Security and Privacy Preservation in Vehicular Social Networks

by

Rongxing Lu

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2012

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Improving road safety and traffic efficiency has been a long-term endeavor for the government, automobile industry and academia. Recently, the U.S. Federal Communication Commission (FCC) has allocated a 75 MHz spectrum at 5.9 GHz for vehicular communications, opening a new door to combat the road fatalities by letting vehicles communicate to each other on the roads. Those communicating vehicles form a huge Ad Hoc Network, namely Vehicular Ad Hoc Network (VANET). In VANETs, a variety of applications ranging from the safety related (e.g. emergence report, collision warning) to the non-safety related (e.g., delay tolerant network, infortainment sharing) are enabled by vehicle-to-vehicle (V-2-V) and vehicle-to-roadside (V-2-I) communications. However, the flourish of VANETs still hinges on fully understanding and managing the challenging issues over which the public show concern, particularly, security and privacy preservation issues. If the traffic related messages are not authenticated and integrity-protected in VANETs, a single bogus and/or malicious message can potentially incur a terrible traffic accident. In addition, considering VANET is usually implemented in civilian scenarios where locations of vehicles are closely related to drivers, VANET cannot be widely accepted by the public if VANET discloses the privacy information of the drivers, i.e., identity privacy and location privacy. Therefore, security and privacy preservation must be well addressed prior to its wide acceptance. Over the past years, much research has been done on considering VANET's unique characteristics and addressed some security and privacy issues in VANETs; however, little of it has taken the social characteristics of VANET into consideration. In VANETs, vehicles are usually driven in a city environment, and thus we can envision that the mobility of vehicles directly reflects drivers' social preferences and daily tasks, for example, the places where they usually go for shopping or work. Due to these human factors in VANETs, not only the safety related applications but also the non-safety related applications will have some social characteristics.

In this thesis, we emphasize VANET's social characteristics and introduce the concept of vehicular social network (VSN), where both the safety and non-safety related applications in VANETs are influenced by human factors including human mobility, human self-interest status, and human preferences. In particular, we carry on research on vehicular delay tolerant networks and infortainment sharing — two important non-safety related applications of VSN, and address the challenging security and privacy issues related to them. The main contributions are, i) taking the human mobility into consideration, we first propose a novel social based privacy-preserving packet forwarding protocol, called SPRING, for vehicular delay tolerant network, which is characterized by deploying roadside units (RSUs) at high social intersections to assist in packet forwarding. With the help

of high-social RSUs, the probability of packet drop is dramatically reduced and as a result high reliability of packet forwarding in vehicular delay tolerant network can be achieved. In addition, the SPRING protocol also achieves conditional privacy preservation and resist most attacks facing vehicular delay tolerant network, such as packet analysis attack, packet tracing attack, and black (grey) hole attacks. Furthermore, based on the "Sacrificing the Plum Tree for the Peach Tree" — one of the Thirty-Six Strategies of Ancient China, we also propose a socialspot-based packet forwarding (SPF) protocol for protecting receiver-location privacy, and present an effective pseudonyms changing at social spots strategy, called PCS, to facilitate vehicles to achieve high-level location privacy in vehicular social network; ii) to protect the human factor — interest preference privacy in vehicular social networks, we propose an efficient privacy-preserving protocol, called FLIP, for vehicles to find like-mined ones on the road, which allows two vehicles sharing the common interest to identify each other and establish a shared session key, and at the same time, protects their interest privacy (IP) from other vehicles who do not share the same interest on the road. To generalize the FLIP protocol, we also propose a lightweight privacy-preserving scalar product computation (PPSPC) protocol, which, compared with the previously reported PPSPC protocols, is more efficient in terms of computation and communication overheads; and iii) to deal with the human factor – self-interest issue in vehicular delay tolerant network, we propose a practical incentive protocol, called Pi, to stimulate self-interest vehicles to cooperate in forwarding bundle packets. Through the adoption of the proper incentive policies, the proposed Pi protocol can not only improve the whole vehicle delay tolerant network's performance in terms of high delivery ratio and low average delay, but also achieve the fairness among vehicles.

The research results of the thesis should be useful to the implementation of secure and privacy-preserving vehicular social networks.

Acknowledgements

I would like to thank all the people who made this possible. This thesis would not have been possible without the help and support of my supervisor, my thesis committee members, and my colleagues in the Broadband Communications Research (BBCR) group. During my PhD research I learned many new things, and without the people surrounding me I could not enjoy from this period of my life.

First of all, I gratefully acknowledge my supervisor, Professor Xuemin (Sherman) Shen. He made available his support and aid in a number of ways. He always does care about his students, and I had this opportunity to discuss the obstacles encountered me in my study and research openly with him. He not only helps me to develop the academic skills, but also guides me to strive for excellence. I would also like to thank Prof. Nirwan Ansari for serving as my thesis external examiner and sharing his invaluable insight on computer and communication security with me. I would also like to extend my appreciation to the other members of my examining committee, Prof. Guang Gong, Prof. M. Anwar Hasan, and Prof. Liping Fu, for the time and efforts to read my thesis. In spite of their busy schedules, all have been readily available for advice, reading and encouragement.

I am indebted to my colleagues for supporting me at the BBCR group. Among them, Dr. Xiaodong Lin, Dr. Haojin Zhu, Dr. Minghui Shi, Dr. Chenxi Zhang, Dr. Yanfei Fan, Dr. Xu Li, Dr. Ho-Ting Cheng, Dr. Jiming Chen, Dr. Zhiguo Shi, Mrs. Xiaoting Sun, Mr. Xiaohui Liang, Mr. Tom. H. Luan, Mr. Kuan Zhang, Mr. Bin Cao, Ms. Miao Wang, and Mr. Mrinmoy Barua are notable. We worked collaboratively at the BBCR group, and we had many discussions to brainstorm and review our work, especially for the case studies.

There are many other people whose names are not mentioned here. It does not mean that I have forgotten them or their help. It is a privilege for me to work and share life with so many bright and energetic people. Their talent and friendship have made Waterloo such a great place to live.

Grateful acknowledgements are made for financial support from the Ontario Graduate Scholarship (OGS), Ontario Research & Development Challenge Fund Bell Scholarship, and Research In Motion (RIM) Graduate Scholarship. Thanks to the University of Waterloo for numerous assistantship awards.

I would never get this far without the support of my parents. I think them for always believing in me and supporting me. Their love and encouragement have been and will always be a great source of inspiration in my life.

Finally, my special thanks go to my wife, Ellen Zhang, for the loving support and patience she has for me to fulfil my career goals.

Dedication

To my family and teachers from whom I have learned so much

Table of Contents

List of Tables

List of Figures

List of Abbreviations

VANET	Vehicular Ad Hoc Network
VSN	Vehicular Social Network
MANET	Mobile Ad Hoc Network
OBU	On-board Unit
RSU	Roadside Unit
PKI	Public Key Infrastructure
V-2-V	Vehicle-to-Vehicle Communication
V-2-I	Vehicle-to-Infrastructure Communication
IVC	Intervehicle Communication
DSRC	Dedicated Short Range Communications
WAVE	Standards for Wireless Access in Vehicular Environments
FCC	Federal Communications Commission
TA	Trust Authority
CA	Certificate Authority
DTN	Delay Tolerant Network
CPPA	Conditional Privacy-preserving Authentication Technique
SPRING	Social-based Privacy-preserving Packet Forwarding Protocol
SPF	Socialspot-based Packet Forward Protocol
PCS	Pseudonyms Changing at Socialspots Strategy
FLIP	Finding Like-Minded Vehicle Protocol
PPSPC	Privacy-preserving Scalar Product Computation
Pi	Practical Incentive

Chapter 1

Introduction

With the advancement and pervasiveness of wireless communication technologies, it is envisioned that vehicles can communicate with each other as well as with the roadside infrastructure located at some critical sections of the road. With the wireless communication devices equipped in vehicles (also known as On-Board Units (OBUs)) and the Roadside Units (RSUs), a self-organized network can be formed, which is called a Vehicular Ad Hoc Network (VANET). Due to various envisioned vehicle safety application scenarios and emerging service demands, VANETs have attracted extensive attentions from all aspects of governments, car manufacture industry, and research community [1, 2, 3, 4, 5]. Extensive lists of potential applications are compiled and assessed by the various projects and consortia. Typically, applications are categorized as safety related and non-safety related [6]. Examples for each category are: 1) Cooperative forward collision warning, namely, to avoid rear-end collisions; traffic light optimal speed advisory, namely, to assist the driver to arrive during a green phase; 2) Mobile peer-to-peer network, namely, to share interesting multimedia files; or other value-added services.

However, before the VANET applications can be put into commercial product lines and start to serve our lives, security and privacy preservation issues must be well tackled [7, 8, 9]. This is not only because vehicles/drivers may not be willing to expose their identities and their corresponding location information to the public, but also because that any misbehavior of a vehicle in a VANET, such as launching a single bogus and malicious traffic related message, could potentially lead to a serious traffic accident. In order to protect the integrity and authenticity of the message, message authentication code (MAC) and digital signature technologies, such as RAISE [10], have provided efficient methods, which prevent an adversary from tampering messages in the middle. On the other hand, in order to provide conditional privacy preservation, some promising techniques, such as

GSIS, ECPP and others [11, 12, 13, 14], are suggested. Although these previously reported schemes have utilized some unique characteristics of VANETs, another special feature of VANETs — social characteristic, has not been well exploited, which may make previously reported schemes unsuitable for some practical VANET applications. In this thesis, we propose the concept of vehicular social network (VSN) in which the social characteristics of VANETs are considered, and develop a suite of security and privacy-preserving solutions to adapt to this promising network's needs.

1.1 Vehicular Ad Hoc Networks

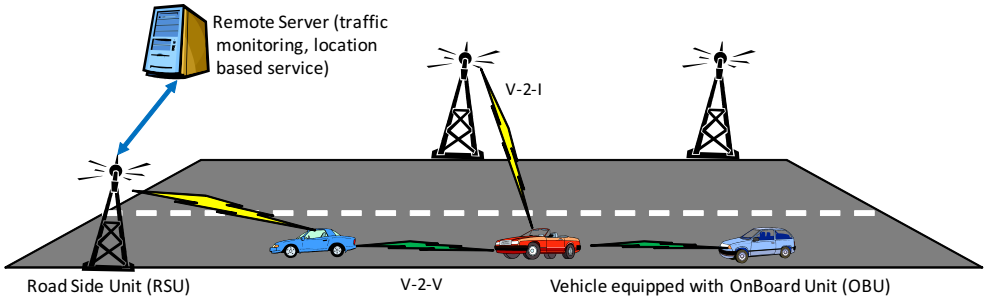


Figure 1.1: Vehicular Ad hoc Network (VANET)

The advancement and wide deployment of wireless communication technologies have revolutionized our lifestyles by providing the best ever convenience and flexibility in accessing Internet services and various types of personal communication applications. Recently, car manufactories and telecommunication industries have geared up to equip every car with the technology that allows drivers and passengers from different cars to communicate with each other in order to improve the driving experience. By using those communication devices equipped in vehicles, also known as onboard units (OBUs), the vehicles can communicate with each other, as well as with roadside units (RSUs) located at the critical points on the road, such as a traffic light at a road intersection. With the OBUs and the RSUs, a self-organized network can be formed, which is called a vehicular ad hoc network (VANET), as shown in Fig. 1.1. Due to low cost and easy deployment of wireless technology in the near future, the roadside will be expected to be densely covered with a variety of RSUs, like traffic lights, traffic signs, and wireless routers, which will provide wireless access to vehicles on the road. In addition, the RSUs could be connected to the Internet backbone to support diversified services, such as transmission control protocols and

real-time multimedia streaming applications [15]. Thus, an increasing interest has been enhanced by both industry and academia in the applications of roadside-to-vehicle communication and intervehicle communication (IVC), aiming to improve the driving safety and traffic management and also provide drivers and passengers with Internet access.

1.1.1 Characteristics of VANETs

Vehicular Ad Hoc Networks (VANETs) is a special case of mobile ad hoc networks (MANETs) [16, 17], where the mobile nodes are instantiated with vehicles equipped with OBU communication devices, as shown in Fig. 1.2. Therefore, VANETs have some unique characteristics different from MANETs [7, 18].

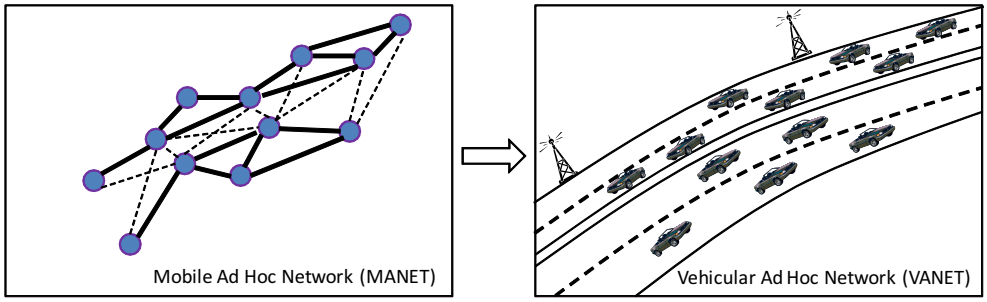


Figure 1.2: The relation between VANET and MANET

1. *Rapid change in topology:* Since vehicles are moving at high speeds, the topology of VANETs is prone to frequent and rapid changes, and usually follows the freeway, Manhattan, streets in real world.

2. *No power constraint:* Since the batteris of the vehicles are self-charging, vehicles in VANET do not have the conventional power constraints of the hand-held devices in MANETs.

3. *Large scale:* VANETs constitute the largest instance of MANETs that the world have ever seen, where the order of the number of vehicles is around 10^7 in reality.

4. *Variable network density:* The number of vehicles in one area of the road is temporally changing during the day, e.g., roads in the rush hours are busier than other times of the day.

5. *High predictable mobility:* The velocity of vehicles in cities ranges from 0 to 60 km/h, and the average velocity can reach up to 100 km/h on a highway. Therefore, the road geometric topology regulates the mobility of vehicles.

In summary, we use Table 1.1 to give a brief comparison between VANET and MANET in terms of topology, architecture, connectivity, resource, scalability and application.

Table 1.1: Comparisons between VANET and MANET

	VANET	MANET
Topology	Freeway, Manhattan, Streets in real world	Random waypoint
Architecture	Vehicle-to-vehicle, Vehicle-to-RSU	Node-to-node
Connectivity	Random and Intermittent	Random
Resource	Almost unlimited	Limited hardware, Power limited by battery
Scalability	Huge	50-100 nodes
Application	Safety, traffic, payment, electronic toll collection (daily life)	Military, disaster (specific)

1.1.2 Applications of VANETs

VANET applications can be divided into two major categories. Applications that increase vehicle safety on the roads are called safety related applications. Applications that provide value-added services, for example, entertainment, vehicular delay tolerant networks (DTNs), are called non-safety related applications [19, 3, 15].

Safety related applications

Safety related applications can decrease the number of road accidents significantly . According to some studies [20], 60 percent of accidents could be avoided if a driver were provided with a warning half a second before the moment of collision. There are three major scenarios in which safety applications could be very useful [3].

1. *Accidents:* When vehicles travel at a high speed on major roads,drivers have very little time to react when a vehicle in front of them make an unexpected move. If an accident occurs, the approaching vehicles often crash before they can a stop. Safety related applications of VANET could be used to warn cars of an accident

that occurred further along the road, thus preventing a pile-up from occurring. In addition, a safety related application could also be used to provide drivers with early warnings and prevent an accident from happening in the first place.

2. *Intersections:* In reality, driving near and through intersections is one of the most complex challenges that drivers face because when two or more traffic flows intersect, and the possibility of collisions is high. In 2003, according to the U.S. Department of Transportation (DoT), intersection crashes accounted for more than 45 percent of all reported crashes and 21 percent of fatalities, that is, 9213 fatalities occurred at intersections in the United States [21]. The number of accidents would be reduced if a safety related application of VANET warned the driver of an impending collision. The driver then could take action to avoid it (see the European project PReVENT/Intersafe [22] for more details).

3. *Road Congestion:* Safety related applications also could be used to provide drivers with the best routes to their destinations [23]. This would decrease congestion on the road and maintain a smooth flow of traffic, thus increasing the capacity of the roads and preventing traffic jams. It also could have the indirect effect of reducing traffic accidents because drivers would feel less frustrated and more inclined to follow traffic regulations.

Non-safety related applications

Non-safety related applications can provide road users with information, advertisements, and entertainment during their journey. Three basic non-safety related applications are described as follows [3].

1. *Internet Connectivity:* Constant Internet access has become a daily requirement for many of us, and since many user applications also require Internet connectivity, providing this facility to vehicle occupants and other VANET applications are important. Moreover, this means that the usual business framework will be presented seamlessly in vehicles, without a requirement for specific redevelopment.

2. *Peer-to-Peer Applications:* To alleviate boredom, peer-to-peer applications are also interesting ideas for VANETs. Passengers in the vehicles could share music, movies, and so on, and chat with each other and play games. They could also stream music or movies from special servers during long journeys.

3. *Vehicular DTNs:* In vehicular DTNs, each DTN node is instantiated by vehicle driven by people running in a city environment to cooperatively relay messages in an opportunistic fashion [24].

1.1.3 Security Threats

The adoption of a variant of the widely deployed IEEE 802.11 protocol by the vehicle manufacturers makes the attacker's task easier in VANETs, and the VANET could be a double-edged sword if the security concern is not well addressed. In general, the security threats in VANETs can be classified into the following categories [19, 7, 15, 1, 25, 2, 26].

1. *Jamming:* An attacker deliberately generates a huge of bogus messages to jam the communication channel so that preventing other vehicles form normal communications.

2. *Forgery:* For malevolent goals, an attacker can launch forgery attack, which could potentially cause the accidents. Therefore, the freshness and correctness of the transmitted messages in V-2-V communication are very important to ensure that the received messages are not forged.

3. *In-transit traffic tampering:* In this attack, an attacker deliberately delays, drops, corrupts, or modifies messages to damage the normal V-2-V communications.

4. *Impersonation:* In this attack, an attacker aims to let others convince he/she is a legitimate vehicle for his/her own benefits. For example, an attacker can claim that he/she is an emergence vehicle to make other vehicles yield the road in front of him/her.

5. *Privacy violation:* In VANETs, the collection of vehicle-specific information from overheard vehicular communications will become particularly easy. Then, if an attacker can collect enough messages from vehicles, the inferences on the drivers personal data could be made, and thus violate drivers' privacy.

6. *On-board tampering:* Beyond abuse of the communication protocols, an attacker may select to tinker with data (e.g., velocity, location, status of vehicle parts) at their source, tampering with the on-board sensing and other hardware. For example, an attacker can by-pass a sensor or put some ice around the sensor that senses temperature to make the vehicle to send bogus warning of icy road.

1.1.4 Security Requirements

In order to protect VANETs against the threats mentioned above, the security mechanisms employed in VANETs should satisfy the following security requirements [19, 27, 6, 28, 2, 29, 30, 31]:

1. *Authentication:* Authentication is the ability to ascertain that a thing is indeed the one that it claims to be. Message authentication is of vital importance in VANETs because it ensures that a received message is indeed sent from a legitimate and authorized vehicle in the networks.

2. *Integrity:* Integrity is the ability to assure that messages exchanged between vehicles have not been subject to modifications, additions, or deletions. Integrity ensures that all messages sent by vehicles should be delivered unaltered.

3. *Non-repudiation:* Non-repudiation is the ability to prevent an authorized vehicle from denying the existence or contents of the message sent by itself. Non repudiation is a critical property for VANETs because it can prevent an attacker from denying the attacks that he/she have launched.

4. *Access control:* Access control is necessary to ensure reliable and secure operations of a system. In VANETs, any misbehaving entity should be revoked from the network to protect the safety of other legitimate entities in the network. Moreover, any actions taken by that misbehaving entity should be canceled.

5. *Privacy:* Privacy is the ability to protect private information from an unauthorized party. In VANETs, the real identity of any individual vehicle is only blind to other vehicles and road side units, but should be transparent to a trusted authority (TA). This security requirement is also called "conditional privacy preservation". Currently, IEEE 802.11p [32] and IEEE 1609.x are called wireless access in vehicular environments (WAVE) standards [33, 34] since their goal, as a whole, is to facilitate the provision of wireless access in vehicular environments.

As shown in Table 1.2, WAVE [33, 34] develops a family of standards for vehicular communications, including resource manager, service services, networking services, multi-channel operation, architecture, communication manager, and facilities, where the IEEE 1609.2 standard mainly addresses the security issues in message processing in VANET. However, as we know, the current version of IEEE 1609.2 standard has not provided any mechanism to achieve privacy preservation in VANETs. Therefore, anonymous and authenticated communication is still a challenging issue in VANETs.

Table 1.2: Standard for Wireless Access in Vehicular Environments (WAVE)

Standard document	Protocols	
IEEE 802.11p	WAVE PHY and MAC	Underdevelopment [32]
IEEE 1609.1	Resource Manager	Trial-use standard Approved 15/09/06
IEEE 1609.2	Security Services	Trial-use standard Approved 8/06/06
IEEE 1609.3	Network Services	Trial-use standard Approved 23/03/07
IEEE 1609.4	Multi-Channel Operation	Trial-use standard Approved 30/10/06
IEEE 1609.0	Architecture	Underdevelopment
IEEE 1609.5	Communication Manager	Underdevelopment
IEEE 1609.6	Facilities	Underdevelopment

1.1.5 Related Work on Securing VANETs

In the following, we review some previously reported works concerning the security and privacy preservation issues in VANETs [35, 36, 1, 37, 7, 38, 12, 39, 40, 14]. Security and privacy issues on VANETs have attracted extensive attentions from both academia and industry since 2004. J. Hubaux *et al.* [36, 1] first identify security and privacy issues of VANETs by claiming that an appropriate public key infrastructure must be well devised to protect the transmitted information and to do mutual authentication among network entities. To address the privacy issue, they suggested to rely on temporary pseudonyms to achieve anonymity.

The IEEE 1609.2 standard [35] is an IEEE trial-use standard for wireless access in vehicular environments particularly for security services. It can be used to protect messages from attacks such as spoofing, eavesdropping and alteration. Also, with this standard, vehicles can send encrypted messages to each other or to roadside infrastructures. Nevertheless, as presented in the last section, the communication overhead caused by the security mechanism is large. Although this standard mentions the necessity to respect the drivers' privacy, such as not leaking personal, identifying, or linkable information to unauthorized parties, it does not provide detailed approaches to achieve this privacy requirement.

To achieve both message authentication and anonymity, Raya *et al.* [37, 7] propose that each vehicle should be pre-loaded with a large number of anonymous public and private key pairs together with the corresponding public key certificates. All traffic related messages are signed with a public key based scheme. To achieve privacy, each public and private key pair is used in a short life time and a pseudo ID is used in each public key certificate. Moreover, the authors computed a safe time interval, in which each vehicle should change its pseudo ID at least once so that two consecutive pseudo IDs of the vehicle cannot be linked by an adversary. Clearly, this scheme to protect privacy is straightforward and

efficient. However, it requires a large storage capacity to store these security information in each vehicle. Further more, on the side of the trust authority (TA), it should keep the record of all pseudo IDs and their corresponding key pairs of all vehicles. This is not only inconvenient for the TA to find the real identity of an abusing vehicle, but is also inconvenient to manage these pseudo identities.

In order to overcome the weakness of the above issues, Lin *et al.* [12] developed a group signature based scheme. With this scheme, vehicles do not require any identities at all. All vehicles within the same group share the same public key, while their private keys are different. When a vehicle receives a signed message, the vehicle verifies it with the group public key. The verifier only knows whether the signer is a legitimate group member or not, but the verifier does not know who the signer exactly is. In this way, the identity privacy is well protected. In case that a dispute happens, TA working as the group manager is capable to trace the real identity of the sender by using TA's secret key. For example, a group member (as an attacker) broadcasts a malicious bogus message with his/her group private key. Notice that the signature of the message is valid, but the content is fake. Suppose a neighbor of the attacker finds out that the message is bogus. The neighbor reports the message along with the corresponding message signature to TA. The TA using its private key can compute the private key of the attacker from the signed signature. Then, by looking up the table in which a private key maps to a real world identity of a user, eventually TA can trace the real identity of the attacker. The disadvantage of the group signature based scheme lies on its inefficiency of revocation. If TA revokes a private key of a vehicle, TA has to update the entire security keys of the whole group. The key materials of all group members have to be renewed. In addition, although the group signature based scheme achieves the conditional privacy preservation, the computational cost of verifying a group signature is high, as compared with the traditional PKI based signature scheme. Therefore, the group signature based scheme could result in a high message loss ratio under a high traffic density scenario.

To alleviate the high computational overhead of the group signature based security scheme, Calandriello *et al.* [39] introduce a hybrid scheme that integrates the traditional PKI based scheme and the group signature based scheme. Similar to the above group signature based scheme, each vehicle is assigned a private key and a group public key. The group public is the same for all group members, and each member holds a distinct private key. Unlike the above scheme of Lin *et al.* [12], the private key is not used for signing messages. Instead, a signer uses the private key to generate temporary public key certificates. In particular, vehicles generate multiple private and public key pairs. Each pair has a public key certificate, which mainly contains a pseudo identity and a lifetime, as well as a signature. The signature in the certificate is signed using its group private key by each

vehicle itself instead of the trust authority. Here, the lifetime in the certificate indicates how long the certificate is valid. The lifetime should be short such that an adversary cannot link two distinct pseudo identities. The temporary public and private key pairs work the same as in [7]. They are used to sign traffic related messages. The revocation process is similar to the group signature scheme in [12]. In case that a dispute happens, TA can trace the real identity of the attacker through the public key certificate because the certificate is signed by vehicles using their group private key. The hybrid scheme makes a tradeoff between the traditional PKI based scheme and the group signature based scheme. Although the hybrid scheme has less computational overhead than the group signature based scheme, it still has higher computation overhead than the traditional PKI based scheme. Therefore, this hybrid scheme cannot address the scalability issues.

The presence of roadside infrastructure units (RSUs) is one of the unique characteristics of VANETs. Some related works take advantage of this feature to achieve privacy requirements. 1) J. Freudiger *et al.* [40] introduce a mix-zone scheme to protect the location privacy for vehicles. An RSU manages a mix-zone, in which vehicles change their pseudo IDs and corresponding public keys. An adversary cannot link two pseudo IDs from the same vehicle when the vehicle passes through a mix-zone. In the mix-zone scheme, RSUs are located at intersections. Vehicles that go through an intersection process mutual authentication with the RSU, and then obtain a secret key from the RSU. All legitimate vehicles share the same secret key. When vehicles within an intersection send safety messages, vehicles first sign them with their temporary public keys and then encrypt the whole message with the secret keys. An adversary without the secret key cannot see the content of the message including the used public certificates, and thus the adversary cannot link two pseudo IDs used before and after a vehicle going through a mix-zone. However, the secret key cannot stop a legitimate vehicle from linking such two IDs because any legitimate vehicle has the same secret key. Therefore, the mix-zone scheme cannot thwart an internal attack. 2) Lu *et al.* [14] develope a conditional privacy preservation scheme, called ECPP, which divides privacy requirements into three levels. The first level is anticipated by TA. TA is capable to trace the real identity of vehicles from a signed safety message. From the users' perspective, no privacy is defined in the first level. In the second level, each safety message is anonymously authenticated, but an adversary can trace a vehicle by collecting messages. The third level is the strongest privacy level. An individual cannot be traced by collecting messages of the vehicles. In ECPP, RSUs play an important role to achieve the three privacy levels. RSUs are responsible to issue a temporary public key certificate, which vehicles use to sign safety messages. The temporary certificate does not reveal the real identity of a vehicle. Vehicles determine the lifetime of a certificate. When vehicle requests a certificate from an RSU, the vehicle indicates the lifetime of the certificate. Then

the RSU issues the requested certificate. Before an issued certificate is expired, a vehicle should request a new certificate from a nearby RSU. The unfixed lifetime is to prevent an adversary from linking multiply certificates of an individual vehicle. RSUs sign each temporary certificate using a group signature scheme, which prevents an adversary from having any knowledge of location information during the use of a certificate. However, TA has the highest authority, which is not only able to recover the real identity of a vehicle from its message signature, but also is able to reveal the real identity of an issuer (RSU) from a temporary public key certificate. The abilities of TA are used in case of a dispute.

Although all the above previously reported works considered VANET's unique characteristics and addressed some security and privacy issues in VANETs, as we know, few of them takes the social characteristics of VANET into consideration. Therefore, in this thesis, we will emphasize VANET's social characteristics and present the concept of vehicular social network (VSN). In specific, we will address the following issues: what is the social characteristic of VANET? what benefits can it bring in VANETs? and what are the more challenging security and privacy issues lying in it?

1.2 Vehicular Social Networks: Research Motivations and Objectives

1.2.1 Motivations

Today, everyone on the Internet knows the buzzword *social networking*. Sites such as Facebook, LinkedIn, as well as content-sharing sites that also offer social networking functionality such as YouTube, have captured the attention of millions of users and made millions of dollars from venture capitalists [41, 42]. *Social networking* over Internet offers us many interesting functionalities including network of friends lists, person surfing, private messaging, discussion forums or communities, and media uploading. In a word, *social networking* can help us work together over common activities or interests. Although *social networking* is a buzzword popular in Internet, social networks exist everywhere around us — at workplace as well as within families and social groups, e.g., pocket switched network [43], where users exchange data related games, rumors, and interesting information using their mobile devices with short range wireless interfaces.

In VANETs, vehicles are usually driven by citizens in a city environment, and thus we can envision that the mobility of vehicles directly reflects people's intentions. Then, since people's intentions (human factors) are involved in VANETs, not only the safety

related applications but also the non-safety related applications will show some social characteristics. As a result, we are motivated to study the social characteristics and their impacts on VANETs.

What Is the Matter of Vehicular Social Network?

Vehicular Social Network (VSN) is a VANET, which includes the traditional V-2-V communications as well as V-2-I communications. Most importantly, it takes the "human factors" into consideration, as shown in Fig. 1.3. To illustrate the concept of VSN clearly, let us consider the following aspects.

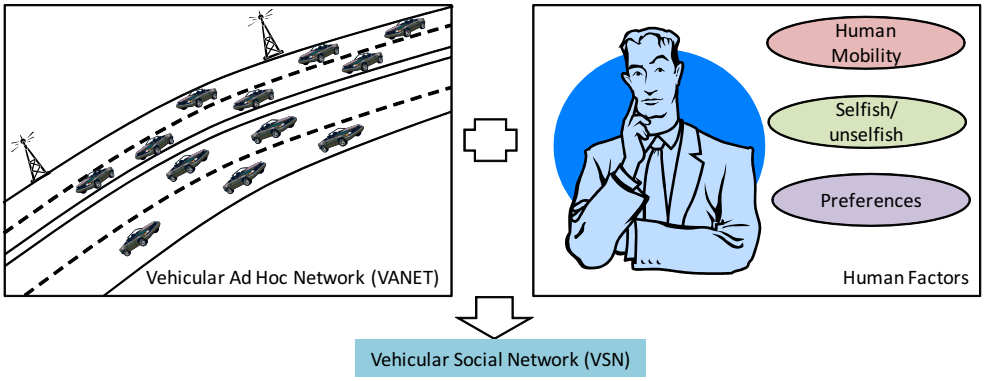


Figure 1.3: Vehicular social network (VSN)

- *Human Mobility Model:* In VSNs, vehicles are driven by people. Then, the mobility model is no longer a random waypoint. Instead, some realistic human mobility models in a city environment may be adopted. In the following, we list some of them [44].

    1. *Shortest Path Map Based Movement:* A vehicle chooses its destination by randomly selecting a map point on the map. Then, it calculates the shortest path to the destination using the Dijkstra's algorithm.
    2. *Community Based Mobility Model:* In the community based mobility model, it is assumed that there exist several spots having high social attractivity in a map, where the social attractivity is calculated from the number of vehicles that are currently stopping in the spot. Then, a vehicle prefers to choose a spot with higher social attractivity as its destination.

3. *Time-variant Mobility Model:* The time-variant mobility model is somewhat similar to the community based model. However, a vehicle prefers to different spots at different times of a day, thereby moving between them in a periodic manner. For example, people goes to the office to work in the morning, to a restaurant to eat at noon and home after working in the evening.

Based on these realistic models, we can develop more efficient non-safety related applications in VSNs.

- *Human Selfish Status:* In VSNs, vehicles are driven and controlled by rational entities — human beings. Since not all people in reality is nonselfish, some vehicles will behave selfishly and may not be willing to participate in some non-safety related applications. For example, in order to conserve buffer and computing resources, a selfish vehicle may be reluctant in the cooperation that is not directly beneficial to it, which could make a well designed routing protocol in VANET useless. Therefore, the selfishness is a very challenging issue for non-safety related applications in VSNs.

- *Human Preferences:* In VSNs, a great number of vehicles driven by people move between home and office for one or more hours each day. Since these commutes typically take people over highways and other frequented corridors, they are highly predicable and regular. Day after day, the same people travel along the same roadways at the same time. Then, based on their preferences, they could form some virtual communities to discuss some interesting topics.

1.2.2 Objectives

Since the social characteristics (human mobility, selfish status, and preferences) are considered in VSNs, the study of VSNs become more promising and richer than that in the pure VANET. For example, many social based VANET routing protocols and non-safety related applications can be developed in VSNs. Nevertheless, like in traditional online social networks, more challenging security and privacy issues are also reared up in VSNs, especially for the privacy.

In VSNs, the human mobility model can be used for designing efficient routing protocols. However, if poorly treated, the mobility model could disclose people's location privacy. On the other hand, when the privacy technique, even the conditional privacy preservation technique, is employed in the VSN, the identification of a self-interest oriented vehicle becomes challenging, for a vehicle's behaviors may be unlinkable. In addition, the favorite of people

is obviously a privacy issue, it is unrealistic for a person to disclose his/her favorites to everyone else on road. Therefore, security requirements including authentication, integrity, non-repudiation, access control and privacy should be paid more attention in VSNs.

In this thesis, we will study safety and non-safety related applications based on human factors (mobility model, self-interest status and preferences) in VSNs, and devote ourselves to resolving the challenging security and privacy issues amongst of them.

1.3 Research Contributions

The research in this thesis focuses on developing a suite of protocols to deal with the challenging security and privacy-preserving issues in vehicular social networks. Specifically, the main contributions lie in the following aspects:

- First, by taking the human factor — *human mobility* into consideration, we propose a novel social based privacy-preserving packet forwarding protocol, called SPRING [24], for vehicular delay tolerant networks, which is characterized by deploying RSUs at high social intersections to assist in packet forwarding between vehicles by temporarily storing packets through V-2-I communications. With the assistances of high-social RSUs, the probability of packet drop is reduced and as a result high reliability of packet forwarding in vehicular DTNs can be achieved. In addition, the SPRING protocol also achieves conditional privacy preservation and resist most attacks facing vehicular DTNs, such as packet analysis attack, packet tracing attack, and black (grey) hole attacks. Further, based on the "*Sacrificing the Plum Tree for the Peach Tree*" — one of the Thirty-Six Strategies of Ancient China, we also propose a socialspot-based packet forwarding (SPF) protocol for protecting receiver-location privacy in VANETs [45]. In addition, an effective pseudonyms changing at social spots strategy, called PCS [46], is proposed, which facilitates vehicles to achieve high-level location privacy in VANET.

- Second, to protect the human factor — *interest preference* privacy in vehicular social networks, an efficient privacy-preserving protocol for finding like-minded vehicles on the road, called FLIP [47], which allows two vehicles with the common interest to identify each other and establish a shared session key, and at the same time, protects their Interest-Privacy (IP) from other vehicles who do not have the same interest on the road. To generalize the FLIP protocol, a lightweight privacy-preserving scalar

product computation (PPSPC) protocol is also proposed, which, compared with previous PPSPC protocols, is much efficient in terms of computation and communication overheads.

- Third, to deal with the human factor – *self-interest* issue in vehicular delay tolerant networks, a practical incentive protocol, called Pi [48], is proposed to stimulate selfish nodes to cooperate in forwarding bundle packets. Through adoption of the proper incentive policies, the proposed Pi protocol can not only improve the whole vehicle DTN network's performance in terms of high delivery ratio and low average delay but also achieve the fairness among nodes.

1.4 Outline of the Thesis

The organization of the remainder of the thesis is as follows. Chapter 2 recalls some related work, including basic concepts in social theory, bilinear groups and conditional privacy-preserving authentication for VANET. Chapter 3 presents a novel social based privacy-preserving packet forwarding (SPRING) protocol for vehicular delay tolerant networks. Chapter 4 presents a socialspot-based packet forwarding (SPF) protocol for protecting receiver-location privacy in VANETs. Chapter 5 presents an effective pseudonyms changing at social spots (PCS) strategy for facilitating vehicles to achieve high-level location privacy in VANET. Chapter 6 presents an efficient privacy-preserving protocol (FLIP) for finding like-minded vehicle on the road and its generalization — a lightweight privacy-preserving scalar product computation (PPSPC) protocol. Chapter 7 presents a practical incentive (Pi) protocol to stimulate selfish nodes to cooperate in forwarding bundle packets in vehicular DTNs. Finally, conclusions and future research work are described in Chapter 8.

Chapter 2

Related Work: Social Theory and Cryptography

In this chapter, we discuss some related work, including basic concepts in social theory, bilinear groups, and conditional privacy-preserving authentication (CPPA) technique, which serve as the basis of the schemes/protocols presented in the remaining of this thesis.

2.1 Basic Concepts in Social Theory

Since our research — vehicular social network (VSN) uses some social properties to study the vehicular communications, we first review some basic concepts in social theory [49, 50].

- *Propinquity*: In social theory, propinquity is defined that, under equal conditions, if two nodes are geographically near to each other, they are more likely to be connected.
- *Homophyly*: Homophyly is defined as having one or more common social attributes, like the same organization, favorites. The greater the homophyly, the more likely two nodes will be connected. At the individual level, persons are more likely to have a connection, friendship or association, if they have common attributes.
- *Degree Centrality:* In social networks, a central node is one that relates to a large quantity of other nodes in the network. Therefore, the centrality of a node can be measured as the number of out-links connecting a node to its neighbors or as the number of in-links that a certain node is receiving from adjacent nodes. We can

measure a node's centrality by counting the number of links (in-links or out-links) that the node possesses. By definition, the degree of a node n_i is the number of links incident to it and is represented as $d(n_i)$. When there is no distinction between in-degree and out-degree, the centrality of a node n_i can be denoted by

$$C_D(n_i) = d(n_i) = \sum_{\forall j \neq i} x_{ij} \tag{2.1}$$

where $C_D(n_i)$ is the degree centrality of node i; $d(n_i)$ is the degree of node i; x_{ij} is 1 if node i is indicent to node j; and 0 otherwise. Clearly, this magnitude depends on the size of the network and it becomes complex to use once it is compared with other networks. Therefore, a way to standardize degree centrality is to divide Eq. (2.1) by the maximum number of nodes that a node can be connected to. Since the number of nodes in a network is n, then a node can be connected at maximum to all other nodes but itself

$$C'_D(n_i) = \frac{\sum_{\forall j \neq i} x_{ij}}{n-1} \tag{2.2}$$

- *Closeness Centrality*: Degree centrality just shows how many nodes are directly joined to a central node, nevertheless it does not consider indirect ties by which a node can reach others using paths available in the network. A node that is central-close can reach other nodes through short distance paths. Therefore, the notion of closeness-centrality is related to the inverse of distance between nodes (e.g., the higher the distance, the less the central-close). In social networks, a shortest path between two nodes is defined as a geodesic. As a result, a closeness index has to account for the geodesics that a given node has to all other nodes in the network. Assuming the network is fully connected, the central-closeness of a node n_i can be measured by i) adding up all individual geodesics related to n_i; and ii) calculating the inverse of this value. Specifically, the standardized formula of a node n_i's closeness centrality can be defined

$$C'_C(n_i) = \frac{n-1}{\left(\sum_{j=1, i \neq j} d(n_i, n_j)\right)} \tag{2.3}$$

where $C'_c(n_i)$ is the standardized closeness centrality of node n_i; and $d(n_i, n_j)$ is the Geodesic between node n_i and node n_j [49].

- *Betweenness Centrality*: Betweenness, which is focusing on nodes that lie in the path between other nodes, is another one to measure the centrality. To calculate an index

that measures betweenness centrality, several considerations should be taken into account [49]. First, it is assumed that a node which wants to reach another node uses preferably shortest paths available between them; second, if two or more geodesics are available, then the node can choose between them with equal probability; third, if A communicates with B, then B must communicate with A; finally, adding up the proportion that a node is between others gives the standardized betweenness centrality index as follows

$$C'_B(n_i) = \frac{\sum_{j<k, i\neq j, i\neq k} \frac{g_{jk}(n_i)}{g_{ik}}}{\frac{(n-2)(n-1)}{2}} \tag{2.4}$$

where $C'_B(n_i)$ is the standardized betweenness centrality of node i; $g_{jk}(n_i)$ is the number of geodesics linking j and k that contains i in between; and g_{jk} is the total number of geodesics linking j and k.

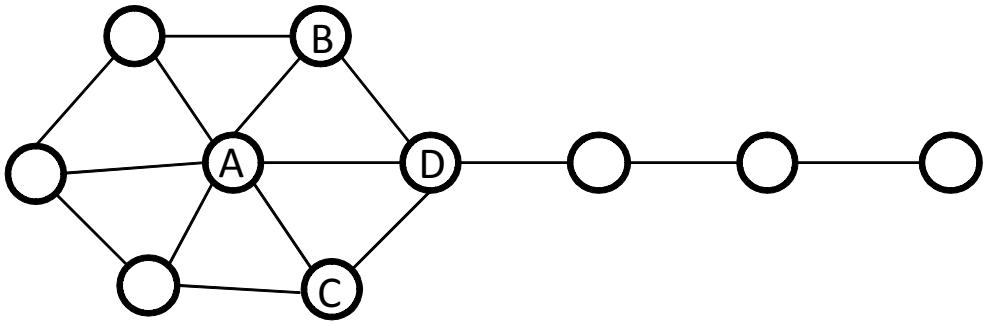


Figure 2.1: The degree, closeness, and betweenness centralities in social networks

Fig. 2.1 shows the differences among the degree, closeness, and betweenness centralities in social networks. According to the degree centrality $C'_D(n_i) = \frac{\sum_{\forall j\neq i} x_{ij}}{n-1}$, node A has the highest degree centrality, i.e., it has the most direct connections, serving as "Central connector" or"hub" in the network; while from the closeness centrality $C'_C(n_i) = \frac{n-1}{\left(\sum_{j=1, i\neq j} d(n_i, n_j)\right)}$, nodes B, C have the highest closeness centrality, i.e., they have the shortest paths to all others; and from the betweenness centrality $C'_B(n_i) = \frac{\sum_{j<k, i\neq j, i\neq k} \frac{g_{jk}(n_i)}{g_{ik}}}{\frac{(n-2)(n-1)}{2}}$, node D has the highest betweenness centrality, i.e., it plays a "broker" role in the network, having great influence over what flows in the network. In this thesis, we would utilize these social properties in vehicular social networks to improve the efficiency, security and privacy.

2.2 Bilinear Groups

2.2.1 Notations

Let $\mathbb{N} = \{1, 2, 3, \ldots\}$ be the set of positive integers. If x is a string, then $|x|$ denotes its length, while if $\mathbb{S}$ is a set then $|\mathbb{S}|$ denotes its size. If $k \in \mathbb{N}$ then 1^k denotes the string with k ones. If $\mathbb{S}$ is a set then $s \leftarrow \mathbb{S}$ denotes the operation of picking a random element s of $\mathbb{S}$ uniformly.

2.2.2 Bilinear Groups of Prime Order

Bilinear pairing is an important cryptographic primitive and has been widely adopted in many positive applications in cryptography [51, 52]. Let $\mathbb{G}$ be a cyclic additive group and $\mathbb{G}_T$ be a cyclic multiplicative group of the same prime order q. We assume that the discrete logarithm problems in both $\mathbb{G}$ and $\mathbb{G}_T$ are hard. A bilinear pairing is a mapping $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ which satisfies the following properties:

1. Bilinearity: For any $P, Q \in \mathbb{G}$ and $a, b \in \mathbb{Z}_q^*$, we have $e(aP, bQ) = e(P, Q)^{ab}$.
2. Non-degeneracy: There exists $P \in \mathbb{G}$ and $Q \in \mathbb{G}$ such that $e(P, Q) \neq 1_{\mathbb{G}_T}$.
3. Computability: There exists an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in \mathbb{G}$.

From Reference [51], we note that such a bilinear pairing may be realized using the modified Weil pairing associated with supersingular elliptic curve.

Definition 1 *(Bilinear Generator) A bilinear parameter generator $\mathcal{G}en$ is a probability algorithm that takes a security parameter κ as input and outputs a 5-tuple $(q, P, \mathbb{G}, \mathbb{G}_T, e)$, where q is a κ-bit prime number, $(\mathbb{G}, +)$ and $(\mathbb{G}_T, \times)$ are two groups with the same order q, $P \in \mathbb{G}$ is a generator, and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is an admissible bilinear map.*

In the following, we define the quantitative notion of the complexity assumptions, including Computational Diffie-Hellman (CDH) Problem, Decisional Diffie-Hellman (DDH) Problem, Bilinear Diffie-Hellman (BDH) Problem, Decisional Diffie-Hellman (DBDH) Problem, Strong Diffie-Hellman (SDH) Problem, and Strong Diffie-Hellman-2 (SDH2) Problem.

Definition 2 *(Computational Diffie-Hellman (CDH) Problem) The Computational Diffie-Hellman (CDH) problem in $\mathbb{G}$ is defined as follows: Given P, aP, $bP \in \mathbb{G}$ for unknown $a, b \in \mathbb{Z}_q^*$, compute $abP \in \mathbb{G}$.*

Definition 3 *(CDH Assumption) Let $\mathcal{A}$ be an adversary that takes an input of $(P, aP, bP) \in \mathbb{G}$ for unknown $a, b \in \mathbb{Z}_q^*$, and returns abP. We consider the following random experiment.*

$$\begin{array}{l}\texttt{Experiment } \mathbf{Exp}_{\mathcal{A}}^{\mathrm{CDH}} \\ \quad a, b \xleftarrow{R} \mathbb{Z}_q,\ \alpha \leftarrow \mathcal{A}(P, aP, bP) \\ \quad \texttt{if } \alpha = abP, \texttt{ then } \beta \leftarrow 1, \texttt{ else } \beta \leftarrow 0 \\ \quad \texttt{return } \beta\end{array}$$

We define the corresponding success probability of $\mathcal{A}$ in solving the CDH problem via

$$\mathbf{Succ}_{\mathcal{A}}^{\mathrm{CDH}} = \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{CDH}} = 1\right]$$

Let $\tau \in \mathbb{N}$ and $\epsilon \in [0, 1]$. We say that the CDH is (τ, ϵ)-secure if no polynomial algorithm $\mathcal{A}$ running in time τ has success $\mathbf{Succ}_{\mathcal{A}}^{\mathrm{CDH}} \geq \epsilon$.

Definition 4 *(Decisional Diffie-Hellman (DDH) Problem) For $a, b, c \in \mathbb{Z}_q^*$, given P, aP, bP, $cP \in \mathbb{G}$, decide whether $c = ab \in \mathbb{Z}_q$. The DDH problem is easy in $\mathbb{G}$, since we can compute $e(aP, bP) = e(P, P)^{ab}$ and decide whether $e(P, P)^{ab}$=$e(P, P)^c$ [51].*

Definition 5 *(Bilinear Diffie-Hellman (BDH) Problem) The Bilinear Diffie-Hellman (BDH) problem in $\mathbb{G}$ is as follows: Given P, aP, $bP, cP \in \mathbb{G}$ for unknown $a, b, c \in \mathbb{Z}_q^*$, compute $e(P, P)^{abc} \in \mathbb{G}_T$.*

Definition 6 *(BDH Assumption) Let $\mathcal{A}$ be an adversary that takes an input of $(P, aP, bP, cP) \in \mathbb{G}$ for unknown $a, b, c \in \mathbb{Z}_q^*$, and returns $e(P, P)^{abc}$. We consider the following random experiment.*

$$\begin{array}{l}\texttt{Experiment } \mathbf{Exp}_{\mathcal{A}}^{\mathrm{CDH}} \\ \quad a, b, c \xleftarrow{R} \mathbb{Z}_q,\ \alpha \leftarrow \mathcal{A}(P, aP, bP, cP) \\ \quad \texttt{if } \alpha = e(P, P)^{abc} \texttt{ then } \beta \leftarrow 1 \texttt{ else } \beta \leftarrow 0 \\ \quad \texttt{return } \beta\end{array}$$

We define the corresponding success probability of $\mathcal{A}$ in solving the BDH problem via

$$\mathbf{Succ}_{\mathcal{A}}^{\mathrm{BDH}} = \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{BDH}} = 1\right]$$

Let $\tau \in \mathbb{N}$ and $\epsilon \in [0,1]$. We say that the BDH is (τ, ϵ)-secure if no polynomial algorithm $\mathcal{A}$ running in time τ has success $\mathbf{Succ}_{\mathcal{A}}^{\mathrm{BDH}} \geq \epsilon$.

Definition 7 *(Decisional Diffie-Hellman (DBDH) Problem) The Decisional Bilinear Diffie-Hellman (DBDH) problem in $\mathbb{G}$ is as follows: Given an element P of $\mathbb{G}$, a tuple (aP, bP, cP, T) for unknown $a, b, c \in \mathbb{Z}_q^*$ and $T \in \mathbb{G}_T$, decide whether $T = e(P,P)^{abc}$ or a random element R drawn from $\mathbb{G}_T$.*

Definition 8 *(DBDH Assumption) Let $\mathcal{A}$ be an adversary that takes an input of (aP, bP, cP, T) for unknown $a, b, c \in \mathbb{Z}_q^*$ and $T \in \mathbb{G}_T$, and returns a bit $\beta' \in \{0,1\}$. We consider the following random experiments.*

$\mathtt{Experiment}\ \mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}}$
$a, b, c \xleftarrow{R} \mathbb{Z}_q; R \xleftarrow{R} \mathbb{G}_T$
$\widetilde{\beta} \leftarrow \{0,1\}$
if $\widetilde{\beta} = 0$, *then* $T = e(P,P)^{abc}$; *else if* $\widetilde{\beta} = 1$ *then* $T = R$
$\widetilde{\beta}' \leftarrow \mathcal{A}(aP, bP, cP, T)$
$\mathtt{return}\ 1$ *if* $\widetilde{\beta}' = \widetilde{\beta}$, 0 *otherwise*

We then define the advantage of $\mathcal{A}$ via

$$\mathbf{Adv}_{\mathcal{A}}^{\mathrm{DBDH}} = \Big|\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}} = 1 | \widetilde{\beta} = 0\right] - \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}} = 1 | \widetilde{\beta} = 1\right]\Big| \geq \epsilon$$

Let $\tau \in \mathbb{N}$ and $\epsilon \in [0,1]$. We say that the DBDH is (τ, ϵ)-secure if no adversary $\mathcal{A}$ running in time τ has an advantage $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{DBDH}} \geq \epsilon$.

Definition 9 *(Strong Diffie-Hellman (SDH) Problem) The Strong Diffie-Hellman (SDH) problem in $\mathbb{G}$ is defined as follows: Given an element P of $\mathbb{G}$, and an l-tuple $(xP, x^2P, \cdots, x^lP)$ for some unknown $x \in \mathbb{Z}_q$, compute a new tuple $(c, \frac{1}{x+c}P)$, where $c \in \mathbb{Z}_q$.*

Definition 10 *(Strong Diffie-Hellman (SDH) Assumption) Let $\mathcal{A}$ be an adversary that takes an input of $(P, xP, x^2P, \cdots, x^lP)$ for some unknown $x \in \mathbb{Z}_q$, and returns a new tuple $(c, \frac{1}{x+c}P)$. We consider the following random experiment.*

$$
\begin{array}{l}
\texttt{Experiment } \mathbf{Exp}_{\mathcal{A}}^{\mathrm{SDH}} \\
\quad x \xleftarrow{R} \mathbb{Z}_q,\ (c, \alpha) \leftarrow \mathcal{A}\left(P, xP, x^2P, \cdots, x^lP\right) \\
\quad \texttt{if } \alpha = \frac{1}{x+c}P \texttt{ then } b \leftarrow 1 \texttt{ else } b \leftarrow 0 \\
\quad \texttt{return } b
\end{array}
$$

We define the corresponding success probability of $\mathcal{A}$ in solving the SDH problem via

$$\mathbf{Succ}_{\mathcal{A}}^{\mathrm{SDH}} = \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SDH}} = 1\right]$$

Let $\tau \in \mathbb{N}$ and $\epsilon \in [0, 1]$. We say that the SDH is (τ, ϵ)-secure if no polynomial algorithm $\mathcal{A}$ running in time τ has success $\mathbf{Succ}_{\mathcal{A}}^{\mathrm{SDH}} \geq \epsilon$.

Definition 11 *(Strong Diffie-Hellman-2 (SDH2) Problem) The strong Diffie-Hellman-2 (SDH2) problem in $\mathbb{G}$ is defined as follows: Let P be an element of $\mathbb{G}$. Given $(xP, yP, \frac{1}{x+y}P)$ for unknown $x, y \in \mathbb{Z}_q$, l_1 distinct tuples $(c_i, \frac{1}{x+c_i}P)$, where $c_i \in \mathbb{Z}_q$, $i \in \{1, 2, \cdots, l_1\}$, and another tuple $(y^2P, \cdots, y^{l_2}P)$, compute a new tuple $(m, \frac{1}{y+m}P)$, where $m \in \mathbb{Z}_q$.*

Definition 12 *(SDH2 Assumption) Let $\mathcal{A}$ be an adversary that takes an input of P, xP, yP, $\frac{1}{x+y}P$, $c_1, \frac{1}{x+c_1}P$, $c_2, \frac{1}{x+c_2}P$, $\cdots$, $c_{l_1}, \frac{1}{x+c_{l_1}}P$ for some unknown $x, y, c_1, \cdots, c_{l_1} \in \mathbb{Z}_q$, and another tuple $(y^2P, \cdots, y^{l_2}P)$, returns a new tuple (c, α). We consider the following random experiment.*

$$
\begin{array}{l}
\texttt{Experiment } \mathbf{Exp}_{\mathcal{A}}^{\mathrm{SDH2}} \\
\quad x, y, c_1, \cdots, c_{l_1} \xleftarrow{R} \mathbb{Z}_q \\
\quad (m, \alpha) \leftarrow \mathcal{A}\left(\begin{array}{l} P, xP, yP, \frac{1}{x+y}P \\ c_1, \frac{1}{x+c_1}P, c_2, \frac{1}{x+c_2}P, \cdots, c_{l_1}, \frac{1}{x+c_{l_1}}P \\ y^2P, \cdots, y^{l_2}P \end{array}\right) \\
\quad \texttt{if } \alpha = \frac{1}{y+m}P \texttt{ then } b \leftarrow 1 \texttt{ else } b \leftarrow 0 \\
\quad \texttt{return } b
\end{array}
$$

We define the corresponding success probability of $\mathcal{A}$ in solving the SDH2 problem via

$$\mathbf{Succ}_{\mathcal{A}}^{\mathrm{SDH2}} = \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SDH2}} = 1\right]$$

Let $\tau \in \mathbb{N}$ and $\epsilon \in [0, 1]$. We say that the SDH2 is (τ, ϵ)-secure if no polynomial algorithm $\mathcal{A}$ running in time τ has success $\mathbf{Succ}_{\mathcal{A}}^{\mathrm{SDH2}} \geq \epsilon$.

2.2.3 Bilinear Groups of Composite Order

Let p, q be two distinct large primes, and $n = pq$. Groups $(\mathbb{G}, \mathbb{G}_T)$ of composite order n are called *bilinear map groups of composite order* if there is a mapping $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ with the following properties [53, 54]:

1. Bilinearity: $e(g^a, h^b) = e(g, h)^{ab}$ for any $(g, h) \in \mathbb{G}^2$ and $a, b \in \mathbb{Z}_n$;
2. Non-degeneracy: there exists $g \in \mathbb{G}$ such that $e(g, g)$ has order n in $\mathbb{G}_T$. In other words, $e(g, g)$ is a generator of $\mathbb{G}_T$, whereas g generates $\mathbb{G}$.
3. Computability: there exists an efficient algorithm to compute $e(g, h) \in \mathbb{G}_T$ for all $(g, h) \in \mathbb{G}$.

Note that 1) we use the multiplicative group to represent the group $\mathbb{G}$, which, however, can be instantiated by the elliptic curve addition group, i.e., the modified Weil pairing or Tate pairing [53]; 2) the vast majority of cryptosystems based on pairings assume for simplicity that bilinear groups have prime order q. In composite order case, it is important that the pairing is defined over a group $\mathbb{G}$ containing $|\mathbb{G}| = n$ elements, where $n = pq$ has a (ostensibly hidden) factorization in two large primes, $p \neq q$; 3) those complexity assumptions above in bilinear group of prime order also hold in bilinear group of composite order.

Definition 13 *(Composite Bilinear Generator) A composite bilinear parameter generator $\mathcal{CG}en$ is a probabilistic algorithm that takes a security parameter k as input, and outputs a 5-tuple $(n, g, \mathbb{G}, \mathbb{G}_T, e)$, where $n = pq$ and p, q are two k-bit prime numbers, $\mathbb{G}, \mathbb{G}_T$ are two groups with order n, $g \in \mathbb{G}$ is a generator, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a non-degenerated and efficiently computable bilinear map.*

Let $\mathbf{g}$ be a generator of $\mathbb{G}$, then $g = \mathbf{g}^q \in \mathbb{G}$ can generate the subgroup $\mathbb{G}_p = \{g^0, g^1, \cdots, g^{p-1}\}$ of order p, and $g' = \mathbf{g}^p \in \mathbb{G}$ can generate the subgroup $\mathbb{G}_q = \{g'^0, g'^1, \cdots, g'^{q-1}\}$ of order q in $\mathbb{G}$. In the following, we define the quantitative notion of the complexity of the SubGroup Decision (SGD) Problem [53].

Definition 14 *(SubGroup Decision (SGD) Problem) The SubGroup Decision (SGD) problem in $\mathbb{G}$ is as follows: Given a tuple $(e, \mathbb{G}, \mathbb{G}_T, n, h)$, where the element h is randomly drawn from either $\mathbb{G}$ or subgroup $\mathbb{G}_q$, decide whether or not $h \in \mathbb{G}_q$.*

Definition 15 *(SGD Assumption) Let $\mathcal{A}$ be an adversary that takes an input of h drawn from either $\mathbb{G}$ or subgroup $\mathbb{G}_q$, and returns a bit $b' \in \{0,1\}$. We consider the following random experiments.*

$$\begin{array}{l}\texttt{Experiment } \mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}} \\ \quad \widetilde{b} \leftarrow \{0,1\} \\ \quad \textit{if } \widetilde{b} = 0\textit{, then } h \stackrel{R}{\leftarrow} \mathbb{G}_q\textit{; else if } \widetilde{b} = 1 \textit{ then } h \stackrel{R}{\leftarrow} \mathbb{G} \\ \quad \widetilde{b}' \leftarrow \mathcal{A}(e, \mathbb{G}, \mathbb{G}_T, n, h) \\ \quad \texttt{return } 1 \textit{ if } \widetilde{b}' = \widetilde{b}\textit{, } 0 \textit{ otherwise}\end{array}$$

We then define the advantage of $\mathcal{A}$ via

$$\begin{aligned}\mathbf{Adv}_{\mathcal{A}}^{\mathrm{SGD}} = \Big|&\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}} = 1 | \widetilde{b} = 0\right] \\ &- \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}} = 1 | \widetilde{b} = 1\right]\Big| \geq \epsilon\end{aligned}$$

Let $\tau \in \mathbb{N}$ and $\epsilon \in [0,1]$. We say that the SGD is (τ, ϵ)-secure if no adversary $\mathcal{A}$ running in time τ has an advantage $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{SGD}} \geq \epsilon$.

2.2.4 Asymmetric Bilinear Groups of Prime Order

Let $\mathbb{G}$, $\mathbb{G}'$, and $\mathbb{G}_T$ be three cyclic multiplicative groups of the same prime order q, i.e., $|\mathbb{G}| = |\mathbb{G}'| = |\mathbb{G}_T| = q$. Let g be a generator of $\mathbb{G}$, g' be a generator of $\mathbb{G}'$, and ψ be an isomorphism from $\mathbb{G}'$ to $\mathbb{G}$, with $\psi(g') = g$. An efficient admissible bilinear map $e : \mathbb{G} \times \mathbb{G}' \rightarrow \mathbb{G}_T$ with the following properties:

1. Bilinearity: for all $g_1 \in \mathbb{G}$, $g_2 \in \mathbb{G}'$ and $a, b \in \mathbb{Z}_q^*$, $e({g_1}^a, {g_2}^b) = e(g_1, g_2)^{ab}$;
2. Non-degeneracy: there exist $g_1 \in \mathbb{G}$ and $g_2 \in \mathbb{G}'$ such that $e(g_1, g_2) \neq 1_{\mathbb{G}_T}$;
3. Computability: there exists an efficient algorithm to compute $e(g_1, g_2) \in \mathbb{G}_T$ for any $g_1 \in \mathbb{G}$, $g_2 \in \mathbb{G}'$.

Such an admissible asymmetric bilinear map e can be constructed by the modified Weil or Tate pairings on the elliptic curves. As mentioned in [55, 56], the Tate pairing on MNT curves gives the efficient implementation, where $\mathbb{G} \neq \mathbb{G}'$, the *one-way* isomorphism ψ can be implemented by the trace map, and the representations of $\mathbb{G}$ can be expressed in 171 bits when the order q is a 170-bit prime. By this construction, the discrete logarithm problem in $\mathbb{G}$ can reach as hard as the discrete logarithm in $\mathbb{Z}_p^*$ where p is 1020 bits.

Definition 16 (Asymmetric Bilinear Generator) *An asymmetric bilinear parameter generator $\mathcal{AG}en$ is a probabilistic algorithm that takes a security parameter k as input and outputs a 7-tuple $(q, \mathbb{G}, \mathbb{G}', \mathbb{G}_T, e, g, g')$ as the bilinear parameters, including a prime number q with $|q| = k$, three cyclic groups $\mathbb{G}$, $\mathbb{G}'$, $\mathbb{G}_T$ of the same order q, an admissible bilinear map $e : \mathbb{G} \times \mathbb{G}' \to \mathbb{G}_T$ and generators g, g' of $\mathbb{G}$, $\mathbb{G}'$, respectively.*

Note that those complexity assumptions in symmetric bilinear groups above also hold in asymmetric bilinear groups.

2.3 Conditional Privacy-preserving Authentication

The Conditional Privacy-preserving Authentication (CPPA) technique is one kind of group signature [12], which is dedicated for vehicular communications to achieve conditional privacy-preserving authentication. Here, we first formalize the CPPA technique and its security notions, then present a provable secure CPPA scheme [57], followed by the formal security proofs with provable security technique.

2.3.1 Definition of CPPA

Definition 17 *(CPPA) A Conditional Privacy-preserving Authentication $\mathcal{CPPA}$ scheme is defined by the following algorithms:* system setup*,* key generation*,* anonymous authentication*, and* conditional tracking*.*

- *a system setup algorithm* **Setup***: it is a probabilistic algorithm run by a trusted authority (TA), which takes a security parameter k as input and outputs the system public parameters* params *and* master key*.*
- *a key generation algorithm* **KGen***: it is an algorithm run by TA, which takes the public parameters* params*,* master key *and an identifier of a vehicle $v_i \in V$ as input, and outputs a corresponding private key sk_i. This algorithm can be either probabilistic or deterministic.*
- *an anonymous authentication algorithm* **AnonyAuth***: a vehicle v_i takes as input a fresh nonce m_i, the private key sk_i, and the public parameters* params*, and outputs an anonymous signature σ of m_i. Then, any verifier takes as input the purported signature σ, the fresh nonce m_i, and the public parameters* params *and tests whether σ is valid. If it is valid, the anonymous authentication is passed; otherwise rejected.*

- *a conditional tracking algorithm* **CTrack***: Once a sensitive operation is disputed, the TA can provide* (σ, T_i) *to the TA. Then, the TA can track the real identity of the vehicle with* (σ, T_i)*.*

Correctness: *A conditional privacy-preserving authentication* $\mathcal{CPPA}$ *scheme should satisfy the following properties: a registered vehicle must be anonymously authenticated by the* **AnonyAuth** *algorithm with* σ*; once a vehicle executes a disputed operation, the real identity of the vehicle must be tracked by the* **CTrack** *algorithm.*

2.3.2 Security Notions

Full Anonymity on Signature σ

A typical approach to define full anonymity σ is the following experiment in the spirit of cpa-indistinguishability [53, 54]. In the $\mathcal{CPPA}$ setting, an adversary $\mathcal{A}$ is given the public parameters, as well as an access to the key generation oracle $\mathcal{O}_K$ on some registered vehicles in $V = \{v_1, v_2, \cdots\}$, to the anonymous signature oracle $\mathcal{O}_S$ on some valid signatures. Then, $\mathcal{A}$ outputs (v_0, v_1, m). After an anonymous signature σ^b corresponding to (v_b, m) is signed, where $b \in \{0, 1\}$, $\mathcal{A}$ must decide which vehicle in (v_0, v_1) signed the signature σ^b.

Definition 18 *(Full Anonymity) Let* $V = \{v_1, v_2, \cdots\}$ *be a group of registered vehicles,* k *and* t *be integers and* ϵ *be a real in* $[0, 1]$*, let* $\mathcal{CPPA}$ *be a conditional privacy-preserving authentication scheme with security parameter* k*. Let* $\mathcal{A}$ *be an adversary against the anonymity of* $\mathcal{CPPA}$*. We consider the following random experiment:*

$\texttt{Experiment } \mathbf{Exp}^{\mathit{anony}}_{\mathcal{CPPA},\mathcal{A}}(k)$
$\quad params, masterkey \xleftarrow{R} \mathbf{Setup}(k)$
$\quad$ *for each* $v_i \in V$ *do*
$\quad\quad sk_i \xleftarrow{R} \mathbf{KGen}(params, masterkey)$
$\quad (v_0, v_1, M) \leftarrow \mathcal{A}$
$\quad b \xleftarrow{R} \{0, 1\}$, $\sigma^b \leftarrow (v_b, M)$
$\quad b' \leftarrow \mathcal{A}^{\mathcal{O}_K, \mathcal{O}_S}(params, \sigma^b)$
$\quad \texttt{if } b = b' \textit{ is valid } \texttt{then return } \ b^* \leftarrow 1 \texttt{ else } b^* \leftarrow 0$
$\quad \texttt{return } b^*$

We define the success probability of $\mathcal{A}$ via

$$\mathbf{Succ}^{\mathsf{anony}}_{\mathcal{CPPA},\mathcal{A}}(k) = 2\Pr\left[\mathbf{Exp}^{\mathsf{anony}}_{\mathcal{CPPA},\mathcal{A}}(k) = 1\right] - 1 = 2\Pr\left[b = b'\right] - 1$$

$\mathcal{CPPA}$ is said to be (k, t, ϵ)-anony secure, if no adversary $\mathcal{A}$ running in time t has a success $\mathbf{Succ}^{\mathsf{anony}}_{\mathcal{CPPA},\mathcal{A}}(k) \geq \epsilon$.

Full Traceability on Signature σ

Full traceability on signature σ can be described in the following game [53, 54]. In the $\mathcal{CPPA}$ setting, an adversary $\mathcal{A}$ is given the public parameters and tracking information ($\mathsf{trackinfo}$), as well as an access to the key generation oracle $\mathcal{O}_K$ on some registered users in $V = \{v_1, v_2, \cdots\}$, to the anonymous signature oracle $\mathcal{O}_S$ on some valid signatures, where the validity of signature and identity tracing can be checked by $\mathcal{A}$. At some point, $\mathcal{A}$ outputs a forged anonymous signature $\sigma^\star$ with its tracking identity $v^\star \in V$ and a message $m^\star$. There is the natural restriction that in the returned signature $\sigma^\star$, the identity $v^\star$ has not been queried on $\mathcal{O}_K$ and $(v^\star, m^\star)$ has not been obtained from $\mathcal{O}_S$.

Definition 19 *(Full Traceability) Let $V = \{v_1, v_2, \cdots\}$ be a group of registered vehicles, k and t be integers and ϵ be a real in $[0, 1]$; let $\mathcal{CPPA}$ be a conditional privacy-preserving authentication scheme with security parameter k. Let $\mathcal{A}$ be an adversary against the traceability of $\mathcal{CPPA}$. We consider the following random experiment:*

$\mathtt{Experiment}\ \mathbf{Exp}^{\mathsf{trace}}_{\mathcal{CPPA},\mathcal{A}}(k)$
  $params, masterkey \xleftarrow{R} \mathsf{Setup}(k)$
  for each $v_i \in V$ *do*
    $sk_i \xleftarrow{R} \mathsf{KGen}(params, masterkey)$
  $\sigma^\star \leftarrow \mathcal{A}^{\mathcal{O}_K, \mathcal{O}_S}(params, v^\star)$
  $\mathtt{if}\ \sigma^\star$ *is valid* $\mathtt{then}\ b^* \leftarrow 1\ \mathtt{else}\ b^* \leftarrow 0$
  $\mathtt{return}\ b^*$

We define the success probability of $\mathcal{A}$ via

$$\mathbf{Succ}^{\mathsf{trace}}_{\mathcal{CPPA},\mathcal{A}}(k) = \Pr\left[\mathbf{Exp}^{\mathsf{trace}}_{\mathcal{CPPA},\mathcal{A}}(k) = 1\right]$$

$\mathcal{CPPA}$ is said to be (k, t, ϵ)-trace secure, if no adversary $\mathcal{A}$ running in time t has a success $\mathbf{Succ}^{\mathsf{trace}}_{\mathcal{CPPA},\mathcal{A}}(k) \geq \epsilon$.

2.3.3 A Provably Secure CPPA Scheme

In the following, we introduce a provably secure CPPA scheme [57], which includes the following four parts: system setup Setup, key generation KGen, anonymous authentication AnonyAuth, and conditional tracking CTrack.

Setup: Given the security parameter k, the bilinear map groups $(\mathbb{G}, \mathbb{G}_T, e)$ of composite order $n = pq$ are generated by running $\mathcal{CG}en(k)$, where $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ and p, q are two large primes with $|p| = |q| = k$. Let $\mathbb{G}_p$, $\mathbb{G}_q$ denote $\mathbb{G}$'s two subgroups of orders p and q, respectively. Then, a trusted authority (TA) chooses two elements (g, u) of $\mathbb{G}$, one generator h of $\mathbb{G}_q$, two random exponents $\alpha, a \in \mathbb{Z}_n^*$, and a collision-resistant hash function $H : \{0,1\}^* \to \mathbb{Z}_n^*$. After these, the TA sets the *master key* (g^α, a, q) and the system public parameters $params = (\mathbb{G}, \mathbb{G}_T, e, n, g, u, e(g,g)^\alpha, A = g^a, h, H)$.

KGen: When a vehicle $v_i \in V$ registers to the system, the TA first chooses a random number $s_i \in \mathbb{Z}_n^*$, and computes the private key $sk_i = (s_i, g^{\frac{1}{a+s_i}})$ for v_i. In addition, TA also stores $(v_i, g^{s_i q})$ in the tracking list used for CTrack.

AnonyAuth: Assume that the vehicle v_i wants to anonymously authenticate itself at time T_i, it uses the secret key $sk_i = (s_i, g^{\frac{1}{a+s_i}})$ to execute the following steps:

- Use the anonymous authentication key s_i to compute ρ, where

$$\rho = (\rho_1, \rho_2, \rho_3) = \left(g^{s_i}, g^{\frac{1}{a+s_i}}, u^{\frac{1}{s_i + H(T_i)}}\right) \tag{2.5}$$

- Choose three random numbers $z_1, z_2, z_3 \in \mathbb{Z}_n^*$, and compute $(\sigma_1, \sigma_2, \sigma_3)$ and the proof (π_1, π_2)

$$\sigma_1 = \rho_1 \cdot h^{z_1}; \sigma_2 = \rho_2 \cdot h^{z_2}; \sigma_3 = \rho_3 \cdot h^{z_3}; \tag{2.6}$$

$$\begin{aligned} \pi_1 &= \rho_2^{z_1} (A\rho_1)^{z_2} h^{z_1 z_2}; \\ \pi_2 &= \rho_3^{z_1} (g^{H(T_i)} \rho_1)^{z_3} h^{z_1 z_3}; \end{aligned} \tag{2.7}$$

- Set $\text{CPPA}(T_i) = \sigma = (\sigma_1, \sigma_2, \sigma_3, \pi_1, \pi_2)$ as the authentication information.

After receiving $\text{CPPA}(T_i)$ at time T_i', any verifier checks whether or not $T_i' - T_i \leq \Delta T$, where ΔT is the expected legal time interval for transmission delay. If it does not hold, the authentication fails. Otherwise, the verifier executes the following steps:

- Compute T_1 and T_2, where

$$T_1 = \frac{e(\sigma_1 A, \sigma_2)}{e(g,g)}; T_2 = \frac{e(\sigma_1 g^{H(T_i)}, \sigma_3)}{e(g,u)}; \tag{2.8}$$

- Verify T_1 and T_2,

$$T_1 \stackrel{?}{=} e(\pi_1, h); T_2 \stackrel{?}{=} e(\pi_2, h); \tag{2.9}$$

  If the above two equations hold, the vehicle v_i is anonymously authenticated, i.e., v_i is a valid registered user in the system, but the real identity is not disclosed; Otherwise, the authentication fails.

CTrack: Once the dispute occurs, the TA can fast track the real identity of vehicle with $\text{CPPA}(T_i) = \sigma = (\sigma_1, \sigma_2, \sigma_3, \pi_1, \pi_2)$ as follows:

- Use the master key q to compute

$$\sigma_1^q = \left(g^{s_i} h^{t_1}\right)^q = g^{s_i q} \quad (\because h \text{ is a generator of } \mathbb{G}_q) \tag{2.10}$$

- Trace the real identity v_i by looking up the entry $(v_i, g^{s_i q})$ in the tracking list. As a result, the CPPA technique achieves conditional privacy-preserving authentication. Note that during the conditional tracking, no pairing computation is needed, the tracking is faster than other previously reported ones [12].

2.3.4 Security Proofs

In this section, under the complex assumptions in Section 2.2, we analyze the security of the proposed CPPA scheme in the standard model.

Theorem 1 *(*Anonymity of σ*) Let $\mathcal{A}$ be an adversary against the proposed $\mathcal{CPPA}$ scheme in the standard model. Assume that $\mathcal{A}$ has the success probability $\mathbf{Succ}^{\mathsf{anony}}_{\mathcal{CPPA},\mathcal{A}} \geq \epsilon$ to break the full anonymity of σ within the running time τ. Then, there exist $\epsilon' \in [0,1]$ and $\tau' \in \mathbb{N}$ as follows*

$$\epsilon' \geq \frac{\epsilon}{2}; \quad \tau' \leq \tau + \Theta(.) \tag{2.11}$$

such that the SGD problem can be solved with probability ϵ' within time τ'.

Proof. We define a sequence of games Game_0, Game_1, $\cdots$ of modified attacks starting from the actual adversary $\mathcal{A}$ [58]. All the games operate on the same underlying probability space: the system parameters and master key, the coin tosses of $\mathcal{A}$. Let $(e, \mathbb{G}, \mathbb{G}_T, n, \tilde{h})$ be a random instance of SGD, we will use these incremental games to reduce the SGD instance to the adversary $\mathcal{A}$ against the full anonymity of σ in the proposed $\mathcal{CPPA}$ scheme.

Game_0: This is the real attack game. In the game, the TA chooses the master key (g^α, a, q) and the system public parameters $params = (\mathbb{G}, \mathbb{G}_T, e, n, g, u,\ e(g,g)^\alpha,\ A = g^a, h, H)$, and feeds the adversary $\mathcal{A}$ with these system public parameters. First, the adversary $\mathcal{A}$ exactly makes q_K queries to $\mathcal{O}_K$ on group members' private keys, q_S queries to $\mathcal{O}_S$ on anonymous signatures, and at some point, $\mathcal{A}$ chooses a message M and two vehicles (v_0, v_1) for challenge. Note that the adversary $\mathcal{A}$ can know the private keys of (v_0, v_1) in the game. Then, we flip a coin $b \in \{0, 1\}$ and produce v_b's anonymous signature $\sigma = (\sigma_1, \sigma_2, \sigma_3, \pi_1, \pi_2)$ as the challenge to the adversary $\mathcal{A}$. The anonymous signature comes from sk_i and three random numbers $z_1, z_2, z_3 \in \mathbb{Z}_n^*$, and $\sigma_1 = \rho_1 \cdot h^{z_1}$, $\sigma_2 = \rho_2 \cdot h^{z_2}$, $\sigma_3 = \rho_3 \cdot h^{z_3}$, $\pi_1 = \rho_2^{z_1}(A\rho_1)^{z_2} h^{z_1 z_2}$ and $\pi_2 = \rho_3^{z_1}(g^{H(M)}\rho_1)^{z_3} h^{z_1 z_3}$ with $(\rho_1,\ \rho_2,\ \rho_3) = \left(g^{s_b}, g^{\frac{1}{a+s_b}}, u^{\frac{1}{s_b+H(M)}}\right)$. Finally, the adversary $\mathcal{A}$ outputs a bit $b' \in \{0, 1\}$. In any Game_j, we denote by Guess_j the event $b = b'$. Then, by definition, we have

$$\begin{aligned} \epsilon \leq \mathbf{Succ}^{\mathsf{anony}}_{\mathcal{SPP},\mathcal{A}} = 2\Pr[b = b'] - 1 = 2\Pr[\mathsf{Guess}_0] - 1, \\ \Pr[\mathsf{Guess}_0] \geq \frac{\epsilon}{2} + \frac{1}{2}. \end{aligned} \tag{2.12}$$

Game_1: In this game, we modify the simulation by replacing the master key and system parameters with the SGD challenge $(e, \mathbb{G}, \mathbb{G}_T, n, \tilde{h})$. In specific, we choose $\alpha, a \in \mathbb{Z}_n^*$, compute the master key $(g^\alpha, a, \sqcup)$, and publish the systems parameters as $params = (\mathbb{G}, \mathbb{G}_T, e, n, g, u,\ e(g,g)^\alpha,\ A = g^a, \tilde{h}, H)$. Since the distribution of $(e(g,g)^\alpha, A = g^a, \tilde{h})$ is unchanged in the eye of the adversary, we thus have

$$\Pr[\mathsf{Guess}_1] = \Pr[\mathsf{Guess}_0] \tag{2.13}$$

Game_2: In this game, we will simulate the key generation oracle $\mathcal{O}_K$ on q_K group members' anonymous authentication keys queries and q_S anonymous signature oracle $\mathcal{O}_S$ queries. Because we have the master key $(g^\alpha, a, \sqcup)$, it is not difficult for us to simulate $\mathcal{O}_K$ and $\mathcal{O}_S$. Thus, this game is perfectly indistinguishable from the previous one, and we have

$$\Pr[\mathsf{Guess}_2] = \Pr[\mathsf{Guess}_1]. \tag{2.14}$$

Game_3: In this game, we consider, if $\tilde{h}$ in the SGD challenge $(e, \mathbb{G}, \mathbb{G}_T, n, \tilde{h})$ actually belongs to the subgroup $\mathbb{G}_q$, i.e., $\widetilde{b} = 0$ in experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}}$, we know this game is indistinguishable from the previous one, we have

$$\Pr[\mathsf{Guess}_3|\widetilde{b} = 0] = \Pr[\mathsf{Guess}_2], \tag{2.15}$$

and

$$\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}} = 1|\widetilde{b} = 0\right] = \Pr[\mathsf{Guess}_3|\widetilde{b} = 0]. \tag{2.16}$$

When $\tilde{h}$ does not belong to $\mathbb{G}_q$, i.e., $\tilde{h} \in \mathbb{G}$ and $\widetilde{b} = 1$ in experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}}$, $(\rho_1, \rho_2, \rho_3) = \left(g^{s_b}, g^{\frac{1}{a+s_b}}, u^{\frac{1}{s_b+H(M)}}\right)$ is masked by random numbers z_1, z_2, z_3 in $\mathbb{Z}_n^*$, i.e., $\sigma_1 = \rho_1 \cdot \widetilde{h}^{z_1}$, $\sigma_2 = \rho_2 \cdot \widetilde{h}^{z_2}$, $\sigma_3 = \rho_3 \cdot \widetilde{h}^{z_3}$, $(\sigma_1, \sigma_2, \sigma_3)$ reveals nothing about the challenge b. In the following, we show that (π_1, π_2) also does not reveal the challenge b.

For a specific $(\sigma_1, \sigma_2, \sigma_3)$, we know there exist random numbers $z_{1(b)}$, $z_{2(b)}$, $z_{3(b)}$, $z_{1(1-b)}$, $z_{2(1-b)}$, $z_{3(1-b)} \in \mathbb{Z}_n^*$ such that

$$\begin{cases} \sigma_1 = g^{s_b} \cdot \widetilde{h}^{z_{1(b)}} = g^{s_{1-b}} \cdot \widetilde{h}^{z_{1(1-b)}} \\ \sigma_2 = g^{\frac{1}{a+s_b}} \cdot \widetilde{h}^{z_{2(b)}} = g^{\frac{1}{a+s_{1-b}}} \cdot \widetilde{h}^{z_{2(1-b)}} \\ \sigma_3 = u^{\frac{1}{s_b+H(M)}} \cdot \widetilde{h}^{z_{3(b)}} = u^{\frac{1}{s_{1-b}+H(M)}} \cdot \widetilde{h}^{z_{3(1-b)}} \end{cases} \tag{2.17}$$

For example, suppose $\widetilde{h} = g^{\eta} = u^{\xi}$, $\varepsilon = \frac{a+s_b}{a+s_{1-b}}$, $\zeta = \frac{s_b+H(M)}{s_{1-b}+H(M)}$ for some unknown $\eta, \xi \in \mathbb{Z}_n^*$, we know

$$\begin{cases} z_{1(1-b)} = z_{1(b)} + \dfrac{s_b - s_{1-b}}{\eta} \\ z_{2(1-b)} = z_{2(b)} + \dfrac{1}{\eta}\left(\dfrac{1}{a+s_b} - \dfrac{1}{a+s_{1-b}}\right) \\ \qquad\quad = z_{2(b)} + \dfrac{1-\varepsilon}{\eta(a+s_b)} \\ z_{3(1-b)} = z_{3(b)} + \dfrac{1}{\xi}\left(\dfrac{1}{s_b+H(M)} - \dfrac{1}{s_{1-b}+H(M)}\right) \\ \qquad\quad = z_{3(b)} + \dfrac{1-\zeta}{\xi(s_b+H(M))} \end{cases} \tag{2.18}$$

Then,

$$\begin{cases} \pi_{1(b)} = g^{\frac{z_{1(b)}}{a+s_b}} g^{(a+s_b)z_{2(b)}} \widetilde{h}^{z_{1(b)}z_{2(b)}} \\ \pi_{1(1-b)} = g^{\frac{z_{1(1-b)}}{a+s_{1-b}}} g^{(a+s_{1-b})z_{2(1-b)}} \widetilde{h}^{z_{1(1-b)}z_{2(1-b)}} \end{cases} \tag{2.19}$$

$$
\begin{aligned}
&\log_g^{\pi_{1(1-b)}} \\
=&\frac{z_{1(b)}+\frac{s_b-s_{1-b}}{\eta}}{a+s_{1-b}}+(a+s_{1-b})\left(z_{2(b)}+\frac{1-\varepsilon}{\eta(a+s_b)}\right) \\
&+\eta\left(z_{1(b)}+\frac{s_b-s_{1-b}}{\eta}\right)\left(z_{2(b)}+\frac{1-\varepsilon}{\eta(a+s_b)}\right) \\
=&\frac{z_{1(b)}}{a+s_{1-b}}+\frac{s_b-s_{1-b}}{\eta(a+s_{1-b})}+az_{2(b)}+s_{1-b}z_{2(b)} \\
&+\frac{(1-\varepsilon)(a+s_b)}{\eta(a+s_b)}+\eta z_{1(b)}z_{2(b)}+s_bz_{2(b)} \\
&-s_{1-b}z_{2(b)}+\frac{z_{1(b)}(1-\varepsilon)}{a+s_b}+\frac{(1-\varepsilon)(s_b-s_{1-b})}{\eta(a+s_b)} \\
=&\frac{z_{1(b)}}{a+s_b}+(a+s_b)z_{2(b)}+\eta z_{1(b)}z_{2(b)}=\log_g^{\pi_{1(b)}} \\
\Rightarrow&\pi_{1(b)}=\pi_{1(1-b)}
\end{aligned}
\tag{2.20}
$$

In addition,

$$
\begin{cases}
\pi_{2(b)}=u^{\frac{z_{1(b)}}{s_b+H(M)}}g^{(s_b+H(M))z_{3(b)}}\widetilde{h}^{z_{1(b)}z_{3(b)}} \\
\pi_{2(1-b)}=u^{\frac{z_{1(1-b)}}{s_{1-b}+H(M)}}g^{(s_{1-b}+H(M))z_{3(1-b)}}\widetilde{h}^{z_{1(1-b)}z_{3(1-b)}}
\end{cases}
\tag{2.21}
$$

$$
\begin{aligned}
&\log_g^{\pi_{2(1-b)}} \\
=&\left(\frac{z_{1(b)}+\frac{s_b-s_{1-b}}{\eta}}{s_{1-b}+H(M)}\right)\frac{\eta}{\xi}+(s_{1-b}+H(M)) \\
&\cdot\left(z_{3(b)}+\frac{1-\zeta}{\xi(s_b+H(M))}\right)+\eta\left(z_{1(b)}+\frac{s_b-s_{1-b}}{\eta}\right) \\
&\cdot\left(z_{3(b)}+\frac{1-\zeta}{\xi(s_b+H(M))}\right) \\
=&\left(\frac{z_{1(b)}}{s_{1-b}+H(M)}\right)\frac{\eta}{\xi}+\left(\frac{s_b-s_{1-b}}{s_{1-b}+H(M)}\right)\frac{1}{\xi} \\
&+s_{1-b}\cdot z_{3(b)}+H(M)\cdot z_{3(b)}+\left(\frac{s_{1-b}+H(M)}{s_b+H(M)}\right)\frac{1-\zeta}{\xi} \\
&+\eta z_{1(b)}z_{3(b)}+s_b z_{3(b)}+s_{1-b}z_{3(b)} \\
&+\left(\frac{z_{1(b)}(1-\zeta)}{s_b+H(M)}\right)\frac{\eta}{\xi}+\left(\frac{s_b-s_{1-b}}{s_b+H(M)}\right)\frac{1-\zeta}{\xi} \\
=&\left(\frac{z_{1(b)}}{s_b+H(M)}\right)\frac{\eta}{\xi}+(s_b+H(M))z_{3(b)}+\eta z_{1(b)}z_{3(b)} \\
=&\log_g \pi_2(b) \\
\Rightarrow&\pi_2(b)=\pi_2(1-b)
\end{aligned}
\tag{2.22}
$$

Since $\pi_1(b)=\pi_1(1-b)$ and $\pi_2(b)=\pi_2(1-b)$ are independent of b, we have

$$
\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}}=1|\widetilde{b}=1\right]=\Pr[\mathsf{Guess}_3|\widetilde{b}=1]=\frac{1}{2} \tag{2.23}
$$

As a result, from Eqs. (2.12)-(2.23), we have

$$
\begin{aligned}
\epsilon' &= \mathbf{Adv}_{\mathcal{A}}^{\mathrm{SGD}} \\
&= \left|\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}}=1|\widetilde{b}=0\right]-\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{SGD}}=1|\widetilde{b}=1\right]\right| \\
&\geq \left|\frac{\epsilon}{2}+\frac{1}{2}-\frac{1}{2}\right|=\frac{\epsilon}{2}
\end{aligned}
\tag{2.24}
$$

In addition, we can obtain the claimed bound for $\tau' \leq \tau+\Theta(.)$, where $\Theta(.)$ is the time cost used for queries on $\mathcal{O}_K$ and $\mathcal{O}_S$ in the simulation. Thus, the proof is completed. □

Theorem 2 *(*Traceability of σ*) Let $V = \{v_1, v_2, \cdots\}$ be a group of registered vehicles. Let $\mathcal{A}$ be an* ef-cma *adversary against the proposed $\mathcal{CPPA}$ scheme in the standard model. Assume that $\mathcal{A}$ has the success probability $\mathbf{Succ}^{trace}_{\mathcal{CPPA},\mathcal{A}} \geq \epsilon$ to existentially forge the anonymous signature σ (in a weak chosen message attack), within the running time τ, after making q_K, q_S queries to the key generation oracle $\mathcal{O}_K$ and the signature oracle $\mathcal{O}_S$, respectively. Then, there exist $\epsilon' \in [0,1]$ and $\tau' \in \mathbb{N}$ as follows*

$$\epsilon' \geq \epsilon; \quad \tau' \leq \tau + \Theta(*) \tag{2.25}$$

such that the SDH2 problem can be solved with probability ϵ' within time τ', where $\Theta(*)$ is the time cost in the simulation.

Proof. We define a sequence of games Game_0, Game_1, $\cdots$ of modified attacks starting from the actual adversary $\mathcal{A}$ [58]. All the games operate on the same underlying probability space: the system parameters and master key, the coin tosses of $\mathcal{A}$. Let

$$\begin{pmatrix} g, g^x, g^y, g^{\frac{1}{x+y}} \\ c_1, g^{\frac{1}{x+c_1}}, c_2, g^{\frac{1}{x+c_2}}, \cdots, c_{l_1}, g^{\frac{1}{x+c_{q_K}}} \\ g^{(y^2)}, \cdots, g^{(y^{q_S})} \end{pmatrix}$$

be a random instance of SDH2, where g is an element of the subgroup $\mathbb{G}_p$ of $\mathbb{G}$, we will use these incremental games to reduce the SDH2 instance to the adversary $\mathcal{A}$ against the existential forgery under a weak chosen message attack in the proposed $\mathcal{CPPA}$ scheme.

Game_0: This is the real attack game. In the game, the TA chooses the master key (g^α, a, q) and the system public parameters $params = (\mathbb{G}, \mathbb{G}_T, e, n, g, u,\ e(g,g)^\alpha,\ A = g^a, h, H)$, and feeds the adversary $\mathcal{A}$ with these system public parameters and the master key q for tracking. First, the adversary $\mathcal{A}$ outputs a list of q_S distinct messages $m_1, m_2, \cdots, m_{q_S} \in \mathbb{Z}_n^*$, where $m_i = H(M_i)$ for $i = 1, 2, \cdots, q_S$, and exactly makes q_K queries to $\mathcal{O}_K$ on group members' anonymous authentication keys, q_S queries to $\mathcal{O}_S$ on a specific vehicle $v^\star$'s signatures, where $v^\star$ has not been queried on $\mathcal{O}_K$. Note that the signatures queried from $\mathcal{O}_S$ can be traced by $\mathcal{A}$ with q and tracking list. In the end, the adversary $\mathcal{A}$ outputs a valid signature $(\sigma^\star, m^\star)$ of $v^\star$, where $m^\star$ is not queried to $\mathcal{O}_S$ before. In any Game_j, we denote by $\Pr[\mathsf{Forge}_j]$ the forgery success probability of $\mathcal{A}$ in Game_j. Then, by definition, we have

$$\Pr[\mathsf{Forge}_0] = \mathbf{Succ}^{\mathsf{trace}}_{\mathcal{CPPA},\mathcal{A}} \geq \epsilon \tag{2.26}$$

Game_1: In this game, we confine the elements (g, u) in the subgroup $\mathbb{G}_p$, i.e., $g^p = 1$, $u^p = 1$. Since $g, u \in \mathbb{G}_p$ still belong to $\mathbb{G}$ in accordance with the scheme design, the

adversary $\mathcal{A}$ cannot detect this trick. Therefore, we have

$$\Pr[\mathsf{Forge}_1] = \Pr[\mathsf{Forge}_0] \tag{2.27}$$

Game_2: In this game, we modify the simulation by replacing the system parameter $(A = g^a, u)$ with (g^x, u'), where u' is generated by adopting the same simulating approach in [59], i.e., given $(g^y, g^{(y^2)}, \cdots, g^{(y^{q_S})}$ and $m_1, m_2, \cdots, m_{q_S} \in \mathbb{Z}_n^*$, we can generate $(u', (u')^y, (u')^{\frac{1}{y+m_1}}, \cdots, (u')^{\frac{1}{y+m_{q_S}}})$. Since the distribution of $(A = g^a, u)$ is unchanged, we thus have

$$\Pr[\mathsf{Forge}_2] = \Pr[\mathsf{Forge}_1] \tag{2.28}$$

Game_3: In this game, we will simulate the key generation oracle $\mathcal{O}_K$ on q_K group members' anonymous authentication keys queries. Concretely, when a fresh query on vehicle $v_i \in V$ is queried, we randomly chooses an un-queried c_i, records $(v_i, c_i, g^{c_i q})$, and return $(c_i, g^{\frac{1}{x+c_i}})$ as the answer to the oracle query. Since $(c_i, g^{\frac{1}{x+c_i}})$ is uniformly distributed, this game is therefore perfectly indistinguishable from the previous one. Hence,

$$\Pr[\mathsf{Forge}_3] = \Pr[\mathsf{Forge}_2]. \tag{2.29}$$

Game_4: In this game, we simulate a total of q_s times signing oracle $\mathcal{O}_S$. Note that, since the adversary $\mathcal{A}$ has already gotten other group members' private keys, $\mathcal{A}$ only queries the signing oracle $\mathcal{O}_S$ on the challenged v^*. For any fresh query on m_i, we first pick the corresponding $(u')^{\frac{1}{y+m_i}}$, set $\rho = (\rho_1, \rho_2, \rho_3) = (g^y, g^{\frac{1}{x+y}}, (u')^{\frac{1}{y+m_i}})$. Then, we choose three random numbers $z_1, z_2, z_3 \in \mathbb{Z}_n^*$, and compute $\sigma_1 = \rho_1 \cdot h^{z_1}$, $\sigma_2 = \rho_2 \cdot h^{z_2}$, $\sigma_3 = \rho_3 \cdot h^{z_3}$; and $\pi_1 = \rho_2^{z_1}(A\rho_1)^{z_2} h^{z_1 z_2}$, $\pi_2 = \rho_3^{z_1}(g^{m_i}\rho_1)^{z_3} h^{z_1 z_3}$. In the end, we return $\sigma = (\sigma_1, \sigma_2, \sigma_3, \pi_1, \pi_2)$ as the answer to the oracle query. In this game, $\mathcal{O}_S$ perfectly simulates the anonymous signature, and we will have

$$\Pr[\mathsf{Forge}_4] = \Pr[\mathsf{Forge}_3]. \tag{2.30}$$

We consider the adversary $\mathcal{A}$ outputs a valid anonymous signature $\sigma^\star = (\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, \pi_1^\star, \pi_2^\star)$ on $(m^\star, v^\star)$ at some point. Then, the forgery should satisfy the verification equations

$$\begin{aligned} e(\sigma_1^\star A, \sigma_2^\star) &= e(g,g)e(\pi_1^\star, h) \\ e(\sigma_1^\star g^{m^\star}, \sigma_3^\star) &= e(g,u)e(\pi_2^\star, h) \end{aligned} \tag{2.31}$$

We generate a random number λ such that $\lambda \equiv 1 \bmod p$ and $\lambda \equiv 0 \bmod q$. Then, we know $g^\lambda = g$, $u^\lambda = u$ and $h^\lambda = 1$. Thus, from Eq. (2.31), we have

$$\begin{aligned} & e(\sigma_1^\star A, \sigma_2^\star)^{\lambda^2} = (e(g,g)e(\pi_1^\star, h))^{\lambda^2} \\ \Rightarrow & e(\sigma_1^{\star\lambda} A^\lambda, \sigma_2^{\star\lambda}) = e(g^\lambda, g^\lambda)e(\pi_1^{\star\lambda}, h^\lambda) \\ \Rightarrow & e(\sigma_1^{\star\lambda} A, \sigma_2^{\star\lambda}) = e(g,g) \end{aligned} \tag{2.32}$$

$$
\begin{aligned}
& e(\sigma_1^\star g^{m^\star}, \sigma_3^\star)^{\lambda^2} = (e(g,u)e(\pi_2^\star, h))^{\lambda^2} \\
\Rightarrow & e(\sigma_1^{\star\lambda} g^{m^\star \lambda}, \sigma_3^{\star\lambda}) = e(g^\lambda, u^\lambda) e(\pi_2^{\star\lambda}, h^\lambda) \\
\Rightarrow & e(\sigma_1^{\star\lambda} g^{m^\star}, \sigma_3^{\star\lambda}) = e(g,u)
\end{aligned} \tag{2.33}
$$

and can get $\sigma_1^{\star\lambda} = g^y$, $\sigma_2^{\star\lambda} = g^{\frac{1}{x+y}}$, and $\sigma_3^{\star\lambda} = (u)^{\frac{1}{y+m^\star}} = (u')^{\frac{1}{y+m^\star}}$. Then, using the same approach in [59], we convert $(u')^{\frac{1}{y+m^\star}}$ back to $g^{\frac{1}{y+m^\star}}$ and output $(m^\star, g^{\frac{1}{y+m^\star}})$ as the challenge of SDH2 problem. In the end, from Eqs. (2.26)-(2.33), we have

$$
\epsilon' = \mathbf{Succ}_{\mathcal{A}}^{\mathrm{SDH2}} = \mathbf{Succ}_{\mathcal{SP},\mathcal{A}}^{\mathsf{trace}} \geq \epsilon \tag{2.34}
$$

In addition, we can obtain the claimed bound for $\tau' \leq \tau + \Theta(*)$, where $\Theta(*)$ is the time cost used for generating $(u', (u')^y, (u')^{\frac{1}{y+m_1}}, \cdots, (u')^{\frac{1}{y+m_{q_S}}})$ and $g^{\frac{1}{y+m^\star}}$. Thus, this completes the proof. □

2.4 Summary

In this chapter, we have discussed some related works, including basic concepts in social theory, bilinear groups, and conditional privacy-preserving authentication (CPPA) technique. In the following chapters, we will present several schemes/protocols one by one to address some security and privacy challenges in VSNs.

Chapter 3

Privacy-preserving Packet Forwarding Protocol for Vehicular DTNs

3.1 Introduction

In recent years, delay tolerant networks (DTNs), such as space communication and networking in sparsely populated areas [60], and vehicular ad hoc networks [61], have been subject to extensive research efforts. Unlike traditional tethered networks like the Internet, a DTN is a sparse mobile network where the connection between nodes in the network changes over time, and as a result the communication constantly suffers from higher delays and disconnections. Since a contemporaneous end-to-end path may never exist in DTNs, effective communication in DTNs requires cooperation of all the nodes for routing and forwarding, where, the intermediate nodes on a communication path are expected to store, carry and forward the packets in an opportunistic way, which is also named as opportunistic data forwarding. However, in most cases, DTNs could consist of many resource-constrained nodes, i.e., limited storage. If carried for a certain of time without an available downstream node, the packets have to be dropped by the carrying node, which thus incurs very unreliable forwarding in DTNs. Therefore, efficient packet forwarding in DTNs becomes an especially challenging issue, and a number of DTN packet forwarding schemes recently have been proposed to improve the reliability [62].

Over the past few years, vehicular ad hoc network, as a special case of DTNs and also known as vehicular DTN, has become increasingly attractive to the public due to its promis-

ing ability of improving road safety and traffic efficiency. In vehicular networks, a variety of applications can be enabled by vehicle-to-vehicle (V-2-V) and vehicle-to-infrastructure (V-2-I) communications to improve transportation systems. Unlike other forms of DTNs, there exists a fixed infrastructure in VANETs, i.e., Roadside Units (RSUs) deployed along the roadside. Recent efforts to improve the reliability of DTNs show that the introduced infrastructure in DTNs could dramatically enhance wireless networks in terms of packet delivery ratio [63]. Obviously, it is viable to adopt an RSU-aided packet forwarding mechanism in VANETs, where RSUs are used to assist in forwarding packets. However, deploying infrastructure is very costly, preventing from making RSUs widely available, for example, in rural areas or in the early stage deployment phase of VANET. Thus, effectively deploying RSUs is crucial to improving packet forwarding efficiency in VANETs. Since people are involved in Vehicular DTNs, human factors, in particular human mobility, will affect the network characteristics. Therefore, Vehicular DTN can be regarded as one kind of vehicular social networks.

Heavy traffic is a common occurrence in some areas on the roads, for example, at intersections, Taxi loading/unloading areas. Despite dynamics of traffic flow, traffic pattern is relatively static in an area. For example, during morning rush hours, overwhelming traffic can be observed inside a certain area, such as, in the downtown area because of its core focus on business. In other words, the area becomes a popular social place for vehicles to connect to each other. Obviously, if a roadside infrastructure is deployed in these high traffic areas and then used to assist in forwarding data packets in vehicular (social) DTNs, reliability in vehicular communications can be dramatically improved with incurred costs under control. Furthermore, security and privacy issue is crucial to the full adoption of any networks, but security and privacy issues in vehicular DTNs have been subject to little attention.

Based on the above observations, in this chapter, we propose a novel Social-based PRivacy-preserving packet forwardING (SPRING) protocol for vehicular DTNs [24]. The proposed SPRING protocol is characterized by deploying RSUs at high social intersections to assist in packet forwarding between vehicles by temporarily storing packets through V-2-I communication during the period when the proper next-hop vehicles of these packets are not available. With such kind of RSU assistance, the probability of packet drop is reduced and as a result a high reliability of packet forwarding in vehicular DTNs can be achieved. Specifically, the contributions of this chapter are threefold.

First, we heuristically define the *social degree* of intersections in vehicular DTNs. Based on the social information, we place RSUs at these *high social* intersections. To the best of our knowledge, this is the first attempt to investigate social-based RSU deployment in vehicular DTNs.

Second, we propose the SPRING protocol, a social-based privacy-preserving packet forwarding protocol for vehicular DTNs. In SPRING, because the stationary RSUs are deployed at high social intersections, a large number of vehicles will pass by these RSUs. Then, RSUs can provide tremendous assistance in temporarily storing some packets and helping packet forwarding to achieve a high transmission reliability. In addition, SPRING can also achieve conditional privacy preservation and resist most attacks existing in vehicular DTNs, such as packet analysis attack, packet tracing attack, and black (grey) hole attacks [64], that are crucial to success of such networks.

Third, we develop a simulator to show the substantial improvement of the SPRING protocol in terms of high reliability, resistance to packet tracing attack, and black (grey) hole attacks. The simulation results demonstrate its effectiveness and security.

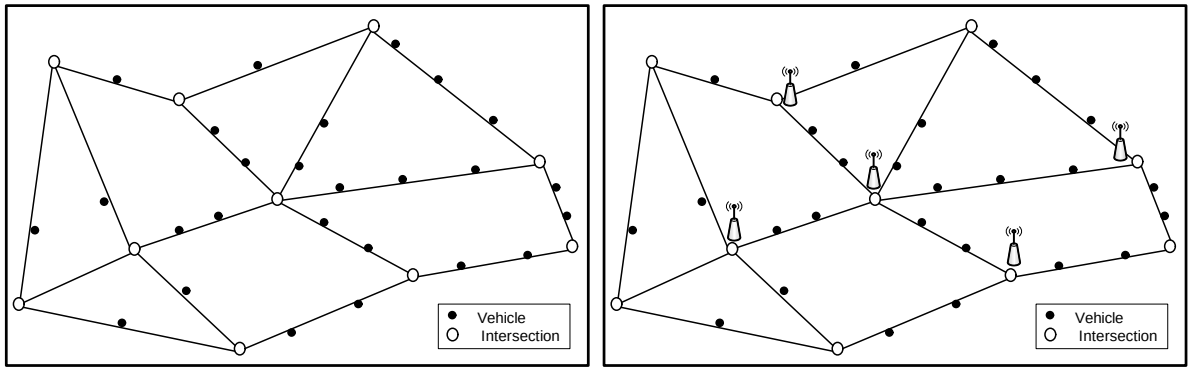
The remainder of this chapter is organized as follows. In Section 3.2, we formalize the network, node and threat models and identify our design goal. Then, we present the SPRING protocol in Section 3.3, followed by the security analysis and performance evaluation in Section 3.4 and Section 3.5, respectively. We also review related works in Section 3.6. Finally, we draw our summary in Section 3.7.

3.2 Models and Design Goal

In this section, we formalize the network model, node model and threat model, and identify our design goal.

3.2.1 Random Graph-Based Network Model

Consider a large number of vehicles $\mathcal{V} = \{v_1, v_2, \cdots\}$ moving around in a city by following map-based shortest path routing algorithm. Then, a vehicular DTN can be represented as a directed random graph $\mathcal{G} = (\mathcal{V}^*, \mathcal{E})$, as shown in Fig. 3.1(a), where $\mathcal{V}^*$ is a union between the set of vehicle nodes $\mathcal{V}$ and a set of intersection nodes $\mathcal{C} = \{c_1, c_2, \cdots\}$, i.e., $\mathcal{V}^* = \mathcal{V} \cup \mathcal{C}$, and $\mathcal{E}$ is the set of directed random edges between any intersections $c_i, c_j \in \mathcal{C}$, where $i \neq j$. For any edge $e_{ij} \in \mathcal{E}$ from c_i to c_j, we denote the flow of e_{ij} as $F(e_{ij}) = \sigma_{ij} \cdot \lambda_{ij}$, where $\sigma_{ij} = 1$ if c_i, c_j are connected by a direct road (i.e., no intermediate intersection between c_i and c_j), and 0 otherwise; λ_{ij} is the Poisson arrival rate of the road $(c_i \rightarrow c_j)$ if $\sigma_{ij} = 1$ and assuming the arrival follows the Poisson distribution, which can realistically capture the average number of vehicles passing from c_i to c_j during a unit of time. If $F(e_{ij}) = 0$, the edge e_{ij} does not exist. Note that in reality, $\sigma_{ij} = \sigma_{ji}$, while λ_{ij} may differ from λ_{ji}.



(a) (b)

Figure 3.1: Vehicular DTN model with social-based RSU deployment

In-Degree of interaction vertex $c_i \in \mathcal{C}$ is the number of roads with c_i as their terminal vertex, and is denoted as $KI_i = \sum_{j \in \mathcal{C}} \sigma_{ji}$. *Out-Degree* of interaction vertex $c_i \in \mathcal{C}$ is the number of roads with vertex c_i as their initial vertex, and is denoted as $KO_i = \sum_{j \in \mathcal{C}} \sigma_{ij}$. Because $\sigma_{ji} = \sigma_{ij}$, we have $KI_i = KO_i$. Generally, in a directed graph, both *in-degree* and *out-degree* of a vertex can capture its impact in the whole graph. However, in the defined random directed graph, the impact of an interaction vertex c_i is contingent upon the number of contacts between c_i and other vehicle nodes in $\mathcal{V}$. Therefore, the *Social Degree* of interaction vertex is introduced.

Social Degree of an intersection vertex $c_i \in \mathcal{C}$ is defined as

$$SD_i = \frac{\sum_{v_j \in \mathcal{V}} \delta_j(c_i)}{\sum_{v_j \in \mathcal{V}} \delta_j} \tag{3.1}$$

where δ_j is the number of shortest paths that a vehicle node $v_j \in \mathcal{V}$ drives during a unit of time, and $\delta_j(c_i)$ is the number of shortest paths that passes through the intersection vertex c_i.

In the defined random graph, it is easy to show that, for any intersection vertex $c_i \in \mathcal{C}$, $\sum_{c_j \in \mathcal{C}, \sigma_{ji}=1} \lambda_{ji} = \sum_{c_j \in \mathcal{C}, \sigma_{ij}=1} \lambda_{ij}$, although $\lambda_{ji} \neq \lambda_{ij}$ for some c_j. With this observation, the flow of the intersection vertex c_i can be defined as $F(c_i) = \sum_{c_j \in \mathcal{C}} F(e_{ji}) = \sum_{c_j \in \mathcal{C}} \sigma_{ji} \cdot \lambda_{ji}$ Because all vehicles $V = \{v_1, v_2, \cdots\}$ follow the map-based shortest path routing, the *social degree* SD_i of c_i can be rewritten as

$$SD_i = \frac{F(c_i)}{\sum_{v_j \in \mathcal{V}} \delta_j} = \frac{\sum_{c_j \in \mathcal{C}} \sigma_{ji} \cdot \lambda_{ji}}{\sum_{v_j \in \mathcal{V}} \delta_j} \tag{3.2}$$

Deployment of RSUs. Let ST denote the *social threshold* of a given random directed graph $\mathcal{G} = (\mathcal{V}^*, \mathcal{E})$. We choose a set of *high social* intersection vertexes as follows

$$HS = \{c_i \in \mathcal{C} | SD_i \geq ST\} \tag{3.3}$$

Note that, by adjusting the social threshold ST, we can determine the cardinality of HS. After the set HS is determined, we place an RSU R_i at each intersection $c_i \in HS$, as shown in Fig. 3.1(b). Then, each RSU has high social capability and can effectively assist vehicles to store-forward packets in vehicular DTNs.

3.2.2 Node Model

Vehicular DTNs, as distinct from general DTNs, are characterized by two kinds of DTN nodes, i.e., vehicles and RSUs, each kind of which has unique characteristics.

- *Vehicles:* Apart from the mobility, each vehicle node driven by people is also resource-constrained, i.e., buffer constraints. In general, a vehicle node will help forwarding the packets when it has available storage. However, once the storage is insufficient, the vehicle node no longer serves the relay node to help forwarding.
- *Roadside Units (RSUs):* Different from the vehicle node, each RSU node is stationary but has huge storage capacity. Once it is deployed at some intersection, each RSU node can temporarily help store some bundle packets till passing-by vehicle nodes carry them close to their destinations. However, since each RSU is costly, it is impractical to erect RSU nodes at all intersections, especially at the early deployment of VANETs. Therefore, as discussed in the network model, only a small number of RSU nodes will place at some *high social* intersections.

Let T_R and T_V, where $\mathsf{T}_R > \mathsf{T}_V$, be the transmission ranges of RSU and vehicle nodes, respectively. Then, the wireless interfaces between vehicle nodes are bidirectional, i.e., if v_i hears the transmission of v_j, then v_j is also able to hear v_i. However, the wireless interfaces between vehicle node and RSU node are usually unidirectional unless they are very close to each other, as shown in Fig. 3.2. Therefore, packets in vehicular DTNs will be i) either stored-carried by the vehicle nodes or stored in RSU nodes if no other node is encountered; ii) forwarded when other node is encountered.

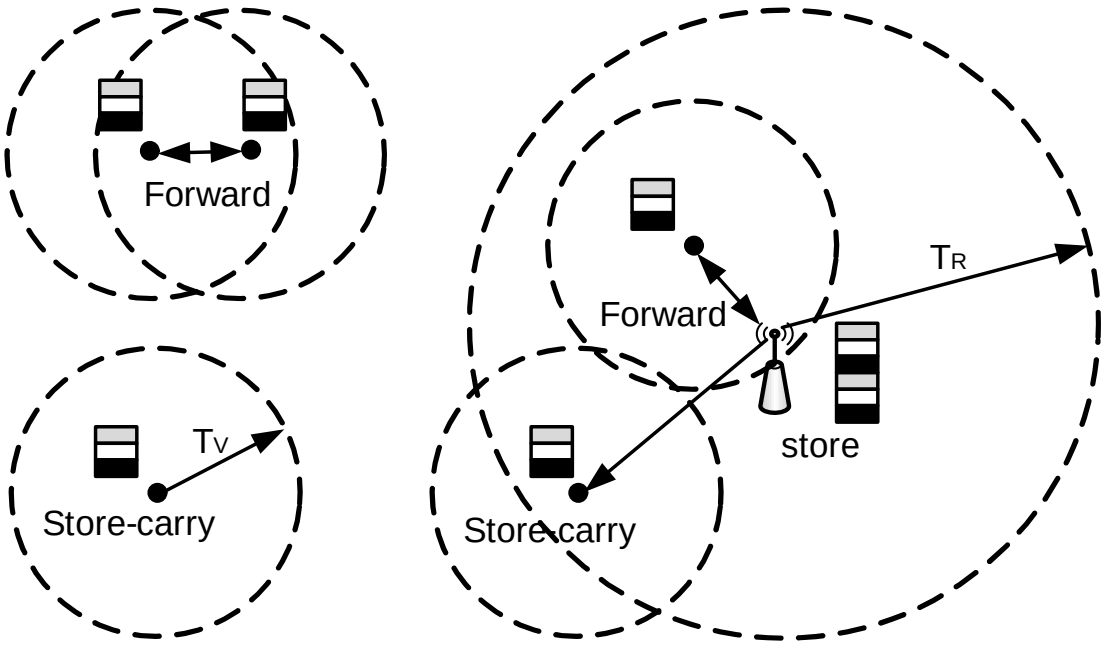


Figure 3.2: Store-Carry-Forward in Vehicle DTNs

3.2.3 Threat Model

In our threat model, RSU nodes are trustable, and non-compromisable. However, a small fraction of vehicle nodes may be compromised. We consider a global external adversary $\mathcal{A}$ with limited control capability, where

- *Global* shows the adversary $\mathcal{A}$ has full traffic information of the whole vehicular DTN;
- *External* denotes the adversary $\mathcal{A}$ generally can only capture the communications between DTN nodes, but has no idea about the internal information stored in these nodes.
- *Limited control* means the adversary $\mathcal{A}$ can control a very small fraction, (i.e., less than 0.1), of vehicle nodes to launch some kinds of active attacks. (Note that, limited controlling a small fraction of vehicles nodes does not help the adversary $\mathcal{A}$ to gain other vehicles' key materials.)

In specific, we consider the adversary $\mathcal{A}$ can launch the following attacks to either subvert privacy or degrade the performance of the whole vehicle DTN.

- *Packet analysis attack:* After eavesdropping a packet, the adversary $\mathcal{A}$ tries to identify the source identity by analyzing the packet, i.e., recover the packet content and infer the source.

- *Packet tracing attack:* The adversary $\mathcal{A}$ eavesdrops the transmission of a single packet as it traverses around the vehicular DTN. In such a way, the source and destination locations of the packet can be traced. Note that the adversary $\mathcal{A}$ does not need to recover the packet content to infer the source and destination locations of the flow.

- *Black hole attack:* In vehicular DTN, the adversary $\mathcal{A}$ first lures packets by claiming that it can help forward them close to their destinations. However, all packets are actually dropped by the adversary $\mathcal{A}$. Clearly, the black hole attack is one kind of Denial of Service (DoS) attacks, which can largely degrade the performance of the whole vehicular DTN, especially when the adversary $\mathcal{A}$ controls some compromised vehicle DTN nodes to launch the attack.

- *Grey hole attack:* Grey hole attack is a variant of black hole attack in vehicular DTN, where the adversary $\mathcal{A}$ selectively forwards some packets but not all packets. This kind of attack is hardly to detect because it is indistinguishable from the normal packet dropping event when the vehicular DTN is poor-connected.

3.2.4 Design Goal

Based on the above models, our design goal in this work is to develop a social-based privacy-preserving packet forwarding protocol for vehicular DTNs. Specifically, the following three desirable objectives will be achieved.

- *Optimizing vehicular DTN with RSU assistance.* In a large vehicular DTN, when the vehicle density is sparse, the contacting opportunity of vehicle DTN nodes is low, which will incur the low delivery ratio in vehicular DTN, especially when the single-copy technique is adopted. In order to prevent the overall performance degradation, we introduce *high social* RSU deployment into vehicular DTNs. Because RSUs have huge storage capabilities, they can temporarily store packets when the next-hop vehicle node is not available. In addition, the *high social* capacities of these RSUs can ensure they can contact many more vehicles in a very short time. As a result, the delay due to RSU temporary storing can be confined, and the performance of the vehicular DTN is optimized.

- *Resisting privacy-related attacks on vehicle DTN nodes.* Because vehicular DTN is usually implemented in civilian scenarios, where the locations of vehicle nodes are tightly related to the citizens who are driving them. If the vehicular DTN discloses the privacy information of the citizens, i.e., identity and location privacy, vehicular DTN cannot be widely accepted by the public. Therefore, the citizens' privacy must be protected in order for wide acceptance to the public.

- *Achieving conditional privacy preservation:* Following the threat model discussed earlier, if the adversary $\mathcal{A}$ launches the black/grey hole attacks by controlling a small fraction of compromised vehicle nodes, these attacks are hard to resist, because these compromised nodes have their valid key materials. Therefore, the *absolute privacy preservation* is insufficient, and the *conditional privacy preservation* is expected. In specific, once a compromised vehicle node launches the attack, a trust authority (TA) should have ability to identity the compromised node and punish it under the applicable law.

3.3 Proposed SPRING Protocol

In this section, we present our social-based privacy-preserving packet forwarding (SPRING) protocol for vehicular DTNs, which mainly consists of the following two phases: system initialization phase and opportunistic RSU-aided packet forwarding phase.

System Initialization Phase

Based on the system requirements, the following steps are performed to bootstrap the whole system:

- Assume that there exists a TA in the system, which initializes the whole system. Given the security parameter k, the composite bilinear parameters $(n, g, \mathbb{G}, \mathbb{G}_T, e)$ are first generated by running $\mathcal{CG}en(k)$. Then, the TA chooses two elements (g, u) of $\mathbb{G}$, one generator h of $\mathbb{G}_q$, two random exponents $\alpha, a \in \mathbb{Z}_n^*$, a collision-resistant hash function $H : \{0,1\}^* \rightarrow \mathbb{Z}_n^*$, and a secure symmetric encryption algorithm $Enc()$. After these, the TA sets the *master key* (g^α, a, q) and the system public parameters $params = (\mathbb{G}, \mathbb{G}_T, e, n, g, u, e(g,g)^\alpha, A = g^a, h, H, Enc())$.

- For each vehicle $v_i \in \mathcal{V} = \{v_1, v_2, \cdots\}$, the TA chooses a random number $x_i \in \mathbb{Z}_n^*$ such that $a + x_i \neq 0$ as a secret key, and computes $A_i = g^{\frac{1}{a+x_i}} \in \mathbb{G}$, $B_i = \mathbf{g}^{x_i} \in \mathbb{G}_T$,

where $\mathbf{g} = e(g, g)$. Then, TA sets (x_i, A_i) as v_i's anonymous credential and stores duplet $(v_i, g^{x_i q})$ in the tracing list. In addition, TA authorizes $B_i \in \mathbb{G}_T$ as the public key of v_i.

- For a specific area, the TA first investigates a set of intersection nodes $\mathcal{C} = \{c_1, c_2, \cdots\}$ and computes the social degree SD_j of each intersection c_j. Then, by setting a social threshold ST, the TA derives a set of *high social* intersection nodes $HS = \{c_j \in \mathcal{C} | SD_j \geq ST\}$. At each *high social* intersection, TA places an RSU, authorizes a secret key $x_j \in \mathbb{Z}_n^*$ and the corresponding public key $C_j = \mathbf{g}^{x_j} \in \mathbb{G}_T$ for the RSU. Note that the public key C_j here is associated with the intersection c_i attested with a certificate issued by the TA.

Opportunistic RSU-aided packet forwarding phase

Suppose that a source node v_1 wants to send a sensitive message $m \in \mathbb{G}_T$ to the destination node v_2, where the location L_2 of v_2 is assumed stationary and known by v_1. To fulfill such sensitive packet forwarding in vehicular DTN, the following steps will be executed.

Step 1. The source node v_1 first uses the destination node v_2's public key $B_2 = \mathbf{g}^{x_2}$ and two random numbers $k_0, k_1 \in \mathbb{Z}_n^*$ to encrypt the message m as

$$M = (\alpha_0, \beta_0, \alpha_1, \beta_1) = (m \cdot B_2^{k_0}, \mathbf{g}^{k_0}, B_2^{k_1}, \mathbf{g}^{k_1}) \tag{3.4}$$

the latter part of ciphertext (α_1, β_1) will be used for any future possible re-encryption on M by RSUs in order to build up a mix network [65].

Step 2. When a passing-by vehicle node v_i is willing to help forwarding the message M, the source node v_1 and node v_i execute the following interactive operations.

- The passing-by vehicle v_i first gets the current timestamp T_i and computes $\mathbf{g}^x$, where x is randomly chosen from $\mathbb{Z}_n^*$. Then, v_i uses the CPPA technique in Chapter 2.3.3 to construct $\mathsf{CPPA}(T_i || \mathbf{g}^x)$ and sends it to the source node v_1.

- After checking the validity of $\mathsf{CPPA}(T_i || \mathbf{g}^x)$, v_1 chooses another random number $y \in \mathbb{Z}_q^*$, encrypts the destination location L_2 as $D = (\alpha_2, \beta_2) = (L_2 \cdot \mathbf{g}^{xy}, \mathbf{g}^y)$, and sends $M || D$ back to the passing-by vehicle v_i.

- After recovering the destination location L_2 from $D = (\alpha_2, \beta_2)$ as $\frac{\alpha_2}{{\beta_2}^x} = \frac{L_2 \cdot \mathbf{g}^{xy}}{\mathbf{g}^{xy}} = L_2$, the passing-by vehicle v_i tries its best to help carrying the message M closer to the destination.

Step 3. After the vehicle v_i carries the message M for a period of time, the destination L_2 is no longer on the vehicle v_i's way. Then, v_i invokes the Algorithm 1 to forward the message M to a proper next-hop node. Because RSU's transmission range T_R is larger than vehicle's transmission range T_V, if there exists an RSU at some nearby intersection c_i on its way, the vehicle v_i can first detect it. Then, v_i will drive close to the RSU and forward the message M with V-2-I communication as follows.

Algorithm 1 Packet forwarding in vehicular DTN

procedure PACKET FORWARDING
    when the vehicle node v_i thinks it cannot help carrying message packet M any more, it will first set a *holding time to wait next-hope node* (T_h) and try to forward M to the next-hop DTN node within T_h
    if v_i detects an RSU located in a nearby intersection **then**
        v_i will drive close and forward the message M to the RSU
    else if v_i detects an available vehicle node v_j nearby **then**
        v_i will forward the message M to v_j
    else if no next-hop node is available **then**
        the message M has to be discarded
    end if
end procedure

Vehicle-to-RSU (V-2-I) communication: Considering an average vehicle velocity of $\mathsf{V} = 60$ km/h (≈ 16.6 m/s) and a transmission range $\mathsf{T}_V = 300$ m, the communication interval (CI) for the vehicle v_i and a stationary RSU located at intersection c_i, as shown in Fig. 3.3, can be roughly calculated by $\mathsf{CI} = \frac{2 \cdot \mathsf{T}_V}{\mathsf{V}} = \frac{2 \times 300}{16.6} = 36.1$ s. Note that, if there are traffic lights at the intersection, the CI can be longer than 36.1 s. Then, within the CI, the V-2-I communication is executed by the follow steps:

- The RSU periodically broadcasts the beacon message within its coverage. Concretely, the RSU first chooses a random number $a \in \mathbb{Z}_n^*$, signs the current timestamp T_j as (α_3, β_3), where
$$\alpha_3 = \mathbf{g}^a, \beta_3 = a + x_j \cdot H(\alpha_3, T_j) \bmod n \tag{3.5}$$
and broadcasts the beacon information (T_j, α_3, β_3).
- After receiving the beacon message (T_j, α_3, β_3) at time T_j', the vehicle v_i checks whether $T_j' - T_j \leq \Delta T$. If it does not hold, v_i believes it is a replay attack and neglect it. Otherwise, v_i checks $\mathbf{g}^{\beta_3} \stackrel{?}{=} \alpha_3 \cdot C_j^{H(\alpha_3, T_j)}$ with the RSU's public key

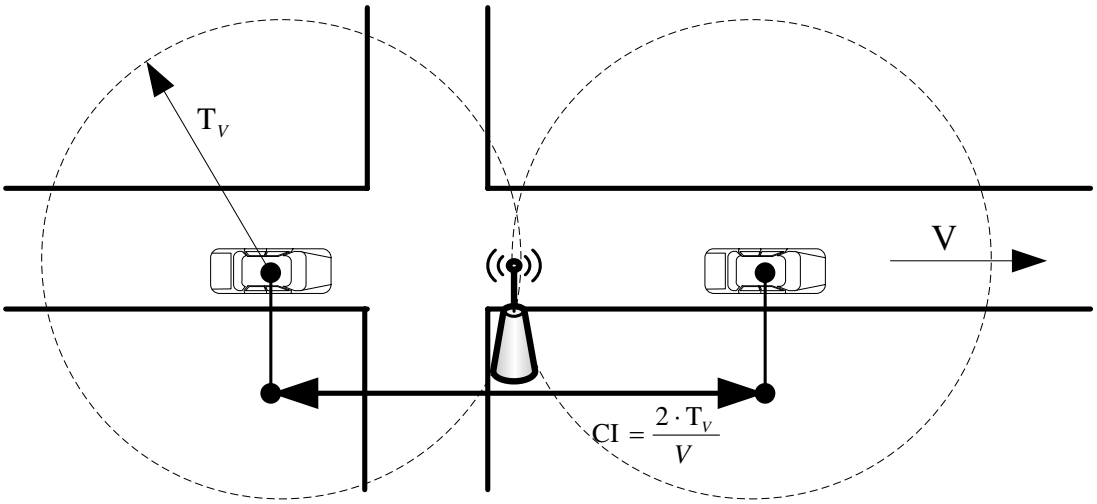


Figure 3.3: Vehicle-to-RSU (V-2-I) communication

$C_j = \mathbf{g}^{x_j}$. If it holds, the beacon message is accepted. The correctness and security can be referred to [66].

- The vehicle v_i chooses a random number $b \in \mathbb{Z}_n^*$ to encrypt the destination location L_2 as $D = (\alpha_2, \beta_2) = (L_2 \cdot \mathbf{g}^{ab}, \mathbf{g}^b)$ and sends M together with D to the RSU.

- Upon receiving $M||D$, the RSU first recovers L_2 from $\frac{\alpha_2}{{\beta_2}^a}$, and chooses $k_0', k_1' \in \mathbb{Z}_q^*$ to re-encrypt the message $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ as

$$\begin{aligned} M &= (\alpha_0 \cdot {\alpha_1}^{k_0'}, \beta_0 \cdot {\beta_1}^{k_0'}, {\alpha_1}^{k_1'}, {\beta_1}^{k_1'}) \\ &= (m \cdot B_2^{k_0 + k_1 \cdot k_0'}, \mathbf{g}^{k_0 + k_1 \cdot k_0'}, B_2^{k_1 k_1'}, \mathbf{g}^{k_1 k_1'}) \end{aligned} \tag{3.6}$$

  Then, the RSU stores $L_2||M$ and waits for the proper next-hop vehicle to carry it.

- Because the RSU is deployed at a *high social* intersection, the RSU may have already stored many messages. Therefore, if the vehicle v_i is willing to carry some of them to their destinations or other high social RSUs closer to their destinations, it will use the CPPA technique $\mathsf{CPPA}(T_i||\mathbf{g}^x)$ to again anonymously authenticate itself, where T_i is a new timestamp here. Then, after checking the validity of $\mathsf{CPPA}(T_i||\mathbf{g}^x)$, the RSU, like the source node v_1, will forward some messages to v_i. We assume each message packet is 2 M and the packet transmission bitrate is 5 Mbps. (Note that, the 802.11p physical layer offers different bitrates, ranging from 3 to 27 Mbps, from which we

can choose [67].) By deducing the cost around 2 seconds used for authentication between vehicle and RSU, we have $(\mathsf{CI} - 2) \cdot 5$ Mbps/2 M $= (36.1 - 2) \cdot 5/2 \approx 85$. This result shows that almost 85 message packets can be forwarded between V-2-I communications.

Vehicle-to-Vehicle (V-2-V) communication: If no nearby RSU is found but an available vehicle v_j is passing by, then v_i will forward the message M to v_j with the V-2-V communications. The concrete interactive operations are the same as those between v_1 and v_i above. Consider both v_i and v_j having the same velocity $\mathsf{V} = 60$ km/h (≈ 16.6 m/s) and transmission range $\mathsf{T}_V = 300$ m, the communication interval (CI) between them on a straight road could become $\mathsf{CI} = \frac{2 \cdot \mathsf{T}_V}{2 \cdot \mathsf{V}} = \frac{2 \times 300}{2 \times 16.6} = 18.0$ s. This calculation indicates that forwarding between vehicle and vehicle should be fulfilled within 18.0 s. Because vehicle has no huge storage and is also not as *social* as RSU in our network model, then the 18-second CI can fit for fewer packet forwarding between V-2-V communications.

Packet dropping case: As show in Algorithm 1, if none of RSU or vehicle is found as an available next-hop node in vehicular DTN, the message M has to be dropped.

Step 4. If the message $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ is not dropped, it will eventually be relayed to the destination node v_2 at location L_2. Then, the destination node v_2 can use its secret key x_2 to recover m by the following operations:

$$m_0 = \frac{\alpha_0}{{\beta_0}^{x_2}}, \quad m_1 = \frac{\alpha_1}{{\beta_1}^{x_2}} \tag{3.7}$$

If $m_0 \neq 1$ and $m_1 = 1$, the destination node accepts m_0 as the valid plaintext m; otherwise, the message $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ is invalid and will be rejected.

Correctness. Suppose $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ is temporarily stored at RSU only once, then it has the form of $(m \cdot B_2^{k_0 + k_1 \cdot k_0'}, \mathbf{g}^{k_0 + k_1 \cdot k_0'}, B_2^{k_1 k_1'}, \mathbf{g}^{k_1 k_1'})$. Thus,

$$\frac{\alpha_0}{{\beta_0}^{x_2}} = \frac{m \cdot B_2^{k_0 + k_1 \cdot k_0'}}{\left(\mathbf{g}^{k_0 + k_1 \cdot k_0'}\right)^{x_2}} = m; \frac{\alpha_1}{{\beta_1}^{x_2}} = \frac{B_2^{k_1 k_1'}}{\left(\mathbf{g}^{k_1 k_1'}\right)^{x_2}} = 1 \tag{3.8}$$

If $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ was stored in RSUs more than once, with simple deduction, the correctness on recovering the plaintext m can also be checked. The details on this universal re-encryption technique can be referred to [65].

3.4 Security Analysis

In this section, we analyze the security properties of the proposed SPRING protocol. In specific, following the threat model discussed earlier, our analysis will focus on how the

proposed SPRING protocol can resist to the packet analysis attack, packet tracing attack, and the black (grey) hole attack, respectively.

3.4.1 Resilience to Packet Analysis Attack

In the proposed SPRING protocol, the source node v_1 has encrypted the sensitive message m into $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$. Without knowing the destination node v_2's secret key x_2, the adversary cannot recover m from packet analysis. In addition, because the CPPA anonymous authentication is adopted, no identity information will be disclosed. Therefore, the proposed SPRING protocol can resist the packet analysis attack.

3.4.2 Resilience to Packet Tracing Attack

First, we consider no vehicle nodes controlled by the adversary $\mathcal{A}$ participate in the packet forwarding. Then, the capability of the adversary is limited to eavesdrop the interactions among V-2-V communications and V-2-I communications. Because the destination is encrypted in each interaction, the adversary cannot know the destination information. In addition, in the proposed SPRING protocol, when an RSU receives $M = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ from a vehicle, it will use the universal re-encrypt technique to convert M into another form $M = (\alpha_0 \cdot {\alpha_1}^{k'_0}, \beta_0 \cdot {\beta_1}^{k'_0}, {\alpha_1}^{k'_1}, {\beta_1}^{k'_1})$. Since k'_0 and k'_1 are randomly chosen from $\mathbb{Z}_q^*$, $(\alpha_0, \beta_0, \alpha_1, \beta_1)$ and $(\alpha_0 \cdot {\alpha_1}^{k'_0}, \beta_0 \cdot {\beta_1}^{k'_0}, {\alpha_1}^{k'_1}, {\beta_1}^{k'_1})$ are unlinkable. More importantly, since the RSU is located at a *high social* intersection, a large number of vehicles pass by and many packets will be temporally stored at the RSU, the RSU can naturally serve as a mix server, as shown in Fig. 3.4. Then, for a specific message packet, only if it had been temporally stored in a *high social* RSU at least once, the adversary cannot trace it only by eavesdropping.

Second, we consider some vehicles controlled by the adversary $\mathcal{A}$ participate in the packet forwarding. Clearly, in this case, the destination information is disclosed to the adversary. However, in the proposed SPRING protocol, 1) the source node v_1 is indistinguishable from other intermediate nodes during the V-2-V and V-2-I communications; 2) the real identity will not be disclosed in the CPPA anonymous authentication, the adversary still cannot determine who is the indeed source node. By summarizing the above, the proposed SPRING protocol can resist the packet tracing attack.

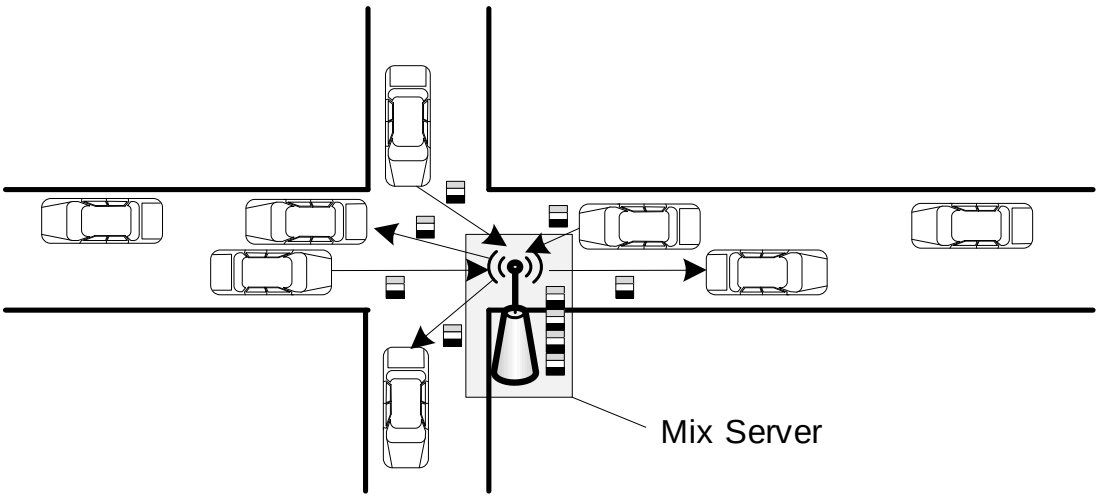


Figure 3.4: High social RSU serves as a mix server in vehicular DTNs

3.4.3 Resilience to Black (Grey) Hole Attack

Because of the CPPA anonymous authentication, the black (grey) attacks launched by the external adversary can be efficiently resisted in the proposed SPRING protocol. However, once the vehicle nodes controlled by the adversary $\mathcal{A}$ launch the black (grey) attacks, because they know the valid key materials, and at the same time, the CPPA anonymous authentication also makes them unlinkable, the black (grey) attacks in this case are serious and hard to resist. Fortunately, the privacy preservation provided by the CPPA technique is conditional, which provides the second line of defense. Once the *witness* $\mathsf{CPPA}(T_i||\mathbf{g}^x)$ is submitted to the TA, the TA can reveal the real identity by using the conditional tracking algorithm in Chapter 2.3.3. Thus, if a message packet does not reach the destination L_2, then with the chain tracking policy shown in Fig. 3.5, each next-hop node participating in such packet's forwarding can be identified by the TA with the *witness* $\mathsf{CPPA}(T_i||\mathbf{g}^x)||L_2$ provided by the current node, where the destination L_2 is used to assist the current node to identify the involved next-hop node among many next-hop nodes. If the current node cannot provide any *witness*, it becomes suspicious.

As shown in Algorithm 1, if no next-hop node is available, the message packet can also be dropped. However, this packet dropping event is less than the event caused by the packet dropping due to black (grey) hole attacks. Therefore, with the detection process in Algorithm 2, the vehicle nodes who launched the black (grey) hole attacks can be identified. Note that, in the Algorithm 2, the thresholds T_B, T_G must be carefully defined. Otherwise,

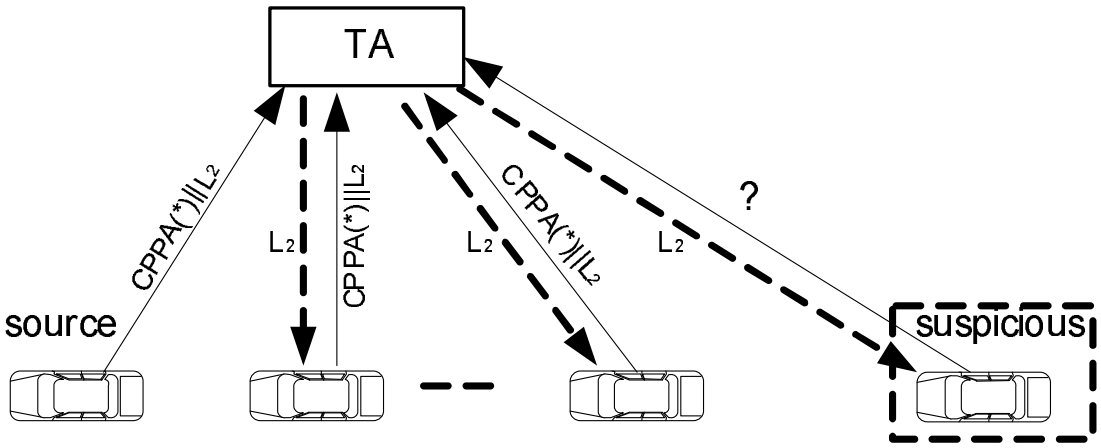


Figure 3.5: Detect the suspicious vehicle nodes with chain tracking

false positive and false negative could be high.

3.5 Performance Evaluation

In this section, we study the average-case performance of the proposed SPRING protocol, using a custom simulator built in Java. The performance metrics used in the evaluation are average *delivery ratio* (DR) and packet *average delay* (AD), where the DR is defined as the average ratio of the packets successfully delivered to the destinations with respect to those generated by the sources within a given time period, and the AD is defined as the average time between when a packet is generated at some source and when it is successfully delivered to the destination. In addition, following the earlier design goal, we also evaluate the resistance to packet tracing attack and detection of black (grey) hole attacks in the simulations.

3.5.1 Simulation Setup

To simulate a sparse vehicular DTN, 50 vehicle nodes with transmission radius of 300 meters are first deployed to cover an interest Kitchener-Waterloo (K-W) region of $6,000$ m $\times$ $15,000$ m, as shown in Fig. 3.6. In addition, 12 intersections are chosen as the candidates for RSU deployment in the region.

Algorithm 2 Detection of black (grey) hole attacks

procedure BLACKGREYHOLEATTACKDECTION
    With the chain tracking in Fig. 3.5, the TA can obtain each vehicle node v_i's packet dropping number, denoted as X_i.
    Compute the mean $\overline{X}$ of all vehicle nodes $\mathcal{V} = \{v_1, v_2, \cdots\}$ as $\overline{X} = \frac{1}{|\mathcal{V}|}\sum_{i=1}^{|\mathcal{V}|} X_i$, where $\mathcal{V}$ is the cardinality of $\mathcal{V}$.
    Compute the distance of each X_i to the mean $\overline{X}$ as $d(X_i) = |X_i - \overline{X}|$.
    Define the thresholds T_B, T_G for black hole attack and grey hole attack, respectively.
    for each vehicle node $v_i \in \mathcal{V} = \{v_1, v_2, \cdots\}$ **do**
        if $d(X_i) > T_B$ **then**
            v_i is considered as a black hole attacker.
        else if $d(X_i) > T_G$ **then**
            v_i is considered as a grey hole attacker.
        else
            v_i is considered as a normal vehicle node.
        end if
    end for
end procedure

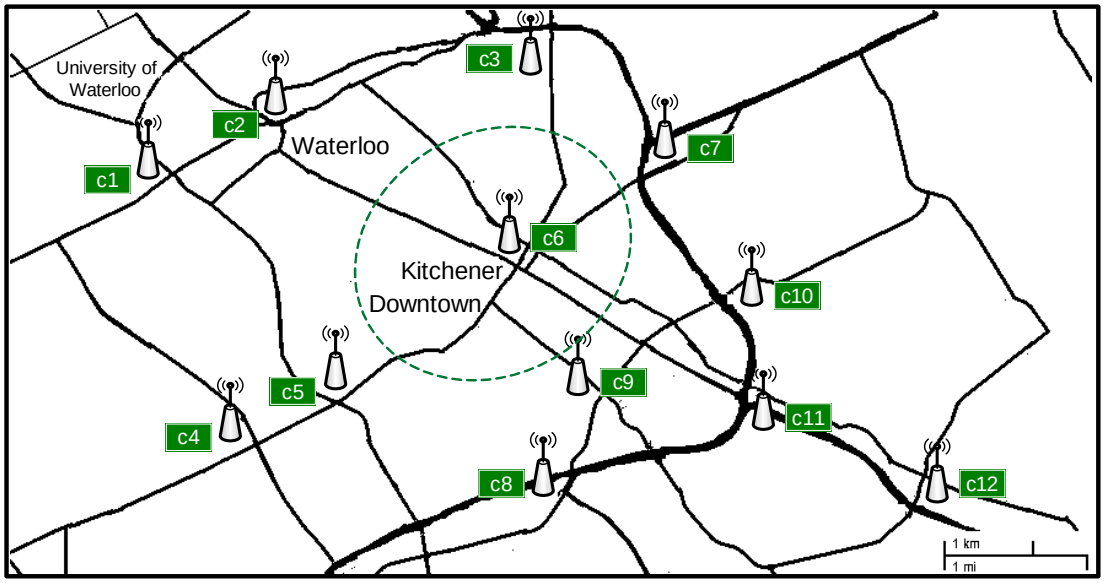


Figure 3.6: Kitchener-Waterloo (K-W) region considered for simulation

Mobility model. In vehicular DTN, the performance of packet forwarding is highly contingent upon the mobility of the vehicle nodes. Because the vehicle nodes are mostly driven by the citizens, modeling the mobility patterns of citizens in a specific area (i.e., K-W region) can achieve a relatively accurate performance evaluation. Let s_0 denote the state that a person is located at any spot in the K-W region, and state s_1, where $\mathsf{s}_1 \subset \mathsf{s}_0$, denote the person is stationary at some spot in the downtown. A person stays at each state s_i, $i \in \{0, 1\}$, for 120 ± 60 seconds, and then chooses the next state. If the current state is s_0, s/he will choose s_1 as the next state with the probability $\rho = 0.5$ and s_0 with the probability $1 - \rho$. If the current state is s_1, s/he will choose s_0 with the probability 1. Once the next state is determined, the person will drive the vehicle to the destination by following the map-based shortest path routing with the velocity 60 km/h.

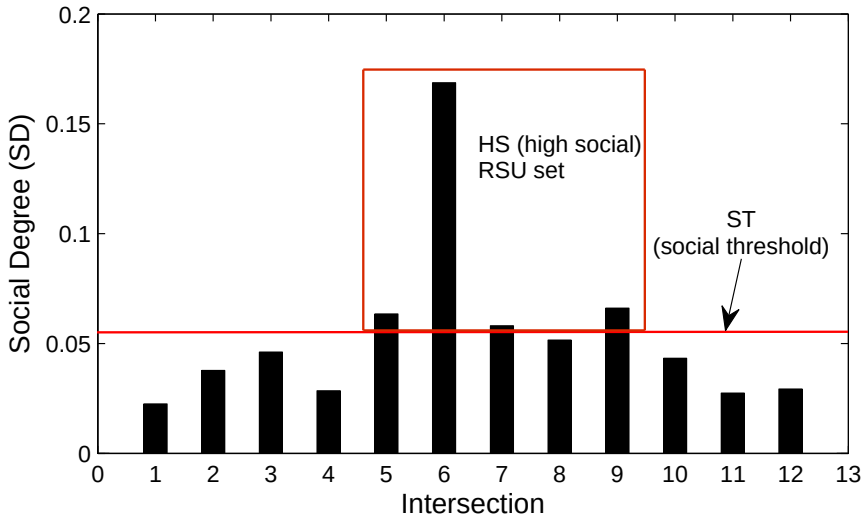


Figure 3.7: Selection of high social intersections

High-social intersection selection & RSU deployment. Because each vehicle node follows the map-based shortest path routing, we can use the simulations to statistically test the *social degree* of each intersection in Fig. 3.6. Based on the definition of *social degree* in Eq. (3.2), the duration for each simulation is set as one hour — a unit of time, and the results, as shown in Fig. 3.7, are averaged over $1,000$ runs. After setting a proper social threshold (ST), we can select *high-social* intersections $HS = \{c_5, c_6, c_7, c_9\}$, and place an RSU at each intersection $c_i \in HS$. To examine the outstanding performance of social-based RSU deployment in the proposed SPRING protocol, we compare it with random RSU deployment, complete RSU deployment and non RSU deployment in vehicular DTN. The detailed parameter settings are summarized in Table 3.1.

Table 3.1: Simulation Settings in SPRING

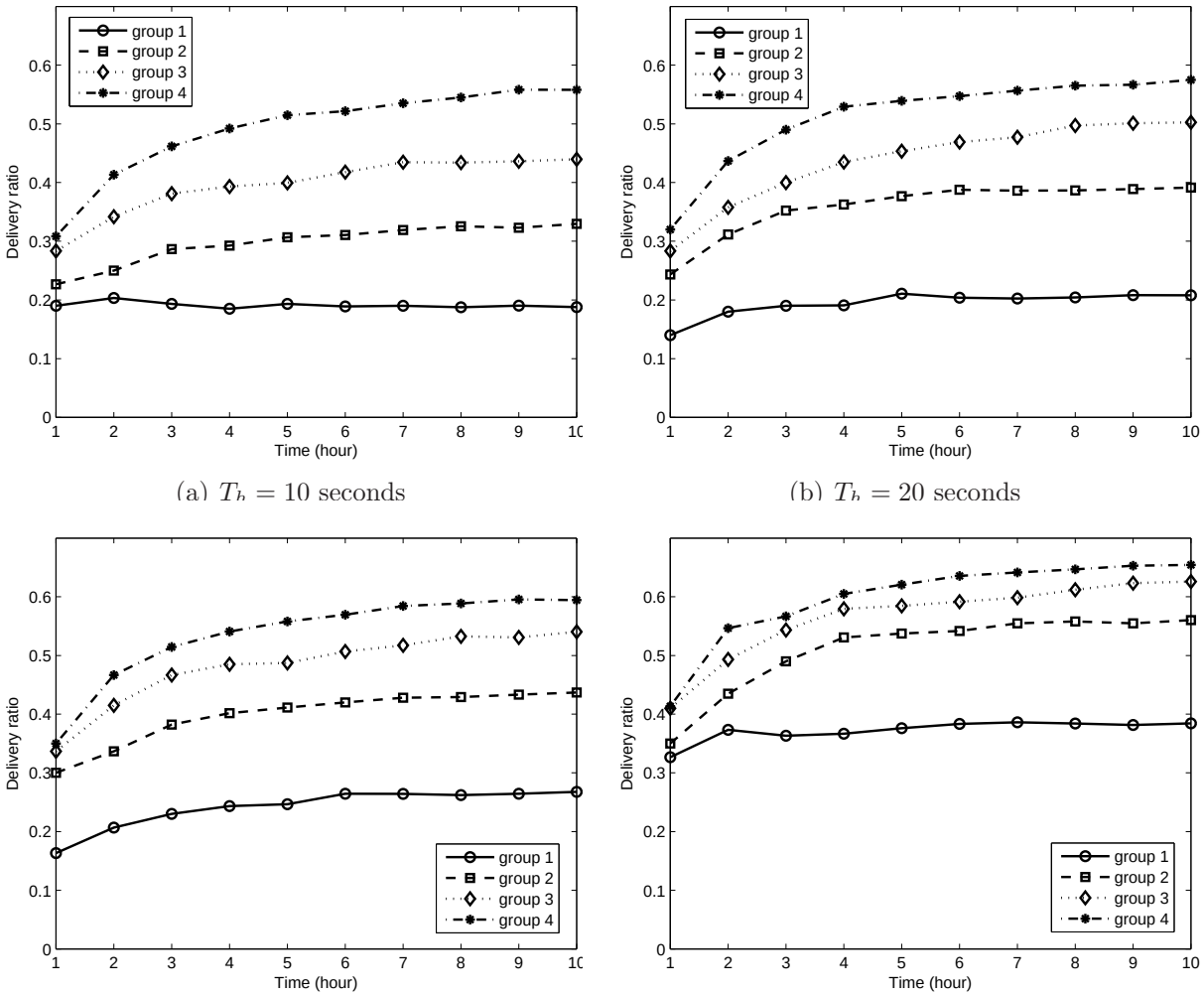
Parameter	Setting
Simulation	
duration time; area	10 hours; $6,000$ m $\times$ $15,000$ m
Vehicle node	
number; storage; velocity	50; 20 MB; 60 km/h
transmission radius; mobility model	300 m; map-based shortest path
holding time to wait next-hop node	$T_h = [10; 20; 40; 80]$ seconds
RSU	
number; storage; transmission radius	50; 10000 MB; 1000 m
group 1 (no deployment)	—
group 2 (random deployment)	$\{c_2, c_4, c_8, c_{11}\}$
group 3 (social-based deployment)	$HS = \{c_5, c_6, c_7, c_9\}$
group 4 (complete deployment)	$\{c_1, c_2, \cdots, c_{11}, c_{12}\}$
Message size; generation interval	2 ± 0.5 MB, 120 ± 20 seconds

In the following, we conduct the simulations with different parameter T_h and different RSU deployments. For each case, we run the simulation for 10 hours, and the average performance results over 20 runs are reported.

3.5.2 Simulation Results

Delivery Ratio & Average Delay

Fig. 3.8 shows the *delivery ratio* varies with the specified period from 1 hour to 10 hours. From the figure, we can see the delivery ratio in group 1 is lower than that in groups 2, 3 and 4. This observation validates that the V-2-V plus V-2-I based packet forwarding is more reliable than the pure V-2-V based forwarding in vehicular DTN. Comparing the delivery ratio in groups 2, 3 and 4, we can also observe that the more the RSUs are deployed, the higher the deliver ratio is; when the number of deployed RSUs are constrained, the social-based RSU deployment can achieve better deliver ratio than that with the random RSU deployment. In addition, Fig. 3.8 also shows that, when the parameter T_h increases, a vehicle node has more chances to contact next-hop DTN nodes (vehicle and RSU), and the delivery ratio will visibly increase.



(a) $T_h = 10$ seconds

(b) $T_h = 20$ seconds

(c) $T_h = 40$ seconds

(d) $T_h = 80$ seconds

Figure 3.8: Delivery ratio versus specified time period

Fig. 3.9 depicts the *average delay* within 10 hours in different RSU deployments. From the figure, we can see the average delay in groups 2, 3, and 4 is larger than that in group 1. As discussed above, the delivery ratio in pure V-2-V based forwarding is very low, and many packets will be dropped, while the delivery ratio in V-2-V plus V-2-I based forwarding is high, i.e., a larger number of successfully delivered packets can be temporarily stored in the RSUs for achieving high delivery ratio. Therefore, for some delay tolerant applications, high delivery ratio gained from some tolerant delay is an acceptable option. Because the RSUs deployed at high social intersections can contact more vehicle nodes, the packets stored in RSUs can be quickly forwarded. Thus, the average delay in social-based RSU deployment is slightly lower than that in random RSU deployment. Another interesting observation shown in Fig. 3.9 is that the average delay in social-based RSU deployment is also lower than that in complete RSU deployment. The reason is that when the RSUs are deployed at all intersections, many packets could be temporarily stored in some low-social RSUs. Then, although the delivery ratio is improved, it possibly takes a slightly long time to forward these packets to the next-hop vehicle nodes.

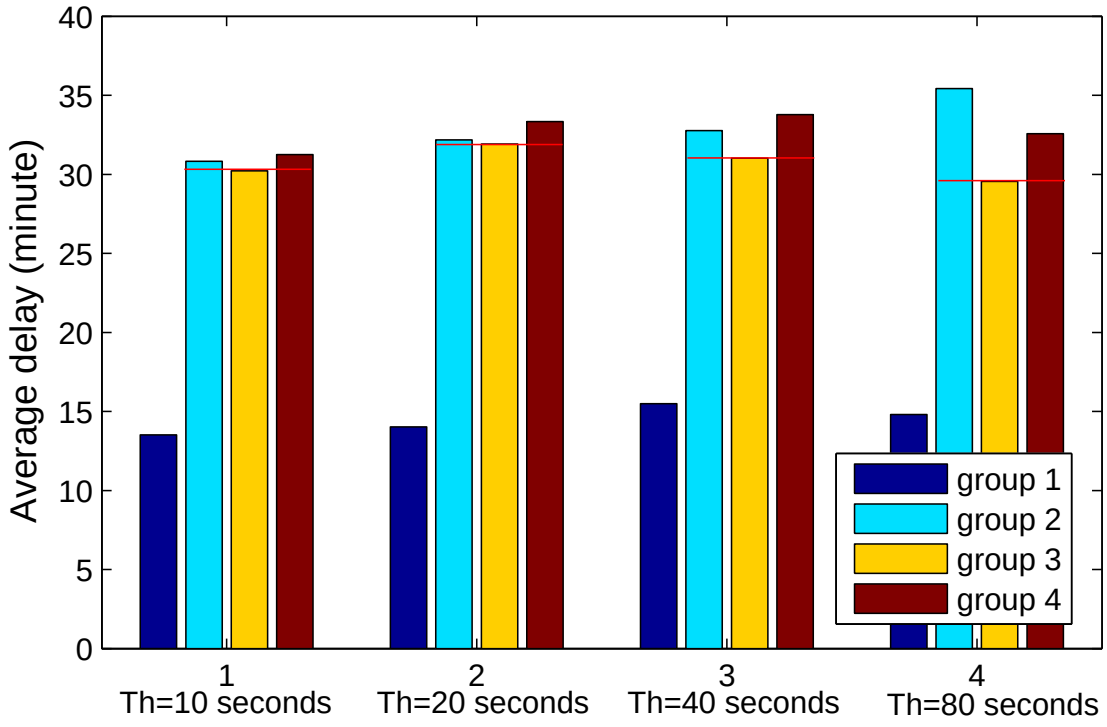
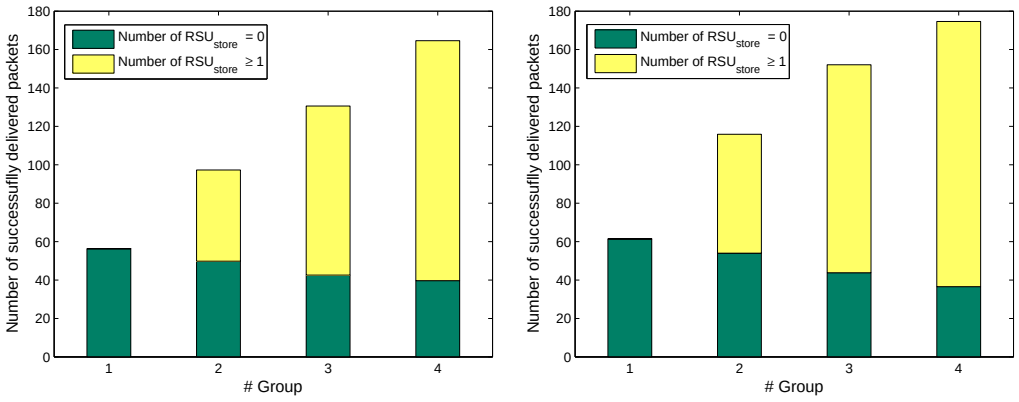


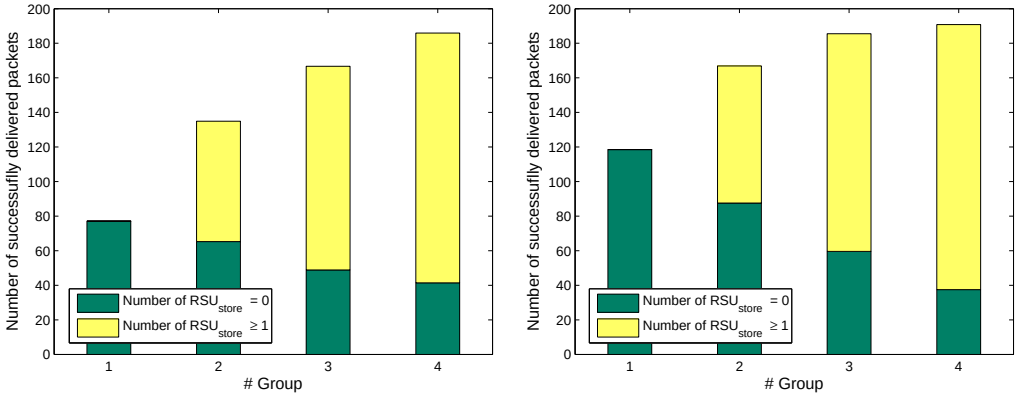
Figure 3.9: Average delay within 10 hours in different RSU deployments

Free Packets from Tracing Attack

As we mentioned earlier in Section 3.4, the RSUs naturally serve as Mix servers in the proposed SPRING protocol. Once a packet was temporarily stored at least in one RSU, it can resist the packet tracing attack launched by the external adversary. Fig. 3.10 shows the average number of successfully delivered packets in different RSU deployments. From the figure, we can observe, the more the RSUs are deployed, the more the successfully delivered packets are and the more the packets are temporarily stored at least in one RSU, as a consequence, the more packets can get rid of the packet tracing attack. Further observing the results in groups 2 and 3, we can conclude that the social-based RSU deployment can achieve better effects than the random RSU deployment in the proposed SPRING protocol.



(a) $T_h = 10$ seconds (b) $T_h = 20$ seconds



(c) $T_h = 40$ seconds (d) $T_h = 80$ seconds

Figure 3.10: Average number of successfully delivered packets

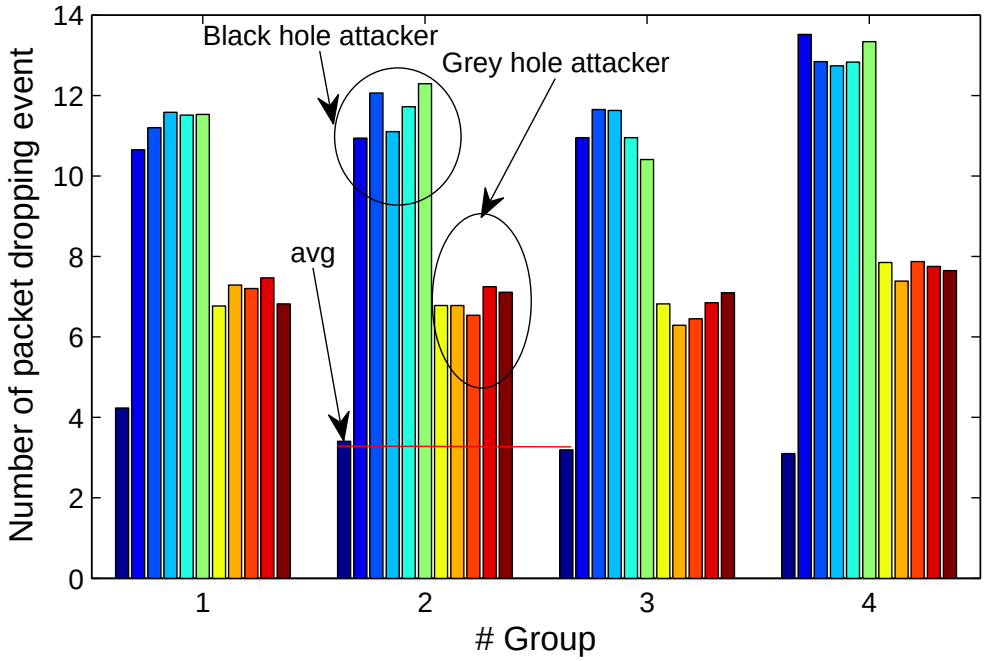


Figure 3.11: Black (grey) hole attack detection

Detection of Black (Grey) Hole Attacks

The proposed SPRING protocol provides the second line of defense to detect black (grey) hole attacks in privacy-preserving forwarding in vehicular DTNs. To evaluate the detection effect, we consider 5 black hole attackers and 5 grey hole attackers (with packet dropping probability (PDP)= 50%) among the total 50 vehicle nodes in the simulations. Then, Fig. 3.11 depicts the detection effects in different groups. From the figure, we can see, i) when more RSUs are deployed, the average dropped events will decrease; ii) the social-based RSU deployment has low average dropped events than that in the random RSU deployment. Therefore, when we choose proper thresholds T_B and T_G in Algorithm 2, these black (grey) hole attackers can be detected. Note that, because the grey hold attackers selectively drop the packets, the threshold T_G should be more carefully chosen than T_B, especially when the PDP is low in grey hold attacks.

3.6 Related Work

Recently, two research works on packet forwarding in DTNs are appeared, which are closely related to the proposed SPRING protocol [63, 68]. In [63], Banerjee *et al.* perform an experimental and analytical study of mobile networks enhanced with relays, meshes, and wired base stations. In specific, they first deploy a large-scale vehicular network and use wired base stations, meshes and relay nodes as the stationary nodes deployed in an interesting area to temporarily store packets for delivery to other mobile nodes, propagating information towards the final destination. This work has the same idea on "stationary node's assistance" as the proposed SPRING protocol. However, the security and privacy preservation issues are not addressed in the work. In [68], Hui *et al.* show that it is possible to detect characteristic properties of social grouping in a decentralized fashion from a diverse set of real world traces, and demonstrate that such characteristics can be effectively applied in packet forwarding decisions in DTN. Concretely, based on the observations that human interaction is heterogeneous both in terms of popular individuals and groups or communities, Hui *et al.* propose a social based forwarding algorithm (BUBBLE) for pocket switched networks (PSNs). The experimental results show that the BUBBLE algorithm can significantly improve the forwarding efficiency. Nevertheless, the security and privacy preservation issues are still not discussed in BUBBLE. Distinct from the above works, the proposed SPRING protocol not only heuristically studies the social-based RSU deployment for enhancing the delivery ratio in vehicular DTNs, but also discusses the privacy preservation issues as well as black hole and grey hole attacks in vehicular DTNs.

3.7 Summary

In this chapter, we have presented a social-based privacy-preserving packet forwarding (SPRING) protocol for vehicular DTNs. Based on social-based RSU deployment, the proposed SPRING protocol has been identified to be not only capable of significantly improving the reliability with V-2-V and V-2-I communications, but also able to achieve the privacy preservation and resist the black (grey) hole attacks in packet forwarding. Through extensive performance evaluation, we have demonstrated that the proposed SPRING protocol can achieve much better efficiency in terms of delivery ratio in vehicular DTNs.

Chapter 4

Socialspot Strategy for Protecting Receiver Location Privacy

4.1 Introduction

In the previous chapter, we introduced SPRING, a social-based privacy-preserving packet forwarding protocol for vehicular DTNs, which can not only improve the reliability with V-2-V and V-2-I communications, but also achieve the conditional vehicle identity privacy preservation and resist the black (grey) hole attacks in packet forwarding. However, the location of the destination is assumed stationary and known to the source in SPRING, and as a result, the receiver's location privacy is not protected. Since the flourish of VANET still hinges upon fully understanding and managing the security and privacy challenges including the location privacy that the public concerns [12, 69], in this chapter, we will focus on how to protect receiver-location privacy in packet forwarding application [24] in VANET.

Location privacy is one of important privacy requirements in VANET, since the locations of vehicles are tightly related to the drivers. Therefore, if VANET does not protect vehicle's location privacy, it cannot be accepted by the public. As for the packet forwarding application in VANET [24], to protect the receiver-location privacy, i.e., the receiver's location is unknown, a possible solution is adopting the flooding technique. However, as we know, the flooding technique will result in a large number of duplicate packets in the network. Though the flooding technique can protect the receiver-location privacy, it is very inefficient, especially when the storage at each vehicle is constrained. Therefore, how to simultaneously protect the receiver-location privacy and improve the performance of

packet delivery in VANET has become a new challenging issue. Unfortunately, to the best of our knowledge, this challenging issue has not be explored.

"Sacrificing the Plum Tree for the Peach Tree" is one of the Thirty-Six Strategies of Ancient China, which means sacrificing something non-critical to ensure the overall interests. In this chapter, we will use this tactic to propose an efficient socialspot-based packet forwarding (SPF) protocol to address the above challenging issue [45], where the socialspots are referred to as the locations in a city environment that many vehicles often visit such as a shopping mall, a restaurant, or a cinema. Since socialspots are usually low sensitive to the vehicles, we can utilize the socialspot as the relay node for packet forwarding. In such a way, the performance of packet delivery can be significantly improved. Meanwhile, since many vehicles visit the same socialspot, the socialspot cannot be used to trace a specific vehicle's other sensitive locations. Therefore, the socialspot tactic can protect the receiver-location privacy in VANETs. The main contributions of this chapter are two-fold.

- Firstly, based on the socialspot tactic, we propose an efficient SFP protocol aiming at packet forwarding application in VANETs, and also conduct the comprehensive security analysis to validate its security to protect the receiver-location privacy in VANETs. To the best of our knowledge, we are the *first* to utilize the socialspot tactic to resolve the above challenging issue.
- Secondly, we develop a custom simulator built in Java to examine the performance of the proposed SPF protocol. Extensive simulation results show that, the socialspot tactic can achieve good performance of packet forwarding in terms of packet delivery ratio and average delay in VANETs.

The remainder of this chapter is organized as follows. In Section 4.2, we introduce the system model, privacy model and design goal. Our proposed SPF protocol is presented in Section 4.3, followed by its security analysis and performance evaluation in Section 4.4 and Section 4.5, respectively. We also discuss the related work in Section 4.6. Finally, we draw our summary in Section 4.7.

4.2 Models and Design Goal

In this section, we formalize the system model, privacy model, and identify our design goal.

4.2.1 System Model

We consider a typical VANET which consists of a trusted authority (TA), a large number of vehicles and some socialspots in a city environment, as shown in Fig. 4.1.

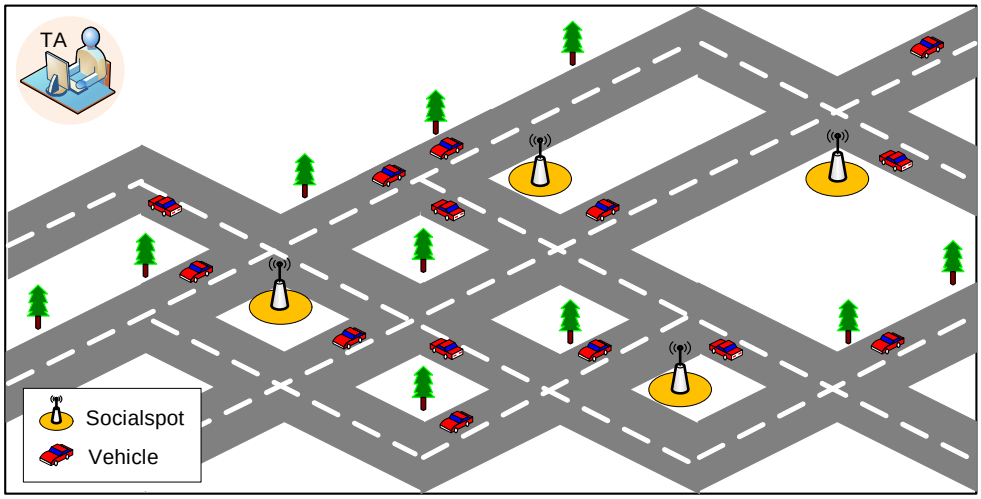


Figure 4.1: System model under consideration for SPF

- Trusted Authority (TA): TA is a trustable and powerful entity, whose duties include initializing the system, deploying RSUs at some socialspots, and registering vehicles by granting a family of pseudo-IDs and the corresponding key materials.
- Socialspots $\mathcal{S} = \{\mathsf{ss}_1\mathsf{ss}_2, \cdots\}$: Socialspots are referred to as the locations where many vehicles will visit, for example, a shopping mall, a restaurant, or a cinema. At each socialspot $\mathsf{ss}_i \in \mathcal{S}$, TA will deploy a trusted and identified storage-huge RSU, so that it can temporarily store some packets in packet forwarding application.
- Vehicles $\mathcal{V} = \{V_1, V_2, \cdots\}$: Each vehicle $V_i \in \mathcal{V}$ is equipped with the OBU device, which allows them to communicate with each other as well as those RSUs at socialspots for cooperative packet delivery in VANET. In general, the OBU device in VANET has no power-limited issue, however the storage is assumed constrained.

4.2.2 Privacy Model

In our privacy model, we consider how to protect a vehicle receiver's location privacy against an *external*, *global*, and *passive* adversary $\mathcal{A}$, where the adversary $\mathcal{A}$ does not com-

promise any RSUs or vehicles, but has a complete view to eavesdrop all packets forwarding in VANET. Note that, the adversary $\mathcal{A}$ could launch some active attacks such as black hole attack, grey hole attack to degrade the performance in cooperative packet delivery application [24]. However, since the focus of our work is on protecting receiver-location privacy, these active attacks are beyond the scope of this chapter.

4.2.3 Design Goal

By utilizing the socialspot strategy, our design goal is to develop an efficient socialspot-based packet forwarding (SPF) protocol to protect receiver-location privacy in VANETs. Specifically, since not all locations in a vehicle V_i's trajectory $\mathsf{Tr}_i = \{\mathsf{tr}_1, \mathsf{tr}_2, \cdots\}$ are sensitive to it, it is possible to reveal a non-sensitive socialspot that V_i often visits as a stationary relay node so that the packet delivery performance can be improved. At the same time, since many vehicles often visit the same socialspot, the RSU at the socialspot can serve as a mix server [24], then the adversary $\mathcal{A}$ cannot link a specific packet to its receiver. In addition, since each vehicle V_i periodically changes his pseudo-IDs on the road, the receiver's sensitive locations are unlinkable and privacy-preserving.

4.3 Proposed SPF Protocol

In this section, we will present our Socialspot-based Packet Forwarding (SPF) protocol for protecting receiver-location privacy in VANET. Before proceeding the SPF protocol, the rationale of socialspot strategy is first introduced.

4.3.1 Rationale of Socialspot Strategy

In reality, the locations in a driver's trajectory is almost fixed. For example, a driver may often drive to his home, school, and shopping mall. As for a driver, his home and school could be privacy locations, which are sensitive to him; while the shopping mall is a socialspot, which is usually not cared about. Therefore, it is possible to apply "Sacrificing the Plum Tree for the Peach Tree" strategy to reveal a receiver's socialspot as a stationary relay node to improve the performance of packet forwarding in VANET while protecting the receiver's other locations privacy, as shown in Fig. 4.2.

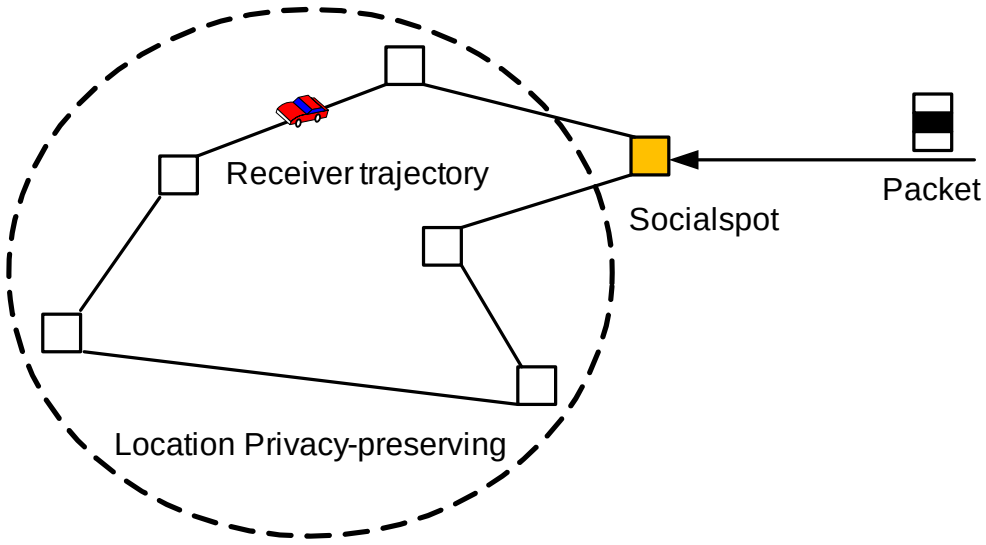


Figure 4.2: Socialspot strategy to improve the performance of packet forwarding

4.3.2 Description of SPF Protocol

The SPF protocol consists of four phases: system initialization phase, packet generation phase, packet forwarding phase, and packet receiving phase.

System Initialization Phase

In the system initialization phase, the TA first configures the system parameter, chooses social spots in a city environment, and registers vehicles in the system. Specifically, the TA runs the following steps.

Step 1. Given the security parameter κ, the bilinear map groups $(\mathbb{G}, \mathbb{G}_T, e, P, q)$ are generated by running $\mathbb{G}en(\kappa)$, where $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, P is a generator of $\mathbb{G}$ and q is a large prime with $|q| = \kappa$. Then, TA chooses an random element $Q \in \mathbb{G}$, and a random number $s \in \mathbb{Z}_q^*$ as the *master key*, and computes the corresponding system public key $P_{pub} = sP$. In addition, TA chooses two secure cryptographic hash functions $\mathcal{H}_0, \mathcal{H}_1$, where $\mathcal{H}_i : \{0,1\}^* \to \mathbb{Z}_q^*$, for $i = 0, 1$, and a secure symmetric encryption algorithm $\mathbf{Enc}()$. In the end, TA sets the system public parameters *params* as $(\mathbb{G}, \mathbb{G}_T, e, P, q, Q, P_{pub}, \mathcal{H}_0, \mathcal{H}_1, \mathbf{Enc}())$.

Step 2. TA chooses a set of socialspots $\mathcal{S} = \{\mathsf{ss}_1, \mathsf{ss}_2, \cdots\}$ in a city environment. Then, at each socialspot $\mathsf{ss}_i \in \mathcal{S}$, a storage-huge RSU is placed, which can be identified and trusted by passing-by vehicles.

Step 3. Assume the trajectory of a vehicle $V_i \in \mathcal{V}$ is $\mathsf{Tr}_i = \{\mathsf{tr}_1, \mathsf{tr}_2, \cdots\}$ such that $\mathsf{Tr}_i \cap \mathcal{S} \neq \phi$, i.e., at least there exists one location $\mathsf{tr}_a = \mathsf{ss}_b$, with $\mathsf{tr}_a \in \mathsf{Tr}_i$ and $\mathsf{ss}_b \in \mathcal{S}$. Then, when V_i registers himself, he submits his identity and the socialspot ss_b to TA. TA then grants a family of pseudo-IDs $\mathsf{PID} = \{\mathsf{pid}_0, \mathsf{pid}_1, \cdots\}$ and the corresponding pseudo-ID-based key materials for V_i by invoking Algorithm 3. In such a way, V_i can use pid_0 at socialspot ss_b and constantly change its pseudo-IDs $\mathsf{pid}_j \in \mathsf{PID}$, $j \geq 1$, at other places to achieve identity privacy [70] and location privacy in a city environment.

Algorithm 3 Vehicle Registration Algorithm

1: **procedure** VehicleRegistration
Input: a vehicle $V_i \in \mathcal{V}$ and a socialspot $\mathsf{ss}_b \in \mathcal{S}$
Output: a family of pseudo-IDs and the corresponding pseudo-ID based key materials
2: choose a family of unlinkable pseudo-IDs $\mathsf{PID} = \{\mathsf{pid}_0, \mathsf{pid}_1, \cdots\}$
3: compute the private key $S_0 = \frac{1}{s+\mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)}Q$ with respect to the pseudo-ID $\mathsf{pid}_0 \in \mathsf{PID}$ and the socialspot ss_b
4: **for** other pseudo-ID $\mathsf{pid}_j \in \mathsf{PID}$, $j \geq 1$ **do**
5: compute the corresponding private key $S_j = \frac{1}{s+\mathcal{H}_0(\mathsf{pid}_0)}Q$
6: **end for**
7: **return** all tuples (pid_j, S_j) to V_i
8: **end procedure**

Packet Generation Phase

Assume that a stationary source wants to send a message M to a vehicle $V_i \in \mathcal{V}$ in the city environment. However, the source does not know the exact location of V_i, what he knows is only V_i's pseudo-ID pid_0 and the socialspot ss_b. Then, the source executes the following steps to generate a packet on M.

Step 1. The source first chooses a random number $x \in \mathbb{Z}_q^*$, and computes $k = e(P, Q)^x$, and C_1, C_2, C_3, where

$$\begin{cases} C_1 = x(P_{pub} + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P) \\ C_2 = \mathcal{H}_1(k||0), C_3 = \mathbf{Enc}(k, M) \end{cases}$$

Note that, (C_1, C_3) is a ciphertext of the anonymous identity-based encryption [71], which thus can achieve the receiver-identity anonymous.

Step 2. The source then packs the packet $\mathcal{P}$ with the format shown in Fig. 4.3, where $\mathsf{Head} := C_1$, $\mathsf{Auth} := C_2$, $\mathsf{Encrypted\text{-}Payload} := C_3$, and $\mathsf{Socialspot} := \mathsf{ss}_b$, and waits for some vehicles to help with forwarding the packet $\mathcal{P}$ to ss_b.

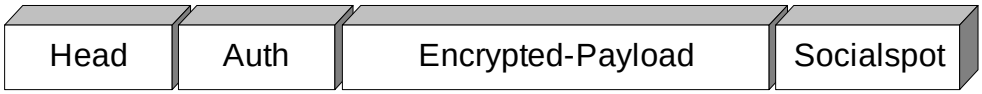


Figure 4.3: The format of packet in the SPF protocol

Packet Forwarding Phase

Generally, the law of proximity shows that if a packet is located close to the socialspot ss_b, it is more likely that it can be carried to ss_b by some vehicles as soon as possible. Therefore, in this phase, the source will ask a passing-by vehicle to help with carrying the packet $\mathcal{P}$ to the socialspot ss_b or other socialspots close to ss_b. Specifically, the source invokes the Algorithm 4 to forward the packet $\mathcal{P}$.

Algorithm 4 Packet Forwarding Algorithm

1: **procedure** PacketForwarding
2: When a vehicle is passing-by the source, the source asks for the help. If vehicle can forward $\mathcal{P}$ to ss_b or other socialspots close to ss_b, the source forwards $\mathcal{P}$ to the vehicle.
3: **end procedure**

If the packet $\mathcal{P}$ is successfully forwarded to the socialspot ss_b, this phase is ended. Otherwise, when the packet is forwarded to other socialspots close to ss_b, then the RSU at the socialspot will temporally store the packet $\mathcal{P}$, and also invoke the Algorithm 4 to help with forwarding $\mathcal{P}$ to ss_b. The above RSU forwarding is iterative, and the packet $\mathcal{P}$ can be forwarded to ss_b eventually.

Packet Receiving Phase

Once the packet $\mathcal{P}$ reaches the socialspot ss_b and is stored at the RSU, the vehicle V_i can pick up the packet $\mathcal{P}$ by the following steps, when it comes across the socialspot ss_b.

Step 1. The vehicle V_i first establishes a secure channel with the RSU by the following interactions, as shown in Fig. 4.4.

- V_i sends his pseudo-ID pid_0 to the RSU;
- RSU chooses a random number $r \in \mathbb{Z}_q^*$, computes the session key $\mathsf{sk} = e(P, Q)^r$, and sends the challenge $\mathsf{Cha} = r(P_{pub} + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P)$ back to V_i;

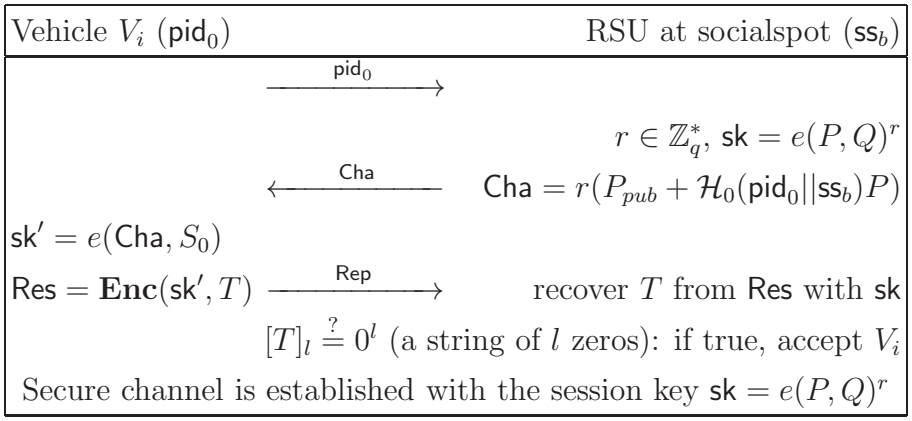


Figure 4.4: Secure channel establishment between V_i and a trusted RSU

- After receiving the challenge Cha, V_i first computes $\mathsf{sk}' = e(S_0, \mathsf{Cha})$, and the response $\mathsf{Res} = \mathbf{Enc}(\mathsf{sk}', T)$, where T is a random number with the least significant l bits $[T]_l = 0^l$, and then sends Rep back to RSU.
- Upon receiving the response Res, RSU recovers T from Res, and checks whether the least significant l bits $[T]_l \stackrel{?}{=} 0^l$. If it is true, V_i is authenticated; and the secure channel with the session key $\mathsf{sk} = e(P,Q)^r$ is established. The correctness is as follows

$$\begin{aligned}\mathsf{sk}' &= e(\mathsf{Cha}, S_0)\\ &= e(r(P_{pub} + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P, \frac{1}{s + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)}Q)\\ &= e(P,Q)^r = \mathsf{sk}\end{aligned}$$

Note that, because the RSU is trusted and can be identified by the vehicle, the unilateral authentication on vehicle here is suitable for the application scenarios.

Step 2. Once the secure channel is established, the vehicle V_i picks up each packet's Head and Auth from RSU by checking the relation

$$\mathsf{Auth} = C_2 \stackrel{?}{=} \mathcal{H}_1(k'||0), \text{ where } k' = e(\mathsf{Head}, S_0)$$

If the relation holds, V_i requests the packet's $\mathsf{Encrypted\text{-}Payload}$ from RSU and recovers the message M with k' from the encrypted-payload $C_3 = \mathbf{Enc}(k, M)$. The correctness is also

as follows

$$\begin{aligned} k' &= e(\mathsf{Head}, S_0) = e(C_1, S_0) \\ &= e(x(P_{pub} + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P, \frac{1}{s + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)}Q) \\ &= e(P, Q)^x = k \end{aligned}$$

In such a way, the message M is successfully received by the receiver V_i, and the SPF protocol ends. Note that, because the receiver V_i is mobile, when V_i happens to move to the location of the stationary source, V_i can also establish a secure channel with the source and directly get the message M.

Computational costs. The computational costs of the SPF protocol is dominant by the process that the vehicle V_i checks all packets stored in RSU. Let T_{pair} denote the time to perform one pairing operation. Since the pairing operation dominates the speed of each check, we only consider T_{pair} and neglect the hash operation in measure of the process. Then, we can see the whole process requires $n \cdot T_{\mathsf{pair}}$ to complete checking all packets, where n is the number of packets in the RSU. When we implement the SPF protocol by using the bilinear maps in [14], we know that $T_{\mathsf{pair}} \approx 4.5$ ms, and then the costs are $n \cdot 4.5$ ms. If each packet is labeled with a timestamp, then the vehicle V_i only needs to check the packets with the specific timestamp, and the computational costs can be reduced.

4.4 Security Analysis

In this section, we will discuss the security issues of the proposed SPF protocol, i.e., the receiver-location privacy-preservation against an *external*, *global* and *passive* adversary $\mathcal{A}$ in VANETs.

• *The packet $\mathcal{P}$ in the proposed SPF protocol can protect the receiver's identity privacy.* Since Head and Encrypted-Payload in the packet $\mathcal{P}$ is (C_1, C_3), where

$$\begin{cases} C_1 = x(P_{pub} + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P) \\ C_3 = \mathbf{Enc}(k, M)) \end{cases}$$

is a valid ciphertext of the anonymous identity-based encryption [71], (C_1, C_3) is provably secure and will not disclose the receiver's identity. At the same time, the Auth is $C_2 = \mathcal{H}_1(k||0)$, where $k = e(P, Q)^x$ is also irrelative to the receiver identity pid_0. Due to these two reasons, the packet $\mathcal{P}$ can protect the receiver's identity privacy, which is a prerequisite for protecting receiver-location privacy.

• *The session key* $\mathsf{sk} = e(P,Q)^r$ *between the vehicle and RSU is semantic secure and can protect the receiver's session privacy.* Because the RSU is deployed at the socialspot, it thus will store many different vehicles' packets. If the session key is secure, it is hard for an adversary to link a packet to a receiver. In the following, based on the DBDH assumption, we first prove that the session key $\mathsf{sk} = e(P,Q)^r$ is semantic secure, which serves as the necessary condition for receiver's session privacy. Assume that there is an adversary $\mathcal{A}'$ which runs in a polynomial time and has a non-negligible advantage ϵ' to break the semantic security of the session key sk in the proposed SPF protocol, then we can use the capability of $\mathcal{A}'$ to construct another adversary $\mathcal{A}$ to break the DBDH problem, i.e., given $(x\widetilde{P}, y\widetilde{P}, z\widetilde{P}, V)$, decide whether or not $V = e(\widetilde{P},\widetilde{P})^{xyz}$ for unknown $x, y, z \in \mathbb{Z}_q^*$. First, $\mathcal{A}$ sets the system parameters $P = x\widetilde{P}$ and $Q = z\widetilde{P}$, which implicitly shows that $Q = z\widetilde{P} = \frac{z}{x}P$. Then, $\mathcal{A}$ chooses a random number $t \in \mathbb{Z}_q^*$, and implicitly define the master key $s = \frac{t}{x} - \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b) \bmod q$. Note that, due to unknown x, the implicitly defined s is also unknown to $\mathcal{A}$, but it does not affect the interactions between $\mathcal{A}$ and $\mathcal{A}'$. Next, $\mathcal{A}$ also implicitly defines a random number $r = \frac{y}{t} \bmod q$ used in the challenge Cha. Then, the challenge Cha is

$$\begin{aligned}
\mathsf{Cha} &= r(P_{pub} + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P) \\
&= \frac{y}{t}\left(\left(\frac{t}{x} - \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)\right)P + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P\right) \\
&= \frac{y}{t}\left(\frac{t}{x}P - \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P + \mathcal{H}_0(\mathsf{pid}_0||\mathsf{ss}_b)P\right) \\
&= \frac{y}{x}P = y\widetilde{P}
\end{aligned}$$

In addition, $\mathcal{A}$ computes $V^{\frac{1}{t}} \in \mathbb{G}_T$ and uses it to encrypt a random number T with $[T]_l = 0^l$ as the response $\mathsf{Res} = \mathbf{Enc}(V^{\frac{1}{t}}, T)$. In the end, $\mathcal{A}$ sends $(P, Q, \mathsf{Cha}, \mathsf{Res})$ to $\mathcal{A}'$ for creating the attack environment of $\mathcal{A}'$.

Upon receiving $(P, Q, \mathsf{Cha}, \mathsf{Res})$, $\mathcal{A}'$ sends a request of guess on the session key. Then, A flips a coin $b \in \{0,1\}$ and sends $V^{\frac{1}{t}}$ to $\mathcal{A}'$. When $\mathcal{A}'$ receives the $V^{\frac{1}{t}} \in \mathbb{G}_T$, he returns a bit b' as the guess of b.

Let E be the event that $(P, Q, \mathsf{Cha}, \mathsf{Res})$ are all valid. When the event E occurs, $\mathcal{A}'$ can launch his attacking capability, and we define $\mathbf{Adv}_{\mathcal{A}'}^{\mathsf{sk}} = 2\Pr[b' = b|\mathsf{E}] - 1 \geq \epsilon'$ to be the advantage probability of $\mathcal{A}'$, i.e.,

$$\Pr[b' = b|\mathsf{E}] = \frac{\mathbf{Adv}_{\mathcal{A}'}^{\mathsf{sk}}}{2} + \frac{1}{2} \geq \frac{\epsilon'}{2} + \frac{1}{2}$$

If the DBDH challenge $(x\widetilde{P}, y\widetilde{P}, z\widetilde{P}, V)$ is actually a bilinear pairing tuple $(x\widetilde{P}, y\widetilde{P}, z\widetilde{P}, V = e(\widetilde{P}, \widetilde{P})^{xyz})$, i.e., $\widetilde{b} = 0$ in $\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}}$, then

$$V^{\frac{1}{t}} = e(\widetilde{P}, \widetilde{P})^{\frac{xyz}{t}} = e(x\widetilde{P}, z\widetilde{P})^{\frac{y}{t}} = e(P, Q)^r$$

and $\mathsf{Res} = \mathbf{Enc}(V^{\frac{1}{t}}, T)$ is also valid. Therefore,

$$\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}} = 1 | \widetilde{b} = 0\right] = \Pr[b' = b | \mathsf{E}] = \frac{\mathbf{Adv}_{\mathcal{A}'}^{\mathsf{sk}}}{2} + \frac{1}{2}$$

However, if the DBDH challenge $(x\widetilde{P}, y\widetilde{P}, z\widetilde{P}, V)$ is a random tuple $(x\widetilde{P}, y\widetilde{P}, z\widetilde{P}, V = R)$, i.e., $\widetilde{b} = 1$ in $\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}}$, we know $V^{\frac{1}{t}} \neq e(P, Q)^r$, the response Res is not valid, and the event E does not occur. Then, the guess of $\mathcal{A}$ is independent of b. Therefore,

$$\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}} = 1 | \widetilde{b} = 1\right] = \Pr[b' = b | \neg\mathsf{E}] = \frac{1}{2}$$

Based on the above relations, we have

$$\begin{aligned}\mathbf{Adv}_{\mathcal{A}}^{\mathrm{DBDH}} &= \left|\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}} = 1 | \widetilde{b} = 0\right]\right. \\ &\quad \left. - \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{\mathrm{DBDH}} = 1 | \widetilde{b} = 1\right]\right| \\ &= \left|\frac{\mathbf{Adv}_{\mathcal{A}'}^{\mathsf{sk}}}{2} + \frac{1}{2} - \frac{1}{2}\right| = \frac{\mathbf{Adv}_{\mathcal{A}'}^{\mathsf{sk}}}{2} \geq \frac{\epsilon'}{2}\end{aligned}$$

This result indicates that the session key $\mathsf{sk} = e(P, Q)^r$ is semantic secure in the proposed SPF protocol, as required. Furthermore, because the session key $\mathsf{sk} = e(P, Q)^r$ is semantic secure, when it is used to encrypt the communications between the vehicle and the RSU, the vehicle's session privacy is protected, i.e., an adversary $\mathcal{A}$ cannot know which packet the vehicle has picked up from the RSU.

• *The receiver's sensitive locations are unlinkable in the proposed SPF protocol.* As we know, each vehicle V_i holds a family of unlinkable pseudo-IDs and the corresponding key materials, and only pseudo-ID pid_0's key is related to the socialspot ss_b, other keys are independent of the locations. Therefore, when V_i periodically changes its pseudo-IDs on the road, other sensitive locations of V_i are unlinkable.

In summary, we can clearly see that the proposed SPF protocol can protect the receiver-location privacy in VANETs.

4.5 Performance Evaluation

In this section, we evaluate the average-case performance of the proposed SPF protocol, using a custom simulator built in Java. The performance metrics gauged in the evaluation are average packet delivery ratio (DR) and average packet delay (AD), where the DR is defined as the average ratio of the packets successfully reach their destinations with respect to those generated by the sources within a given time period, and the AD is defined as the average between when a packet is generated at source and when it is successfully delivered to its destination.

4.5.1 Simulation Settings

In the simulation, $N = \{40, 80\}$ vehicles with transmission radius of 300 meters and velocity varying from 40 km/h to 60 km/h are moving in an interest area of $10,000 \times 10,000$ m^2, as shown in Fig. 4.5. In addition, 6 socialspots are randomly chosen in the region, each socialspot is deployed with a storage-huge RSU to help with temporarily storing the packets.

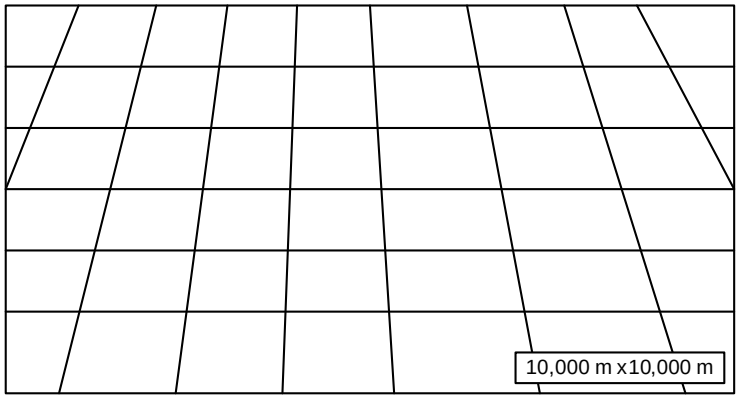


Figure 4.5: Interest area considered for simulation in SPF

Mobility model. In VANET, the performance of packet forwarding is highly contingent upon the mobility of the vehicles. Meanwhile, since the rationale of socialspot tactic is based on the assumption that each vehicle often visits at least one socialspot, we consider a special but more realistic mobility model in the simulation. Set each vehicle V_i's trajectory $\mathsf{Tr}_i = \{\mathsf{tr}_1, \mathsf{tr}_2, \cdots\}$ contains the same number of locations, i.e., $|\mathsf{Tr}_i| = S = \{3, 5\}$. In the trajectory Tr_i, at least one location belongs to the socialspots, and the rest locations

are randomly distributed in the area. In the simulation, each vehicle moves around his individual trajectory. Specifically, each vehicle V_i first equally chooses one location in his trajectory $\mathsf{Tr}_i = \{\mathsf{tr}_1, \mathsf{tr}_2, \cdots\}$, and gets there using the map-based shortest path routing. After reaching the destination, with 2-minute pause time, the vehicle again equally chooses a new destination in Tr_i and repeats the above.

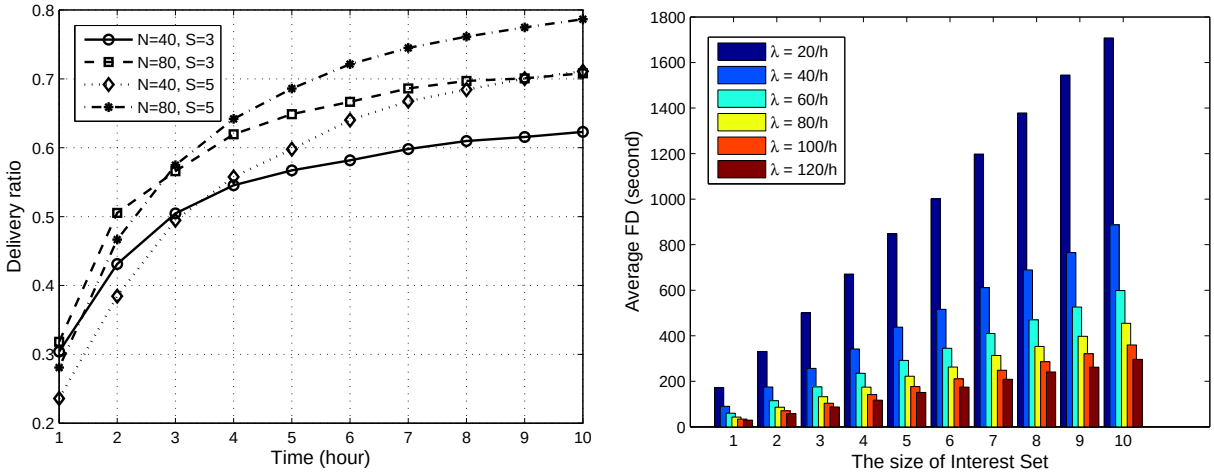
The detailed parameter settings in the simulations are summarized in Table 4.1. We perform the experiments with different $N = \{40, 80\}$ and different $S = \{3, 5\}$. For each case, we run the simulation 10 hours, and the average delivery ratio and average packet delay over 50 runs are reported.

Table 4.1: Simulation Settings in SPF

Parameter	Setting
Simulation area	$10,000 \times 10,000$ m^2
Simulation duration	10 hours
Number of socialspots, RSU storage	6, 10000 M
Number of vehicles	$N = \{40, 80\}$
Number of locations in vehicle's trajectory	$S = \{3, 5\}$
Vehicle velocity and transmission	$40 \sim 60$ km/h, 300 m
Vehicle storage and waittime	20 M, 2 minutes
Package size, generation interval	1 M, 5 minutes

4.5.2 Simulation Results

Fig. 4.6(a) shows the average DR varies with the time period from 1 hour to 10 hours. From the figure, we can observe that, with the increase of time, the DR will increase accordingly. When the number of socialspots S is fixed, the DR in $N = 80$ case is higher than that in $N = 40$ case. The reason is that, when more vehicles move around the area, more packets can be carried to the socialspots, then the receivers can get their packets when they visit the socialspots. Furthermore, when the number of vehicles N is fixed, the DR in $S = 5$ case is lower than that in $S = 3$ case at the initial stage and will be higher in the late stage. The reason for the phenomena is that at the initial stage, the number of generated packets is small, fewer packets should be carried to the socialspots. When $S = 5$, receivers have lower frequency to visit the socialspots to pick up their packets. Therefore, the DR is lower than that in $S = 3$ case. However, in the late stage, the number of generated packets is larger. When $S = 5$, vehicles can carry more packets to the socialspots. Accordingly, when the receivers visit the socialspots, they can receive more packets. Therefore, the DR is larger than that in $S = 3$ case.



(a) delivery ratio (b) average delay

Figure 4.6: The average packet delivery ratio and average delay

Fig. 4.6(b) shows the AD varies with the time period from 1 hour to 10 hours corresponding to the DR in Fig. 4.6(b). From the figure, we can see, with the increase of time, the AD will also increase, but the increased delay can improve the DR. When the number of socialspots S is fixed, 80 vehicles can carry packets more quickly to the socialspots than 40 vehicles. As a result, the AD in $N = 80$ case is lower than that in $N = 40$ case. Meanwhile, when the number of vehicles N is fixed, vehicles will visit more locations in $S = 5$ than that in $S = 3$ case. Therefore, the AD in $S = 3$ is lower than that in $S = 5$ case, but the corresponding DR is also lower.

4.6 Related Work

Recently, several research works have been reported [72, 73, 24], which are closely related to the proposed SPF protocol. Jian *et al.* [72] study the packet-tracing attack and propose a location privacy routing (LPR) protocol in combination with fake packet injection technique to protect receiver-location privacy in wireless sensor networks. In [73], Cheng *et al.* propose an efficient packet cloaking routing mechanism to protect the privacy of a receiver. The main idea in packet cloaking is to transmit multiple copies of a sent packet to a selected group of k receivers, so that an adversary may only identify the true receiver with a probability of $1/k$. Although both LPR protocol and packet cloaking mechanism can protect the receiver-location privacy, they cannot be applied to VANET, since the re-

ceivers in VANETs are vehicles, which move around in the city environment. Our previous work SPRING [24] studies how to utilize the vehicle mobility model, i.e., map-based shortest routing, to improve the performance of packet forwarding in vehicular DTN. However, SPRING only addresses the stationary receiver, and does not consider the receiver-location privacy. Different from SPRING, the proposed SPF protocol considers the more realistic mobility model, which not only protects the receiver-location privacy, but also improves the performance of packet delivery.

4.7 Summary

In this chapter, based on the "Sacrificing the Plum Tree for the Peach Tree" — one of the Thirty-Six Strategies of Ancient China, we have proposed a socialspot-based packet forwarding (SPF) protocol for protecting receiver-location privacy in VANETs. Detailed security analysis has shown that, only when a receiver sacrifices one socialspot that he often visits, all his other sensitive locations can be protected against an external, global, passive adversary. In addition, through extensive performance evaluation, we have demonstrated that the temporarily storing packets at socialspots can achieve much better efficiency in terms of delivery ratio and average delay in VANETs.

Chapter 5

Effective Pseudonym Changing Strategy for Location Privacy

5.1 Introduction

In the previous chapter, we presented SPF protocol, where we introduced "sacrificing the plum tree for the peach tree" strategy in VANETs to achieve receiver location privacy in packet forwarding application. Specifically, in SPF protocol, each vehicle holds a family of pseudo-IDs, one is public and used at social spot to efficiently receive packets, and others are constantly changed by the vehicle to achieve identity privacy and location privacy. Now, the question is how, when and where the vehicle changes these pseudo-IDs? And is the pseudonyms changing effective? To answer this question, in this chapter, we proposed an effective pseudonyms changing at social spots strategy to facilitate a vehicle to achieve high-level location privacy.

It is well known that, to achieve location privacy, a popular approach recommended in VANETs is that vehicles periodically change their pseudonyms when they are broadcasting *safety messages* (where each *safety message* is a 4-tuple including `Time`, `Location`, `Velocity`, `Content`, and is authenticated with a `Signature` with respect to a `Pseudonym`) [7, 2, 28]. Because a vehicle uses different pseudonyms on the road, the *unlinkability* of pseudonyms can guarantee a vehicle's location privacy. However, if a vehicle changes its pseudonyms in an improper occasion, changing pseudonyms has no use to protect location privacy, since an adversary could still link a new pseudonym with the old one [74]. As an example shown in Fig. 5.1, when three vehicles are running on the road, if only one vehicle changes its pseudonyms during Δt, an adversary can still monitor the

pseudonyms' link. Even though all three vehicles change their pseudonyms simultaneously, the `Location` and `Velocity` information embedded in *safety messages* could still provide a clue to the adversary to link the pseudonyms, making the privacy protection fail. Therefore, it is imperative for us to exploit the accuracy of location privacy achieved by frequent changing pseudonyms in VANETs [75, 76, 77, 78, 79, 80]. Formally, we let $\overrightarrow{F} = \{F_1, F_2, F_3, \cdots\}$ be multi-dimensional character factors associated with a pseudonym changing process. For example, the vector $\overrightarrow{F} = \{F_1, F_2, F_3, \cdots\}$ can represent factors $\{\texttt{Time}, \texttt{Location}, \texttt{Velocity}, \cdots\}$. In some specific scenarios, an adversary has the ability to monitor a subset $\overrightarrow{F}_n = \{F_1, F_2, \cdots, F_n\} \subset \overrightarrow{F}$ and use it for identifying a vehicle pseudonym changing process. Suppose $\overrightarrow{b}_0 = (x_1, x_2, \cdots, x_n)$ and $\overrightarrow{b}_1 = (y_1, y_2, \cdots, y_n)$ be the characteristic vectors of two vehicles' pseudonym changing processes observed by an adversary. Then, the cosine-based similarity between $\overrightarrow{b}_0$ and $\overrightarrow{b}_1$ can be given by

$$\cos(\overrightarrow{b}_0, \overrightarrow{b}_1) = \frac{\overrightarrow{b}_0 \odot \overrightarrow{b}_0}{|\overrightarrow{b}_0| \cdot |\overrightarrow{b}_1|} = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}}$$

Obviously, when $\overrightarrow{b}_0$ and $\overrightarrow{b}_1$ are identical, $\cos(\overrightarrow{b}_0, \overrightarrow{b}_1) = 1$. Due to the monitoring inaccuracy, if $|1 - \cos(\overrightarrow{b}_0, \overrightarrow{b}_1)| \leq \epsilon$, for some small confusion value $\epsilon > 0$, two pseudonyms changing processes can be regarded as indistinguishable in the eye of the adversary. Therefore, in order to protect location privacy with high quality, a vehicle should choose a proper scenario where as many as possible indistinguishable pseudonyms changing processes are taken place simultaneously.

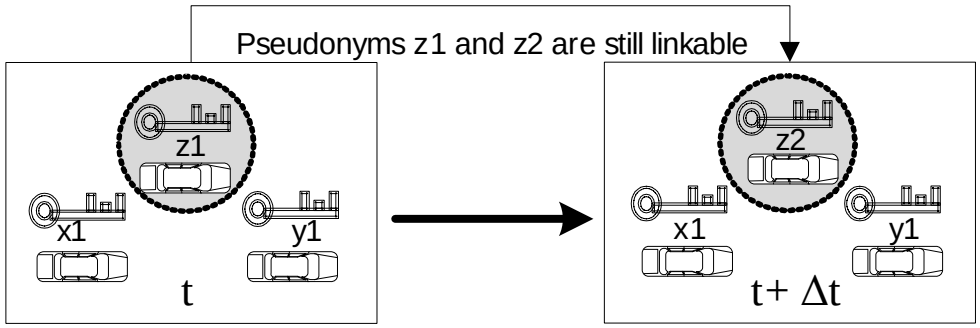


Figure 5.1: Pseudonyms link due to changing pseudonyms at an improper occasion

In this chapter, to facilitate vehicles to achieve high-level location privacy in VANETs, we propose an effective pseudonyms changing at social spots strategy, called PCS [46]. In the PCS strategy, the social spots are the places where many vehicles temporarily gather,

e.g., the road intersection when the traffic light turns red, or a free parking lot near a shopping mall. If all vehicles change their pseudonyms before leaving the spot, the first broadcasted *safety message* includes indistinguishable information `Location`= social spot, `Velocity`=0, and unlinkable `Pseudonym`. Then, the social spot naturally becomes a *mix zone*, and the location privacy can be achieved. Specifically, in this work, our contributions are threefold.

First, we utilize the unique feature of social spots, i.e., many vehicles temporarily stop at the social spot, to propose the PCS strategy. In addition, as an important technical preliminary of PCS strategy, we present a practical key-insulated pseudonym self-delegation (KPSD) model, which securely generates many on-demand short-life keys and can mitigate the hazards due to vehicle theft.

Second, we take the anonymity set size (ASS) as the privacy metric (the larger the anonymity set size, the higher the anonymity achieved [81, 74]) to measure the Quality of Privacy (QoP) achieved in PCS strategy. To the best of our knowledge best knowledge, most previously reported schemes [74, 80] use the simulations to gauge the achieved location privacy in VANETs, and thus our anonymity set analytic models will shed light on this research line.

Third, to guarantee the PCS strategy can be effectively adopted in practice, we use the simplified game theoretic techniques to formally prove the feasibility of the PCS strategy. As a result, the PCS strategy can really guide vehicles to intelligently change their pseudonyms for better location privacy at the right moment and place.

The remainder of this chapter is organized as follows. In Section 5.2, we formalize the problem by describing the network model, threat model, and identifying the requirements of location privacy in VANETs. Then, we present the PCS strategy in Section 5.3, followed by the performance evaluations in Section 5.4. We also review some related work in Section 5.5. Finally, we draw our summary in Section 5.6.

5.2 Problem Definition

In this section, we define the problem by formalizing the network model, threat model, and identifying the requirements of location privacy in VANETs.

5.2.1 Network Model

We consider a VANET in the urban area, which consists of a large number of vehicles and a collection of social spots[1] as

- *Vehicles:* in the urban area, a large number of vehicles are running on the road everyday. Each vehicle is equipped with an OnBoard Unit (OBU) device, which allows the vehicle to communicate with other vehicles for sharing local traffic information to improve the whole safety driving conditions.
- *Social Spots:* the social spots in the urban area refer to the places where many vehicles gather, for example, a road intersection when the traffic light is red or a free parking lot near the shopping mall, as shown in Fig. 5.2. Since the session of red traffic light is typically short, (i.e., 30 or 60 seconds), the road intersection is called a *small social spot*. As a shopping mall usually operates for a whole day, indicating that a number of customers' vehicles will stop at the parking lot for a long period, the free parking lot near the mall is hence called a *large social spot*. Notice that as social spots usually hold many vehicles, if all vehicles indistinguishably change their pseudonyms in the spots, the social spots naturally become *mix zones*.

5.2.2 Threat Model

Unlike other wireless communication devices, the OBU devices equipped on the vehicles cannot be switched off once vehicles are running on the road [82]. Then, an eavesdropper, through the *safety messages* broadcasted by the OBU, can monitor the location information of a specific vehicle all the time. Concretely, in our threat model, we consider a global external adversary $\mathcal{A}$ equipped with radio devices to trace the vehicles' locations, where

- *Global* means the adversary $\mathcal{A}$ has the ability to monitor and collect all *safety messages* in the network with radio devices plus some special eavesdropping infrastructure mentioned in [15], where each safety message includes `Time`, `Location`, `Velocity`, `Content` as well as `Pseudonym`. Since `Pseudonym` is unlinkable and `Content` could be set as irrelevant, the adversary $\mathcal{A}$ primarily tracks a vehicle in terms of `Time`, `Location`, `Velocity`, i.e., in a spatial-temporal way in our model.

[1] We confine our problem to pseudonym changing in the V-2-V communication mode, and do not include Roadside Units (RSUs) in the current network model, although RSUs are still deployed to support V-2-R communication in the urban area.

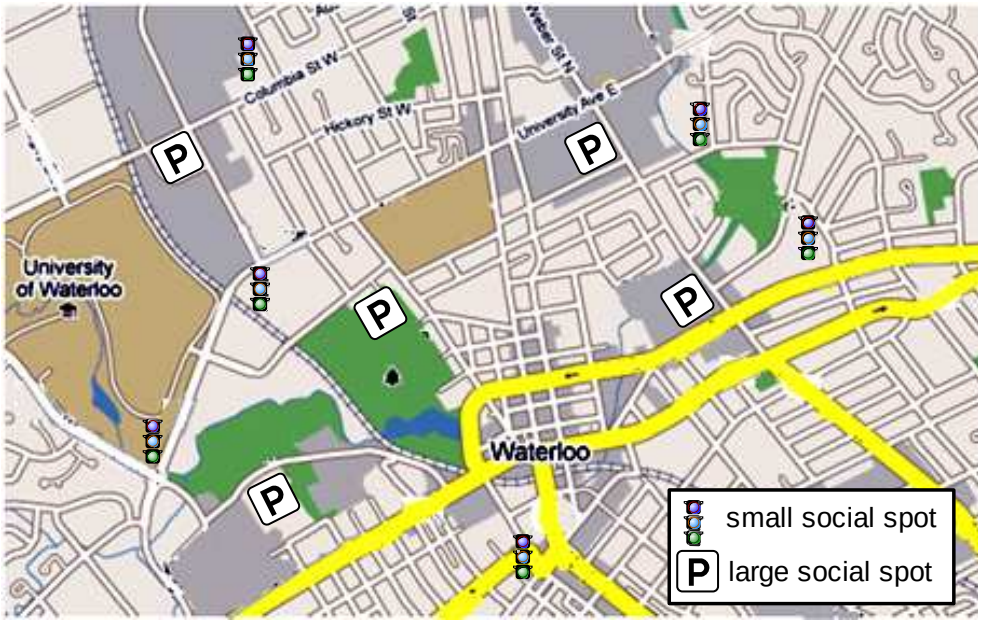


Figure 5.2: Social spots including the road intersection and free parking lots

- *External* denotes the adversary $\mathcal{A}$ can only passively eavesdrop the communications, but does not actively attempt to compromise the running vehicles.

Notice that an adversary $\mathcal{A}$ of course can track vehicles by using cameras in the urban area. However, the cost of *global* eavesdropping with cameras is much higher than that of radio based eavesdropping [80]. Therefore, the camera based global eavesdropping is beyond the scope of this chapter.

5.2.3 Location Privacy Requirements

To resist the global external adversary's tracking and achieve the location privacy in VANETs, the following requirements must be satisfied.

- *R-1.* Identity privacy is a prerequisite for the success of location privacy. Therefore, each vehicle should use pseudonym in place of real identity to broadcast messages. Then, by concealing the real identity, the identity privacy can be achieved.

- *R-2.* Each vehicle should also periodically change its pseudonyms to cut down the relation between the former location and the latter location. In addition, the pseudonyms changing should be performed at the appropriate time and location to ensure that the location privacy is achieved.

- *R-3.* Location privacy should be *conditional* in VANETs. If a broadcasted *safety message* is in dispute, the trusted authority (TA) can disclose the real identity, i.e., TA has the ability to determine the location where a specific vehicle broadcasted a disputed *safety message*.

Recall that the social spots can serve as *mix zones* naturally. In what follows, we explore this feature and propose the PCS strategy for achieving location privacy in VANETs.

5.3 Proposed PCS Strategy for Location Privacy

In this section, we present our PCS strategy for achieving location privacy in VANETs. Specifically, we develop two anonymity set analytic models to investigate the location privacy level achieved in the PCS strategy, and use simplified game theoretical techniques to discuss its feasibility. Before delving into the details of the PCS strategy, we first present a practical key-insulated pseudonym self-delegation (KPSD) model, which securely generates many on-demand short-life keys and serves as the basis of the proposed PCS strategy.

5.3.1 KPSD Model for PCS Strategy

To support the PCS strategy, a vehicle must hold a certain amount of pseudonyms. In [7], a simple and straightforward solution is proposed, where an OBU device equipped on a vehicle possesses a large number of anonymous short-time keys authorized by a Trusted Authority (TA). Obviously, this solution can achieve conditional location privacy when periodically changing the pseudonyms. However, it may take a large storage space to store these short-time keys in OBU device. GSIS [12] is a group signature based technique which can achieve conditional location privacy without pseudonyms changing. However, the pure group signature verification is usually time-consuming which may be not suitable for some time-stringent VANET applications. ECPP [14] is another anonymous authentication technique which combines group signature and ordinary signature. In ECPP, when a legal vehicle passes by an RSU, the RSU will authorize a group signature based short-life anonymous certificate to the vehicle. Then, the vehicle can use it to sign messages with

ordinary signature techniques [83]. Once receiving a signed message, anyone can verify the authenticity of message by checking both the anonymous certificate and message signature. Note that, when the vehicle signs many messages, any verifier only needs execute one group signature verification operation on certificate, thus it is more efficient than GSIS. Similar to ECPP, Calandriello *et al.* [84], inspired by the idea of pseudonymous PKI for ubiquitous computing [62], also combine group signature and ordinary signature techniques to achieve anonymous authentication in VANETs. Because the short-life anonymous certificate is generated by the vehicle itself, their scheme is very flexible. However, once a vehicle is stolen, the vehicle thief can arbitrarily generate valid short-life anonymous certificates before being detected. Then, the potential hazards could be large. To mitigate such negative affects, we propose a practical key insulated pseudonym self-delegation (KPSD) model.

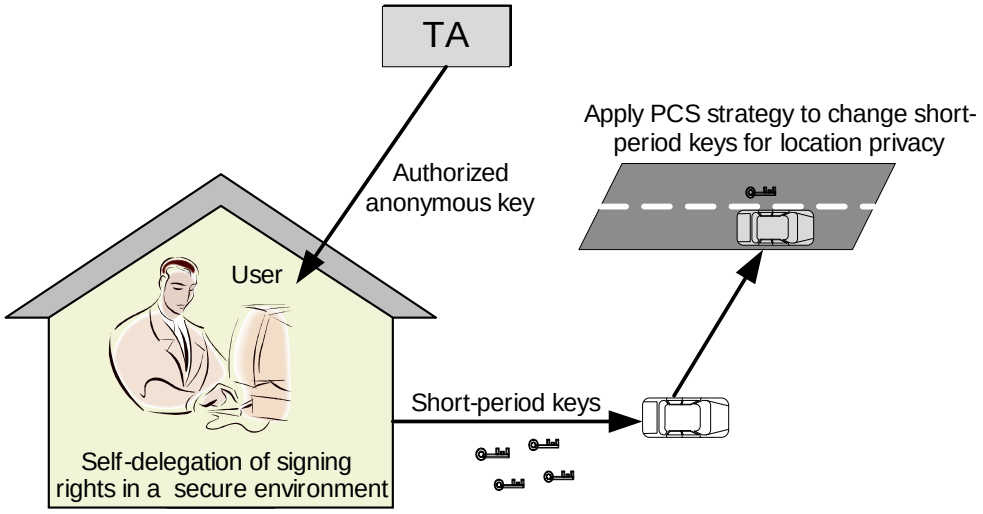


Figure 5.3: Practical KPSD model for location privacy in VANETs

As shown in Fig. 5.3, in KPSD model, TA does not directly preload authorized anonymous key to the vehicle, instead, it provides the authorized anonymous key to the user — the owner of the vehicle. The user usually stores the authorized anonymous key in a secure environment, i.e., at home. When s/he is ready to go out for a travel, like feuling enough gasoline, s/he first generates required self-delegated short-life keys, and installs them in the OBU device. Later, when the vehicle is running in the urban area, these short-life keys can be used to sign messages. Because vehicle theft is still a serious concern currently, e.g., statistics show that there have been over 170,000 vehicles stolen each year in Canada [61], these short-life keys could be abused by the thieves, once the vehicle is stolen. However,

different from previous works [7, 12, 14, 84], the authorized anonymous key in KPSD model is not stored in the vehicle. Thus, the vehicle thieves cannot generate more short-life keys. As a result, the hazards due to vehicle theft can be mitigated in KPSD model. Note that if the authorized anonymous key is protected by a password-based tamper-proof device, Calandriello *et al.*'s scheme [84] can fall into our key insulated pseudonym self-delegation model, but the cost will increase accordingly.

In the following, we construct an efficient KPSD scheme with asymmetric bilinear groups [52], which serves as the basis of the PCS strategy.

Construction

Our proposed KPSD scheme is based on Boneh-Boyen short signature [59] and the conditional privacy preservation authentication technology [14, 24], which mainly consists of the following four parts: system initialization, key generation, pseudonym self-delegated generation, and conditional tracking.

System Initialization: Similar to the notations used in [52], let k be a security parameter, $\mathbb{G}$, $\mathbb{G}'$ and $\mathbb{G}_T$ be three (multiplicative) cyclic groups of the same large prime order q generated by $\mathcal{AG}en(k)$, where $|q| = k$. Suppose $\mathbb{G}$, $\mathbb{G}'$ and $\mathbb{G}_T$ are equipped with a pairing, i.e., a non-degenerated and efficiently computable bilinear map $e : \mathbb{G} \times \mathbb{G}' \rightarrow \mathbb{G}_T$ such that $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab} \in \mathbb{G}_T$ for all $a, b \in \mathbb{Z}_q^*$ and any $g_1 \in \mathbb{G}_1, g_2 \in \mathbb{G}_2$. We denote by ψ the isomorphism from $\mathbb{G}'$ onto $\mathbb{G}$, that we assume to be one-way (easy to compute, but hard to invert). TA first chooses two random numbers $u, v \in \mathbb{Z}_q^*$ as the *master-key*, and computes $U_1 = {g_1}^u$, $U_2 = {g_2}^u$, and $V_1 = {g_1}^v$. In addition, TA also chooses a public collision-resistant hash function: $H : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$. In the end, TA publishes the system parameters $params = (q, \mathbb{G}, \mathbb{G}', \mathbb{G}_T, e, g_1, g_2, U_1, U_2, V_1, H)$.

Key Generation: When a user $\mathcal{U}_i$ with identity ID_i joins the system, TA first chooses a random number $s_i \in \mathbb{Z}_q^*$ such that $s_i + u \neq 0 \bmod q$, computes $A_i = g_1^{\frac{1}{s_i+u}}$. Then, TA stores (ID_i, A_i^u) in the tracking list and returns $ASK_i = (s_i, A_i = g_1^{\frac{1}{s_i+u}})$ as the authorized anonymous key to the user.

Pseudonym Self-Delegated Generation: After receiving the authorized anonymous key ASK_i, $\mathcal{U}_i$ places it in a secure environment (e.g., at home). When $\mathcal{U}_i$ starts to travel in the city, he first runs the following steps to generate the required anonymous short-life keys used for the travel, which is very analogous to the fueling of a vehicle before a travel.

1. $\mathcal{U}_i$ first chooses l random numbers $x_1, x_2 \cdots, x_l \in \mathbb{Z}_n^*$ as the short-life private keys

and computes the corresponding public keys $Y_j = g^{x_j}$, for $j = 1, 2, \cdots, l$ for the travel.

2. For each short-life public key Y_j, $\mathcal{U}_i$ computes the anonymous self-delegated certificate $Cert_j$ as follows

   - Randomly choose $\alpha, r_\alpha, r_x, r_\delta \in \mathbb{Z}_q^*$ and compute $T_U, T_V, \delta, \delta_1, \delta_2, \delta_3$, where

   $$\begin{cases} T_U = {U_1}^{\alpha}, T_V = A_i \cdot {V_1}^{\alpha}, \delta = \alpha \cdot x_i \bmod q \\ \delta_1 = {U_1}^{r_\alpha}, \delta_2 = T_U^{r_x} / {U_1}^{r_\delta} \\ \delta_3 = e(T_V, {g_2}^{r_x}) / e(V_1, {U_2}^{r_\alpha \cdot {g_2}^{r_\delta}}) \end{cases} \tag{5.1}$$

   - Compute $c = H(U_1||V_1||Y_j||T_U||T_V||\delta_1||\delta_2||\delta_3)$ and s_α, s_x, $s_\delta \in \mathbb{Z}_q^*$, where

   $$\begin{cases} s_\alpha = r_\alpha + c \cdot \alpha \bmod q, s_x = r_x + c \cdot x_i \bmod q \\ s_\delta = r_\delta + c \cdot \delta \mod q \end{cases} \tag{5.2}$$

   - Set $Cert_j = \{Y_j||T_U||T_V||c||s_\alpha||s_x||s_\delta\}$ as the certificate.

3. After all anonymous self-delegated certificates $Cert_j$, $j = 1, 2, \cdots, l$, are generated, $\mathcal{U}_i$ installs them to the vehicle, i.e., implanting all $x_j||Y_j||Cert_j$, $j = 1, 2, \cdots, l$, into the OBU device.

Later, when $\mathcal{U}_i$ is driving the vehicle in the city, he can use one short-life key $x_j||Y_j||Cert_j$ to authenticate a message M by signing $\sigma = {g_2}^{\frac{1}{x_j + H(M)}}$, and broadcast

$$msg = (M||\sigma||Y_j||Cert_j) \tag{5.3}$$

Upon receiving $msg = (M||\sigma||Y_j||Cert_j)$, everyone can check the validity by the following.

1. If the certificate $Y_j||Cert_j$ has not been checked, the verifier first computes

   $$\begin{cases} \delta_1' = U_1^{s_\alpha} / {T_U}^c, \delta_2' = {T_U}^{s_x} / U_1^{s_\delta} \\ \delta_3' = \frac{e(T_V, {g_2}^{s_x} \cdot {U_2}^c)}{e(V_1, {U_2}^{s_\alpha} \cdot {g_2}^{s_\delta}) e(g_1, {g_2}^c)} \end{cases} \tag{5.4}$$

   and checks whether

   $$c = H(U_1||V_1||Y_j||T_U||T_V||\delta_1'||\delta_2'||\delta_3') \tag{5.5}$$

   If it does hold, the certificate $Y_j||Cert_j$ passes the verification. The corrections are as follows: i) $\delta_1' = U_1^{s_\alpha} / T_U^c = U_1^{r_\alpha + c \cdot \alpha} / U_1^{c \cdot \alpha} = \delta_1$; ii) $\delta_2' = T_U^{s_x} / U_1^{s_\delta} = T_U^{r_x + c x_i} / U_1^{r_\delta + c\delta} = \delta_2$; iii) $\delta_3' = e(T_V, {g_2}^{s_x} \cdot {U_2}^c) / e(V_1, {U_2}^{s_\alpha} \cdot {g_2}^{s_\delta}) e(g_1, {g_2}^c) = e(T_V, {g_2}^{r_x}) / e(V_1, {U_2}^{r_\alpha} \cdot {g_2}^{r_\delta}) = \delta_3$.

2. Once the certificate $Y_j||Cert_j$ passes the verification, the verifier checks

$$e(Y_j \cdot {g_1}^{H(M)}, \sigma) \stackrel{?}{=} e(g_1, g_2) \tag{5.6}$$

If it holds, the message M is accepted, otherwise, M is rejected, since $e(Y_j \cdot {g_1}^{H(M)}, \sigma) = e(g_1^{x_j+H(M)}, {g_2}^{\frac{1}{x_j+H(M)}}) = e(g_1, g_2)$. Note that the value of $e(g_1, g_2)$ can be pre-computed in advance.

Conditional Tracking: Once an accepted message M under the certificate

$$Cert_j = \{Y_j||T_U||T_V||c||s_\alpha||s_x||s_\delta\}$$

is disputed, TA uses the master key (u, v) to compute

$$T_V^u / T_U^v = A_i^u \cdot V_1^{u\alpha} / U_1^{v\alpha} = A_i^u \cdot g^{uv\alpha} / g^{uv\alpha} = A_i^u \tag{5.7}$$

and then can efficiently trace the real identity ID_i by looking up the entry (ID_i, A_i^u) in the tracking list.

Security

Since both the short signature [59] and conditional privacy preservation authentication [14] are secure, the security of the proposed KPSD scheme can be guaranteed, i.e., it can effectively achieve anonymous authentication with conditional tracking to fulfill the requirements of location privacy. In addition, the proposed KPSD scheme can also mitigate the hazards due to vehicle theft, since the authorized anonymous key ASK_i is *key-insulated*, i.e., it is stored in a secure environment, then vehicle thieves can not obtain ASK_i from the stolen vehicle, and consequently can not generate new self-delegated short-life keys arbitrarily.

Performance

In VANETs, it is a very challenging issue for a vehicle to verify too many signed messages in a stringent time, e.g., within 300 msec. Let T_{pair}, $T_{\mathsf{exp\text{-}1}}$, $T_{\mathsf{exp\text{-}2}}$ be the time costs for pairing operation, exponentiation in $\mathbb{G}$ and $\mathbb{G}'$, respectively. Then, to check n messages from the same source, where $n \geq 1$, the verification cost of the proposed KPSD anonymous authentication and the pure group signature-based (GSB) anonymous authentication are $(3+n)T_{\mathsf{pair}} + (4+n)T_{\mathsf{exp\text{-}1}} + 5T_{\mathsf{exp\text{-}2}}$ and $3nT_{\mathsf{pair}} + 4nT_{\mathsf{exp\text{-}1}} + 5nT_{\mathsf{exp\text{-}2}}$, respectively. Since T_{pair} is dominant over $T_{\mathsf{exp\text{-}1}}$ and $T_{\mathsf{exp\text{-}2}}$, we set T_{pair} as 4.5 ms as in [14] and make the comparison in Fig. 5.4. Clearly, it can be seen, when n is large, the proposed anonymous authentication is much more efficient than the pure GSB anonymous authentication.

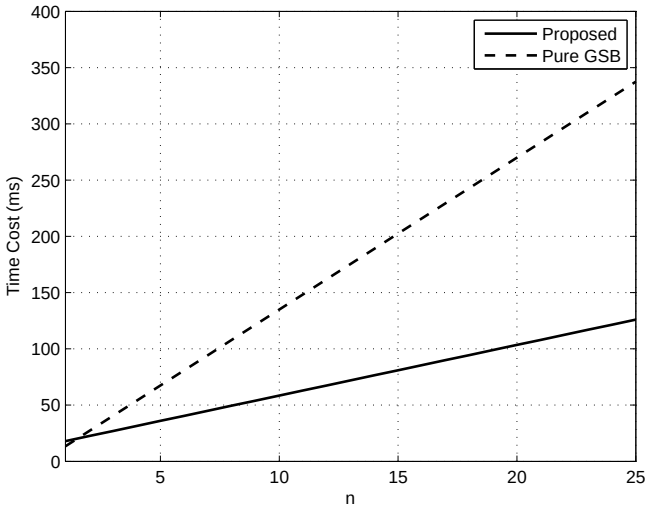


Figure 5.4: Time cost comparison

Algorithm 5 Pseudonym Changing at Social Spots Strategy

1: **procedure** PCS STRATEGY
2: **Case 1:** Small social spot
3: A vehicle V_i stops at road intersection when the traffic light turns red. When the traffic light turns to green, V_i changes its pseudonym.
4: **Case 2:** Large social spot
5: A vehicle V_i stops at a free parking lot near a shopping mall. When leaving the parking lot, V_i changes its pseudonym.
6: **end procedure**

5.3.2 Anonymity Set Analysis for Achieved Location Privacy

With the above KPSD scheme, each vehicle can hold a number of pseudonyms on the road, then it can apply the PCS strategy, as shown in Algorithm 5, to protect its location privacy. To gauge the benefits from the PCS strategy, we next develop two anonymity set analytic models to investigate the location privacy achieved in small social spots and large social spots, respectively.

Anonymity set analysis at small social spots

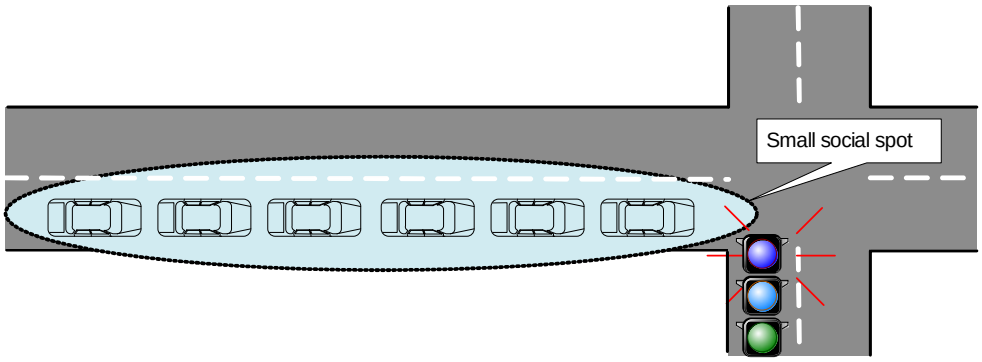


Figure 5.5: Pseudonym changing at an intersection

As shown in Fig. 5.5, when the traffic light turns red, the road intersection can be regarded as a *small social spot*, since a fleet of vehicles will stop at the intersection [80]. Consider all vehicles will simultaneously change their pseudonyms when the traffic light turns to green. Then, the road intersection naturally becomes a *mix zone*. Let S_a be the number of vehicles stopped at the intersection, we will have the expected anonymity set size (ASS) = S_a. Clearly, the larger the anonymity set size ASS is, the greater the anonymity is offered in the small social spot. We can use a trivial anonymity set analytic model on ASS to investigate the anonymity level provided by the small social spot.

Let $T_s = t$, where $t = 30, 60$ seconds, be the fixed stop time period of a specific road intersection. Let *vehicle arrival* (VA) at the road intersection be a Poisson process, and t_a be the inter-arrival time for VA, where t_a has an exponential distribution with the mean $\frac{1}{\lambda}$. Let X be the random variable of vehicles arriving at the road intersection during the period T_s. Then, based on [85, 86], the probability $X = x$ during $T_s = t$ can be expressed as

$$\Pr[X = x | T_s = t] = \frac{(\lambda t)^x}{x!} e^{-\lambda t} \tag{5.8}$$

and the expected number of X can be computed as

$$\mathsf{E}[X|T_s = t] = \sum_{x=1}^{\infty} x \Pr[X = x|T_s = t] = \lambda t \tag{5.9}$$

Since all vehicles leave the intersection after the traffic light turns to green[2], the anonymity set size ASS is

$$ASS = S_a = \mathsf{E}[X|T_s = t] = \lambda t \tag{5.10}$$

if all vehicles follow the PCS strategy.

Anonymity set analysis at large social spots

As shown in Fig. 5.6, a large social spot could be a free parking lot near a shopping mall [61]. Because a parking lot usually holds many vehicles, and each vehicle randomly leaves the parking lot at the user own will, such a parking lot also naturally becomes a *mix zone* if all users change their pseudonyms in the parking lot and leave the parking lot after a random delay. Because a parking lot can obfuscate the relation between the arriving and leaving vehicles, the location privacy of user can be achieved.

Let S_a be the number of vehicles in the parking lot when a vehicle is ready to leave. Then, the anonymity set size denotes ASS $= S_a$. In the following, we propose an anonymity analytic model on ASS to investigate the anonymity level provided by the large social spot.

For a specific vehicle $\mathcal{V}$ that has entered a parking lot near a shopping mall for changing pseudonyms, we consider the time period from the mall's opening time, e.g., 8:00 AM, to the vehicle $\mathcal{V}$'s leaving time after pseudonyms changing, T_S, as shown in Fig. 5.7, is exponentially distributed with the density function $f(t)$, the mean $\frac{1}{\mu}$, and the Laplace transform $f^*(s) = \left(\frac{\mu}{\mu+s}\right)$. On the other hand, other vehicles enter/leave a parking lot at the drivers' own willing, for example, a driver determines when and how long he will stop at the mall. Let *vehicle arrival* (VA) at the parking lot be a Poisson process, and t_a be the inter-arrival time for VA. Then, t_a has an exponential distributions with the mean $\frac{1}{\lambda}$. In addition, the time period between the time when a vehicle arrives at the parking lot

[2]Note that when the number of waiting vehicles is larger than some threshold, only part of the waiting vehicles can leave the intersection after the traffic light turns to green, and some vehicles have to wait for the next green light. In this case, the number of waiting vehicles (N_v) can be regarded as the initial value for the next anonymity set size at intersection, i.e., $ASS = N_v + \lambda t$.

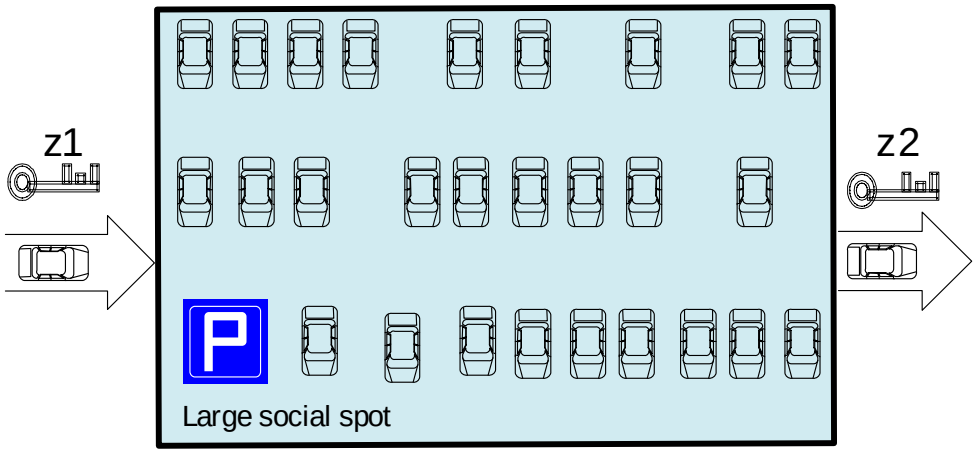


Figure 5.6: Pseudonym changing at a free parking lot

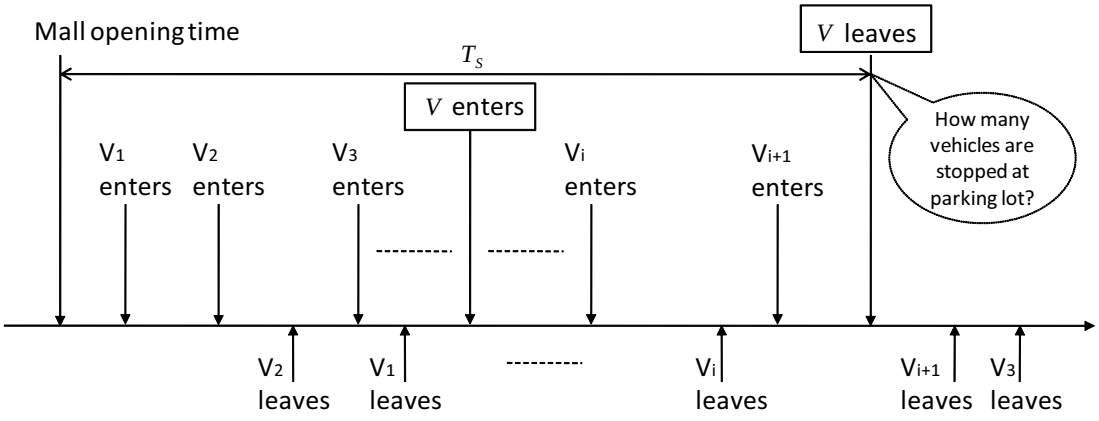


Figure 5.7: Timing diagram (there is no vehicle stopping in the parking lot initially)

and the time when it leaves, t_u, is assumed having the density function $f_u(\cdot)$, the mean $\frac{1}{\omega}$ and the Laplace transform $f_u^*(s)$. Let X be the random variable of vehicles arriving at the parking lot during the time period T_s. Then, the probability $X = x$ during the period $T_s = t$ follows $\Pr[X = x|T_s = t] = \frac{(\lambda t)^x}{x!}e^{-\lambda t}$, and for $t \geq 0$,

$$\begin{aligned}
\Pr[X = x] &= \int_{t=0}^{\infty} \Pr[X = x|T_s = t] f(t) dt \\
&= \int_{t=0}^{\infty} \frac{(\lambda t)^x}{x!} e^{-\lambda t} f(t) dt \\
&= \left(\frac{\lambda^x}{x!}\right) \int_{t=0}^{\infty} t^x e^{-\lambda t} f(t) dt \\
&= \left(\frac{\lambda^x}{x!}\right) \left[(-1)^x \frac{d^x f^*(s)}{ds^x}\right]\Big|_{s=\lambda} \\
&= \frac{\mu \lambda^x}{(\mu + \lambda)^{x+1}}
\end{aligned} \tag{5.11}$$

and the expected number of X can be computed as

$$\mathsf{E}[X] = \sum_{x=1}^{\infty} x \Pr[X = x] = \frac{\lambda}{\mu} \tag{5.12}$$

Let χ be the time period between the time when a vehicle arrives at the parking lot and the time when the specific vehicle $\mathcal{V}$ leaves the parking lot after pseudonyms changing. Since T_s is exponentially distributed, the density function $\sigma(\chi)$ for the distribution χ can be expressed as

$$\sigma(\chi) = \mu \int_{t=\chi}^{\infty} f(t) dt = \mu[1 - F(t)]\Big|_{t=\chi} = \mu e^{-\mu\chi} \tag{5.13}$$

During the period T_s, many vehicles may leave the parking lot before $\mathcal{V}$'s leaving, i.e., $t_u < \chi$, while others leave after $\mathcal{V}$, i.e., $t_u \geq \chi$. Assume that Y is the number of vehicles leaving the parking lot before $\mathcal{V}$, then the probability $\Pr[Y = y|X = x]$ can be computed as

$$\Pr[Y = y|X = x] = \binom{x}{y} (\Pr[t_u < \chi])^y (\Pr[t_u \geq \chi])^{x-y} \tag{5.14}$$

Then, the probability $\Pr[t_u \geq \chi]$ can be calculated as

$$\begin{aligned}\Pr[t_u \geq \chi] &= \int_{t_u=0}^{\infty} \int_{\chi=0}^{t_u} \mu e^{\mu\chi} d\chi f_u(t_u) dt_u \\ &= \int_{t_u=0}^{\infty} (1 - e^{-\mu t_u}) f_u(t_u) dt_u \\ &= 1 - \int_{t_u=0}^{\infty} f_u(t_u) e^{-\mu t_u} dt_u = 1 - f_u^*(\mu)\end{aligned} \tag{5.15}$$

and $\Pr[t_u < \chi]$ can be derived from $\Pr[t_u \geq \chi]$ as

$$\Pr[t_u < \chi] = 1 - \Pr[t_u \geq \chi] = 1 - (1 - f_u^*(u)) = f_u^*(u) \tag{5.16}$$

After that, Eq. (5.14) can be rewritten as

$$\Pr[Y = y | X = x] = \binom{x}{y} (f_u^*(u))^y (1 - f_u^*(u))^{x-y} \tag{5.17}$$

and the expected number of Y can be computed as

$$\begin{aligned}\mathsf{E}[Y] =& \sum_{x=1}^{\infty} \sum_{y=1}^{x} \{ y \Pr[Y = y | X = x] \Pr[X = x] \} \\ =& \sum_{x=1}^{\infty} \left\{ \left\{ \sum_{y=1}^{x} y \binom{x}{y} (f_u^*(u))^y (1 - f_u^*(u))^{x-y} \right\} \right. \\ & \left. \times \left[\frac{\mu \lambda^x}{(\mu + \lambda)^{x+1}} \right] \right\}\end{aligned} \tag{5.18}$$

Therefore, the expected anonymity set size ASS for the specific vehicle $\mathcal{V}$'s pseudonyms changing is

$$\begin{aligned}ASS =& S_a = \mathsf{E}[X] - \mathsf{E}[Y] \\ =& \frac{\lambda}{\mu} - \sum_{x=1}^{\infty} \left\{ \left\{ \sum_{y=1}^{x} y \binom{x}{y} (f_u^*(u))^y (1 - f_u^*(u))^{x-y} \right\} \right. \\ & \left. \times \left[\frac{\mu \lambda^x}{(\mu + \lambda)^{x+1}} \right] \right\}\end{aligned} \tag{5.19}$$

Since the exponential distribution has been widely used in modeling many realistic scenarios [85], we assume that t_u also follows the exponential distribution. Then, the

Laplace transform $f_u^*(u)$ becomes

$$f_u^*(u) = \left(\frac{\omega}{\omega+\mu}\right) \tag{5.20}$$

As a result, S_{anony} can be rewritten as

$$\begin{aligned} ASS =& \frac{\lambda}{\mu} - \sum_{x=1}^{\infty}\left\{\left\{\sum_{y=1}^{x} y\binom{x}{y}\left(\frac{\omega}{\omega+\mu}\right)^{y}\left(1-\frac{\omega}{\omega+\mu}\right)^{x-y}\right\}\right. \\ & \left.\times\left[\frac{\mu\lambda^x}{(\mu+\lambda)^{x+1}}\right]\right\} \\ =& \frac{\lambda}{\mu} - \sum_{x=1}^{\infty}\left\{x\cdot\frac{\omega}{\omega+\mu}\times\left[\frac{\mu\lambda^x}{(\mu+\lambda)^{x+1}}\right]\right\} \\ =& \frac{\lambda}{\mu} - \frac{\omega\mu}{(\omega+\mu)(\mu+\lambda)}\sum_{x=1}^{\infty} x\cdot\left(\frac{\lambda}{\mu+\lambda}\right)^{x} \\ =& \frac{\lambda}{\mu} - \frac{\omega\lambda}{\mu(\omega+\mu)} = \frac{\lambda}{\omega+\mu} \end{aligned} \tag{5.21}$$

5.3.3 Feasibility Analysis of PCS Strategy

The above anonymity set analyses are under the assumption that all vehicles change their pseudonyms. In this subsection, we use the simplified game theoretic techniques to show the feasibility of PCS strategy, i.e., we prove that each vehicle is really willing to change the pseudonym at social spots for achieving its location privacy in practice.

Let the anonymity set size ASS be $N = n + 1$, where $n \geq 0$, at social spots, which can be estimated by the above anonymity set analysis. Then, we investigate the scenario where all vehicles are rational to protect their location privacy. At social spots, each vehicle V_j, $1 \leq j \leq N$ has two possible actions: change (C) the pseudonym with probability p_j and keep (K) the pseudonym with probability $1 - p_j$. If V_j keeps its pseudonym at the social spot, it will still be tracked with probability 1. Then, the loss of V_j's location privacy is unchanged, and the payoff in this action is a normalized location privacy loss of $-d_j$, where $d_j \in (0, 1)$ is the V_j's self-evaluation on the importance of location privacy. On the other hand, when V_j changes its pseudonym at the social spot, if there are other vehicles taking the same action as well, the anonymity set size will become S. After this social spot, V_j remains being tracked only with probability $\frac{1}{S}$. As such, the loss of location privacy in this

case is reduced to $-\frac{d_j}{S}$. Let $c_j \in (0,1)$ be V_j's normalized cost of changing a pseudonym, so the payoff in this action is $-\frac{d_j}{S} - c_j$. For all vehicles except V_j, let p_m be the minimum of all probabilities $\{p_i | 1 \leq i \leq N, i \neq j\}$. Then, when V_j is ready to change its pseudonym at social spots, it can estimate the low bound of average anonymity set as

$$\begin{aligned} S &= \sum_{i=0}^{n} \binom{n}{i} \cdot p_m^i \cdot (1-p_m)^{n-i} \cdot (i+1) \\ &= np_m + 1 \end{aligned}$$

As a result, the payoff function of vehicle V_j can be summarized as

$$Payoff = \begin{cases} -\frac{d_j}{np_m+1} - c_j, & \text{if action C is taken;} \\ -d_j, & \text{else if action K is taken.} \end{cases} \tag{5.22}$$

Since vehicle V_j is rational and its goal is to protect its location privacy, the condition that V_j changes its pseudonym at the social spot is

$$-\frac{d_j}{np_m+1} - c_j > -d_j \Rightarrow c_j < \frac{np_m d_j}{np_m+1} \tag{5.23}$$

With the adopted KPSD scheme, all vehicles generate and manage their pseudonyms by themselves, they can generate enough pseudonyms before a travel, then the cost of changing pseudonym can be very low. Nevertheless, when np_m is 0, Eq. (5.23) does not hold, which indicates when there is no neighboring vehicle changing its pseudonym, V_j also does not change its pseudonym. However, when np_m is large than 0, V_i is always able to reduce the cost c_j such that $c_j < \frac{np_m d_j}{np_m+1}$. Then, V_j can actively change the pseudonym at social spots. We define each vehicle V_j's location privacy gain (LPG) function as

$$LPG_j = -\frac{d_i}{np_m+1} - (-d_i) = \frac{np_m}{np_m+1} \cdot d_j$$

Then, LPG_j is an increase function in terms of p_m. When $p_m = 1$, i.e., all vehicles change their pseudonyms at social spots, LPG_j can reach its maximal gain $\frac{n}{n+1} \cdot d_j = \frac{(N-1)}{N} \cdot d_j$. Since each vehicle is rational to maximize its location privacy gain, it would be a win-win situation when they all change their pseudonyms. As a result, the feasibility of PCS strategy in practice is shown.

5.4 Performance Evaluation

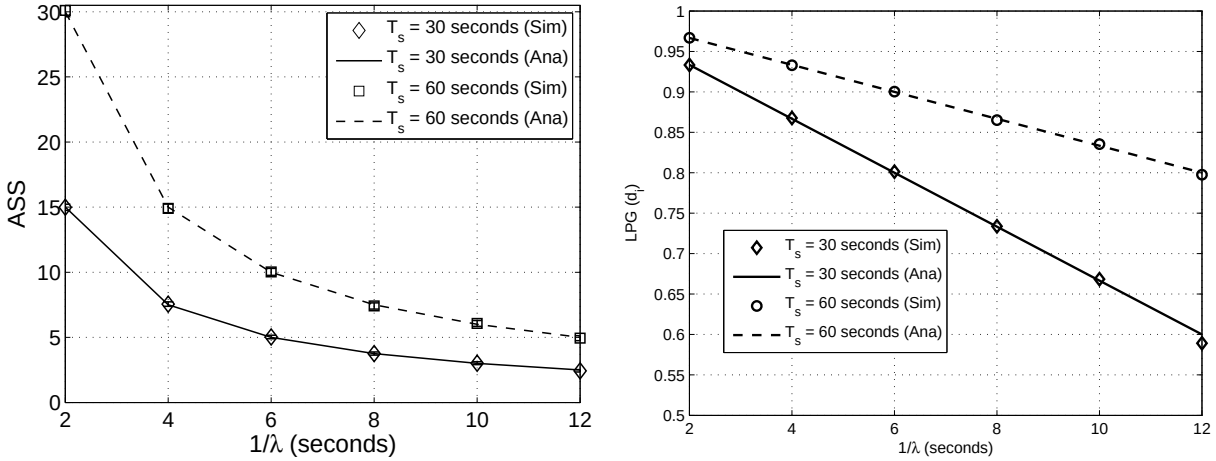
In this section, we evaluate the location privacy level achieved in the PCS strategy. In particular, extensive simulations are conducted to demonstrate the impacts of different parameters on the performance metrics in terms of the anonymity set size (ASS) and location privacy gain (LPG). Our simulations are based on a discrete event simulator coded in C++, where the simulation parameters are listed in Table 5.1 for two scenarios: the small social spot and the large social spot. For each case, we repeat the simulation 100 times with different random seeds and calculate the average value with 95% confidence intervals. In addition, we compare the simulation results (denoted as Sim) with the numerical ones (denoted as Ana) to validate the developed analytical models.

Table 5.1: Simulation Settings in PCS

Parameter	Values
T_S: time period at small social spot	30, 60 seconds
$1/\lambda$: at small social spot	$[2, 4, 6, 8, 10, 12]$ seconds
$1/\mu$: mean of T_S at large social spot	$[1, 2, \cdots, 10]$ hours
$1/\lambda$: at large social spot	$[2, 4, 6]$ minutes
$1/\omega$: at large social spot	$[10, 20, \cdots, 90]$ minutes
d_i: a vehicle's self-evaluation on the importance of its location privacy	normalized

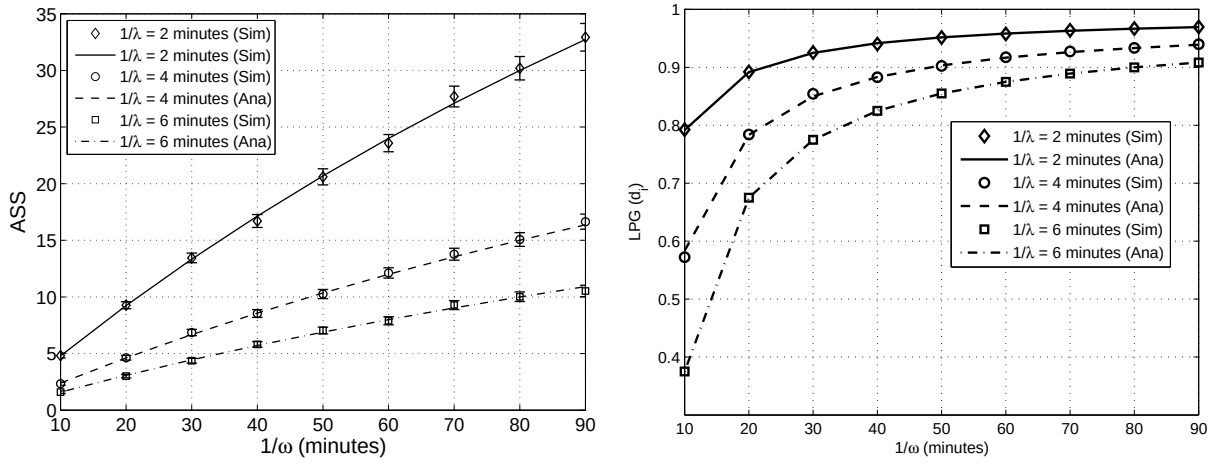
We first validate the location privacy level achieved at small social spot, i.e., a road intersection when the traffic light turns red. Consider the stopping time period $T_S = 30, 60$ seconds for a low traffic intersection and a high traffic intersection, respectively. Fig. 5.8 shows the ASS and LPG versus $1/\lambda$ varying from 2 seconds to 10 seconds with increase of 2. From the figure, it can be seen that ASS and LPG decrease with the increase of $1/\lambda$. The reason is that with a large $1/\lambda$, less vehicles drive at the road intersection when traffic light is red, which leads to a small number of vehicles gather at the intersection, as a result, it causes a smaller ASS as well as a lower LPG. In addition, a large T_S also has a positive impact on ASS and LPG. Therefore, to achieve a high location privacy level, a large intersection with high traffic is a good choice for vehicles, which tallies with our common sense.

To evaluate the location privacy level achieved at large social spot, we consider a free parking lot near a shopping mall. Parameterized with $1/\mu = 4$ hours, Fig. 5.9 shows the



(a) ASS versus $1/\lambda$ (b) LPG versus $1/\lambda$

Figure 5.8: ASS and LPG versus $1/\lambda$ with different T_S at small social spot

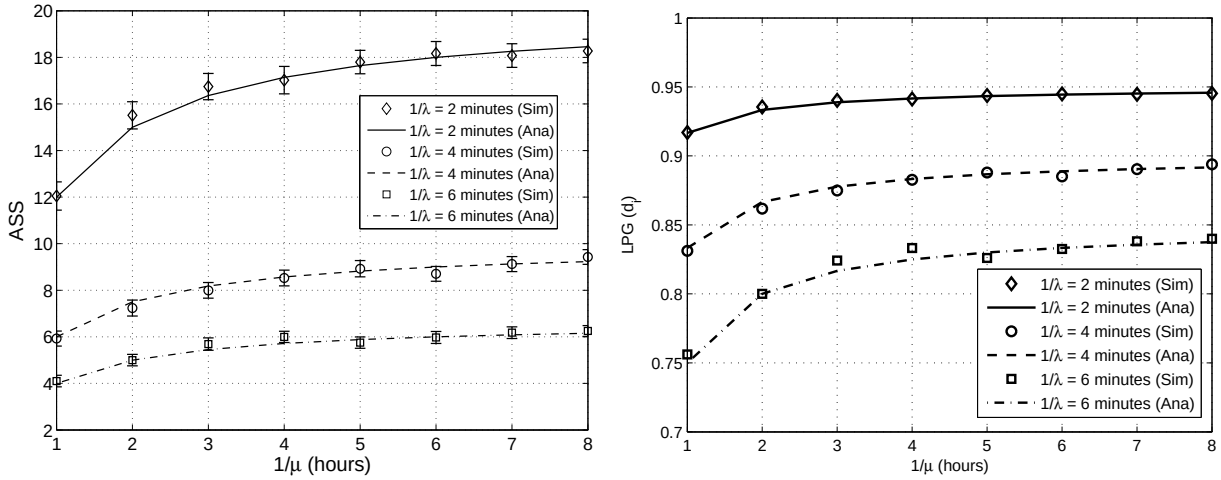


(a) ASS versus $1/\omega$ (b) LPG versus $1/\omega$

Figure 5.9: ASS and LPG versus $1/\omega$ with $1/\mu = 4$ hours at large social spot

impacts of $1/\omega$ on the performance metrics in terms of ASS and LPG. From the figure, it can be seen, as $1/\omega$ increases, both ASS and LPG also increase. The reason is that the larger $1/\omega$, the more vehicles will park at the parking lot. In addition, the smaller $1/\lambda$ also achieves a larger ASS and a higher LPG. Therefore, when a vehicle changes its pseudonyms in a parking lot near a prosperous shopping mall (with small $1/\lambda$ and large $1/\omega$), the high location privacy level can be guaranteed. From the figure, it can also be seen that the simulation and analysis results match very well, which justifies the accuracy of the analytical model.

Fig. 5.10 shows the impact of the parameter $1/\mu$ on ASS and LPG. We can see, except the first two hours, with the increase of $1/\mu$, both ASS and LPG smoothly increase. The results indicate that a vehicle can change its pseudonyms at most of daytime for better location privacy at large social spot, no matter in the morning or afternoon. In the figure, the gaps between the simulation results and the analytical results are small, which can be further reduced if larger number of simulation runs are conducted.



(a) ASS versus $1/\mu$

(b) LPG versus $1/\mu$

Figure 5.10: ASS and LPG versus $1/\mu$ with $1/\omega = 40$ minutes at large social spot

5.5 Related Work

There have been a few prior efforts on frequently changing pseudonyms in mix zones to achieve location privacy in VANETs. In the following, some research works closely related

to ours are reviewed. In [87], Gerlach proposes an approach, called *context mix*, to protect the location privacy of vehicles. In *context mix*, a vehicle permanently assesses its neighborhood, and changes its pseudonyms only if the vehicle detects k vehicle with a similar direction in a confusion radius. The *context mix* is an intuitive approach for achieving location privacy in VANETs. However, how to detect k vehicles in neighborhood and how to guarantee neighboring vehicles to react similarly should be further exploited. In [78], Li *et al.* propose two user-centric location tracking mitigation schemes called *Swing* and *Swap*, where *Swing* can increase location privacy by enabling the nodes to loosely synchronize updates when changing their velocity, and *Swap* enables the vehicle to exchange their identifiers to potentially maximize the location privacy provided by each update. In [74], Butyan *et al.* define a model to study the effectiveness of changing pseudonyms to provide location privacy in VANET. Concretely, they characterize the tracking strategy of the adversary in the model, and introduce a metric to quantify the level of location privacy enjoyed by the vehicles. Additionally, they also use extensive simulations to study the relationship between the strength of the adversary model and the level of the privacy achieved by changing pseudonyms. In [80], Freudiger *et al.* use cryptographic techniques to create mix zones at road intersections and combine these mix zones into vehicular mix networks, then leverage on the mobility of the vehicles and the dynamics of road intersections to mix vehicle identifiers. Finally, they evaluate the effectiveness of the proposed mix system by simulations. Different from the above works, our PCS strategy suggests the vehicles to change pseudonyms at social spots (as mix zones), to maximize the location privacy, and theoretically analyze the achieve location privacy.

In the research line of the placement of mix zones, Freduiger *et al.* [88] analyze the optimal placement of mix zones with combinational optimization techniques, and show that the optimal mix zone placement performs comparatively well to the fully deployment scenarios. This work is instructive, which guides the placement of mix zones in VANETs. In our PCS strategy, due to the characteristics of social spots, and at the same time, since the KPSD model can provide each vehicle enough secure pseudonyms for changing, social spots are in nature of mix zones for achieving better location privacy.

The size of the anonymity set and the entropy of the anonymity set are two popular quantitative measurements of location privacy in VANETs [89]. Following Beresford and Stajano's seminal work [75], the location privacy of a vehicle corresponding to a pseudonyms changing (PC) event is the entropy of $P_{i\to PC}$, i.e., $H(PC) = -\sum_{i=1}^{N} P_{i\to PC} \cdot \log_2(P_{i\to PC})$, where $P_{i\to PC}$ is the probability of the mapping of a vehicle i to PC event and N is the total number of vehicles in the mix zone. When N increases, and $P_{i\to PC}$ is uniformly distributed, i.e., $P_{i\to PC} = 1/N$, the entropy reaches the maximum $H(PC) = \log_2 N$. Therefore, when pseudonyms changing events are indistinguishable in social spots, both

the size and the entropy of the anonymity set size can measure the achieved location privacy. In this work, our PCS strategy adopts anonymity set size as the metric, and focuses on developing anonymity set analytical models to investigate the location privacy level.

In [90], Freudiger *et al.* observe that self-interested mobile nodes may not cooperate in changing pseudonyms in mix zone and would jeopardize the achieved location privacy. To address this issue, they use the game-theoretical techniques to analyze the non-cooperative behavior of mobile nodes. In our PCS strategy, we also use game theory to analyze the feasibility. Since the adopted KPSD scheme provides each vehicle with enough pseudonyms, each vehicle is willing to change its pseudonym at social spot for achieving better location privacy. As a result, the feasibility is easily analyzed.

5.6 Summary

In this chapter, we have proposed an effective pseudonym changing at social spots (PCS) strategy for location privacy in VANETs. In particular, we developed two anonymity set analytical models in terms of ASS to formally analyze the achieved location privacy level, and we used game theoretic techniques to prove its feasibility. In addition, we introduced a practical KPSD model to mitigate the hazards caused by vehicle theft. To the best of our knowledge, most previously reported works on *mix-zone* based pseudonyms changing *only* use the simulations to evaluate the achieved location privacy. Therefore, our analytical models on location privacy at social spot *shed light on* this research line.

Chapter 6

Privacy-preserving Protocol for Finding Like-minded Vehicles

6.1 Introduction

Vehicular Ad hoc Networks (VANETs), as a special instantiate of mobile ad hoc network, have been subject to extensive research efforts not only from the government, but also from the academia and automobile industry in recent years [12]. In VANETs, each vehicle is equipped with OnBoard Unit (OBU) device, which allows them communicate with each other, i.e., vehicle-to-vehicle (V-2-V) communication, as well as to the Roadside Units (RSUs), i.e., vehicle-to-infrastructure (V-2-I) communication. Compared with traditional ad hoc networks, the hybrid of V-2-V and V-2-I communications makes VANETs more promising, and can provide a board of safety-related (e.g., emergence report, collision warning) and non-safety-related (vehicle chatting, downloading and sharing files on the road) applications close to our daily lives. Due to these salient applications, VANETs have been increasingly attractive to the public.

Vehicle chatting is one of the most promising applications in VANETs, which allows vehicles moving along the same road to chat with each other on some topics of common interest, for the purpose of passing the time during the commute or asking for a help on the road [91]. However, the success of vehicle chatting application in VANETs still hinges up the fully understanding and managing the security and privacy challenges that the public concerns, for example, the identity privacy, location privacy, and interest privacy. Because VANETs are usually implemented in civilian scenarios, where the locations of vehicles are tightly related to people who drive them. If the vehicle chatting application

discloses the vehicle's identity privacy and location privacy, it cannot be accepted by the public. In recent years, these two kinds of privacy have been deeply discussed in VANETs [12, 14, 24]. However, to the best of our knowledge, the interest privacy, as a special privacy requirement in vehicle chatting application, has not been explored. Therefore, how to identify a vehicle who is like-minded and establish a shared session key for secure chatting, and how to prevent other vehicles who are not like-minded from knowing one vehicle's interest have become two newly emerging privacy challenges in vehicle chatting application.

In this chapter, to address the above challenging privacy issues in vehicle chatting application, we propose an efficient privacy-preserving finding like-minded vehicle protocol, called FLIP [47], which allows two vehicles with the common interest to identify each other and establish a shared session key, and at the same time, protects their Interest-Privacy (IP) from other vehicles who do not have the same interest on the road. Specifically, the contribution of this chapter are two-fold.

- Firstly, we propose an efficient IP-preserving FLIP protocol aiming at vehicle chatting application in VANETs, and formalize its security model as well. Then, we apply the provable security technique to validate its security within the defined model.
- Secondly, we develop a custom simulator built in Java to measure the relation between the IP-preserving level and the delay for finding the like-minded vehicle. The extensive simulation results show that, after setting a required IP-preserving level, a vehicle can find a like-minded vehicle within an expected time.

The remainder of this chapter is organized as follows. In Section 6.2, we introduce the system model and design goal, as well as the security model of FLIP. Then, we present our IP-preserving FLIP protocol in Section 6.3, followed by its security analysis and performance evaluation in Section 6.4 and Section 6.5, respectively. We also generalize the FLIP protocol with a lightweight Privacy-preserving Scalar Product Computation (PPSPC) in Section 6.6 and discuss the related work in Section 6.7. Finally, we draw our conclusions in Section 6.8.

6.2 System Model and Design Goal

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

6.2.1 System Model

We consider a VANET in a city environment, which consists of a large number of vehicles $\mathcal{V} = \{V_1, V_2, \cdots\}$ and a single offline trusted authority (TA), as shown in Fig. 6.1. Since we confine our problem to the scenario where vehicles find like-minded vehicles with common interest on the road without the assistance of RSUs, we do not include RSUs in our current model, although they are still deployed to support V-2-I communication.

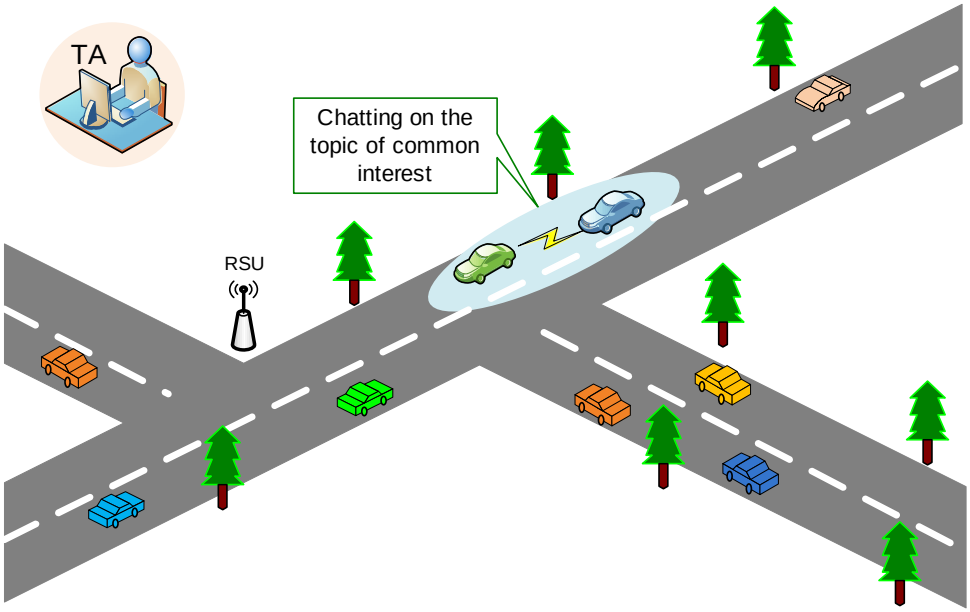


Figure 6.1: System model under consideration

- Trust Authority (TA): TA is a trustable and powerful entity. The responsibility of TA is in charge of management of the whole network, for example, initializing the system, registering the vehicles in the system by assigning a finite set of pseudo-IDs and the corresponding key materials to each vehicle. Note that TA is an offline entity, which is not directly involved in the V-2-V communications.

- Vehicles $\mathcal{V} = \{V_1, V_2, \cdots\}$: Each vehicle $V_i \in \mathcal{V}$ is equipped with the OBU device, which allows them to communicate with each other for sharing some information of common interest. Different from the mobile nodes in the general ad hoc network, the OBU device in VANET has no power-constrained issue and at the same time, is equipped with powerful computational and communication capabilities. According

to [67], the medium used for communications among the neighboring vehicles is 5.9 GHz Dedicated Short Range Communication (DSRC) identified as IEEE 802.11p, and the transmission range of each vehicle is 300 m. When two vehicles V_a, and $V_b \in \mathcal{V}$ are within their transmission range, they can chat on the topics of common interest on the road.

6.2.2 Design Goal

Security requirements and design goal

Security and privacy are always of vital importance to the flourish of vehicle chatting application in VANETs. Without the guarantee of vehicle's privacy including identity privacy, location privacy and interest privacy, vehicle chatting application cannot be widely accepted by the public. Therefore, it is essential to protect vehicle's privacy. Specifically, the following security requirements should be ensured in vehicle chatting application: i) vehicle's real identity should be protected; ii) vehicle's location privacy should be guaranteed; and iii) vehicle's interests should be protected against others who does not have the common interest.

In regard of the former two security requirements, each vehicle can use pseudo-ID to conceal the real identity, and periodically change multiple pseudo-IDs to achieve the location privacy [12, 14]. However, for the third security requirement, vehicles should use some IP-preserving protocols to find other vehicle who has the common interest on the road. Concretely, when a vehicle V_a wants to talk with another vehicle V_b nearby, if V_b has the common interest with V_a, V_a and V_b can establish a shared session key used for secure chatting on the topics of common interest. However, if V_b does not have the common interest with V_a, neither V_a nor V_b can know the counterpart's interest.

To satisfy the above security requirements, our design goal is to develop an efficient privacy-preserving finding like-minded vehicle protocol (FLIP) in VANET environments. With FLIP, vehicles who have the common interest can establish a shared session key without violating IP to others who have non-common interest. To subtly check the security of FLIP protocol in terms of IP-preserving, we should formally define its security model as follows.

Security model of FLIP

To model all possible leakages of IP in finding like-minded vehicle protocol on the road, we define the security model of FLIP by borrowing some ideas from security model of authen-

ticated key exchange (AKE) protocols [92] to describe some possible attacks. Specifically, in the model, the vehicles do not deviate from the FLIP protocol, while an adversary $\mathcal{A}$, whose attack capabilities are modelled by a set of pre-defined oracle queries, can passively monitor and/or actively control all the inter-vehicle communications. We assume that two vehicles V_a and V_b participate in FLIP for common interest $I_\alpha \in \mathcal{I} = \{I_1, I_2, \cdots, I_k\}$. Each of them has several *instances* called oracles involved in distinct executions of FLIP, where the common interest I_α varies in different executions. We denote an instance s of $V_i \in \{V_a, V_b\}$ by $\Pi^s_{V_i}$ for an integer $s \in \mathbb{N}$, and use the notation $\Pi^s_{V_a,V_b}$ to define the s-th instance V_a executing FLIP with V_b on the common interest I^s_α, where $\alpha \in \mathbb{N}$ and $1 \leq \alpha \leq k$.

Adversarial Model: We allow the adversary $\mathcal{A}$ to access to all transcripts in the FLIP. All oracles only communicate with each other via $\mathcal{A}$. The adversary $\mathcal{A}$ can replay, modify, delay, interleave or delete transcripts.

- $\mathsf{Execute}(\Pi^s_{V_a,V_b})$: This query models passive attacks, where $\mathcal{A}$ accesses an honest execution of FLIP between V_a and V_b by eavesdropping.
- $\mathsf{SendReq}(\Pi^s_{V_a}, *)$: This query models $\mathcal{A}$ can send a transcript m to the requestor-instance $\Pi^s_{V_a}$, and get back the answer of $\Pi^s_{V_a}$ by following FLIP. The adversary $\mathcal{A}$ can use this query to perform active attacks by modifying and inserting the transcript of the protocol to identify the IP of the requestor V_a. A query $\mathsf{SendReq}(\Pi^s_{V_a}, \mathsf{init})$ initializes the protocol, and thus the adversary $\mathcal{A}$ receives the transcripts sent out by V_a to V_b.
- $\mathsf{SendRes}(\Pi^s_{V_b}, *)$: This query models $\mathcal{A}$ can send a transcript m to the responsor-instance $\Pi^s_{V_b}$, and get back the answer of $\Pi^s_{V_b}$ by following FLIP. The adversary $\mathcal{A}$ can use this query to perform active attacks by modifying and inserting the transcript of the protocol to identify the IP of the responsor V_b.
- $\mathsf{Reveal}(\Pi^s_{V_a,V_b})$: This query models the known session key attack. The adversary $\mathcal{A}$ can get access to an old session key that has been previously established. Once $\Pi^s_{V_a,V_b}$ is valid and holds some session key, then $\Pi^s_{V_a,V_b}$ will send the session key and the common interest I^s_α to $\mathcal{A}$ when it receives the query.
- $\mathsf{Corrupt}(V_i)$: This query models exposure of the private key corresponding to pid_i held by $V_i \in \{V_a, V_b\}$ to the adversary $\mathcal{A}$. In reality, the scenarios that V_i may discard some outdated pseudo-ID and its corresponding key materials are modelled by this query.

- $\mathsf{Test}(\Pi^s_{V_a,V_b})$: This query is used to define the advantage of the adversary $\mathcal{A}$. When the adversary $\mathcal{A}$ queries on an instance $\Pi^s_{V_a,V_b}$ based on the common interest I^s_α, where $1 \leq \alpha \leq k$, $\mathcal{A}$ is given either the actual session key or a random value drawn from the session key space, according to a random bit $\beta \in \{0,1\}$, i.e., actual session key is given when $\beta = 0$ and a random value is drawn when $\beta = 1$. The Test query can be asked at most once by $\mathcal{A}$.

Freshness: The freshness is a useful notion, which identifies the session keys about which the adversary $\mathcal{A}$ ought not to know anything since $\mathcal{A}$ has not revealed any oracles that have accepted the session key and has not corrupted $V_i \in \{V_a, V_b\}$. An oracle $\Pi^s_{V_a,V_b}$ is said fresh if i) $\Pi^s_{V_a,V_b}$ has accepted a session key and $\Pi^s_{V_a,V_b}$ has not been asked for a Reveal query; ii) No $\mathsf{Corrupt}$ query has been asked before a query of the form $\mathsf{SendReq}(\Pi^s_{V_a}, *)$ or $\mathsf{SendRes}(\Pi^s_{V_b}, *)$.

Definition of Security: The security of FLIP is defined using the following game, played between $\mathcal{A}$ and a collection of $\Pi^s_{V_a,V_b}$ oracle for vehicles V_a, V_b and $s \in \mathbb{N}$, where V_a, V_b are first assigned pseudo-IDs and the corresponding key materials, respectively.

- In the game, $\mathcal{A}$ may ask some queries and get back the answers from the corresponding oracles.
- At certain point, $\mathcal{A}$ asks a Test query to a fresh oracle, and outputs its guess α' for α, where $1 \leq \alpha \leq k$, and β' for the bit β in the Test query.

The success of $\mathcal{A}$ in the game is quantified in terms of $\mathcal{A}$'s advantage in distinguishing whether $\mathcal{A}$ guesses the correct common interest I^s_α, and receives a real session key or not, i.e., its ability guessing α, β. We define A's advantages as

$$\mathbf{Adv}^\alpha_{\mathcal{P}}(\mathcal{A}) = k \cdot \Pr[\alpha = \alpha'] - 1, \mathbf{Adv}^\beta_{\mathcal{P}}(\mathcal{A}) = 2 \cdot \Pr[\beta = \beta'] - 1$$

We say that the FLIP is secure if both $\mathbf{Adv}^\alpha_{\mathcal{P}}(\mathcal{A})$ and $\mathbf{Adv}^\beta_{\mathcal{P}}(\mathcal{A})$ are negligible.

6.3 Our Proposed FLIP Protocol

In this section, we present our efficient privacy-preserving finding like-minded vehicle protocol (FLIP), which mainly consists of two parts: system initialization and privacy-preserving finding like-minded vehicle on the road.

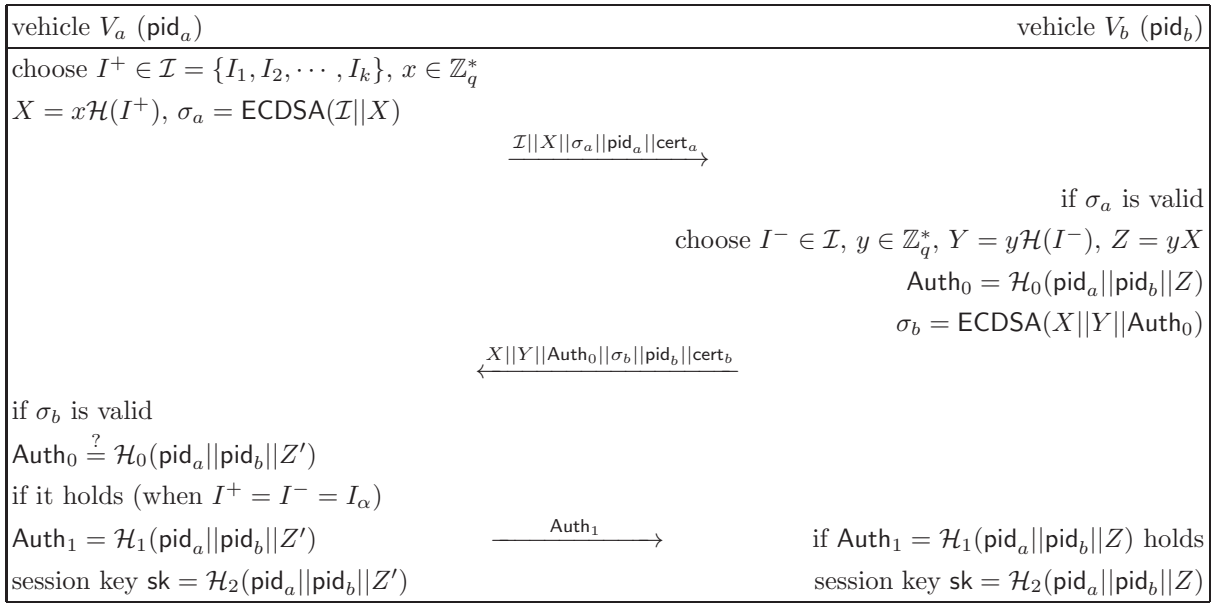


Figure 6.2: Proposed Privacy-preserving Finding Like-minded Vehicle Protocol

6.3.1 System Initialization

In the system initialization phase, the Trusted Authority (TA) first initializes the whole system by running the following steps. Given the security parameter l, TA generates an elliptic curve group $\mathbb{G} = \langle P \rangle$, where the generator P has a large prime order q with $|q| = l$. Then, TA chooses a random number $s \in \mathbb{Z}_q^*$ as the *master key* and compute the corresponding system public key $P_{pub} = sP$. In addition, TA also chooses four secure hash functions $\mathcal{H}, \mathcal{H}_0, \mathcal{H}_1$, and $\mathcal{H}_2$, where $\mathcal{H} : \{0,1\}^* \to \mathbb{G}$ and $\mathcal{H}_i : \{0,1\}^* \to \mathbb{Z}_q^*$, for $i = 0, 1, 2$. In the end, TA publishes the public system parameters *params* as $\{\mathbb{G}, P, q, P_{pub}, \mathcal{H}, \mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2\}$ and keeps the *master key* secretly.

When a vehicle $V_i \in \mathcal{V}$ itself to the system, TA first checks the vehicle V_i's validity. If V_i is valid, TA generates a family of pseudo-IDs and the corresponding key materials for V_i using Algorithm 6. In such a way, V_i can constantly change its pseudo-IDs to achieve identity privacy and location privacy on the road.

6.3.2 Privacy-preserving Finding Like-minded Vehicle

When a vehicle $V_a \in \mathcal{V}$ is on the road and wants to find a like-minded vehicle $V_b \in \mathcal{V}$ on the common interest I_α nearby, as shown in Fig. 6.2, they will run the following steps to

Algorithm 6 Vehicle Registration Algorithm

1: **procedure** VEHICLEREGISTRATION
   Input: a verified vehicle $V_i \in \mathcal{V}$
   Output: a family of pseudo-IDs and the corresponding key materials
2: choose a family of unlinkable pseudo-IDs $\mathsf{PID} = \{\mathsf{pid}_1, \mathsf{pid}_2, \cdots\}$
3: **for** each pseudo-ID $\mathsf{pid}_j \in \mathsf{PID}$ **do**
4: randomly choose a private key $x_j \in \mathbb{Z}_q^*$
5: compute the corresponding public key $Y_j = x_j P$
6: assert (pid_j, Y_j) with certificate cert_j signed by TA with s
7: **end for**
8: **return** all tuples $(\mathsf{pid}_j, x_j, Y_j, \mathsf{cert}_j)$ to V_i
9: **end procedure**

establish a shared session key sk regarding the common interest I_α.

Step 1. V_a first sets an interest set $\mathcal{I}$, which consists of k kinds of interests $\{I_1, I_2, \cdots, I_k\}$, where V_a's actual interest I^+ is involved. Then, V_a chooses a random number $x \in \mathbb{Z}_q^*$, computes $X = x\mathcal{H}(I^+)$, and uses the ECDSA algorithm to make a signature $\sigma_a = \mathsf{ECDSA}(\mathcal{I}||X)$ on $\mathcal{I}||X$ with regard to the pseudo-ID pid_a and the certificate cert_a. In the end, V_a broadcasts the request $\mathcal{I}||X||\sigma_a||\mathsf{pid}_a||\mathsf{cert}_a$ to the nearby vehicles.

Step 2. Upon receiving the request $\mathcal{I}||X||\sigma_a||\mathsf{pid}_a||\mathsf{cert}_a$, a nearby vehicle V_b first checks the validity of σ_a with $\mathsf{pid}_a||\mathsf{cert}_a$. If it is invalid, V_b neglects the request. Otherwise, V_b chooses his interest $I^- \in \mathcal{I}$ and a random number $y \in \mathbb{Z}_q^*$, computes $Y = y\mathcal{H}(I^-)$, $Z = yX$, and $\mathsf{Auth}_0 = \mathcal{H}_0(\mathsf{pid}_a||\mathsf{pid}_b||Z)$. In addition, V_b makes a signature $\sigma_b = \mathsf{ECDSA}(X||Y||\mathsf{Auth}_0)$ on $X||Y||\mathsf{Auth}_0$ with regard to the pseudo-ID pid_b and the certificate cert_b, and returns the response $X||Y||\mathsf{Auth}_0||\sigma_b||\mathsf{pid}_b||\mathsf{cert}_b$ to V_a. Note that, in the protocol, V_b is only allowed to make at most one response for the same request.

Step 3. After receiving the responsor V_b's response $X||Y||\mathsf{Auth}_0||\sigma_b||\mathsf{pid}_b||\mathsf{cert}_b$, the requestor V_a first checks the validity of σ_b with $\mathsf{pid}_b||\mathsf{cert}_b$. If it is invalid, V_a neglects the response. Otherwise, V_a computes $Z' = xY$, and checks whether $\mathsf{Auth}_0 \stackrel{?}{=} \mathcal{H}_0(\mathsf{pid}_a||\mathsf{pid}_b||Z')$. If it holds, V_a computes and sends $\mathsf{Auth}_1 = \mathcal{H}_1(\mathsf{pid}_a||\mathsf{pid}_b||Z')$ to V_b, and calculate the session key $\mathsf{sk} = \mathcal{H}_2(\mathsf{pid}_a||\mathsf{pid}_b||Z')$.

Step 4. When V_b receives $\mathsf{Auth}_1 = \mathcal{H}_1(\mathsf{pid}_a||\mathsf{pid}_b||Z')$, he checks whether $\mathsf{Auth}_1 \stackrel{?}{=} \mathcal{H}_1(\mathsf{pid}_a||\mathsf{pid}_b||Z)$. If it holds, V_b calculates the session key $\mathsf{sk} = \mathcal{H}_2(\mathsf{pid}_a||\mathsf{pid}_b||Z)$. If $I^+ = I^- = I_\alpha$ for some $1 \leq \alpha \leq k$, V_a and V_b have the shared session key sk, i.e., the vehicle V_a successfully finds an like-minded vehicle V_b on the road.

Correctness. If the interests I^+ and I^- are same, i.e., $I^+ = I^- = I_\alpha \in \mathcal{I}$, then

$\mathcal{H}(I^-) = \mathcal{H}(I^+)$,

$$Z' = xY = xy\mathcal{H}(I^-) = xy\mathcal{H}(I_\alpha) = yx\mathcal{H}(I^+) = yX = Z$$

and both the authenticators $\mathsf{Auth}_0 = \mathcal{H}_0(\mathsf{pid}_a||\mathsf{pid}_b||Z)$, $\mathsf{Auth}_1 = \mathcal{H}_1(\mathsf{pid}_a||\mathsf{pid}_b||Z')$ and the session key sk are valid. However, if $I^+ \neq I^-$, then $\mathcal{H}(I^-) \neq \mathcal{H}(I^+)$ and

$$Z' = xY = xy\mathcal{H}(I^-) \neq xy\mathcal{H}(I^+) = yX = Z$$

which indicates that Auth_0, and Auth_1 are not valid, and the shared session key sk cannot be established. Therefore, the correctness of the proposed FLIP protocol follows. Note that, the responsor V_b can only respond once for the same request. Otherwise, by successive responses to the same request, the requestor's IP can be guessed by non-like-minded vehicles. In reality, the requestor V_a usually can detect whether a nearby vehicle V_b has responded more than once based on V_b's relative location and other correlative information on the road. Thus, the successive-response attack can be prevented.

6.4 Security Analysis

In this section, we will demonstrate the IP can be protected against non-like-minded vehicles without collusion in the proposed FLIP protocol. Note that, since the ECDSA signature is unforgeable, all transcripts in FLIP are detectable if they are altered by the adversary. Therefore, we should only consider an adversary cannot break the proposed FLIP protocol without altering the transcripts.

Theorem 3 *Let $\mathcal{A}$ be an adversary against the proposed FLIP protocol in the random oracle model, where the hash functions $\mathcal{H}, \mathcal{H}_0, \mathcal{H}_1$, and $\mathcal{H}_2$ behave as random oracles. Assume that $\mathcal{A}$ has the advantage $\mathbf{Adv}^{\alpha}_{\mathcal{P}}(\mathcal{A}) = \varepsilon$ to guess the correct interest I_α, and the advantage $\mathbf{Adv}^{\beta}_{\mathcal{P}}(\mathcal{A}) = \epsilon$ breaks the proposed FLIP protocol without altering the transcripts, within the running time τ, after several oracles defined in the adversarial model. Then, there exist $\epsilon' \in [0, 1]$ and $\tau' \in \mathbb{N}$ as follows*

$$\epsilon' = \mathbf{Succ}^{\mathrm{CDH}}_{\mathcal{A}} \geq \frac{\epsilon}{q_s q_{H_2}}, \quad \tau' \leq \tau + \Theta(.)$$

such that the CDH problem can be solved with probability ϵ' and within time τ', where $\Theta(.)$ is the time complexity for the simulation, q_{H_2} is the total number of $\mathcal{H}_2$ oracle queries, and q_s is the total number of session instances $\Pi^1_{V_a,V_b}, \Pi^2_{V_a,V_b}, \cdots, \Pi^{q_s}_{V_a,V_b}$.

▷ **sim**-$\mathcal{H}$
On input of an interest $I_i \in \mathcal{I} = \{I_1, \cdots, I_k\}$
choose a fresh random number $r_i \xleftarrow{R} \mathbb{Z}_q^*$
compute $H_i = r_i P$, set $\mathcal{H}(I_i) = H_i$
add (I_i, r_i, H_i) to $\mathcal{H}$-list
return H_i

▷ **sim**-SendRes($\Pi^s_{V_b}, \mathcal{I}||X||\sigma_a||\mathsf{pid}_a||\mathsf{cert}_a$)
if $s = \gamma$ **then**
  choose the same interest I_α identified with s
  obtain the tuple $(I_\alpha, r_\alpha, H_\alpha)$ in $\mathcal{H}$-list
  compute $Y = r_\alpha y P$, set $Z = \sqcup$
  choose a fresh random number $u_{\alpha 0} \xleftarrow{R} \mathbb{Z}_q^*$
  set $\mathsf{Auth}_0 = \mathcal{H}_0(\mathsf{pid}_a||\mathsf{pid}_b||Z) = u_{\alpha 0}$
  add $(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{\alpha 0})$ to $\mathcal{H}_0$-list
  compute $\sigma_b = \mathsf{ECDSA}(X||Y||\mathsf{Auth}_0)$
  return $X||Y||\mathsf{Auth}_0||\sigma_b||\mathsf{pid}_b||\mathsf{cert}_b$
else if $s \neq \gamma$ **then**
  choose the same interest I_i identified with s
  obtain the tuple (I_i, r_i, H_i) in $\mathcal{H}$-list
  chooses $y_i \in \mathbb{Z}_q^*$, set $Y = y_i H_i$, $Z = x_i y_i H_i$
  choose a fresh random number $u_{i0} \xleftarrow{R} \mathbb{Z}_q^*$
  set $\mathsf{Auth}_0 = \mathcal{H}_0(\mathsf{pid}_a||\mathsf{pid}_b||Z) = u_{i0}$
  add $(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{i0})$ to $\mathcal{H}_0$-list
  compute $\sigma_b = \mathsf{ECDSA}(X||Y||\mathsf{Auth}_0)$
  return $X||Y||\mathsf{Auth}_0||\sigma_b||\mathsf{pid}_b||\mathsf{cert}_b$

▷ **sim**-$\mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2$
On input of $\mathsf{pid}_a||\mathsf{pid}_b||Z_i$ on $\mathcal{H}_j$, $j \in \{0, 1, 2\}$
choose a fresh random number $u_{ij} \xleftarrow{R} \mathbb{Z}_q^*$
set $\mathcal{H}_j(\mathsf{pid}_a||\mathsf{pid}_b||Z_i) = u_{ij}$
add $(\mathsf{pid}_a||\mathsf{pid}_b||Z_i, u_{ij})$ to $\mathcal{H}_j$-list
return u_{ij}

▷ **sim**-Corrupt(V_i) with $V_i \in \{V_a, V_b\}$
return private key $\widetilde{x}_i$ of V_i with respect to pid_i

▷ **sim**-Reveal($\Pi^s_{V_a,V_b}$) with $s \neq \gamma$
obtain the tuple $(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{i2})$ in $\mathcal{H}_2$-list
  and the interest I_i identified with s
return $\mathsf{sk} = u_{i2}$ and I_i

▷ **sim**-SendReq($\Pi^s_{V_a}, X||Y||\mathsf{Auth}_0||\sigma_b||\mathsf{pid}_b$
  $||\mathsf{cert}_b$)
if $s = \gamma$ **then**
  obtain the tuple$(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{\alpha 0})$ in $\mathcal{H}_0$-list
    on condition that $\mathsf{Auth}_0 = u_{\alpha 0}$
  choose a fresh random number $u_{\alpha 1} \xleftarrow{R} \mathbb{Z}_q^*$
  set $\mathsf{Auth}_1 = \mathcal{H}_1(\mathsf{pid}_a||\mathsf{pid}_b||Z) = u_{\alpha 1}$
  add $(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{\alpha 1})$ to $\mathcal{H}_1$-list
  return Auth_1
else if $s \neq \gamma$ **then**
  obtain the tuple$(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{i0})$ in $\mathcal{H}_0$-list
    on condition that $\mathsf{Auth}_0 = u_{i0}$
  choose random numbers $u_{\alpha 1}, u_{\alpha 2} \xleftarrow{R} \mathbb{Z}_q^*$
  set $\mathsf{Auth}_1 = \mathcal{H}_1(\mathsf{pid}_a||\mathsf{pid}_b||Z) = u_{i1}$
  add $(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{i1})$ to $\mathcal{H}_1$-list
  set session key $\mathsf{sk} = \mathcal{H}_2(\mathsf{pid}_a||\mathsf{pid}_b||Z) = u_{i2}$
  add $(\mathsf{pid}_a||\mathsf{pid}_b||Z, u_{i2})$ to $\mathcal{H}_2$-list
  return Auth_1

▷ **sim**-SendReq($\Pi^s_{V_a}, \mathsf{init}$)
if $s = \gamma$ **then**
  randomly choose an interest $I_\alpha \in \mathcal{I}$
  obtain the tuple $(I_\alpha, r_\alpha, H_\alpha)$ in $\mathcal{H}$-list
  compute $X = r_\alpha x P$, $\sigma_a = \mathsf{ECDSA}(\mathcal{I}||X)$
  return $\mathcal{I}||X||\sigma_a||\mathsf{pid}_a||\mathsf{cert}_a$
else if $s \neq \gamma$ **then**
  randomly choose $I_i \in \mathcal{I}$, and $x_i \in \mathbb{Z}_q^*$
  obtain the tuple (I_i, r_i, H_i) in $\mathcal{H}$-list
  compute $X = x_i H_i$, $\sigma_a = \mathsf{ECDSA}(\mathcal{I}||X)$
  return $\mathcal{I}||X||\sigma_a||\mathsf{pid}_a||\mathsf{cert}_a$

▷ **sim**-Execute($\Pi^s_{V_a,V_b}$)
successively simulate SendReq($\Pi^s_{V_a}, \mathsf{init}$),
 SendRes($\Pi^s_{V_b}, \mathcal{I}||X||\sigma_a||\mathsf{pid}_a||\mathsf{cert}_a$), and
 SendReq($\Pi^s_{V_a}, X||Y||\mathsf{Auth}_0||\sigma_b||\mathsf{pid}_b||\mathsf{cert}_b$)

▷ **sim**-Test($\Pi^s_{V_a,V_b}$) with $s = \gamma$
randomly flip a coin $\beta \in \{0, 1\}$
choose a random number $u_{\alpha 2} \xleftarrow{R} \mathbb{Z}_q^*$
return $u_{\alpha 2}$

□

Figure 6.3: Simulations of oracles in FLIP

Proof. Since the adversary $\mathcal{A}$ can, with non-negligible advantage $\mathbf{Adv}_{\mathcal{P}}^{\beta}(\mathcal{A})$, break the proposed FLIP protocol, we can use $\mathcal{A}$'s attack capabilities to construct another algorithm $\mathcal{B}$ to solve the CDH problem. In specific, $\mathcal{B}$ is given a random instance of the CDH problem (P, xP, yP), where $x, y \in \mathbb{Z}_q^*$. Then, $\mathcal{B}$ runs $\mathcal{A}$ as a subroutine and simulates the attack environment required by $\mathcal{A}$.

At first, for each vehicle $V_i \in \{V_a, V_b\}$ involved in the FLIP protocol, $\mathcal{B}$ sets V_i's pseudo-ID pid_i, generates valid key materials by choosing a random number $\widetilde{x}_i \in \mathbb{Z}_q^*$ as the private key, computes the corresponding public key $\widetilde{Y}_i = \widetilde{x}_i P$, as well as the certificate cert_i with the resort of TA. Then, $\mathcal{B}$ interacts with $\mathcal{A}$ and simulates all the instances with queries of oracles $\mathsf{SendReq}$, $\mathsf{SendRes}$, $\mathsf{Execute}$, Reveal, $\mathsf{Corrupt}$, and Test. In order to make use of $\mathcal{A}$'s attack capability, $\mathcal{B}$ first guesses γ such that $\mathcal{A}$ asks the Test query in the γ-th session. Because there are total q_s session instances, the probability for successful guessing γ is $1/q_s$. Besides the above oracles, $\mathcal{B}$ should also simulates the random oracles $\mathcal{H}, \mathcal{H}_0, \mathcal{H}_1$, and $\mathcal{H}_2$ by maintaining the lists $\mathcal{H}$-list, $\mathcal{H}_0$-list, $\mathcal{H}_1$-list and $\mathcal{H}_2$-list to deal with the identical queries as shown in Fig. 6.3.

After receiving $u_{\alpha 2}$ from $\mathsf{Test}(\Pi^s_{V_a,V_b})$, the adversary $\mathcal{A}$ guesses $\alpha' \in \{1, 2, \cdots, k\}$ for α and $\beta' \in \{0, 1\}$ for β, and returns (α', β') to $\mathcal{B}$. Then, we analyze $\mathcal{A}$'s successful guess probability on α' and β'.

First, we consider the transcripts $X = x\mathcal{H}(I_\alpha)$ and $Y = y\mathcal{H}(I_\alpha)$ with unknown $x, y \in \mathbb{Z}_q^*$. Because $\mathbb{G}$ is a cyclic group, we can see there always exist other $x_i, y_i \in \mathbb{Z}_q^*$ such that $X = x_i\mathcal{H}(I_i)$ and $Y = y_i\mathcal{H}(I_i)$. Therefore, the transcripts (X, Y) can be linked to each interest $I_i \in \{I_1, I_2, \cdots, I_k\}$ equally. Therefore, we can know $\Pr[\alpha' = \alpha] = \frac{1}{k}$ and

$$\mathbf{Adv}_{\mathcal{P}}^{\alpha}(\mathcal{A}) = k \cdot \Pr[\alpha = \alpha'] - 1 = 0, \quad i.e., \varepsilon = 0$$

Second, we analyze the advantage probability $\mathbf{Adv}_{\mathcal{P}}^{\beta}(\mathcal{A})$ on guessing the correct β, where

$$\Pr[\beta = \beta'] = \frac{1}{2} + \frac{\mathbf{Adv}_{\mathcal{P}}^{\beta}(\mathcal{A})}{2}$$

Let E denote the event that $\mathsf{pid}_a||\mathsf{pid}_b||Z$ has been queried by $\mathcal{A}$ on $\mathcal{H}_2$ oracle, where $Z = r_\alpha xyP$. If the event E does not occur, $\mathcal{B}$ has no idea on the session key sk, then we have $\Pr[\beta' = \beta|\neg\mathsf{E}] = \frac{1}{2}$, and

$$\begin{aligned}
\Pr[\beta = \beta'] &= \Pr[\beta = \beta'|\mathsf{E}] \cdot \Pr[\mathsf{E}] + \Pr[\beta = \beta'|\neg\mathsf{E}] \cdot \Pr[\neg\mathsf{E}] \\
&= \Pr[\beta = \beta'|\mathsf{E}] \cdot \Pr[\mathsf{E}] + \frac{1}{2} \cdot (1 - \Pr[\mathsf{E}]) \\
&\leq \Pr[\mathsf{E}] + \frac{1}{2} \cdot (1 - \Pr[\mathsf{E}]) = \frac{1}{2} + \frac{\Pr[\mathsf{E}]}{2}
\end{aligned}$$

Therefore, based on the above relations, we have

$$\Pr[\mathsf{E}] \geq \mathbf{Adv}_{\mathcal{P}}^{\beta}(\mathcal{A})$$

Because $\mathcal{H}_2$-list contains q_{H_2} entries, we can pick up the correct $\mathsf{pid}_a||\mathsf{pid}_b||Z$, where $Z = r_\alpha xyP$, with the success probability $1/q_{H_2}$. Then, by computing $Z/r_\alpha = xyP$, where r_α is included in the entry $(I_\alpha, r_\alpha, H_\alpha)$ in $\mathcal{H}$-list, we can get the CDH challenge xyP. Combining the probability $1/q_s$ for guessing the correct γ, we have

$$\epsilon' = \mathbf{Succ}_{\mathcal{A}}^{\text{CDH}} \geq \frac{\mathbf{Adv}_{\mathcal{P}}^{\beta}(\mathcal{A})}{q_s q_{H_2}} = \frac{\epsilon}{q_s q_{H_2}}$$

In addition, we can obtain the claimed bound for $\tau' \leq \tau + \Theta(.)$ in the above simulation. In summary, the IP can be protected in FLIP. Thus, the proof is completed.

6.5 Performance Evaluation

In the proposed FLIP protocol, to prevent successive-guessing attack from non-like-minded vehicle, the responsor V_b is only allowed to respond once for the same request. Therefore, the larger the Interest Set $\mathcal{I} = \{I_1, I_2, \cdots, I_k\}$ that the requestor V_a chooses, the harder the actual interest $I_\alpha \in \mathcal{I}$ can be guessed by non-like-minded vehicle, and thus the IP can be protected. However, when the set $\mathcal{I}$ becomes large, multiple interests $\mathcal{I}' = \{I_\alpha, I_\beta, I_\gamma, \cdots\}$ of the like-minded vehicle V_b could belong to $\mathcal{I}$. Then, it is hard for V_b to choose the correct $I_\alpha \in \mathcal{I}'$, which thus causes the long delay for finding the like-minded vehicle. Therefore, in this section, we use a custom simulator built in Java to study how the interest set $\mathcal{I}$ affects the delay for finding the like-minded vehicle on the road. In specific, the performance metric used in the evaluation is the average delay for finding the like-minded vehicle, denoted by FD, which is defined as the average time between when the requestor V_a sends a request and when V_b successfully finds a like-minded vehicle V_b on the road.

6.5.1 Simulation Settings

We consider a large number of vehicles $\mathcal{V} = \{V_1, V_2, \cdots\}$ are moving on a multi-lane same-direction road with velocity varying from 40 km/h to 80 km/h. Consider other vehicles passing-by a vehicle $V_a \in \mathcal{V}$ follows a Poisson process, and the inter-passing-by time t_a has an exponential distribution with the mean $1/\lambda$. In the simulation, the vehicle V_a will

broadcast the request with Interest set $\mathcal{I}$ of different size $|\mathcal{I}|$ varying from 1 to 10, to find the like-minded vehicle.

The detailed parameter settings in the simulations are summarized in Table 6.1. We perform the simulations for the specified interest set size $|\mathcal{I}|$ varying from 1 to 10 with increment of 1. For each case, we run the simulation 10,000 times, and the average FD is reported.

Table 6.1: Simulation Settings in FLIP

Parameter	Setting
Simulation area, duration	a multi-lane same-direction road, 1 hour
Vehicle velocity, transmission	40 km/h - 80 km/h, 300 m
Mean passing-by rate λ	$[20/h, 40/h, 60/h, 80/h, 100/h, 120/h]$
Interest set size $\|\mathcal{I}\|$	$[1, 2, \cdots, 10]$

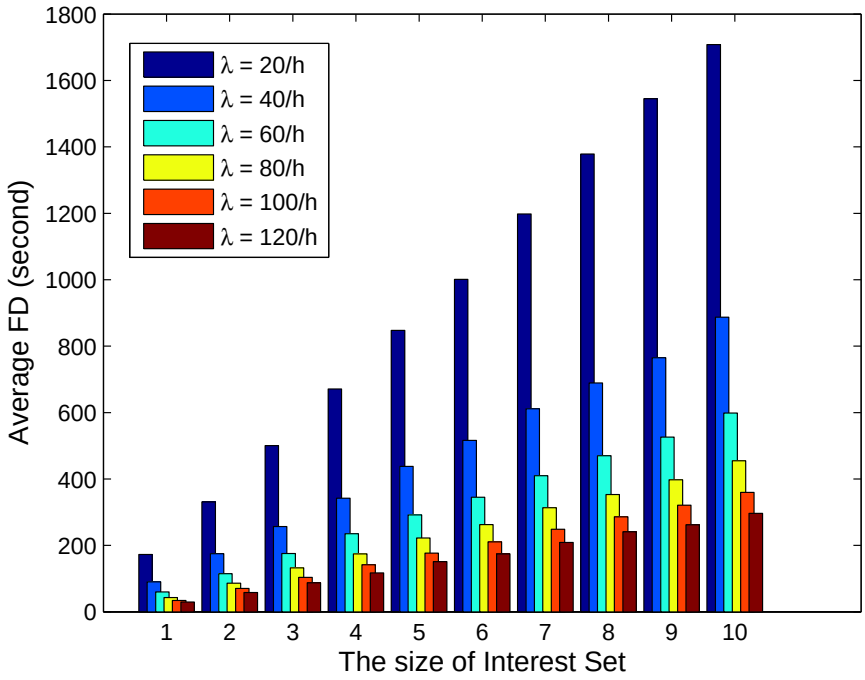


Figure 6.4: The average FD in different interest set size $|\mathcal{I}|$ within 1 hour

6.5.2 Simulation Results

Fig. 6.4 shows the average FD under the different $|\mathcal{I}|$ and λ within 1 hour. From the figure, we can see, the larger the $|\mathcal{I}|$, the longer the average FD; but at the same time, the average FD can be reduced with the increase of λ. Therefore, by setting a proper size of $|\mathcal{I}|$ on considering of λ, a vehicle V_a can find a like-minded vehicle within an expected time on the road while keeping his IP from non-like-minded vehicles.

6.6 Generalization of FLIP

The goal of the proposed FLIP protocol is to exactly find a like-minded vehicle. However, this kind of exact finding sometimes is not efficient, especially when the size of $|\mathcal{I}|$ is large. To address this issue, we propose a lightweight Privacy-preserving Scalar Product Computation (PPSPC) to generalize the FLIP.

Given an interest set $\mathcal{I} = \{I_1, I_2, \cdots, I_n\}$, each vehicle's interest preferences can be represented by a binary vector $\vec{\mathbf{a}} = (a_1, a_2, \cdots, a_n)$, where $a_i \in \vec{\mathbf{a}}$ indicates one interest, i.e., $a_i = 1$ if the vehicle has the corresponding interest, and $a_i = 0$ otherwise. Now, assume vehicles V_a and V_b have their interest vectors $\vec{\mathbf{a}} = (a_1, a_2, \cdots, a_n)$ and $\vec{\mathbf{b}} = (b_1, b_2, \cdots, b_n)$, respectively. If the number of their common interests reaches a threshold τ, which can be derived from $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$, they can launch vehicle chatting application on the road. In order to compute $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$ in a privacy-preserving way, V_a and V_b can invoke our newly designed lightweight PPSPC protocol in Algorithm 7. Since the PPSPC protocol ensures neither V_a nor V_b will disclose their interest preferences to each other during the computation of $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$, it can efficiently find like-mined vehicle in a privacy-preserving way. For example, if the returned value $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} \geq th$, V_a nor V_b can launch a chatting session. However, if the returned value $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} < th$, the common interests of V_a nor V_b is not enough to launch a session. Note that the threshold th is not fixed, it will vary with the road condition.

Correctness of PPSPC Protocol. The correctness of our proposed PPSPC protocol can be clearly illustrated by the following typical example. Assume two binary vectors are $\vec{\mathbf{a}} = (a_1, a_2, a_3, a_4, a_5) = (1, 1, 0, 0, 1)$ and $\vec{\mathbf{b}} = (b_1, b_2, b_3, b_4, b_5) = (1, 0, 1, 0, 1)$. After Step-1 is performed, we have $C_1 = \alpha + c_1 + r_1 \cdot \beta$, $C_2 = \alpha + c_2 + r_2 \cdot \beta$, $C_3 = c_3 + r_3 \cdot \beta$, $C_4 = c_4 + r_4 \cdot \beta$, and $C_5 = \alpha + c_5 + r_5 \cdot \beta$.

After Step-2 is executed, we have $D_1 = \alpha^2 + c_1 \cdot \alpha + r_1 \cdot \alpha \cdot \beta$, $D_2 = \alpha + c_2 + r_2 \cdot \beta$, $D_3 = c_3 \cdot \alpha + r_3 \cdot \alpha \cdot \beta$, $D_4 = c_4 + r_4 \cdot \beta$, $D_5 = \alpha^2 + c_5 \cdot \alpha + r_5 \cdot \alpha \cdot \beta$, and $D = \sum_{i=1}^{5} D_i$.

Based on the returned D and the secret $K = \sum_{i=1}^{5} k_i$, the value of E can be calculated in Step-3 as

$$\begin{aligned} & E = D + K = \textstyle\sum_{i=1}^{5}(D_i + k_i) \\ = \; & [\alpha^2 + c_1 \cdot (\alpha - 1) + r_1 \cdot \alpha \cdot \beta + c_1 + k_1] + (\alpha + \\ & r_2 \cdot \beta + c_2 + k_2) + [c_3 \cdot (\alpha - 1) + r_3 \cdot \alpha \cdot \beta + c_3 + \\ & k_3] + (r_4 \cdot \beta + c_4 + k_4) + [\alpha^2 + c_5 \cdot (\alpha - 1) + \\ & r_5 \cdot \alpha \cdot \beta + c_5 + k_5] \bmod \beta \\ = \; & [\alpha^2 + c_1 \cdot (\alpha - 1) + r_1 \cdot (\alpha + 1) \cdot \beta] + (r_2 \cdot 2 \cdot \beta \\ & + \alpha) + [c_3 \cdot (\alpha - 1) + r_3 \cdot (\alpha + 1) \cdot \beta] + r_4 \cdot 2 \cdot \beta \\ & + [\alpha^2 + c_5 \cdot (\alpha - 1) + r_5 \cdot (\alpha + 1) \cdot \beta] \bmod \beta \\ = \; & 2 \cdot \alpha^2 + \alpha + (c_1 + c_3 + c_5) \cdot (\alpha - 1) \bmod \beta \end{aligned} \tag{6.1}$$

Since $\alpha - n = \alpha - 5 > \sum_{i=1}^{n} c_i = \sum_{i=1}^{5} c_i$, $\beta > (n+1) \cdot \alpha^2 = 6 \cdot \alpha^2$ when $n = 5$, the value

$$\begin{aligned} & 2 \cdot \alpha^2 + \alpha + (c_1 + c_3 + c_5) \cdot (\alpha - 1) \\ < \; & 2 \cdot \alpha^2 + \alpha + \textstyle\sum_{i=1}^{5} c_i \cdot \alpha < 2 \cdot \alpha^2 + \alpha(1 + \alpha - 5) \\ < \; & 2 \cdot \alpha^2 + \alpha^2 = 3 \cdot \alpha^2 < \beta \end{aligned} \tag{6.2}$$

Therefore, we can remove " mod β" from Eq.(6.1) and have

$$\begin{aligned} E = & 2 \cdot \alpha^2 + \alpha + (c_1 + c_3 + c_5) \cdot (\alpha - 1) \bmod \beta \\ = & 2 \cdot \alpha^2 + \alpha + (c_1 + c_3 + c_5) \cdot (\alpha - 1) \end{aligned} \tag{6.3}$$

Again, since $\alpha + (c_1 + c_3 + c_5) \cdot (\alpha - 1) < \alpha^2$, we have

$$\frac{E - (E \bmod \alpha^2)}{\alpha^2} = \frac{2 \cdot \alpha^2}{\alpha^2} = 2 \tag{6.4}$$

According to the line-19 in Algorithm 7, only when both a_i and b_i are 1, an α^2 can be produced. Then, the coefficient of α^2 is just the required scalar product $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$. As a result, the correctness of PPSPC protocol is verified.

Algorithm 7 Privacy-preserving Scalar Product Computation

procedure PPSPC PROTOCOL
  Input: V_a's binary vector $\vec{\mathbf{a}} = (a_1, a_2, \cdots, a_n)$ and V_b's binary vector $\vec{\mathbf{b}} = (b_1, b_2, \cdots, b_n)$, where $n \leq 2^6$
  Output: The scalar product $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} = \sum_{i=0}^{n} a_i \cdot b_i$

  Step-1: V_a first does the following operations:
  choose two large primes α, β, where α is of the length $|\alpha| = 256$ bits and $\beta > (n+1) \cdot \alpha^2$, e.g., the length $|\beta| > 518$ bits if $n = 2^6$
  set $K = 0$ and choose n positive random numbers $(c_1, c_2, c_3, \cdots, c_n)$ such that $\sum_{i=1}^{n} c_i < \alpha - n$
  for each element $a_i \in \vec{\mathbf{a}}$ **do**
    choose a random number r_i, compute $r_i \cdot \beta$ such that $|r_i \cdot \beta| \approx 1024$ bits, and calculate $k_i = r_i \cdot \beta - c_i$
    if $a_i = 1$ **then**
      $C_i = \alpha + c_i + r_i \cdot \beta, \quad K = K + k_i$
    else if $a_i = 0$ **then**
      $C_i = c_i + r_i \cdot \beta, \quad K = K + k_i$
    end if
  end for
  keep (β, K) secret, and send $(\alpha, C_1, C_2, C_3, \cdots, C_n)$ to V_b

---

  Step-2: V_b then executes the following operations:
  for each element $b_i \in \vec{\mathbf{b}}$ **do**
    if $b_i = 1$ **then**
      $D_i = \alpha \cdot C_i = \begin{cases} \alpha^2 + c_i \cdot \alpha + r_i \cdot \alpha \cdot \beta, & \text{if } a_i = 1; \\ c_i \cdot \alpha + r_i \cdot \alpha \cdot \beta, & \text{if } a_i = 0. \end{cases}$
    else if $b_i = 0$ **then**
      $D_i = C_i = \begin{cases} \alpha + c_i + r_i \cdot \beta, & \text{if } a_i = 1; \\ c_i + r_i \cdot \beta, & \text{if } a_i = 0. \end{cases}$
    end if
  end for
  compute $D = \sum_{i=1}^{n} D_i$ and return D back to V_a

  Step-3: V_a continues to do the following operations:
  compute $E = D + K \bmod \beta$
  return $\frac{E - (E \bmod \alpha^2)}{\alpha^2}$ as the scalar product $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} = \sum_{i=0}^{n} a_i \cdot b_i$
end procedure

Extension of PPSPC protocol. Although Algorithm 7 deals with the PPSPC for binary vectors, it can be easily extended for the general vector's PPSPC. For example, to calculate the PPSPC of the general vectors $\vec{\mathbf{a}} = (a_1, a_2, \cdots, a_n)$, $\vec{\mathbf{b}} = (b_1, b_2, \cdots, b_n)$, where any $a_i, b_i \in \mathbb{Z}_m$ with $2 < m < 2^8$, we only make the following modifications in Algorithm 7, and its correctness can be easily verified as well.

5: choose two large primes α, β, where α is of the length $|\alpha| = 256$ bits and $\beta > (n \cdot m^2 + 1) \cdot \alpha^2$

6: set $K = 0$ and choose n positive random numbers $(c_1, c_2, c_3, \cdots, c_n)$ such that $m \cdot \sum_{i=1}^{n} c_i < \alpha - m \cdot n$

9: **if** $a_i \neq 0$ **then**

10: $C_i = a_i \cdot \alpha + c_i + r_i \cdot \beta, \quad K = K + k_i$

18: **if** $b_i \neq 0$ **then**

19: $D_i = b_i \cdot \alpha \cdot C_i$

Comparison between the proposed PPSPC and Paillier Cryptosystem based PPSPC protocol. The currently popular Paillier Cryptosystem (PC)-based PPSPC is described as follows. Given the Paillier cryptosystem $\mathcal{E}(x) = g^x r^N \bmod N^2$ [93], where $N = pq$ and the base g are public, V_a keeps (p, q) secretly and performs the following steps with V_b: *i*) for each element $a_i \in \vec{\mathbf{a}} = (a_1, a_2, \cdots, a_n)$, V_a first uses a random number r_i to encrypt a_i as $\mathcal{E}(a_i) = g^{a_i} r_i^N \bmod N^2$. Then, V_a sends $\mathcal{E}(\vec{\mathbf{a}}) = (\mathcal{E}(a_1), \mathcal{E}(a_2), \cdots, \mathcal{E}(a_n))$ to V_b; and *ii*) after receiving $\mathcal{E}(\vec{\mathbf{a}}) = (\mathcal{E}(a_1), \mathcal{E}(a_2), \cdots, \mathcal{E}(a_n))$, V_b uses his vector $\vec{\mathbf{b}} = (b_1, b_2, \cdots, b_n)$ to compute $\mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})$ as

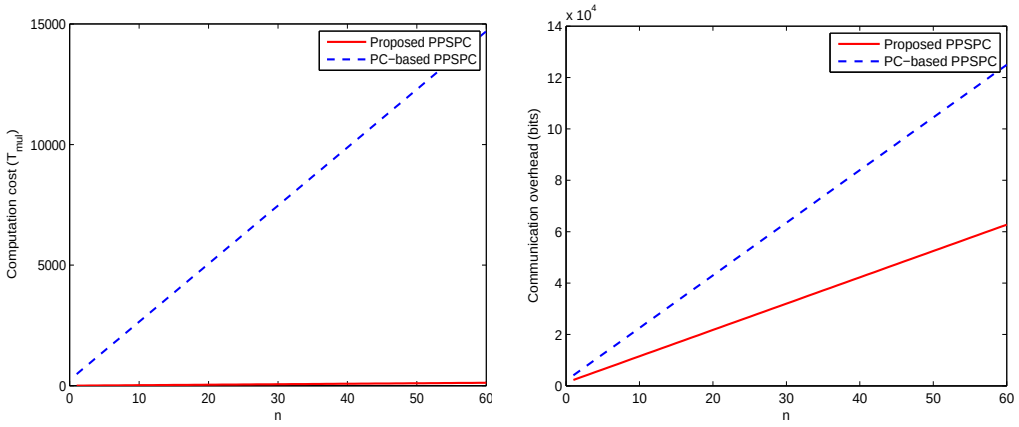
$$\begin{aligned} & \textstyle\prod_{i=1}^{n} \mathcal{E}(a_i)^{b_i} \equiv \prod_{i=1}^{n} \left(g^{a_i} r_i^N\right)^{b_i} \equiv \prod_{i=1}^{n} g^{a_i b_i} \left(r_i^{\,b_i}\right)^N \\ \equiv \;& g^{\sum_{i=1}^{n} a_i \cdot b_i} \cdot \left(\prod_{i=1}^{n} \left(r_i^{\,b_i}\right)\right)^N \equiv \mathcal{E}\left(\textstyle\sum_{i=1}^{n} a_i \cdot b_i\right) \\ \equiv \;& \mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}) \end{aligned}$$

and returns $\mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})$ back to V_a; *iii*) upon receiving $\mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})$, V_a uses the secret (p, q) to recover $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$ from $\mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})$.

Computational and communication costs: For binary vectors $(\vec{\mathbf{a}}, \vec{\mathbf{b}})$, V_a should take at least n exponentiations to compute $\mathcal{E}(\vec{\mathbf{a}})$. Then, V_b takes around $(n-1)$ multiplications to calculate $\mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})$. Finally, V_a takes one more exponentiation to recover $\vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$. Therefore, the computational cost is around $(n+1) \cdot exp + (n-1) \cdot mul$. Note that, if $(\vec{\mathbf{a}}, \vec{\mathbf{b}})$ are general vectors, the computational cost should be $(3n+1) \cdot exp + (n-1) \cdot mul$. The security of the Paillier cryptosystem relies on the unknown factorization of modulus $N = pq$. When

$N = pq$ is set as 1024, each $\mathcal{E}(a_i)$ and $\mathcal{E}(\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})$ will be expanded to 2048 bits, and then the communication cost will be $(n+1) \cdot 2048$ bits.

Different from the PC-based PPSPC, the proposed lightweight PPSPC does not use any "homomorphic encryption", but is very efficient in terms of computational and communication costs, i.e., the computational cost only takes $2n$ multiplications (mul), and the communication cost is only $(n+1) \cdot 1024 + 256$ bits. Let T_{mul}, T_{exp} denote the time needed to execute a modulus multiplication and a modulus exponentiation, respectively. When we roughly estimate $T_{exp} \approx 240 T_{mul}$ [94], we use Fig. 6.5 to compare the computation and communication costs of the proposed PPSPC protocol and the PC-based PPSPC protocol. From the figure, we can obviously observe that our proposed PPSPC protocol is much efficient, especially in computation costs. To the best of our knowledge, our proposed PPSPC is the most efficient privacy-preserving scalar product computation protocol till now.



(a) Computation comparison (b) Communication comparison

Figure 6.5: Comparisons between the proposed PPSPC and the PC-based PPSPC

6.7 Related Work

Recently, several research works have been reported [95, 96], which are closely related to the proposed FLIP protocol, but focus on other scenarios different from VANETs. Atallah and Du [95] present secure multi-party computation problems and several privacy-preserving applications including privacy-preserving database query, intrusion detection, data mining, and geometric computation. Although they take some initial attempts to tackle these problems, their solutions are less than satisfactory because a semi-trusted

third party is required to be involved in the privacy-preserving computations. Based on the homomorphic encryption, Zhong *et al.* [96] propose three protocols called Louis, Lester and Pierre to resolve the nearby friend problem, where a mobile user can determine whether or not one of his friends is in a nearby location in a privacy-preserving way. However, Louis still requires an online semi-trusted third party, and both Louis and Lester are found insecure [97]. Recently, Chatterjee *et al.* [97] propose a new efficient protocol for the nearby friend problem without resorting to a semi-honest third party. However, due to lack of authentication, their protocol could suffer from replay attack and man-in-the-middle attack, and thus cannot be directly applied in VANET scenarios. Different from the above works, our proposed FLIP protocol can provide mutual authentication and establish a shared session key between two like-minded vehicles. However, more importantly, it is a provably secure protocol suitable for VANET scenarios.

6.8 Summary

Secure finding like-minded vehicles protocol (FLIP) can protect vehicle's IP and is of vital importance to the success of vehicle chatting application on the road, yet it has not been paid enough attention in VANETs. In this chapter, based on the elliptic-curve technique, we have proposed an efficient IP-preserving FLIP protocol. With the provable security technique, the proposed FLIP protocol has been demonstrated to be secure in the VANET scenarios. In addition, extensive simulations have also been conducted to its practical considerations, i.e., how to balance the level of IP-preserving and the delay of finding like-minded vehicles on the road. Because the proposed FLIP protocol keeps each other's IP-preserving if two vehicles do not have the common interest, it can be widely accepted by the public.

Chapter 7

Practical Incentive Protocol for Vehicular DTNs

7.1 Introduction

In the previous chapter, we discussed privacy-preserving protocols for finding like-minded vehicles in vehicular social networks. However, due to the resource constraints, some vehicles could behavior selfishly in vehicle social networks. To address this issue, in this chapter, we will present a practical incentive (Pi) protocol for packet forwarding applications in Vehicular DTNs [48].

The opportunistic data propagation in DTNs has been well studied so far, and several efficient opportunistic routing protocols have been proposed under the hypothesis that each individual DTN node is willing to forward bundles for others [98, 99, 100]. However, when DTN nodes are controlled by rational entities, such as human or organization [101, 102], some DTN nodes will behave selfishly and may not be willing to help others to forward bundles, so the hypothesis will be violated [103, 104]. For example, in order to conserve power, buffer and computing resources, a selfish DTN node may be reluctant in the cooperation that is not directly beneficial to it, which could make a well designed opportunistic routing useless. Therefore, how to efficiently and effectively resolve the selfishness problem in DTNs has become a very challenging issue to achieve better packet delivery performance of DTNs.

To stimulate the possible selfish nodes to forward packets, many reputation-based and credit-based incentive protocols for wireless ad hoc networks have been proposed [105,

106, 107, 108, 109, 110]. However, due to the unique features of DTNs, such as the lack of contemporaneous path and high variation in network conditions, it is hard to detect DTN nodes' selfish behaviors or predetermine a routing path. Therefore, these challenges in DTNs make the existing incentive protocols, which usually rely on a contemporaneous routing, not applicable to DTNs.

In this chapter, in order to improve the performance of DTNs in terms of high delivery ratio and low average delay, we propose a Practical incentive (Pi) protocol to address the selfishness problem in DTNs. In the proposed protocol, when the source DTN node sends a bundle, it does not set a routing path in advance, but only needs to attach some incentive on the bundle. Then, the selfish DTN nodes on the road could be stimulated to help with forwarding the bundle to improve the delivery ratio and reduce the average delay of the whole DTNs. Specifically, the contributions of this chapter are threefold.

- First, we provide a fair incentive model in which selfish DTN nodes are stimulated to help forward bundles with credit-based incentive as well as reputation-based incentive. In the reward model, to achieve fairness, if and only if the bundles arrive at the destination node, the intermediate forwarding nodes can get credits from the source node. Furthermore, for the failure of bundle forwarding, those intermediate forwarding nodes still can get good reputation values from a trusted authority (TA). Therefore, with this stimulation, the packet delivery performance of DTNs can be improved. To the best of our knowledge, no previously reported stimulation schemes provide the fairness in DTNs.

- Second, in order to guarantee the feasibility of the fair incentive model, we use the layered coin model [104, 28] and verifiably encrypted signature techniques [111, 112] to provide authentication and integrity protection in the proposed Pi protocol.

- Third, to confirm the effectiveness of the proposed Pi protocol, we also develop a custom simulator built in Java to substantially show that the proposed Pi protocol can achieve the high delivery ratio and low average delay of DTNs when the high incentive is provided.

The remainder of this chapter is organized as follows. In Section 7.2, we formalize the network model, the node model, and identify the design goal. Then, we present the Pi protocol in Section 7.3, followed by the security analysis and performance evaluation in Section 7.4 and Section 7.5, respectively. We also review related work in Section 7.6. Finally, we draw our summary in Section 7.7.

7.2 Models and Design Goal

In this section, we formalize the network model, the node model, and identify the design goal.

7.2.1 Network Model

Delay Tolerant Networks (DTNs) are typically characterized by the unguaranteed connectivity and the low frequency of encounters between a given pair of nodes within the network [60]. In our model, we consider a DTN as a directed graph $G = (V, E)$, where V and E represent the set of DTN nodes and opportunistic contact edges, respectively. In the DTN, a source S can deliver packets to a destination D via the movement of DTN nodes with proper data forwarding algorithm. Currently, contingent upon whether they allow multiple copies of a message relaying within the network, the existing data forwarding algorithms may be categorized into single-copy and multi-copy algorithms. In the single-copy algorithm [100], only one copy is relayed in the network until it arrives at the destination. While in the multi-copy algorithms, such as flooding or spray routing [98], more than one copy are relayed in the networks. Due to large number of message copies in the networks, this kind of approach consumes a high amount of resources which are scarce in DTNs. In this work, in order to clearly illustrate the practical incentive, we just consider a single-copy data forwarding algorithm, i.e., for each bundle B, only one copy is initially spread by the source S, then the only copy is opportunistically relayed from one forwarding node to another until its reaching the destination D.

7.2.2 Node Model

In DTNs, the selfish behaviors of DTN nodes are naturally caused by human entities who control them [101, 102]. In our model, in order to study the selfish DTN nodes in a non-abstract fashion, we take the vehicular ad hoc network as a concrete delay tolerant network — vehicular DTN, where each DTN node is instantiated by vehicle driven by people running in a city environment with some velocity. In the rest of this chapter, we will use the terms "node" and "vehicle" interchangeably to refer to the same DTN entity.

In vehicular DTNs, each vehicle is equipped with On Board Unit (OBU) communication device, which allows different vehicles to communicate with each other based on the 802.11p protocol [14]. Note that the 802.11p physical layer offers different bitrates, ranging from 3 to 27 Mbps, from which OBU devices can choose [67]. Therefore, when two vehicles

are within the transmission range, e.g., 300 meters, they can exchange bundles [14]. In general, a vehicle is almost resource-unlimited, while the equipped OBU communication device is considered resource-constrained, i.e., buffer and computation power constraints [113]. Therefore, there may exist many selfish DTN nodes in the networks. In order to conserve buffer space, these selfish DTN nodes may be very reluctant in the cooperation that is not directly beneficial to them. As a result, the selfishness would be against the goal of the vehicular DTN to cooperatively deliver a bundle from its source S to the destination D. Therefore, the cooperation probability of a selfish DTN node can be modeled as follows

$$P_c = \alpha P_s + (1-\alpha)P_u = \alpha P_s + 1 - \alpha \tag{7.1}$$

where $0 \leq \alpha \leq 1$ is the *selfish factor*, $P_s < 1$ is the cooperation probability under selfish condition, i.e., $P_s = 0.01$, while $P_u = 1$ denotes the unselfish cooperation probability. Clearly, if $\alpha = 0$, a DTN node is unselfish, i.e., it is always willing to help with forwarding with probability $P_c = 1$. On the contrary, if $\alpha = 1$, the DTN node is selfish, the cooperation probability is just $P_c = P_s = 0.01$. Therefore, the smaller the *selfish factor* α, the better the cooperation in DTNs.

7.2.3 Design Goal

Our design goal is to develop a practical incentive protocol to stimulate the selfish DTN nodes to improve the cooperation probability P_c in the networks. Specifically, the following two desirable objectives will be achieved.

- *Improving DTN's performance with stimulation:* In order to prevent the overall performance degradation, i.e., low delivery ratio and high average delay, due to the selfish DTN nodes in DTNs, the credit-based incentive strategy is adopted. Similar to [104], the basic strategy is to provide incentives for intermediate forwarding DTN nodes to faithfully forward bundles. Generally, the intermediate nodes will get paid for bundle forwarding from the other nodes, and will take the same payment mechanism to pay for their bundle forwarding requests, by which the overall performance (i.e., high delivery ratio and low average delay) of the DTNs can be assured.

- *Fairness:* In the practical incentive protocol, the fairness is also considered. Concretely, the intermediate forwarding DTN nodes can receive credits if and only if the destination node receives the bundles, which is fair to the source node. At the same time, even though the bundles do not arrive at the destination, those intermediate DTN nodes who participated in relaying still can get good reputation values for their

cooperations. Because a good reputation can build other DTN nodes' confidence in helping forward the bundles (when the reputation value is higher than a reputation threshold R_{th}), the fairness can further stimulate DTN nodes to improve the DTN's packet delivery performance.

Incentive Strategy

To achieve the above objectives, the following hybrid incentive strategy is adopted.

- There exists a trusted authority (TA) in the system similar to [110]. Although it does not participate in bundle forwarding in DTNs, TA performs trusted fair credit and reputation clearance for DTN nodes. Therefore, before joining the DTNs, each DTN node should register itself to the TA and obtain its personal credit account (PCA) and personal reputation account (PRA) in the initialization phase. Later, when a DTN node has an available fast connection to the TA, it can report to the TA for credit and/or reputation clearance [110]. For example, in the vehicular DTN, a vehicle can communicate with TA for clearance when it makes contact with some RoadSide Units (RSUs). For each DTN node, PCA stores its credits, while PRA records its dynamic reputation value as follows: Let $\mathrm{R}_{IP(n-1)}$ be the DTN node's reputation value at time T_{n-1}. Then, the new reputation value $\mathrm{R}_{IP(n)}$ at time T_n is formulated as $\mathrm{R}_{IP(n)} = e^{-\lambda T_i} \cdot \mathrm{R}_{IP(n-1)} + C_{T_i}$, where $T_i = T_n - T_{n-1}$, λ is the rate at which the reputation value would decrease, and C_{T_i} denotes the reputation cumulative function, which is the summation of new gained reputation values in the time period T_i.

- It is not mandatory for the intermediate DTN node to forward bundles. All intermediate nodes in the DTN network can self-determine whether or not to participate in bundle forwarding.

- However, once an intermediate DTN node participates in forwarding bundle, it can get the credits from the source node as well as reputation values from the TA.

- If the bundle does not arrive at the destination node, the source node will not need to pay credits. However, those intermediate nodes who helped forward can still get good reputation values from the TA. Based on the above reputation calculation, if no new reputation value is gained in T_i, i.e., $C_{T_i} = 0$, then $\mathrm{R}_{IP(n)} = e^{-\lambda T_i} \cdot \mathrm{R}_{IP(n-1)}$ will decrease with the time. The larger the parameter λ, the quicker the reputation value $\mathrm{R}_{IP(n)}$ decreases. Therefore, in order to keep/increase good reputation values, this fair incentive strategy is attractive to each DTN node.

The design of reward calculation is the pivot of a practical incentive protocol, which should guide the selfish DTN nodes to follow the protocol to help with forwarding bundles. In the incentive model, the following reward calculation is exercised: once an intermediate DTN node N_i helped forward a bundle for Dis_i distance, it can get a reward either $Dis_i \cdot \mathrm{C}_{IP} + Dis_i \cdot \mathrm{R}_{IP}$ if the bundle B arrives at the destination D finally or $Dis_i \cdot \mathrm{R}_{IP}$ otherwise, i.e.,

$$\mathrm{Reward}_i = \begin{cases} Dis_i \cdot \mathrm{C}_{IP} + Dis_i \cdot \mathrm{R}_{IP}, & \text{if B arrives at } D; \\ Dis_i \cdot \mathrm{R}_{IP}, & \text{otherwise.} \end{cases} \tag{7.2}$$

where C_{IP} is a unit incentive credit provided by the source S, R_{IP} is a fixed unit reputation value defined by the TA for optimizing the network. Assume that C_F is the unit resource cost used for forwarding. We define the *gaining factor* of DTN node N_i as

$$\zeta_i = \frac{Dis_i \cdot \mathrm{C}_{IP} - Dis_i \cdot \mathrm{C}_F}{Dis_i \cdot \mathrm{C}_F} = \frac{\mathrm{C}_{IP} - \mathrm{C}_F}{\mathrm{C}_F} \tag{7.3}$$

and redefine the cooperation probability of N_i with reputation value R_{IP} as

$$P_c = \begin{cases} 1, & \textbf{if } R_{IP} < R_{th}; \\ & \textbf{else if } R_{IP} \geq R_{th} \\ 1, & \alpha_i - \zeta_i \leq 0; \\ (\alpha_i - \zeta_i)P_s + 1 - (\alpha_i - \zeta_i), & \alpha_i - \zeta_i > 0. \\ & \textbf{end if} \end{cases} \tag{7.4}$$

Then, with the cooperation probability P_c, the DTN node N_i is interested in helping forward the bundle. Note that, when $R_{IP} \geq R_{th}$, different intermediate DTN node may have different initial selfish factor α_i. Therefore, to guarantee the success of stimulation on all intermediate DTN nodes, the source S can choose a large C_{IP} (i.e., large gaining factor ζ_i) in its incentive policy such that each $\alpha_i - \zeta_i$ can be minimal. In addition, since Reward_i is a linear increase function of Dis_i in Eq. (7.2), the longer the Dis_i, the more the Reward_i. Therefore, the intermediate node is willing to forward the bundle as long as possible.

Layered Coin Model

To guarantee the incentive strategy working well, the incentive must be secure. Therefore, in the implementation, we use the layered coin to stimulate the bundle delivery [104, 28]. A

typical layered coin usually consists of a *base layer* formed by the source node and multiple *endorsed layers* formed by the intermediate nodes. Fig. 7.1 shows an example of layered coin architecture, where (S, L_s), (D, L_d), (N_i, L_i) are the source node and its location, the destination node and its location, and the i-th intermediate node and the location that it contacts with the $i+1$-th node, respectively. IP is the incentive policy provided by the source node S, TTL, TS, and Sig_i refer to the time-to-live information, the timestamp, and the signature, respectively. IP includes the source's reputation value R_{IP} signed by TA and the incentive policy in this bundle packet forwarding, i.e., the incentive in Eq. (7.2), and the signatures $Sig_0, Sig_1, \cdots$ can witness the cooperation among DTN nodes while preventing possible malicious nodes from disrupting the system.

Overhead of layered coin. Except the signature fields, we assume the IP field is 64-byte length, and all other fields are 8-byte length, then the overhead of a n-layered coin is around $120 + 32 \cdot n + |Sig| \cdot (n+2)$ bytes, where $|Sig|$ denotes the length of adopted signature.

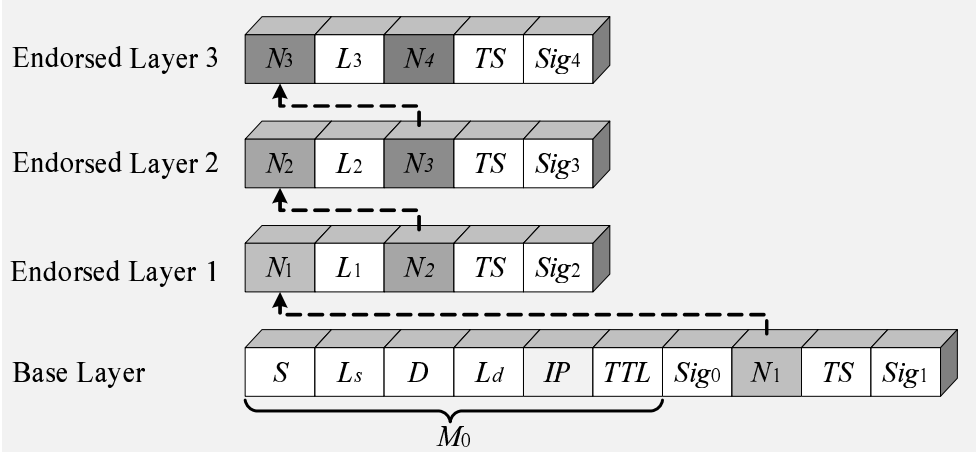


Figure 7.1: An example of layered coin architecture

7.3 Practical Incentive Protocol

In this section, we propose Pi protocol, which consists of four parts: system initialization, bundle generation, bundle forwarding, and charging and rewarding.

7.3.1 System Initialization

We assume that all DTN nodes $\mathcal{N} = \{N_1, N_2, \cdots\}$ and TA are using the same suite of system parameters. Given the security parameter k, the bilinear parameters $(q, g, \mathbb{G}, \mathbb{G}_T, e)$ are first generated by running $\mathcal{G}en(k)$. Then, a cryptographic hash function $H : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and a secure symmetric encryption algorithm $\mathcal{E}()$ are chosen [51]. In the end, the system parameter $\mathsf{params} = (q,\ g, \mathbb{G},\ \mathbb{G}_T, e, H, \mathcal{E})$ are published.

Each DTN node with a unique identity $N_i \in \mathcal{N}$ chooses a random number $x_i \in \mathbb{Z}_q^*$ as its private key and computes the corresponding public key as $y_i = g^{x_i}$. At the same time, each DTN node $N_i \in \mathcal{N}$ also registers its personal credit account (PCA) and personal reputation account (PRA) to the TA. Note that, all public keys in the system should be certified by public key certificates issued by certificate authority (CA). In addition, each DTN node's reputation value R_{IP} during a period is signed by TA and anyone can check it.

7.3.2 Bundle Generation

When a source node S with the private-public key pair $(x_s, y_s = g^{x_s})$ at location L_s wants to send a bundle m to the destination node D with the key pair $(x_d, y_d = g^{x_d})$ at location L_d, S will run the following steps.

Step 1. Compute the static shared key $k_{sd} = y_d^{x_s} = g^{x_s x_d}$ between S and D, and encrypt the bundle m into $\mathrm{B} = \mathcal{E}_{k_{sd}}(m)$ to achieve confidentiality.

Step 2. Determine a proper incentive policy (IP) as in Eq. (7.2), and make a verifiably encrypted signature σ_0 on $M_0 = S||L_s||D||L_d||IP||TTL$ and B as $\sigma_0 = y_d^{(H(M_0||B)+x_s)^{-1}}$.

When an intermediate node N_1 is interested in the IP and willing to forward the bundle to a possible location L_1, it first checks the source's reputation value R_{IP} and verifies the validity of σ_0 with the equation $e(\sigma_0, g^{H(M_0||B)} \cdot y_s) \stackrel{?}{=} e(y_d, g)$. If the source's reputation is acceptable, i.e., $R_{IP} \geq R_{th}$, and the equation holds, N_1 signs $\sigma_1^* = g^{(H(M_0||N_1||L_s||TS)+x_1)^{-1}}$ as an *Interest Acknowledgement* (ACK), and sends σ_1^* and L_1 to the source node S. After receiving σ_1^* and L_1, the source node S runs the next steps.

Step 3. Verify the validity of ACK by checking the equation $e(\sigma_1^*, g^{H(M_0||N_1||L_s||TS)} \cdot y_1) \stackrel{?}{=} e(g, g)$. If it holds, S makes the signature σ_1 on $M_0||N_1||L_s||TS$ as $\sigma_1 = g^{(H(M_0||N_1||L_s||TS)+x_s)^{-1}}$. Otherwise, S neglects the ACK.

Step 4. Set the *base layer* as $\mathrm{BL} = (\mathrm{M}_0||\sigma_0||N_1||TS\ ||\sigma_1)$ and forward the bundle B together with the base layer BL to the intermediate node N_1 as follows

$$S \rightarrow N_1 : \mathrm{B}, \mathrm{BL} \tag{7.5}$$

After verifying $\sigma_1 = g^{(H(M_0||N_1||L_s||TS)+x_s)^{-1}}$ by checking $e(\sigma_1, g^{H(M_0||N_1||L_s||TS)} \cdot y_s) \stackrel{?}{=} e(g, g)$, N_1 begins to forward the bundle.

7.3.3 Bundle Forwarding

When approaching to the location L_1, the intermediate node N_1 considers it cannot carry the bundle B close to the destination node D any more and forwards the bundle to the next-hop DTN node by running the Algorithm 8. Likewise, each subsequent forwarding node also uses the Algorithm 8 to forward the bundles. Without loss of generality, the bundle B finally arrives at the destination node D by opportunistic bundle forwarding with the routing $S \rightarrow N_1 \rightarrow N_2 \rightarrow \cdots \rightarrow N_l \rightarrow D$, as shown in Fig. 7.2. In the following, the detailed bundle forwarding protocol is described.

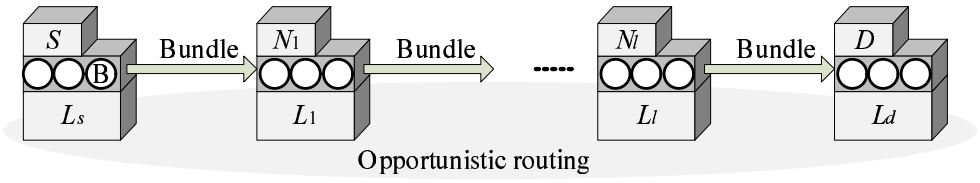


Figure 7.2: An opportunistic routing in DTN

At the location L_i, the intermediate node N_i, $1 \leq i \leq l-1$, is ready to forward the bundle to the next-hop node N_{i+1}, the following steps are executed.

Step 1. When the intermediate node N_{i+1} is interested in forwarding the bundle B, it first checks the source S's reputation value embedded in IP and the validity of $(\sigma_0, \cdots, \sigma_i)$. If the source's reputation value is acceptable and $(\sigma_0, \cdots, \sigma_i)$ are valid, N_{i+1} signs

$$\sigma_{i+1}^* = \sigma_0^{x_{i+1}} \cdot \sigma_1^{x_{i+1} H(N_i||L_i||N_{i+1}||TS)} \tag{7.6}$$

as an ACK to the N_i.

Step 2. After receiving the ACK σ_{i+1}^*, the intermediate node N_i checks

$$e(\sigma_{i+1}^*, g) \stackrel{?}{=} e(\sigma_0, y_{i+1}) \cdot e\left(\sigma_1, y_{i+1}^{H(N_i||L_i||N_{i+1}||TS)}\right) \tag{7.7}$$

Algorithm 8 Bundle forwarding

Data: When approaching to the location L_1, the node N_1 sets a *holding time to wait next-hop node* (T_h), and tries to forward the bundle B to the next-hop DTN node within T_h
procedure BUNDLE FORWARDING
    if a DTN node N_2 is interested in forwarding within T_h **then**
        N_1 checks the possible location L_2 that N_2 can carry the bundle B to
        if location L_2 is closer to the destination D than L_1 **then**
            N_1 forwards the bundle B to N_2
        else
            N_1 continues to wait other DTN node which is interested in forwarding
        end if
    else
        when there is no DTN node which is interested in forwarding the bundle at location L_1, N_1 has to drop the bundle packet, since the next-hop route is not immediately available
    end if
end procedure

If it holds, N_i computes

$$\sigma_{i+1} = \sigma_0^{x_i} \cdot \sigma_1^{x_i H(N_i||L_i||N_{i+1}||TS)} \tag{7.8}$$

and sets the i-th endorsed layer as $\mathrm{EL}_i = (N_i||L_i\,||N_{i+1}||TS\,||\sigma_{i+1})$ and forwards the bundle packet B to the next node N_{i+1} as follows

$$N_i \rightarrow N_{i+1} : \mathrm{B}, \mathrm{BL}, \mathrm{EL}_1, \cdots, \mathrm{EL}_i \tag{7.9}$$

Step 3. After verifying the validity of σ_{i+1} by checking

$$e(\sigma_{i+1}, g) \stackrel{?}{=} e(\sigma_0, y_i) \cdot e\left(\sigma_1, y_i^{H(N_i||L_i||N_{i+1}||TS)}\right) \tag{7.10}$$

the intermediate node N_{i+1} forwards the bundle packet B.

At the location L_d, the last intermediate node N_l forwards the bundle $(\mathrm{B}, \mathrm{BL}, \mathrm{EL}_1, \cdots, \mathrm{EL}_{l-1})$ to the destination node D. After the destination node D checks the signatures $(\sigma_0, \cdots, \sigma_l)$ and correctly recovers m from $\mathrm{B} = \mathcal{E}_{k_{sd}}(m)$, it signs a special signature $\sigma_{l+1} = \sigma_0^{x_d^{-1}}$ such that

$$\sigma_{l+1} = y_d^{(x_d \cdot (H(M_0||B)+x_s))^{-1}} = g^{(H(M_0||B)+x_s)^{-1}} \tag{7.11}$$

and sends σ_{l+1} back to the last intermediate node N_l. After verifying the validity of σ_{l+1} by checking $e(\sigma_{l+1}, g^{H(M_0||B)} \cdot y_s) \stackrel{?}{=} e(g, g)$, N_l can submit $(\sigma_0, \cdots, \sigma_{l+1})$ to the TA for clearance in the future.

7.3.4 Charging and Rewarding

When the last intermediate node N_l has an available fast connection to the TA, N_l reports $(\sigma_1, \cdots, \sigma_l)$ to the TA, then the TA performs the fair credit and reputation clearance as the following steps.

Step 1. TA checks the freshness and the validity of $(\sigma_0, \cdots, \sigma_{l+1})$. If they are fresh and valid, TA continues; otherwise terminates the operation.

Step 2. Based on the locations $(L_s, L_1, \cdots, L_l, L_d)$ in the signatures, TA measures the actual relay distance of each intermediate node. Then, according to the incentive policy in IP, TA stores the merited credits and reputation values in each intermediate node's PCA and PRA, and withdraws the corresponding credit values from the source node's PCA, as shown in Algorithm 9.

Algorithm 9 Credit and reputation clearance

1: **Data:** The TA obtains valid signatures $(\sigma_1, \cdots, \sigma_l)$ from the last intermediate node N_l
2: **procedure** CREDIT AND REPUTATION CLEARANCE
3: get the location information $(L_s, L_1, \cdots, L_l, L_d)$ from these signatures
4: measure each intermediate node N_i's actual relay distance Dis_i, where $Dis_1 = |L_1 - L_s|$, $Dis_l = |L_d - L_l|$ and $Dis_i = |L_i - L_{i-1}|$, where $2 \leq i \leq l$
5: **for** $i = 0$ to l **do**
6: according to the incentive policy in IP, withdraw $C_i = L_i \times C_{\mathrm{IP}}$ from the source node S's PCA, and store the merited credits C_i in N_i's PCA
7: store $R_i = Dis_i \times R_{IP}$ reputation values in N_i's PRA based on the reputation calculation
8: **end for**
9: **end procedure**

If the bundle packet does not arrive at the destination node D, each intermediate node $N_i \in \mathcal{N}$, which helped forwarding, still can get the good reputation value by submitting σ_i and σ^*_{i+1}. As shown in Algorithm 10, from the locations L_{i-1} in $\sigma_i = \sigma_0^{x_{i-1}} \cdot \sigma_1^{x_{i-1}H(N_{i-1}||L_{i-1}||N_i||TS)}$ and L_i in $\sigma^*_{i+1} = \sigma_0^{x_{i+1}} \cdot \sigma_1^{x_{i+1} \cdot H(N_i||L_i||N_{i+1}||TS)}$, TA can compute the relay distance, and store the merited reputation values to N_i's PRA.

Correctness. The correctness of σ_0, σ_1 and σ_{i+1} are given as follows:

$$\begin{aligned} & e(\sigma_0, g^{H(M_0||B)} \cdot y_s) \\ =& e\left(y_d^{\frac{1}{H(M_0||B)+x_s}}, g^{H(M_0||B)} \cdot y_s\right) = e(y_d, g) \end{aligned} \tag{7.12}$$

Algorithm 10 Reputation clearance

1: **Data:** The TA obtains valid signatures $(\sigma_i, \sigma_{i+1}^*)$ from the intermediate node N_i
2: **procedure** REPUTATION CLEARANCE
3: get the location information LN_{i-1} in σ_i and LN_i in σ_{i+1}^*
4: measure the intermediate node N_i's actual relay distance $Dis_i = |L_i - L_{i-1}|$
5: store $R_i = Dis_i \times R_{IP}$ reputation values in N_i's PRA based on the reputation calculation
6: **end procedure**

$$\begin{aligned} & e(\sigma_1, g^{H(M_0||N_1||L_s||TS)} \cdot y_s) \\ =& e\left(g^{(H(M_0||N_1||L_s||TS)+x_s)^{-1}}, g^{H(M_0||N_1||L_s||TS)} \cdot y_s\right) \\ =& e(g,g) \end{aligned} \tag{7.13}$$

$$\begin{aligned} e(\sigma_{i+1}, g) &= e\left(\sigma_0^{x_i} \cdot \sigma_1^{x_i H(N_i||L_i||N_{i+1}||TS)}, g\right) \\ &= e\left(\sigma_0^{x_i}, g\right) \cdot e\left(\sigma_1^{x_i H(N_i||L_i||N_{i+1}||TS)}, g\right) \\ &= e(\sigma_0, y_i) \cdot e\left(\sigma_1, y_i^{H(N_i||L_i||N_{i+1}||TS)}\right) \end{aligned} \tag{7.14}$$

Similarly, the correctness of σ_i^* can also be checked. Then, due to the hybrid incentives, the DTN nodes will be stimulated to faithfully forward the bundles to the destination nodes in a cooperative fashion.

Communication Overhead. Similar to the BLS signature [114], each signature $\sigma_i, i = 0, 1, \cdots$, can be implemented as short as 160 bits (= 20 bytes). Then, the overhead of l-layered coin is $160 + 52 \cdot l$ bytes. When $l = 20$ is assumed, the overhead of layered coin is only $1,200$ bytes (≈ 1.17 Kb). Assume each bundle is 2 Mb or more, then the overhead of layered coin is much smaller than 2 Mb and acceptable for providing security in vehicular DTNs.

Aggregation and Batch Verification. In the proposed Pi protocol, each signature's signing cost is very low, only exponentiation operation is required. However, since the verification requires pairing operation, the computation cost becomes a little higher, but still less than 20 ms [14]. In order to further reduce the communication and computation overheads, the signatures $\sigma_2, \sigma_3, \cdots, \sigma_l$ in the proposed Pi protocol can be aggregated as

$$\sigma = \sigma_2 \cdot \sigma_3 \cdot \cdots \cdot \sigma_l = \sum_{i=2}^{l} \sigma_i \tag{7.15}$$

Then, the aggregated signature σ can be batch-verified as

$$e(\sigma, P) = e\left(\sigma_0, \prod_{i=1}^{l-1} y_{i+1}\right) \cdot e\left(\sigma_1, \prod_{i=1}^{l-1} y_{i+1}^{H(N_i||L_i||N_{i+1}||TS)}\right) \tag{7.16}$$

Clearly, the correctness of Eq. (7.16) directly follows from Eq. (7.14). Because the signatures σ_0, σ_1 are provably secure in the random oracle model [111, 112] and the CDH problem is also assumed hard in $\mathbb{G}$, the signature in Eq. (7.14) is secure. Then, the security of $\sigma = \sum_{i=1}^{l-1} \sigma_{i+1}$ also follows. More details on security proof of σ can be found in [111, 112].

7.4 Security Analysis

In this section, we discuss security issues of the proposed Pi protocol, i.e., the fairness issue in stimulation, the free ride attack [115], the layer removing attack [104], and the layer adding attack. Note that, since the proposed Pi protocol only deals with the selfish DTN nodes in DTNs, other attacks launched by malicious DTN nodes are out of the scope of this chapter.

• *The proposed Pi protocol provides fair incentive.* In the charging and rewarding phase, if *i*) a bundle is really relayed to the destination node, the source node S will pay credits to those intermediate nodes for forwarding. However, if *ii*) the bundle fails to reach the destination node, the source node S will not pay any credits. Therefore, it is fair to the source node. For the intermediate nodes, although they cannot get credits for their forwarding in case *ii*), they still can increase their good reputation values from the TA. When the gaining factor ζ_i is large, those intermediate nodes still feel fair for bundle forwarding. In addition, since the provably secure short signature schemes are employed [111, 112], the authentications from the signatures can provide strong witnesses. If an intermediate node didn't participate in forwarding, it cannot get any reward. Therefore, from the above analysis, the proposed Pi protocol can provide fair incentive in the DTN network.

• *The proposed Pi protocol is resistent to the free riding attack.* The free riding attack is a notorious selfish attack in DTN, which is conducted by two selfish DTN nodes that attempt to exchange messages without paying their credits [115]. If these two DTN selfish nodes are neighbor, this attack makes no sense, since they can directly exchange messages without the aid of others. When there is at least one normal node residing between them, launching such an attack is possible. Assume that the intermediate node N_i

wants to send message m' to N_{i+2} by piggybacking it with the forwarded bundle packet $(\mathrm{B}, \mathrm{BL}, \mathrm{EL}_1, \cdots, \mathrm{EL}_i)$. Since the signature $\sigma_0 = y_d^{(H(M_0||B)+x_s)^{-1}}$ can provide the integrity protection on $(\mathrm{M}_0, \mathrm{B})$, the free riding message m' will not pass the verification equation. Thus, the intermediate node N_{i+1} can detect the free riding message m' and delete it before passing the bundle message to the node N_{i+2}. As a result, the proposed Pi protocol is resistant to the free riding attack in the DTN network.

• *The proposed Pi protocol is resistent to the layer removing attack.* The layer removing attack [104] refers to *i*) a selfish intermediate node removes previous layers on the forwarding path or *ii*) two selfish intermediate nodes remove the layers between them to maximize their credits. However, this attack can be thwarted by the proposed Pi protocol. In the bundle forwarding phase in Section 7.3.3, each intermediate node N_i holds two valid witnesses σ_i and σ_{i+1}^*, where $\sigma_i = \sigma_0^{x_{i-1}} \cdot \sigma_1^{x_{i-1}H(N_{i-1}||L_{i-1}||N_i||TS)}$ is signed by N_{i-1}, and $\sigma_{i+1}^* = \sigma_0^{x_{i+1}} \cdot \sigma_1^{x_{i+1}H(N_i||L_i||N_{i+1}||TS)}$ is signed by N_{i+1}. Note that, the first intermediate node N_1 gets the witness $\sigma_1 = g^{(H(M_0||N_1||L_s||TS)+x_s)^{-1}}$ from the source node, and the last intermediate node N_l gets the witness σ_{l+1} from the destination node. If a selfish intermediate node N_i launches the first kind of removing layer attack, after removing the previous layers, it cannot get $\sigma_i = g^{(H(M_0||N_i||L_s||TS)+x_s)^{-1}}$ from the source node. Therefore, it can be detected. If two selfish intermediate nodes launch the second kind of removing layer attack, those removed intermediated nodes can provide their witnesses to prove their participation. Thus, the selfish nodes will also be detected, and their reputation values will decrease.

A special removing layer attack, as shown in Fig. 7.3, is the last intermediate node N_l colludes with the source node S to remove all previous layers for enabling the source node to pay less rewarding credits. However, this special attack is still hard to launch. This is because the source node S does not know the last intermediate node N_l in advance in the DTN network. Even though S knows N_l and provides $\sigma_l = g^{(H(M_0||N_l||L_s||TS)+x_s)^{-1}}$ to N_l, it cannot deny its signing on $\sigma_1 = g^{(H(M_0||N_1||L_s||TS)+x_s)^{-1}}$. Therefore, the selfish behaviors of S and N_l in this special case can also be detected.

• *The proposed Pi protocol is resistent to the layer adding attack.* If a system allows a DTN node with multiple identities, then the layer adding attack could be launched. The layer adding attack refers to a selfish intermediate node with multiple identities adds some additional layers with its different identities on *i*) the same forwarding path or *ii*) detour the forwarding path to maximize its credits, as shown in Fig. 7.3. However, in the proposed Pi protocol, the $\mathrm{Reward}_i = Dis_i \cdot \mathrm{C}_{IP} + Dis_i \cdot \mathrm{R}_{IP}$ increases linearly with Dis_i. If these additional layers do not enlarge the actual distance Dis_i as in case *i*), the selfish node still cannot get more credits. In case *ii*), although Dis_i increases, TA can detect these

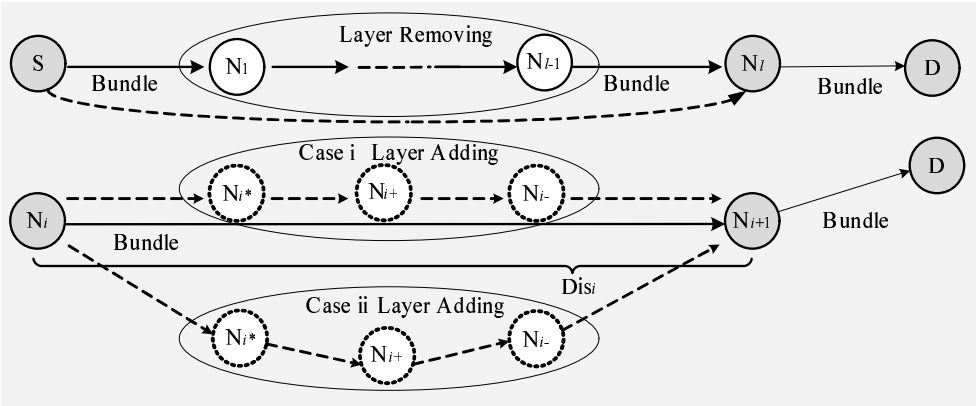


Figure 7.3: Layer removing / adding attacks in DTN

forwarding nodes N_{i^*}, N_{i+}, N_{i-} are the same node N_i at charging and rewarding phase, since the trusted authority TA knows all DTN node's PCA and PRA. In our system, since one DTN node holds only one unique identifer, and multiple identities are not allowed, this attack is prevented. Note that in DTN network, more than one DTN nodes collude with each other to launch layer adding attack is a malicious attack, how to resist it is still a challenging issue.

7.5 Performance Evaluation

In this section, we study the performance of the proposed Pi protocol using a custom simulator built in Java. The performance metrics used in the evaluation are i) the *delivery ratio*, which is the fraction of generated messages that are correctly delivered to the final destination within a given time period; 2) the *average delay*, which is defined as the average time between when a message is generated at some source and when it is successfully delivered to its destination. Both *delivery ratio* and *average delay* can be used to examine the ability of the proposed Pi protocol with some incentive strategy to deliver the bundle to the destination within a specified period.

7.5.1 Simulation Settings

In the simulations, total n DTN nodes with a transmission radius of 300 meters are first uniformly deployed in an area of 6,000 m $\times$ 15,000 m, as shown in Fig. 7.4, to simulate a

sparse vehicular DTN.

Mobility model. In vehicular DTNs, the performance of bundle forwarding is highly contingent upon the mobility of vehicles. Since vehicles are usually driven along the roads in a city, we assume each DTN node follows the *shortest path map based movement* routing. Specifically, each vehicle first randomly chooses a destination in the area, and gets there using the shortest route with the average velocity v. After reaching the destination, with 2-minute pause time, the vehicle randomly chooses a new destination and repeats the above.

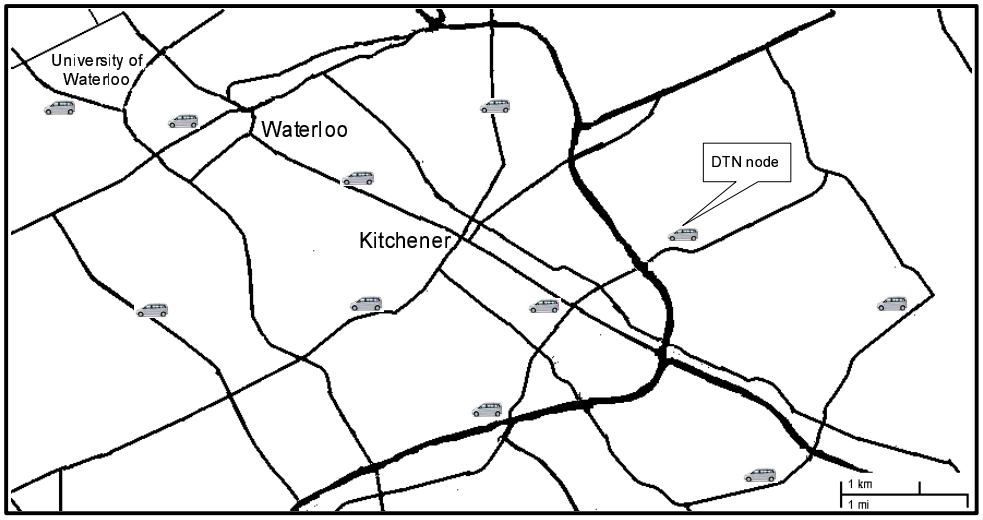


Figure 7.4: Vehicular DTN considered for simulation

Selfish ratio. Let $\rho = \frac{\text{the number of selfish DTN nodes}}{\text{the total number of DTN nodes}}$ be the selfish ratio (SR) among these DTN nodes, which usually is a variable based on how many DTN nodes that behave selfishly in the network [116]. Once a DTN node is selfish, then according to Eq. (7.4), it may refuse to forward the bundle packets if the gaining factor ζ is less than its selfish factor α when $R_{IP} \geq R_{th}$. However, with some incentives, i.e., the gaining factor ζ in Eq. (7.4) is increased, the selfish node may faithfully forward. Note that, in our simulation, we do not consider the case that $R_{IP} < R_{th}$. The reason is that, when $R_{IP} < R_{th}$, the selfish nodes will faithfully forward the bundles, which is equivalent to lowering the selfish ratio ρ in the simulation.

The detailed parameter settings in the simulations are summarized in Table 7.1. We perform the experiments for the specified period varying from 1 hour to 12 hours with increment of 1 hour. For each case, we run the simulation 10 times, and the average

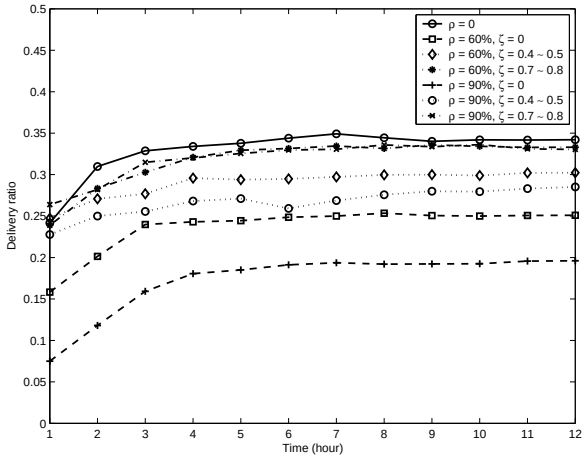
delivery ratio and *average delay* are reported.

Table 7.1: Simulation Settings in Pi

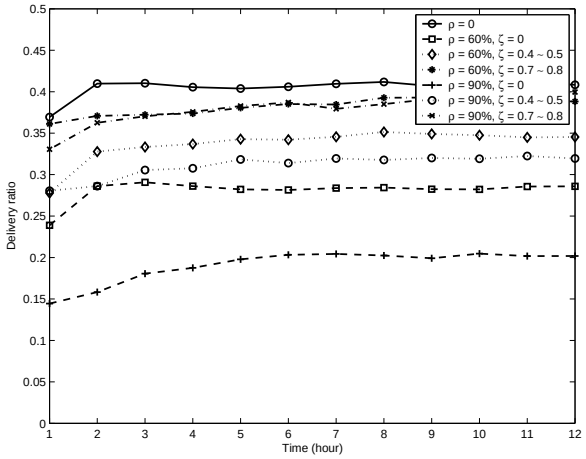
Parameter	Setting
Simulation area, duration	$6,000$ m $\times$ $15,000$ m, 12 hours
DTN nodes	
Number	$n = 60, 120$
Velocity	$v = 40 \pm 5$km/h, 80 ± 5km/h
Transmission range, buffer size	300 m, 20 Mb
Mobility model	shortest path map based movement
Holding time to wait next node	$T_h = 3$ minutes
Selfish factor of each DTN node	$\alpha \in [0.2, 0.8]$
Selfish ratio (SR)	$\rho = [0, , 60\%, 90\%]$
Bundle messages	
Generation interval, size, TTL	120 ± 20 s, 2 ± 0.5 Mb, 12 hours
Gaining factor of each bundle	$\zeta = 0, 0.4 \sim 0.5, 0.7 \sim 0.8$

7.5.2 Simulation Results

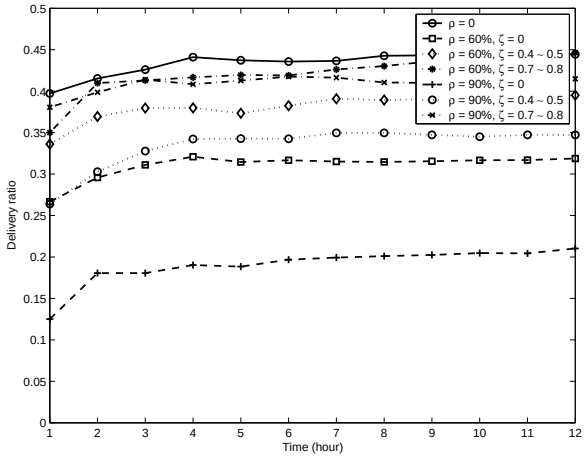
In Fig. 7.5, we compare the *delivery ratio* of the sampled DTN in different incentive policies, i.e., *without incentive* $\zeta = 0$, *with low incentive* $\zeta \in [0.4, 0.5]$ and *with high incentive* $\zeta \in [0.7, 0.8]$, under different selfish ratio $\rho = 0, 60\%, 90\%$. From the figure, we can see the delivery ratio without incentive is very low, especially when the selfish ratio $\rho = 90\%$. The reason is that many selfish DTN nodes move around the network, then there exist many dropping events in which *when a forwarding node seeks a next forwarding node at some location but only meets selfish nodes who are not willing to forward, the bundle message has to be dropped, since the next hop is not immediately available due to the selfishness*. Therefore, Fig. 7.5 shows that the larger the selfish ratio ρ, the more the dropping events take place and the lower the delivery ratio. On the other hand, when the network is stimulated with some incentive, the delivery ratio will increase. Because different selfish node has different selfish factor, the same incentive cannot satisfy all selfish nodes' stimulation conditions in Eq. (7.4). Therefore, there still exists a small fraction of selfish nodes. Intuitively, when the incentive is higher, the fraction of selfish nodes becomes smaller. By observing the figure, this intuition is corroborated, where the delivery ratio with high incentive is much higher than that with low incentive, and almost approaches to that with no selfish nodes, i.e., $\rho = 0$, in the DTN. Therefore, we can be sure that, when



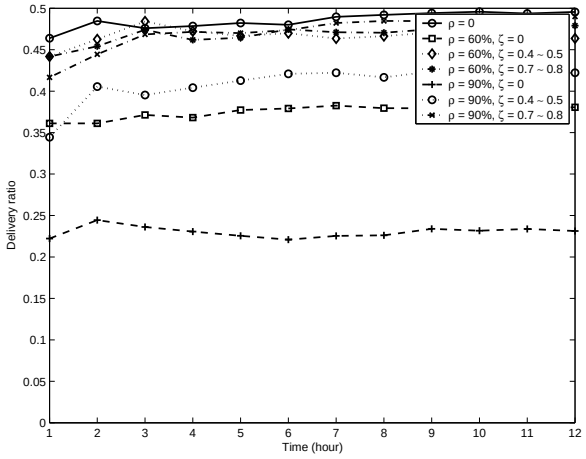
(a) Group 1: $n = 60$, $v = 40 \pm 5$km/h



(b) Group 2: $n = 60$, $v = 80 \pm 5$km/h



(c) Group 3: $n = 120$, $v = 40 \pm 5$km/h



(d) Group 4: $n = 120$, $v = 80 \pm 5$km/h

Figure 7.5: Delivery ratio varies with the specified period from 1 hour to 12 hours

choosing a proper incentive, the proposed Pi protocol can effectively stimulate the selfish nodes and improve the performance the DTN network in terms of high delivery ratio.

We further compare the delivery ratios in Group 1 and Group 2 in terms of different velocity. From the comparisons, we can see the delivery ratios in Group 2 are higher than those in Group 1. The reason is that the faster the velocity v, the more chances a DTN node can contact with other unselfish DTN nodes in time period T_h. As a result, the number of dropping events becomes small, and the delivery ratio increases. We also compare the delivery ratios in Group 1 and Group 3 in terms of different number of DTN nodes, and the comparisons show that the increase of DTN node's number will bring a positive affect on the delivery ratios. When the total number of DTN nodes increases, the density of unselfish DTN nodes subsequently increases. Then, a DTN node has more chances to contact with other unselfish DTN nodes, and the delivery ratio increases. The high delivery ratios in Group 4 with $n = 120$, $v = 80 \pm 5$ km/h further confirm our observations.

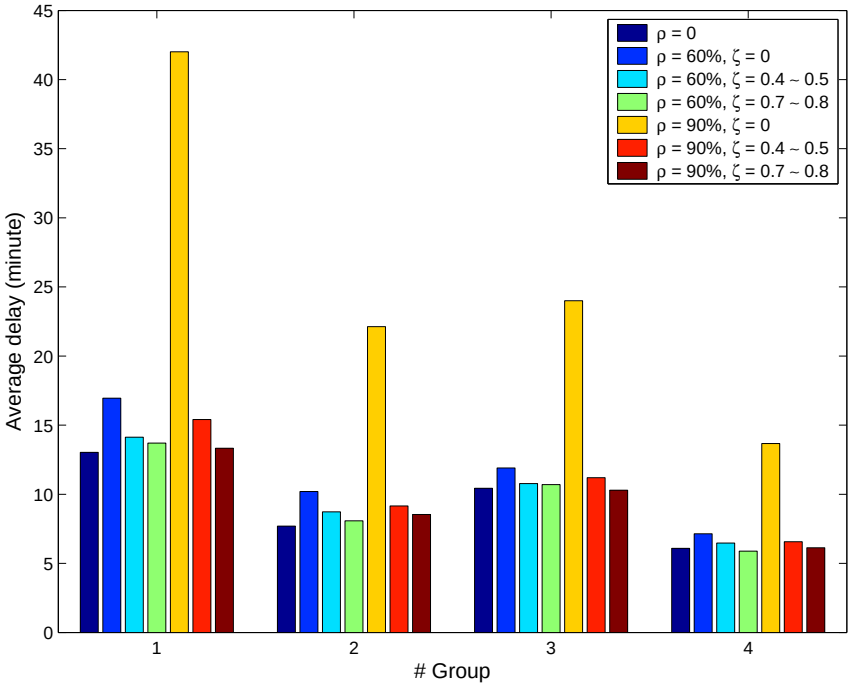


Figure 7.6: Average delay within 12 hours with different parameter settings

Fig. 7.6 depicts the average delay with 12 hours with different parameter settings. From the figure, we can see when there exist selfish nodes in DTN network, the average delay will decrease. The higher the selfish ratio ρ, the longer the average delay. However,

when the network is stimulated with some incentive, the average delay will decrease quickly. Especially, when the *high incentive* is exercised, i.e., the gaining factor ζ is around $0.7 \sim 0.8$, the average delays can approach to that with no selfish nodes, i.e., $\rho = 0$, in the DTN network. In addition, comparing the average delays in Groups 1, 2, 3 and 4, when the number of DTN nodes n and/or the velocity v increase, the average delay can be further reduced.

7.6 Related Work

In DTNs, the lack of contemporaneous routing and high variation in network conditions make the selfishness problem very different from the one in traditional wireless ad hoc network, and many existing incentive solutions can not be directly applied to DTNs. Recently, two research works on incentive-aware routing in DTNs have been appeared [103, 104], which are closely related to the proposed Pi protocol.

In [103], Shevade *et al.* first study the impact of selfish behaviors in DTNs. Based on the simulation results, they show that the presence of selfish DTN nodes can greatly degrade total delivered traffic. To mitigate the damage caused by selfish DTN nodes, they use the pair-wise tit-for-tat (TFT) as a simple, robust and practical incentive mechanism for DTNs, and develop an incentive-aware routing protocol that allows selfish DTN nodes to maximize their individual utilities while conforming to TFT constraints. Extensive simulation results are given to show that the TFT mechanism can increase total delivered traffic in the whole DTN network. Although Shevade *et al.*'s scheme is the first practical incentive-aware routing scheme for DTNs, the security issues lying in the incentive-based DTNs are not addressed in the work. In [104], Zhu *et al.* propose a secure multilayer credit-based incentive (SMART) scheme for DTNs affiliated with selfish nodes. In SMART, layered coins are used to provide incentives to selfish DTN nodes for bundle forwarding. In addition, compared with Shevade *et al.*'s scheme, several security issues lying in DTNs, i.e., credit forgery attack, nodular tontine attack, and submission refusal attack, are addressed in the SMART protocol, and the corresponding countermeasures are also briefly discussed.

Different from the SMART protocol, the proposed Pi protocol focuses on the fairness issue in DTNs. Specifically, we propose a hybrid (credit plus reputation) incentive model with verifiably encrypted signature technique to stimulate the selfish DTN nodes to help forward bundles. To achieve fairness, if and only if the bundles arrive at the destination node, the intermediate forwarding nodes can get credits from the source node. Furthermore, for the failure of bundle forwarding, those intermediate DTN nodes still can get good

reputation values from the trusted authority. Therefore, DTN nodes will be more confident in participating in bundle forwarding.

7.7 Summary

In this chapter, we have developed a practical incentive (Pi) protocol to stimulate selfish nodes in order to cooperate in forwarding bundle packets in vehicular DTNs. By adopting the proper incentive policy, the proposed Pi protocol can not only improve the whole vehicle DTN network's performance in terms of *high delivery ratio* and *low average delay* but also achieve the fairness among DTN nodes. Detailed security analyses have shown that the proposed Pi protocol can resist most attacks launched by selfish DTN nodes. In addition, extensive simulations have been conducted to demonstrate the effectiveness of the proposed Pi protocol. In our future work, we will integrate Pi with anonymity to provide each DTN node's privacy protection.

Chapter 8

Conclusions and Future Work

In this chapter, we summarize our contributions in this thesis, propose our future research work, and give our final remarks.

8.1 Contributions

The major contributions of this thesis can be summarized as follows:

- First, a novel social based privacy-preserving packet forwarding protocol, called SPRING, is proposed for vehicular DTNs, which is characterized by deploying RSUs at high social intersections to assist in packet forwarding between vehicles by temporarily storing packets through V-2-I communication during the period when the proper next-hop vehicles of these packets are not available. With such kind of RSU assistance, the probability of packet drop is reduced and as a result high reliability of packet forwarding in vehicular DTNs can be achieved. In addition, SPRING can also achieve conditional privacy preservation and resist most attacks existing in vehicular DTNs, such as packet analysis attack, packet tracing attack, and black (grey) hole attacks.

- Second, based on the "Sacrificing the Plum Tree for the Peach Tree" — one of the Thirty-Six Strategies of Ancient China, a socialspot-based packet forwarding (SPF) protocol for protecting receiver-location privacy, called SPF, is proposed for VANETs. Detailed security analysis on SPF has shown that, only when a receiver sacrifices one socialspot that it often visits, all its other sensitive locations can be protected against

an external, global, passive adversary. In addition, through extensive performance evaluation, we have demonstrated that the temporarily storing packets at socialspots can achieve much better efficiency in terms of delivery ratio and average delay in VANETs.

- Third, to facilitate vehicles to achieve high-level location privacy in VANETs, an effective pseudonyms changing at social spots strategy, called PCS, is proposed, where we have developed two anonymity set analytical models in terms of ASS to formally analyze the achieved location privacy level, and used game theoretic techniques to prove its feasibility. In addition, we introduced a practical KPSD model to mitigate the hazards caused by vehicle theft. Our analytical models on location privacy at social spot shed light on this research line.

- Fourth, to address the privacy issues in vehicle chatting application in vehicular social networks, an efficient privacy-preserving finding like-minded vehicle protocol, called FLIP, is proposed, which allows two vehicles with the common interest to identify each other and establish a shared session key, and at the same time, protects their Interest-Privacy (IP) from other vehicles who do not have the same interest on the road. To generalize the FLIP protocol, a lightweight privacy-preserving scalar product computation (PPSPC) protocol is also proposed, which, compared with previous PPSPC protocols, is more efficient in terms of computation and communication overheads.

- Fifth, to stimulate selfish nodes to cooperate in forwarding bundle packets in vehicular DTNs, a practical incentive protocol, called Pi, is proposed. By adopting the proper incentive policy, the proposed Pi protocol can not only improve the whole vehicle DTN network's performance in terms of high delivery ratio and low average delay but also achieve the fairness among DTN nodes. Detailed security analyses have shown that the proposed Pi protocol can resist most attacks launched by selfish DTN nodes. In addition, extensive simulations have been conducted to demonstrate the effectiveness of the proposed Pi protocol.

8.2 Future Work

Our research has already made significant progress in secure vehicular social networks. However, since vehicular social network is a promising platform in pervasive environments,

there still exist several research directions to be explored to complement this thesis. Therefore, the following two research topics will be investigated as a continuation of my Ph.D. thesis work.

- *Privacy-Preserving Communications with Blacklists for Vehicular Social Networks:* In privacy-preserving communications, once the misbehaviors caused by users are detected by system manager, tracing those unique identifiers and revoking their legitimate authority seem to be an ideal method and a natural consequence for maintaining the network working. However, in vehicular social network, the high dynamical topology and limited resource deployed on each vehicle may lead to the malfunctions on communication with some vehicles temporarily. As a result, those vehicles should not be considered as malicious ones and immediate revocation by system manager is not fair towards them. Moreover, if defining malfunction equals to misbehavior in a highly dynamic network, such as vehicular social network, the valid vehicles may become less and less, and eventually the network may collapse. Therefore, as one of our future works, for vehicular social networks, we propose a privacy-preserving communications with blacklists as follows

  1. For each vehicle, it maintains a blacklist which records all the identifiers of vehicles they cannot communicate successfully in previous period. From the vehicle's view, those vehicles on the blacklist are considered as malicious ones for this specific period and it will not communicate with them during this period.
  2. For system manager, it will periodically collect the blacklists from all the vehicles. If one's appearing times on different blacklists exceeds a preset threshold, the system manager will revoke the legitimate authority; otherwise, this vehicle's behavior is recognized as malfunction and the system manager will clarify its validity.

  With implementation of such concept in vehicular social networks, the communication between vehicles becomes more reliable from the whole system and meanwhile the personal blacklist does help for improving the local efficiency by avoiding the time delay caused by malfunction.

- *Supporting Reputation-Based Trust in Privacy-Preserving Vehicular Social Networks:* Based on the first future work, each vehicle could maintain a blacklist which is helpful for selecting nearby trusted peers. In addition, the vehicles are more likely to share their blacklists with others in vehicular social networks. Then, the so-called blacklist can be regarded as a list of trust value of contacted vehicles instead of those

malicious ones. Such distributed reputation-based scheme could help the system exclude misbehaviors more effectively, i.e., the vehicular communication could be made of more reliable vehicles. Moreover, the scheme should also provide privacy-preserving for each vehicle, which may lead more challenge in designing reputation-based mechanism.

8.3 Final Remarks

In this thesis, we have presented a suite of security and privacy-preserving protocols for secure vehicular social networks. In addition, we have also identified two future research topics to complement of this thesis. To facilitate our research accomplishments and findings to benefit the real world situations, we will carry out experiments to further confirm our research findings.

APPENDICES

Appendix A

Author's Publications

A.1 Journal Papers

[J1] **Rongxing Lu**, Xiaohui Liang, Xu Li, Xiaodong Lin, and Xuemin (Sherman) Shen, "EPPA: An Efficient and Privacy-Preserving Aggregation Scheme for Secure Smart Grid Communications", *IEEE Transactions on Parallel and Distributed Systems*, to appear.

[J2] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "A Dynamic Privacy-Preserving Key Management Scheme for Location Based Services in VANETs", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 13, No. 1, pp. 127-139, 2012.

[J3] **Rongxing Lu**, Xiaodong Lin, Tom. H. Luan, Xiaohui Liang, and Xuemin (Sherman) Shen, "Pseudonym Changing at Social Spots: An Effective Strategy for Location Privacy in VANETs", *IEEE Transactions on Vehicular Technology*, Vol. 61, No. 1, pp. 86-96, 2012.

[J4] **Rongxing Lu**, Xiaodong Lin, Haojin Zhu, Xiaohui Liang, and Xuemin (Sherman) Shen, "BECAN: A Bandwidth-Efficient Cooperative Authentication Scheme for Filtering Injected False Data in Wireless Sensor Networks", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 23, No. 1, pp. 32 - 43, 2012.

[J5] **Rongxing Lu**, Xu Li, Xiaohui Liang, Xiaodong Lin, and Xuemin (Sherman) Shen, "GRS: The Green, Reliability, and Security of Emerging Machine to Machine Com-

munication", *IEEE Communications Magazine*, Vol. 49, No. 4, pp. 28 - 35, 2011. (One of ten most popular articles published in ComSoc periodicals in April 2011)

[J6] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "A Secure Handshake Scheme with Symptoms-Matching for mHealthcare Social Network", *Mobile Networks and Applications*, Vol. 16, No. 6, pp. 683 - 694, 2011.

[J7] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "An Efficient and Provably Secure Public Key Encryption Scheme Based on Coding Theory", *Security and Communication Networks (Wiley)*, Vol. 4, No. 12, pp. 1440 - 1447, 2011.

[J8] **Rongxing Lu**, Xiaodong Lin, Haojin Zhu, Xuemin (Sherman) Shen, and Bruno Preiss, "Pi: A Practical Incentive Protocol for Delay Tolerant Networks", *IEEE Transactions on Wireless Communications*, Vol. 9, No. 4, pp. 1483-1493, 2010.

[J9] **Rongxing Lu**, Xiaodong Lin, Haojin Zhu, and Xuemin (Sherman) Shen, "An Intelligent Secure and Privacy-Preserving Parking Scheme through Vehicular Communications", *IEEE Transactions on Vehicular Technology*, Vol. 59, No. 6, pp. 2772-2785, 2010.

[J10] Z. Md. Fadlullah, Nei Kato, **Rongxing Lu**, Xuemin (Sherman) Shen, and Yousuke Nozaki, "Towards secure targeted broadcast in smart grid", *IEEE Communications Magazine*, to appear.

[J11] Xu Li, **Rongxing Lu**, Xiaohui Liang, Xuemin (Sherman) Shen, Jiming Chen, and Xiaodong Lin, "Smart Community: an Internet of Things Application", *IEEE Communications Magazine*, Vol. 49, No. 11, pp. 68 - 75, 2011.

[J12] M. Fouda, Z. Md. Fadlullah, Nei Kato, **Rongxing Lu**, and Xuemin (Sherman) Shen, "A Light-weight Message Authentication Scheme for Smart Grid Communications", *IEEE Transactions on Smart Grid*, Vol. 2, No. 4, pp. 675 - 685, 2011.

[J13] Albert Wasef, **Rongxing Lu**, Xiaodong Lin, and Xuemin (Sherman) Shen, "Complementing Public Key Infrastructure to Secure Vehicular Ad Hoc Networks", *IEEE Wireless Communications*, Vol. 17, No. 5, pp. 22 - 28, 2010.

[J14] Yipin Sun, **Rongxing Lu**, Xiaodong Lin, Xuemin (Sherman) Shen, and Jinshu Su, "An Efficient Pseudonymous Authentication Scheme with Strong Privacy Preservation for Vehicular Communications", *IEEE Transactions on Vehicular Technology*, Vol. 59, No. 7, pp. 3589-3603, 2010.

[J15] Xiaodong Lin, **Rongxing Lu**, Xuemin (Sherman) Shen, Yoshiaki Nemoto, and Nei Kato, "SAGE: A Strong Privacy-Preserving Scheme Against Global Eavesdropping for eHealth Systems", *IEEE Journal on Selected Areas in Communications*, Vol. 27, No. 4, pp. 365 - 378, 2009.

[J16] Haojin Zhu, Xiaodong Lin, **Rongxing Lu**, Yanfei Fan, and Xuemin (Sherman) Shen, "SMART: A Secure Multi-Layer Credit based Incentive Scheme for Delay-Tolerant Networks", *IEEE Transactions on Vehicular Technology*, Vol. 58, Issue 8, pp. 4628 - 4639, 2009.

[J17] Haojin Zhu, **Rongxing Lu**, Xiaodong Lin, and Xuemin (Sherman) Shen, "Security in Service-Oriented Vehicular Networks", *IEEE Wireless Communications*, Vol. 16, No. 4, pp. 16-22, 2009.

A.2 Conference Papers

[C1] **Rongxing Lu**, Xiaodong Lin, Tom Luan, Xiaohui Liang, Xu Li, Le Chen, and Xuemin (Sherman) Shen, "PReFilter: An Efficient Privacy-preserving Relay Filtering Scheme for Delay Tolerant Networks", Proc. IEEE INFOCOM 2012, Orlando, Florida, USA, March 25-30, 2012. (Acceptance rate 278/1547 = 17.97%)

[C2] Xiaohui Liang, Xu Li, Qinghua Shen, **Rongxing Lu**, Xiaodong Lin, Xuemin (Sherman) Shen, and Weihua Zhuang, "Exploiting Prediction to Enable Secure and Reliable Routing in Wireless Body Area Networks", Proc. IEEE INFOCOM 2012, Orlando, Florida, USA, March 25-30, 2012. (Acceptance rate 278/1547 = 17.97%)

[C3] Xiaohui Liang, Xu Li, **Rongxing Lu**, Xiaodong Lin, and Xuemin (Sherman) Shen, "A Secure and Efficient Service Review System for Service-Oriented Mobile Social Networks", Proc. ICDCS'12, Macau, China, June 18-21, 2012. (acceptance rate is 13%)

[C4] Xiaodong Lin, **Rongxing Lu**, Xiaohui Liang, and Xuemin (Sherman) Shen, "STAP: A Social-Tier-Assisted Packet Forwarding Protocol for Achieving Receiver-Location Privacy Preservation in VANETs", Proc. IEEE INFOCOM 2011, Shanghai, China, April 10 -15, 2011. (Acceptance rate 291/1823 = 15.96%)

[C5] **Rongxing Lu**, Xiaodong Lin, Tom H. Luan, Xiaohui Liang, and Xuemin (Sherman) Shen, "Anonymity Analysis on Social Spot based Pseudonym Changing for Location Privacy in VANETs", Proc. IEEE ICC'11, Kyoto, Japan, June 5 - 9, 2011.

[C6] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "Sacrificing the Plum Tree for the Peach Tree: A Socialspot Tactic for Protecting Receiver-location Privacy in VANET", Proc. IEEE Globecom 2010, Miami, Florida, USA, December 6-10, 2010.

[C7] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "FLIP: An Efficient Privacy-preserving Protocol for Finding Like-minded Vehicles on the Road", Proc. IEEE Globecom 2010, Miami, Florida, USA, December 6-10, 2010.

[C8] Xu Li, Xiaohui Liang, **Rongxing Lu**, Shibo He, Jiming Chen, and Xuemin (Sherman) Shen, "Toward Reliable Actor Service in Wireless Sensor and Actor Networks", Proc. IEEE MASS'11, Valencia, Spain, October 17-22, 2011. (Acceptance rate 20%)

[C9] Xiaodong Lin, **Rongxing Lu**, Kevin Foxton, and Xuemin (Sherman) Shen, "An Efficient Searchable Encryption Scheme and Its Application in Network Forensics", *The Third International Conference on Forensic Applications and Techniques in Telecommunications, Information and Multimedia* (e-Forensics 2010), Shanghai, China, November 11-12, 2010.

[C10] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "Secure Handshake with Symptoms-matching: The Essential to the Success of mHealthcare Social Network", *The Fifth International Conference on Body Area Networks* (BodyNets 2010), Corfu Island, Greece, September 10-12, 2010. (Best Paper Award)

[C11] **Rongxing Lu**, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing", *The 5th ACM Symposium on Information, Computer and Communications Security* (ASIACCS 2010), Beijing, China, April 13-16, 2010. (Acceptance rate 25/166 = 15.06%)

[C12] **Rongxing Lu**, Xiaodong Lin, and Xuemin (Sherman) Shen, "SPRING: A Social-based Privacy-preserving Packet Forwarding Protocol for Vehicular Delay Tolerant Networks", *The 29th IEEE International Conference on Computer Communications* (INFOCOM 2010), San Diego, California, USA, March 14 - 19, 2010. (Acceptance rate 276/1575 = 17.52%)

[C13] Haojin Zhu, Xiaodong Lin, **Rongxing Lu**, Xuemin (Sherman) Shen, Dongsheng Xing, and Zhenfu Cao, "An Opportunistic Batch Bundle Authentication Scheme for Energy Constrained DTNs", , *The 29th IEEE International Conference on Computer*

Communications (INFOCOM 2010), San Diego, California, USA, March 14 - 19, 2010. (Acceptance rate 276/1575 = 17.52%)

[C14] **Rongxing Lu**, Xiaodong Lin, Haojin Zhu, and Xuemin (Sherman) Shen, "SPARK: A New VANET-based Smart Parking Scheme for Large Parking Lots", *The 28th IEEE International Conference on Computer Communications* (INFOCOM 2009), Rio de Janeiro, Brazil, April 19-25, 2009. (Acceptance rate 282/1435 = 19.65%)

[C15] Xiaodong Lin, **Rongxing Lu**, and Xuemin (Sherman) Shen, "Location-release Signature for Vehicular Communications", *The 18th International Conference on Computer Communications and Networks* (ICCCN 2009), San Francisco, CA, USA, Aug. 2-6, 2009. (Best Paper Award)

[C16] Haojin Zhu, Xiaodong Lin, **Rongxing Lu**, and Xuemin (Sherman) Shen, "A Secure Credit Based Incentive Scheme for Delay Tolerant Networks", *The Third International Conference on Communications and Networking in China* (Chinacom 2009), Hangzhou, China, Aug. 25-27, 2008. (Best Paper Award)

References

[1] M. Raya and J. Hubaux, "Security aspects of inter-vehicle communications," in *Swiss Transport Research Conference (STRC)*, 2005.

[2] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing vehicular communications," *IEEE Communications Magazine*, vol. 44, no. 10, pp. 8–15, 2006.

[3] Y. Toor, P. Mühlethaler, A. Laouiti, and A. de La Fortelle, "Vehicle ad hoc networks: Applications and related technical issues," *IEEE Communications Surveys and Tutorials*, vol. 10, no. 1-4, pp. 74–88, 2008.

[4] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1557–1568, 2007.

[5] N. Ristanovic, P. Papadimitratos, G. Theodorakopoulos, J.-P. Hubaux, and J.-Y. L. Boudec, "Adaptive message authentication for vehicular networks," in *Vehicular Ad Hoc Networks*, 2009, pp. 121–122.

[6] H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, June 2008.

[7] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security, Special Issue on Security of Ad Hoc and Sesnor Networks*, vol. 15, no. 1, pp. 39–68, 2007.

[8] P. Papadimitratos and J.-P. Hubaux, "Secure vehicular communication systems," in *Encyclopedia of Cryptography and Security (2nd Ed.)*, 2011, pp. 1140–1143.

[9] T. Moore, M. Raya, J. Clulow, P. Papadimitratos, R. Anderson, and J.-P. Hubaux, "Fast exclusion of errant devices from vehicular networks," in *SECON*, 2008, pp. 135–143.

[10] C. Zhang, X. Lin, R. Lu, and P.-H. Ho, "RAISE: An efficient rsu-aided message authentication scheme in vehicular communication networks," in *Proc. IEEE ICC'08*, Beijing, China, May 19-23 2008.

[11] A. Wasef, Y. Jiang, and X. Shen, "DCS: An efficient distributed certificate service scheme for vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 2, pp. 533–549, 2010.

[12] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: A secure and privacy-preserving protocol for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3442–3456, 2007.

[13] A. Wasef and X. Shen, "Efficient group signature scheme supporting batch verification for securing vehicular networks," in *ICC*, 2010, pp. 1–5.

[14] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications," in *The 27th Conference on Computer Communications (INFOCOM 2008)*, Phoenix, Arizona, USA, April 2008, pp. 1229–1237.

[15] H. Zhu, R. Lu, X. Lin, and X. Shen, "Security in service-oriented vehicular networks," *IEEE Wireless Communications*, vol. 16, no. 4, pp. 16–22, 2009.

[16] K. Hoeper and G. Gong, "Preventing or utilising key escrow in identity-based schemes employed in mobile ad hoc networks," *IJSN*, vol. 2, no. 3/4, pp. 239–250, 2007.

[17] X. Fan and G. Gong, "Key revocation based on dirichlet multinomial model for mobile ad hoc networks," in *LCN*, 2008, pp. 958–965.

[18] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *IEEE Vehicular Technology Magazine*, vol. 2, no. 2, pp. 12–22, 2007.

[19] X. Lin, R. Lu, C. Zhang, H. Zhu, P.-H. Ho, and X. Shen, "Security in vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 88–95, 2008.

[20] C. D. Wang and J. P. Thompson, "Apparatus and method for motion detection and tracking of objects in a region for collision avoidance utilizing a real-time adaptive probabilistic neural network," U.S. patent no. 5,613,039, 1997.

[21] "U.S. department of transportation," http://safety.fhwa.dot.gov/facts/road fact-sheet.htm.

[22] "European project prevent-intersafe," http://www.preventip.org/en/prevent_subprojects/intersection_safety /intersafe.

[23] "European project react," www.react-project.org.

[24] R. Lu, X. Lin, and X. Shen, "SPRING: A social-based privacy-preserving packet forwarding protocol for vehicular delay tolerant networks," in *INFOCOM'10*, San Diego, California, USA, March 2010, pp. 1229–1237.

[25] C. Laurendeau and M. Barbeau, "Threats to security in dsrc/wave," in *Ad-Hoc, Mobile, and Wireless Networks*, 2006, pp. 266–279.

[26] B. Parno and A. Perrig, "Challlenges in securing vehicular networks," in *Proc. HotNets-IV, 2005*, 2005.

[27] X. Lin, X. Sun, X. Wang, C. Zhang, P.-H. Ho, and X. Shen, "TSVC: timed efficient and secure vehicular communications with privacy preserving," *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 4987–4998, December 2008.

[28] C. Zhang, X. Lin, R. Lu, P.-H. Ho, and X. Shen, "An efficient message authentication scheme for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3357–3368, 2008.

[29] R. Brooks, S. Sander, J. Deng, and J. Taiber, "Automobile security concerns," *IEEE Vehicular Technology Magazine*, vol. 4, no. 2, pp. 52–64, June 2009.

[30] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure vehicular communication systems: design and architecture," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 100–109, November 2008.

[31] F. Kargl, P. Papadimitratos, L. Buttyan, M. Muter, E. Schoch, B. Wiedersheim, T.-V. Thong, G. Calandriello, A. Held, A. Kung, and J.-P. Hubaux, "Secure vehicular communication systems: implementation, performance, and research challenges," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 110–118, November 2008.

[32] "Draft amendment to standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications-amendment 7: Wireless access in vehicular environment," IEEE P802.11p/D3.0, Tech. Rep., 2007.

[33] R. A. Uzcategui, A. J. D. Sucre, and G. Acosta-Marum, "WAVE: a tutorial," *IEEE Communications Magazine*, vol. 47, no. 5, pp. 126–133, 2009.

[34] "IEEE 1609 - family of standards for wireless access in vehicular environments (wave)," Website, http://www.standards.its.dot.gov/fact_sheet.asp?f=80.

[35] "IEEE standard 1609.2 - ieee trial-use standard for wireless access in vehicular environments - security services for applications and management messages," July 2006.

[36] J. P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security and Privacy Magazine*, vol. 2, no. 3, pp. 49–55, 2004.

[37] M. Raya and J. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks (SASN'05)*, Nov. 2005.

[38] X. Sun, X. Lin, and P.-H. Ho, "Secure vehicular communications based on group signature and id-based signature scheme," in *Proceedings of IEEE International Conference on Communications (ICC'07)*, 2007, pp. 1539–1545.

[39] G. Calandriello, P. Papadimitratos, A. Lioy, and J.-P. Hubaux, "Efficient and robust pseudonymous authentication in vanet," in *Proceedings of the International workshop on Vehicular ad hoc networks (VANET'07)*, 2007, pp. 19–28.

[40] J. Freudiger, M. Raya, M. Feleghhazi, P. Papadimitratos, and J.-P. Hubaux, "Mix zones for location privacy in vehicular networks," in *Proceedings of WiN-ITS 2007*, Vancouver, British Columbia, August 2007.

[41] J. Breslin and S. Decker, "The future of social networks on the internet: The need for semantics," *IEEE Internet Computing*, vol. 11, no. 6, pp. 86–90, 2007.

[42] S. Staab, P. Domingos, P. Mika, J. Golbeck, L. Ding, T. Finin, A. Joshi, A. Nowak, and R. R. Vallacher, "Social networks applied," *IEEE Intelligent Systems*, vol. 20, no. 1, pp. 80–93, 2005.

[43] A.-K. Pietiläinen, E. Oliver, J. LeBrun, G. Varghese, and C. Diot, "Mobiclique: middleware for mobile social networking," in *WOSN '09: Proceedings of the 2nd ACM workshop on Online social networks*. New York, NY, USA: ACM, 2009, pp. 49–54.

[44] F. Ekman, "Mobility models for mobile ad hoc network simulations," Mater's Thesis, Helsinki University of Technology, May 2008.

[45] R. Lu, X. Lin, X. Liang, and X. Shen, "Sacrificing the plum tree for the peach tree: A socialspot tactic for protecting receiver-location privacy in vanet," in *GLOBECOM*, 2010, pp. 1–5.

[46] R. Lu, X. Lin, T. H. Luan, X. Liang, and X. Shen, "Pseudonym changing at social spots: An effective strategy for location privacy in vanets," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 86–96, January 2012.

[47] R. Lu, X. Lin, X. Liang, and X. Shen, "Flip: An efficient privacy-preserving protocol for finding like-minded vehicles on the road," in *GLOBECOM*, 2010, pp. 1–5.

[48] R. Lu, X. Lin, H. Zhu, X. Shen, and B. R. Preiss, "Pi: a practical incentive protocol for delay tolerant networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 4, pp. 1483–1493, 2010.

[49] D. A. Batallas and A. A. Yassine, "Information leaders in product development organizational networks: Social network analysis of the design structure matrix," *IEEE Transactions on Engineering Management*, vol. 53, no. 4, pp. 570–582, 2006.

[50] T. Snijders and S. Borgatti, "Non-parametric standard errors and tests for network statistics," *Connections*, vol. 22, no. 2, pp. 161–170, 1999.

[51] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Lecture Notes in Computer Science, Advances in Cryptology - CRYPTO 2001*, vol. 2139. Springer-Verlag, 2001, pp. 213–229.

[52] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[53] X. Boyen and B. Waters, "Full-domain subgroup hiding and constant-size group signatures," in *Lecture Notes in Computer Science, PKC 2007*, 2007, pp. 1–15.

[54] X. Liang, Z. Cao, J. Shao, and H. Lin, "Short group signature without random oracles," in *ICICS*, 2007, pp. 69–82.

[55] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proceedings of Crypto 2004, LNCS 3152*, 2004, pp. 41–55.

[56] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for fr-reduction," *IEICE Transactions on Fundamentals*, vol. E84-A, no. 5, pp. 1234–1243, 2001.

[57] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure provenance: the essential of bread and butter of data forensics in cloud computing," in *ASIACCS*, 2010, pp. 282–292.

[58] V. Shoup, "Oaep reconsidered," *Journal of Cryptology*, vol. 15, no. 4, pp. 223–249, 2002.

[59] D. Boneh and X. Boyen, "Short signatures without random oracles and the sdh assumption in bilinear groups," *Journal of Cryptology*, vol. 21, no. 2, pp. 149–177, 2008.

[60] K. Fall, "A delay tolerant networking architecture for challenged internet," in *Proceedings of SIGCOMM '03*, Karlsruhe, Germany, 2003, pp. 27–34.

[61] R. Lu, X. Lin, H. Zhu, and X. Shen, "SPARK: a new vanet-based smart parking scheme for large parking lots," in *The 28th Conference on Computer Communications (INFOCOM 2009)*, Rio de Janeiro, Brazil, April 2009.

[62] K. Zeng, "Pseudonymous pki for ubiquitous computing," in *Proceedings of EuroPKI' 06*, Turin, Italy, June 2006, pp. 207–222.

[63] N. Banerjee, M. D. Corner, D. Towsley, and B. N. Levine, "Relays, base stations, and meshes: enhancing mobile networks with infrastructure," in *Proceedings of MobiCom 2008*, San Francisco, USA, 2008, pp. 81–91.

[64] F. Li, J. Wu, and A. Srinivasan, "Thwarting blackhole attacks in disruption-tolerant networks using encounter tickets," in *Proceedings of INFOCOM 2009*, Rio de Janeiro, Brazil, April 2009.

[65] P. Golle, M. Jakobsson, A. Juels, and P. Syverson, "Universal re-encryption for mixnets," in *Proceedings of CT-RSA 2004*, 2004, pp. 163–178.

[66] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161–174, 1991.

[67] P. Shankar, T. Nadeem, J. Rosca, and L. Iftode, "Cars: Context-aware rate selection for vehicular networks," in *Proceedings of ICNP 2008*, 2008, pp. 1–12.

[68] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *Proceedings of MobiHoc 2008*, 2008, pp. 241–250.

[69] A. Wasef and X. Shen, "REP: Location privacy for vanets using random encryption periods," *ACM Mobile Networks and Applications (MONET)*, vol. 15, no. 1, pp. 172–185, 2010.

[70] Z. Chai, Z. Cao, and R. Lu, "Efficient password-based authentication and key exchange scheme preserving user privacy," in *WASA*, 2006, pp. 467–477.

[71] M. Izabachene and D. Pointcheval, "New anonymity notions for identity-based encryption," in *SCN '08*, ser. LNCS 5229, 2008, pp. 375–391.

[72] Y. Jian, S. Chen, Z. Zhang, and L. Zhang, "Protecting receiver-location privacy in wireless sensor networks," in *INFOCOM 2007*, Anchorage, AK, May 2007, pp. 1955 – 1963.

[73] R. Cheng, D. K. Y. Yau, and J. Fu, "Packet cloaking: Protecting receiver privacy against traffic analysis," in *3rd IEEE Workshop on Secure Network Protocols*, 2007, pp. 1–6.

[74] L. Buttyan, T. Holczer, and I. Vajda, "On the effectiveness of changing pseudonyms to provide location privacy in vanets," in *ESAS 2007*, ser. Lecture Notes In Computer Science, vol. 4572. Springer-Verlag, 2007, pp. 129–141.

[75] A. Beresford and F. Stajano, "Mix zones: user privacy in location-aware services," in *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, March 2004, pp. 127 – 131.

[76] L. Huang, H. Yamane, K. Matsuura, and K. Sezaki, "Towards modeling wireless location privacy," in *PET 2005*, ser. Lecture Notes In Computer Science, vol. 3856. Springer-Verlag, 2005, pp. 59–77.

[77] ——, "Silent cascade: Enhancing location privacy without communication qos degradation," in *SPC 2006*, ser. Lecture Notes In Computer Science, vol. 3934. Springer-Verlag, 2006, pp. 165–180.

[78] M. Li, K. Sampigethaya, L. Huang, and R. Poovendran, "Swing & swap: user-centric approaches towards maximizing location privacy," in *WPES*, 2006, pp. 19–28.

[79] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "CARAVAN: Providing location privacy for vanet," in *Proceedings of Embedded Securit in Cars (ESCAR)*, 2005.

[80] J. Freudiger, M. Raya, and M. Feleghhazi, "Mix zones for location privacy in vehicular networks," in *Proceedings of WiN-ITS 2007*, Vancouver, British Columbia, August 2007.

[81] A. Pfitzmann and M. Köhntopp, "Anonymity, unobservability, and pseudonymity - a proposal for terminology," in *Workshop on Design Issues in Anonymity and Unobservability*, ser. Lecture Notes In Computer Science, vol. 2009. Springer-Verlag, 2000, pp. 1–9.

[82] P. Papadimitratos, A. Kung, J. P. Hubaux, and F. Kargl, "Privacy and identity management for vehicular communication systsms: a position paper," in *Proceedings of Workshop on Standards for Privacy in User-Centric Identity Management*, 2006.

[83] W. Mao, *Modern Cryptography: Theory and Practice*. Prentice Hall PTR, 2003.

[84] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in vanet," in *Proceedings of VANET' 07*, Montreal, Quebec, Canada., September 2007, pp. 19–28.

[85] L. Kleinrock, *Queueing Systems Vol. 1: Theory*. Wiley, 1975.

[86] S.-M. Cheng, W.-R. Lai, P. Lin, and K.-C. Chen, "Key management for umts mbms," *IEEE Transactions on Wireless Communications*, vol. 7, no. 9, pp. 3619–3628, 2008.

[87] M. Gerlach, "Assessing and improving privacy in vanets," in *Proceedings of Fourth Workshop on Embedded Security in Cars (ESCAR)*, November 2006.

[88] J. Freudiger, R. Shokri, and J.-P. Hubaux, "On the optimal placement of mix zones," in *Privacy Enhancing Technologies*, 2009, pp. 216–234.

[89] Z. Ma, F. Kargl, and M. Weber, "Measuring long-term location privacy in vehicular communication systems," *Computer Communications*, vol. 33, no. 12, pp. 1414–1427, 2010.

[90] J. Freudiger, M. H. Manshaei, J.-P. Hubaux, and D. C. Parkes, "On non-cooperative location privacy: a game-theoretic analysis," in *ACM Conference on Computer and Communications Security*, 2009, pp. 324–337.

[91] S. Smaldone, L. Han, P. Shankar, and L. Iftode, "Roadspeak: enabling voice chat on roadways using vehicular social networks," in *SocialNets '08*, 2008, pp. 43–48.

[92] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *EUROCRYPT*, 2000, pp. 139–155.

[93] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. of EUROCRYPT'99*, 1999, pp. 223–238.

[94] K.-H. Huang, Y.-F. Chung, C.-H. Liu, F. Lai, and T.-S. Chen, "Efficient migration for mobile computing in distributed networks," *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 40–47, 2009.

[95] M. Atallah and W. Du, "Secure multi-party computation problems and their applications: a review and open problems," in *NSPW*, 2001, pp. 13–22.

[96] G. Zhong, I. Goldberg, and U. Hengartner, "Louis, lester and pierre: Three protocols for location privacy," in *PET 2007*, ser. LNCS 4776, 2007, pp. 62–76.

[97] S. Chatterjee, K. Karabina, and A. Menezes, "A new protocol for the nearby friend problem," in *Cryptography and Coding*, ser. LNCS 5921, 2009, pp. 236–251.

[98] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 77–90, 2008.

[99] Delay tolerant networking research group, http://www.dtnrg.org, November 2008.

[100] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: the single-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 63–76, 2008.

[101] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "MaxProp: Routing for vehicle-based disruption-tolerant networks," in *The 25th Conference on Computer Communications (INFOCOM 2006)*, Barcelona, Spain, April 2006.

[102] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on the design of opportunistic forwarding algorithms," in *The 25th Conference on Computer Communications (INFOCOM 2006)*, Barcelona, Spain, April 2006.

[103] S. Upendra, H. H. Song, L. Qiu, and Y. Zhang, "Incentive-aware routing in dtns," in *Prof. of IEEE ICNP 2008*, Orlando, FL, USA, 2008, pp. 238–247.

[104] H. Zhu, X. Lin, R. Lu, Y. Fan, and X. Shen, "SMART: a secure multi-layer credit based incentive scheme for delay-tolerant networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4628–4639, 2009.

[105] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. of The Sixth International Conference on Mobile Computing and Networking Networking (MobiCom 2000)*, Boston, MA, Aug. 2000, pp. 255–265.

[106] S. Buchegger and J.-Y. L. Boudec, "Performance analysis of the confidant protocol: Cooperation of nodes - fairness in dynamic ad-hoc networks," in *Proc. of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHoc 2002)*, Lausanne, Switzerland, Jun. 2002, pp. 226–236.

[107] Y. Liu and Y. R. Yang, "Reputation propagation and agreement in mobile ad-hoc networks," in *Proc. of IEEE WCNC 2003*, vol. 3, New Orleans, LA, March 2003, pp. 1510–1515.

[108] L. Buttyan and J.-P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *Proc. of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHoc 2000)*, Boston, MA, Aug. 2000, pp. 87–96.

[109] J.-P. Hubaux, T. Gross, J.-Y. L. Boudec, and M. Vetterli, "Toward self-organized mobile ad hoc networks: the terminodes project," *IEEE Communications Magazine*, vol. 31, no. 1, pp. 118–124, Jan. 2001.

[110] S. Zhong, J. Chen, and Y. Yang, "Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *The 22th Conference on Computer Communications (INFOCOM 2003)*, vol. 3, Barcelona, Spain, Mar.-Apr. 2003, pp. 1987–1997.

[111] F. Zhang, R. Safavi-Nani, and W. Susilo, "An efficient signature scheme from bilinear pairings and its applications," in *Lecture Notes in Computer Science, Proc. PKC 2004*, vol. 2947. Springer-Verlag, 2004, pp. 277–290.

[112] ——, "Efficient verifiably encrypted signature and partially blind signature from bilinear pairings," in *Lecture Notes in Computer Science, Proc. INDOCRYPT 2003*, vol. 2904. Springer-Verlag, 2003, pp. 191–204.

[113] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, and efficient VANET authentication," in *Proceedings of the 6th Embedded Security in Cars Conference (ESCAR 08)*, 2008.

[114] D. Boneh, H. Shacham, and B. Lynn, "Short signatures from the weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[115] Y. Zhang, W. Lou, W. Liu, and Y. Fang, "A secure incentive protocol for mobile ad hoc networks," *Wireless Networks*, vol. 13, no. 5, pp. 569–582, 2007.

[116] P. Kar, S. Sen, and P. Dutta, "Effect of individual opinions on group interactions," *Connection Science*, vol. 14, no. 4, pp. 335–344, 2002.