SECURITY AND PRIVACY IN VEHICULAR AD-HOC NETWORKS

LI KANG
PhD

The Hong Kong Polytechnic University

2021

The Hong Kong Polytechnic University
Department of Computing

Security and Privacy in Vehicular Ad-Hoc Networks

Li Kang

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

August 2020

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____(Signed)

_____Li Kang_____(Name of student)

Abstract

The automotive and transportation industry is currently developing smart vehicles to improve driving safety and build an intelligent transportation system in modern cities. As a very promising technology to achieve these goals, the vehicular ad hoc networks (VANETs) are proposed to improve safety on roads by providing real-time traffic information for vehicles. However, VANETs cause many information security and privacy issues to the transportation system, such as bogus information attacks, message modification attacks, or eavesdropping attacks that may undermine user safety and privacy. For example, an attacker could broadcast bogus messages to mislead nearby vehicle to take wrong actions, which could cause a traffic accident. Hence, VANETs can only be deployed successfully after resolving the security and privacy issues.

Message authentication is the most important mechanism to ensure information security in VANETs. Typically, message authentication is realized using a digital signature scheme, by which a vehicle produces a signature on traffic-related messages and broadcasts the pairs of messages and signatures over the network. The message recipient verifies the validity of the signature to ensure that the message is signed by a legitimate vehicle and has not been altered during the transmission. Various kinds of digital signature schemes are proposed to realize secure message authentication in VANETs. However, they all have different benefits and drawbacks in terms of security and efficiency. In order to improve the security and efficiency of the message authentication for VANETs, a secure online/offline certificateless signature scheme is proposed in this thesis. The proposed authentication scheme based on certificateless signature not only satisfies the basic security and privacy requirements but also has a better efficiency in terms of signature generation and verification. Furthermore, it supports the techniques of signature aggregation and batch verification, which can improve the efficiency

of message authentication.

For an authentication scheme to be useful in practice, a secure and efficient revocation mechanism, which revokes malicious or compromised users in the network, is necessary. However, most of the certificateless signature-based authentication schemes lack a secure and efficient revocation mechanism. Hence, based on the proposed online/offline certificateless signature, a revocable online/offline certificateless signature is proposed to solve the revocation problem. Compared with conventional revocation approaches in many other authentication schemes for VANETs, the proposed revocable mechanism eliminates the delay caused by checking against the revocation list and does not require a secret channel. Hence, the proposed revocable approach is practical to be used in the scenario of VANETs. Besides, the revocation burden is alleviated by employing the well-known KUNodes algorithm. Moreover, in order to enhance the overall authentication efficiency in VANETs, a process where the roadside units assist the signature verification of nearby vehicles using cuckoo filter is developed.

Even though the proposed authentication scheme prevents many potential security and privacy attacks, certain privacy issues still exist. For example, an adversary could collect transmitted messages and employ a data analysis technique to extract some sensitive information, such as the home address, driving preference, etc. Hence, unlinkability and minimum information disclosure are two desirable features that are required to ensure strong privacy protection. As a promising approach to provide strong privacy for the drivers in VANETs, anonymous credential and its necessary component range proof are investigated in this thesis. Range proof is a cryptographic protocol that has many applications in VANETs, such as the anonymous credentials used in the vehicle registration process, and applications in parking navigation services. In order to develop practical range proof protocol that is secure and efficient, we specifically study the range proof protocol used in cryptocurrency Monero and identify its security flaws. Then, we develop an improved range proof protocol for Monero and give a rigorous proof to prove its security.

Publications Arising from the Thesis

[1] Kang Li, Rupeng Yang, Man Ho Au, and Qiuliang Xu. "Practical range proof for cryptocurrency Monero with provable security." In International Conference on Information and Communications Security, pp. 255-262. Springer, Cham, 2017.

[2] Kang Li, Man Ho Au, Wang Hei Ho, and Yilei Wang. "An efficient conditional privacy-preserving authentication scheme for vehicular ad hoc networks using online/offline certificateless aggregate signature." In International Conference on Provable Security, pp. 59-76. Springer, Cham, 2019.

[3] Kang Li, Wang Fat Lau, and Man Ho Au. "A secure and efficient privacy-preserving authentication scheme for vehicular networks with batch verification using cuckoo filter." In International Conference on Network and System Security, pp. 615-631. Springer, Cham, 2019.

[4] Peng Li, Kang Li, Yilei Wang, Ying Zheng, Dongfeng Wang, Guoyu Yang, and Xiaomei Yu. "A systematic mapping study for blockchain based on complex network." Concurrency and Computation: Practice and Experience (2020): e5712.

[5] Kang Li, Wang Fat Lau, Man Ho Au, Ivan Wang-Hei Ho, and Yilei Wang. "Efficient message authentication with revocation transparency using blockchain for vehicular networks." Computers & Electrical Engineering 86 (2020): 106721.

Acknowledgements

First and foremost, I would like to express my sincere and deepest gratitude to Dr Man Ho Allen Au for his continuous support and patient guidance during my PhD study. His encouragements and guidance help me overcome many difficulties. I will never be able to complete my dissertation without his guidance and help. I would also like to thank Dr Daniel Xiapu Luo, Dr Ivan Wang-Hei Ho and Dr Yilei Wang for their help.

Also, I would like to thank my colleagues and friends that I met in PolyU.

Lastly, thank my mother, father and brother for their continuous support over these years.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CA Certificate Authority

CDH Computational Diffie-Hellman

CRL Certificate Revocation List

DL Discrete Logarithm

DoS Denial-of-Service

DSRC Dedication Short Range Communications

HMAC Hash Message Authentication Code

ID-PKC Identity-based Public Key Cryptography

KGC Key Generation Center

MAC Message Authentication Code

Mod Modulo Operation

NIZKPoK Non-interactive Zero-knowledge Proof of Knowledge

OBU Onboard Unit

PKC Public Key Cryptography

PKI Public Key Infrastructure

PPT Probabilistic Polynomial Time

RSU Roadside Unit

TA Trusted Authority

TRA Trace Authority

V2I Vehicle-to-Infrastructure

V2V Vehicle-to-Vehicle

V2X Vehicle-to-Everything

VANETs Vehicular Ad Hoc Networks

ZKP Zero-knowledge Proof

Chapter 1

Introduction

1.1 Background and Motivation

Nowadays, vehicles are used by many people daily and have become an indispensable component of our life. However, the increasing number of vehicles causes a lot of traffic accidents on the road, which could lead to great loss of our life. For example, according to road traffic accident statistics from the transport department of the Hong Kong government, there were around 16000 traffic accidents happened every year from 2015-2018. And according to the CARE-European Road Accident Database Report, around 43,000 deaths and 1.8 million injuries occurred at regular intervals [1]. Research [2, 3] show that about 60 percent of the traffic accidents could be avoided if the driver is aware of the warning message just a few seconds before the accidents. Hence, developing technologies to improve driving safety and efficiency is very meaningful. In the past several years, VANETs have aroused significant interest in both the industry and academia and has been proposed to be a very promising technology to achieve road safety and high traffic efficiency. Three major factors have led to the development of VANETs. The first one is the wide adoption of IEEE 802.11p standard and a new technology named as Dedicated Short-Range Communications (DSRC) protocol is designed. DSRC is specifically designed to facilitate the communications of VANETs and it can provide high data transfer rates of up to 27 Mb/s over a range of 1 km while maintaining low overhead in the spectrum, allowing efficient emergency communications between vehicles [4]. This leads to efficient emergency communications

in highly dynamic vehicle networks. The second factor is that car manufactures realize the great potential of using information technology to improve the driving safety and start to cooperate with the telecommunications companies to develop novel approaches to address the safety issues of vehicles. The last factor is the commitment of large countries and regional governments to allocate wireless spectrum for vehicular wireless communication [5].

VANETs enable vehicles to exchange real-time information with each other and facilitates rich applications to enhance road safety and traffic efficiency. However, such a promising technology imposes information security and privacy issue to the transportation system, such as message modification attack, denial-of-service attack, replay attack or some other attacks that may cause serious damage to the drivers or undermine location privacy or even leak the personal information of the driver. For example, a malicious vehicle could send fake messages to cause a traffic jam or an accident. An attacker could collect the messages sent by a vehicle and extract personal information, such as home address and driving preference, from the collected messages. Any such attack could lead to an accident which may cause great loss. Hence, information security and privacy issues should be addressed before deploying VANETs for practical applications. And this thesis is motivated to address the information security and privacy issues of VANETs.

1.2 Overview of Vehicular Ad Hoc Networks

1.2.1 Component and Architecture of VANETs

The VANET is a type of wireless network and is proposed to enhance the driving safety and efficiency by facilitating real-time information exchange among vehicles. Typically, it is made up of three main components, which are the trusted authorities (TAs), such as the key generation center (KGC), the onboard unit (OBU), and the roadside unit (RSU). The TAs are in charge of the registration and management of the network users. For example, the TAs need to authenticate the identities of network users and has the ability to reveal the identities of any users. Moreover, the TAs has high computation power and large storage size to handle a large number of users. The OBU is a communication device installed on every vehicle and is used to receive and send messages for the vehicle. OBU con-

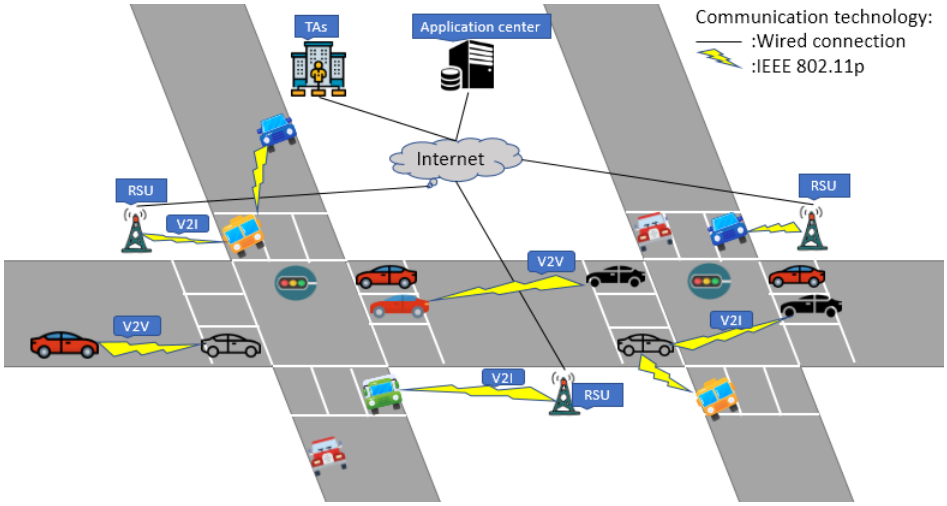


Figure 1.1: Vehicular Ad Hoc Networks

sists of many electronic components such as resource command processor (RCP), sensor devices, the user interface and read/write storage for retrieving storage information [6]. The RSU is located along the roadside or at the critical points of the roads to assist the message dissemination in the network. Both OBUs and RSUs are equipped with a network device that is responsible for wireless communication using IEEE 802.11p radio technology. In VANETs, Vehicle-to-Everything (V2X) communication, Vehicle-to-Vehicle (V2V) communication and the Vehicle-to-Infrastructure (V2I) communication, are realized. V2V allows vehicles to send traffic-related information to each other. V2I allows a vehicle to communicate with a roadside infrastructure (RSU) mainly for information and data gathering applications. Typically, the computation power and storage capacity of RSU is much higher than that of OBU. And, V2I has longer communication range, thus vehicles connect to RSU can sent information to a longer range.

A two-layer network model is suitable for vehicular networks, as presented in prior research work [7]. The upper layer is composed of TAs, application centers and RSUs. The lower layer consists of OBUs and RSUs. The communication of the upper layer network is realized using secure wired connections, whereas the communication of the lower layer network is realized using wireless technology, specifically the DSRC radio technology. The typical architecture of VANETs is shown in Figure 1.1.

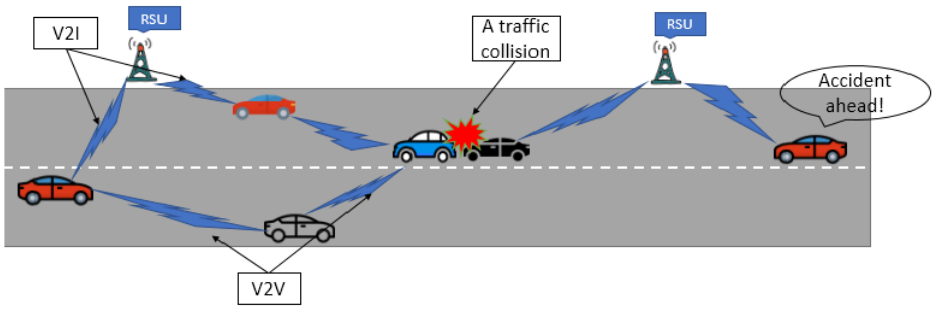


Figure 1.2: A typical use case of VANETs

1.2.2 Characteristics and Applications of VANETs

The VANET belongs to mobile ad hoc network (MANET) and most of the network nodes are vehicles that are equipped with an OBU. Hence, it has some unique characteristics. Firstly, in VANETs, the moving direction of a vehicle is constrained by the road layout, which means mobility of vehicles is predictable. Secondly, network topology changes very rapidly, as the vehicles moves at high speed towards different directions on the road. Thirdly, the network density is variable, such as the roads become very busy during the rush hours. Another characteristic is that due to wireless connections and high moving speed towards different directions of vehicles, frequent network disconnections exist. Moreover, the transmission power is limited, which leads to limited wireless transmission distance [8]. Lastly, the computing capacity of VANET is limited. For example, the computing capacity of an OBU is limited.

The communication of VANETs facilitates various kinds of applications for drivers and the public. These applications can be categorized to be safety-oriented applications and commercial/convenience applications. The main propose of developing VANETs is to provide safety applications for drivers to enhance road safety. The V2V and V2I communications in VANETs enable drivers to exchange traffic-related information with each other so that drivers can know the traffic status from the transmitted messages well and then choose a better route to circumvent a traffic jam or take actions promptly to avoid an accident, which is depicted in Figure 1.2. According to [9], safety applications based on V2V and V2I communications are further categorized into four types, which are intersection collision avoidance, public safety [10], sign extension, vehicle diagnosis and maintenance.

Apart from the safety-oriented applications, VANETs can also provide additional applications, such as commercial applications and convenience applications. Commercial applications may include internet access, digital map downloading, value-added advertisement and remote diagnosis. Convenience application means to enhance traffic efficiency by providing more convenience for the drivers. For example, VANETs facilitate the electronic toll collection, which means that the payment of toll can be done automatically through a toll collection point by communicating with the OBU of the vehicle.

1.2.3 Security and Privacy Requirements of VANETs

In the work of [11], the authors suggested some basic security and privacy requirements, such as source authentication, message integrity and user anonymity, for VANETS. There are some other literatures that propose additional requirements, such as scalability in [12], sufficient reaction time in [13]. Below is a list of basic security requirements.

1. **Availability:** Availability means that the network resource should be available for any users at any time. If an attacker broadcasts a large amount of unnecessary messages to occupy most of the bandwidth, the legitimate user cannot use the network. This is especially important in case of an emergency, where network unavailability could result in loss of lives. Hence, availability is an important requirement that needs to be satisfied.

2. **Access Control:** Access control ensures that only authorized users are able to use the network resource. It prevents malicious vehicles from accessing network services and information. Moreover, access control also guarantees that any misbehaving or malicious users will be revoked from the network.

3. **Authentication:** Authentication is the most important requirement for the security of VANETs. For a network user, authentication means that any legitimate user can be identified and verified by its identity and obtains the authorization from trusted parties to access the network. For the transmitted messages, it means that each message is sent from a legitimate user. Hence authentication ensures that the transmitted messages are reliable and sent from a legitimate sender. Typically, vehicles use digital signatures to realize

message authentication. Hence, digital signature schemes should be secure and practical, which means that they should meet the security requirements, incur low computation and communication overhead, and have an efficient revocation mechanism.

4. **Message Integrity:** Message integrity means that a message has not been tampered with or altered during the transmission.

5. **Nonrepudiation:** Nonrepudiation means that if a vehicle broadcasted a message, this vehicle cannot later deny having sent the message. This is very important when there is a dispute. For example, if an accident occurs because of a malicious vehicle sends a fake message, then this malicious car should not be able to deny that the fake message is sent by him/her. Hence, nonrepudiation prevents the attacker from denying the misbehaviour done by him/her.

6. **Privacy:** Privacy means to protect private information, such as location privacy, personal information, from an unauthorized party. More specifically, privacy requires minimum disclosure, anonymity and unlinkability [14]. Minimum disclosure means that the user only needs to reveal sufficient information for the basic functionality in VANETs and keeps the amount of revealed information to the minimum. Anonymity means that the real identity of a sender should be kept hidden. However, due to the accountability requirement, anonymity should be conditional, which means that the privacy of a malicious user should be revealed. Unlinkability is to ensure that the relation between two or more items of interest cannot be linked. A possible linkability example may be that an attacker can link a message to a vehicle, and then further link a vehicle to a person, which could reveal the real identity of the driver.

1.2.4 Security and Privacy Attacks in VANETs

Since the VANET is an open wireless network, it is easy for an attacker to inject or modify messages. Moreover, due to the characteristic of frequent disconnections, VANETs are vulnerable to various attacks and the suspect adversaries are difficult to identify. This section presents the different types of adversaries and attacks.

1.2.4.1 Types of Attackers

According to [15], an attacker can be categorised according to four major dimensions in VANETS.

1. **Insider and Outsider Attacker:** The insider attacker is authenticated and knows the network configurations very well. The outsider attacker is not authenticated. It acts as an intruder, and its capacity to launch an attack is smaller than that of an insider attacker.

2. **Active and Passive Attacker:** The active attacker can broadcast bogus messages or reject to transmit received messages to launch an attack. The passive attacker does not engage in the message communication of the network, and can only eavesdrop on the communication channel.

3. **Malicious and Rational Attacker:** The malicious attacker does not seek personal benefits and only aims to destroy the network, while a rational attacker attacks the network for personal benefits.

4. **Local and Extended Attacker:** A local attacker only has limited control over a specific network area, while an extended attacker can extend his scope by using several entities in different network regions.

1.2.4.2 Types of Attacks

1. **Denial of Service Attack:** The Denial of Service (DoS) attack breaks the availability requirement of VANETs. This attack aims to disturb the network to make it unavailable to legitimate users. More specifically, DoS attacker floods the network by injecting enormous irrelevant bulk messages into the network channel to occupy the bandwidth resource and consume the computational power of vehicles. If the attack is launched by distributed adversaries, it is the distributed denial of service (DDoS) attack. In DDoS attack, the adversaries could distribute in different locations and inject irrelevant messages in different time slots, hence DDoS attack is more difficult to prevent than DoS attack in VANETs.

2. **Sybil Attack:** This attack can also be named as multiple identity attack. An attacker launches this attack by creating many identities to act as multi-

ple vehicles. The attacker broadcasts multiple messages using fake identities to mislead the received vehicles to make incorrect decisions. For instance, if a vehicle receives many messages from a Sybil attacker with different identities at a road intersection, it may feel that there is congestion and decide to choose another route. Hence, a Sybil attack could cause a substantial negative influence on the network. This attack could affect the authentication requirement of VANETs.

3. **Impersonation Attack:** This attack happens when an adversary acts as an authenticated vehicle or RSU of the network [16]. For instance, a malicious vehicle may pretend to be a legitimate vehicle and get into the network, then sent malicious information to mislead other vehicles.

4. **Message Modification Attack:** This is the attack that the safety-related messages are altered during or after the transmission. For instance, an adversary that is involved in a car accident may wish to alter the message about its position or speed that had been broadcasted, to escape from the responsibility of the accident.

5. **Message Replay Attack:** The attacker resends the valid messages previously sent by a legitimate source in order to disturb the traffic [4].

6. **Bogus Information Attack:** An attacker could send fake or false messages to mislead the decisions of other vehicles. For instance, an attacker may broadcast fake warning messages about a traffic jam on its route to misguide other vehicles to choose another route.

7. **Eavesdropping:** In VANETs, it is easy for the eavesdropper to collect any specific information from the transmitted messages. For instance, an attacker could track the target vehicle and extract some personal information, such as driving preference and driver's identity, from the collected messages. This may greatly undermine user privacy.

8. **Repudiation Attack:** In this attack, an adversary attempts to deny sending malicious or fake messages which have caused damages to the network.

9. **GPS Spoofing Attack:** This attack is launched by hiding the real location and broadcasting a false GPS message to misguide the legitimate users. The

adversary creates fake GPS information by manipulating the GPS signal and modifying the information arbitrarily.

Moreover, there exist some attacks that violate user privacy, including location, identity, etc. Below are the two main types of privacy issues.

1. **Personal Information Leakage:** Personal information about the driver including real identity, address, could be easily collected by the adversary.

2. **Location Privacy:** In order to enhance operative awareness, vehicles need to broadcast position information over the network. After collecting and analyzing the position information, the attacker may obtain the moving pattern of the target vehicle.

1.3 Research Focus and Contributions

Security and privacy are the major concerns in the development and acceptance of services of VANETs [17], as any attack could cause life-threatening accidents. Hence, the information security and privacy problems in VANETs must be addressed before VANETs can be deployed successfully. This research project focuses on addressing the security and privacy problems of VANETs by employing appropriate cryptographic techniques.

The main idea of this dissertation can be divided into two parts. The first part is to address the problem of secure message authentication. Since the VANET is an open wireless network, the communication messages could be easily monitored, modified and forged by an attacker. Hence, a security mechanism to tackle this security issue is needed. And message authentication is the most critical mechanism to ensure information security in VANETs. Typically, message authentication is realized by using a digital signature scheme, where messages are signed by legitimate senders and the signatures are verified by the receivers. A secure message authentication scheme ensures that the authenticated message is sent by a legitimate user and has not been altered during transmission. Many authentication schemes, which are based on various kinds of cryptographic primitives, are proposed to address security and privacy issued for VANETs. And these authentication schemes

have their own advantages and disadvantages. In this thesis, we develop a secure and efficient authentication scheme based on a certificateless signature for VANETs. The proposed certificateless signature-based authentication scheme has several advantages, such as enhanced security, improved efficiency, etc. In order to make the authentication scheme practical, we address the revocation problem by using a revocable certificateless signature. And, the revocation efficiency and transparency of the key generation center is enhanced by using the well-known KUNodes algorithm and the blockchain technology respectively. Furthermore, we propose an RSU-assisted authentication process using the data structure, named as cuckoo filter, to improve the overall authentication efficiency.

The second part focuses on addressing the privacy issues of VANETs, especially on protecting the privacy of drivers by developing range proof protocols for anonymous credentials. Since an attacker can collect the exchanged messages from vehicles, the attacker could derive some privacy-sensitive information of the driver or vehicle, such as the home address, driving preference, by using some advanced data analysis techniques. In VANETs, revealing more information than necessary could lead to privacy risks [14], which means that minimum information disclosure should be ensured. Moreover, the transmitted credentials or signatures should be unlinkable, which means that an adversary cannot derive personal information by linking two or more collected messages. Anonymous credential is a promising technique to protect user privacy in VANETs, as it enables the user to prove certain statements about their identity attributes selectively without revealing the corresponding data, which means that minimum information disclosure and unlinkability are ensured and the identity privacy is controlled by the user. For example, one user can use the anonymous credential to selectively prove that his age is greater than 25 years old without revealing his real age to others. And a driver can also use this technique to prove that the mileage of a vehicle lies within a specific range so that he can get some services from a service provider. Actually, anonymous credentials use the range proof technique, which allows someone to prove that he knows a secret value in the interval range, to realize such kind of functionality. Hence, range proof is an important technique of building anonymous credentials to enhance user privacy in VANETs. Range proof uses zero-knowledge proof, which is an advanced cryptographic technique, to prove the statement that a number is in certain range without revealing the number. By using zero-knowledge

proof, the verifier learns nothing but the truth of statement. Hence, strong privacy-preserving is guaranteed. Range proof is widely used in many scenarios, such as E-cash and muti-coupon systems [18], electronic voting [19], cryptocurrency [20], or any protocol that requires to prove that an input value is from a valid range. We specifically, investigated the range proof technique used in the cryptocurrency Monero. We found that the proposed range proof in Monero has security flaws, especially, it lacks a formal security proof. We show that the range proof may not be a proof-of-knowledge by giving a counterexample. Then, we propose an improved version of the range proof protocol and give a formal security proof.

1.4 Thesis Organization

The following chapters of this dissertation are organized as follows. In Chapter 2, I will present a literature review of the message authentication schemes and anonymous credential techniques for VANETs. Then, I present the proposed message authentication scheme for VANETs based on a secure and efficient certificateless signature scheme in Chapter 3. Security and efficiency analysis of the certificateless signature scheme will be presented. Afterwards, in Chapter 4, I focus on addressing the revocation problem by utilizing a revocable certificateless signature, and enhancing the efficiency of authentication by developing the RSU-assisted authentication process. Moreover, the blockchain technology is proposed in the authentication scheme to improve the revocation transparency of the key generation center. In Chapter 5, I describe the range proof protocol used in cryptocurrency Monero, explain the flaws of this range proof protocol, and present the improved range proof protocol with a formal security proof. Lastly, in Chapter 6, the conclusions and further work are presented.

Chapter 2

Literature Review on Security and Privacy of VANET

2.1 Overview of Security and Privacy Mechanisms for VANETs

In the last decade, many research works have emerged to address the security and privacy issues in VANETs. These papers can be roughly classified into two main categories, which are the trust management approaches and cryptography-based approaches. Trust management is the process where vehicles evaluate the quality of the messages transmitted by peers to model the trustworthiness of peers in VANETs. By incorporating trust in VANETs, vehicles can detect dishonest peers or bogus messages. Trust management can also enhance message dissemination by giving incentives to honest peers that help to transmit messages. The existing proposed models for trust management can be categorized into three types, namely, entity oriented trust models, which focus on modelling trust relationships of entities in VANETs, data-oriented trust models, which aim to evaluate the data trustworthiness, and hybrid trust models that combine the trust of both entities and data. However, only limited numbers of trust models are being proposed for assuring trust among neighbouring vehicles in VANETs [21]. And most of the proposals which address the security and privacy issues in VANETs belong to the category which is based on cryptographic techniques.

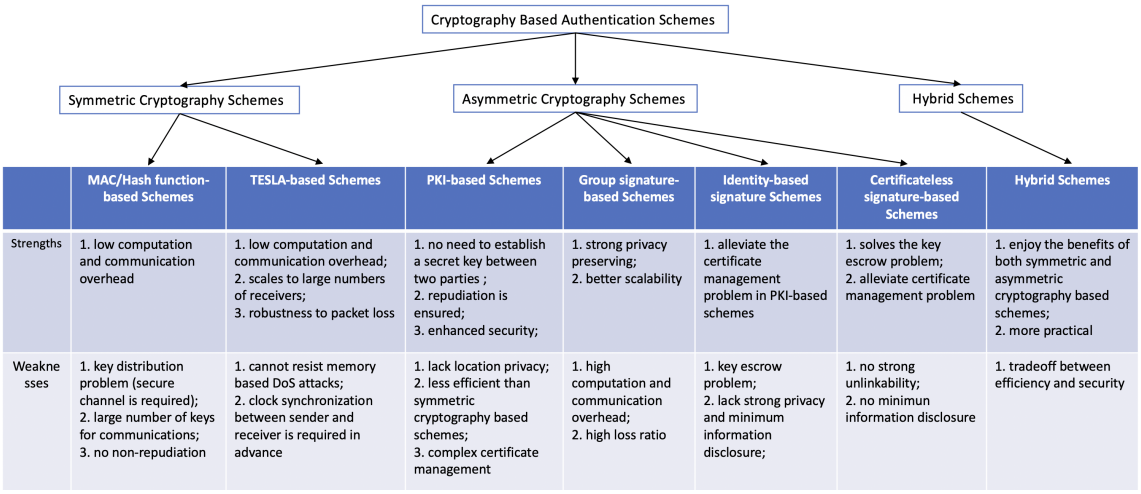
Message authentication is the most important mechanism to achieve secure

communication in VANETs. In a typical message authentication process, a legitimate user generates a signature on traffic-related messages and the corresponding signature is verified by the message receiver. A secure message authentication scheme ensures that the message is sent by an authentic user and has not been altered during the transmission. Hence, both the source legitimacy and message integrity are guaranteed. Therefore, many research works were proposed to use various cryptographic techniques to realize message authentication in VANETs.

Anonymous credential is a promising technique to protect the privacy of the driver in VANETs. Anonymous credential is also known as privacy-preserving attribute-based credential, and it allows a driver to prove certain statements about his/her attributes while keeping the attributes hidden. In VANETs, there are situations where one vehicle may be involved in a communication protocol with another party or vehicle to get certain services. During the process, a vehicle may be required to submit identity information or a credential to prove a certain statement. For example, in order to register into VANETs, a driver needs to provide his personal information such as age, name, addresses to a party get an electronic license plate, which means that identity information is revealed to a third party. Actually, registration could turn out to be a major privacy concern in VANETs [22]. This privacy issue can be solved by using an anonymous credential scheme to allow the driver to prove that his attributes satisfy the requirements without revealing the attributes. Moreover, the identity management of drivers requires privacy protection, and one of the most promising approaches to fulfil this privacy-preserving requirement is the anonymous credential system [23]. In terms of location privacy in VANETs, the anonymous credential is also a commonly used approach to protect location privacy of the driver [24].

2.2 Review on Authentication Schemes of VANETs

In this section, the literature on cryptography-based authentication schemes for VANETs is classified and presented. However, it is not possible to classify these cryptography-based schemes strictly, as many schemes use a combination of several different cryptography techniques and could belong to multiple categories. According to the cryptographic primitives used in the schemes, these authentication schemes for VANETs are classified into symmetric cryptography based schemes,



	MAC/Hash function-based Schemes	TESLA-based Schemes	PKI-based Schemes	Group signature-based Schemes	Identity-based signature Schemes	Certificateless signature-based Schemes	Hybrid Schemes
Strengths	1. low computation and communication overhead	1. low computation and communication overhead; 2. scales to large numbers of receivers; 3. robustness to packet loss	1. no need to establish a secret key between two parties ; 2. repudiation is ensured; 3. enhanced security;	1. strong privacy preserving; 2. better scalability	1. alleviate the certificate management problem in PKI-based schemes	1. solves the key escrow problem; 2. alleviate certificate management problem	1. enjoy the benefits of both symmetric and asymmetric cryptography based schemes; 2. more practical
Weaknesses	1. key distribution problem (secure channel is required); 2. large number of keys for communications; 3. no non-repudiation	1. cannot resist memory based DoS attacks; 2. clock synchronization between sender and receiver is required in advance	1. lack location privacy; 2. less efficient than symmetric cryptography based schemes; 3. complex certificate management	1. high computation and communication overhead; 2. high loss ratio	1. key escrow problem; 2. lack strong privacy and minimum information disclosure;	1. no strong unlinkability; 2. no minimun information disclosure	1. tradeoff between efficiency and security

Figure 2.1: Cryptography based authentication schemes

asymmetric cryptography based schemes, which can be further divided into public key infrastructure based schemes, identity-based schemes, certificateless signature based schemes, and group signature based schemes, and the hybrid cryptography schemes which employ both symmetric and asymmetric cryptographic techniques. The overall differences between these cryptography-based authentication schemes is shown in Figure 2.1. The following section will present a survey of recent literature about the different types of authentication schemes for VANETs.

2.2.1 Symmetric Cryptography Based Authentication Schemes

In a symmetric cryptography system, the sender and receiver share the same secret key and use the same secret key for encryption and decryption or message signing and signature verification. Hence, before processing the communication messages, the sender and the receiver should establish a shared secret key with each other securely and then use the same key for message exchange.

Compared with asymmetric cryptography which uses pairs of keys in the algorithm, symmetric cryptography is time and space-efficient, as only a single secret key is used in the cryptography algorithm. However symmetric cryptographic primitives have several shortcomings. The first one is the key distribution problem, which is that the two communication entities must establish a shared secret key through a secure channel before communication. However, it is not secure to

exchange secret keys between two users over the open wireless network of VANETs. The second problem is the large number of keys for communications. Since every two different users need to establish a unique secret key for communications, the number of keys that needs to be tackled will become very large in a system. For example, if there are 2000 users in the network communicating with each other using symmetric cryptographic primitives, millions of key pairs need to be created and exchanged between two users through a secure channel. The third drawback is the lack of nonrepudiation, which is a basic security requirement for VANETs. This is due to the fact that both communication parties own the same key and can generate the same message using the key. Given the three drawbacks, message authentication schemes for VANETs only based on pure symmetric cryptography are rare. However, symmetric cryptographic primitives, such as message authentication code, are used with other mechanisms to achieve efficient message authentication for VANETs in many research works, which are roughly classified into the following two categories.

2.2.1.1 MAC and Hash Function Based Authentication Schemes

Message authentication code (MAC), also known as a cryptographic checksum, is a short piece of information which is used to ensure the message integrity and authenticity. In the MAC algorithm, the message sender calculates the MAC value on the message using the secret key and sends the message with the MAC to the receiver. On receiving the message, the recipient firstly uses the shared secret key to calculate the MAC value, then check if the computed MAC value is equal to the received MAC value. If they are the same, it means that the message is sent by a valid user who also possesses the secret key and the message has not been altered during transmission. The computation overhead of MAC algorithm is much smaller than a digital signature scheme. Hash function is an algorithm which takes data of arbitrary size as input and outputs a fixed string. It is widely used to ensure the message integrity by sending the message together with the hash value. Hash function can be computed very efficiently and it incurs very little computation overhead to the authentication scheme.

Both MAC and hash function are widely used in many authentication schemes for VANETs. In [25], the author proposed to use symmetric primitives to improve

authentication efficiency. However, the authentication scheme only considers the scenario where roadside units assist the message authentication of vehicles. Hence it is not suitable for message authentication between vehicles where roadside units are not available. Lin et al. [26] proposed a secure and efficient authentication approach for vehicular communications by attaching a MAC tag to each message. By employing MACs and hash functions to realize message authentication, both the computation and communication overhead are decreased. In [27], a decentralized lightweight message authentication scheme using only XOR operations and hash functions is proposed for VANETs. Besides the use of lightweight cryptographic primitives which ensure the high efficiency of authentication, the proposed scheme utilizes the trust-extended mechanism to enhance the efficiency of the authentication process. Rhim et al. [28] proposed a MAC-based authentication scheme for VANETs. However, it cannot resist the replay attack. Hu et al. [29] proposed to employ the Hash-based Message Authentication Code (HMAC) to achieve secure message authentication for VAENTs. The secure V2I communication is realized by symmetric encryption and simple HMAC checking. The communication of V2V within a group is also secured using symmetric encryption and HMAC verification based on a shared secret key. In [30], a lightweight secure communication framework based on symmetric cryptography was proposed for VANETs. This scheme employs lightweight cryptographic primitives, such as HMAC, and uses XOR operation to substitute point addition to minimize the computation and communication cost. The key leak problem due to the joining or leaving of a vehicle is resolved by the proposed group key agreement protocol in this framework. Wang et al. [31] also developed a lightweight authentication scheme for VANETs. The conditional privacy is preserved by using self-generated pseudo-identity and the message authentication is realized by symmetric encryption and MAC calculation, which results in much higher performance than the schemes based on public-key cryptography. Besides, the proposed scheme does not require the vehicles to keep a certificate revocation list (CRL), thus avoiding the overhead incurred by the CRL. Recently, Benyamina et al. [32] proposed a novel lightweight authentication scheme based on MAC for VANETs. The MAC-based authentication scheme not only offers high performance but also ensures the security requirements of privacy preservation and non-repudiation. Moreover, security services, such as biological password login and update, are also provided by the proposed scheme.

2.2.1.2 TESLA-Based Authentication Schemes

The TESLA (timed efficient stream loss-tolerant authentication) based authentication protocol was firstly proposed in [33] to ensure source authentication in broadcast communications. It purely used symmetric cryptographic primitives, such as MACs and one-way hash chain, to achieve efficient message authentication. The sender calculates the MAC value using its key and then broadcasts the message with the MAC for authentication. On receiving the message, the recipient does not check the MAC immediately, but stores the received message in the buff and waits for the message sender to disclose the MAC key at a later time. The key disclosure mechanism is realized by using the one-way hash chain. The message sender computes the MAC tag using the key that corresponds to the next element of the hash chain. After the predefined time delay, the sender broadcasts the next hash chain element corresponding to the MAC key, then the message receiver firstly validates the authenticity of the MAC key by performing a hash operation and then uses the MAC key to verify the MAC tag.

Since the authentication protocol only uses lightweight MAC and hash operations, the computation and communication cost is much smaller than the authentication protocols based on public-key cryptography. However, TESLA has a drawback that it cannot resist the denial-of-service (DoS) attack. Since the received messages are buffed before being authenticated, a DoS attacker can send a large amount of messages to occupy the receiver's memory. In order to solve this problem, some TESLA-based authentication schemes that can resist DoS attack are proposed, such as [34–36]. For example, in [36], a modified version of TESLA, called TESLA++ was proposed to achieve efficient message authentication in VANETs. This modified version inherits the advantage of high computation efficiency of TESLA and is able to resist DoS attack by decreasing the memory requirements for authentication using self-generated MACs from the receiver. However, this enhanced version has the drawbacks that it cannot support multi-hop authentication and the basic security requirement non-repudiation is not ensured. Jahanian et al. [37] presented an analysis of TESLA protocol in VANETs to investigate attacks and improve the security of the protocol. More specifically, it investigated the timeliness of TESLA protocol by a model checking method using timed colored Petri nets and CPN Tools model checking. The analy-

sis result showed that timely attacks could be resisted by improving the awareness of the sender about the situation if loose synchronization exists in the network [37]. Based on TESLA, Lyu et al. [38] proposed a prediction-based authentication (PBA) for V2V communications. Same as TESLA, it utilized the hash chain to store private keys and used each hash chain element for authentication. By only storing shortened MACs of signatures in the authentication process, the storage requirement is reduced and the DoS attack is prevented. Recently, inspired by [38], Bao et al. [39] proposed a lightweight authentication scheme for VANETs based on TESLA and bloom filter. Compared with the authentication scheme in [38], this scheme has two main improvements. Firstly, instead of using Elliptic Curve Digital Signature Algorithm (ECDSA) to verify the keys, this scheme proposed to use a bloom filter to authenticate keys to reduce overhead. Secondly, the proposed scheme enhanced the privacy of the authentication process by using the joint pseudonym changing mechanism.

2.2.2 Asymmetric Cryptography Based Authentication Schemes

Asymmetric cryptography, which is also known as public-key cryptography, is a cryptographic system that uses a pair of keys: the public key, which is known to all, and the private key, which is kept secret by the owner. Public key encryption and digital signature are two public key cryptography techniques that are widely used in many protocols. Unlike symmetric key encryption in which one secret key is used for both encryption and decryption, in a public key encryption algorithm, the message is encrypted by the sender using the receiver's public key, and then the resulting ciphertext is decrypted by the recipient using the corresponding private key. In a digital signature algorithm, a message is signed using the signer's private key and the resulting signature is verified using the corresponding public key. Since only the sender, who owns the private key, can generate a valid signature, the successful signature verification ensures that the message is indeed signed by a sender that associated with the public key and the message has not been modified during transmission. A significant issue of public-key cryptography is to ensure the authenticity of the public key. In other words, we should guarantee that the public key indeed belongs to a certain user. Typically, this problem is solved by using a certificate, that is issued by a certificate authority (CA), to bind the public

key to the corresponding user.

Public key cryptography has three main advantages over the conventional symmetric cryptography. Firstly, public-key cryptography eliminates the key distribution problem, as there is no need to establish a shared secret key between the message sender and receiver. Secondly, public-key cryptography ensures non-repudiation, which is a necessary security requirement in VANETs. Thirdly, since the private key is only kept by the message sender and will not be transmitted over a network, public key cryptography provides enhanced security. The main disadvantage of public-key cryptography is that it offers relatively lower computation efficiency than asymmetric cryptography. Public-key cryptography is the fundamental tools used in many modern cryptosystems and protocols to provide various security services, such as message confidentiality, entity authentication, non-repudiation, etc. Over the last two decades, many research works proposed to employ public-key cryptography based authentication schemes to address the security and privacy issues in VANETs have emerged. Based on different cryptographical techniques used for authentication, these research works can be roughly classified into four types, which are public-key infrastructure (PKI)-based schemes, identity-based signature schemes, group signature-based scheme and certificateless signature-based schemes.

2.2.2.1 PKI-based Authentication Schemes

Public key infrastructure (PKI) is a centralized trusted authority that creates, manages, and revokes the certificate which binds the public key with the corresponding legitimate entity. The binding process is realized by the registration process, where a certificate authority (CA) validates the legitimacy of the user and generate a digital signature (certificate) on the user's public key. Then the certificate is stored in the local repository of PKI. Moreover, PKI keeps a certificate revocation list (CRL) of the certificates that should be revoked and broadcasts the latest CRL over the network periodically.

The PKI-based authentication scheme is the most traditional scheme proposed to secure V2V and V2I communications in VANETs. Typically, in a PKI-based authentication scheme for VANETs, the CA works as a third trusted party that is in charge of user registration, certificates generation and revocation. Typically,

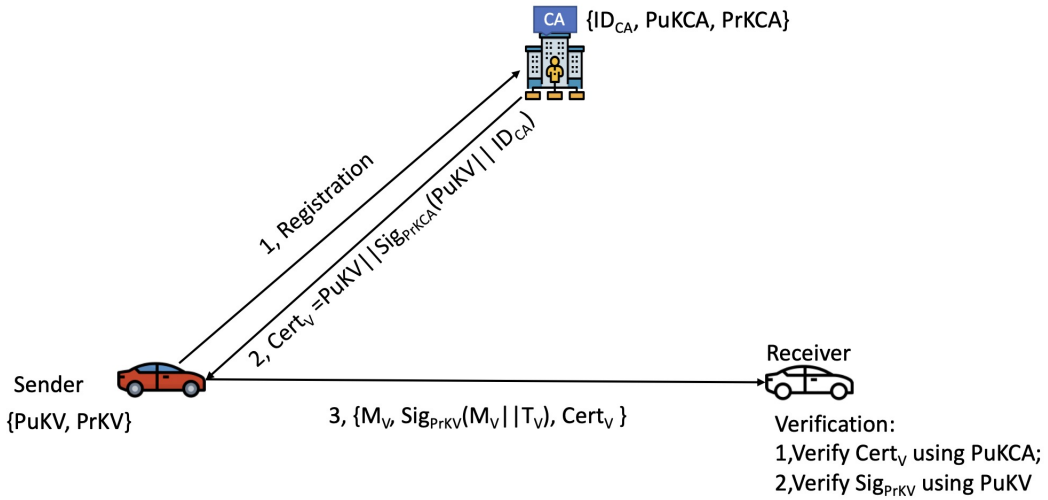


Figure 2.2: Typical PKI-based authentication process

a vehicle can communicate with the CA directly or through a nearby RSU. The typical PKI-based message authentication process is depicted in Figure 2.2. Firstly, every vehicle should register with the CA to obtain a certificate. After verifying the legitimacy of the vehicle, CA generates a signature (certificate) on the public key of the vehicle using its own private key and issues the certificate ($Cert_V$) to the vehicle. The CA also stores the certificate with the map to the corresponding public key in its local storage for user revocation. After registration, the vehicle uses its own private key ($PrKV$) to generate a signature (Sig_{PrKV}) on the message (M_V) and the timestamp (T_V) and broadcasts the signature together with the certificate for V2V and V2I communications. Once a vehicle or RSU receives the message, it firstly checks whether the certificate has been revoked or not by searching the CRL. If the certificate does not exist in the CRL, the message receiver will check the validity of the certificate using the public key ($PuKCA$) of the CA and further verify the signature on the safety message using the public key ($PuKV$) of the sender. Finally, the receiver accepts the safety message once both the certificate and signature are verified as valid.

In 2005, Raya et al. [40] introduced a PKI-based anonymous message authentication scheme for VANETs. In this authentication scheme, vehicles register with the PKI to obtain pseudonyms and certificates and use them for message authentication in VANETs. In order to protect user privacy, vehicles need to use different

pseudonyms and certificates for each communication message to avoid tracking. Hence, vehicles are required to preload a large number of pseudonyms, certificates and the corresponding private keys, which need large storage space. Another drawback is that the certificate revocation list (CRL) will become very large, hence the updating and checking of CRL will incur high communication overhead. Hence, many research works were proposed to address various issues related to burdensome certificate management. Calandriello et al. [41] proposed a hybrid scheme using baseline pseudonym and group signatures to achieve efficient and robust message authentication in VANETs. The baseline pseudonym is obtained from the certificate authority, and it is attached with the corresponding certificate in each message for authentication. By using the group signature to allow legitimate vehicles to generate their pseudonyms on-the-fly and self-certification, the complex credential management problem is alleviated and the communication overhead is reduced. Lin et al. [42] specifically addressed the two significant problems, certification revocation and conditional privacy-preserving. Instead of using the central repository to handle CRL, it proposed the RSU-assisted certificate revocation mechanism, in which RSU is also trusted, to improve the revocation efficiency. Moreover, it used a group signature and an identity-based signature to achieve conditional privacy-preserving. Wasef et al. [43] proposed an enhanced PKI-based authentication scheme to protect location privacy and realize distributed revocation. More specifically, it used random encryption periods mechanism to allow vehicles to change certificates in an encrypted communication zone to protect the location privacy, and used a secret sharing scheme to realize distributed revocation, and proposed to resist DoS attack by allowing each vehicle to keep tracking of all the invalid signatures received in a period. However, as stated in the conclusion, efficient revocation and privacy-preserving of PKI-based schemes are still challenging issues that need to be tackled. Moreover, Wasef et al. [44] developed an expedite message authentication protocol for VANETs which aims to decrease the delay incurred by CRL checking in the PKI system. The proposed EMAP protocol used a keyed Hash Message Authentication Code (HMAC) to realize the efficient revocation checking process. By decreasing the delay due to the CRL checking, the loss ratio of EMAP is much smaller than that of conventional PKI-based schemes using CRL checking. In [45], the problem of information leaking due to revocation checking was addressed. The author proposed to utilize a one-

way accumulator to enable CA to accumulate multiple revoked certificates into one value and allow a vehicle to prove that its certificate has not been accumulated, which means that its certificate has not been revoked. The analysis showed that this scheme not only preserved user privacy but also greatly reduced the revocation cost. In [46], Ganan et al. proposed a privacy-aware revocation mechanism using Merkle Hash Tree (MHT) and a crowds-based anonymous protocol. It used MHT to generate a positive proof which allows a vehicle to prove that a given certificate has not been revoked, hence the vehicle did not need to download the large CRL and the revocation cost was greatly reduced. Another advantage is that the proposed scheme also enhanced user privacy, as the broadcast messages of vehicles cannot be traced by an attacker. In 2016, Islam et al. [47] proposed a novel revocation mechanism to decrease the long delay caused by CRL checking. It avoids CRL checking by using a secret key sharing mechanism which allows non-revoked vehicles to update their secret information. More recently, Junior [48] developed a flexible revocation mechanism to allow temporary certification revocation. It also supported the linkage of pseudonym certificates. These functionalities are useful to implement suspension mechanisms or to assist the investigations by law-enforcement authorities [48]. In [49], the author specifically addressed the issue of efficient distribution of certificate revocation list. The proposed mechanism achieves timely distribution of CRL with small computation and communication overhead and can protect strong user privacy against honest-but-curious attackers. The efficient CRL checking is realized by appending a fingerprint of CRL to the corresponding pseudonyms for verification. Experiment evaluation shows that the proposed CRL distribution mechanism is not only efficient and scalable but also secure against DoS attacks.

The pseudonym change is another significant issue that closely influences privacy-preserving of VANETs. There are many schemes proposed to address the issue of pseudonym change. Lu et al. [50] addressed the issue of pseudonym change in PKI-based authentication schemes. In order to tackle of improper pseudonym change, Lu et al. [50] proposed an effective pseudonym change strategy at social spots, which worked as a mix zone to allow several vehicles to temporarily change their pseudonyms. Moreover, a simplified game-theoretic technique was utilized to prove that the pseudonym change strategy is feasible and effective. However, the strategy cannot work well when the density of vehicles is very low. Boualouache

et al. [51] proposed traffic-aware pseudonym changing strategy based for VANETs on the radio silence technique, aiming to prevent various pseudonym linking attacks by a strong passive adversary. It is based on a traffic congestion detection technique to trigger the change strategy, which is made up of five phases. The strategy based on radio silence technique is more effective in resisting both external and internal attackers than the strategy based on the technique of encrypting messages. However, the synchronization of this strategy is based on the protocol instead of infrastructure, which can control a mix zone, hence, this strategy cannot ensure that whether the vehicles will perform the pseudonym change or not. According to [52], pseudonym changing strategy can be classified as the following four types: applying silence, periodical synchronous change, the mix zones and location obfuscation. They have their own advantages and disadvantages. For example, approaches based on mix-zone has high privacy protection level, but it must depend on a crowd or an infrastructure. This paper develops a strategy that enjoys the privacy protection of being within a crowd but reduces the linkability when the road density is low using the technique of obfuscation. Moreover, it also avoids the overhead incurred by synchronization between vehicles.

2.2.2.2 Group Signature Based Authentication Schemes

Group signature was firstly proposed by Chaum and van Heyst [53] in 1991, to provide signer anonymity against the receiver. The main idea of group signatures is that any member of a group signs messages on behalf of the whole group. Hence, after verifying the validity of the signature, the verifier only learns that the message is signed by someone belongs to the group and the anonymity of the actual signer is protected. However, usually, there exists a trusted group manager in the system, who is responsible for group initialization, new member admission and has the ability to reveal the identity of the actual signer of a group signature. The group manager initializes the system by selecting its own private key and group public key, which is used for verifying the group signature, and defining the public group parameters. After the initialization, the group manager uses its private key to generate membership certificates for group members. Then, any group member can generate a group signature on any message using the certificate issued by the group manager. And any verifier can verify the validity of the group signature

using the group public key.

The role of group manager in a group signature scheme is similar to the role of a certificate authority in a PKI-based signature scheme, as both of them are responsible for new user admission, issuing certificates, and user revocation. However, the difference is that the certificates in the PKI-based signature scheme are public for verification, whereas the certificates in a group signature scheme are secret and should not be revealed by a group member. The main advantage over the PKI-based signature is that the group signature offers signer anonymity, which means that after the signature verification, the verifier is only convinced that the message is signed by a member of the group but cannot identify the actual signer. By using group signatures to realize secure communication in VANETs, the privacy of the sending vehicle is ensured. Moreover, group signature also offers better scalability, as only one public group key is used for verification. The main shortcoming of group signatures is the high computation cost of the signature verification, which causes high message loss ratio. Another drawback is that due to the fact a vehicle could join or leave a group at any time, the frequent key generation and distribution could also incur high computation and communication overhead to the system. Due to the main feature of privacy protection of the signer, group signature schemes are widely used in many privacy-preserving authentication schemes for VANETs.

Lin et al. [54] proposed a message authentication scheme with privacy-preserving for VANETs using the group signature and ID-based signature. The secure V2V communication is realized by allowing the sending vehicle anonymously sign messages using the group signature. The V2I communication is secured using the ID-based signature scheme. In [55], Guo et al. proposed a secure and privacy-preserving communication framework for vehicular communications using group signatures. The message authentication and privacy-preserving are ensured by employing a group signature scheme. Moreover, this framework also offers a scalable role-based access control method based on a trusted temple-resistant device to resist various attacks on VANETs. In [56], a short group signature scheme is used in the distributed key management framework to ensure privacy-preserving of VANETs. More specifically, a group of vehicles is formed near an RSU, which is semi-trusted and distributes the private key for each group member. Aimed to reduce the computation cost of group signature verification, a cooperative mes-

sage authentication protocol is proposed in this framework. However, the issue of high overhead caused by frequent key establishment process, where a vehicle approaches a new RSU, was not addressed in the framework. In 2017, Lim et al. [57] specifically addressed the scalability and efficiency issues of key distribution of group signature schemes for VANETs. It employed the Diffie-Hellman protocol to securely establish a shared symmetric key between a vehicle and an RSU. In order to decrease the frequency of key establishment between a vehicle and a RSU, it introduced the concept of domain with multiple RSUs. Shao et al. [58] proposed a threshold anonymous authentication protocol for VANETs using a new group signature. Efficient traceability and linkability are both supported by this protocol. In order to increase the verification efficiency, the scheme supports batch verification, which allows multiple signatures to be verified in a single instance. However, the proposed protocol does not consider the high computation and communication cost caused by the certificate management problem in VANETs. Alimohammadi et al. [59] proposed to use the Boneh-Shacham short group signature and the batch verification to realize efficient and secure message authentication for VANETs. Moreover, the proposed protocol can detect double registration of a vehicle and prevent Sybil attack by using retransmission checking. And the Sybil attack detection process does not involve the participation of CA or RSUs, hence, no extra computation overhead is imposed on CA or RSU. Zhang et al. [60] proposed a secure and privacy-preserving protocol to address the privacy issues in location-based service (LBS) for VANETs. A vehicle uses the group member key to generate a verifier-local group signature, which will be verified by an LBS provider without undermining the privacy of the sending vehicle. Moreover, in order to improve efficiency, the LBS scheme uses asymmetric pairings instead of symmetric pairings.

2.2.2.3 Identity-based Signature Authentication Schemes

Identity-based public key cryptography (ID-PKC) was firstly proposed by Shamir to avoid the key management problem of conventional PKI-based cryptography in [61]. In ID-PKC, the user can use its own identities, such as name and email address, as its public key, for encryption and signature verification. The corresponding private key is obtained from a trusted key generation center (KGC),

which generates the private key using its master secret key. Hence, no certificates are used in ID-PKC and the complexity of certificates generation and management is eliminated. The typical process of an ID-based signature is shown as follows:

- Setup: The KGC generates the master private key msk_{KGC}, and public parameters. KGC publishes the public parameters over the network.
- Key Extraction: The sender authenticates with the KGC using its identity ID_{sender} and obtains its private key $sk_{ID_{sender}}$, which is computed using KGC's master secret key msk_{KGC}.
- Signature Generation: The sender produces a signature σ on message M using $sk_{ID_{sender}}$, and transmits $\{\sigma, M\}$ to the verifier.
- Signature Verification: After receiving the message from the sender, the recipient verifies the validity of the signature σ using the sender's identity (public key) ID_{sender} and the KGC's public key. If the signature is verified to be valid, the recipient accepts the message. Otherwise, the recipient rejects the message.

In PKI-based cryptography schemes, the CA only issues a certificate on the public key of a user and does not know the corresponding secret key. However, in an ID-PKC scheme, the secret keys of all the users are derived by the KGC using its master secret key. This indicates that the KGC has the ability to generate signatures on behalf of any user, and this breaks the nonrepudiation of the signature scheme. This is known as the key escrow problem, which sis the major shortcoming of the ID-based signature.

ID-based signature is widely used in many message authentication schemes for VANETs. In 2008, Zhang et al. [62] proposed an efficient message authentication scheme for vehicular sensor networks using ID-based signature and batch verification technique. In this scheme, each vehicle firstly generated a pseudo-identity, then further used it to produce an ID-based one-time signature for authentication to avoid the cost of public key certificate. The use of batch verification reduced the verification cost. However, every vehicle used a tamper-resistant device to keep a long-term master secret key. This could be a serious security issue, as the master secret key could be learned by an attacker through a side-channel attack. Sun et

al. [63] proposed an identity-based privacy-preserving authentication scheme for VANETs. The privacy and traceability are ensured using pseudonyms. It utilized the ID-based cryptosystem, which avoids the certificate management problem, to achieve secure message authentication. The performance analysis showed that the computation and communication cost of the proposed scheme was lower than that of PKI-based schemes. Shim et al.[64] employed a new ID-based signature to achieve efficient privacy-preserving message authentication for VANETs. This signature scheme does not use the computation expensive map-to-point hash function, which reduces the computation cost. The conditional privacy-preserving is ensured by using the pseudo-identity and TA's ability to reveal the real identity of any vehicle using its private key. Another advantage of this scheme is that it does not require the long-term master secret key to be stored in a tamper-resistant device. However, this signature scheme has the drawback that the expensive pairing operations are used and degrade the computation performance. Lo et al. [65] proposed an efficient authentication scheme using a pairing-free ID-based signature scheme. The signature scheme offers high efficiency, as both the expensive map-to-point hash function and pairing operations are not required. The basic security requirements including anonymous authentication, privacy-preserving, and traceability are satisfied by this scheme. Zhang et al. [66] employed an identity-based signature to realize message authentication in VANETs. Due to the feature of ID-based signature, no certificate is used for communication. Hence, the computation and communication overhead caused by certificate management and CRL checking is eliminated. Moreover, the proposed scheme utilizes the technique of hierarchical signature aggregation and batch verification, by which multiple signatures can be aggregated together and verified in a batch, to further enhance the efficiency of the message authentication. Wang et al. [67] proposed a local ID-based anonymous authentication scheme for VANETs. It utilized both the PKI-based certificate and ID-based signature to realize efficient and secure message authentication. Specifically, each vehicle and RSU maintains a long-term PKI certificate for mutual authentication. Then a vehicle can obtain the local master private key from a nearby RSU and use it to generate its local anonymous identity, which is used to produce the ID-based signature to realize authentication of a safety-related message.

2.2.2.4 Certificateless Signature Based Schemes

In order to solve the inherent key escrow problem of ID-based cryptography, certificateless cryptography was firstly proposed by Al-Riyami in [68]. In a certificateless signature scheme, the private key of a user is computed by two different parties, not just by the KGC alone. Firstly, the user obtains a partial private key generated by the KGC. Then the user chooses a secret value by itself and uses the secret value and the partial private key to generate its private key for signature generation. Hence, in certificateless cryptography, the KGC does not know the private key for all the users, which means that the key escrow problem in ID-based cryptography is eliminated. Similar to ID-based cryptography, no certificate is required to authenticate the public key of a user in certificateless based cryptography, thus the complex certificate management problem is avoided. A typical certificateless signature scheme is made up of the following algorithms:

- Setup: The KGC uses the security parameters to generates the master secret key msk, master public key mpk and publish the public parameters $param$.
- Partial Private Key Extraction: The KGC generates a partial private key psk_{ID} for an identity ID using msk, mpk and $param$.
- Secret Value Generation: The user with identity ID generates its own secret value x_{ID} using the system parameters $param$.
- Private Key Generation: The user computes the private key sk_{ID} using partial private key psk and secret value x_{ID}.
- Public Key Generation: The user generates its public key pk_{ID} using $params$, mpk and ID.
- Signature Generation: The user uses the private key sk_{ID} to compute a certificateless signature σ on message m.
- Signature Verification: On receiving the message m and signature σ, the receiver checks the validity of the signature σ using public parameters $param$, master public key mpk, identity ID, and the sender's public key pk_{ID}.

Due to the advantages of no certificate management problem and no key escrow problem, certificateless signature was proposed in many authentication schemes to realize efficient and secure communications in VANETs. In [69], Mohanty et al. proposed to use a certificateless signature scheme to realize RSU-assisted message authentication. The privacy is preserved by using pseudonyms. Signature aggregation and batch verification are utilized to improve authentication efficiency. In [7], an efficient message authentication scheme based on certificateless signature was proposed to secure V2I communications. Unlike most of the certificateless signature scheme, in which many pairing operations are required and the number of pairing operations increases linearly as the number of the signers, only constant number of pairing operations are required in the proposed certificateless signature scheme for VANETs. Hence, the proposed certificateless signature scheme has a lower computation overhead compared with other schemes. Moreover, the techniques of signature aggregation and batch verification are also supported in the certificateless signature scheme. The security analysis shows that the certificateless signature-based authentication scheme provides anonymous authentication, message integrity and unlinkability. More recently, Cui et al. [70] proposed a pairing-free certificateless signature scheme to realize efficient and secure message authentication for VANETs. Without requiring expensive pairing operations and map-to-point hash functions, the proposed scheme achieves higher performance in terms of signature computation over than many other schemes.

2.2.3 Hybrid Cryptography Based Schemes

Given that each cryptographic technique has its own advantages and disadvantages, an increasing number of literature proposing to use hybrid cryptographic techniques to achieve efficient and secure message authentication for VANETs. Hybrid authentication schemes may combine symmetric cryptography, such as MAC, and asymmetric cryptography, such as PKI-based cryptography, group signature, ID-based signature, certificateless signature.

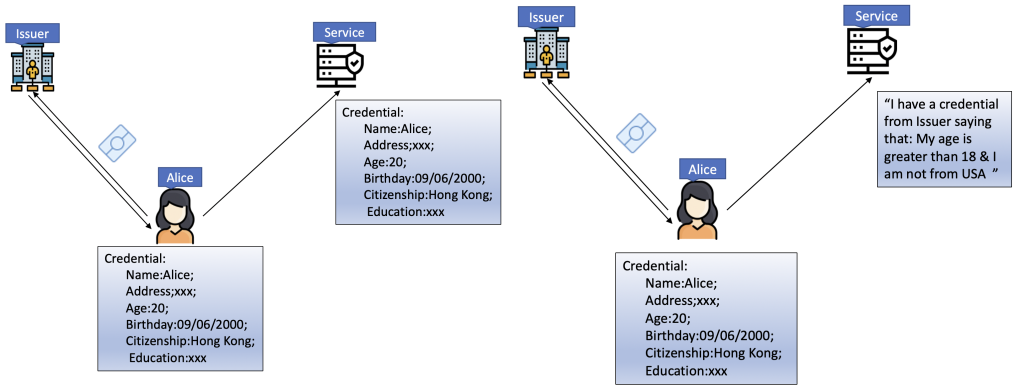
Wang et al. [71] proposed a two-factor lightweight authentication scheme to address the security issues in VANETs by employing the decentralized certificate authority and the biological password. The message authentication process in V2V communications only requires lightweight hash functions and MACs opera-

tions. Hence, this scheme is much more efficient than previous schemes, which are based on complex symmetric cryptography, in terms of both computation and communication overhead. Moreover, due to the nature of CA decentralization, the certificate distribution is not needed, which means that the complexity of certificate management and CRL checking is eliminated. The biological password-based authentication also offers non-repudiation and can resist DoS attacks. In [72], Jiang et al. aim to solve the CRL checking problems using group signature and HMAC. The CRL checking is replaced by computing HMAC, which is calculated locally by vehicles. Hence, the computation and communication overhead due to CRL checking is eliminated. Moreover, the group signature is used with HMAC to guarantee that all the group members are legitimate users. In order to enhance the performance, the proposed scheme utilizes an ID-based signature to realize batch authentication. Asl et al. [73] proposed a symmetric non-repudiated message authentication scheme for VANETs by combining the techniques of symmetric key encryption and digital signature. The symmetric cryptography primitive named as message authentication code (MAC), is attached to the traffic-related message to ensure the message integrity. The PKI-based signature scheme is used by vehicles to sign messages to guarantee message authenticity and non-repudiation. The simulation results show that this scheme has a better performance than that is based on asymmetric cryptographic techniques. In [74], a privacy-preserving authentication scheme for VANETs based on hybrid techniques is proposed. The hybrid scheme utilizes pseudonym-based cryptography and group signatures. The combined scheme eliminates the certificate management problem and does not incur overhead due to group management. The proposed pseudonym approach can offer a trapdoor mechanism to realize the detection of malicious group members and ensure conditional anonymity. Tangade et al. [75] proposed a decentralized message authentication scheme for VANETs using hybrid cryptographic techniques. Specifically, it utilizes the asymmetric ID-based signature scheme to realize secure V2I communications and employs the symmetric HMAC scheme to achieve efficient and secure V2V communications. Performance analysis shows that the proposed hybrid scheme offers higher efficiency in terms of computation and communication overhead than the other schemes. Another novel advantage of the proposed scheme is that the feature of decentralization suit the rapid change of network topology of VANETs.

2.3 Review on Anonymous Credentials

A credential is issued by an organization to attest that the credential holder has certain features, such as nationality, age, or high education degree. It can be regarded as a digital signature of the issuer on certain attribute-value pairs. A typical process of a credential system is that a user obtains a credential from an organization using the corresponding proofs and then shows the credential to a service provider to get access to the service. In order to enhance user privacy, the anonymous credential system was firstly introduced by Chaum [76] in 1985. It is one of the most promising techniques that can satisfy the need for a combination of strong security and privacy protection [23]. It allows the user to selectively prove statements about certain identity attribute to another party while still keeping the corresponding data hidden. Anonymous credential is constructed by a cryptographic technique, called zero-knowledge proof of knowledge, which is used to prove the ownership of certain private information anonymously. Compared with conventional credential systems, anonymous credentials have four main advantages. The first one is that by using anonymous credentials, user does not need to transmit the credentials itself, but rather just convinces the verifier that his attributes satisfy certain properties without revealing the real identity to the verifier. The second advantage is that the user can selectively reveal any subset of the attributes. For example, someone may only want to prove his birthday day is within a certain range while keeping others attributes hidden. The third advantage is that it allows users to prove some complex predicate over the attributes, such as predicate that includes logical operation OR. The last advantage is that anonymous credential guarantees strong privacy-preserving, as the unlinkability is ensured by zero-knowledge proof.

A basic comparison between use case of traditional credentials and anonymous credentials is shown in the following figure, where Figure 2.3b shows working model of anonymous credentials. The main difference is that by using anonymous credentials, Alice can get the services by proving the required statement instead of showing the original credential to others. Another benefit is that Alice can choose to prove any subset of the attributes to the service provider instead of all the attributes. This means that minimum information is revealed for the authentication, and the attribute privacy is self-controlled by the user. For example, as shown in



(a) Credential (b) Anonymous Credential

Figure 2.3: Working models of different credentials

Figure 2.3b, the service may only require Alice to prove that she is not a USA citizen and is over 18 years old. Then, Alice can obtain the service by proving the statement that her age is greater than 18, and she is not a USA citizen.

Many of the proposed authentication schemes for VANETs have some privacy issues. In a typical PKI-based authentication scheme for VANETs, the use of certificates could leak some privacy-related information, such as the identity attributes of the driver [77]. The commonly employed approach to enhance the unlinkability is to use a pseudonym change strategy, in which a vehicle generates many pseudonyms and uses a different pseudonym for each message. However, this approach does not work well when a vehicle changes pseudonyms in a scarce network. In order to achieve minimum information disclosure and strong unlinkability, anonymous credential was proposed in some research works to enhance the privacy in VANETs.

In [78], the author proposed to use an anonymous attribute-based credential to represent vehicle authorization. Specifically, by using anonymous credentials, a vehicle can on-the-fly prove ownership of the required credential while ensuring minimum information disclosure and credential unlinkability. It employed the Persiano and Visconti anonymous credential to make it non-interactive so that it is practical to be used in vehicular networks. A prototype was implemented to analyze the computation and time cost of the proposed solution and the result showed that the proposed scheme is suitable for the scenarios of vehicular networks. In [79], Forster et al. aimed to achieve full anonymity by using an anonymous creden-

tial scheme. It is built based on the basic pseudonym scheme and only the process where a vehicle obtains a pseudonym is changed by using a periodic n-show credential to achieve full anonymity. It also supports the voluntary revocation controlled by the user. In 2017, Fuentes et al. [80] provided the first assessment of the feasibility of three anonymous credential systems in preserving privacy in VANETs. Specifically, it focused on investigating the well-known anonymous credentials, namely, Idmix and U-Prove, and the VANET-update Persiano proposed in [78]. It assessed the three anonymous credentials under a set of use cases for smart cities, which are set by the European Telecommunications Standards Institute (ETSI). Moreover, the authors also adapted the attribute-based credential mechanisms to the scenario of vehicular networks and analyzed the performance. The analysis and experiment results showed that Idemix was the most suitable technique to be used in vehicular networks. Singh et al. [81] proposed a privacy-preserving authentication framework with misbehaviour detection based on restricted usage of anonymous credentials. Honest vehicles enjoy strong anonymity protection by anonymous credential. However, if misbehaving vehicles attempt to use multiple pseudonyms within a time interval, these pseudonyms will be linkable by the embedded tracking ID in the signature, so that those misbehaving vehicles will be detected. A prototype of the proposed anonymous credential framework was implemented in Java based on Idemix library to analyze its feasibility. However, the revocation efficiency is too low, as it is estimated to cost 11 hours to revoke 100000 vehicles. Hence, further optimization is needed to enhance the performance, for example, a hardware accelerator could be employed to perform the revocation process. Neven et al. [82] provided a generalization of the anonymous credential scheme in [81]. In the proposed scheme, a vehicle generates one valid pseudonym locally at a time interval for anonymous authentication. It allows the increased frequency of pseudonym change without increasing the threat of Sybil attacks. However, this paper does not conduct implementation of the proposed anonymous credential, and only provides a conceptual insight on the feasibility of using attribute-based anonymous credentials to protect privacy in VANETs.

2.3.1 Zero-Knowledge Range Proof

From Figure 2.3b, we can see that zero-knowledge range proof is needed to protect the privacy of information that is related to an integer, such as age, birthday, mileage of a vehicle, etc. Hence, as the necessary component of building anonymous credentials, zero-knowledge range proof should be studied. Range proof is an applied cryptographic technique to enable a party to prove that a secret integer is in an interval range [83]. It is needed in a variety of cryptographic protocols, such as anonymous credentials, e-cash, etc [84]. In recent years, range proof technique is becoming widely used in the area of cryptocurrency, as it could be developed to enhance the privacy, which is a very important requirement for cryptocurrency. For example, the cryptocurrency Monero developed a range proof protocol that is used to hide the transaction amount. Range proof is also used in Ethereum [20], which is the cryptocurrency with the second largest market cap in the world.

2.3.1.1 Zero-knowledge Proof

Zero-knowledge proof (ZKP) is a cryptographic protocol that allows the prover to prove to the verifier that a statement about certain secret information is true, without revealing the secret. For example, ZKP allows you to prove that your age is over 25 while still keeping your actual age hidden. Zero-knowledge proofs were first conceived in 1985 by Shafi Goldwasser, Silvio Micali, and Charles Rackoff in their paper [85]. There are three basic requirements of a zero-knowledge proof protocol, which are described as follows.

1. Completeness. If both prover and verifier are honest and the statement is true, the verifier should be convinced of the truth of the statement by the honest prover.

2. Soundness. If the statement is false, no cheating prover can convince the verifier that the statement is true, except with negligible probability.

3. Zero-knowledge. The proof protocol does not leak any information to the verifier. In other words, if the statement is true, the verifier only learns the fact that the statement is true.

ZKP is very useful in preserving privacy in the area of distributed ledger technology (DLT). This is because, in DLT all the transactions are visible to all the

participant nodes, in other words, there is no data privacy. And ZKP can be used to ensure that the transactions are valid while information about the sender the receiver and the transaction amount remain hidden. Furthermore, ZKP can be used to prove statements in various scenarios to protect user privacy. For example, a user can use a ZKP to prove that his/her account has enough available balance for a transaction, to prove membership without revealing his/her identity, to prove he/she knows a solution to a puzzle without revealing the solution, etc.

2.3.1.2 Range Proof

Range proof allows the prover to prove that a committed integer lies in a specific interval without revealing any information about the integer. It has many cryptographic applications, such as anonymous credentials, E-cash, multi-coupon system, electronic voting and any other zero-knowledge protocol that requires to prove certain integer lies in a specific integer range. The problem of range proof has been studied extensively. Early constructions of range proof [86, 87] are inexact, i.e. the proved interval is much larger than the actual interval. Then, in [88], an exact range proof relying on the fact that a number lies in $[0, 2^n]$ iff it can be represented as a n bits binary string is given. Our range proof for Monero, which is presented in Chapter 4, is also adapted from this range proof.

Subsequently, there are also many works constructing more efficient exact range proofs. However, we observe that these improved constructions are either no better than our range proof in the scenario of Monero or not compatible with the other part of Monero. In particular, the range proof presented in [89] is more efficient than the range proof in [88] (and our range proof) only when the interval is $[0, H]$ for H being not a power of 2, which is not considered in many cases. The range proofs presented in [90, 91] work only in unknown order groups, and the range proofs presented in [84,92–95] works only in bilinear groups, which means it cannot support the classical Elliptical-Curve groups. Besides, the range proofs presented in [92–94] need a trusted setup[1].

[1]Trusted setup is viewed as a drawback because for a distributed currency since, ideally, the system should be without any trusted party.

Chapter 3

Message Authentication based on Certificateless Signature Scheme

Privacy-preserving authentication protocol is one important tool to satisfy the security and privacy requirements. Many such schemes employ the certificateless signature, which not only avoids the key management issue of the PKI-based scheme but also solves the key escrow problem of the ID-based signature scheme. However, many schemes have the drawback that the computational expensive bilinear pairing operation or map-to-point hash function are required. In order to enhance efficiency, certificateless signature schemes for VANETs are usually constructed to supporting signature aggregation or online/offline computation. In this chapter, we propose an efficient privacy-preserving authentication scheme using an online/offline certificateless aggregate signature, which does not require bilinear pairings or map-to-point hash functions, to address the security and privacy issues of VANETs. The proposed scheme is proven to be secure with a rigorous security proof, and it satisfies all the security and privacy requirements with a better performance compared with other related schemes.

3.1 Introduction

The transmitted message, which may include sensitive data concerning the drivers' privacy, in DSRC wireless protocol could be easily monitored, altered and forged [96]. For example, a malicious vehicle may broadcast a fake message to cause a

traffic accident. For message security, the receiver should verify the legitimacy and integrity of the received message before taking further action. Moreover, message non-repudiation should also be guaranteed, which means that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message. For privacy protection, the real identity of a vehicle should be hidden from the adversaries. However, privacy protection should be conditional, as traceability should also be guaranteed, which indicates that the TA should be able to reveal the real identity of a malicious vehicle when it is necessary.

Many privacy-preserving authentication schemes based on traditional public key infrastructure (PKI) [11,97] have been proposed to address the security and privacy issues. However, in PKI-based authentication scheme, a certificate is required for every public key of the vehicle and the RSU, which means that a certificate authority needs manage all the certificates and vehicles may have to preload a large number of public/private key pairs together with the corresponding certificates in the local storage. This causes huge storage burden and also makes it difficult for the authority to manage the certificates. Due to this drawback, PKI-based scheme is not practical and still infeasible for use in VANETs. In order to remove the burden of certificates, many papers [62, 98, 99] proposed ID-based authentication scheme to enhance the computation and communication efficiency. However, these mechanisms are considered suitable only for private networks, because of the key escrow problem [7].

To solve the key escrow problem of ID-based signature scheme, the concept of certificaletless signature was firstly introduced by Al-Riyami and Paterson [68]. The key idea to handle the key escrow problem is that the key generation center in certificateless signature only provides a part of the user's full private key, of which another part comes from the user's own choice. Since then, many authentication schemes using certificateless signatures have been proposed to tackle the security and privacy problems in VANET [7, 70, 100–102].

Since the OBU only has limited computation capacity and the communication window of VANET is very short, participants in VANETs need to handle a large flow of messages. Hence, improving message authentication efficiency is critical. Various kinds of techniques or schemes have been proposed to improve the authentication efficiency in VANETs, for instance, certificateless signature, which avoids

the certificates management and key escrow problem, signature aggregation, batch verification, online/offline signature, signatures without using bilinear pairings or map-to-point hash functions, etc. In this chapter, I combine all these techniques to protect security and privacy as well as enhancing the authentication efficiency in VANETs. More specifically, I propose an efficient pairing-free online/offline certificateless aggregated signature scheme to address the information security and privacy issues for VANETs.

3.1.1 Related Works

The introduction of the first certificateless signature (CL-PKS) by Al-Riyami and Paterson [68] has inspired a large body of research work on improving the CL-PKS scheme. Yum and Lee [103] described a general method to construct a CL-PKS scheme from any ID-based signature scheme. Later, Li et al. [104] proposed the first CL-PKS scheme using bilinear pairings. Au et al. [105] presented a new security model for CL-PKS schemes, in which a malicious KGC attack is considered. He et al. [106] developed the first CL-PKS without using bilinear pairings. However, in [107, 108], the scheme in [106] is found to be insecure against a strong type II attack. More recently, Yeh et al.[109] proposed a CL-PKS scheme for IoT deployment. However, Jia et al. [110] pointed out that it has security flaws, as any malicious KGC can impersonate the KGC and it cannot resist a public key replacement attack.

In order to further reduce the computation and communication cost, which is crucial for resource-constrained scenario, aggregated signature for CL-PKC is proposed [111]. Signature aggregation means that given n signatures on n distinct messages from n distinct users, it is possible to aggregate all these signatures into a single short signature [112]. Most of the aggregated signature schemes require complex bilinear pairing operations, which are very expensive and are not suitable for lightweight devices, such as the OBU. In order to reduce the impact of bilinear pairing computations, Xiong et al. [113] proposed a certificateless aggregate signature scheme which only requires a small constant number of pairing operations. However, it is showed to be insecure in [114]. Apart from the aggregated signature, an online/offline signature is another approach to further decrease the computation cost. The first online/offline signature scheme was introduced by

Even, Goldreich and Micali [115]. But, the method is impractical since the size of the signature increases by a quadratic factor [116]. Recently, Cui et al.[70] proposed an efficient certificateless aggregated signature scheme without pairing for VANETs. However, Kamil et al. [102] found a security flaw in [70].

3.1.2 Overview of The Contributions of This Chapter

In this chapter, we propose an efficient online/offline certificateless signature scheme for VANET aiming to solve its security and privacy issues. The contributions are presented as follows:

- Firstly, we propose a certificateless online/offline signature scheme. Our scheme is efficient as it does not require the complex pairing operation and the map-to-point hash function. It also supports signature aggregation and batch verification, which can improve performance.
- Secondly, we present a rigorous security proof of our signature scheme. And we further perform a security analysis to show that the proposed scheme meets all the security and privacy requirements of VANETs.
- Thirdly, we analyse its computation efficiency, specifically the signing, verifying and aggregated verifying cost and make comparisons with some other similar schemes to show that the efficiency of our scheme is better than most of the other related schemes.

3.2 Preliminaries and Background

3.2.1 Elliptic Curve Cryptosystem and Assumptions

Below we briefly recap the fundamentals of elliptic curve cryptosystem.

Let F_p be a finite field, which is determined by a λ-bit prime number p. Let a set of elliptic curve points E over F_p be defined by the curve form: $y^2 = x^3 + ax + b$, where $p > 3$, $a, b \in F_p$, and $(4a^3 + 27b^2)$ mod p$\neq 0$, and the point at infinity be O. All the points on E including O form an additive group G with order q and generator P. The point addition '+' of element in cyclic group G is defined as follows: Let $P, Q \in G$, l be the line containing P, Q (tangent line to E if $P = Q$),

and R is the third point of the intersection of l and E. Let l' be the line connecting R and O. Then P '+' Q is defined as the third point such that l' intersects with E at R and O, which is -R. Scalar multiplication over E/F_p can be defined as follows:

$$mP{=}P{+}P{+}P{+}...{+}P \text{ (m times), where } m \in Z_q^*$$

The following complexity assumptions are used in security proof of the proposed scheme. We will use the Discrete Logarithm (DL) assumption and the Computational Diffie-Hellman (CDH) assumption over the additive cyclic group G, which can be defined as follows.

Definition 3.2.1 (The DL Assumption)**.** Discrete Logarithm (DL) Assumption: Given a random point $Q \in G$ on E, it is hard to compute an integer $x \in Z_q^*$ in polynomial time such that $Q = xP$ with non-negligible probability.

Definition 3.2.2 (The CDH Assumption)**.** Computational Diffie-Hellman (CDH) Assumption: Given two random points $Q, R \in G$ on E, where $Q = xP$, $R = yP$, $x, y \in Z_q^*$, it is hard to compute xyP in polynomial time with non-negligible probability.

3.2.2 System Model

Typically, a two-layer vehicular ad hoc network model is suitable for VANETs, as presented in prior research work [7, 62]. Figure 3.1 shows the typical architecture of VANETs. The lower layer is composed of vehicles and roadside units (RSUs) located at the critical points along the road. Each vehicle is equipped with an onboard unit (OBU), which enables vehicles to communicate with other vehicles or RSUs. There are three main types of communications in a vehicular ad hoc network, namely, V2V communication, V2I communication and hybrid way of communication. The communication among them is based on dedicated short-range communications (DSRC) protocol, which is identified as IEEE 802.11p. Each vehicle has a real identity, a number of pseudo identities, public/private key pairs.

The upper layer of VANETs consists of an application server(such as traffic control and analysis center), and key generation center (KGC) and trace authority (TRA). The TRA is responsible for RSU and vehicle registration by generating pseudo identities for them and can reveal the real identity of a vehicle from its

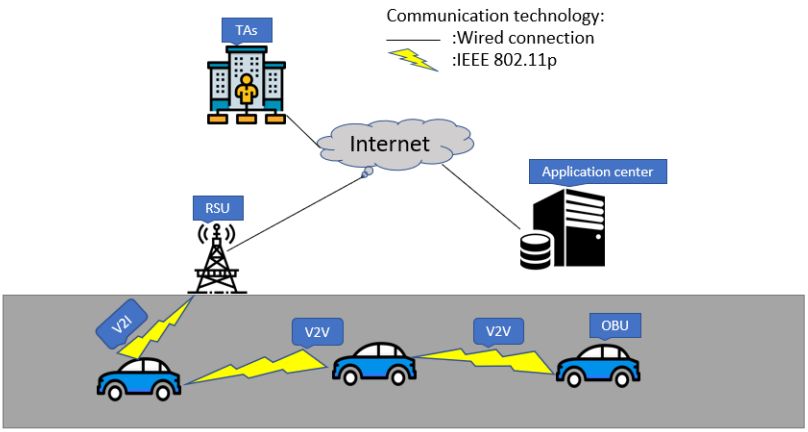


Figure 3.1: A typical architecture of VANETs

signed message. The KGC is in charge of generating public and private keys for RSU and vehicles. Besides, we assume the following hold:

1. The KGC and TRA are always trusted and cannot be compromised, which is usually assumed in VANETs scheme as in [62, 65]. The KGC and TRA have sufficient computation power and storage capacity. KGC and TRA are two separate authorities, which can communicate with each other securely using wired networks and secure protocols, such as the Transport Layer Security(TLS) protocol.

2. Each vehicle is equipped with a tamper-proof device, which can prevent the adversary from extracting data from the device. The OBU only has limited computation power, and RSU has greater computation power than OBU. The OBU and RSU are not trusted, and the message sent by them should be authenticated.

3.2.3 Security and Privacy Requirement

The proposed authentication scheme should satisfy the following security and privacy requirements.

1. **Identity Privacy Preserving**: RSUs, vehicles and any third participants cannot extract a vehicle's real identity from its pseudo identity and the transmitted message.

2. **Message Authentication and Integrity**: RSUs and vehicles should be able to check the validity of the signed message, and verify that the message is not modified during transmission.

3. **Traceability**: The TRA can reveal the vehicle's real identity from its pseudo identity and revoke its membership from VANETs in some cases. For example, TRA needs to reveal a malicious vehicle which sends a false message to mislead other vehicles.

4. **Resistance to Various Attacks**: The proposed authentication scheme can resist various possible attacks, such as the impersonation attack, modification attack, replay attack, the stolen verifier table attack.

3.2.4 Framework of the Signature Scheme

The proposed authentication scheme consists of the following eight phases: Setup, Pseudo-Identity-Generation, Partial-Private-Key-Extract, Vehicle-Key-Generation, Offline-Sign, Online-Sign, Individual-Verify, Aggregate-Verify.

Setup. In this phase, the TRA and KGC accept a security parameter and generate the master public/private key pair (mpk/msk) and the public system parameters ($params$).

Pseudo-Identity-Generation. In this phase, a vehicle and the TRA perform the registration process. The TRA receives the real identity of a vehicle and generate a pseudo identity (PID) using its private key, and assign the pseudo identity to the vehicle securely. Only the TRA has the ability to reveal the real identity of a vehicle.

Partial-Private-Key-Extraction. This process is run by a vehicle and the KGC, which receives the pseudo identity from a vehicle and generate a partial private key using its master secret key for the vehicle. Then the KGC delivers the partial private key to the vehicle through a secure channel.

Vehicle-Key-Generation. In this phase, the vehicle generates its own public/secret key (vpk, vsk).

Offline-Sign. In this phase, the vehicle takes the $params$ as input and generate the offline component of the certificateless signature using its partial private key and secret key. In this process, the vehicles pre-compute a set of tuples without

knowing the messages and store them in local storage for use in the online-sign phase. Each time the offline component used in the online-sign should be different.

Online-Sign. In this phase, given a traffic-related message, the offline signature component and the full private key, the vehicle generates the certificateless signature and broadcasts the message with the signature over the network.

Individual-Verify. In this phase, the RSUs or vehicles verify the certificateless signature by using public $params$, messages, the signer's pseudo identity and its full public key. If the signature is valid, the verifier outputs $true$, otherwise outputs $false$ and rejects the message.

Aggregate. In this phase, on receiving n different message and signatures pairs $\{m_i, \sigma_i\}$ from n different vehicles, the RSUs aggregate these different signatures into a single signature, and broadcast the aggregated signature to other participants in VANETs.

Aggregate Verify. In this phase, the verifier takes an aggregated signature, a list of the corresponding messages, pseudo identities, public keys, and $params$ as input, outputs $true$ if the certificateless aggregated signature is valid. This is also assumed to be performed by the RSUs or the application centers, such as a traffic control center in the system.

3.3 The Proposed Authentication Scheme

This section presents our proposed authentication scheme in detail, which is based on an efficient online/offline certificateless aggregate signature scheme. First, we define some notations that will be used in the scheme as listed in Table 3.1.

3.3.1 System Parameter Setup

In this phase, the TRA and KGC will generate the system parameters, such as a finite field, an elliptic curve, public keys, etc.

- Given a security parameter τ, the TAs will generate two large primes p and q, and will choose a non-singular elliptic curve E, which is defined by the equation $y^2 = x^3 + ax + b$, where $p > 3$, $a, b \in F_p$, and $(4a^3 + 27b^2)$ mod p$\neq$ 0.

Table 3.1: Notations and descriptions

Notation	Description
V_i	The i-th vehicle
OBU	An onboard unit
RSU	A trace authority
p, q	Two large prime numbers
E	An elliptic curve
G	An additive group with the order of q
P	A generator of group G
psk_i	A partial private key of vehicle V_i
x_{ID_i}	A secret key of vehicle V_i
vpk_{ID_i}	A public key of vehicle V_i
(P_{pub}, α)	The public/private key pair of KGC
(T_{pub}, β)	The public/private key pair of TRA
RID_i	The real identity of a vehicle V_i
PID_i	The pseudo identity of a vehicle V_i
H_1, H_2, H_3	Secure hash functions
T_i	A valid period of the pseudo identity
t_i	A current timestamp
m_i	A traffic-related message
$\oplus$	The exclusive **OR** operation
$\|\|$	The message concatenation operation

- The TAs will choose a generator P of the additive group G with the order of q. And it will also choose three secure hash functions which are H_1: $G \times \{0,1\}^* \times \{0,1\}^* \rightarrow Z_q^*$, H_2: $\{0,1\}^* \times G \rightarrow Z_q^*$, H_3: $\{0,1\}^* \times \{0,1\}^* \times G \times G \times \{0,1\}^* \rightarrow Z_q^*$.

- The TRA will randomly choose number $\beta \in Z_q^*$ as its master private key for traceability, and compute $T_{pub} = \beta \cdot P$ as its public key.

- The KGC will randomly choose number $\alpha \in Z_q^*$ as its master private key for partial private key extraction, and compute $P_{pub} = \alpha \cdot P$ as its public key.

- Then, the public parameters are $params$=$\{P, p, q, E, G, H_1, H_2, H_3, P_{pub}, T_{pub}\}$. Finally, each vehicle pre-loads the public parameters into its temper-proof device and RSU stores $params$ into its local storage.

3.3.2 Pseudo-Identity-Generation and Partial-Private-Key-Extraction

In this phase, a vehicle registers with the TRA and KGC to obtain its pseudo identity and partial private key.

- The vehicle chooses a random value $k_i \in Z_q^*$, and calculates $PID_{i,1}$=k_iP. Then the vehicle sends its real identity RID_i and $PID_{i,1}$ to the TRA in a secure way.

- Once the TRA receives $(RID_i, PID_{i,1})$ from the vehicle, it first checks whether RID_i is valid or not. If RID_i exists in its local database, then TRA computes $PID_{i,2} = RID_i \oplus H_1((\beta \cdot PID_{i,1})||T_i||T_{pub})$ and sends the $PID_{i,2}$ to the vehicle. Then, the pseudo identity of the vehicle is PID_i=$(PID_{i,1}, PID_{i,2}, T_i)$ where T_i is the valid period of the pseudo identity.

- A vehicle will use its pseudo identity PID_i to communicate with other participants in the VANET. Since only TRA knows its master private key β, it has the ability to reveal the real identity of a vehicle by computing $RID_i = PID_{i,2} \oplus H_1((\beta \cdot PID_{i,1})||T_i||T_{pub})$ in some situation. Then, the TRA will also send the pseudo identity PID_i to KGC in a secure way.

- After the KGC receives the pseudo identity, it chooses a random number $d_i \in Z_q^*$ and compute Q_{ID_i}=d_iP. Then it calculates the partial private key as $psk_{ID_i} = d_i + H_2(PID_i||Q_{ID_i}) \cdot \alpha \pmod q$.

- Then the KGC transmits (Q_{ID_i}, psk_{ID_i}) to the vehicle via a secure channel. Finally the vehicle obtains its pseudo identity PID_i and partial private key psk_{ID_i}. And the vehicle can check the validity of the partial private key using the public parameters by verifying whether the equation $psk_{ID_i} \cdot P$=Q_{ID_i} + $H_2(PID_i||Q_{ID_i}) \cdot P_{pub}$ holds or not. If it holds, then the vehicle will store the pseudo identity (PID_i) and partial private key(psk_{ID_i}) in its temper-proof device for further use. Note that the value Q_{ID_i} should be public.

3.3.3 Vehicle-Key-Generation

In this phase, the vehicle chooses a random number $x_{ID_i} \in Z_q^*$ as its secret key and compute $vpk_{ID_i} = x_{ID_i} \cdot P$ as its public key.

3.3.4 Offline-Sign

In order to maintain the message authentication and integrity, the traffic-related message should be signed before transmitted. Since the computation power of the OBU is limited, we propose to use the online-offline signature technique, which allows the vehicles to offline compute some part of the signature when OBU is idle or the traffic density is not high, to enhance the efficiency of generating signatures. The offline signature is generated as follows:

- V_i randomly selects a number $r_i \in Z_q^*$

- V_i computes $R_i = r_i \cdot P$

- V_i stores the offline $\phi_i = (r_i, R_i)$ locally

Generating offline signature does not require the message, thus a large set of these offline signature pairs could be pre-generated and stored locally for future use.

3.3.5 Online-Sign

Firstly, it randomly picks a pseudo identity PID_i from its storage and selects the latest timestamp t_i, which is used to prevent the replay message attacks. On input a traffic-related message m_i, it signs the message as the followings steps.

- V_i obtains a fresh offline signature tuple $\phi_i = (r_i, R_i)$ from its storage.
- V_i computes the full private key $sk_i = x_{ID_i} + psk_{ID_i}$
- V_i computes $h_{3i} = H_3(m_i||PID_i||vpk_{ID_i}||R_i||t_i)$.
- V_i computes $s_i = h_{3i} \cdot r_i + sk_i \ (mod\ q)$
- The output signature is $\sigma_i = (R_i, s_i)$. Finally, the vehicle V_i broadcasts $\{m_i, PID_i, \sigma_i, t_i, vpk_{ID_i}, Q_{ID_i}\}$ to nearby RSUs and vehicles for verification.

3.3.6 Individual-Verify

In this phase, RSUs or vehicles verify the validity of an individual received message. Once it receives the message $\{m_i, PID_i, \sigma_i, t_i, vpk_{ID_i}, Q_{ID_i}\}$, the verifier checks the validity of the signature as follows.

- Firstly, the verifier will check the freshness of the timestamp t_i. If it is not fresh, then the verifier reject the message and stop the verifying process.
- Then, the verifier calculates $h_{3i} = H_3(m_i||PID_i||vpk_{ID_i}||R_i||t_i)$ and $h_{2i} = H_2(PID_i||Q_{ID_i})$
- Then, it checks whether the equation $s_i \cdot P = h_{3i} \cdot R_i + vpk_{ID_i} + Q_{ID_i} + h_{2i} \cdot P_{pub}$ holds or not. If this equation holds, then the verifier accepts this message, otherwise rejects this message.

Proof of Correctness:

Since $h_{3i} = H_3(m_i||PID_i||vpk_{ID_i}||R_i||t_i)$, $h_{2i} = H_2(PID_i||Q_{ID_i})$, $sk_i = x_{ID_i} + psk_{ID_i}$, $r_i \cdot P = R_i$, $x_{ID_i} \cdot P = vpk_{ID_i}$, and $psk_{ID_i} \cdot P = Q_{ID_i} + h_{2i} \cdot P_{pub}$, if the signature is generated correctly, then the following equation will hold

$$\begin{aligned} s_i \cdot P &= h_{3i} \cdot r_i \cdot P + x_{ID_i} \cdot P + psk_{ID_i} \cdot P \\ &= h_{3i} \cdot R_i + vpk_{ID_i} + Q_{ID_i} + h_{2i} \cdot P_{pub} \end{aligned}$$

3.3.7 Aggregate

In some scenarios where the density of transmitted messages is very high, RSUs need to aid the communication by aggregating a collection of certificateless signatures into one. Signature aggregation is the process that on receiving a set of messages $\{m_i, PID_i, \sigma_i, t_i, vpk_{ID_i}, Q_{ID_i}\}$ from n vehicles $\{V_i, V_2, ..., V_n\}$, where $i = 1, 2, 3, ...n$, the RSU aggregates the signature by calculating $S = \sum_{i=1}^{n} s_i$. Then RSUs output $\sigma = (R_1, R_2, R_3...R_n, S)$ as the aggregated signature.

3.3.8 Aggregate-Verify

This algorithm is assumed to be performed by RSUs or the application centers, such as a traffic control center. Once receiving the aggregated signature $\sigma = (R_1, R_2, R_3...R_n, S)$ from a set of vehicles $\{V_1, V_2, V_3, ..., V_n\}$, with the corresponding parameters $\{m_i, PID_i, t_i, vpk_{ID_i}, Q_{ID_i}\}$, where $i = 1, 2, 3, ...n$, the RSUs or application centers check the validity of the aggregated signature by performing the following steps.

- Firstly, the verifier will check the freshness of the timestamp t_i, for $i = 1, 2, 3, ...n$. If it is not fresh, then the verifier rejects the message and stops the verifying process.
- Calculate $h_{3i} = H_3(m_i||PID_i||vpk_{ID_i}||R_i||t_i)$ and $h_{2i} = H_2(PID_i||Q_{ID_i})$, for $i = 1, 2, 3, ...n$
- Check whether the following equation holds or not: $S \cdot P = \sum_{i=1}^{n}(h_{3i} \cdot R_i) + \sum_{i=1}^{n} Q_{ID_i} + \sum_{i=1}^{n} vpk_{ID_i} + (\sum_{i=1}^{n} h_{2i}) \cdot P_{pub}$. If this equation holds, the verifier will accept the aggregated signature.

Proof of Correctness:

Since we have $h_{3i} = H_3(m_i||PID_i||vpk_{ID_i}||R_i||t_i)$, $h_{2i} = H_2(PID_i||Q_{ID_i})$, $sk_i = x_{ID_i} + psk_{ID_i}$, $r_i \cdot P = R_i$, $x_{ID_i} \cdot P = vpk_{ID_i}$, and $psk_{ID_i} \cdot P = Q_{ID_i} + h_{2i} \cdot P_{pub}$, then we can check the correctness as follows:

$$
\begin{aligned}
S \cdot P &= \sum_{i=1}^{n} s_i \cdot P \\
&= \sum_{i=1}^{n}(h_{3i} \cdot r_i \cdot P + x_{ID_i} \cdot P + psk_{ID_i} \cdot P) \\
&= \sum_{i=1}^{n}(h_{3i} \cdot R_i) + \sum_{i=1}^{n} Q_{ID_i} + \sum_{i=1}^{n} vpk_{ID_i} + (\sum_{i=1}^{n} h_{2i}) \cdot P_{pub}
\end{aligned}
$$

3.3.9 Batch Verification

Sometimes, a participant in VANETs needs to verify multiple signatures in a single instance instead of aggregating them. In this scenario, we need to use the batch verification technique, which allows multiple signatures to be verified at a time. To ensure the non-repudiation of signatures using batch verification, we use the small exponent test technology [99]. On receiving multiple messages $\{m_i, PID_i, \sigma_i, t_i, vpk_{ID_i}, Q_{ID_i}\}$ where $i = 1, 2, 3, ...n$, the verifier checks the signature validity using public parameters. The verification process is shown as follows.

- Firstly, the verifier will check the freshness of the timestamp t_i, for $i = 1, 2, 3, ...n$. If it is not fresh, then the verifier rejects the message and stops the verifying process.
- The verifier randomly chooses a vector v=$\{v_1, v_2, v_3, ..., v_n\}$,, where v_i is a small random integer in $[1, 2^t]$ and t is a small integer that incurs very little computation head.
- The verifier checks whether the following equation holds, if it holds, it accepts the messages, otherwise rejects the messages.

  $(\sum_{i=1}^{n} s_i \cdot v_i) \cdot P$=$\sum_{i=1}^{n}(h_{3i} \cdot R_i \cdot v_i) + \sum_{i=1}^{n}(vpk_{ID_i} \cdot v_i) + \sum_{i=1}^{n}(Q_{ID_i} \cdot v_i) + (\sum_{i=1}^{n} h_{2i} \cdot v_i) \cdot P_{pub}$

Proof of Correctness: The process is similar to that in the aggregated verify. We have $h_{3i} = H_3(m_i||PID_i||vpk_{ID_i}||R_i||t_i)$, $h_{2i} = H_2(PID_i||Q_{ID_i})$, $sk_i = x_{ID_i} + psk_{ID_i}$, $r_i \cdot P = R_i$, $x_{ID_i} \cdot P = vpk_{ID_i}$, and $psk_{ID_i} \cdot P = Q_{ID_i} + h_2 \cdot P_{pub}$. We obtain that:

$(\sum_{i=1}^{n} s_i \cdot v_i) \cdot P$
$=\sum_{i=1}^{n}((h_{3i} \cdot r_i + x_{ID_i} + psk_{ID_i}) \cdot v_i) \cdot P$
$=\sum_{i=1}^{n}(h_{3i} \cdot v_i \cdot r_i \cdot P)+\sum_{i=1}^{n}(v_i \cdot x_{ID_i} \cdot P) +\sum_{i=1}^{n}(v_i \cdot psk_{ID_i} \cdot P)$
$=\sum_{i=1}^{n}(h_{3i} \cdot R_i \cdot v_i)+\sum_{i=1}^{n}(psk_{ID_i} \cdot v_i)+ \sum_{i=1}^{n}((Q_{ID_i} + h_{2i} \cdot P_{pub}) \cdot v_i)$
$=\sum_{i=1}^{n}(h_{3i} \cdot R_i \cdot v_i) + \sum_{i=1}^{n}(vpk_{ID_i} \cdot v_i) + \sum_{i=1}^{n}(Q_{ID_i} \cdot v_i) + (\sum_{i=1}^{n} h_{2i} \cdot v_i) \cdot P_{pub}$

3.4 Security Proof

In this section, we give a formal security proof on the proposed certificateless signature scheme. We use a similar approach in [106] to prove the security of the proposed signature scheme. The security proof shows that the proposed signature scheme is secure against public key replacement attack and the malicious-but-passive KGC attack. The detailed security proof is shown in Appendix A.

3.5 Discussion

In this section, we first present the security and privacy analysis with respect to the identity privacy-preserving, message authentication, and integrity, traceability, unlinkability and resistance to various attacks. Then we analyze the performance of the proposed online/offline certificateless signature scheme and compare with some other similar schemes.

3.5.1 Security Analysis

1. **Identity Privacy Preserving**: Each participant in VANETs needs to register with the TRA to obtain a pseudo identity, which is generated by the TRA using its master private key β. The only way for an adversary to reveal the real identity is to compute $RID_i = PID_{i,2} \oplus H_1((\beta \cdot PID_{i,1})||T_i||T_{pub})$, which means that the adversary has to know the master private key β to calculate $\beta \cdot PID_{i,1}$. However, it is infeasible for the adversary to obtain β from $T_{pub} = \beta \cdot P$, as this contradicts the DL assumption. Therefore, our scheme meets the requirement of identity privacy preserving.

2. **Message Authentication and Integrity**: Each transmitted message is signed by a legitimate user before broadcasting in VANETs. According to Theorem 3 and Theorem 4 in Appendix A, there is no polynomial-time adversary can forge a valid signature based on the DL assumption. Hence the verifier can check the validity and integrity of the signature, which guarantees that the message comes from a legitimate user and it is not modified during transmission, by verifying the equation $s_i \cdot P = h_{3i} \cdot R_i + vpk_{ID_i} + Q_{ID_i} +$

$h_{2i} \cdot P_{pub}$. Hence, the proposed scheme ensures the message authentication and integrity.

3. **Traceability**: The pseudo identity is generated using the master private key of the TRA. From the pseudo identity PID_i=$(PID_{i,1}, PID_{i,2}, T_i)$, where $PID_{i,1}$=k_iP, $PID_{i,2} = RID_i \oplus H_1((\beta \cdot PID_{i,1})||T_i||T_{pub})$, the TRA can extract the real identity by computing $RID_i = PID_{i,2} \oplus H_1((\beta \cdot PID_{i,1})||T_i||T_{pub})$. Hence, the traceability is also provided by our scheme.

4. **Resistance to Various Attacks**: In this part, we show that our scheme can resist various attacks, including reply attack, modification attack, impersonation attack and stolen verifier table attack.

    - **Reply Attack**: Before verifying the validity of the signature, the verifier will check the freshness of the timestamp t_i. If it is not a fresh timestamp, the message will be rejected. Hence, the reply attack is avoided in our scheme by using the timestamp.
    - **Message Modification Attack**: Due to the message integrity ensured by the signature scheme, any modification of the message will lead to the result that equation $s_i \cdot P = h_{3i} \cdot R_i + vpk_{ID_i} + Q_{ID_i} + h_{2i} \cdot P_{pub}$ does not hold when the verifier checks the validity of the signature. Then the modified message will be disregarded. Hence, our scheme can resist modification attack.
    - **Impersonation Attack:** In order to launch a successful impersonation attack, an attacker should output a message $\{m_i, PID_i, \sigma_i, t_i, vpk_{ID_i}, Q_{ID_i}\}$ that can pass the verification of the receiver. This means that the adversary should be able to forge a valid signature. However, this is infeasible according to the Theorem 3 and Theorem 4 in Appendix A. Hence the impersonation attack is impossible for our scheme.
    - **Stolen Verifier Table Attack:** In our scheme, OBU and RSU do not maintain a verifier table for message authentication. Therefore, stolen verifier table attack is also impossible for our scheme.

3.5.2 Performance Evaluation

In this section, we will analyze the computation performance of our scheme in terms of signing cost, verification cost, and make comparisons with some other related schemes, for instance, certificateless signature schemes that require bilinear pairing. We adopt a similar approach in [117] to analyze the performance. Below we define the benchmark and security level for comparisons.

For bilinear pairing-based authentication schemes, we use a bilinear pairing $\bar{e} : G_1 \times G_1 \rightarrow G_2$ with the security level of 80-bits, where G_1 is an additive group generated by a point $\bar{P}$ with the order of $\bar{q}$ on the super singular elliptic curve $\bar{E} : y^2 = x^3 + x \bmod \bar{p}$ with the embedding group degree 2, $\bar{p}$ is a 512-bit prime number, $\bar{q}$ is a 160-bit Solinas prime number and the equation $\bar{p} + 1 = 12\bar{q}\bar{r}$ holds. For ECC-based identity-based authentication scheme, we achieve the security level of 80-bits by using an additive group G generated by a point P with the order q on a non-singular elliptic curve E, which is defined by the equation $y^2 = x^3 + ax + b$, where $p > 3$, $a, b \in F_p$, p, q are 160-bit prime number, and $(4a^3 + 27b^2) \bmod \text{p} \neq 0$.

3.5.3 Computation Cost Analysis

We first define some notations about the execution time of the cryptographic operations. The execution time is evaluated using the famous MIRACL cryptographic library. The execution time of the cryptographic operations is given in Table 3.2, where the timing is from [117]. Note that some very light operations, such as addition operation in Z_q^* and multiplication operation in Z_q^* are ignored, as the execution time is relatively small.

•T_{bp}: The operation time of a bilinear pairing operation $\bar{e}(P, Q)$, where $\bar{P}, \bar{Q} \in G_1$;

•T_{bp-m}: The operation time of a scalar multiplication $x \cdot \bar{P}$ related to a bilinear pairing, where $\bar{P} \in G_1, x \in Z_{\bar{q}}^*$;

•T_{bp-a}: The operation time of a point addition $\bar{P} + \bar{Q}$ related to a bilinear pairing, where $\bar{P}, \bar{Q} \in G_1$;

•T_{ecc-m}: The operation time of a scalar multiplication $x \cdot P$ related to the ECC, where $P \in G$ and $x \in Z_q^*$;

•T_{ecc-a}: The operation time of a point addition $P + Q$ related to the ECC, where $P, Q \in G$;

Table 3.2: Execution time of different cryptographic operations

Cryptographic operation	Execution time (ms)
T_{bp}	4.2110
T_{bp-m}	1.7090
T_{bp-a}	0.0071
T_{ecc-m}	0.4420
T_{ecc-a}	0.0018
T_H	4.406
T_h	0.0001

Table 3.3: Computation cost comparisons of the proposed scheme with others

Schemes	Sign(ms)	Individual Verify(ms)	Total(ms)
[118]	$4T_{bp-m} + 2T_{bp-a} + T_h \approx 6.8503$	$3T_{bp} + 3T_{bp-m} + T_{bp-a} + 2T_h \approx 17.7673$	24.6176
[7]	$2T_{bp-m} + T_{bp-a} + T_h \approx 3.4252$	$3T_{bp} + T_{bp-m} + T_{bp-a} + T_H + T_h \approx 18.7552$	22.1804
[100]	$3T_{bp-m} \approx 5.127$	$3T_{bp} + 2T_H + 2T_{bp-m} \approx 24.863$	29.99
[101]	$3T_{bp-m} \approx 5.127$	$3T_{bp} + T_H + 2T_{bp-m} \approx 20.457$	25.584
[70]	$T_{ecc-m} + T_h + T_{ecc-a} \approx 0.4439$	$3T_{ecc-m} + 2T_{ecc-a} + 2T_h \approx 1.3298$	1.7737
[102]	$3T_{ecc-m} + 3T_h + 2T_{ecc-a} \approx 1.3299$	$2T_{ecc-m} + T_{ecc-a} + T_h \approx 0.8859$	2.2158
Our scheme	$T_{ecc-m} + T_h \approx 0.4421$	$3T_{ecc-m} + 3T_{ecc-a} + 2T_h \approx 1.3316$	1.7737

•T_H: The execution time of a map-to-point hash function operation;

•T_h: The execution time of an ordinary one-way hash function operation.

Typically, the operation of bilinear pairing is much more costly than that of the ECC, and a map-to-point hash operation is also expensive than an ordinary one-way hash function. Hence, the singing phase that does not require pairing operation and map-to-point hash function has higher computation efficiency.

We make comparisons with the recent authentication schemes in VANETs [7, 70, 100–102, 118]. The comparisons of computation cost of signing, verifying one message and aggregated verify are given in Table 3.3 and Table 3.4.

From Table 3.3 and Table 3.4, it is obvious to see that schemes[7, 100, 101, 118] with pairing operation and map-to-point hash functions are much more compu-

Table 3.4: Computation cost comparisons of the proposed scheme with others

Schemes	Aggregated Verify(ms)
[118]	$3T_{bp} + 3nT_{bp-m} + nT_{bp-a} + 2nT_h$
[7]	$3T_{bp} + nT_{bp-m} + nT_{bp-a} + nT_H + nT_h$
[100]	$3T_{bp} + (n+1)T_H + 2nT_{bp-m}$
[101]	$3T_{bp} + nT_H + 2nT_{bp-m}$
[70]	$(n+2)T_{ecc-m} + 2nT_{ecc-a} + 2nT_h$
[102]	$2T_{ecc-m} + nT_{ecc-a} + nT_h$
Our scheme	$(n+2)T_{ecc-m} + 3nT_{ecc-a} + 2nT_h$

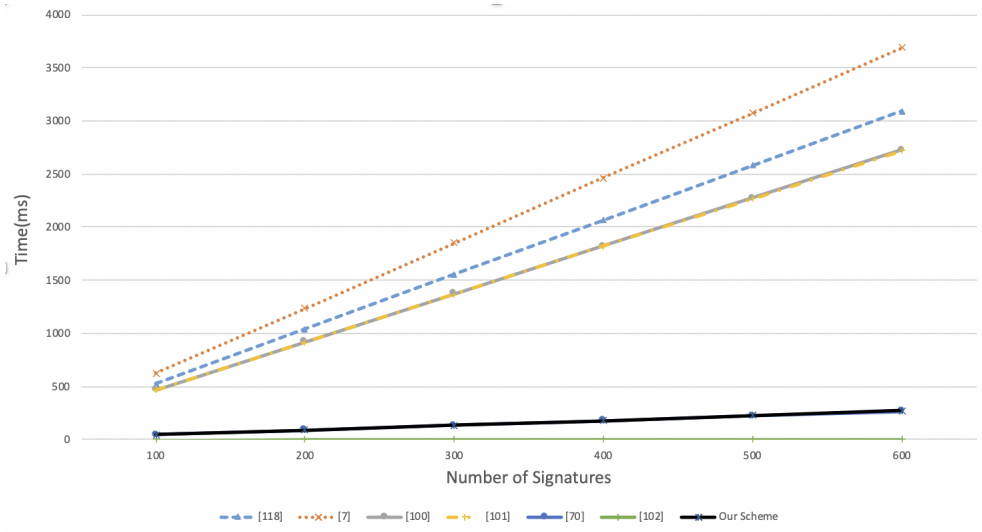


Figure 3.2: Aggregated verification time vs. Number of signatures

tationally expensive than schemes based on ECC cryptographic primitives and simple one-way hash functions. Then, comparing to similar schemes [70, 102], which also does not require pairing and map-to-point hash function, our scheme also has some advantages. Even though [70] almost has the same computation efficiency as our scheme, it is shown to be insecure under the existing security model in [102]. Kamil et al. [102] proposed an improved scheme after its cryptanalysis of Cui's scheme [70]. Although the individual verifying phase of our scheme is more expensive than that in [102], the signing cost of our scheme is much lower than that in [102]. And note that, the total cost of signing and verifying a single message is also smaller than that in [102]. More importantly, our scheme supports online/offline sign, which means that some cryptographic operations can be pre-computed and used directly when signing a message. Hence in our scheme, the signing cost could be lower and only be T_h, as the operation of the relatively expensive scalar multiplication corresponding to T_{ecc-m} can be pre-computed and does not incur computation overhead.

In Figure 3.2, we investigate the aggregated verification time with respect to the number of signatures. Figure 3.2 indicates that the aggregated verification time with regards to the number of signatures of the schemes, which require bilinear pairings and map-to-point hash functions, increases much faster than that of the schemes without pairings and map-to-point hash functions. The aggregated verification time with regards to the number of signatures of our scheme grows relatively a little faster than that of [102]. However, we argue that typically an RSU is assumed to have more computation power than the OBU. Hence, in many scenarios, the need to enhance the signing efficiency is more significant than the need to improve the aggregated verification efficiency, which means that the advantage of an efficient sign phase outweighs the advantage of an efficient aggregated verification phase. Therefore, our scheme has a slight edge comparing to the scheme [102] in the sense that the signing efficiency is higher than that in [102].

3.6 Chapter Summary

In this chapter, we propose an efficient privacy-preserving authentication scheme using online/offline certificateless aggregate signature to address the security and privacy issues of VANETs. The proposed scheme is proven to be secure with a

rigorous security proof, and it satisfies all the security and privacy requirements of VANETs. The online/offline signature allows some computational expensive operations to be pre-computed offline, thus reducing the computation overhead when signing a message online. Moreover, the proposed scheme does not require the computational expensive bilinear pairing operation and map-to-point hash function, and it supports the techniques of signature aggregation and batch verification, which are very useful for VANETs scenario. As a result of using these techniques, the proposed scheme offers a better computation efficiency compared with many other related schemes.

Chapter 4

Efficient Revocation based on a Revocable Certificateless Signature Scheme

In this chapter, we focus on addressing the efficient revocation problem of the authentication scheme based on a certificateless signature, improving the overall authentication efficiency, and enhancing the revocation transparency. More specifically, the efficient revocation is realized by using a revocable certificateless signature scheme. To handle revocation, the key generation center will periodically update the time keys of the non-revoked users. To reduce the workload of the key generation center, we apply a node selection algorithm known as KUNodes. Moreover, in order to enhance the transparency of revocation, the blockchain technology is employed to store the revocation list, so that the vehicles can check the state of revocation list from the immutable blockchain. In terms of improving the authentication efficiency, we propose the RSU-assisted authentication process, where roadside unit assists nearby vehicles to verify signatures through the use of cuckoo filter.

4.1 Introduction

Certificateless signature schemes have been identified to address the security and privacy issues of vehicular networks in various works [7, 70, 100, 102]. The on-

line/offline certificateless signature scheme proposed in Chapter 3 has the benefit of efficient signature generation and verification. However, like many other certificateless signatures schemes, it lacks an efficient and practical revocation mechanism. The existing revocation methods either require an additional online security mediator (SEM) to update the private key, which implies the need for a secure channel, or requires the key generation center (KGC) to conduct computation linear in the number of users in the system which is not scalable.

To tackle this problem, the revocation of our proposed scheme is realized by a time key mechanism. Specifically, the full private key of the certificateless signature scheme consists of an initial partial private key and a time key from the KGC, together with a secret value selected by the user. The time key is generated and updated by the KGC and is broadcasted over the public channel, while the partial private key remains unchanged. Hence, no additional security mediator or secure channel is required. In many other authentication schemes, such as PKI-based schemes, a revocation list is used to record the revoked user and the corresponding certificates or pseudonyms. The revocation list is public and updated by the certificate authority periodically. And, every verifier should locally check the revocation state of the certificate against the revocation list before checking the corresponding signature. This revocation checking process incurs delays to the authentication process. Compared with the conventional revocation approaches, the proposed revocation mechanism in the certificateless signature scheme has the advantage that the verifier does not need to perform revocation check before verifying the signature. The first reason is that no certificate is needed in certificateless signature scheme. Another reason is that, in the PKI-based schemes, a revoked user still has the private key to produce a valid signature, which can be verified successfully if the verifier has not checked the revocation list. However, this problem is avoided, as the revoked user will not receive the time key from the KGC, and cannot generate a valid signature.

A problem of the proposed revocation approach is that the number of revocation updates performed by KGC grows linearly with the number of the non-revoked users, which means that the workload of KGC may become the bottleneck if the number users become very large. In order to solve this problem, we employ the well-known node selection algorithm, KUNodes algorithm, for the key update. Hence, the resulting revocation complexity of the KGC depends logarithmically

on the number of non-revoked users in the vehicular network. Another issue is that the revocation of all the users is controlled by the KGC, which indicates that the revocation transparency is not ensured. For example, occasionally, a system error could happen and the KGC may unintentionally revoke a user that should not be revoked. In this case, the revoked user should be able to check the state of the revocation list from a trustworthy source to resolve this dispute. Hence, in order to enhance the revocation transparency, the blockchain technology is used to record the state of the revocation list for inspection.

Apart from solving the revocation issues, an RSU-assisted authentication process is proposed to enhance the overall efficiency of message authentication in vehicular networks. More specifically, the proposed scheme supports batch verification, by which an RSU can verify multiple signatures simultaneously. And, after batch verification, the RSU generates a notification message to assist the signature verification of the nearby vehicles using the cuckoo filter [119]. Therefore, the proposed online/offline revocable certificateless signature authentication scheme is efficient not only in terms of the signature generation and verification but also in terms of user revocation.

This chapter is organized as follows. Firstly, we introduce the related works and contributions of this chapter. In the next section, we present basic preliminaries. Then, we show the message authentication scheme in section 3, and the RSU-assisted verification process in section 4. Afterwards, we perform the security analysis of the proposed scheme in section 5. Lastly, we discuss performance issues and draw conclusions.

4.1.1 Related Works

Certificateless signature was first proposed in [68] to eliminate the key escrow problem in identity-based signatures. Many improvements and new constructions were proposed since then. Au et al. [105] proposed a new security model to consider the case when the KGC could maliciously generate the public parameters. He et al. [106] developed the first pairing-free construction. Jia et al.[110] proposed to use certificateless signature scheme to address the security and privacy problems for the Internet-of-Things.

Revocation in certificateless cryptography has been widely studied. Existing

schemes with revocation fall into one of the following approaches. The first one is to rely on an online security mediator (SEM) [120]. However, the SEM has to maintain private information for each user and could be the target of the attacker. Another approach is to add a time key to the private key of a user [121]. The time key is valid for a specific time interval and has to be updated periodically with the help of the KGC. The KGC can revoke user by refusing to update the time key for the revoked users. Another approach is to maintain a revocation list [122]. Finally, some also consider outsourcing the expensive process to the cloud [123].

Since the number of signatures to be verified is possibly high, batch verification has also been widely investigated in the vehicular networks [117]. Also, some considered using bloom filter for set-membership check in the batch verification to enhance the overall message authentication efficiency [124–126]. For example, [126] proposed to use two bloom filters to check the MAC address of the pseudonyms. Recently, cuckoo filter is also introduced to message authentication in vehicular networks [127], albeit in the identity-based setting which inherits the key escrow problem as stated in [128].

4.1.2 Overview of The Contributions of This Chapter

Based on the online/offline certificateless signature scheme, which is presented in the previous chapter, we add the functionalities of efficient user revocation using the KUNodes algorithm and enhanced transparency of revocation of the trusted authorities using the blockchain technology. The contributions of this chapter are summarised as follows. Firstly, we propose an efficient revocable pairing-free certificateless online/offline signature scheme for vehicular networks. The complexity of our revocation mechanism is logarithmic to the number of users and can be conducted through a public channel. We also propose to use blockchain to improve the revocation transparency of the KGC. Secondly, the proposed system supports batch verification and allows the RSUs to assist message authentication of nearby vehicles through the use of cuckoo filter. Thirdly, we conduct security and efficiency analysis of our proposed revocable signature scheme and compare it with existing proposals in terms of efficiency. Finally, we analyze the false positive of the cuckoo filter and conduct an experiment to evaluate its computation cost.

4.2 Background and Preliminaries

4.2.1 System Model and Blockchain

Typically, a vehicular network consists of two layers. The upper layer includes application centers, trusted authorities (TAs), and roadside units (RSUs), and the lower layer consists of vehicles and RSUs. Trusted authorities, such as the key generation centers (KGCs) are responsible for user registration and revocation. RSUs are located at the critical points of roads to assist the message exchange of the vehicles. Each vehicle is equipped with an onboard unit (OBU) to process messages for vehicles. This paper mainly considers the two basic communication types, namely, Vehicle-to-Vehicle (V2V) communication and Vehicle-to-Infrastructure (V2I) communication, which are realized by Dedicated Short Range Communication (DSRC) radio technology specifically developed to facilitate the communications of vehicular networks.

A blockchain is a distributed, immutable, append-only ledger which maintains a growing list of data blocks. Each block contains transaction data, and is securely connected to the previous block using a hash pointer. Blocks are validated by all nodes in the blockchain network before being added to the blockchain. This process is governed by the consensus protocol of the blockchain network. Blockchain can be classified according to its access control model. In a public blockchain, anyone can access the blockchain and participate in the consensus process. On the other hand, in a permissioned blockchain, participation is subject to access control. While public blockchains such as Bitcoin often rely on proof-of-work consensus, permissioned blockchains can make use of some kind of Byzantine fault tolerant protocol or crash fault tolerant protocol for more efficient consensus. Looking ahead, we propose to use a permissioned blockchain, such as Hyperledger Fabric, to record the activities of the KGC. It has a much higher transaction throughput and is thus more practical to be used in vehicular networks. For instance, Hyperledger fabric could achieve a transaction throughput of over 2000 transactions/second, while Ethereum (a public blockchain) can only support 10-30 transactions/second. Figure 4.1 shows the proposed architecture of our system. In this paper, we make the following assumptions.

1. The KGC is a trusted entity and cannot be compromised. It has high compu-

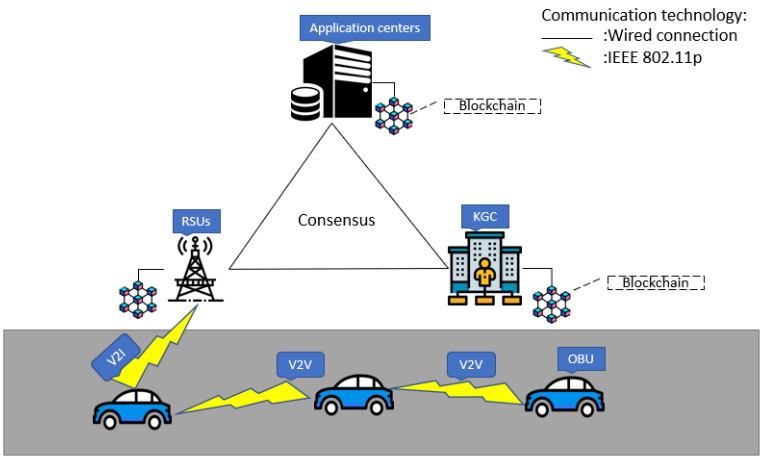


Figure 4.1: The architecture of a vehicular network with blockchain

tation power and large storage capacity. The communication between KGC and RSUs is secured and efficient. The RSUs are also trusted. They have a much larger communication range and higher computation capability than that of vehicles.

2. The permissioned blockchain has two kinds of nodes, namely, validators and clients. Validators are full nodes that are played by the application centers, the KGC and the RSUs. They participate in the consensus process and propose new blocks. Clients (i.e. vehicles in our system) have read access to the blockchain and can send transactions to the blockchain through the RSUs. They do not maintain the blockchain in their local storage nor take part in the consensus process. The blockchain system ensures the correctness, integrity, and trust of the data.

3. The blockchain is an append-only ledger offering three following basic functions:

   - **Initialize**: The KGC prepares an empty genesis block (i.e., the first block) and initializes the blockchain L with this empty block.
   - **Store**: Validator P_i stores data into the ledger by submitting $(Store, P_i, Msg_i)$ to the blockchain network. All the validators perform consensus algorithm and store the tuple (P_i, Msg_i) to its local copy of the blockchain.

- **Retrieve and View**: As a full node of the blockchain, any validator can retrieve data from the blockchain easily from its local storage. For a client, such as a vehicle, it can view blockchain data by submitting $(View, PID_i)$, where PID_i is the pseudo identity of the vehicle, to any validator, such as an RSU. Then this validator will send the blockchain data to the vehicle.

4.2.2 Cuckoo Filter

Recently, a new index data structure called cuckoo filter [119] was proposed as an improvement to the conventional bloom filter. Compared to bloom filter, cuckoo filter has the advantages of supporting dynamic addition and deletion. It is constructed based on cuckoo hash tables, which store fingerprints of the items. The basic set membership check for item x involves computing its fingerprint and searching for it in the cuckoo hash table. If the fingerprint is found, item x exists in the hash table. The basic structure of cuckoo hash table and cuckoo filter are shown in Figure 4.2 For the insertion operation of item x (see Figure 4.2(a)), two candidate buckets for the item are determined by computing two hash values of x. If one of the buckets is empty, item x is stored in this empty bucket. However, if both candidate buckets are filled, one candidate bucket will be chosen to store the new item. The original item, say a, being kicked out, will be re-inserted into its another candidate bucket (e.g, bucket 4) following the same insertion process. If bucket 4 is also not empty, the existing item, say c, will be kicked out to accommodate insertion of a. Hence, in some cases, the re-insertion process needs to repeat until an available bucket for the item being kicked out is found. Cuckoo filter composes of a set of cuckoo hash tables, as shown in Figure 4.2(b). The indexes of the candidate buckets for item x are computed as follows:

$$\begin{aligned} i_1 &= \text{hash}(x) \mod M \\ i_2 &= i_1 \oplus \text{hash}(\text{Fingerprint}(x)) \mod M, \end{aligned}$$

where M is the number of buckets in the cuckoo filter.

The cuckoo filter algorithm includes three basic functions: query, insert and delete. The query and insert functions are depicted in Algorithm 1 and Algorithm 2. In this paper, cuckoo filter will be used in the batch verification process to

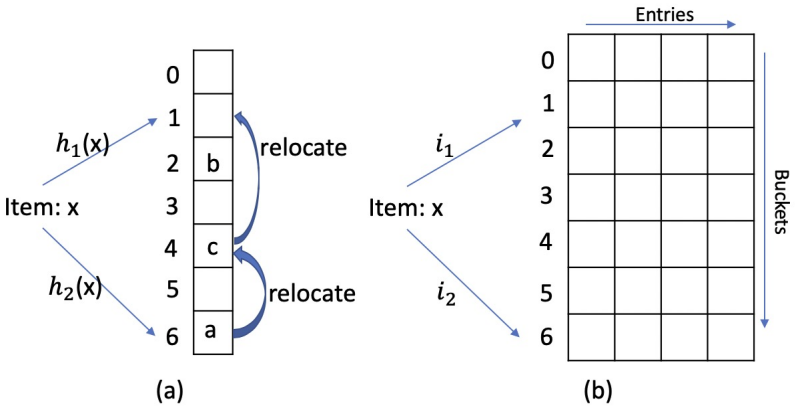


Figure 4.2: (a) Insertion at cuckoo hash table; (b)A cuckoo filter with four entries per bucket

generate a notification message to assist the message authentication of vehicles and enhance the overall efficiency.

4.2.3 Binary Tree and KUNodes Algorithm

We employ the binary tree data structure and KUNodes algorithm in [129] to realize scalable and efficient revocation. We use BT, $root$ and v to denote a binary tree, its root and a node respectively. $Path(v)$ denotes the set of nodes on the path from v to $root$, including v and $root$. KUNodes algorithm is performed by the KGC. Upon registration, the KGC firstly prepares a binary tree BT and then assigns an empty leaf node to each user. The algorithm accepts binary tree BT, revocation list RL, and time t and outputs the minimal set Y of nodes for which the key update needs to be published. The mechanics of the KUNodes algorithm is illustrated in Figure 4.3. The formal specification of KUNodes algorithm is given as follow.

$$
\begin{aligned}
&\text{KUNodes}(BT, RL, t)\\
&\quad X, Y \leftarrow \phi\\
&\quad \forall(v_i, t_i) \in RL\\
&\qquad \text{if } t_i \leq t \text{ then add } Path(v_i) \text{ to } X\\
&\quad \forall(x \in X)\\
&\qquad \text{if } \forall(x_l \notin X) \text{ then add } x_l \text{ to } Y
\end{aligned}
$$

Algorithm 1 Insert(x)

1: $f = Fingerprint(x)$;
2: i_1=hash(x) mod M;
3: i_2=($i_1 \oplus$hash(f)) mod M;
4: **if** bucket[i_1] or bucket[i_2] has an empty entry **then**;
5: add f to that bucket;
6: return done
7: **else**
8: i= randomly pick i_1 or i_2;
9: **for** $n = 0; n < MaxNumKicks; n++$ **do**
10: randomly select an entry e from bucket[i];
11: swap f and the fingerprint stored in ehtry e;
12: i=$i \oplus hash(f)$
13: **if** bucket[i] has an empty entry **then**
14: add f to bucket[i]
15: return done
16: **end if**
17: **end for**
18: return failure
19: **end if**

Algorithm 2 Query(x)

1: $f = Fingerprint(x)$
2: i_1=hash(x) mod M
3: i_2=($i_1 \oplus$hash(f)) mod M
4: **if** bucket[i_1] or bucket[i_2] has f **then**
5: return true
6: **else**
7: return false
8: **end if**

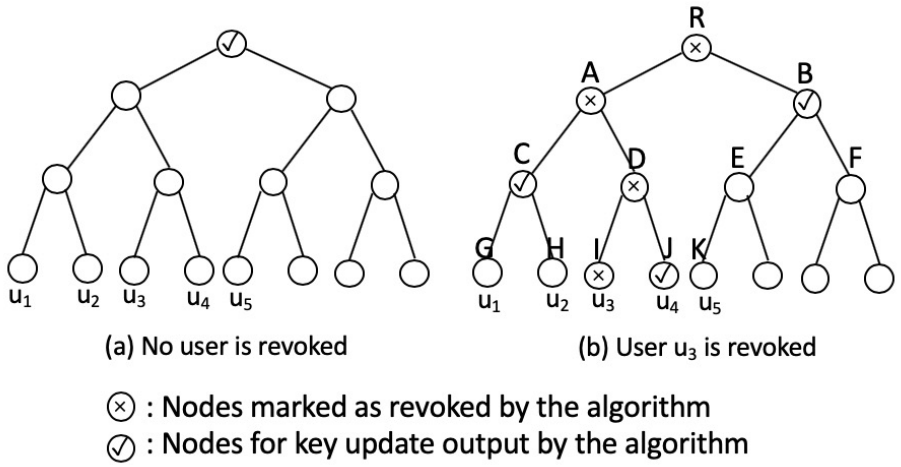


Figure 4.3: The KUNodes algorithm

if $\forall(x_r \notin X)$ then add x_r to Y
if $Y = \phi$ then add root to Y
return Y

As Figure 4.3 shows, the root node will be returned by the algorithm in the case that no user has been revoked. Suppose, users u_1, u_2, u_3, u_4, u_5 correspond to the nodes G, H, I, J, K in BT. If the user u_3 has been revoked, the algorithm outputs the set $Y = \{C, J, B\}$ which does not contain any ancestors of the revoked user. For the non-revoked user u_1 that corresponds to node G, $Path(G)$ has an intersection with KUNodes(BT, RL, t) at node C. Whereas, for the revoked user u_3 at node I, $Path(I)$ does not intersects with KUNodes(BT, RL, t). Consequently, all the non-revoked users have at least one node in Y that is on the path from the root to the corresponding nodes of the users, which indicates that they will receive the update key from the KGC. By employing this algorithm, the key update workload of KGC decreases from linear $O(n)$ to logarithmic $O(log_2{}^n)$ in the number of users. Hence, instead of updating the time key for every vehicle, the KGC only needs to broadcast the time keys for a minimum set of nodes for the non-revoked vehicles.

4.2.4 Security and Privacy Threats and Requirements

4.2.4.1 Threats

1. **Bogus Information Attack**: The adversary sends fake messages on purpose to launch an attack. For example, an adversary may broadcast wrong traffic warning messages to mislead other vehicles to the wrong directions.

2. **Impersonation Attack**: The attacker pretends to be a legitimate vehicle or an RSU and sends malicious messages to cause some damage to others.

3. **Message Modification Attack**: The safety-related message may be modified by an adversary during the transmission.

4. **Message Replay Attack**: The attacker collects the valid messages sent from a legitimate user and resends the messages to mislead the receiver.

4.2.4.2 Basic Requirements

1. **Identity Privacy Preserving**: The real identity of the network user should be hidden. In particular, it cannot be extracted from the transmitted messages by any adversaries. In this paper, vehicles obtain their pseudo identities from the KGC and use pseudo identities to communicate with others.

2. **Message Authentication and Integrity**: The message authentication scheme should enable the users to ensure that the messages are sent by a legitimate user of the network and are not modified by the adversaries.

3. **Non-repudiation**: The sender of a message cannot deny having sent the message at a later time.

4. **Traceability and Revocation**: Traceability means that the KGC can reveal the real identity of a malicious or misbehaving user of the network from its messages. Also, once a misbehaving or compromised user is identified, it should be possible to have its credential revoked.

5. **Revocation Transparency**: Since the KGC is the only trusted authority to register and revoke the vehicles of the vehicular networks, its activities about

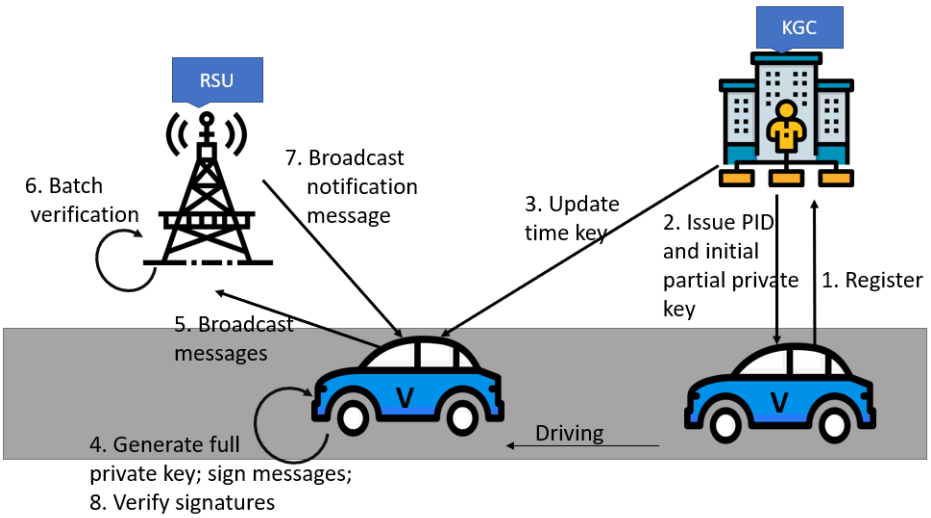


Figure 4.4: Process of message authentication

user revocation should be transparent and accountable. In other words, the revocation actions of the KGC should be available for inspection.

4.2.5 Overview of the Message Authentication Process

Figure 4.4 illustrates the scenario where a vehicle signs and authenticates messages in vehicular networks. In phase 1 and 2, a vehicle firstly registers with the KGC to obtain its initial partial private key and pseudo identity. It then generates the full private key using the initial partial private key, the update time key and a secret value. Afterwards, in phase 4 and 5, the vehicle signs messages and broadcasts the message-signature pairs over the network. In terms of signature verification, three kinds of scenarios that are supported exist. Firstly, verification of a single signature could be performed by a vehicle. Secondly, an RSU performs batch verification over multiple signatures at the same time. Thirdly, upon successful verification of multiple signatures, the RSU generates a notification message with the set of message-signature pairs represented using Cuckoo filter. A vehicle can make use of the verification result of a nearby RSU. This is known as RSU-assisted authentication, which corresponds to phase 7 and 8. In terms of revocation, the KGC runs the KUNodes algorithm by itself to realize scalable time key update for the non-revoked users.

4.2.6 Framework of the Revocable Signature Scheme

Different from the traditional online/offline certificateless signature schemes, a time-key, which is one part of the full private key of a vehicle, is generated by the KGC to revoke a misbehaving or compromised vehicle. The time-key is transmitted to the vehicles over the public channel. In order to revoke a vehicle, the KGC stops issuing new time-key for that vehicle. Without the new time-key, the vehicle is not able to obtain its full private key to sign new messages, thus being revoked from the vehicular network. The proposed authentication scheme consists of the following nine phases: Setup, Pseudo-Identity-Generation, Initial-Partial-Private-Key-Generation, Time-Key-Generation, Partial-Private-Key-Generation, Vehicle-Key-Generation, Offline-Sign, Online-Sign, Individual-Verify, Aggregate-Verify, Revocation.

Setup. In this phase, the KGC accepts a security parameter τ and the number of users n, generates the master public/private key pair (mpk/msk), the initial empty revocation list RL, state st, and the public system parameters ($params$).

Pseudo-Identity-Generation. In this phase, a vehicle registers with the KGC. The KGC receives the real identity of a vehicle and generate a pseudo identity (PID) using its private key, and assign the pseudo identity to the vehicle securely.

Initial-Partial-Private-Key-Generation. In this phase, the KGC takes public parameters $params$, master secret key msk, pseudo identity PID and state (binary tree) st as input, and outputs an initial partial private key $ipsk_{PID}$ and an updated state st'. Then the KGC delivers the initial partial private key to the vehicle through a secure channel.

Time-Key-Generation. In this phase, the KGC takes input public parameters $params$, master secret key msk, key update time t, a revocation list RL and state (binary tree) st to output the time key (tsk_t). The KGC broadcasts it over the network in every time period for the vehicles.

Partial-Private-Key-Generation. In this phase, a non-revoked user takes the initial partial private key $ipsk_{PID}$ and the time key (tsk_t) as input, outputs the partial private key ($psk_{PID,t}$). For the revoked users, it outputs a special symbol $\perp$.

Vehicle-Key-Generation. In this phase, a vehicle generates its own pub-

lic/secret key (vpk, vsk).

Offline-Sign. In this phase, the vehicle takes the $params$ as input and generate the offline component of the certificateless signature using its partial private key and secret key. This phase allows the vehicles to pre-compute a set of tuples without knowing the messages and stores them in local storage for use in the online-sign phase. Each time the offline component used in the online-sign should be different.

Online-Sign. In this phase, given a traffic-related message, the offline signature component and the full private key, the vehicle generates the certificateless signature and broadcasts the message with the signature over the network.

Individual-Verify. In this phase, the RSUs or vehicles verify the certificateless signature by using public $params$, messages, the signer's pseudo identity and its full public key. If the signature is valid, the verifier outputs $true$, otherwise outputs $false$ and rejects the message.

Batch Verification. In this phase, the RSUs verify multiple signatures at one time using public parameters and public keys. If the signatures are valid, it outputs $true$. Otherwise, it outputs $false$, and the RSUs find the invalid signatures in the batch. Finally, the RSUs store the information about valid and invalid signatures in two separate cuckoo filters, which are used to generate the notification message to assist message authentication of nearby vehicles.

Revocation. In this phase, the KGC adds the detected malicious or compromised users in vehicular networks. On input the corresponding leaf nodes and revocation time, the KGC updates its revocation list RL.

4.3 The Proposed Message Authentication Scheme of Vehicular Networks

This section presents the authentication process based on revocable certificateless signature scheme and the RSU-assisted authentication process using cuckoo filter. Before presenting our scheme, we describe the notations used. We use V_i to denote the i-th vehicle, use RID_i, PID_i to denote the real identity and pseudo identity of a V_i, and use psk_i, x_{PID_i}, vpk_{PID_i} to denote the partial private key, secret key and public key of V_i respectively. H_1, H_2, H_3, H_4 are four secure hash functions.

Δt_i,t_r,t_i are the valid time period of PID_i, revocation time, timestamp, respectively. We use m_i to denote a traffic-related message and use $\oplus$ to denote the exclusive **OR** operation.

4.3.1 System Parameter Setup

The KGC chooses a cyclic group G with generator P of prime order q. In practice, we will use an elliptic curve group which is usually written in additive notation. It also selects four secure hash functions H_1, H_2, H_3, H_4, all with domain $\{0,1\}^*$ and range Z_q^*. Then, the KGC selects a random number $\alpha \in Z_q^*$ as its master private key and sets $P_{pub} = \alpha \cdot P$ as the corresponding master public key. In order to use the KUNodes algorithm for revocation, the KGC initiates an empty revocation list RL and prepares a binary tree BT with N leaves (assume N is the maximum number of users). Moreover, the KGC also prepares an empty block as the genesis block to initialize a permissioned blockchain. Finally, the KGC sets the public parameters as $params$=$\{P, q, G, H_1, H_2, H_3, H_4, P_{pub}\}$, which will be pre-loaded by the RSUs and vehicles.

4.3.2 Pseudo-Identity-Generation

- The vehicle V_i firstly selects a random value $k_i \in Z_q^*$, and computes $PID_{i,1}$=k_iP. Then it sends RID_i and $PID_{i,1}$ to the KGC.

- On receiving $(RID_i, PID_{i,1})$ from the vehicle, the KGC first checks whether RID_i exists in RL. If it exists, the KGC aborts the registration process. Otherwise, the KGC computes $PID_{i,2} = RID_i \oplus H_1((\alpha \cdot PID_{i,1}), \Delta t_i, P_{pub})$ and sends $PID_{i,2}$ to the vehicle. The pseudo identity is PID_i=$(PID_{i,1}, PID_{i,2}, \Delta t_i)$, of which t_i is the valid period of PID_i. The KGC can reveal the real identity of V_i by computing $RID_i = PID_{i,2} \oplus H_1((\alpha \cdot PID_{i,1}), \Delta t_i, P_{pub})$.

- The KGC submits a "Store" request $(Store, KGC, Msg)$, where Msg contains the latest revocation list RL, to the blockchain network.

4.3.3 Initial-Partial-Private-Key-Generation

After generating the pseudo identity for vehicle V_i, KGC randomly chooses an empty leaf node η_{PID_i} from BT and stores PID_i into this node. The KGC generates the partial private key by performing the following algorithm.

$\forall \theta \in Path(\eta_{PID_i})$

if r_θ is undefined, then $r_\theta \xleftarrow{\$} Z_q^*$,

store r_θ in node θ,

$d_\theta \leftarrow r_\theta \cdot P$; $D_\theta \leftarrow r_\theta + \alpha \cdot H_2(PID_i, d_\theta) \pmod q$

Return $ipsk_{PID_i}=\{(\theta, d_\theta, D_\theta)\}_{\theta \in Path(\eta_{PID_i})}$, st.

The algorithm computes the identity-component of the full private signing key for all the nodes on the path from η_{PID_i} corresponding to the identity PID_i to the $root$. Finally, the KGC transmits $ipsk_{PID_i}=(d_\theta, D_\theta)_{\theta \in Path(\eta_{PID_i})}$ to V_i through a secure channel.

4.3.4 Time-Key-Generation

The KGC firstly runs the KUNodes algorithm to obtain the minimal set of nodes for the key update of the non-revoked users. Then it computes the time-component of the full private key for all the nodes in that set. Finally, the KGC broadcasts the time-component keys through the public channel. This algorithm is presented as follows.

$\forall \mu \in \text{KUNodes}(BT, RL, t_r)$

$m_\mu \xleftarrow{\$} Z_q^*$; $e_\mu \leftarrow m_\mu \cdot P$;

$E_\mu \leftarrow m_\mu + \alpha \cdot H_3(t_r, e_\mu) \pmod q$.

Return $tsk_t=\{(\mu, e_\mu, E_\mu)\}_{\mu \in KUNodes(BT,RL,t_r)}$.

4.3.5 Partial-Private-Key-Generation

On input $ipsk_{PID_i}$ as $\{(\theta, d_\theta, D_\theta)\}_{\theta \in Path(\eta_{PID_i})}$ and tsk_t as $\{(\mu, e_\mu, E_\mu)\}_{\mu \in KUNodes(BT,RL,t_r)}$, non-revoked users execute the following steps:

$\forall (\theta, d_\theta, D_\theta) \in ipsk_{PID_i}, (\mu, e_\mu, E_\mu) \in tsk_t$

If $\exists(\theta, \mu)$ s.t. $\theta = \mu$, then $psk_{PID_i,t} \leftarrow (D_\theta, E_\mu)$

Else($ipsk_{PID_i}, tsk_t$ do not have any node in common), then $psk_{PID_i,t} \leftarrow \perp$.

Return $psk_{PID_i,t}$ as the partial private key. Note that condition "$\exists(\theta, \mu)$ s.t.

$\theta = \mu$" indicates the scenario that the leaf node corresponds to the pseudo identity PID_i has an ancestor(or itself) in the set of $KUNodes(BT, RL, t_r)$. This means that user with the pseudo identity PID_i is a non-revoked user before or, at time t_r, thus is able to obtain their partial private key from the initial partial private key and the time key received from the KGC. However, if this condition does not hold, it means the user with pseudo identity PID_i has been revoked before or, at time t_r, thus will not get time key from the KGC. We denote the partial private key of PID_i as $(D^{(PID_i)}, E^{(PID_i)})$, where $D^{(PID_i)} = r + \alpha \cdot H_2(PID_i, d^{(PID_i)}) \ (mod\ q)$, $d^{(PID_i)} = r \cdot P$ and $E^{(PID_i)} = m + \alpha \cdot H_3(t_r, e^{(PID_i)}) \ (mod\ q)$, $e^{(PID_i)} = m \cdot P$. The user can check the validity of partial secret keys by verifying that $D^{(PID_i)} \cdot P = d^{(PID_i)} + H_2(PID_i, d^{(PID_i)}) \cdot P_{pub}$ and $E^{(PID_i)} \cdot P = e^{(PID_i)} + H_3(t_r, e^{(PID_i)}) \cdot P_{pub}$. Note that $d^{(PID_i)}, e^{(PID_i)}, P_{pub}, PID_i, t_r$ are public.

4.3.6 Vehicle-Key-Generation

The vehicle randomly chooses $x_{PID_i} \in Z_q^*$ as its secret value and computes $vpk_{PID_i} = x_{PID_i} \cdot P$ as its public key. Then, a non-revoked user with pseudo identity PID_i before or, at time t, will use the partial private key $(D^{(PID_i)}, E^{(PID_i)})$ and its secret value x_{PID_i} to generate signatures on messages. The public keys for signature verification should contain $\{vpk_{PID_i}, d^{(PID_i)}, e^{(PID_i)}, t_r\}$.

4.3.7 Offline-Sign

This algorithm is performed by a non-revoked user (vehicle) V_i in the vehicular network. V_i chooses a random number $w_i \in Z_q^*$, computes $W_i = w_i \cdot P$ and stores the offline $\phi_i = (w_i, W_i)$ locally. The vehicle does not need to know the message to generate the offline signature, hence the offline tuples can be pre-computed when the OBU is idle.

4.3.8 Online-Sign

On input message m_i, current timestamp t_i, partial private key $(D^{(PID_i)}, E^{(PID_i)})$, secret value x_{PID_i} and an offline tuple ϕ_i, V_i (with pseudo identity PID_i) generates the signature as follows. Firstly, it parses ϕ_i as (w_i, W_i). Then, it computes $h_{4i} = H_4(m_i, PID_i, d^{(PID_i)}, e^{(PID_i)}, vpk_{PID_i}, W_i, t_i)$, and $s_i = w_i + h_{4i} \cdot (D^{(PID_i)} +$

$E^{(PID_i)} + x_{PID_i})$ $(modq)$. The output signature is $\sigma_i = (W_i, s_i, d^{(PID_i)}, e^{(PID_i)})$. Finally, V_i broadcasts $\{m_i, PID_i, \sigma_i, t_i, vpk_{ID_i}\}$ over the network.

4.3.9 Individual-Verify

Upon receiving message $\{m_i, PID_i, \sigma_i, t_i, vpk_{PID_i}, d^{(PID_i)}, e^{(PID_i)}\}$, the receiver firstly checks the freshness of t_i. If the timestamp is not fresh, it drops the message. Then, the verifier computes $h_{4i} = H_4(m_i, PID_i, d^{(PID_i)}, e^{(PID_i)}, vpk_{PID_i}, W_i, t_i)$, $h_{2i} = H_2(PID_i, d^{(PID_i)})$ and $h_{3i} = H_3(t_r, e^{(PID_i)})$ and checks the following equation: $s_i \cdot P = W_i + h_{4i} \cdot (d^{(PID_i)} + h_{2i} \cdot P_{pub} + e^{(PID_i)} + h_{3i} \cdot P_{pub} + vpk_{PID_i})$. If this equation holds, the verifier accepts the message, otherwise it rejects the message.

4.3.10 Batch Verification

In batch verification process, a RSU receives multiple pairs of messages, such as $\{m_i, PID_i, \sigma_i, t_i, vpk_{PID_i}, d^{(PID_i)}, e^{(PID_i)}\}$ where $i = 1, 2, 3, ...n$, and verifies all the signatures together using this algorithm. The RSU firstly checks freshness of t_i, for $i = 1, 2, 3, ...n$ before proceeding. Then, it randomly selects a vector v=$\{v_1, v_2, v_3, ..., v_n\}$, where each v_i is small, and validates the following equation: $(\sum_{i=1}^{n} s_i \cdot v_i) \cdot P$=$\sum_{i=1}^{n}(W_i \cdot v_i) + \sum_{i=1}^{n}(h_{4i} \cdot v_i \cdot d^{(PID_i)}) + (\sum_{i=1}^{n}(h_{4i} \cdot h_{2i} \cdot v_i + h_{4i} \cdot v_i \cdot h_{3i})) \cdot P_{pub} + \sum_{i=1}^{n}(h_{4i} \cdot v_i \cdot e^{(PID_i)}) + \sum_{i=1}^{n}(h_{4i} \cdot v_i \cdot vpk_{PID_i})$. The RSU accepts the message if the equation holds. Otherwise, the RSU extracts the valid signatures from the batch and stores the information about signature validity in two separate cuckoo filters.

4.3.11 Revocation

When a user should be revoked, the leaf node η_{PID} associated with this user PID with the revocation time period t will be added into the revocation list of the KGC. The KGC will update its revocation list by $RL \leftarrow RL \cup \{\eta_{PID}, t\}$. Each time the RL is updated, the KGC will broadcast a new "Store" request to the blockchain network to keep the revocation record. Any vehicle can check the revocation list by submitting a "View" request with its pseudo identity to a nearby RSU (who acts as a full node).

4.4 RSU-assisted Verification

This section describes the process in which the RSUs generate a notification message using cuckoo filter after executing batch verification to assist nearby vehicles to verify signatures efficiently.

4.4.1 Generating Notification Message

In order to avoid the drop of the batch due to few invalid signatures, the RSU could use the commonly employed technique of binary search as in [70, 124, 127] to identify invalid signatures from the batch. After running Algorithm 3, where $List$ (resp. $List1$) denotes a list of the (resp. invalid) message-signature pairs in the batch, the RSU obtains the list of invalid signatures, and arranges valid and invalid signatures together with the corresponding pseudo identities into two lists $validList(V_i)$ and $invalidList(V_i)$. Then, the RSU uses two cuckoo filters, namely, $posFilter$ and $negFilter$ to store the concatenation of the messages, pseudo identities and timestamps corresponding to the valid and invalid signatures respectively. The specific algorithm for notification message generation is given as Algorithm 4. Finally, the RSU broadcasts the notification message as $\{posFilter, negFilter, SIG_{sk_{RSU}}(posFilter, negFilter)\}$ over the network.

4.4.2 Signature Verification Using Cuckoo Filters

Using notification message from a nearby RSU, vehicle V_i can verify message-signature pair (m_j, σ_j) quickly. Specifically, vehicle V_i firstly calculates the fingerprint as $f_j = Fingerprint(x_j)$, where $x_j = (PID_j||t_j||m_j)$. Then, it computes the indexes for the item x_j as i_1=hash(x_j) mod M, i_2=($i_1 \oplus$hash(Fingerprint(x_j))) mod M. Afterwards, it queries the cuckoo filters in the notification messages for the item x_j. The process of message authentication using cuckoo filters is shown in Algorithm 5.

Since the cuckoo filter inherently has false positive, four possible results with different implications exist, as shown in Table 4.1. The first case is that the $posFilter$ outputs true and $negFilter$ outputs false, which indicates that σ_j is valid. The second case is that $posFilter$ outputs false and $negFilter$ outputs true, which indicates that σ_j is invalid. The false positive happens in case 3, where

Algorithm 3 $signatureExtract(List, List1, low, high)$

if $batchVerity(List, low, high) == true$ **then**
    return 1
else
    if $low == high$ **then**
        $List1.add(List[low])$
        return 1
    else
        $mid = (low + high)/2$
        $signatureExtract(List, List1, low, mid)$
        $signatureExtract(List, List1, mid + 1, high)$
        return 1
    end if
end if

Algorithm 4 The RSU generates notification message

for $PID_i \in validList(V_i)$ **do**
    $x_i \leftarrow (PID_i||t_i||m_i)$
    $posFilter.Insert(x_i)$
end for
for $PID_i \in invalidList(V_i)$ **do**
    $x_i \leftarrow (PID_i||t_i||m_i)$
    $negFilter.Insert(x_i)$
end for
return $\{posFilter, negFilter, SIG_{sk_{RSU}}(posFilter, negFilter)\}$

Algorithm 5 V_i verifies σ_j of V_j

$x_j \leftarrow (PID_j||t_j||m_j)$
while $t_j$is fresh **do**
    V_i queries $posFilter, negFilter$ on f_j
    if $posFilter.Query(x_j) == true$ **then**
        if $negFilter.Query(x_j) == false$ **then**
            V_i accepts the validity of σ_j; break;
        else
            if $negFilter.Query(x_j) == true$ **then**
                V_i resends σ_j to the RSU or V_i verifies the σ_j by itself; break;
            end if
        end if
    else
        if $negFilter.Query(x_j) == true$ **then**
            V_i rejects the validity of σ_j; break;
        end if
        if $negFilter.Query(x_j) == false$ **then**
            V_i waits for next notification broadcast or V_i verifies the σ_j by itself;break;
        end if
    end if
end while

Table 4.1: Possible query results and their implications

Cases	Positive Filter	Negative Filter	Implications
1	Ture	False	σ_j is valid
2	False	True	σ_j is invalid
3	True	True	False positive happens
4	False	False	σ_j has not been verified

both two filters yield true for the query operation. The fourth case happens when both filters output false. In case 3, vehicle V_i can either send the signature back to the nearby RSU for re-confirmation or verify the signature by itself. For the re-confirmation process, the RSU is assumed to store the verified signatures with its corresponding pseudo identities for at least one more batch after the broadcast of the notification message. Hence, once received the re-confirmation message from a vehicle, the RSU can check the validity of the signature and insert the information into the corresponding filter of the next notification message. This re-confirmation process actually happens very rarely, as the probability of the false positive of cuckoo filter is very low. Its analysis is given in the last section. Typically, case 1 or 2 happens and V_i can be sure about the validity of the message. Case 4 means that σ_j has not been verified by the RSU. In this case, V_i may opt for individual verification or wait for the next notification message from RUS.

4.5 Security Analysis

1. **Identity Privacy Preserving**: Each user of the network will use a pseudo identity, which is computed using the master private key α of the KGC in the registration phase, to communicate with the others. The real identity can only be extracted using α by the equation $RID_i = PID_{i,2} \oplus H_1((\alpha \cdot PID_{i,1}), \Delta t_i, P_{pub})$. However, it is infeasible to extract private key α from pubic key P_{pub} or any other messages. Therefore, our scheme preserves the identity privacy of users.

2. **Message Authentication and Integrity**: Only legitimate user that has

registered with the KGC can generate a valid signature which can be verified successfully. And any adversary attempts to alter the transmitted message will lead to failure of the signature verification.

3. **Non-repudiation**: In vehicular networks, each message is signed by a vehicle before broadcasting to others. Hence, once a vehicle generates a signature on a message, it cannot deny that the signature is generated by itself later.

4. **Traceability and Revocation**: From PID_i=($PID_{i,1}, PID_{i,2}, \Delta t_i$), where $PID_{i,1}$=k_iP, $PID_{i,2} = RID_i \oplus H_1((\alpha \cdot PID_{i,1}), \Delta t_i, P_{pub})$, the KGC can extract the real identity by computing $RID_i = PID_{i,2} \oplus H_1((\alpha \cdot PID_{i,1}), \Delta t_i, P_{pub})$. In terms of revocation, the KGC can revoke the malicious or compromised users by updating the time key for non-revoked users. And any detected attackers will be added into the revocation list of the KGC, thus cannot generate signatures on messages anymore. Therefore, traceability and scalable revocation are ensured in our scheme.

5. **Revocation Transparency**: Once the revocation list is updated, the KGC stores the updated revocation list into the blockchain. Since blockchain is an immutable ledger with a consensus algorithm, all the revocation actions of the KGC stored in the blockchain are trusted and available for inspection. Hence, revocation transparency are ensured in our scheme.

4.6 Performance Issues

4.6.1 Complexity

We evaluate the computation and communication cost of our signature scheme. Let T_{bp}, T_{bp-m}, T_{ecc-m}, T_H and T_h denotes the execution time of a bilinear paring, scalar multiplication in a pairing-friendly group, scalar multiple in an ecc-group, a map-to-point hash function and an ordinary hash function respectively. Using parameters and benchmark results from [117], T_{bp}, T_{bp-m}, T_{ecc-m}, T_H and T_h is 4.2110 ms, 1.7090 ms, 0.4420 ms, 4.406 ms and 0.0001 ms, respectively. Let $|G_{pr}|$ and $|G_{ecc}|$ respectively denotes the length of a pairing-based group element and an ecc-based group element. And $|G_{pr}|$ are 128 bytes and $|G_{ecc}|$ are 40 bytes. Denote

Table 4.2: Comparisons of computation cost and signature size with existing schemes

Schemes	Sign(ms)	Verify(ms)	Signature Tuple Size (bytes)
[121]	$2T_{bp-m} + 2T_H \approx 12.23$	$3T_{bp} + 2T_H \approx 21.445$	$2\|G_{pr}\|$=256
[130]	$2T_{bp-m} + 2T_H \approx 12.23$	$4T_{bp} + 3T_H + T_{bp-m} \approx 31.771$	$3\|G_{pr}\|$=384
[123]	$T_{ecc-m} + 2T_h \approx 0.4422$	$5T_{ecc-m} + 4T_h \approx 2.2104$	$3\|G_{ecc}\|+\|Z_q^*\|$=140
Our scheme	$T_{ecc-m} + T_h \approx 0.4421$	$4T_{ecc-m} + 3T_h \approx 1.7683$	$3\|G_{ecc}\|+\|Z_q^*\|$=140

$\left|Z_q^*\right|$ as the length of a group element in Z_q^* with q is 160-bits. We compare our scheme with other revocable certificateless digital signature schemes and the result is shown in Table 4.2.

Table 4.2 shows that the computation and communication cost of the schemes [121, 130], which use pairings and map-to-point hash functions, are higher than schemes that are pairing-free, such as [123] and our scheme. While our scheme and the pairing-free revocable certificateless signature scheme [123] offer similar complexity, our scheme has a number of advantages. Firstly, our scheme supports online/offline signature generation to reduce the computation cost. Secondly, our scheme supports efficient revocation using the KUNodes algorithm, which reduces the revocation burden of the KGC to logarithmic complexity.

4.6.2 Analysis of the Cuckoo Filter

1. Comparing with the bloom filter, cuckoo filter has several advantages. Cuckoo filter supports adding and removing items dynamically, which is impossible for conventional bloom filter. This function is necessary for message authentication in VANETs scenarios, where the RSU can update the signature validity information timely by adding and removing items into the two filters. For example, if the timestamp of a signature is not fresh, the RSU will remove its signature validity information timely. Moreover, cuckoo filter provides higher lookup performance than the traditional bloom filters.

2. Cuckoo filter has a very small false positive rate. We evaluate the false positive using the upper bound of the total probability of a false fingerprint

collision as $1-(1-1/2^f)^{2b} \approx 2b/2^f$, where f is the fingerprint length in bits and b is the number of entries per bucket. We set $b = 4$ to achieve the best or close-to-best space efficiency for false positive rate that suits our needs. If f is set to be 13 bits, which is very short, the false positive is calculated to be 0.000976. This means that less than 1 out of 1000 signatures may need a re-confirmation process, which corresponds to the case 3. Moreover, if we increase f only by a few bits, the false positive will decrease exponentially. Hence, we can achieve negligible false positive with a relatively short fingerprint length.

3. We further conduct an experiment to evaluate the computation cost of the basic operations of the cuckoo filter. We perform the experiment using a MacBook Pro notebook, with an Intel i5 processor with 3.1GHz clock frequency and 16 gigabytes memory. We use cuckoo filter parameters in [131] to perform the experiment. Specifically, we set the number of entries per bucket(b) to be 4, fingerprint length to be 12 bits, load factor to be 95.36%, filter capacity to be 1000000, false positive rate to be 0.0944%, and the number of operations to be 1000000. We implement a simple program to execute the query, insert and delete operations for 1 million times utilizing the C++ library in [119], and record the execution time. The result shows that the execution time of query, insert and delete operations is 56ms, 75ms and 67ms respectively. Comparing to the time of verifying a single signature, as shown in Table 4.2, the computation overhead of cuckoo filter is small. Hence using cuckoo filter for message authentication can improve the overall efficiency.

4.7 Chapter Summary

We employ a revocable pairing-free online/offline certificateless signature scheme to achieve secure and efficient message authentication for vehicular networks. The revocation mechanism does not require a secret channel or an additional online security mediator, hence it is practical to be applied in vehicular networks. We analyze the performance of the revocable signature scheme by comparing the computation time and signature size of the proposed scheme with that of existing schemes, and the result shows that our scheme offers better performance than ex-

isting schemes. More importantly, our proposal offers three additional desirable features. The first feature is the use of cuckoo filter to support RSU-assisted message authentication. This allows the vehicles to authenticate messages using the notification message from the RSU without verifying individual signature by itself. Hence, the overall efficiency of the message authentication process is improved. The second feature is the use of KUNodes algorithm to support KGC-decided user revocation. By using the KUNodes algorithm, the revocation workload of the KGC is reduced, as it grows logarithmically instead of linearly with the number of users. The third feature is the use of blockchain to allow network participants to check the transaction data about the revocation list from the blockchain. Hence, this feature improves the transparency of the KGC's operation about user revocation. Finally, security analysis shows that the proposed signature scheme meets all the security and privacy requirements of vehicular networks.

Chapter 5

Range Proof

In this chapter, as an important component of an anonymous credential system and could be used to protect user privacy in VANETs, the technique of range proof is investigated. Specifically, we studied the range proof protocol in the cryptocurrency Monero and found security flaws in the protocol. Then we propose an improved range proof protocol and present a rigorous security proof. The analysis shows that the improved range proof protocol is practical and secure.

5.1 Range Proof of Monero

Monero is an open-source decentralized cryptocurrency with strong privacy. The core technique used in Monero to achieve strong privacy is called ring confidential transaction (RingCT) protocol. As wallet balance is stored in committed forms, RingCT requires the sender to issue a proof that (1) the input of a ringCT transaction (it may involve multiple input wallets) is equal to that of the output (again, it may involve multiple outputs); and (2) the balance of each wallet involved is within the range of permitted values. The second part is a range proof and importance has been explained in detail in the Monero white paper [132]. Briefly, the cryptographic primitives of Monero work in a cyclic group of known orders, say q. Then, a commitment of -1 is equivalent to $q-1$. Thus, a sender with 1 dollar in his wallet could send a transaction with the output of 2 and $q-1$ if such a range proof is not enforced. It is prevented by requiring the sender to prove that

all commitments involved are confined within a small range[1].

Despite being one of the most important components and accounts for over 50% of the total bandwidth of a ringCT transaction, the range proof employed in ringCT is not well-studied. This range proof combines bit decomposition technique with ring signatures in an unconventional way [133]. The Monero white paper [132] proposed a new ring signature called ANSL, and discussed two other options, namely, a well-studied scheme from Abe et al. [134], and a newly proposed scheme called Borromean ring signature [133].

Remarks. We would like remark that our results do not imply Monero is broken. Firstly, perhaps a witness-indistinguishable proof, instead of a zero-knowledge proof-of-knowledge, provides sufficient security guarantee. Secondly, while the range proof is not a zero-knowledge proof-of-knowledge, it may be the case that the overall ringCT protocol can still be proven secure. However, this will involve a comprehensive analysis of the protocol and we leave it as an open problem to actually quantify the security and privacy guarantee of the current version of Monero.

5.2 Preliminaries

5.2.1 Notations

Let a be a string, we use $|a|$ to denote the length of a. Let $\mathcal{S}$ be a finite set, then we use $s \xleftarrow{\$} \mathcal{S}$ to denote sampling an element s uniformly from set $\mathcal{S}$. We write $negl(\cdot)$ to denote a negligible function, and write $poly(\cdot)$ to denote a polynomial. For integers $a \leq b$, we write $[a, b]$ to denote all integers that is not less than a and not greater than b. For two random variables, say $\mathcal{X}$ and $\mathcal{Y}$, we write $\mathcal{X} \overset{c}{\approx} \mathcal{Y}$ to denote that $\mathcal{X}$ and $\mathcal{Y}$ are computationally indistinguishable.

5.2.2 Complexity Assumptions

Let $\mathbb{G}_1$ and $\mathbb{G}_T$ be groups of order q for some large prime q. Let e be a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ that:

1. For any $g, h \in \mathbb{G}_1$ and any $a, b \in \mathbb{Z}_q$, $e(g^a, h^b) = e(g, h)^{ab}$.

[1] In the current version of monero, the range is $[0, 2^{64} - 1]$.

2. Let g, h be generators of $\mathbb{G}_1$, then $e(g, h)$ is a generator of $\mathbb{G}_T$.

We will use the Discrete Logarithm (DL) assumption and the Computational Diffie-Hellman (CDH) assumption over the bilinear groups $(\mathbb{G}_1, \mathbb{G}_T, q, e)$, which can be defined as follows.

Definition 5.2.1 (The CDH assumption.)**.** Let $(\mathbb{G}_1, \mathbb{G}_T, q, e)$ be bilinear groups that are sampled according to some generator algorithm $\mathcal{G}(1^\lambda)$. Let g be generators of $\mathbb{G}_1$, then it is hard to compute g^{ab} given (g^a, g^b) for $a, b \xleftarrow{\$} \mathbb{Z}_q$.

Definition 5.2.2 (The DL assumption.)**.** Let $(\mathbb{G}_1, \mathbb{G}_T, q, e)$ be bilinear groups that are sampled according to some generator algorithm $\mathcal{G}(1^\lambda)$. Let g be generators of $\mathbb{G}_1$, then it is hard to compute a given g^a for $a \xleftarrow{\$} \mathbb{Z}_q$.

Although a few works have been done to prove the equivalence of these two problems (see [135] and references therein), they only work in a specific group and may rely on unproven mathematical conjectures. So, in general, it is believed that the CDH assumption and the DL assumption are not equal.

5.2.3 Syntax of Range Proof

The range proof [86, 88] is nothing more than a non-interactive zero-knowledge proof of knowledge (NIZKPoK) [136] for the special statement that the committed value in a given commitment lies in an interval. More formally, a NIZKPoK for a language L has two algorithms:

- **Prove.** The prove algorithm takes as input a common reference string R, a statement x and a witness w, and outputs a proof π for x.
- **Verify.** The verify algorithm takes as input a common reference string R, a statement x and a proof π, and outputs a bit $b \in \{0, 1\}$.

5.2.4 Properties of NIZKPoK

Generally, a non-interactive zero-knowledge proof of knowledge has (some of) the following properties: completeness, soundness, proof of knowledge, zero-knowledge, witness-indistinguishability. In this section, we recall the definition of the completeness, soundness, proof of knowledge, zero-knowledge, and witness-indistinguishability for NIZKPoKs.

- **Completeness.** For any true statement $x \in L$, and witness w for x, we have

$$\Pr[R \xleftarrow{\$} \{0,1\}^{poly(n)}; \pi \leftarrow Prove(R, x, w) : Verify(R, x, \pi) = 1] = 1.$$

- **Soundness.** For any false statement $x \notin L$ and for any (not necessary PPT) adversary P^*, we have

$$\Pr[R \xleftarrow{\$} \{0,1\}^{poly(n)}; \pi \leftarrow P^*(R, x) : V(R, x, \pi) = 1] \leq negl(n).$$

- **Proof of Knowledge.** There exists an PPT extractor $\mathcal{E}$ that for any statement x, for any PPT adversary P^*, if the adversary could generate a valid proof π that passes the verify algorithm with non-negligible probability, then the extractor $\mathcal{E}$ can extract the witness for x from P^* with a non-negligible probability.

- **Zero-Knowledge.** There exists a PPT simulator $\mathcal{S}$ such that for any $x \in L$ and any witness w for x, we have:

$$(R_1, x, \pi_1) \overset{c}{\approx} (R_2, x, \pi_2)$$

  where $R_1 \xleftarrow{\$} \{0,1\}^{poly(n)}$, $\pi_1 \leftarrow P(R_1, x, w)$ and $(R_2, \pi_2) \leftarrow \mathcal{S}(x)$.

- **Witness-Indistinguishability.** For any $x \in L$ and any w_1, w_2 that are witnesses for x, we have

$$(R, x, \pi_1) \overset{c}{\approx} (R, x, \pi_2)$$

  where $R \xleftarrow{\$} \{0,1\}^{poly(n)}$, $\pi_1 \leftarrow P(R, x, w_1)$ and $\pi_2 \leftarrow P(R, x, w_2)$.

5.2.5 Syntax of Ring Signatures

The ring signature scheme allows a user to sign on behalf of a group of users while protecting the privacy of that user. In this paper, we will use the improved security definition presented in [137].

Formally, a ring signature consists of four algorithms:

- **Setup.** On input a security parameter 1^λ, the setup algorithm outputs the public parameter $param$ for the scheme, which is also set implicitly as the input for the following three algorithms.

- **KeyGen.** The key generation algorithm outputs a secret key/public key pair $(sk, pk) \in \mathcal{SK} \times \mathcal{PK}$, where we use $\mathcal{SK}$ and $\mathcal{PK}$ to denote the secret key space and the public key space respectively.

- **Sign.** On input a message $\mathfrak{m}$, a polynomial-size set $\mathcal{R}$ of public keys, and a secret key sk whose corresponding public key is in $\mathcal{R}$, the signing algorithm outputs a signature σ.

- **Verify.** On input a message $\mathfrak{m}$, a polynomial-size set $\mathcal{R}$ of public keys, and a signature σ, the verification algorithm outputs a bit indicating whether the signature is acceptable.

5.2.6 The Security Definition of Ring Signatures

Security of the ring signature scheme requires that it has the correctness, the unforgeability w.r.t. insider corruption, and the anonymity against full key exposure. In this section, we recall the security definition of ring signature scheme.

The correctness of the ring signature scheme requires that an honestly signed signature should pass the verification. More formally, we require that:

Definition 5.2.3 (Correctness.)**.** Let $param \leftarrow Setup(1^\lambda)$, $(sk, pk) \leftarrow KeyGen()$. Then for any message $\mathfrak{m}$, any polynomial-size set $\mathcal{R} \in 2^{\mathcal{PK}}$ containing pk, we have $\Pr[Verif(\mathfrak{m}, \mathcal{R}, Sign(\mathfrak{m}, \mathcal{R}, sk)) = 1] \geq 1 - negl(\lambda)$.

Security of the ring signature scheme requires that the scheme is unforgeable w.r.t. insider corruption, anonymous against full key exposure, which is formally defined as follows.

Definition 5.2.4 (Unforgeability w.r.t. insider corruption.)**.** A ring signature scheme is unforgeable w.r.t. insider corruption if for any probabilistic polynomial time adversary $\mathcal{A}$ and for any polynomial $n(\cdot)$, the probability that $\mathcal{A}$ succeeds in the following game is negligible in λ.

1. In the beginning, the challenger generates $param \leftarrow Setup(1^\lambda)$ and $(sk_i, pk_i) \leftarrow KeyGen(param; \omega_i)$ for $i \in [1, n]$, where ω_i is the randomness used in the generation of the ith key pair. Then it sends the $param$ and the set $\mathcal{S} = \{pk_i\}_{i=1}^n$ to $\mathcal{A}$. It also initializes two empty sets $\mathcal{SO}$ and $\mathcal{CO}$.

2. Then $\mathcal{A}$ is allowed to access the following two oracles:

   - *The Signing Oracle.* On input an integer $j \in [1, n]$, a message $\mathfrak{m}$ and a ring $\mathcal{R} \subseteq \mathcal{PK}$ containing pk_j, the challenger returns $\sigma \leftarrow Sign(\mathfrak{m}, \mathcal{R}, sk_j)$ and puts $(m, \mathcal{R}, \sigma)$ into $\mathcal{SO}$.
   - *The Corrupt Oracle.* On input an integer $j \in [1, n]$, the challenger returns ω_j and puts pk_j into $\mathcal{CO}$.

3. Finally, $\mathcal{A}$ outputs $(\mathfrak{m}^*, \mathcal{R}^*, \sigma^*)$, and succeeds if $Verify(\mathfrak{m}^*, \mathcal{R}^*, \sigma^*) = 1$, $(\mathfrak{m}^*, \mathcal{R}^*, \sigma^*) \notin \mathcal{SO}$, $\mathcal{R}^* \subseteq \mathcal{S}$, $\mathcal{R}^* \cap \mathcal{CO} = \emptyset$.

Definition 5.2.5 (Anonymity against full key exposure.)**.** A ring signature scheme is anonymous against full key exposure if for any probabilistic polynomial time admissible adversary $\mathcal{A}$ and for any polynomial $n(\cdot)$, the probability that $\mathcal{A}$ succeeds in the following game is negligibly close to $1/2$.

1. In the beginning, the challenger generates $param \leftarrow Setup(1^\lambda)$ and $(sk_i, pk_i) \leftarrow KeyGen(param; \omega_i)$ for $i \in [1, n]$, where ω_i is the randomness used in the generation of the ith key pair. Then it sends the public parameter $param$ and the set $\mathcal{S} = \{pk_i\}_{i=1}^n$ to $\mathcal{A}$.

2. Then $\mathcal{A}$ is allowed to access the following two oracles:

   - *The Signing Oracle.* On input an integer $j \in [1, n]$, a message $\mathfrak{m}$ and a ring $\mathcal{R} \subseteq \mathcal{PK}$ containing pk_j, the challenger returns $\sigma \leftarrow Sign(\mathfrak{m}, \mathcal{R}, sk_j)$.
   - *The Corrupt Oracle.* On input an integer $j \in [1, n]$, the challenger returns ω_j.

3. Next, $\mathcal{A}$ submits a message $\mathfrak{m}^*$, two distinct integers $i_0^*, i_1^* \in [1, n]$, and a ring $\mathcal{R}^* \subseteq \mathcal{PK}$ that $pk_{i_0^*}, pk_{i_1^*} \in \mathcal{R}^*$. Then, the challenger choose a random bit $b \in \{0, 1\}$, and returns a signature $\sigma^* \leftarrow Sign(\mathfrak{m}^*, \mathcal{R}^*, sk_{i_b^*})$.

4. After receiving the challenge signature σ^*, $\mathcal{A}$ is further allowed to access the signing oralce and the corrupt oracle as in the second phase.

5. Finally, $\mathcal{A}$ outputs a bit b' and succeeds if $b = b'$.

5.3 Analysis of the Range Proof in Monero

5.3.1 Review of Monero's Range Proof

We first review the range proof employed by Monero. Since it is using ring signature as a building block, we call it ring signature-based range proof.

Let g, h be two random generators of a cyclic group $\mathbb{G}$ of prime order q. The common input is a Pedersen commitment C, and an interval $[0, 2^\ell - 1]$ for some integer ℓ such that $\ell < |q|$. . The prover's private input include (u, r). The goal of the prover is to convince the verifier that u lies in an interval $[0, 2^\ell - 1]$.

Let $\Pi = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$ be a ring signature scheme such that output of KeyGen is of the form (h^x, x). In other words, it has to be a ring signature scheme whose public-secret key pair matches the domain of the Pedersen commitment.

- The prover first conducts a binary decomposition of u and r. That is, it finds $u_i \in \{0, 1\}$ for $i = 0$ to $\ell - 1$ such that $u = \sum_{i=0}^{\ell-1} 2^i u_i$. Next, it chooses $r_i \in_R \mathbb{Z}_q$ uniformly at random, and compute $r_{\ell-1} = r - \sum_{i=0}^{\ell-2} r_i$. It computes $C_i = g^{2^i u_i} h^{r_i}$ for $i = 0$ to ℓ. Note that the set of $\{C_i\}_{i=0}^{\ell-1}$ satisfies the following equation:
$$C = \prod_{i=0}^{\ell-1} C_i$$

- Let $M = H(C, C_0, \ldots, C_{\ell-1})$ for some hash function H. For each i, the prover generates a ring signature on the ring $\mathcal{R}_i = \{C_i, C_i/g^{2^i}\}$ by invoking $\sigma_i = \Pi.\mathsf{Sign}(\mathcal{R}_i, r_i, M)$. The range proof π for C is $(C_0, \sigma_0, \ldots, C_{\ell-1}, \sigma_{\ell-1})$.

- Upon receiving a proof π, the verifier computes $M = H(C, C_0, \ldots, C_{\ell-1})$

and outputs accept if and only the following holds:

$$\begin{aligned} C &= \prod_{i=0}^{\ell-1} C_i \\ 1 &= \Pi.\mathsf{Verify}(\{C_0, C_0/g\}, M, \sigma_0) \\ &\vdots \\ 1 &= \Pi.\mathsf{Verify}(\{C_{\ell-1}, C_{\ell-1}/g^{2^{\ell-1}}\}, M, \sigma_{\ell-1}) \end{aligned}$$

Discussions. The monero range proof follows the folklore approach in bit decomposition while replacing the standard 0/1 OR proof for each commitment C_i with a range signature. The design philosophy, as explained in [133] is that:
"If C was a commitment to 1 then I do not know its discrete log, but C' becomes a commitment to 0 and I do know its discrete log (just the blinding factor). If C was a commitment to 0 I know its discrete log, and I don't for C'. If it was a commitment to any other amount, none of the results will be zero and I won't be able to sign.".

After confirming C_i can only be a commitment to 0 or 2^i, the equation $C = \prod_{i=0}^{\ell-1} C_i$ assures the verifier that C is a commitment to a number between 0 to $2^\ell - 1$.

5.3.2 Analysis of Monero's Range Proof

This is one of the core results of this paper in which we illustrate a flaw in the above design philosophy. As no security analysis is provided in [133] nor [132], the ring signature-based range proof may not be secure. Here we give evidence to support our claim by instantiating the ring signature-based range proof with a secure ring signature scheme, and show that existence of an extractor, in any setting, for the resulting range proof implies that the CDH problem is equivalent to the DL problem. Essentially it means it is highly unlikely that such an extractor can be constructed.

5.3.2.1 BKM Ring Signature.

First, we review the ring signature scheme from [137]. The BKM scheme works in group equipped with a bilinear map, $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$.

- KeyGen. Randomly picks a value $x \in_R \mathbb{Z}_q$, computes $Y = h^x$. Pick a hash function $H : \{0,1\}^* \to \mathbb{G}$. The public key is Y, H and the secret key is x.

- Sign. Let the input ring be (Y_0, H_0) and (Y_1, H_1). Without loss of generality, assume the signing key is x_0 such that $Y_0 = h^{x_0}$. To sign message M, the signer randomly picks $r \in_R \mathbb{Z}_q$ and computes

$$\sigma_1 = Y_1^{x_0}(H_0(M)H_1(M))^r, \quad \sigma_2 = h^r.$$

  The signature is $\sigma = (\sigma_1, \sigma_2)$.

- Verify. On input ring $\mathcal{R} = \{(Y_0, H_0), Y_1, H_1\}$, message M, signature (σ_1, σ_2), the verifier outputs 1 if and only if:

$$\hat{e}(Y_0, Y_1)\hat{e}(\sigma_2, H_0(M)H_1(M)) = \hat{e}(\sigma_1, h),$$

  and outputs 0 otherwise.

As shown in [137], this ring signature scheme is unforgeable under the CDH assumption in the standard model when H is a Waters' hash function. It is unconditionally anonymous.

5.3.2.2 An Impossibility Result.

Denote by $(\mathcal{P}, \mathcal{V})$ the non-interactive ring signature-based range proof using BKM ring signature as a building block.

We show that it is impossible to construct ppt extractor $\mathcal{E}$ for $(\mathcal{P}, \mathcal{V})$ capable of extracting witness r from proof Π such that $\mathcal{V}(\Pi)$ outputs accept. More formally, we prove that if ppt $\mathcal{E}$ exists, it is possible to show that the CDH problem is equivalent to the DL problem in the bilinear group pair. In other words, it is highly unlikely that $\mathcal{E}$ exists and thus Π could not be a proof-of-knowledge.

Theorem 1. If ppt $\mathcal{E}$ exists for $(\mathcal{P}, \mathcal{V})$, we show how to construct simulator $\mathcal{S}$ that can solve the DL problem given a CDH oracle.

Proof. We consider the case when $\ell = 1$, i.e., the ring signature-based range proof of which the value committed is either 0 or 1. Assume $(\mathcal{P}, \mathcal{V})$ is an non-interactive zero-knowledge proof-of-knowledge system,

$$NIZKPoK\{(r) : C = h^r \vee C/g = h^r\}.$$

It means there exists ppt $\mathcal{E}$ which can extract from any $\mathcal{P}'$ capable of outputting valid proofs (i.e. proof accepted by $\mathcal{V}$). We show how to construct ppt $\mathcal{S}$, having access to a CDH oracle, which can solve the DL problem through interaction with the ppt $\mathcal{E}$.

$\mathcal{S}$ receives Y as a problem instance and its goal is to output r such that $Y = h^r$. It flips a coin $b \in \{0, 1\}$ and set $C = g^b Y$. Then, it invoked its CDH oracle on input $C, C/g$ to obtain a value Y. It computes $M = C$, generates a random $t \in_R$ uses Y, t to produce a proof π as follows: Compute $\sigma_1 = Y(H_0(M)H_1(M))^t, \sigma_2 = h^t$. Output $\sigma = (\sigma_1, \sigma_2)$ as the proof that $C = h^r \vee C = gh^r$. Invoke $\mathcal{E}$ on σ to obtain witness r satisfying $C = h^r$ or $C = gh^r$. If $b = 0$ and $C = h^r$, $\mathcal{S}$ outputs r as the solution to the DL problem. Likewise, if $b = 1$ and $C = gh^r$, $\mathcal{S}$ outputs r as the solution to the DL problem. Since b is hidden from $\mathcal{E}$, with probability $1/2$, $\mathcal{S}$ is able to solve the DL problem. □

Under the assumption that CDH $\not\Leftarrow$ DL, the above theorem implies that no $\mathcal{E}$ exists. In other words, $(\mathcal{P}, \mathcal{V})$ cannot be a proof-of-knowledge.

5.3.2.3 Additional Observations.

It is straightforward to say that $(\mathcal{P}, \mathcal{V})$ is a proof (but not a proof of knowledge) because the statement $C = h^r \vee C/g = h^r$ is always true. In other words, such r always exists. Furthermore, $(\mathcal{P}, \mathcal{V})$ is witness-indistinguishable because the underlying ring signature is anonymous. However, it is not clear how a zero-knowledge simulator could be constructed and thus it is unclear whether or not $(\mathcal{P}, \mathcal{V})$ is zero-knowledge.

Thus, we conclude that $(\mathcal{P}, \mathcal{V})$ is a witness-indistinguishable proof, but not a witness-indistinguishable proof-of-knowledge, that C is a commitment to 0 or 1 . Whether or not it is zero-knowledge remains unclear.

5.4 Improved Range Proof Protocol

In this section, we present our improved range proof for Monero. Our range proof admits a formal security proof and is even more efficient than the original range proof in Monero.

Formally, the improved (non-interactive) range proof for Monero, which proves that the committed value in a Peterson commitment lies in an interval $[0, 2^l - 1]$ for some l, also works in a group $\mathbb{G}$ of order q, which is generated with a security parameter 1^λ. Let g, h be two random generators of $\mathbb{G}$, which can be sampled with some public randomness and also serve as the public key of the Peterson commitment. Let $\mathtt{H}$ be a cryptographic hash function. The common reference string of the range proof is $crs = (\mathbb{G}, q, g, h, \mathtt{H})$. The range proof Π consists of two algorithms, which are formally described as follows:

- **Prove.** On input an integer l and a commitment $C = g^u h^r$ as well as the value u and the randomness r for the commitment, where $u \in [0, 2^l - 1]$, the prove algorithm works as follows.

  1. The prove algorithm decomposes the value u into a binary vector $\boldsymbol{u} = (u_0, \ldots, u_{l-1})$, where $u_i \in \{0, 1\}$ for $i \in [0, l-1]$, and $u = \sum_{i=0}^{l-1} 2^i u_i$.
  2. It samples $r_0, \ldots r_{l-2}$ uniformly at random from $\mathbb{Z}_q$ and computes $r_{l-1} = r - \sum_{i=0}^{l-2} r_i$.
  3. It generates l commitments $C_0, \ldots C_{l-1}$, where $C_i = g^{2^i u_i} h^{r_i}$ for $i \in [0, l-1]$.
  4. It sets $C_{i,0} = C_i$ and computes $C_{i,1} = C_i / g^{2^i}$ for $i \in [0, l-1]$.
  5. For $i \in [0, l-1]$, it samples $c_{i,\bar{u}_i}, \alpha_i, z_{i,\bar{u}_i} \xleftarrow{\$} \mathbb{Z}_q$, and computes $T_{i,u_i} = h^{\alpha_i}$ and $T_{i,\bar{u}_i} = h^{z_{i,\bar{u}_i}} C_{i,\bar{u}_i}^{c_{i,\bar{u}_i}}$.
  6. It computes the challenge $c = \mathtt{H}(C, C_0, \ldots, C_{l-1}, T_{0,0}, T_{0,1}, \ldots, T_{l-1,0}, T_{l-1,1})$, and for $i \in [0, l-1]$, it computes $c_{i,u_i} = c - c_{i,\bar{u}_i} \mod q$, and $z_{i,u_i} = \alpha_i - c_{i,u_i} r_i$.
  7. It outputs the proof $\pi = (\{C_i\}_{i\in[0,l-2]}, \{c_{i,0}, c_{i,1}\}_{i\in[0,l-1]}, \{z_{i,0}, z_{i,1}\}_{i\in[0,l-1]})$.

  Here, in Step 1 to Step 3, the prove algorithm decompose the commitment C into l auxiliary commitment, and in Step 4 to Step 7, it generates the proof proving that each auxiliary commitment C_i commits either 0 or 2^i for $i \in [0, l-1]$.

- **Verify.** On input a proof $\pi = (\{C_i\}_{i\in[0,l-2]}, \{c_{i,0}, c_{i,1}\}_{i\in[0,l-1]}, \{z_{i,0}, z_{i,1}\}_{i\in[0,l-1]})$, an integer l, and a commitment C, the verify algorithm works as follows.

1. The verify algorithm computes $C_{l-1} = C/(\prod_{i=0}^{l-2} C_i)$.
2. For $i \in [0, l-1]$, it computes $C'_i = C_i/g^{2^i}$, $T_{i,0} = h^{z_{i,0}} C_i^{c_{i,0}}$ and $T_{i,1} = h^{z_{i,1}} C_i'^{c_{i,1}}$.
3. It computes $c = \mathtt{H}(C, C_0, \ldots, C_{l-1}, T_{0,0}, T_{0,1}, \ldots, T_{l-1,0}, T_{l-1,1})$.
4. For $i \in [0, l-1]$, it checks if $c = c_{i,0} + c_{i,1} \mod q$, and outputs 1 iff the equation holds for all $i \in [0, l-1]$.

5.4.1 Security Proof

Theorem 2. The range proof Π is a secure range proof, namely, a secure NIZKPoK system proving that the prover knows an open of a commitment that lies in a particular interval, assuming that $\mathtt{H}$ is modelled as the random oracle.

Proof. We prove Theorem 2 by proving that Π has completeness, special soundness and zero-knowledge property.

5.4.1.0.1 Completeness. For any $l \in \mathbb{Z}$, $u, r \in \mathbb{Z}_q$ and $C \in \mathbb{G}$ that $C = g^u h^r$ and $u \in [0, 2^l]$, let $\pi \leftarrow Prove(l, C, u, r)$, where $\pi = (\{C_i\}_{i\in[0,l-2]}, \{c_{i,0}, c_{i,1}\}_{i\in[0,l-1]}, \{z_{i,0}, z_{i,1}\}_{i\in[0,l-1]})$, let $\boldsymbol{u} = (u_0, \ldots, u_{l-1}), r_0, \ldots, r_{l-1}, C_{l-1}, C_{0,0}, C_{0,1} \ldots, C_{l-1,0}, C_{l-1,1}$, $\alpha_0, \ldots, \alpha_{l-1}, T_{0,0}, T_{0,1} \ldots, T_{l-1,0}, T_{l-1,1}, c$ be variables used in the prove algorithm, and let $\hat{C}_{l-1}, \hat{T}_{0,0}, \ldots \hat{T}_{l-1,1}, \hat{c}$ be variables used in the verify algorithm when running on input (l, C, π). As $u = \sum_{i=0}^{l-1} 2^i u_i$ and $r = \sum_{i=0}^{l-1} r_i$, we have $\prod_{i=0}^{l-1} C_i = g^{\sum_{i=0}^{l-1} 2^i u_i} h^{\sum_{i=0}^{l-1} r_i} = g^u h^r = C$, i.e. $C_{l-1} = \hat{C}_{l-1}$. Also, for each $i \in [0, l-1]$, $T_{i,\bar{u}_i} = h^{z_{i,\bar{u}_i}} C_{i,\bar{u}_i}^{c_{i,\bar{u}_i}} = \hat{T}_{i,\bar{u}_i}$ by definition and $T_{i,u_i} = h^{\alpha_i} = h^{z_{i,u_i} + c_{i,u_i} r_i} = h^{z_{i,u_i}} C_{i,u_i}^{c_{i,u_i}} = \hat{T}_{i,u_i}$. So, $c = \hat{c}$, thus, for $i \in [0, l-1]$, we have $c_{i,0} + c_{i,1} = c = \hat{c} \mod q$. In summary, the verify algorithm will output 1 on input (l, C, π).

5.4.1.0.2 Special Soundness. On input two valid proofs $\check{\pi} = (\{\check{C}_i\}_{i\in[0,l-2]}, \{\check{c}_{i,0}, \check{c}_{i,1}\}_{i\in[0,l-1]}, \{\check{z}_{i,0}, \check{z}_{i,1}\}_{i\in[0,l-1]})$ and $\hat{\pi} = (\{\hat{C}_i\}_{i\in[0,l-2]}, \{\hat{c}_{i,0}, \hat{c}_{i,1}\}_{i\in[0,l-1]}, \{\hat{z}_{i,0}, \hat{z}_{i,1}\}_{i\in[0,l-1]})$ for a statement (l, C), which is obtained by rewinding at the point answering the random oracle and each time the random oracle being answered with randomly sampled responses, [2] the extractor works as follows. First, as different responses,

[2] Here, w.l.o.g., we assume that the cheating prove algorithm only query the random oracle once, as the extractor can succeed in guessing the point where the prove algorithm query the

say, $\check{c}$ and $\hat{c}$, are answered for the random oracle query, and both $\check{\pi}$ and $\hat{\pi}$ are valid, for each $i \in [0, l-1]$, there exists b_i that $\check{c}_{i,b_i} \neq \hat{c}_{i,b_i}$. Here, we set $u_i^* = b_i$ (if both $\check{c}_{i,0} \neq \hat{c}_{i,0}$ and $\check{c}_{i,1} \neq \hat{c}_{i,1}$, we sets u_i^* arbitrarily.), and computes $u^* = \sum_{i=0}^{l-1} 2^i u_i^*$. Obviously, $u^* \in [0, 2^l - 1]$. Then, let $\check{C}_{l-1} = C/(\prod_{i=0}^{l-2} \check{C}_i)$ and $\hat{C}_{l-1} = C/(\prod_{i=0}^{l-2} \hat{C}_i)$, let $\check{C}_{i,0} = \check{C}_i$, $\check{C}_{i,1} = \check{C}_i/g^{2^i}$, $\hat{C}_{i,0} = \hat{C}_i$, and $\hat{C}_{i,1} = \hat{C}_i/g^{2^i}$ for $i \in [0, l-1]$, and let $\check{T}_{i,j} = h^{\check{z}_{i,j}} \check{C}_{i,j}^{\check{c}_{i,j}}$ and $\hat{T}_{i,j} = h^{\hat{z}_{i,j}} \hat{C}_{i,j}^{\hat{c}_{i,j}}$ for $i \in [0, l-1]$ and $j \in \{0, 1\}$. As the extractor rewinds after the prove algorithm queries the random oracle, we have $\check{C}_i = \hat{C}_i$ (and thus, $\check{C}_{i,j} = \hat{C}_{i,j}$) and $\check{T}_{i,j} = \hat{T}_{i,j}$ for $i \in [0, l-1]$ and $j \in \{0, 1\}$. So, we can use $C_{i,j}$ to denote both $\check{C}_{i,j}$ and $\hat{C}_{i,j}$ for $i \in [0, l-1]$ and $j \in \{0, 1\}$. Besides, we also have $h^{\check{z}_{i,u_i^*}} C_{i,u_i^*}^{\check{c}_{i,u_i^*}} = h^{\hat{z}_{i,u_i^*}} C_{i,u_i^*}^{\hat{c}_{i,u_i^*}}$ for $i \in [0, l-1]$. Thus, we have $C_{i,u_i^*} = h^{(\check{z}_{i,u_i^*} - \hat{z}_{i,u_i^*}) \cdot (\check{c}_{i,u_i^*} - \hat{c}_{i,u_i^*})^{-1}}$ for $i \in [0, l-1]$. Then, the extractor sets $r_i^* = (\check{z}_{i,u_i^*} - \hat{z}_{i,u_i^*}) \cdot (\check{c}_{i,u_i^*} - \hat{c}_{i,u_i^*})^{-1}$ for $i \in [0, l-1]$, and computes $r^* = \sum_{i=0}^{l-1} r_i^*$. Finally, as $g^{u^*} h^{r^*} = g^{\sum_{i=0}^{l-1} 2^i u_i^*} h^{\sum_{i=0}^{l-1} r_i^*} = \prod_{i=0}^{l-1} g^{2^i u_i^*} h^{r_i^*} = \prod_{i=0}^{l-1} g^{2^i u_i^*} C_{i,u_i^*} = \prod_{i=0}^{l-1} C_i = C$, the tuple (u^*, r^*) is exactly the witness that C^* commits a value lies in $[0, 2^l - 1]$.

5.4.1.0.3 Zero-Knowledge. On input an integer l and a commitment C, the simulator who controls the random oracle works as follows.

1. samples $C_0, \ldots, C_{l-2} \xleftarrow{\$} \mathbb{G}$ and computes $C_{l-1} = C/(\prod_{i=0}^{l-2} C_i)$.
2. samples $c \xleftarrow{\$} \mathbb{Z}_q$.
3. for $i \in [0, l-1]$, samples $c_{i,0} \xleftarrow{\$} \mathbb{Z}_q$ and computes $c_{i,1} = c - c_{i,0} \mod q$.
4. for $i \in [0, l-1]$ and $j \in \{0, 1\}$, samples $z_{i,j} \xleftarrow{\$} \mathbb{Z}_q$.
5. for $i \in [0, l-1]$, sets $C_{i,0} = C_i$ and $C_{i,1} = C_i/g^{2^i}$.
6. for $i \in [0, l-1]$ and $j \in \{0, 1\}$, computes $T_{i,j} = h^{z_{i,j}} C_{i,j}^{c_{i,j}}$.

random oracle for the challenge of its proof with non-negligible probability. Also, as the cheating prove algorithm could generate a valid proof with non-negligible probability ϵ, with probability $\sigma/2$ over the choice of its random tape, it could generate a valid proof with probability $\sigma/2$, where the latter probability takes over the response of the random oracle. So, even when the extractor fix the random tape of the cheating prove algorithm, it could success twice with non-negligible probability. Besides, we also assume that different responses are sampled for the random oracle, which occurs with all but a negligible probability.

	Prover	Verifier
This paper	5ℓ	4ℓ
This paper (optimised)	3.1ℓ	2.2ℓ
Monero (ANSL)	3ℓ	$2\ell + 2$

Table 5.1: Efficiency of improved range proof

7. sets the random oracle response on $(C, C_0, \ldots, C_{l-1}, T_{0,0}, T_{0,1}, \ldots, T_{l-1,0}, T_{l-1,1})$ to be c.

8. outputs $\pi = (\{C_i\}_{i\in[0,l-2]}, \{c_{i,0}, c_{i,1}\}_{i\in[0,l-1]}, \{z_{i,0}, z_{i,1}\}_{i\in[0,l-1]})$.

It is not hard to check that the simulated proof π is identically distributed to a proof generated by the prove algorithm honestly. That completes the proof. □

5.4.2 Efficiency Analysis

We count the number of exponentiations needed by the prover and the verifier to compute and verify a proof respectively. For a range of $[0, 2^\ell]$, the proof effort is linear in ℓ. This is the same as Monero asymptotically. The following table shows the comparison of our scheme with Monero using the ANSL ring signature as a building block. We would like to remark that there is no formal security analysis for the range proof of Monero. Our range proof can be optimised using the trick that a multi-base exponentiation with 2 bases can be computed in roughly 1.1 times as a single base exponentiation. Comparing with the folklore binary decomposition technique, our protocol saves one commitment in terms of space complexity. As shown in the improved protocol, the proof π only includes the commitment C_0, …, C_{I-2}. This trick works because the verifier could derive C_{l-1} using the equation that $C_{l-1} = C/(\prod_{i=0}^{l-2} C_i)$. In other words, our protocol has a slight edge compared with the range proof protocol of Monero in the sense that one less commitment is needed. The saving is quite small in practice since $\ell = 64$ in Monero's range proof.

5.5 Chapter Summary

As a promising approach to address the security and privacy issues of VANET, anonymous credential and its necessary component range proof are investigated. We specifically studied the range proof protocol of the cryptocurrency Monero by analyzing the security aspect of rang proof protocol, pointing out that the design philosophy of its range proof is not likely to lead to a secure range proof. Then, we design a new range proof protocol and present a formal security proof. And, it is also illustrated that the efficiency is also comparable to that of Monero. Moreover, the improved protocol is compatible with Monero's wallet and the algebraic structure, thus does not require massive modification in the codebase. Hence, our proposed range proof protocol for Monero is practical and secure.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis aims to address the information security and privacy issues in VANETs. The first chapter introduces the various aspects of VANETs, including its architecture, network characteristics, applications, security and privacy attacks and requirements. Then, the second chapter specifically presents a review on the message authentication mechanisms and anonymous credentials, which are the major research focus of this thesis. Afterwards, we focus on developing secure and efficient signature schemes to realize message authentication in VANETs. Specifically, we propose an online/offline certificateless signature scheme to realize secure and efficient message authentication in VANETs. The security of the proposed certificateless signature is proven by a rigorous security proof. Compared with other certificateless signature schemes, the proposed scheme is more efficient in terms of computation cost of signature generation and verification. Moreover, the techniques of signature aggregation and batch verification are supported in the proposed authentication scheme to enhance authentication efficiency. Based on the proposed certificateless signature scheme for VANETs, the revocation problem is addressed by using a revocable certificateless signature scheme. The use of the KUNodes algorithm reduces the revocation workload of the key generation center. And, since the revocation is only decided by the key generation center without transprency, the blockchain technology is used in the authentication scheme to record the update of the revocation list and allow vehicles to check the update from

the blockchain, which improves the revocation transparency. Finally, in order to further enhance the overall authentication efficiency, an RSU-assisted verification process is proposed to allow RSUs to generate notification messages using cuckoo filters to assist the signature verification of nearby vehicles. Hence, the vehicles could use the notification messages to achieve message authentication instead of verifying the signatures by itself.

In the second part of the thesis, in order to further enhance the privacy of the drivers in VANETs, we study the anonymous credential. More specifically, the technique of range proof, which is a necessary component used in anonymous credential to protect identity privacy of drivers, is investigated. By studying the range proof protocol used in cryptocurrency Monero, the security flaws are identified. An improved range proof protocol is developed and its security is proven with a formal security proof. Efficiency and security analysis show that the improved range proof protocol is practical and secure.

6.2 Future Work

In the future, I will focus on investigating advanced cryptographic primitives and develop secure and practical protocols to enhance privacy in VANETs, especially ensuring unlinkability and minimum information disclosure. Firstly, I will continue to study anonymous credentials. Based on the improved version of range proof, I will further develop secure and efficient range prood protocols which can be used practically in the scenarios of VANETs, for example, proving the mileage of a vehicle lies in a certain range, proving the parking time of a vehicle is less than certain minutes in the parking navigation services, etc. I will also implement the developed protocol in VANETs and investigate the efficiency by conducting simulations. Moreover, apart from anonymous credential, attribute-based signature is another promising technique to achieve anonymous authentication with strong privacy. Hence, I will also investigate the technique of attribute-based signature schemes and employ it to develop anonymous authentication schemes for VANETs.

Appendix A

Security Proof

Typically, for the certificateless signature scheme proposed in Chapter 3, we define two types of security, namely Type-I security and Type-II security, which corresponds to two types of adversaries $\mathcal{A}_1$ and $\mathcal{A}_2$.

- **Type-I Adversary**: $\mathcal{A}_1$ can launch a public key replacement attack by replacing the public key of any vehicle with a value of its choice. $\mathcal{A}_1$ does not know the master secret key or the partial private key.
- **Type-II Adversary**: $\mathcal{A}_2$ acts as a malicious-but-passive KGC, which knows the master key and the partial private key, but cannot replace any user's public key.

Theorem 3. The proposed online/offline certificateless signature scheme in Chapter 3 in is $(\varepsilon, t, q_c, q_s, q_h)$- secure against the adversary $\mathcal{A}_1$ in the random oracle model, assuming that DL assumption hold in G, where q_c, q_h, q_s are the numbers of **Create**,**Hash** and **Sign** queries that the adversary is allowed to make.

Proof. Assume there is a probabilistic polynomial-time forger $\mathcal{A}_1$, we construct an algorithm $\mathcal{F}$ that make use of $\mathcal{A}_1$ to solve the discrete logarithm problem(DLP). Suppose $\mathcal{F}$ is given the DLP instance (P, Q) to compute $x \in Z_q^*$ such that $Q = xP$. $\mathcal{F}$ chooses a random identity ID^* as the challenged ID and answers the oracle queries from $\mathcal{A}_1$ as follows:

- **Setup(ID) query:** $\mathcal{F}$ sets $P_{pub} = Q$ and sends $\{P, p, q, E, G, H_2, H_3, P_{pub}\}$ to $\mathcal{A}_1$.

- **Create(ID) query:** $\mathcal{F}$ maintains a hash list L_c of tuple $(ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2)$. When $\mathcal{A}_1$ makes a query on ID, if ID is in L_c, $\mathcal{F}$ responds with $(ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2)$. Otherwise, $\mathcal{F}$ will simulate the oracle as follows. It randomly selects three value $a, b, c \in Z_q^*$, and sets $Q_{ID} = a \cdot P_{pub} + b \cdot P$, $vpk_{ID} = c \cdot P$, $psk_{ID} = b$, $x_{ID} = c$, $h_2 = H_2(ID||Q_{ID}) \leftarrow -a(modq)$. Then it responds with $(ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2)$, and inserts (ID, Q_{ID}, h_2) to L_{H_2}. Note that the equation $psk_{ID} \cdot P = Q_{ID} + h_2 \cdot P_{pub}$ holds, which means that the partial secret key is valid.

- H_2 **query:**When adversary makes a H_2 query with (ID, Q_{ID}), if ID is already in the hash list L_{H_2}, $\mathcal{F}$ just returns the corresponding h_2. Otherwise, $\mathcal{F}$ runs Create(ID) to get h_2, and send h_2 to $\mathcal{A}_1$.

- **Partial-Private-Key-Extract(ID) query:** If $ID = ID^*$, $\mathcal{F}$ stops the simulation. Otherwise, $\mathcal{F}$ checks the hash list L_c, if ID in the list, then $\mathcal{F}$ responds with psk_{ID}. If ID is not in L_c, $\mathcal{F}$ queries Create(ID) to get the psk_{ID}, and sends it to $\mathcal{A}_1$.

- **Public-Key(ID) query:** On receiving the query on ID, if ID is already in L_c, $\mathcal{F}$ responds with pk_{ID}=(Q_{ID}, vpk_{ID}). Otherwise, $\mathcal{F}$ queries Create(ID) to get the (Q_{ID}, vpk_{ID}), and sends it to $\mathcal{A}_1$.

- **Public-Key-Replacement(ID, pk'_{ID}) query:** $\mathcal{F}$ maintains a hash list L_R of tuple $(ID, d_i, Q_{ID}, x_{ID}, vpk_{ID})$. When $\mathcal{A}_1$ queries with (ID, pk'_{ID}), where $Q'_{ID} = d'_i \cdot P$,$vpk'_{ID} = x'_{ID} \cdot P$ and $pk'_{ID} = (Q'_{ID}, vpk'_{ID})$, $\mathcal{F}$ sets $Q_{ID} = Q'_{ID}$, $vpk_{ID} = vpk'_{ID}$, $psk_{ID} = \perp$, and $x_{ID} = x'_{ID}$. Then $\mathcal{F}$ updates the list L_R to be $(ID, d'_i, Q'_{ID}, vpk'_{ID}, x'_{ID})$

- H_3 **query:** $\mathcal{F}$ maintains a hash list L_{H_3} of tuple $(m, ID, R, vpk_{ID}, t, h_3)$. If the queries ID is in this list, $\mathcal{F}$ just responds with h_3. Otherwise it chooses a random h_3, sets $h_3 = H_3(m||ID||vpk_{ID}||R||t)$, add it into L_{H_3} and responds with h_3.

- **Sign(ID, m) query:** When $\mathcal{A}_1$ makes a sign query on (ID, m), if ID is in L_R, $\mathcal{F}$ generates random numbers $a, b, c \in Z_q^*$, sets $s = a, R = P, h_3 = H_3(m||ID||vpk_{ID}||R||t) \leftarrow (a - b - c)mod(q)$, inserts $(m, ID, R, vpk_{ID}, t, h_3)$

into L_{H_3}. The output signature is (R, s). If ID is not in L_R, $\mathcal{F}$ acts like the description of the scheme.

Finally, $\mathcal{A}_1$ outputs a forged signature σ=$(R, s_{\{1\}})$ on (ID, m), which satisfies the verification process of the verifier. If $ID \neq ID^*$,$\mathcal{F}$ fails and aborts. From the forking lemma in [138], $\mathcal{F}$ rewinds $\mathcal{A}_1$ to the point where it queries H_3, and use a different value. $\mathcal{A}_1$ will output another valid signatures (R, $s_{\{2\}}$) with the same R. Then we have $s_{\{i\}} \cdot P$=$h_{3_{\{i\}}} \cdot R + vpk_{ID} + Q_{ID} + h_2 \cdot P_{pub}$, where $i = 1, 2$

From these two linear equations, we can derive the value r by $\frac{s_2 - s_1}{h_{3_{\{2\}}} - h_{3_{\{1\}}}}$. Another rewind on H_2 will allow computation on x.

Probability Analysis: The simulation of Create(ID) oracle fails when the random oracle assignment $H_2(ID||Q_{ID})$ causes inconsistency, which happens with the probability at most q_h/q. The probability of successful simulation of q_c times is at least (1-(q_h/q))^{q_c}$\geqq$1-$(q_h q_c/q)$. Also, the simulation is successful q_h times with the probability at least (1-(q_h/q))^{q_h}$\geqq$1-(q_h^2/q). And $ID = ID^*$ with the probability $1/q_c$. Therefore, the overall successful simulation probability is (1-$q_h q_c/q$)(1-(q_h^2/q))$(1/q_c)\varepsilon$.

The time complexity of the algorithm $\mathcal{F}$ is dominated by the exponentiations performed in the Create and Sign queries, which is equal to $t + O(q_c + qs)S$, where S is the time of a scalar multiplication operation.

Theorem 4. The proposed online/offline certificateless signature scheme in Chapter 3 is $(\varepsilon, t, q_c, q_s, q_h)$- secure against the adversary $\mathcal{A}_2$ in the random oracle model, assuming that DL assumption hold in G, where q_c, q_h, q_s are the numbers of **Create**,**Hash** and **Sign** queries that the adversary is allowed to make.

Proof. Assume there is a probabilistic polynomial-time forger $\mathcal{A}_2$, we construct an algorithm $\mathcal{F}$ that make use of $\mathcal{A}_2$ to solve the discrete logarithm problem (DLP). Suppose $\mathcal{F}$ is given the DLP instance (P, Q) to compute $y \in Z_q^*$ such that $Q = yP$. $\mathcal{F}$ chooses a random identity ID^* as the challenged ID and answers the oracle queries from $\mathcal{A}_2$ as follows:

- **Setup(ID) query:** $\mathcal{F}$ sets $P_{pub} = x \cdot P, x \in Z_q^*$ and sends the parameters $\{P, p, q, E, G, H_2, H_3, P_{pub}\}$ to $\mathcal{A}_2$.
- **Create(ID) query:** $\mathcal{F}$ maintains a hash list L_c of tuple $(ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2)$. When $\mathcal{A}_1$ makes a query on ID, if ID is in L_c, $\mathcal{F}$ responds

with $(ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2)$. If $ID = ID^*$, $\mathcal{F}$ choose $a, b \in Z_q^*$ randomly, sets $Q_{ID} = aP, vpk_{ID} = Q, h_2 = H_2(ID||Q_{ID}) \leftarrow b, psk_{ID} = a+x \cdot h_2, x_{ID} = \perp$. If $ID \neq ID^*$, $\mathcal{F}$ select three random number a, b, c, and sets $Q_{ID} = aP, vpk_{ID} = bP, h_2 = H_2(ID||Q_{ID}) \leftarrow c, psk_{ID} = a + x \cdot h_2, x_{ID} = b$. Finally, $\mathcal{F}$ responds the query with $ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2$ and add ID, Q_{ID}, h_2 into the hash list L_{H_2}

- H_2 **query:**When adversary makes a H_2 query with (ID, Q_{ID}), if ID is already in the hash list L_{H_2}, $\mathcal{F}$ just returns the corresponding h_2. Otherwise, $\mathcal{F}$ runs Create(ID) to get h_2, and send h_2 to $\mathcal{A}_1$.

- **Partial-Private-Key-Extract(ID) query:** On receiving the query on ID, $\mathcal{F}$ checks the hash list L_c, if ID in the list, then $\mathcal{F}$ responds with psk_{ID}. If ID is not in L_c, $\mathcal{F}$ queries Create(ID) to get the psk_{ID}, and sends it to $\mathcal{A}_1$.

- **Public-Key(ID) query:** On receiving the query on ID, if ID is already in L_c, $\mathcal{F}$ responds with pk_{ID}=(Q_{ID}, vpk_{ID}). Otherwise, $\mathcal{F}$ queries Create(ID) to get the (Q_{ID}, vpk_{ID}), and sends it to $\mathcal{A}_1$.

- **Secret-Key-Extract(ID) query:** If $ID = ID^*$,$\mathcal{F}$ aborts the simulation. Otherwise, if ID is already in L_c, $\mathcal{F}$ responds with x_{ID}.If ID is not alreary in L_c, $\mathcal{F}$ runs Create(ID) to get $ID, Q_{ID}, vpk_{ID}, psk_{ID}, x_{ID}, h_2$, and sends $x_{ID}$to the adversary.

- H_3 **query:** $\mathcal{F}$ maintains a hash list L_{H_3} of tuple $(m, ID, R, vpk_{ID}, t, h_3)$. If the queries ID is in this list, $\mathcal{F}$ just responds with h_3. Otherwise it chooses a random h_3, sets $h_3 = H_3(m||ID||vpk_{ID}||R||t)$, add it into L_{H_3} and responds with h_3.

- **Sign(ID, m) query:** If $ID \neq ID^*$, $\mathcal{F}$ acts like the description of the scheme. Otherwise, $\mathcal{F}$ generates random numbers $a, b, f \in Z_q^*$, sets $s = a, h_3 = H_3(m||ID||vpk_{ID}||R||t) \leftarrow f, R = h_3^{-1} \cdot (bP_{pub} - Q)$, and responds with the signature as (R, s). This signature is valid as the equation $s \cdot P = h_3 \cdot R + Q_{ID} + vpk_{ID} + h_2 \cdot P_{pub}$ holds.

Finally, $\mathcal{A}_2$ outputs a forged signature σ=$(R, s_{\{1\}})$ on (ID, m), which satisfies the verification process of the verifier. From the forking lemma in [138], $\mathcal{F}$ rewinds $\mathcal{A}_2$

to the point where it queries H_3, and use a different value. $\mathcal{A}_2$ will output another valid signature (R, $s_{\{2\}}$) with the same R.Then we have:

$$s_{\{i\}} \cdot P = h_{3_{\{i\}}} \cdot R + vpk_{ID} + Q_{ID} + h_2 \cdot P_{pub}, \text{ where } i = 1, 2$$

$$s_{\{i\}} = h_{3_{\{i\}}} \cdot r + y + d_i + h_2 x, i = 1, 2$$

Only y, r are unknown. Hence, from these two linear equations, we can derive the two unknown value r, y, and output y as the solution of the DL problem.

Probability Analysis: The simulation of Create(ID) oracle fails when the random oracle assignment $H_2(ID||Q_{ID})$ causes inconsistency, which happens with the probability at most q_h/q. The probability of successful simulation of q_c times is at least $(1\text{-}(q_h/q))^{q_c} \geqq 1\text{-}(q_h q_c/q)$. Also, the simulation is successful q_h times with the probability at least $(1\text{-}(q_h/q))^{q_h} \geqq 1\text{-}(q_h^2/q)$. And $ID = ID^*$ with the probability $1/q_c$. Therefore, the overall successful simulation probability is $(1\text{-}q_h q_c/q)(1\text{-}(q_h^2/q))(1/q_c)\varepsilon$.

The time complexity of the algorithm $\mathcal{F}$ is dominated by the exponentiations performed in the Create and Sign queries, which is equal to $t + O(q_c + qs)S$, where S is the time of a scalar multiplication operation.

Bibliography

[1] I. Ali, A. Hassan, and F. Li, "Authentication and privacy schemes for vehicular ad hoc networks (vanets): A survey," *Vehicular Communications*, 2019.

[2] F. Bai and H. Krishnan, "Reliability analysis of dsrc wireless communication for vehicle safety applications," in *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pp. 355–362, IEEE, 2006.

[3] M. A. Moharrum and A. A. Al-Daraiseh, "Toward secure vehicular ad-hoc networks: a survey," *IETE Technical Review*, vol. 29, no. 1, pp. 80–89, 2012.

[4] X. Lin and R. Lu, *Vehicular ad hoc network security and privacy.* John Wiley & Sons, 2015.

[5] H. Hartenstein and L. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications magazine*, vol. 46, no. 6, 2008.

[6] M. S. Sheikh, J. Liang, and W. Wang, "A survey of security services, attacks, and applications for vehicular ad hoc networks (vanets)," *Sensors*, vol. 19, no. 16, p. 3589, 2019.

[7] S.-J. Horng, S.-F. Tzeng, P.-H. Huang, X. Wang, T. Li, and M. K. Khan, "An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks," *Information Sciences*, vol. 317, pp. 48–66, 2015.

[8] A. K. Jadoon, L. Wang, T. Li, and M. A. Zia, "Lightweight cryptographic techniques for automotive cybersecurity," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[9] C. V. S. C. Consortium *et al.*, "Vehicle safety communications project: Task 3 final report: identify intelligent vehicle safety applications enabled by dsrc," *National Highway Traffic Safety Administration, US Department of Transportation, Washington DC*, 2005.

[10] S. Olariu and M. C. Weigle, *Vehicular networks: from theory to practice*. Chapman and Hall/CRC, 2009.

[11] J.-P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security & Privacy*, vol. 2, no. 3, pp. 49–55, 2004.

[12] K. Sha, Y. Xi, W. Shi, L. Schwiebert, and T. Zhang, "Adaptive privacy-preserving authentication in vehicular networks," in *Communications and Networking in China, 2006. ChinaCom'06. First International Conference on*, pp. 1–8, IEEE, 2006.

[13] F. A. Ghaleb, M. Razzaque, and I. F. Isnin, "Security and privacy enhancement in vanets using mobility pattern," in *Ubiquitous and Future Networks (ICUFN), 2013 Fifth International Conference on*, pp. 184–189, IEEE, 2013.

[14] F. Schaub, Z. Ma, and F. Kargl, "Privacy requirements in vehicular communication systems," in *Computational Science and Engineering, 2009. CSE'09. International Conference on*, vol. 3, pp. 139–145, IEEE, 2009.

[15] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of computer security*, vol. 15, no. 1, pp. 39–68, 2007.

[16] P. Kamat, A. Baliga, and W. Trappe, "Secure, pseudonymous, and auditable communication in vehicular ad hoc networks," *Security and Communication Networks*, vol. 1, no. 3, pp. 233–244, 2008.

[17] H. Moustafa and Y. Zhang, *Vehicular networks: techniques, standards, and applications*. Auerbach publications, 2009.

[18] R. Cramer, *Advances in Cryptology-EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, vol. 3494. Springer Science & Business Media, 2005.

[19] A. Juels, D. Catalano, and M. Jakobsson, "Coercion-resistant electronic elections," in *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, pp. 61–70, ACM, 2005.

[20] T. Koens, C. Ramaekers, and C. van Wijk, "Efficient zero-knowledge range proofs in ethereum,"

[21] A. K. Malhi, S. Batra, and H. S. Pannu, "Security of vehicular ad-hoc networks: A comprehensive survey," *Computers & Security*, p. 101664, 2019.

[22] F. Dötzer, "Privacy issues in vehicular ad hoc networks," in *International Workshop on Privacy Enhancing Technologies*, pp. 197–209, Springer, 2005.

[23] J. Camenisch and T. Groß, "Efficient attributes for anonymous credentials," *ACM Transactions on Information and System Security (TISSEC)*, vol. 15, no. 1, p. 4, 2012.

[24] X. S. Shen, "Cloud-based privacy-preserving parking navigation through vehicular communications," in *Security and Privacy in Communication Networks: 12th International Conference, SecureComm 2016, Guangzhou, China, October 10-12, 2016, Proceedings*, vol. 198, p. 85, Springer, 2017.

[25] J. Y. Choi, M. Jakobsson, and S. Wetzel, "Balancing auditability and privacy in vehicular networks," in *Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks*, pp. 79–87, 2005.

[26] 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, 2008.

[27] M.-C. Chuang and J.-F. Lee, "Team: Trust-extended authentication mechanism for vehicular ad hoc networks," *IEEE systems journal*, vol. 8, no. 3, pp. 749–758, 2013.

[28] W. Rhim, "A study on mac-based efficient message authentication scheme for vanet," *Hanyang University, Seoul, South Korea*, 2012.

[29] C. Hu, T. W. Chim, S.-M. Yiu, L. C. Hui, and V. O. Li, "Efficient hmac-based secure communication for vanets," *Computer Networks*, vol. 56, no. 9, pp. 2292–2303, 2012.

[30] X. Zhu, Y. Lu, X. Zhu, and S. Qiu, "Lightweight and scalable secure communication in vanet," *International Journal of Electronics*, vol. 102, no. 5, pp. 765–780, 2015.

[31] M. Wang, D. Liu, L. Zhu, Y. Xu, and F. Wang, "Lespp: lightweight and efficient strong privacy preserving authentication scheme for secure vanet communication," *Computing*, vol. 98, no. 7, pp. 685–708, 2016.

[32] Z. Benyamina, K. Benahmed, and F. Bounaama, "Anel: A novel efficient and lightweight authentication scheme for vehicular ad hoc networks," *Computer Networks*, vol. 164, p. 106899, 2019.

[33] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "The tesla broadcast authentication protocol," *Rsa Cryptobytes*, vol. 5, no. 2, pp. 2–13, 2002.

[34] Q. Li and W. Trappe, "Staggered tesla: a multicast authentication scheme resistant to dos attacks," in *GLOBECOM'05. IEEE Global Telecommunications Conference, 2005.*, vol. 3, pp. 6–pp, IEEE, 2005.

[35] P. Ning, A. Liu, and W. Du, "Mitigating dos attacks against broadcast authentication in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 4, no. 1, pp. 1–35, 2008.

[36] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, and efficient vanet authentication," *Journal of Communications and Networks*, vol. 11, no. 6, pp. 574–588, 2009.

[37] M. H. Jahanian, F. Amin, and A. H. Jahangir, "Analysis of tesla protocol in vehicular ad hoc networks using timed colored petri nets," in *2015 6th International Conference on Information and Communication Systems (ICICS)*, pp. 222–227, IEEE, 2015.

[38] C. Lyu, D. Gu, Y. Zeng, and P. Mohapatra, "Pba: Prediction-based authentication for vehicle-to-vehicle communications," *IEEE transactions on dependable and secure computing*, vol. 13, no. 1, pp. 71–83, 2015.

[39] S. Bao, W. Hathal, H. Cruickshank, Z. Sun, P. Asuquo, and A. Lei, "A lightweight authentication and privacy-preserving scheme for vanets using tesla and bloom filters," *ICT Express*, vol. 4, no. 4, pp. 221–227, 2018.

[40] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pp. 11–21, 2005.

[41] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in vanet," in *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, pp. 19–28, 2007.

[42] 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.

[43] A. Wasef, R. Lu, X. Lin, and X. Shen, "Complementing public key infrastructure to secure vehicular ad hoc networks [security and privacy in emerging wireless networks]," *IEEE Wireless Communications*, vol. 17, no. 5, pp. 22–28, 2010.

[44] A. Wasef and X. Shen, "Emap: Expedite message authentication protocol for vehicular ad hoc networks," *IEEE transactions on Mobile Computing*, vol. 12, no. 1, pp. 78–89, 2011.

[45] C. Gañán, J. L. Muñoz, O. Esparza, J. Mata-Díaz, and J. Alins, "Pprem: privacy preserving revocation mechanism for vehicular ad hoc networks," *Computer Standards & Interfaces*, vol. 36, no. 3, pp. 513–523, 2014.

[46] C. Gañán, J. L. Muñoz, O. Esparza, J. Mata-Díaz, and J. Alins, "Epa: An efficient and privacy-aware revocation mechanism for vehicular ad hoc networks," *Pervasive and Mobile Computing*, vol. 21, pp. 75–91, 2015.

[47] N. Islam, "Certificate revocation in vehicular ad hoc networks: a novel approach," in *2016 International Conference on Networking Systems and Security (NSysS)*, pp. 1–5, IEEE, 2016.

[48] M. A. S. Junior, E. L. Cominetti, H. K. Patil, J. Ricardini, L. Ferraz, and M. V. Silva, "Privacy-preserving method for temporarily linking/revoking pseudonym certificates in vanets," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 1322–1329, IEEE, 2018.

[49] M. Khodaei and P. Papadimitratos, "Efficient, scalable, and resilient vehicle-centric certificate revocation list distribution in vanets," in *Proceedings of the 11th ACM conference on security & privacy in wireless and mobile networks*, pp. 172–183, 2018.

[50] 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, 2011.

[51] A. Boualouache and S. Moussaoui, "Tapcs: Traffic-aware pseudonym changing strategy for vanets," *Peer-to-Peer networking and Applications*, vol. 10, no. 4, pp. 1008–1020, 2017.

[52] L. Benarous, B. Kadri, and S. Boudjit, "Alloyed pseudonym change strategy for location privacy in vanets," in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–6, IEEE, 2020.

[53] D. Chaum and E. Van Heyst, "Group signatures," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 257–265, Springer, 1991.

[54] 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.

[55] J. Guo, J. P. Baugh, and S. Wang, "A group signature based secure and privacy-preserving vehicular communication framework," in *2007 Mobile Networking for Vehicular Environments*, pp. 103–108, IEEE, 2007.

[56] Y. Hao, Y. Cheng, C. Zhou, and W. Song, "A distributed key management framework with cooperative message authentication in vanets," *IEEE Journal on selected areas in communications*, vol. 29, no. 3, pp. 616–629, 2011.

[57] K. Lim, K. M. Tuladhar, X. Wang, and W. Liu, "A scalable and secure key distribution scheme for group signature based authentication in vanet," in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pp. 478–483, IEEE, 2017.

[58] J. Shao, X. Lin, R. Lu, and C. Zuo, "A threshold anonymous authentication protocol for vanets," *IEEE Transactions on vehicular technology*, vol. 65, no. 3, pp. 1711–1720, 2015.

[59] M. Alimohammadi and A. A. Pouyan, "Sybil attack detection using a low cost short group signature in vanet," in *2015 12th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC)*, pp. 23–28, IEEE, 2015.

[60] L. Zhang, Q. Wu, B. Qin, J. Domingo-Ferrer, and B. Liu, "Practical secure and privacy-preserving scheme for value-added applications in vanets," *Computer Communications*, vol. 71, pp. 50–60, 2015.

[61] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the theory and application of cryptographic techniques*, pp. 47–53, Springer, 1984.

[62] C. Zhang, R. Lu, X. Lin, P.-H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *IEEE INFOCOM 2008-The 27th Conference on Computer Communications*, pp. 246–250, IEEE, 2008.

[63] J. Sun, C. Zhang, Y. Zhang, and Y. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1227–1239, 2010.

[64] K.-A. Shim, "Cpas: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 4, pp. 1874–1883, 2012.

[65] N.-W. Lo and J.-L. Tsai, "An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 5, pp. 1319–1328, 2015.

[66] L. Zhang, C. Hu, Q. Wu, J. Domingo-Ferrer, and B. Qin, "Privacy-preserving vehicular communication authentication with hierarchical aggregation and fast response," *IEEE Transactions on Computers*, vol. 65, no. 8, pp. 2562–2574, 2015.

[67] S. Wang and N. Yao, "Liap: A local identity-based anonymous message authentication protocol in vanets," *Computer Communications*, vol. 112, pp. 154–164, 2017.

[68] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *International conference on the theory and application of cryptology and information security*, pp. 452–473, Springer, 2003.

[69] S. Mohanty, D. Jena, and S. K. Panigrahy, "A secure rsu-aided aggregation and batch-verification scheme for vehicular networks," in *Intemational Conference on Soft Computing and its Applications (ICSCA2012), pp174-178*, 2012.

[70] J. Cui, J. Zhang, H. Zhong, R. Shi, and Y. Xu, "An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks," *Information Sciences*, vol. 451, pp. 1–15, 2018.

[71] F. Wang, Y. Xu, H. Zhang, Y. Zhang, and L. Zhu, "2flip: A two-factor lightweight privacy-preserving authentication scheme for vanet," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 2, pp. 896–911, 2015.

[72] S. Jiang, X. Zhu, and L. Wang, "An efficient anonymous batch authentication scheme based on hmac for vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2193–2204, 2016.

[73] F. R. Asl and R. Samavi, "Synorm: Symmetric non repudiated message authentication in vehicular ad hoc networks," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, IEEE, 2017.

[74] U. Rajput, F. Abbas, H. Eun, and H. Oh, "A hybrid approach for efficient privacy-preserving authentication in vanet," *IEEE Access*, vol. 5, pp. 12014–12030, 2017.

[75] S. Tangade, S. S. Manvi, and P. Lorenz, "Decentralized and scalable privacy-preserving authentication scheme in vanets," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8647–8655, 2018.

[76] D. Chaum, "Security without identification: Transaction systems to make big brother obsolete," *Communications of the ACM*, vol. 28, no. 10, pp. 1030–1044, 1985.

[77] M. Gerlach and F. Guttler, "Privacy in vanets using changing pseudonyms-ideal and real," in *2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*, pp. 2521–2525, IEEE, 2007.

[78] A. I. González-Tablas, A. Alcaide, J. M. de Fuentes, and J. Montero, "Privacy-preserving and accountable on-the-road prosecution of invalid vehicular mandatory authorizations," *Ad hoc networks*, vol. 11, no. 8, pp. 2693–2709, 2013.

[79] D. Förster, F. Kargl, and H. Löhr, "Puca: A pseudonym scheme with user-controlled anonymity for vehicular ad-hoc networks (vanet)," in *2014 IEEE Vehicular Networking Conference (VNC)*, pp. 25–32, IEEE, 2014.

[80] J. M. d. Fuentes García Romero de Tejada, L. González Manzano, J. Serna Olvera, and F. Veseli, "Assessment of attribute-based credentials for privacy-preserving road traffic services in smart cities," 2017.

[81] A. Singh and H. C. S. Fhom, "Restricted usage of anonymous credentials in vehicular ad hoc networks for misbehavior detection," *International Journal of Information Security*, vol. 16, no. 2, pp. 195–211, 2017.

[82] G. Neven, G. Baldini, J. Camenisch, and R. Neisse, "Privacy-preserving attribute-based credentials in cooperative intelligent transport systems," in *2017 IEEE Vehicular Networking Conference (VNC)*, pp. 131–138, IEEE, 2017.

[83] K. Peng and L. Yi, "Studying a range proof technique—exception and optimisation," in *International Conference on Cryptology in Africa*, pp. 328–341, Springer, 2013.

[84] R. Chaabouni, H. Lipmaa, and B. Zhang, "A non-interactive range proof with constant communication.," in *Financial Cryptography*, vol. 2012, pp. 179–199, Springer, 2012.

[85] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM Journal on computing*, vol. 18, no. 1, pp. 186–208, 1989.

[86] E. F. Brickell, D. Chaum, I. B. Damgård, and J. van de Graaf, "Gradual and verifiable release of a secret," in *Conference on the Theory and Application of Cryptographic Techniques*, pp. 156–166, Springer, 1987.

[87] A. Chan, Y. Frankel, and Y. Tsiounis, "Easy come—easy go divisible cash," *Advances in Cryptology—EUROCRYPT'98*, pp. 561–575, 1998.

[88] W. Mao, "Guaranteed correct sharing of integer factorization with off-line shareholders," in *Public Key Cryptography*, pp. 60–71, Springer, 1998.

[89] H. Lipmaa, N. Asokan, and V. Niemi, "Secure vickrey auctions without threshold trust," in *International Conference on Financial Cryptography*, pp. 87–101, Springer, 2002.

[90] F. Boudot, "Efficient proofs that a committed number lies in an interval," in *Advances in Cryptology—EUROCRYPT 2000*, pp. 431–444, Springer, 2000.

[91] H. Lipmaa, "On diophantine complexity and statistical zero-knowledge arguments," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 398–415, Springer, 2003.

[92] J. Camenisch, R. Chaabouni, *et al.*, "Efficient protocols for set membership and range proofs," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 234–252, Springer, 2008.

[93] R. Chaabouni, H. Lipmaa, and A. Shelat, "Additive combinatorics and discrete logarithm based range protocols.," in *ACISP*, vol. 10, pp. 336–351, Springer, 2010.

[94] A. Rial, M. Kohlweiss, and B. Preneel, "Universally composable adaptive priced oblivious transfer," *Pairing-Based Cryptography–Pairing 2009*, pp. 231–247, 2009.

[95] J. Groth, "Efficient zero-knowledge arguments from two-tiered homomorphic commitments," *Advances in Cryptology–ASIACRYPT 2011*, pp. 431–448, 2011.

[96] X. Cheng, C.-X. Wang, D. I. Laurenson, S. Salous, and A. V. Vasilakos, "An adaptive geometry-based stochastic model for non-isotropic mimo mobile-to-mobile channels," *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, 2009.

[97] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "Ecpp: Efficient conditional privacy preservation protocol for secure vehicular communications," in *IEEE INFOCOM 2008-The 27th Conference on Computer Communications*, pp. 1229–1237, IEEE, 2008.

[98] M. Bayat, M. Barmshoory, M. Rahimi, and M. R. Aref, "A secure authentication scheme for vanets with batch verification," *Wireless networks*, vol. 21, no. 5, pp. 1733–1743, 2015.

[99] S.-J. Horng, S.-F. Tzeng, Y. Pan, P. Fan, X. Wang, T. Li, and M. K. Khan, "b-specs+: Batch verification for secure pseudonymous authentication in vanet," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1860–1875, 2013.

[100] D. LIU, R.-h. SHI, S. ZHANG, and H. ZHONG, "Efficient anonymous roaming authentication scheme using certificateless aggregate signature in wireless network," *Journal on Communications*, vol. 37, no. 7, pp. 182–192, 2016.

[101] H. Zhong, S. Han, J. Cui, J. Zhang, and Y. Xu, "Privacy-preserving authentication scheme with full aggregation in vanet," *Information Sciences*, vol. 476, pp. 211–221, 2019.

[102] I. A. Kamil and S. O. Ogundoyin, "An improved certificateless aggregate signature scheme without bilinear pairings for vehicular ad hoc networks," *Journal of information security and applications*, vol. 44, pp. 184–200, 2019.

[103] D. H. Yum and P. J. Lee, "Generic construction of certificateless signature," in *Australasian Conference on Information Security and Privacy*, pp. 200–211, Springer, 2004.

[104] X.-x. Li, K.-f. Chen, and L. Sun, "Certificateless signature and proxy signature schemes from bilinear pairings," *Lithuanian Mathematical Journal*, vol. 45, no. 1, pp. 76–83, 2005.

[105] M. H. Au, Y. Mu, J. Chen, D. S. Wong, J. K. Liu, and G. Yang, "Malicious kgc attacks in certificateless cryptography," in *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, pp. 302–311, ACM, 2007.

[106] D. He, J. Chen, and R. Zhang, "An efficient and provably-secure certificateless signature scheme without bilinear pairings," *International Journal of Communication Systems*, vol. 25, no. 11, pp. 1432–1442, 2012.

[107] M. Tian and L. Huang, "Cryptanalysis of a certificateless signature scheme without pairings," *International Journal of Communication Systems*, vol. 26, no. 11, pp. 1375–1381, 2013.

[108] J.-L. Tsai, N.-W. Lo, and T.-C. Wu, "Weaknesses and improvements of an efficient certificateless signature scheme without using bilinear pairings," *International Journal of Communication Systems*, vol. 27, no. 7, pp. 1083–1090, 2014.

[109] K.-H. Yeh, C. Su, K.-K. R. Choo, and W. Chiu, "A novel certificateless signature scheme for smart objects in the internet-of-things," *Sensors*, vol. 17, no. 5, p. 1001, 2017.

[110] X. Jia, D. He, Q. Liu, and K.-K. R. Choo, "An efficient provably-secure certificateless signature scheme for internet-of-things deployment," *Ad Hoc Networks*, vol. 71, pp. 78–87, 2018.

[111] L. Zhang, B. Qin, Q. Wu, and F. Zhang, "Efficient many-to-one authentication with certificateless aggregate signatures," *Computer Networks*, vol. 54, no. 14, pp. 2482–2491, 2010.

[112] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 416–432, Springer, 2003.

[113] H. Xiong, Z. Guan, Z. Chen, and F. Li, "An efficient certificateless aggregate signature with constant pairing computations," *Information Sciences*, vol. 219, pp. 225–235, 2013.

[114] D. He, M. Tian, and J. Chen, "Insecurity of an efficient certificateless aggregate signature with constant pairing computations," *Information sciences*, vol. 268, pp. 458–462, 2014.

[115] S. Even, O. Goldreich, and S. Micali, "On-line/off-line digital signatures," in *Conference on the Theory and Application of Cryptology*, pp. 263–275, Springer, 1989.

[116] J. K. Liu, J. Baek, J. Zhou, Y. Yang, and J. W. Wong, "Efficient online/offline identity-based signature for wireless sensor network," *International Journal of Information Security*, vol. 9, no. 4, pp. 287–296, 2010.

[117] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2681–2691, 2015.

[118] A. K. Malhi and S. Batra, "An efficient certificateless aggregate signature scheme for vehicular ad-hoc networks," *Discrete Mathematics and Theoretical Computer Science*, vol. 17, no. 1, pp. 317–338, 2015.

[119] B. Fan, D. G. Andersen, M. Kaminsky, and M. D. Mitzenmacher, "Cuckoo filter: Practically better than bloom," in *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, pp. 75–88, ACM, 2014.

[120] H. S. Ju, D. Y. Kim, D. H. Lee, J. Lim, and K. Chun, "Efficient revocation of security capability in certificateless public key cryptography," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pp. 453–459, Springer, 2005.

[121] Y. Sun, F. Zhang, L. Shen, and R. H. Deng, "A revocable certificateless signature scheme.," *JCP*, vol. 9, no. 8, pp. 1843–1850, 2014.

[122] J. Zhang and X. Zhao, "An efficient revocable certificateless signature scheme," in *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pp. 1852–1857, IEEE, 2015.

[123] H. Du, Q. Wen, and S. Zhang, "A provably-secure outsourced revocable certificateless signature scheme without bilinear pairings," *IEEE Access*, vol. 6, pp. 73846–73855, 2018.

[124] T. W. Chim, S.-M. Yiu, L. C. Hui, and V. O. Li, "Specs: Secure and privacy enhancing communications schemes for vanets," *Ad Hoc Networks*, vol. 9, no. 2, pp. 189–203, 2011.

[125] S.-H. Kim and I.-Y. Lee, "A secure and efficient vehicle-to-vehicle communication scheme using bloom filter in vanets," *International Journal of Security and Its Applications*, vol. 8, no. 2, pp. 9–24, 2014.

[126] A. Malhi and S. Batra, "Privacy-preserving authentication framework using bloom filter for secure vehicular communications," *International Journal of Information Security*, vol. 15, no. 4, pp. 433–453, 2016.

[127] J. Cui, J. Zhang, H. Zhong, and Y. Xu, "Spacf: A secure privacy-preserving authentication scheme for vanet with cuckoo filter," *IEEE transactions on vehicular technology*, vol. 66, no. 11, pp. 10283–10295, 2017.

[128] T. Limbasiya and D. Das, "Secure message confirmation scheme based on batch verification in vehicular cloud computing," *Physical Communication*, vol. 34, pp. 310–320, 2019.

[129] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proceedings of the 15th ACM conference on Computer and communications security*, pp. 417–426, ACM, 2008.

[130] Y.-H. Hung, Y.-M. Tseng, and S.-S. Huang, "A revocable certificateless short signature scheme and its authentication application," *Informatica*, vol. 27, no. 3, pp. 549–572, 2016.

[131] J. Cui, L. Wei, J. Zhang, Y. Xu, and H. Zhong, "An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, 2019.

[132] S. Noether, A. Mackenzie, and M. C. Team, "Ring confidential transactions." Monero research lab report MRL-0005, 2 2016.

[133] G. Maxwell, "Confidential transactions." web, 6 2015.

[134] M. Abe, M. Ohkubo, and K. Suzuki, "1-out-of-n signatures from a variety of keys," in *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings* (Y. Zheng, ed.), vol. 2501 of *Lecture Notes in Computer Science*, pp. 415–432, Springer, 2002.

[135] U. M. Maurer and S. Wolf, "The relationship between breaking the diffie–hellman protocol and computing discrete logarithms," *SIAM Journal on Computing*, vol. 28, no. 5, pp. 1689–1721, 1999.

[136] C. Rackoff and D. R. Simon, "Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack," in *Annual International Cryptology Conference*, pp. 433–444, Springer, 1991.

[137] A. Bender, J. Katz, and R. Morselli, "Ring signatures: Stronger definitions, and constructions without random oracles." Cryptology ePrint Archive, Report 2005/304, 2005. `http://eprint.iacr.org/2005/304`.

[138] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *Journal of cryptology*, vol. 13, no. 3, pp. 361–396, 2000.