Scientific Reports

https://doi.org/10.1038/s41598-025-28854-1

Article in Press

Sybil-resistant and privacy-preserving authentication based on short-term pseudonym for internet of vehicles

Received: 8 February 2025

Accepted: 12 November 2025

Published online: 21 November 2025

Cite this article as: Zhang C., Chang X., Jia X. *et al.* Sybil-resistant and privacy-preserving authentication based on short-term pseudonym for internet of vehicles. *Sci Rep* (2025). https://doi.org/10.1038/s41598-025-28854-1

Chunhua Zhang, Xing Chang, Xiaojun Jia, Sulan Zhang & Xiaolin Wang

We are providing an unedited version of this manuscript to give early access to its findings. Before final publication, the manuscript will undergo further editing. Please note there may be errors present which affect the content, and all legal disclaimers apply.

If this paper is publishing under a Transparent Peer Review model then Peer Review reports will publish with the final article.

ARTICLE IN PRESS

© The Author(s) 2025. **Open Access** This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

ARTICLE IN PRESS

Sybil-Resistant and Privacy-Preserving Authentication Based on Short-Term Pseudonym for Internet of Vehicles

Chunhua Zhang[1,*], Xing Chang[2], Xiaojun Jia[1], Sulan Zhang[1], and Xiaolin Wang[1]

[1]College of Artificial Intelligence, Jiaxing University, Jiaxing, 314001, China
[2]College of Electronic and Information Engineering, Tongji University, Shanghai, 201804, China
[*]zhangchunhua89@yeah.net

ABSTRACT

By sharing basic safety messages (BSMs) containing driving information, internet of vehicles (IoVs) becomes an important part in cooperative intelligent transportation systems (C-ITS). Designing a secure, privacy-preserving and efficient authentication scheme is an imperative and challenging issue. Vehicles should use different anonymous identities to send BSMs to resist trajectory tracking attacks. However, malicious vehicles may use multiple identities at the same time to launch Sybil attacks. In terms of efficiency, some schemes use time-consuming computation operations such as bilinear pairing and have very costly computation time for verifying signatures. To solve these problems, we construct a Sybil-resistant and privacy-preserving (SRPP) authentication scheme. In SRPP, vehicles use different short-term pseudonyms in different road side unit (RSU) jurisdictions, and short-term pseudonyms need to be authorized by RSUs. In terms efficiency, SRPP scheme mainly uses computationally efficient multiplication operations and point addition operations on elliptic curve, and realizes batch verification. Security proof and analysis show that SRPP scheme can meet security requirements and resist multiple types of attacks such as replay and DoS. The performance evaluation demonstrates that compared with recent proposed schemes, SRPP scheme is practical in terms of computation time and communication overhead while resisting Sybil attacks and trajectory tacking attacks.

Introduction

To improve driving experience, road safety and traffic management, researchers have proposed internet of vehicles (IoVs). In IoVs, each vehicle includes real-time location, speed, and other driving information in basic safety messages (BSMs), which are shared with surrounding vehicles and infrastructure via wireless connection[1–3]. However, by exploiting BSM, attackers can launch several types of attacks, such as impersonation and Sybil attacks[4,5]. Therefore, in IoVs, ensuring the correctness and reliability of the exchanged BSMs are essential considerations that must be addressed[6–8]. In addition, since BSMs contain vehicle's identity, position, direction, etc., if the information is exploited by attackers, they may disclose vehicle's travel trajectory, occupation, etc., and hijack a vehicle[9]. Thus, the protection of vehicle's privacy is also an important part in the research field of message authentication in IoVs[7].

Pseudonym and digital signature are widely used techniques to address the above security and privacy issues in IoVs, and researchers proposed some anonymous message authentication schemes[10,11]. However, existing schemes still suffer some limitations[8]. Generally, traditional PKI-based schemes exist certificate management burden[12,13], and key escrow issue[14,15]. In addition, the security of some existing schemes depends on strong assumptions about the ideal tamper-proof devices (TPDs)[16,17]. Besides, some schemes are only designed for vehicle-to-infrastruction (V2I) communication. In fact, vehicle-to-vehicle (V2V) communication is also an important mode for vehicles to share traffic data, and it is urgent to propose authentication scheme for V2V communication. Moreover, existing authentication schemes are less concern about Sybil attacks and trajectory privacy simultaneously. For these reasons, security and privacy protection are still the critical factors restricting the development of IoVs[18].

Considering the above security, privacy protection and efficiency issues in IoVs, we proposes a Sybil-resistant and privacy-preserving (SRPP) authentication scheme. The principal contributions of our work are summarized as follows:

1) We construct a lightweight anonymity message authentication scheme by utilizing computationally efficient multiplication operations and point addition operations on elliptic curve, hash functions and XOR operations. By supporting batch mode, thus receivers can verify multiple signatures simultaneously and efficiently.

2) Furthermore, based on two hash chains, vehicles can generate short-term pseudonyms on demand, and short-term pseudonyms need to be authorized by RSUs. In this way, vehicles can use multiple pseudonyms during their travel, while limiting the abuse of pseudonyms. Thus, SRPP scheme can resist both trajectory tracking and Sybil attacks.

ARTICLE IN PRESS

3) Based on random oracle model, the security of SRPP scheme is proved, and the security in message authentication, dynamic pseudonym, un-linkability and conditional traceability, and resistance to various common attacks such as Sybil and tracking are illustrated. Moreover, the performance evaluation demonstrates that compared with other related work recently proposed, SRPP scheme has reasonable computation time cost in generating signature and verifying BSMs, and practical communication overhead in sending BSMs.

Related Works

Public Key Infrasturcture (PKI)-based Schemes

In PKI-based schemes, when a vehicle sends a BSM, it attaches a corresponding certificate to the message. The receiver needs to verify the certificate in BSM. In a traditional PKI-based scheme[12], each vehicle loads a large number of certificates from a trusted authority (TA) to its own TPD. To minimize the required storage for certificates, RSUs are used to assign short-lifetime pseudonyms and corresponding certificates to requesting vehicles[19]. By introducing fog computing, Kang et al.[20] transfered pseudonym management to resources at the network edge. Certificate revocation is also an issue in PKI based schemes. Wasef and Shen[21] realized revocation by updating the global key used to calculate hash message authentication code (HMAC). The time and resource cost of updating the global key is relatively high. Overall, in such schemes, certificate management increases the storage, communication and computation burden[13].

Identity (ID)-based Schemes

The ID-based schemes avoid the burden of certificate management, because the public key of each vehicle can be easily calculated from its identity[14]. In most ID-based schemes, each vehicle preloads system master secret key (SMSK), its real identity, and password from TA into its own TPD in initialization phase. He et al.[22] proposed a scheme that does not use bilinear pairing. Ali et al.[23] proposed an ID-based batch verification scheme for V2I communication by using bilinear pairing and general hash functions. Without using bilinear pairng, Ali et al.[24] proposed a scheme for V2V communication, but Song et al.[25] demonstrated that this scheme[24] is insecure.

On the whole, most ID-based schemes strongly assume that SMSK must be pre-loaded into each vehicle's TPD and that no attacker will be able to obtain SMSK from TPD[16]. In fact, by launching a side channel attack, an attacker may get SMSK[14]. As a result, using SMSK, the adversary can launch impersonation attacks, bogus message attacks, etc., and the whole system could be exposed to serious security risks[17]. In addition, ID-based schemes generally require a TA to generate and manage vehicles' private keys, which brings the new key escrow problem[26].

Certificateless Signature (CLS)-based Schemes

CLS-based scheme is considered to have the potential to solve the certificate management problem of the PKI-based schemes and the key escrow problem of the ID-based schemes[26]. To achieve secure V2I communication, scholars proposed some schemes. Based on bilinear paring, Horng et al.[26] proposed a CLAS-based scheme with conditional privacy-preserving in which each BSM uses a different pseudonym. Zhong et al.[16] used a trace authority (TRA) responsible for generating pseudonyms for vehicles and, if necessary, tracking real identities of vehicles. Han et al.[27] proposed a CLAS scheme that does not use bilinear pairing, named eCLAS, and proved the security of eCLAS in the random oracle model based on the computational difficulty of elliptic curve discrete logarithm problem (ECDLP). Zhu et al.[1] devised a pairing-free authentication scheme, which has better computational performance than previous authentication schemes, but Yang et al.[28] showed that the proposed scheme[1] is susceptible to coalition assaults launched by malicious RSUs and vehicles, as well as being insecure against public-key replacement (PKR) attacks. Yang et al.[29] put forward a re-cryptanalysis to the security of the CLAS scheme proposed by Zhu et al.[1], and then proposed a provably secure CLAS scheme with a new aggregate algorithm for IoVs. The main difference between the two schemes is that Yang et al.'s scheme[29] changes the input space of a hash function in the signature generation phase. However, the improved scheme[29] still only supports V2I communication, and a vehicle uses only one pseudonym, which makes the vehicle potentially vulnerable to trajectory tracking attacks.

Elliptic curve cryptography (ECC) can provide the same or even higher security level as RSA, achieving a smaller key size, faster computing speed, lower bandwidth and storage overhead. These features perfectly match the demanding requirements of low latency, high throughput and resource constraints in the IoVs. Based on ECC, Kamil and Ogundoyin[30] proposed a CLAS scheme for vehicular communications, and the security of this scheme was reduced to the computational difficulty assumption of ECDLP. Through analysis, Xiong et al.[15] found that the scheme[30] could not resist collusion attacks, and put forward an improvement scheme to resist collusion attacks, however, in this scheme, a vehicle uses only one pseudonym for a long time, so the vehicle's trajectory privacy is not protected[31]. Ali et al.[32] also proposed an ECC-based scheme for V2V communication that uses efficient general hash functions instead of time-consuming MapToPoint hash functions, but it does not provide unforgeability[18]. Thus, Zhou et al.[18] designed a new provably secure scheme that does not use time-consuming

ARTICLE IN PRESS

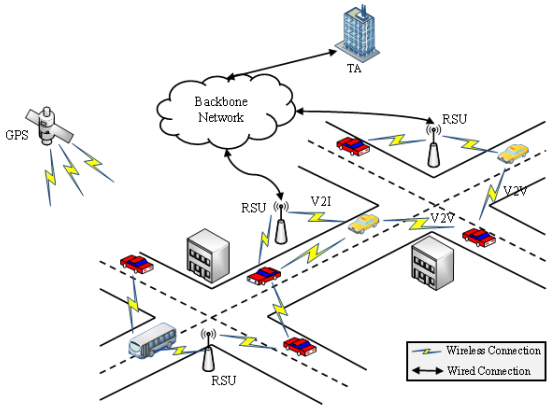


Figure 1. System model.

bilinear pairing and MapToPoint opereations, however, there is also the problem of travel trajectory disclosure. Based on ECC, Wu and Ye[33] proposed a CLAS scheme that does not use bilinear pairing, but there is also a risk of leaking travel trajectory.

Wu et al.[7] proposed a CLS scheme for V2V communication, whose batch verification method is based on bilinear pairing, so its computation time cost is higher compared with those schemes that do not use bilinear paring. During the registration phase, a vehicle applies to TA for a pool of pseudonyms. This provides an opportunity for malicious users to launch Sybil attacks. Based on CLAS and paring-free operations, Zhu et al.[8] proposed a lightweight conditional privacy-preserving identity authentication (LCPIA) scheme with fewer operations and less communication overhead. In LCPIA scheme, vehicles collaborate with TA to generate dynamic pseudonyms and public/private key pairs, and each vehicle may hold multiple usable pseudonyms which can be used by attackers to launch Sybil attacks. In addition, they did not discuss how TA sent the relevant cryptographic materials to a vehicle.

In general, although many important results have been achieved in the research of privacy-preserving authentication scheme for vehicular communications, how to resist Sybil attacks and trajectory tracking attacks while improving verification efficiency is still a challenge.

Preliminaries

System Model

The system model of the proposed SRPP scheme is comprised of a TA, several RSUs and intelligent vehicles[34,35], as shown in Fig. 1.

TA is the upper layer management center having the highest authority, storage, communication and computation capabilities, and has the highest level of security, that is, will not be compromised by attackers. It completes the system setup and the registration of RSUs and vehicles by offline means. TA generates long-term anonymous identities for each vehicle and reveals the true identities of malicious vehicles when needed. TA generally adopts the secure socket layer (SSL) protocol to communicate with RSUs[36].

RSUs are base-stations fixed along roadsides. They have less capabilities than TA and more capabilities than vehicles. In our SRPP scheme, RSUs are responsible for authorizing vehicles' short-term pseudonyms. A RSU communicates with TA as well as other RSUs in real time through secure SSL protocol, and with vehicles through dedicated short range communication (DSRC) protocol.

Each vehicle is required to register its real identity information in TA, and each vehicle can establish a secure channel for the registration. Vehicles periodically broadcast BSMs including driving-related status data (e.g., speed and location) through DSRC protocol. The secret materials used by vehicles to sign and verify messages are kept secret in their respective on-board TPD. Vehicles are untrusted and can launch fake messages, Sybil and other attacks.

Security Requirement

To meet the security and privacy requirements of vehicular communications, a practical BSM authentication scheme need to satisfy the following security goals[8].

1) Message Authentication: In V2V and V2I communications, vehicles and RSUs are exposed to wireless communication channels. To prevent adversaries in open channels from forging and tampering, any receiver in IoVs must have the capability to ensure that the received BSMs are sent from authorized devices and have not been modified by adversaries.

ARTICLE IN PRESS

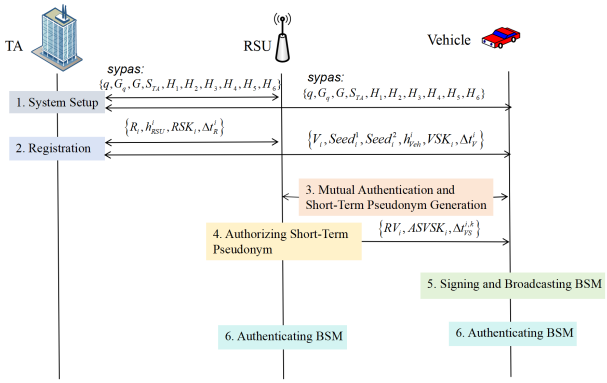


Figure 2. Framework of SRPP.

2) Dynamic Pseudonym: Pseudonym provides some protection for vehicles' real identity. However, if a vehicle uses only one pseudonym for a long time, adversaries can use that pseudonym to launch a trajectory tracking attack. Therefore, dynamic pseudonyms should be guaranteed in IoVs, so that different pseudonyms can be used in different areas to ensure greater privacy and security.

3) Un-Linkability: Any adversary cannot correctly link two or more pseudonyms to the same vehicle. In IoVs, unlinkability ensures that BSMs sent under different pseudonyms are uncorrelated.

4) Conditional Traceability: Conditional traceability requires that TA be able to track the real identity of a vehicle, and that no other entity has this ability.

5) Resistance Against Attacks: In this complex system environment, by analyzing or broadcasting BSMs, an attacker may launch multiple attacks such as tracking, replay, Sybil and denial of service (DoS). Resistance to these attacks is also a fundamental requirement for a BSM authentication scheme.

Mathematical Knowledge

The ECC belongs to public-key cryptography (PKC) designed on elliptic curves over finite fields[37]. The ECC requires a shorter key size than other PKC-oriented approaches while providing the same level of security.

Consider a finite field F_p with a large prime order p. Elliptic curve $E(a,b): y^2 = x^3 + ax + b \pmod p$ is defined on F_p, where $a, b \in F_p$ and $4a^3 + 27b^2 \neq 0$. Let G_q be a finite cyclic additive group on $E(a,b)$ with a generator point G, where q is a large prime order. Especially, the scalar multiplication on G_q is given as $mH = H + H + \cdots + H$ (m times), where $m \in Z_q^*$ and H is a random point on $E(a,b)$. Our SRPP scheme is based on the following computational complexity assumptions[38].

Definition 1: ECDLP: Given two random points $H, K \in G_q$, finding $x \in Z_q^*$ such that $K = x \cdot H$ is computationally hard.

Definition 2: Computational diffie-hellman problem (CDHP): Given $H, K \in G_q$, the computation of $(x \cdot y) \cdot G$ is hard, where $H = x \cdot G$, $K = y \cdot G$ with $x, y \in Z_q^*$.

Definition 3: Collision-resistance: Given a hash function $H() : X \to Y$, if it is impossible to find two distinct values a and b, where $a, b \in X$ and $a \neq b$, such that $H(a) = H(b)$, then it possesses collision-resistance.

Proposed SRPP Scheme

In this section, the overall framework of our proposed SRPP scheme and the implementation details are introduced.

Framework of SRPP

To protect privacy and defend against Sybil attacks, SRPP scheme supports each vehicle to use different short-term anonymous identities in different RSU jurisdictions. The framework of SRPP scheme consists of six phases, namely: 1) system setup; 2) registration; 3) mutual authentication and short-term pseudonym generation; 4) authorizing short-term pseudonym; 5) signing and broadcasting BSM; 6) authenticating BSM, as shown in Fig. 2. For illustration purpose, we show in TABLE 1 the definitions of notations used in the following descriptions.

System Setup

The system setup is accomplished by TA, which inputs a security parameter l and generates system parameters *sypas* and system secret key s_{TA} for the authentication system.

ARTICLE IN PRESS

Table 1. Notations and Definitions

Notations	Definitions
(G_q, q, G)	Elliptic curve parameters
$H_1, H_2, H_3, H_4, H_5, H_6$	Six secure hash functions
(s_{TA}, S_{TA})	The private key and public key of TA respectively
(RSU_i)	The i-th RSU
RID_i	The identification code of RSU_i
(r_i, R_i)	The private key and public key of RSU_i
Δt_R^i	The expiration date of R_i
RSK_i	The secret key of RSU_i
Veh_i	The i-th vehicle
VID_i	The real identification of Veh_i
$Seed_i^1, Seed_i^2$	Two random hash seeds generated by TA for Veh_i
(v_i, V_i, VSK_i)	The private key, public key and long-term secret key of Veh_i respectively
Δt_V^i	The expiration date of Veh_i
$SPID_{i,k}$	The k-th short pseudonym of Veh_i
$\{vpr_i = (va_i, SVSK_i), VPU_i = (VA_i, RV_i)\}$	The short-term private keys and corresponding public keys of Veh_i respectively

1) TA selects a cyclic additive group G_q with order q on the elliptic curve $E(F_p)$ according to the selected security parameter l, where p, q are large primes of length l-bits, and G is a generator point of G_q.

2) TA chooses a number $s_{TA} \in Z_q^*$ at random and computes $S_{TA} = s_{TA} \cdot G$ as its private key and public key respectively.

3) TA selects six one-way collision-resistant hash functions: $H_1 : \{0,1\}^* \times G_q \times \{0,1\}^* \to Z_q^*$, $H_2 : G_q \times \{0,1\}^* \to Z_q^*$, $H_3 : \{0,1\}^* \times G_q \times G_q \times \{0,1\}^* \times \{0,1\}^* \to Z_q^*$, $H_4 : \{0,1\}^* \to Z_q^*$, $H_5 : \{0,1\}^* \times G_q \times G_q \times \{0,1\}^* \to Z_q^*$, $H_6 : \{0,1\}^* \times \{0,1\}^* \times G_q \times \{0,1\}^* \to Z_q^*$. It is worth noting that vehicles communicate through wireless means and are facing long-term and complex threats. A system that relies on a single hash function, once this function is breached, will lead to large-scale and catastrophic security failures. Our approach of using multiple hash functions has significantly raised the threshold and cost of attacks, providing a stronger guarantee for secure vehicle communication and meeting the fundamental requirements of the IoV for high reliability and security resilience.

4) TA publishes the system public parameters $\{q, G_q, G, S_{TA}, H_1, H_2, H_3, H_4, H_5, H_6\}$, and keeps the s_{TA} secret.

ARTICLE IN PRESS

Registration

Before communicating with other entities in IoVs, any RSU and vehicle must register with TA. TA generates public key and secret key for RSUs and vehicles. TA provides long-term public key R_i and corresponding secret key RSK_i for a RSU, and provides long-term public key V_i, corresponding secret key VSK_i and two hash seeds for a vehicle.

1) RSU Registration

For the RSU_i, TA chooses a number $r_i \in Z_q^*$ at random and computes $R_i = r_i \cdot G$, $h_{RSU}^i = H_1\left(RID_i, R_i, \Delta t_R^i\right)$, $RSK_i = (r_i + s_{TA} \cdot h_{RSU}^i) \bmod q$, where R_i is long-term public key of RSU_i, RID_i is identification code of RSU_i, Δt_R^i is expiration date of R_i, and RSK_i is secret key of RSU_i. Then, the TA sends $\left\{R_i, h_{RSU}^i, RSK_i, \Delta t_R^i\right\}$ to RSU_i via a secure channel.

2) Vehicle Registration

For the vehicle Veh_i with real identity VID_i, TA generates two random hash seeds $Seed_i^1$ and $Seed_i^2$, and selects a random number $v_i \in Z_q^*$, and computes $V_i = v_i \cdot G$, $h_{Veh}^i = H_2(V_i, \Delta t_V^i)$, $VSK_i = \left(v_i + s_{TA} \cdot h_{Veh}^i\right) \bmod q$, where V_i is long-term public key of Veh_i, Δt_V^i is the expiration date of V_i, and VSK_i is the long-term secret key of Veh_i. Then, TA stores $\left\{VID_i, V_i, Seed_i^1, Seed_i^2, h_{Veh}^i, VSK_i, \Delta t_V^i\right\}$ in its database, and sends $\left\{V_i, Seed_i^1, Seed_i^2, h_{Veh}^i, VSK_i, \Delta t_V^i\right\}$ to Veh_i via a secure channel.

Mutual Authentication and Short-Term Pseudonym Generation

In SRPP scheme, short-term pseudonyms, which are used by vehicles when sending BSMs, should be authorized by RSUs. In this way, SRPP scheme prevents malicious users from abusing pseudonyms, thus making it resistant to Sybil attacks. Before a RSU authorizes a short-term pseudonym, the RSU and the vehicle authenticate that each other's identities are legitimate. Also, the vehicle generates short-term pseudonym $SPID_{i,k}$ and partial private key va_i.

ARTICLE IN PRESS

1) RSU Broadcasting Hello Message

The corresponding RSU, noted as RSU_i, broadcasts hello messages periodically, e.g., every 5s, which contain information such as identity and public key. Specifically, RSU_i randomly selects $ra_i \in Z_q^*$, computes $RA_i = ra_i \cdot G$, $RH_i = H_3(RID_i, R_i, RA_i, \Delta t_R^i, tim_h)$ and $\delta_R^i = (RSK_i + RH_i \cdot ra_i) \, mod \, q$, where tim_h is timestamp, and then broadcasts hello message $M_h : \{RID_i, R_i, RA_i, \Delta t_R^i, tim_h, \delta_R^i\}$.

2) Vehicle Authenticating Hello Message

To resist replay attacks, when Veh_i receives a hello message, it first checks tim_h to make sure the message is fresh, and checks Δt_R^i to make sure the RSU identity is unexpired. Next, Veh_i calculates $h_{RSU}^i = H_1(RID_i, R_i, \Delta t_R^i)$ and $RH_i = H_3(RID_i, R_i, RA_i, \Delta t_R^i, tim_h)$. Then, Veh_i checks if $\delta_R^i \cdot G \stackrel{?}{=} R_i + h_{RSU}^i \cdot S_{TA} + RH_i \cdot RA_i$ holds. The proof of the formula is demonstrated bellow.

$$\begin{aligned} \delta_R^i \cdot G &= (RSK_i + RH_i \cdot ra_i) \cdot G \\ &= \left(r_i + s_{TA} \cdot h_{RSU}^i + RH_i \cdot ra_i\right) \cdot G \\ &= r_i \cdot G + s_{TA} \cdot G \cdot h_{RSU}^i + RH_i \cdot ra_i \cdot G \\ &= R_i + h_{RSU}^i \cdot S_{TA} + RH_i \cdot RA_i \end{aligned} \tag{1}$$

If the formula holds, Veh_i stores $\{RID_i, R_i, \Delta t_R^i\}$ into its TPD.

3) Vehicle Generating Short-Term Pseudonym and Sending Request Message

Using two hash seeds distributed by TA, Veh_i can locally generate many short-term pseudonyms as needed. Note that, vehicles can request fresh hash seeds from TA when the regular maintenance of the system is coming, or the current hash seeds are no longer secure. Successively applying $H_4(\cdot)$ on $Seed_i^1$ and $Seed_i^2$, two hash chains can be formed as follows.

$$\begin{aligned} S_{i,k}^1 &= H_4^k(Seed_i^1), \\ S_{i,k}^2 &= H_4^k(Seed_i^2) \end{aligned} \tag{2}$$

The kth short-term pseudonym of Veh_i is generated by computing $SPID_{i,k} = H_4(S_{i,k}^1 \oplus S_{i,k}^2)$. Next, Veh_i sends a request message to RSU_i. Given that once an attacker obtains Veh_i's public key V_i, it can track Veh_i's trajectory. In order to avoid disclosing V_i, our scheme utilize XOR operation and CDHP based operation to conceal V_i. Specific operations are as follows.

Veh_i randomly chooses a number $va_i \in Z_q^*$ with secrecy, and computes $VA_i = va_i \cdot G$, $VAR_i = va_i \cdot R_i$, $AV_i = V_i \oplus VAR_i$, $VH_i = H_3(SPID_{i,k}, VA_i, AV_i, \Delta t_V^i, tim_r)$, and $\delta_V^i = (VSK_i + VH_i \cdot va_i) \bmod q$, where va_i is short-term partial private key, VA_i is short-term public key, tim_r is time-stamp. Then, Veh_i sends message $M_r : \left\{SPID_{i,k}, VA_i, AV_i, \Delta t_V^i, tim_r, \delta_V^i\right\}$ to RSU_i.

4) RSU Authenticating Request Message

When receiving the request message M_r, RSU_i first checks the timestamp tim_r in M_r, and then computes $V_i = AV_i \oplus r_i \cdot VA_i$, $h_{Veh}^i = H_2(V_i, \Delta t_V^i)$ and $VH_i = H_3(SPID_{i,k}, VA_i, AV_i, \Delta t_V^i, tim_r)$. After that, RSU_i checks if $\delta_V^i \cdot G \stackrel{?}{=} V_i + h_{Veh}^i \cdot S_{TA} + VH_i \cdot VA_i$ holds to authenticate the signature δ_V^i. The proof of the formula is demonstrated bellow.

$$\begin{aligned} \delta_V^i \cdot G &= (VSK_i + VH_i \cdot va_i) \cdot G \\ &= \left(v_i + s_{TA} \cdot h_{Veh}^i + VH_i \cdot va_i\right) \cdot G \\ &= v_i \cdot G + h_{Veh}^i \cdot s_{TA} \cdot G + VH_i \cdot va_i \cdot G \\ &= V_i + h_{Veh}^i \cdot S_{TA} + VH_i \cdot VA_i \end{aligned} \tag{3}$$

Authorizing Short-Term Pseudonym

In this phase, RSU generates short-term secret key $SVSK_i$ for vehicle Veh_i. SRPP scheme uses XOR operation and CDHP based operation to secure the secret key $SVSK_i$. Specifically, RSU_i generates the authentication materials associated with the short-term pseudonym for V_i. RSU_i sets the expiration date $\Delta t_{VS}^{i,k}$ for $SPID_{i,k}$, selects $rv_i \in Z_q^*$ randomly, and computes $RV_i = rv_i \cdot G$, $rh_i = H_5(SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k})$, $SVSK_i = (RSK_i + rh_i \cdot rv_i) \bmod q$, $ASVSK_i = SVSK_i \oplus rv_i \cdot VA_i$. Then, RSU_i sends $\{RV_i, ASVSK_i, \Delta t_{VS}^{i,k}\}$ to Veh_i. When Veh_i receives the reply message from RSU_i, it calculates $SVSK_i = ASVSK_i \oplus va_i \cdot RV_i$. $SVSK_i$ is the short-term secret key of Veh_i with short-term pseudonym $SPID_{i,k}$. Then, the short-term private key of Veh_i is set as $vpr_i = \{va_i, SVSK_i\}$. Veh_i sets its corresponding public key $VPU_i = \{VA_i, RV_i\}$.

ARTICLE IN PRESS

Signing and Broadcasting BSM

Assume that, Veh_i generates a plain-text $M_i^{'}$ containing driving information such as position, speed and direction. Veh_i selects a random number $vb_i \in Z_q^*$ and calculates $VB_i = vb_i \cdot G$, $h_M^{i,1} = H_6(M_i^{'}, SPID_{i,k}, VB_i, tim_M)$, $h_M^{i,2} = H_6(M_i^{'}, SPID_{i,k}, VB_i, h_M^{i,1})$ and $\delta_M^i = (vb_i + h_M^{i,1} \cdot va_i + h_M^{i,2} \cdot SVSK_i) \bmod q$, where tim_M is the timestamp and δ_M^i is the signature. Then, Veh_i broadcasts the message $M_i : \{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1}, \Delta t_{VS}^{i,k}, \delta_M^i\}$.

Authenticating BSM

If a RSU or vehicle receives a BSM, it first uses an operation based on hash function to quickly check whether the BSM has been tampered. Next, the receiver authenticates the authenticity of the signature in the BSM to achieve message authentication and thus ensure accountability.

1) Quick check

For the BSM M_i sent by Veh_i, the receiver generates a new value of $h_M^{i,1}$ using $(M_i^{'}, SPID_{i,k}, VB_i, tim_M)$ which is included in M_i, and hash function $H_6()$. The receiver compared the calculated new $h_M^{i,1}$ value with the original $h_M^{i,1}$ value in the BSM. If the two values are equal, the receiver continues to authenticate message signature; otherwise, it discards the BSM.

2) Signature verification

The proposed SRPP scheme supports BSM authentication both individually and in batches.

a) Single mode

First, we introduce the single signature verification mode as follows. The receiver calculates $h_M^{i,2} = H_6(M_i^{'}, SPID_{i,k}, VB_i, h_M^{i,1})$, $h_{RSU}^i = H_1(RID_i, R_i, \Delta t_R^i)$ and $rh_i = H_5(SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k})$. The receiver then checks if the following equation holds, and if it does, the signature is correct, otherwise the BSM is discarded.

$$\delta_M^i \cdot G = VB_i + h_M^{i,1} \cdot VA_i + h_M^{i,2} \cdot R_i + h_M^{i,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2} \cdot rh_i \cdot RV_i \tag{4}$$

The correctness of Eq. 4 is proved as:

$$\begin{aligned} \delta_M^i \cdot G &= (vb_i + h_M^{i,1} \cdot va_i + h_M^{i,2} \cdot SVSK_i) \cdot G \\ &= (vb_i + h_M^{i,1} \cdot va_i + h_M^{i,2} \cdot RSK_i + h_M^{i,2} \cdot rh_i \cdot rv_i) \cdot G \\ &= (vb_i + h_M^{i,1} \cdot va_i + h_M^{i,2} \cdot r_i + h_M^{i,2} \cdot s_{TA} \cdot h_{RSU}^i + h_M^{i,2} \cdot rh_i \cdot rv_i) \cdot G \\ &= VB_i + h_M^{i,1} \cdot VA_i + h_M^{i,2} \cdot R_i + h_M^{i,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2} \cdot rh_i \cdot RV_i \end{aligned} \tag{5}$$

b) Batch mode

In RSU_i jurisdiction region, when receiving n message tuples $M_j : \{M_j^{'}, SPID_{j,k}, VA_j, RV_j, VB_j, tim_M, h_M^{j,1}, \Delta t_{VS}^{j,k}, \delta_M^j\}$ for $j = 1, 2, ..., n$, the receiver computes $\delta_\Sigma = \sum_{j=1}^n \delta_M^j$. The receiver then checks if the following equation holds, if it does, then the n signatures are correct, otherwise there is an incorrectly signed message. After the signature authentication is completed, other processing is performed on these messages.

$$\delta_\Sigma \cdot G = \sum_{j=1}^n (VB_j + h_M^{j,1} \cdot VA_j + h_M^{j,2} \cdot rh_j \cdot RV_j) + \sum_{j=1}^n h_M^{j,2} \cdot (R_i + h_{RSU}^i \cdot S_{TA}) \tag{6}$$

The correctness of Eq. 6 is proved as:

$$\begin{aligned} \delta_\Sigma \cdot G &= \sum_{j=1}^n \delta_M^j \cdot G \\ &= \sum_{j=1}^n (vb_j + h_M^{j,1} \cdot va_j + h_M^{j,2} \cdot SVSK_j) \cdot G \\ &= \sum_{j=1}^n (vb_j + h_M^{j,1} \cdot va_j + h_M^{j,2} \cdot RSK_i + h_M^{j,2} \cdot rh_j \cdot rv_j) \cdot G \\ &= \sum_{j=1}^n (vb_j + h_M^{j,1} \cdot va_j + h_M^{j,2} \cdot r_i + h_M^{j,2} \cdot s_{TA} \cdot h_{RSU}^i + h_M^{j,2} \cdot rh_j \cdot rv_j) \cdot G \\ &= \sum_{j=1}^n (VB_j + h_M^{j,1} \cdot VA_j + h_M^{j,2} \cdot R_i + h_M^{j,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{j,2} \cdot rh_j \cdot RV_j) \\ &= \sum_{j=1}^n (VB_j + h_M^{j,1} \cdot VA_j + h_M^{j,2} \cdot rh_j \cdot RV_j) + \sum_{j=1}^n h_M^{j,2} \cdot (R_i + h_{RSU}^i \cdot S_{TA}) \end{aligned} \tag{7}$$

Security Proof and Analysis

In this section, the security of the proposed SRPP scheme is proved. In addition, security analysis demonstrates that the SRPP scheme can satisfy security requirements.

ARTICLE IN PRESS

Security Proof

By using two games played between an adversary and a challenger, we modeled the security of the SRPP scheme. Two types of adversaries are considered: Type I and Type II[30,32].

Type-I Attacker: This kind of attacker simulates a common entity as a malicious vehicle to replace the public key or private key of any vehicle. However, it cannot get the master secret key of TA.

Type-II Attacker: This kind of attacker simulates a curious but honest TA and can approach the TA's master secret key. However, it cannot replace the public key of a vehicle or access a vehicle's private key.

Theorem 1. In the random oracle model, the proposed SRPP scheme is existentially unforgeable against a Type-I adversary A_1, if the ECDLP assumption holds.

Proof. Suppose the adversary A_1 can forge a BSM $M_i : \{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, \Delta t_{VS}^{i,k}, \delta_M^i\}$. By utilizing A_1 as a subroutine, we construct a challenger C that can solve the ECDLP with a non-negligible probability. Given a random instance $W = s_{TA} \cdot G$ of the ECDLP to C , where $s_{TA} \in Z_q^*$ and W, G are points on E/F_q, the ultimate goal of C is to output s_{TA} by interacting with A_1.

(1) Setup: C generates $sypas : \{q, G_q, G, S_{TA}, H_1, H_2, H_3, H_4, H_5, H_6\}$ and sends $sypas$ to A_1. Note that A_1 does not know secret key s_{TA}.

(2) H_2-query: By inputting $\{V_i, \Delta t_V^i\}$, A_1 submits H_2-query to C. If the input exists in the list L_{H_2}, C returns h_{Veh}^i to A_1. Otherwise, C computes $h_{Veh}^i = H_2(V_i, \Delta t_V^i)$. C forwards h_{Veh}^i to A_1, and adds the element $\{V_i, \Delta t_V^i, h_{Veh}^i\}$ to the list L_{H_2}.

(3) RevealLongtermSecetKey: By inputting V_i, A_1 requests this query to C. When receiving the query, C checks list L_{LSK}. If V_i already exists in L_{LSK}, C returns $\{VSK_i, \Delta t_V^i\}$ to A_1. Otherwise, C chooses $VSK_i, \Delta t_V^i \in Z_q^*$ randomly. Then, C forwards $\{VSK_i, \Delta t_V^i\}$ to A_1 and adds the element $\{V_i, VSK_i, \Delta t_V^i\}$ to L_{LSK}.

(4) H_5-query: A_1 submits this query with an input $\{SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k}\}$. Then, C checks list L_{H_5}. If the input exists in L_{H_5}, C returns rh_i to A_1. Otherwise, C computes $rh_i = H_5(SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k})$. C forwards rh_i to A_1 and adds the element $\{SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k}, rh_i\}$ to L_{H_5}.

(5) RevealShorttermPartialPrivateKey: A_1 requests this query with an input $\{SPID_{i,k}, R_i\}$. When receiving the request, C checks list L_{SPPK}. If $\{SPID_{i,k}, R_i\}$ already exists in L_{SPPK}, C returns va_i to A_1. Otherwise, C selects va_i randomly and computes $VA_i = va_i \cdot G$. Finally, C adds the element $\{SPID_{i,k}, R_i, va_i, VA_i\}$ to L_{SPPK} and outputs $\{va_i, VA_i\}$ to A_1.

(6) RevealShorttermPrivateKey: By inputting $\{SPID_{i,k}, R_i\}$, A_1 requests this query to C. When receiving the request, C checks list L_{SPK} as follows.

a) If $\{SPID_{i,k}, R_i\}$ exists in L_{SPK}, C returns short-term private key $vpr_i = \{va_i, SVSK_i\}$ to A_1.

b) If $\{SPID_{i,k}, R_i\}$ does not exist in L_{SPK}, C submits a query on RevealShorttermPartialPrivateKey$(SPID_{i,k}, R_i)$ to produce $\{SPID_{i,k}, R_i, va_i, VA_i\}$, and adds returned values to L_{SPPK}. Next, C randomly selects rv_i, RSK_i, $\Delta t_{VS}^{i,k} \in Z_q^*$, calculates $RV_i = rv_i \cdot G$, $rh_i = H_5(SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k})$ and $SVSK_i = (RSK_i + rh_i \cdot rv_i) \bmod q$. Finally, C adds the element $\{SPID_{i,k}, R_i, va_i, VA_i, rv_i, RV_i, \Delta t_{VS}^{i,k}, rh_i, SVSK_i\}$ to L_{SPK} and outputs the private key $vpr_i = \{va_i, SVSK_i\}$ to A_1.

(7) RevealPublicKey: By inputting $\{SPID_{i,k}, R_i\}$, A_1 submits this query to C. When receiving the request, C checks L_{SPK} as follows.

a) If the input value exists in L_{SPK}, C outputs $\{VA_i, RV_i\}$ to A_1.

b) If there is no $\{SPID_{i,k}, R_i\}$ in L_{SPK}, C submits a query on RevealShorttermPrivateKey$(SPID_{i,k}, R_i)$ to generate $\{SPID_{i,k}, R_i, va_i, VA_i, rv_i, RV_i, \Delta t_{VS}^{i,k}, rh_i, SVSK_i\}$ and adds the element to L_{SPK}. Finally, C outputs $\{VA_i, RV_i\}$ to A_1.

(8) RepalcePublicKey: A_1 submits $\{SPID_{i,k}, R_i, \Delta t_{VS}^{i,k}, VA_i^{'}, RV_i^{'}\}$, where $VA_i^{'} = va_i^{'} \cdot G$, $RV_i^{'} = rv_i^{'} \cdot G$. Then, C uses $\{SPID_{i,k}, R_i, \Delta t_{VS}^{i,k}, VA_i^{'}, RV_i^{'}\}$ to replace the corresponding old tuple in L_{SPK}. In addition, A_1 keeps $\{va_i^{'}, rv_i^{'}\}$.

(9) H_6-query: By inputting $\{M_i^{'}, SPID_{i,k}, VB_i, tim_M\}$, A_1 submits this query to C. Then, C checks list L_{H_6}. If there is a record in L_{H_6} that matches the input value. C outputs $(h_M^{i,1}, h_M^{i,2})$ to A_1. Otherwise, C computes $h_M^{i,1} = H_6(M_i^{'}, SPID_{i,k}, VB_i, tim_M)$ and $h_M^{i,2} = H_6(M_i^{'}, SPID_{i,k}, VB_i, h_M^{i,1})$. C forwards $(h_M^{i,1}, h_M^{i,2})$ to A_1 and adds the element $\{M_i^{'}, SPID_{i,k}, VB_i, tim_M, h_M^{i,1}, h_M^{i,2}\}$ to L_{H_6}.

(10) Signing-query: A_1 submits this query with $\{SPID_{i,k}, R_i, M_i^{'}\}$. C selects $\{va_i, SVSK_i\}$ from the list L_{SPK}. Next, it chooses $vb_i, tim_M \in Z_q^*$ randomly and calculates $VB_i = vb_i \cdot G$. C selects $h_M^{i,1}$ and $h_M^{i,2}$ from the list L_{H_6} by inputting $\{M_i^{'}, SPID_{i,k}, VB_i, tim_M\}$. C calculates $\delta_M^i = (vb_i + h_M^{i,1} \cdot va_i + h_M^{i,2} \cdot SVSK_i) \bmod q$, and returns δ_M^i to A_1. The response to the signing query satisfies Eq. 8

$$\delta_M^i \cdot G = VB_i + h_M^{i,1} \cdot VA_i + h_M^{i,2} \cdot R_i + h_M^{i,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2} \cdot rh_i \cdot RV_i \tag{8}$$

ARTICLE IN PRESS

Finally, A_1 outputs the tuple $\{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1}, \Delta t_{VS}^{i,k}, \delta_M^i\}$. According to Forgery Lemma[37]. A_1 can forge another valid tuple $\{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1}, \Delta t_{VS}^{i,k}, \delta_{M*}^i\}$ that satisfies the following equation:

$$\delta_M^{i*} \cdot G = VB_i + h_M^{i,1} \cdot VA_i + h_M^{i,2*} \cdot R_i + h_M^{i,2*} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2*} \cdot rh_i \cdot RV_i \tag{9}$$

Based on Eq. 8 and Eq. 9, we have

$$\begin{aligned}(\delta_M^i - \delta_M^{i*}) \cdot G &= (h_M^{i,2} - h_M^{i,2*}) \cdot R_i + (h_M^{i,2} - h_M^{i,2*}) \cdot h_{RSU}^i \cdot S_{TA} + (h_M^{i,2} - h_M^{i,2*}) \cdot rh_i \cdot RV_i \\ &= (h_M^{i,2} - h_M^{i,2*})(R_i + h_{RSU}^i \cdot S_{TA} + rh_i \cdot RV_i) \\ &= (h_M^{i,2} - h_M^{i,2*})(r_i^{'} \cdot G + h_{RSU}^i \cdot s_{TA} \cdot G + rh_i \cdot rv_i^{'} \cdot G)\end{aligned} \tag{10}$$

Hence, C returns $[(\delta_M^i - \delta_M^{i*})(h_M^{i,2} - h_M^{i,2*})^{-1} - r_i^{'} - rh_i \cdot rv_i^{'}](h_{RSU}^i)^{-1}$ as the solution to the given instance of ECDLP; otherwise, the game is terminated. However, this contradicts with the difficult assumption of ECDLP. Therefore, in the security model, the proposed SRPP scheme is existentially unforgeable against a Type I adversary A_1.

Theorem 2. In the random oracle model, the proposed SRPP scheme is existentially unforgeable against an Type-II adversary A_2 based on the ECDLP assumption holds.

Proof. Suppose the adversary A_2 can forge a BSM $M_i : \{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, \Delta t_{VS}^{i,k}, \delta_M^i\}$. By utilizing A_2 as a subroutine, we construct a challenger C that can solve the ECDLP with a non-negligible probability. Given a random instance $W = va_i \cdot G$ of the ECDLP to C , where $va_i \in Z_q^*$ and W, G are points on E/F_q, the ultimate goal of C is to output va_i by interacting with A_2.

(1) Setup: C generates $sypas = \{q, G_q, G, S_{TA}, H_1, H_2, H_3, H_4, H_5, H_6\}$, and sends $sypas$ and master secret key s_{TA} to A_2. C randomly selects $\{V^*, SPID^*\}$ as challenge long-term public key and short-term pseudonym for A_2.

(2) H_2-query: By inputting $\{V_i, \Delta t_V^i\}$, A_2 submits H_2-query to C. If there is a record in list L_{H_2} that matches $\{V_i, \Delta t_V^i\}$. C outputs h_{Veh}^i to A_2. Otherwise, C computes $h_{Veh}^i = H_2(V_i, \Delta t_V^i)$. C forwards h_{Veh}^i to A_2, and adds the element $\{V_i, \Delta t_V^i, h_{Veh}^i\}$ to the list L_{H_2}.

(3) RevealLongtermSecetKey: By inputting V_i, A_2 requests this query to C. When receiving the request, C checks L_{LSK} as follows.

a) If there is a tuple that matches V_i in L_{LSK}, C outputs the corresponding $\{VSK_i, \Delta t_V^i\}$ to A_2.

b) If $V_i \neq V^*$ and there is no tuple matching V_i in L_{LSK}, C chooses $VSK_i, \Delta t_V^i \in Z_q^*$ randomly. Then, C forwards $\{VSK_i, \Delta t_V^i\}$ to A_2 and adds the element $\{V_i, VSK_i, \Delta t_V^i\}$ to L_{LSK}.

c) If $V_i = V^*$, C aborts the game process.

(4) H_5-query: A_2 submits this query with an input $\{SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k}\}$. Then, C checks list L_{H_5}. If there is a tuple matching the input value in L_{H_5}, C returns rh_i to A_2. Otherwise, C computes $rh_i = H_5(SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k})$. C forwards rh_i to A_2 and adds the element $\{SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k}, rh_i\}$ to L_{H_5}.

(5) RevealShorttermPartialPrivateKey: By inputting $\{SPID_{i,k}, R_i\}$, A_2 requests this query to C. When receiving the request, C checks list L_{SPPK} as follows.

a) If there is a tuple matching $\{SPID_{i,k}, R_i\}$ in L_{SPPK}, C outputs va_i to A_2.

b) If $SPID_{i,k} \neq SPID^*$ and $SPID_{i,k}$ does not exist in the L_{SPPK}, C selects va_i randomly and computes $VA_i = va_i \cdot G$. Finally, C adds the element $\{SPID_{i,k}, R_i, va_i, VA_i\}$ to L_{SPPK} and outputs $\{va_i, VA_i\}$ to A_2.

c) If $SPID_{i,k} = SPID^*$, C aborts the game process.

(6) RevealShorttermPrivateKey: A_2 requests this query with an input $\{SPID_{i,k}, R_i\}$. Then, C checks list L_{SPK} as follows.

a) If $\{SPID_{i,k}, R_i\}$ already exists in the L_{SPK}, C returns short-term private key $vpr_i = \{va_i, SVSK_i\}$ to A_2.

b) If there is no tuple matching $\{SPID_{i,k}, R_i\}$ in L_{SPK}, C makes a query on RevealShorttermPartialPrivateKey $(SPID_{i,k}, R_i)$ to produce $\{SPID_{i,k}, R_i, va_i, VA_i\}$, and adds the element to L_{SPPK}. Next, C selects $rv_i, RSK_i, \Delta t_{VS}^{i,k} \in Z_q^*$ randomly, calculates $RV_i = rv_i \cdot G$, $rh_i = H_5(SPID_{i,k}, VA_i, RV_i, \Delta t_{VS}^{i,k})$ and $SVSK_i = (RSK_i + rh_i \cdot rv_i) \bmod q$. Finally, C adds the element $\{SPID_{i,k}, R_i, va_i, VA_i, rv_i, RV_i, \Delta t_{VS}^{i,k}, rh_i, SVSK_i\}$ to L_{SPK} and outputs the private key $vpr_i = \{va_i, SVSK_i\}$ to A_2.

(7) RevealPublicKey: By inputting $\{SPID_{i,k}, R_i\}$, A_2 requests this query to C. When receiving the request, C checks list L_{SPK} as follows.

a) If there is a tuple matching the input value in L_{SPK}. C outputs $\{VA_i, RV_i\}$ to A_2.

b) If there is no tuple matching the input value in L_{SPK}, C makes a query on RevealShorttermPrivateKey $(SPID_{i,k}, R_i)$ to generate $\{SPID_{i,k}, R_i, va_i, VA_i, rv_i, RV_i, \Delta t_{VS}^{i,k}, rh_i, SVSK_i\}$ and adds the element to L_{SPK}. Finally, C outputs the $\{VA_i, RV_i\}$ to A_2.

ARTICLE IN PRESS

(8) H_6-query: By inputting $\{M_i^{'}, SPID_{i,k}, VB_i, tim_M\}$, A_2 submits H_6-query to C. Then, C checks list L_{H_6}. If there is a tuple matching the input value in L_{H_6}. C returns $(h_M^{i,1}, h_M^{i,2})$ to A_2. Otherwise, C computes $h_M^{i,1} = H_6(M_i^{'}, SPID_{i,k}, VB_i, tim_M)$ and $h_M^{i,2} = H_6(M_i^{'}, SPID_{i,k}, VB_i, h_M^{i,1})$. C forwards $(h_M^{i,1}, h_M^{i,2})$ to A_2 and adds the element $\{M_i^{'}, SPID_{i,k}, VB_i, tim_M, h_M^{i,1}, h_M^{i,2}\}$ to L_{H_6}.

(9) Signing-query: A_2 submits this query with $\{SPID_{i,k}, R_i, M_i^{'}\}$. C selects $\{va_i, SVSK_i\}$ from the list L_{SPK}. Next, it chooses $vb_i, tim_M \in Z_q^*$ randomly and calculates $VB_i = vb_i \cdot G$. C selects $h_M^{i,1}$ and $h_M^{i,2}$ from the list L_{H_6} by inputting $\{M_i^{'}, SPID_{i,k}, VB_i, tim_M\}$. C calculates $\delta_M^i = (vb_i + h_M^{i,1} \cdot va_i + h_M^{i,2} \cdot SVSK_i) \bmod q$, and returns δ_M^i to A_2 as a valid signature. The response to the signing query satisfies Eq. 11.

$$\delta_M^i \cdot G = VB_i + h_M^{i,1} \cdot VA_i + h_M^{i,2} \cdot R_i + h_M^{i,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2} \cdot rh_i \cdot RV_i \tag{11}$$

Finally, A_2 outputs the tuple $\{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1}, \Delta t_{VS}^{i,k}, \delta_M^i\}$. According to Forgery Lemma[37]. A_2 can forge another valid tuple $\{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1*}, \Delta t_{VS}^{i,k}, \delta_{M*}^i\}$ that satisfies the following equation:

$$\delta_M^{i*} \cdot G = VB_i + h_M^{i,1*} \cdot VA_i + h_M^{i,2} \cdot R_i + h_M^{i,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2} \cdot rh_i \cdot RV_i \tag{12}$$

Based on Eq. 11 and Eq. 12, we have

$$(\delta_M^i - \delta_M^{i*}) \cdot G = (h_M^{i,1} - h_M^{i,1*}) \cdot VA_i = (h_M^{i,1} - h_M^{i,1*}) \cdot va_i \cdot G \tag{13}$$

Hence, C returns $(\delta_M^i - \delta_M^{i*})(h_M^{i,1} - h_M^{i,1*})^{-1}$ as the solution of the given instance of ECDLP; otherwise, the game is terminated. However, this contradicts with the difficult assumption of the ECDLP. Therefore, in the security model, the proposed SRPP scheme is existentially unforgeable against a Type II adversary A_2.

Security Analysis

We analyze the security of the proposed SRPP scheme in this part.

1) Message Authentication: In the SRPP scheme, to generate a BSM signature δ_M^i, $(SVSK_i, h_M^{i,1}, h_M^{i,2}, va_i, vb_i)$ are required, among which $SVSK_i$ is generated by RSU and is securely transmitted to Veh_i based on XOR operation and CDHP. $h_M^{i,1}$ and $h_M^{i,2}$ are hash values. va_i and vb_i are random numbers generated by Veh_i secretly. Also, the SRPP scheme can verify the integrity and validity of the BSM $M_i : \{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1}, \Delta t_{VS}^{i,k}, \delta_M^i\}$ by checking whether the condition $\delta_M^i \cdot G = VB_i + h_M^{i,1} \cdot VA_i + h_M^{i,2} \cdot R_i + h_M^{i,2} \cdot h_{RSU}^i \cdot S_{TA} + h_M^{i,2} \cdot rh_i \cdot RV_i$ is true. In this way, only vehicle Veh_i can generate correct message signature, and BSM signature is unforgeable. Therefore, the proposed SRPP scheme satisfies message authentication.

2) Dynamic Pseudonym: In our scheme, based on two hash chains, vehicles use various short-term pseudonyms in different RSU jurisdictions to send BSMs.

3) Un-Linkability: In the SRPP scheme, attackers cannot link two or more short-term pseudonyms to the same vehicle. Suppose $SPID_{i,k} = H_4(H_4^k(Seed_i^1) \oplus H_4^k(Seed_i^2))$ and $SPID_{i,k+1} = H_4(H_4^{k+1}(Seed_i^1) \oplus H_4^{k+1}(Seed_i^2))$ belong to the same vehicle, to validate their relationship, the attacker should calculate $x_1 = H_4^k(Seed_i^1)$, $x_2 = H_4^k(Seed_i^2)$, $y_1 = H_4(H_4(x_1) \oplus H_4(x_2))$ for each possible values of $Seed_i^1$ and $Seed_i^2$ until the condition $SPID_{i,k+1} = y_1$ is established. Because, hash function is unidirectional, it is computational hardness to link the relationship between two short-term pseudonyms.

4) Conditional Traceability: In the SRPP scheme, Veh_i uses long-term public key to execute mutual authentication with RSU, and uses short-term pseudonyms to broadcast BSMs. The real identity VID_i and long-term public key V_i are stored in TA database. Long-term public key and short-term pseudonym are stored in RSU's database. In special cases, RSU can trace long-term public key V_i based on short-term pseudonym in BSM, and then TA reveals the real identity VID_i based on V_i. In summary, our scheme can provide conditional traceability.

5) Resistance Against Attacks: As in the previous analysis, short-term pseudonyms for vehicles are not linkable in our scheme. Therefore, the SRPP scheme has the ability to resist tracking attacks. When a BSM is signed, the SRPP scheme uses a timestamp tim_M. Upon receiving a BSM, the receiver first check its freshness. The receiver will only proceed to the next step on the BSM whose timestamp has not expired. Thus, our scheme can effectively resist replay attacks. Short-term pseudonyms used by vehicles require authorization from RSUs first. Vehicles cannot use short-term pseudonyms arbitrarily. So, our SRPP scheme can resist Sybil attacks. Our proposed scheme provides batch mode to authenticate message signatures. A vehicle or RSU can verify the correctness of the signatures of a large number of BSMs in a short period of time. Therefore, the SRPP scheme can effectively resist DoS attacks.

ARTICLE IN PRESS

ARTICLE IN PRESS

Table 2. Comparison of Security Features of Related Schemes

Schemes	RS	RTT	TA	RSU
Zhu et al.[1]	✓	×	×	×
Wu et al.[7]	×	✓	×	×
Zhu et al.[8]	×	✓	✓	✓
Zhou et al.[18]	✓	×	×	×
Yang et al.[29]	✓	×	×	×
Our scheme	✓	✓	×	✓

Table 3. Execution Time of Cryptographic Operations

Notations	Definitions	Execution time (ms)
T_{bp}	Bilinear pairing operation	4.2110
T_{sm-bp}	Scale multiplication operation based on bilinear pairing	1.7090
T_{mtp}	MapToPoint hash operation related to the bilinear pairing	4.4060
T_{sm-ecc}	Scale multiplication operation based on elliptic curve	0.4420
$T_{sm-ecc-s}$	Small scale multiplication operation related to elliptic curve	0.0138
T_{pa-ecc}	Point addition operation related to elliptic curve	0.0018
T_h	General hash function operation	0.0001

Performance Evaluation

This section evaluates the performance of the proposed SRPP scheme from the aspects of security feature, computation time cost and communication overhead.

Security Feature

In this section, we evaluate and compare security features including resistance to Sybil attacks and resistance to trajectory tracking attacks. If a vehicle uses an anonymous identity for a long time, it is vulnerable to trajectory tracking attacks. If a vehicle can use multiple pseudonyms at the same time, a malicious user may launch Sybil attacks. According to the security analysis in the previous section and the description of related works, Table 2 summarizes the security features of the proposed SRPP scheme and other relevant schemes in terms of resistance against Sybil (RS) and trajectory tracking attacks (RTT), whether online TA or RSU is needed. From table 2, we can see that our scheme provides better Sybil attack defense and trajectory tracking defense capabilities.

Computation Time Cost

We compare the proposed SRPP scheme with the recent related schemes[1,7,8,18,29] in terms of computation time cost in vehicular communications. We set security level to 80 bits. For ECC-based schemes, the ECC is instance as an additive group G with prime order q on the $E: y^2 = x^3 + ax + b \pmod p$, where p, q are both 160 bits primes and $a, b \in Z_p^*$. For bilinear pairing-based schemes, the bilinear pairing is instanced as $e: G_1 \times G_1 \rightarrow G_T$, where G_1 is an additive group of prime order q on $E: y^2 = x^3 + x \pmod q$, q is a 160 bits Solinas prime number and p is a 512 bits prime number. The execution time of cryptographic operations have been obtained using the MIRACL library[39] on the hardware platform which contains 4-GB memory, an Intel I7-4770 processor, and runs on Windows 7 operating system. Table 3 lists out the time of executing following cryptographic operations[32,40].

In our comparison, we mainly consider the computation time cost for generating one signature and verifying $n(n > 1)$ signatures. In our SRPP scheme, to generate one signature for BSM, one scale multiplication operation based on elliptic curve and 2 general hash function operations are involved, and the computation time cost is $1T_{sm-ecc} + 2T_h = 0.4422\ ms$. In terms of signature verification, our scheme involves $(2n+3)$ scalar multiplication operations, $(3n+1)$ point addition operations related to elliptic curve and $(3n+1)$general hash function operations. Therefore, the total cost of verifying n signatures is $(2n+3)T_{sm-ecc} + (3n+1)T_{pa-ecc} + (3n+1)T_h = (0.8897n + 1.3279)\ ms$. So the total cost of execution is $(0.8897n + 1.7701)ms$. In the similar way, we analyze the computation time cost of other schemes and summarize the results in Table 4 and Fig. 3. Because the scheme[7] proposed by Zhu et al. is based on pairing operations, the computation time cost is significantly higher than that of other schemes. To make the comparison of computation time costs among other schemes clearer, we do not present the computation time cost of this scheme in Fig. 3.

As shown in Table 4 and Fig. 3, in terms of signing one message, the performance of our scheme is similar to other four

Table 4. Comparison on Computation Cost of Related Schemes

Schemes	Sign one message (ms)	Authentication $n(n>1)$ messages (ms)	Total cost (ms)
Zhu et al.[1]	$1T_{sm-ecc}+2T_h=0.4422$	$(2n+3)T_{sm-ecc}+(3n-3)T_{sm-ecc-s}+(3n+1)T_{pa-ecc}+3nT_h=0.9311n+1.2864$	$0.9311n+1.7286$
Wu et al.[7]	$3T_{sm-ecc}+1T_{mtp}+1T_h=9.5331$	$3T_{bp}+3nT_{sm-bp}+2nT_{mtp}+nT_h=13.9391n+12.633$	$13.9391n+22.1661$
Zhu et al.[8]	$1T_{sm-ecc}+1T_h=0.4421$	$(n+2)T_{sm-ecc}+3nT_{pa-ecc}+2nT_h=0.4476n+0.884$	$0.4476n+1.3261$
Zhou et al.[18]	$1T_{sm-ecc}+2T_h=0.4422$	$(2n+2)T_{sm-ecc}+3nT_{pa-ecc}+3nT_h=0.8897n+0.884$	$0.8897n+1.3262$
Yang et al.[29]	$1T_{sm-ecc}+2T_h=0.4422$	$s(4n+1)T_{sm-ecc}+3nT_{pa-ecc}+4nT_h=1.7738n+0.442$	$1.7738n+0.8842$
Our scheme	$1T_{sm-ecc}+2T_h=0.4422$	$(2n+3)T_{sm-ecc}+(3n+1)T_{pa-ecc}+(3n+1)T_h=0.8897n+1.3279$	$0.8897n+1.7701$

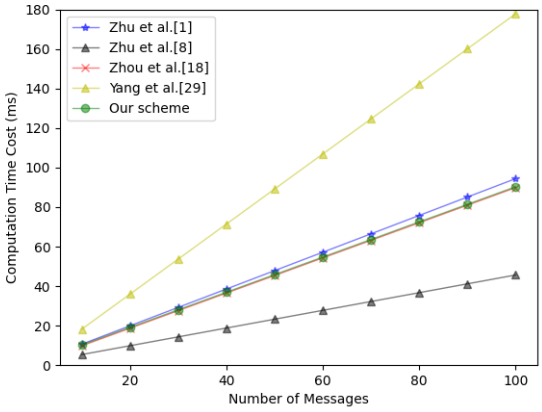


Figure 3. Computation time cost comparison of schemes.

schemes[1,8,18,29]. In terms of verifying messages, the computation time cost of our scheme is lower than Yang's scheme[29], similar to other two schemes[1,18], and slightly higher than Zhu's scheme[8]. In Zhu's scheme[1], a vehicle has only one pseudonym, which makes it vulnerable to trajectory tracking attacks, and this scheme only supports V2I communication. Although Zhu's scheme[8] has a lower computation time cost than ours, their dynamic pseudonym scheme relies on online TA, which can become a system bottleneck when there is a high traffic follow.

Communication Overhead

We analyze and compare our SRPP scheme with other recent related schemes in terms of communication overhead when vehicles send BSMs. To analyze bilinear pairing-based scheme, the size of $|G_1|$ is set to $64\times 2=128$ bytes. For ECC-based schemes, the size of $|G|$ is set to $20\times 2=40$ bytes. In both types of schemes, the elements in Z_q^*, the size of hashing value and the field related to time are 20 bytes, 20 bytes and 4 bytes, respectively. Because all schemes have the same plain-text driving data, we exclude this overhead when comparing performance.

In our scheme, the transmitted message is $M_i:\{M_i^{'}, SPID_{i,k}, VA_i, RV_i, VB_i, tim_M, h_M^{i,1}, \Delta t_{VS}^{i,k}, \delta_M^i\}$, where $M_i^{'}$ is plain-text; $SPID_{i,k}, h_M^{i,1}, \delta_M^i \in Z_q^*$; $VA_i, RV_i, VB_i \in G$; $tim_M, \Delta t_{VS}^{i,k}$ are fields related to time. Therefore, the communication overhead for sending a BSM is $20\times 3+40\times 3+4\times 2=188$ bytes. The communication overhead of other schemes[1,7,8,18,29] to send one message is 208 bytes, 116 bytes, 164 bytes, 228 bytes and 148 bytes respectively. We summarize the results in Fig. 4. As can be seen from Fig. 4, the communication overhead of our proposed scheme is lower than the schemes in studies[1,18] and is higher than other three schemes[7,8,29]. Although the communication overhead of our scheme is not the lowest, based on the previous analysis, Wu's scheme[7] has high computational cost and Zhu's scheme[8] exists the risk of system bottleneck. In

ARTICLE IN PRESS

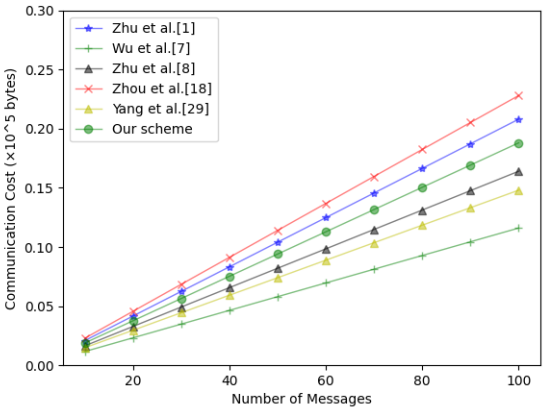


Figure 4. Comparison of communication cost in sending messages.

addition, compared with Yang's scheme[29], our scheme has the advantage of resisting trajectory tracking.

In general, our proposed scheme, assisted by RSU, provides security capabilities against Sybil attacks and trajectory tracking attacks, which are not fully considered in other schemes. Although our scheme is slightly higher than Zhu's scheme[8] in terms of computation time, their scheme relies on an online TA, which can be a system performance bottleneck. On the other hand, although Wu's scheme[7] has lower communication cost than our proposed scheme, their operations of signing and authenticating messages are time-consuming.

Conclusion

IoVs is a complex system due to features such as wireless communication channels and high-speed movements of vehicles. The research of message authentication scheme for IoVs involves many problems such as computing efficiency, communication cost, identity authentication and privacy protection, trajectory privacy protection, anti-Sybil attacks and so on. Based on ECDLP, CDHP, hash operations and XOR operations, we proposed an efficient anonymous BSM authentication scheme with batch verification for vehicular communications. In addition, a dynamic pseudonym method is used to resist Sybil attacks and trajectory tracking attacks. Security proof and analysis show that the proposed SRPP scheme can meet the security requirements of message authentication, dynamic pseudonym, un-linkability and conditional traceability, and resist multiple attacks such as replay and DoS. Moreover, compared with other related work, the proposed SRPP scheme is practical in terms of computation time cost in generating signature and verifying BSMs and communication overhead in sending BSMs.

Data availability

The authors confirm that all data supporting the findings and research data of this study are available in the article.

References

1. Zhu, F. *et al.* A security-enhanced certificateless conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Transactions on Intell. Transp. Syst.* **24**, 10456–10466, DOI: 10.1109/TITS.2023.3275077 (2023).

2. Chen, S., Liu, Y., Ning, J. & Zhu, X. Basrac: An efficient batch authentication scheme with rule-based access control for vanets. *Veh. Commun.* **40**, 100575, DOI: https://doi.org/10.1016/j.vehcom.2023.100575 (2023).

3. Dong, S. *et al.* A comprehensive survey on authentication and attack detection schemes that threaten it in vehicular ad-hoc networks. *IEEE Transactions on Intell. Transp. Syst.* **24**, 13573–13602, DOI: 10.1109/TITS.2023.3297527 (2023).

4. Mdee, A. P., Khan, M. T. R., Seo, J. & Kim, D. Security compliant and cooperative pseudonyms swapping for location privacy preservation in vanets. *IEEE Transactions on Veh. Technol.* **72**, 10710–10723, DOI: 10.1109/TVT.2023.3254660 (2023).

5. Tahir, H. *et al.* Lightweight and secure multi-factor authentication scheme in vanets. *IEEE Transactions on Veh. Technol.* **72**, 14978–14986, DOI: 10.1109/TVT.2023.3286187 (2023).

ARTICLE IN PRESS

6. Hammi, B., Monteuuis, J.-P. & Petit, J. Pkis in c-its: Security functions, architectures and projects: A survey. *Veh. Commun.* **38**, 100531, DOI: https://doi.org/10.1016/j.vehcom.2022.100531 (2022).

7. Wu, Q., Zhang, L., Yang, Y. & Choo, K.-K. R. Certificateless signature scheme with batch verification for secure and privacy-preserving v2v communications in vanets. *IEEE Transactions on Dependable Secur. Comput.* 1–12 (2024).

8. Zhu, D. & Guan, Y. Secure and lightweight conditional privacy-preserving identity authentication scheme for vanet. *IEEE Sensors J.* **24**, 35743–35756, DOI: 10.1109/JSEN.2024.3431557 (2024).

9. Sharma, S. & Kaushik, B. A survey on internet of vehicles: Applications, security issues and solutions. *Veh. Commun.* **20**, 100182, DOI: https://doi.org/10.1016/j.vehcom.2019.100182 (2019).

10. Qi, J., Gao, T., Deng, X. & Zhao, C. A pseudonym-based certificateless privacy-preserving authentication scheme for vanets. *Veh. Commun.* **38**, 100535, DOI: https://doi.org/10.1016/j.vehcom.2022.100535 (2022).

11. Du, H., Wen, Q., Zhang, S. & Gao, M. An improved conditional privacy protection scheme based on ring signcryption for vanets. *IEEE Internet Things J.* **10**, 17881–17892, DOI: 10.1109/JIOT.2023.3279896 (2023).

12. Raya, M. & Hubaux, J.-P. Securing vehicular ad hoc networks. *J. Comput. Secur.* **15**, 39–68 (2007).

13. Zhang, L., Wu, Q., Solanas, A. & Domingo-Ferrer, J. A scalable robust authentication protocol for secure vehicular communications. *IEEE Transactions on Veh. Technol.* **59**, 1606–1617, DOI: 10.1109/TVT.2009.2038222 (2010).

14. Shamir, A. Identity-based cryptosystems and signature schemes. In Blakley, G. R. & Chaum, D. (eds.) *Advances in Cryptology*, 47–53 (Springer Berlin Heidelberg, Berlin, Heidelberg, 1985).

15. Xiong, W. *et al.* Improved certificateless aggregate signature scheme against collusion attacks for vanets. *IEEE Syst. J.* **17**, 1098–1109 (2023).

16. Zhong, H., Han, S., Cui, J., Zhang, J. & Xu, Y. Privacy-preserving authentication scheme with full aggregation in vanet. *Inf. Sci.* **476**, 211–221, DOI: https://doi.org/10.1016/j.ins.2018.10.021 (2019).

17. Zhang, J., Zhong, H., Cui, J., Xu, Y. & Liu, L. An extensible and effective anonymous batch authentication scheme for smart vehicular networks. *IEEE Internet Things J.* **7**, 3462–3473 (2020).

18. Zhou, X., Luo, M., Vijayakumar, P., Peng, C. & He, D. Efficient certificateless conditional privacy-preserving authentication for vanets. *IEEE Transactions on Veh. Technol.* **71**, 7863–7875, DOI: 10.1109/TVT.2022.3169948 (2022).

19. Lu, R., Lin, X., Zhu, H., Ho, P.-H. & Shen, X. Ecpp: Efficient conditional privacy preservation protocol for secure vehicular communications. In *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, 1229–1237, DOI: 10.1109/INFOCOM.2008.179 (2008).

20. Kang, J., Yu, R., Huang, X. & Zhang, Y. Privacy-preserved pseudonym scheme for fog computing supported internet of vehicles. *IEEE Transactions on Intell. Transp. Syst.* **19**, 2627–2637, DOI: 10.1109/TITS.2017.2764095 (2018).

21. Wasef, A. & Shen, X. S. Emap: Expedite message authentication protocol for vehicular ad hoc networks. *IEEE Transactions on Mob. Comput.* **12**, 78–89, DOI: 10.1109/TMC.2011.246 (2013).

22. He, D., Zeadally, S., Xu, B. & Huang, X. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Transactions on Inf. Forensics Secur.* **10**, 2681–2691, DOI: 10.1109/TIFS.2015.2473820 (2015).

23. Ali, I. & Li, F. An efficient conditional privacy-preserving authentication scheme for vehicle-to-infrastructure communication in vanets. *Veh. Commun.* **22**, 100228, DOI: https://doi.org/10.1016/j.vehcom.2019.100228 (2020).

24. Ali, I., Lawrence, T. & Li, F. An efficient identity-based signature scheme without bilinear pairing for vehicle-to-vehicle communication in vanets. *J. Syst. Archit.* **103**, 101692, DOI: https://doi.org/10.1016/j.sysarc.2019.101692 (2020).

25. Song, Y., Xu, C., Zhang, Y. & Li, F. Comments on an identity-based signature scheme for vanets. *J. Syst. Archit.* **112**, 101851, DOI: https://doi.org/10.1016/j.sysarc.2020.101851 (2021).

26. Horng, S.-J. *et al.* An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks. *Inf. Sci.* **317**, 48–66, DOI: https://doi.org/10.1016/j.ins.2015.04.033 (2015).

27. Han, Y., Song, W., Zhou, Z., Wang, H. & Yuan, B. eclas: An efficient pairing-free certificateless aggregate signature for secure vanet communication. *IEEE Syst. J.* **16**, 1637–1648, DOI: 10.1109/JSYST.2021.3116029 (2022).

28. Yang, X., Li, S., Yang, L., Du, X. & Wang, C. Efficient and security-enhanced certificateless aggregate signature-based authentication scheme with conditional privacy preservation for vanets. *IEEE Transactions on Intell. Transp. Syst.* **25**, 12256–12268, DOI: 10.1109/TITS.2024.3367925 (2024).

ARTICLE IN PRESS

ARTICLE IN PRESS

29. Yang, W., Fan, J., Song, K., Zheng, Y. & Zhang, F. An efficient and practical conditional privacy-preserving aggregate authentication for vehicular ad-hoc networks. *IEEE Transactions on Intell. Transp. Syst.* **25**, 20256–20267, DOI: 10.1109/TITS.2024.3474210 (2024).

30. Kamil, I. A. & Ogundoyin, S. O. An improved certificateless aggregate signature scheme without bilinear pairings for vehicular ad hoc networks. *J. Inf. Secur. Appl.* **44**, 184–200, DOI: https://doi.org/10.1016/j.jisa.2018.12.004 (2019).

31. Baza, M. *et al.* Detecting sybil attacks using proofs of work and location in vanets. *IEEE Transactions on Dependable Secur. Comput.* **19**, 39–53, DOI: 10.1109/TDSC.2020.2993769 (2022).

32. Ali, I., Chen, Y., Ullah, N., Kumar, R. & He, W. An efficient and provably secure ecc-based conditional privacy-preserving authentication for vehicle-to-vehicle communication in vanets. *IEEE Transactions on Veh. Technol.* **70**, 1278–1291, DOI: 10.1109/TVT.2021.3050399 (2021).

33. Wu, W. & Ye, F. A secure and efficient certificateless aggregate signature authentication scheme with pseudonyms for vanets. *IEEE Internet Things J.* **12**, 124–139, DOI: 10.1109/JIOT.2024.3459033 (2025).

34. Karabulut, M. A., Shah, A. F. M. S., Ilhan, H., Pathan, A.-S. K. & Atiquzzaman, M. Inspecting vanet with various critical aspects – a systematic review. *Ad Hoc Networks* **150**, 103281, DOI: https://doi.org/10.1016/j.adhoc.2023.103281 (2023).

35. Liu, G. *et al.* Lrcpa: Lattice-based robust and conditional privacy-preserving authentication for vanets. *IEEE Transactions on Veh. Technol.* 1–15, DOI: 10.1109/TVT.2024.3485671 (2024).

36. Mukherjee, S., Gupta, D. S. & Biswas, G. P. An efficient and batch verifiable conditional privacy-preserving authentication scheme for vanets using lattice. *Computing* **101**, 1763–1788, DOI: 10.1007/s00607-018-0689-3 (2019).

37. Imghoure, A., El-Yahyaoui, A. & Omary, F. Ecdsa-based certificateless conditional privacy-preserving authentication scheme in vehicular ad hoc network. *Veh. Commun.* **37**, 100504, DOI: https://doi.org/10.1016/j.vehcom.2022.100504 (2022).

38. Xiong, W., Wang, R., Wang, Y., Zhou, F. & Luo, X. Cppa-d: Efficient conditional privacy-preserving authentication scheme with double-insurance in vanets. *IEEE Transactions on Veh. Technol.* **70**, 3456–3468, DOI: 10.1109/TVT.2021.3064337 (2021).

39. Wenger, E. & Werner, M. Evaluating 16-bit processors for elliptic curve cryptography. In Prouff, E. (ed.) *Smart Card Research and Advanced Applications*, 166–181 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011).

40. Zhang, J., Cui, J., Zhong, H., Chen, Z. & Liu, L. Pa-crt: Chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular ad-hoc networks. *IEEE Transactions on Dependable Secur. Comput.* **18**, 722–735, DOI: 10.1109/TDSC.2019.2904274 (2021).

ARTICLE IN PRESS

Acknowledgements

This work is supported by Public Welfare Research Project of Jiaxing City of Zhejiang Province of China (No. 2024AY40009)

Author contributions

ZHANG Chunhua proposed the original idea and drafting the original manuscript. CHANG Xing and JIA Xiajun analysis and edited the manuscript. ZHANG Sulan and WANG Xiaolin checked the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to C.Z.

To include, in this order: **Accession codes** (where applicable); **Competing interests** (mandatory statement).

The corresponding author is responsible for submitting a competing interests statement on behalf of all authors of the paper. This statement must be included in the submitted article file.