Physically Secure Lightweight and Privacy-Preserving Message Authentication Protocol for VANET in Smart City

Wajdy Othman, Miao Fuyou, Kaiping Xue, *Senior Member, IEEE*, and Ammar Hawbani, *Member, IEEE*

***Abstract*—Secure message transmission in vehicular communications in smart cities is still a challenging task. Most of the related work employed the Public Key Infrastructure, Certification Revocation Lists (CRLs) for ensuring security, privacy. However, these work suffered from some issues such as: 1) the time-consuming checking process, huge size of CRLs, 2) traceability attacks by linking unencrypted Basic Safety Messages (BSMs), 3) extracting secret keys from the storage of parked vehicles or road-side units (RSU) by an adversary. To address the aforementioned issues, we thus propose a physically secure privacy-preserving message authentication protocol using Physical Unclonable Function (PUF), Secret Sharing. The proposed protocol guarantees security, privacy against passive, active attacks even under memory leakage. The entities (i.e., vehicles, RSU) make use of their PUF to reconstruct a secret polynomial-share so that pairwise temporal secret keys (PTKs) can be established with other entities. Unlike existing protocols, BSMs are also encrypted in our protocol (by PTKs) to provide a higher level of security, thwart vehicles traceability attacks. To revoke a vehicle, RSU needs not broadcast CRLs. Instead, RSU distributes only a secure offset key using threshold Secret Sharing. Consequently, our revocation checking process has computation complexity $\mathcal{O}(1)$. Our protocol also eliminates the need for a third party in Vehicle-to-Vehicle communication to ensure expeditious transmission. Security analysis, performance evaluation show that our proposed protocol outperforms existing schemes in terms of security features, computation, communication cost.**

***Index Terms*—Smart city, authentication, privacy-preserving, physical unclonable functions, secret sharing, VANET.**

I. Introduction

IN RECENT years, Internet of Things (IoT) evolved so rapidly that paved the way for smart cities. A Vehicular Ad hoc Network (VANET) is a technology employed in smart cities to establish an intelligent transportation system that provides security to roads, safety to pedestrians, passengers, drivers [1]. There are two basic communication modes in VANET: 1) Vehicle-to-Vehicle (V2V) communication, 2) Vehicle-to-Infrastructure (V2I) communication. Both modes make use of the Dedicated Short Range Communication (DSRC) standard [2]. According to DSRC, each vehicle has to broadcast regular position beacon messages (known as BSMs in the US standard) every 300 ms [2], [3]. In addition, prior to accepting a received message, vehicles must first verify the message validity, to avoid communication with revoked vehicles, the message integrity as the adversary may falsify the original messages during transmission. Furthermore, the privacy of VANET users must be preserved, otherwise, an adversary most likely obtains sensitive information such as a driver's name, travel route, or license plate [4]. According to [2], the privacy of vehicles mightn't well be preserved even with frequently switching pseudonyms. This is due to the fact that the main privacy problem lies within the BSM itself as it contains the vehicle's position, speed, heading, acceleration. However, it is being broadcast up to 10 times per second in plaintext, which facilitates messages linking, traceability attacks by an adversary [2]. VANETs essentially require deploying a revocation mechanism to prevent malicious vehicles from any future communication, thus remove them from the network. A traditional method for the revocation mechanism in VANETs is to deploy *Certification Revocation Lists* (CRLs). Basically, CRLs are lists, containing all (malicious) revoked vehicles' certificates. On the one hand, CRLs are regularly distributed to enhance the overall security, safety of vehicular networks, but on the other hand, they are time-consuming in terms of the checking process, more likely to be very large in size over time [5]. As a result, they cause a heavy burden on computation, communication processes. Nevertheless, in CRLs-based authentication schemes, all Road-Side Units (RSUs), vehicles must store, regularly update CRLs, firstly check them upon receiving a message [6]. According to [7], a vehicle consumes 9 ms to check one identity in CRL, 11 ms to verify an attached signature with a received message. Suppose the number of revoked vehicles in CRL is n, then the total number of messages which can be verified in one second is $1000/(9n + 11)$ [7]. It is obvious that CRL checking alongside signature verification presents an excessive computation, communication delay, considerably degrading VANETs performance [3]. Additionally, the delivery time of BSMs is still another concern in VANET. In other words, a cooperative safety driving system cannot avoid traffic accidents if BSMs

Manuscript received January 19, 2021; revised July 14, 2021 and October 3, 2021; accepted October 14, 2021. Date of publication October 20, 2021; date of current version December 17, 2021. This work was supported in part by the National Key R&D Project of China under Grant 2018YFB2100300. The review of this article was coordinated by Prof. Heejo Lee. *(Corresponding author: Miao Fuyou.)*

Wajdy Othman, Miao Fuyou, and Ammar Hawbani are with the School of Computer Science and Technology, University of Science and Technology of China, Hefei 230000, China (e-mail: wajdy@mail.ustc.edu.cn; mfy@ustc.edu.cn; anmande@ustc.edu.cn).

Kaiping Xue is with the School of Information Science and Technology Hefei 230027, China (e-mail: kpxue@ustc.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TVT.2021.3121449, provided by the authors.

Digital Object Identifier 10.1109/TVT.2021.3121449

delivery time is larger than 0.5 ms [8]. Moreover, if an accident occurred, the accident information could not be transmitted to users who need it as fast as possible, more serious issues such as traffic congestion or more tragic traffic accidents would probably happen [9]. Consequently, it is critical for a vehicular communication system to efficiently handle the issues of processing, fast sharing of BSMs. Furthermore, DoS (de-synchronization) attacks are another possible key issue in VANET when a vehicle requests an update of its secret credentials from RSU or TA. Even though many authentication schemes have been proposed in the literature for VANET, they still suffer from most of the issues mentioned above. In addition, most schemes are designed based on an assumption that a vehicle is equipped with tamper-proof key storage/On-Board Units (OBUs). Nonetheless, an adversary can still mount side-channel attacks, i.e., a power analysis attack, on a vehicle's OBU, obtain all secret materials stored in it since the adversary may easily gain access to parked vehicles or RSUs [10]. Hence it is concluded that the aforementioned issues are critical to VANET, addressing them is of the utmost importance. However, they are not resolved well in the existing literature. To address these issues, this article contributes towards proposing a physically secure privacy-preserving message authentication protocol based on Physical Unclonable Function (PUF), (t, n)-Shamir's Secret Sharing [11]. By utilizing PUF, the authenticity of a signer, e.g., a vehicle, is always proved as an adversary cannot tamper with PUF. The major contributions of our proposed protocol are as follows.

- *PUF, Secret Sharing-based key establishment*, where the entities (vehicle, RSUs) can make use of their PUF to reconstruct a secret *polynomial-share* . Consequently, pairwise temporal secret keys PTKs can be established with other entities. Moreover, an adversary cannot impersonate a legal entity due to employing PUF.
- *Secret Sharing-based revocation mechanism*, compared with the existing protocols that use CRLs for entity revocation, our proposed (t, n)-Secret Sharing-based Session Group Key Distribution (SGKD) reduced the revocation time complexity from $\mathcal{O}(\log N_{rev})$ to $\mathcal{O}(1)$. Moreover, our revocation mechanism is more flexible as it enables temporary revocation besides the permanent one.
- *Polynomial-based broadcast encryption, expeditious message transmission*. According to [2], it is still challenging to encrypt *broadacst*, anonymously authenticate VANET messages. We addressed the aforementioned challenge by introducing two novel algorithms (namely, *Broadcast Encryption*, *Broadcast Decryption*) wherein a polynomial-based encrypted broadcast is used, which is *symmetric, lightweight*. The encrypted broadcast can be authenticated, decrypted by only legitimate vehicles. As a result, our protocol thwarts an eavesdropper from linking messages to trace a vehicle Furthermore, our protocol doesn't require a third party during V2V communication, thus authenticated message transmission is expeditious.
- *One-side secret renewal mechanism*. Unlike many existing schemes, our protocol is secure against de-synchronization attacks, owing to PTKs, secret renewal mechanism.

II. Related Work

Authors in [5] utilized the batch group signature verification wherein a bulk of messages can be authenticated in a time window. They used a keyed Hash Message Authentication Code (HMAC) function to replace CRLs. Nevertheless, a rebatch may lead to an additional verification delay if there exist a couple of invalid messages, then it brings efficiency loss to such schemes [3]. Liu *et al.* [12] presented a privacy-preserving dual authentication, key agreement scheme for different Internet of Vehicles (IoV) scenarios. However, their scheme still has some limitations such as 1) human assistance for log-in into the system is required, 2) a trusted authority involvement is a must during authentication [13], lastly 3) using bilinear pairings rendered the scheme to be computationally expensive. Dang *et al.* [14] designed an ID-based authentication, key agreement scheme to build secure communication between two vehicles. However, their scheme is susceptible to man-in-the-middle (MITM), replay attacks besides the scheme lacks the vehicle anonymity property [13]. For securing communication among vehicles, RSUs, authors in [15] proposed a new certificateless short signature scheme. Their scheme has the advantage of achieving better efficiency in V2I communication, however, the computation cost is high [16]. Wang, Yao [17] proposed a localized anonymous message authentication scheme based on ID-based signatures. Although their scheme supports batch signature verification, it suffers from the overhead of bilinear maps, the issues of managing, distributing certificates [18]. Li *et al.* [19] proposed an identity authentication scheme based on elliptic curve cryptography for unmanned aerial vehicle (UAV) networks. Their design has three stages: 1) ECC certification generation, 2) identity authentication, 3) verification of key compatibility. However, their scheme incurs much computation overhead. Jiang *et al.* [20] proposed a cloud-centric three-factor authentication, key agreement protocol (CT-3FAKA) for autonomous vehicles (AVs). In their scheme, they integrated passwords, biometrics, smart cards to guarantee secure access to both cloud, AVs, wherein two session keys are established. Furthermore, users can accomplish authentication while the privacy of their identity, biometrics are kept preserved. Only a few PUF-based authentication protocols have been proposed in the literature. However, these protocols cannot be directly applied in vehicular communication (e.g., VANETs or IoV) due to various shortcomings. Chatterjee *et al.* [21] introduced an authentication, key exchange protocol for IoT by combining identity-based encryption (IBE), PUFs, keyed hash function. Yet, authors in [22] observed that the scheme of [21] lacks a user anonymity feature, an adversary can easily track a user as well. In [23], Gope *et al.* proposed a lightweight, privacy-preserving two-factor authentication scheme for IoT devices, utilizing PUF. However, their scheme is vulnerable to de-synchronization attacks as it does not consider the loss of messages during transmission [22]. Recently, Aman *et al.* [10] proposed an efficient protocol for authentication in IoV to reduce the overhead of authentication, improve application layer packets throughput. Nonetheless, their scheme does not consider the inherent noisy output from PUFs. Therefore, their scheme is impractical as the PUF response is not uniformly distributed,

may cause immediate rejection for a legitimate entity. Gope *et al.* [24] designed an authenticated Key agreement scheme for edge-assisted Internet of drones. In their scheme, UAVs do not need to store any secret keys. According to authors, UAVs can be authenticated by third-party communication, mobile edge computing service providers without any loss of provacy. Recently, Gope *et al.* [25] proposed PUF-based anonymous authentication scheme for RFID-enabled UAV applications. Their scheme can ensure resiliency against man-in-the-middle attacks, guarantee privacy against eavesdroppers. It's apparent that all aforementioned schemes are still not well suitable for VANETs as they have their own shortcomings such as the difficulty of certifications management, huge overheads resulted from employing CRLs and/or asymmetric cryptography (i.e., bilinear pairings, etc.), the vulnerability to physical attacks as in [5], [9], [12] -[20] (even the few existing PUF-based schemes still ignore the noisy PUF response as in [10], [24]), lastly disregarding the risk of keep sending unencrypted transmitting messages (i.e., traffic BSMs) as in [12] -[25]. Therefore, the proposed protocol is designed to fulfill VANET requirements with respect to security, privacy, efficiency. Note that using PUFs to build cryptographic protocols eliminates the need for storing secret keys in Non-Volatile Memory (NVM) storage of a node, therefore such protocols can resist physical attacks. However, some kinds of PUFs are still vulnerable to Machine-Learning-based modeling-attacks (wherein an adversary can collect a large subset of PUF's possible CRPs, utilize them in building a PUF clone using machine learning algorithm [26]). Hence authors in [26] *et al.* addressed the above issue by proposing an authentication scheme for IoT using PUFs that is secure against machine learning or modeling attacks. In their scheme, they utilized the concept of one-time PUF (OPUF), wherein the behavior of the PUF changes after the execution of each session of the protocol. Consequently, the adversary will have no extra advantage in predicting either the previous or upcoming CRPs. On the other hand, authors in [27] *et al.* proposed a PUF-based protocol for IoT that can overcome the vulnerability of modeling attack in PUF-based protocols. In their scheme, the PUF challenge is split over multiple messages (nodes) to limit the adversary's ability of intercepting the whole challenge bits exchanged with IoT nodes. As a result, the adversary's ability in retrieving the challenge bits of the PUF without reliance on cryptosystems can be hindered. Note that modeling attacks on PUFs are out of the scope of this work.

III. Preliminaries

A. Physical Unclonable Functions

PUF is a physical circuit, which always produces an unpredictable response R when it is stimulated with a challenge C. The Challenge-Response Pairs *CRPs* are a number of pairs of challenges C_i, their corresponding responses R_i [28]. PUF output is noisy by nature, thus processing PUF responses by an error correction such as fuzzy extractor is essential [29].

Definition 1: We say PUF_N, which is embedded in a device N, (d, n, l, λ, ϵ)-secure PUF if for any inputs $C_1, \ldots, C_n \in \{0,1\}^k$, where k denotes a security parameter throughout the paper, the following properties hold [23]:

- $Pr[HD(PUF_{N_1}(C_1), PUF_{N_2}(C_1)) > d] \geq 1 - \epsilon$, HD denotes the hamming distance.
- $Pr[\hat{H}_\infty(PUF_N(C_i), PUF_N(C_j))_{1 \leq i,j \leq n, i \neq j} > \lambda] \geq 1 - \epsilon$. This condition refers the min-entropy of PUF_N is always larger than λ with high probability, when the intra-distance (i.e., the distance between two PUF responses from the same PUF, using the same challenge) is smaller than d, the inter-distance (i.e., the distance between two PUF responses from different PUFs using the same challenge) is greater than d.

B. Fuzzy Extractor, Helper Data

A (d, λ)-fuzzy extractor *FE* [23], [30], [31] consists of two algorithms: $FE.Gen(.)$, $FE.Rep(.)$. $FE.Gen(.)$ is a probabilistic key generation algorithm that takes a bit string R as an input, outputs a key K, helper data hd [30], i.e., $(K, hd) = FE.Gen(R)$. $FE.Rep(.)$ is a deterministic key reproduction algorithm that recovers the key K from a noisy response R', i.e., $K = FE.Rep(R', hd)$, provided that the hamming distance *HD* between R, R' is at most d. A fuzzy extractor guarantees security even with hd is being revealed [30] if the min-entropy of an input R is at least λ, K is close to uniformly random distribution in $\{0,1\}^k$.

C. Secret Sharing

In 1979, secret sharing schemes were introduced by Shamir [11], Blakley [32] separately. (t, n)-Shamir's secret sharing scheme divides a secret s into n shares such that any t or more than t shares can recover s while less that t shares cannot obtain any information about the secret s. It contains of two phases: 1) *Share Generation*. A dealer, say *D*, constructs a polynomial $P(x)$ of degree $(t-1)$ randomly: $f(x) = a_0 + \sum_{i=1}^{t-1} a_i x^i \in \mathbb{F}_q[x]$ in which the secret $s = f(0)$, all coefficients $a_0, a_1, \ldots, a_{t-1}$ are in $\mathbb{F}_q$, a finite field with q elements. After that, *D* computes all shares: $s_i = f(x_i) \pmod q$ for $i = 1, \ldots, n$, securely sends each share s_i to the shareholder with x_i as the public information. 2) *Secret Reconstruction*. This phase can reconstruct the secret s from t shares, i.e., $(s_1, s_2, \ldots, s_t)$, $s = f(0) = \sum_{i=1}^{t} s_i (\prod_{j=1, j \neq i}^{t} \frac{x_j}{x_j - x_i}) \pmod q$.

D. System Model

As shown in Fig. 1, the proposed system model of vehicles communication in smart cities consists of various components explained as follows:

- A Trusted Authority TA is responsible for enrolling all other entities (i.e., vehicles, road-side units), distributing secret keys to them in the network.
- Road-side Units (RSUs) are base stations/signal towers fixed at the roadsides, can communicate with TA securely. RSUs interact with vehicles within their range by a wireless channel, which is the DSRC.

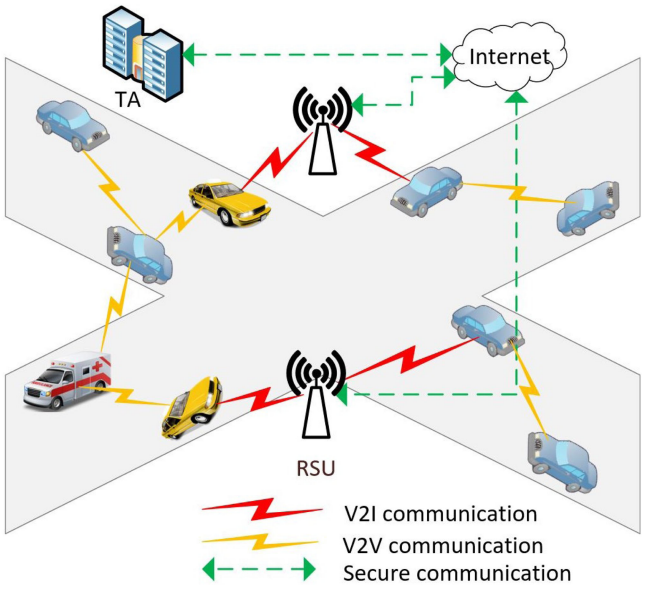

Fig. 1. Vehicles communications.

- Vehicles, which are equipped with On-Board Units *OBUs*. A vehicle utilizes its OBU to communicate with *RSUs*, other vehicles using (DSRC).

TA has high computation, storage, communication capabilities, will never be compromised. It also provides services to the vehicles, RSUs upon their requests. We assume PUF is embedded in each RSU, OBU of each vehicle. A vehicle's OBU periodically broadcasts BSMs that contain traffic-related status information such as its location, speed, direction to other vehicles so that drivers can avoid road accidents, traffic jams. RSUs are mainly fixed in road junctions with certain distances. All communications (i.e., V2V, V2I) among the aforementioned components take place through a public channel (i.e., the Internet), thus are susceptible to security attacks.

E. Design Goals

We briefly list the design goals of the proposed message authentication protocol for VANET:

1) Message authentication, integrity: Verifying the authenticity of received messages, proving they are indeed sent by authorized entities without being altered.
2) Physical protection: To withstand physical attacks on (parked) vehicles or RSUs, which may enable an adversary to extract the secret keys from their storage, then launch other attacks such as entity impersonation attacks.
3) Message confidentiality: To protect the vehicle's sensitive information included in the basic safety messages (BSMs) from leaking to an adversary.
4) Untraceablility: To preserve vehicle's privacy such that an adversary cannot analyze, link the intercepted messages (BSMs) to trace the vehicle.
5) Resistance to security attacks: To withstand the known passive, active attacks, particularly DoS, collusion, impersonation, replay, MITM attacks.

TABLE I
NOTATIONS

Notation	Description
TA	Trusted Authority
RSU_i, u_i	The i-th road side unit and its public index, respectively
$\mathcal{RT}$	Road side-Trusted authority unit
V_i, v_i	The i-th Vehicle and its private index, respectively
BSMs	Basic Safety Messages as in the US standard
K_{TA}	Main long-term secret key of TA
K_j	Secret offset key (Revocation key) in session j
K_{vi}	A uniform key from the PUF of V_i
$p(x,y)$,$q(x,y)$	Symmetric bivariate one-degree polynomials
ΔT	Maximum transmission delay; e.g., $\lvert T_1 - T_2\rvert \leq \Delta T$
$h(.)$	One-way cryptographic hash function
PUF	Physical Unclonable Function
$FE.Gen(.)$	Fuzzy extractor probabilistic generation function
$FE.Rep(.)$	Fuzzy extractor deterministic reproduction function
$\mathcal{E}_k[.]/\mathcal{D}_k[.]$	Symmetric encryption/decryption under key k
PTK_{ij}	Pairwise temporal secret key for entity i and entity j
sid_j	An identifier of a revocation session , $j \geq 1$
$\mathbb{F}_q$	A finite field with q elements, q is a prime
$\mathbb{F}_p^*$	A multiplicative group over finite field with order p
$\mathcal{V}$, $\mathcal{S}$	The set of all vehicles and road-side units, respectively
$\mathcal{R}_j$	The set of all revoked vehicles before and in session j
$\|, \oplus$	Concatenation and bitwise XOR functions, respectively

F. Security Assumptions

The proposed protocol is based on the following security assumptions:

- Each vehicle is equipped with a PUF. The PUF, the vehicle's on board unit (OBU) is considered a system-on chip (SoC). Any attempt to tamper with the device such as PUF separation that changes the device's behavior, consequently destroys the PUF.
- The RSU, TA cannot be compromised.
- Vehicles have limited resources (i.e., less computation capability, shorter transmission power, less storage) whereas the RSU, TA have no such resource limitations.

G. Security, Privacy Model

In this section, we present the formal security definitions, specify the security, privacy models used to analyze the proposed scheme. The most common notations used in this paper are listed in Table I.

1) Security Model: Consider a trusted authority TA, roadside units $\mathcal{S} = \{RSU_1, RSU_2, \ldots, RSU_m\}$, vehicles $\mathcal{V} = \{V_1, V_2, \ldots, V_n\}$. The $\mathcal{S}$ can interact with $\mathcal{V}$ whereas TA runs a setup algorithm $Setup(1^k)$, generates public parameters *pp*, secret parameters *sp*. Here, *pp* represents all public system parameters (i.e., PUF output length, coding mode, pseudo-random function (PRF) algorithm name, etc.), *sp* denotes the secret shared parameters (i.e., K_{TA}, K_j). Since each $RSU_i \in \mathcal{S}$ is connected to the TA through a secure connection such as the wired Transport Layer Security (TLS) protocol, hence we consider them (roadside unit - trusted authority) as a single unit $\mathcal{RT}$. In the authentication phase, mutual authentication is executed

between $V_i \in \mathcal{V}$, $\mathcal{RT}$. At the end of this phase, both parties output 1 (acceptance) or 0 (rejection) as the authentication result. A communication sequence between the $\mathcal{RT}$, the vehicle is called a session, a session identifier *sid* is used for distinguishing each session. A session is said to have a *matching session* if the messages exchanged between $\mathcal{RT}$, the vehicle are honestly transmitted until they eventually authenticate each other. That is to say, all communications have been unmodified by an adversary.

The *correctness* of the authentication protocol requires that both $V_i \in \mathcal{V}$ and $\mathcal{RT}$ always accept the session if it has the *matching session*. Following [30], we consider the security game (denoted by $Exp^{Sec}_{\Pi,\mathcal{A}}(k)$) between a challenger $\mathcal{C}$ and adversary $\mathcal{A}$ against an authentication protocol Π:

$Exp^{Sec}_{\Pi,\mathcal{A}}(k)$:

a) $(pp, sp) \xleftarrow{R} Setup(1^k)$;

b) $(sid^*, V_i) \xleftarrow{R} \mathcal{A}^{Launch, Send\mathcal{RT}, Send\mathcal{V}, Result, Reveal}(pp, \mathcal{RT}, \mathcal{V})$;

c) $b := Result(sid^*, V_i)$;

d) Output b

At the end of the setup phase, $\mathcal{A}$ can issue the following oracle queries

$\mathcal{O} := \{Launch, Send\mathcal{RT}, Send\mathcal{V}, Result, Reveal\}$, explained as follows:

- $Launch(1^k)$: A new session is started by $\mathcal{RT}$.
- $Send\mathcal{RT}(m)$: Send an arbitrary message m to $\mathcal{RT}$.
- $Send\mathcal{V}(V_i, m)$: Send a random message m to V_i.
- $Result(sid, \mathcal{P})$: Output whether the session *sid* of $\mathcal{P}$ is accepted or not, where $\mathcal{P} \in \{\mathcal{V}, \mathcal{RT}\}$.
- $Reveal(V_i)$: Output whole information contained in the OBU of vehicle V_i.

The advantage of an active adversary $\mathcal{A}$ against Π, $Adv^{Sec}_{\Pi,\mathcal{A}}(k)$, is defined by a probability $Pr[Exp^{Sec}_{\Pi,\mathcal{A}}(k)]$ outputs 1, provided that sid^* of $\mathcal{P}$ has no *matching session*.

Definition 2 (Security): An authentication protocol Π is secure against MITM attacks with complete memory leakage if for any probabilistic polynomial time adversary $\mathcal{A}$, $Adv^{Sec}_{\Pi,\mathcal{A}}(k)$ is negligible in k (for large enough k).

2) Privacy Model: Now we consider the indistinguishability-based privacy as in [30] wherein the adversary randomly selects two vehicles and tries to distinguish the communication originated from any one of them. The privacy experiment between the challenger $\mathcal{C}$ and adversary $\mathcal{A}:=(\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ is then described as follows:

$Exp^{IND^{*-b}}_{\Pi,\mathcal{A}}(k)$

a) $(pp, sp) \xleftarrow{R} Setup(1^k)$;

b) $(V_0^*, V_1^*, st_1) \xleftarrow{R} \mathcal{A}_1^{\mathcal{O}}(pp, \mathcal{RT}, \mathcal{V})$;

c) $b \xleftarrow{U} \{0,1\}, \mathcal{V}' := \mathcal{V} \setminus \{V_0^*, V_1^*\}$;

d) $\pi_0 \xleftarrow{R} Execute(\mathcal{RT}, V_0^*), \pi_1 \xleftarrow{R} Execute(\mathcal{RT}, V_1^*)$, $st_2 \xleftarrow{R} \mathcal{A}_2^{\mathcal{O}}(\mathcal{RT}, \mathcal{V}', \mathcal{I}(V_b^*), \pi_0, \pi_1, st_1)$;

e) $\pi_0' \xleftarrow{R} Execute(\mathcal{RT}, V_0^*), \pi_1' \xleftarrow{R} Execute(\mathcal{RT}, V_1^*)$;

f) $b' \xleftarrow{R} \mathcal{A}_3^{\mathcal{O}}(\mathcal{RT}, \mathcal{V}, \pi_0', \pi_1', st_2)$;

g) Output b'

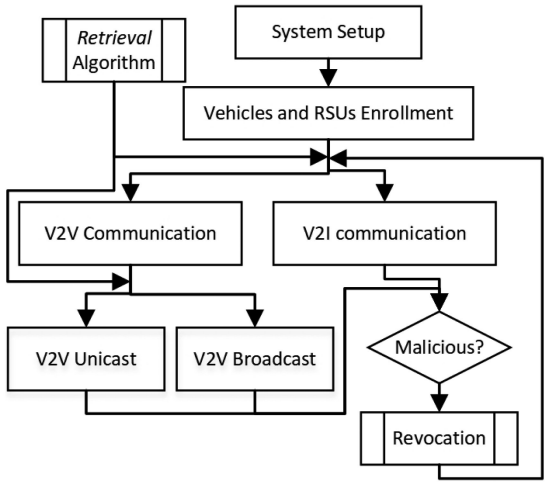

Fig. 2. The workflow of the proposed protocol

Similar to the security game, $\mathcal{A}$ is allowed to interact with $\mathcal{RT}$ and V_i via oracle queries in $\mathcal{O}$. Upon sending two vehicles (V_0^*, V_1^*) by an adversary $\mathcal{A}_1$ to the challenger $\mathcal{C}$, a random coin $b \xleftarrow{U} \{0,1\}$ is flipped by $\mathcal{C}$ and then the adversary can access the challenge vehicle V_b^* anonymously. For attaining anonymous access, $\mathcal{A}_2$ invokes $Send\mathcal{V}$ query with an intermediate algorithm $\mathcal{I}$ that honestly transmits the communication messages between $\mathcal{A}_2$ and V_b^*. When the adversary invokes $Send\mathcal{V}(\mathcal{I}, m)$, $\mathcal{I}$ sends m to the challenge vehicle V_b^* and responds with its output. Subsequent to the challenge phase, $\mathcal{A}_3$ can constantly interact with all vehicles, including (V_0^*, V_1^*), as $\mathcal{A}_1$. Hereafter, V_0^* and V_1^* invoke the *Execute* query, which is the normal protocol execution between the vehicle and the $\mathcal{RT}$ without an active adversary. The communication cannot be modified by the adversary except the transcripts (π_0, π_1) and (π_0', π_1') are delivered to the adversary. Thus, the advantage of the adversary in guessing the correct vehicle bit can be defined as

$Adv^{IND^*}_{\Pi,\mathcal{A}}(k) := |Pr[Exp^{IND^{*-0}}_{\Pi,\mathcal{A}}(k) \to 1] - Pr[Exp^{IND^{*-1}}_{\Pi,\mathcal{A}}(k) \to 1]|$

Definition 3 (Privacy): An authentication protocol Π satisfies the indistinguishability-based privacy under complete memory leakage if for any probabilistic polynomial time adversary $\mathcal{A}$, $Adv^{IND^*}_{\Pi,\mathcal{A}}(k)$ is negligible in k (for large enough k).

IV. Our Proposed Protocol

A. Overview

We propose a lightweight privacy-preserving message authentication protocol for VANETs based on combined PUF and Secret Sharing. The proposed protocol mainly consists of the following phases: (1) System setup; vehicles and RSUs Enrollment; (2) V2I mutual authentication and key renewal; (3) V2V authenticated secure message communication (which is further divided to two sub-phases V2V unicast or broadcast expeditious message authentication); and (4) Revocation mechanism. The main procedure, which has also illustrated in Fig. 2, can be described as follows: a network entity (i.e., a vehicle) first enrolls itself in the system via TA. Prior to sending or receiving a

message in V2I/V2V communication, the entity must invoke Algorithm 1 to make use of its PUF to reconstruct its secret polynomial-share. Hence the entity can establish PTKs and use them to broadcast or unicast encrypted BSMs. During V2V/V2I communication, TA will invoke the revocation mechanism if malicious found.

The advantages of PTKs established can be listed as follows: (a) PTKs ensure the legitimacy of a vehicle (namely, whether it is unrevoked or not), (b) PTKs guarantee confidentiality of BSMs by encrypting any BSMS prior to sending it, (c) traceability attacks mounted based on traditional plaintext/unencrypted BSMs can be also thwarted and hence a higher level of vehicle's privacy can be realized when employing PTKs to encrypt BSMs, (d) with PTKs, a vehicle is capable to broadcast or unicast safety messages with the features of being encrypted aforementioned, (e) owing to PTKs, parties can still recognize each other that renders DoS (de-synchronization) attacks ineffective, and (f) with using PTKs, a third party (i.e., RSU) is not necessarily required in V2V communications, particularly in urgent situations such as accidents or traffic jams, and therefore a vehicle in our design can send expeditious safety messages to a nearby vehicle(s).

B. Our Idea and Solutions

Traditional approaches in the related work still have the following pitfalls: 1) CRL used for revocation, which incurs much computation, communication, and storage overhead, 2) vulnerability to BSMs-based traceability attacks, 3) the need of a third party in V2V communication, 4) vulnerability to physical and DoS attacks. To address and resolve the above issues, our proposed message authentication protocol, which is designed based on combined PUF and Secret Sharing, replaces the costly CRL approach by an efficient revocation mechanism using Secret Sharing (the revocation status is constant). The proposed work employs PUF-based PTKs in encrypting BSMs during transmission to prevent traceability attacks. Furthermore, our scheme eliminates the need of a third party in V2V communication due to using PTKs. Additionally, by employing PUF, our protocol withstands physical and cloning attacks, as well as our work overcomes DoS attack using one-side update mechanism.

C. System Setup Phase

Trusted Authority (TA) first defines the (public) system parameters: a collision-resistant one-way hash function $h(.)$, fuzzy extractor generation/reproduction functions $FE.Gen(.)/FE.Rep(.)$, a finite field $\mathbb{F}_q$ with q elements and a generator g; q is a prime integer, a long-term secret key $K_{TA} \in \mathbb{F}_q$. Next, TA constructs two symmetric bivariate polynomials $p(x,y)$ and $q(x,y)$ over $\mathbb{F}_q[x,y]$ having degree one in both x and y; $p(x,y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{1,1}xy$, $q(x,y) = b_{0,0} + b_{1,0}x + b_{0,1}y + b_{1,1}xy$. The coefficients a_{ij} and b_{ij} are random elements in $\mathbb{F}_q$. $p(x,y)$ is personal secret and thus it is only generated based on PUF on demand. TA also selects its unique identifier τ and stores the tuple $(p(x,y), q(x,y), \tau)$ in its database. Finally, the system parameters $\{h(.), FE.Gen(.), FE.Rep(.), \mathbb{F}_q, \tau\}$ are made public.

D. Vehicle and Road-side Unite Enrollment

Consider n vehicles (denoted as $\mathcal{V} = \{V_1, V_2, \ldots, V_n\}$. For each vehicle $V_i \in \mathcal{V}$, TA performs the following:

a) TA first randomly and uniformly generates an identifier $v_i \in \mathbb{F}_q$ for V_i. Next, TA generates and sends a distinct challenge $C_{v_i} \in \mathbb{F}_q$ to V_i as a stimulus for its PUF_{v_i}. Upon receiving C_{v_i}, V_i inputs C_{v_i} into its PUF_{v_i} to get $R_{v_i} = PUF(C_{v_i})$, where R_{v_i} is the response produced by PUF_{v_i}, and then R_{v_i} is sent back to TA.

b) Upon receiving R_{v_i}, TA initially computes one polynomial-share from $p(x,y)$ as $p_{v_i}(y) = p(v_i, y) = m_{i0} + m_{i1}y \pmod q$, where v_i is a unique random element in $\mathbb{F}_q$ associated with each V_i, and the polynomial-share $p(v_i, y)$ is clearly a univariate polynomial of the same degree of $p(x,y)$.

c) Similarly, TA computes one polynomial-share from $q(x,y)$, $q_{v_i}(y) = q(v_i, y) \pmod q$ for V_i.

d) TA computes $(K_{v_i}, hd_{v_i}) = FE.Gen(R_{v_i})$, $\sigma_{v_i} = p(v_i, K_{v_i}).q(v_i, K_{v_i})$, and $\xi_{v_i} = p(v_i, 0) \oplus K_{v_i}$. It also encrypts the tuple $(v_i, g, h(K_{TA}))$ using K_{v_i}, $\beta_{v_i} = \mathcal{E}_{K_{v_i}}[v_i, g, h(K_{TA})]$ and sets the initial session identifier $sid_j = sid_0 = 0$. TA stores the real identity of V_i together with its corresponding identifier v_i, and then securely sends $(q(v_i, y), \beta_{v_i}, C_{v_i}, hd_{v_i}, \sigma_{v_i}, \xi_{v_i}, sid_j)$ to V_i.

Consider m road-side units (denoted as $\mathcal{S} = \{RSU_1, RSU_2, \ldots, RSU_m\}$. Similar to V_i registration, the same steps are required to enroll $RSU_i \in \mathcal{S}$.

E. Algorithm 1: Retrieval

This algorithm is invoked by a participant, which is a vehicle or RSU say P_i, to reconstruct its secret polynomial-share based on PUF and homomorphism property. Let $p(i,y) = m_0 + m_1 y$ denote the secret polynomial-share of P_i with an identifier i. First, P_i stimulates its PUF_i using a stored challenge C_i to obtain $R_i = PUF(C_i)$ and then computes a PUF-based key using *FE*, $K_i = FE.Rep(R_i, hd_i)$. Hence P_i's private identifier i alongside the secret generator g can be recovered by decrypting β_i using K_i, $[i, g, h(K_{TA})] = \mathcal{D}_{K_i}(\beta_i)$. The entity P_i can then recover the 1^{st} coefficient of its polynomial using K_i, $m_0 = p(i,0) = \xi_i \oplus K_i$. Next, the 2^{nd} coefficient can be recovered as follows. $p(i, K_i) = \sigma_i / q(i, K_i)$, $|m_1| = ((p(i, K_i) - p(i,0))/K_i$. Thus, $p(i,y)$ is simply constructed as $p(i,y) = m_0 + m_1 y \pmod q$. Finally, this algorithm returns $(i, p(i,y))$ to the main protocol.

F. V2I Mutual Authentication and Key Renewal Phase

In this phase, a vehicle V_i gets authenticated by $\mathcal{RT}$ and its personal secrets get updated as well. This phase is triggered when V_i enters a new RSU's coverage area and/or for a periodic updating of V_i's secrets in the same coverage area. The following steps, which are also summarized in Table II, are required for V2I authentication with secret keys updating:

Step 1: (Authentication Request). $\mathcal{RT}$ randomly chooses a nonce N_1 and sends it to a vehicle V_i over a public channel.

Algorithm 1: *Retrieval.*

Input: β_i, C_i, $q(i,y)$, ξ_i, σ_i
Output: i, $p(i,y)$
$R'_i = PUF(C_i)$
$K_i = FE.Rep(R'_i, hd_i)$
$[i, g, h(K_{TA})] = \mathcal{D}_{K_i}(\beta_i)$
$m_0 = p(i,0) = \xi_i \oplus K_i$ /* the 1^{st} coefficient */
$p(i, K_i) = \dfrac{\sigma_i}{q(i, K_i)}$
$|m_1| = \dfrac{p(i,K_i) - p(i,0)}{K_i}$ /* the 2^{nd} coefficient */
Construct $p(i,y) = m_0 + m_1 y \pmod q$
return $(i, p(i,y))$

Step 2: **(Authentication Response)**. Upon receiving N_1, V_i first invokes the *Retrieval* algorithm (see Algorithm 1) to recover $[v_i, p(v_i,y) \leftarrow Retrieval]$. V_i computes its temporary identity $TID_{v_i} = h(h(K_{TA})\|T_1) \oplus v_i$. After that, V_i extracts the offset key K_j, which has been distributed by TA for the latest revocation, from its masked value, $K_j = K_{v_i} \oplus MK_j$. Hereafter, V_i checks the revocation session identifier by the condition (if $sid_j > 0$) that is always valid except the initial authentication session executed after the enrollment in which the two following steps will be skipped over. Thus, V_i adds K_j to its $p(v_i,y)$; shifting $p(v_i,y)$ so as to establish pairwise keys with only legitimate vehicles and/or with *RSUs*, $p(v_i,y) = p(v_i,y) + K_j \pmod q$. Showing it is a valid entity, V_i computes its revocation status $REV_{status} = HMAC(K_j, TID_i\|T_1)$ using HMAC under K_j on $TID_i\|T_1$, where T_1 is the current timestamp. V_i further computes the pairwise temporal key with the nearby RSU_i as $PTK_{vu} = p(v_i, u_i) \pmod q$ and $PTK_{vu} = h(PTK_{vu}\|REV_{status})$. Similarly, a pairwise temporal key is also established with TA to guarantee strong security in case the RSU_i is compromised, $PTK_{v\tau} = p(v_i, \tau) \pmod q$ and $PTK_{v\tau} = h(PTK_{v\tau}\|REV_{status})$. V_i also computes $C^{new}_{v_i} = h(C_{v_i}\|K_{v_i})$, $R^{new}_{v_i} = PUF(C^{new}_{v_i})$. Hereafter, V_i picks N_2 and computes $\Theta_1 = \mathcal{E}_{PTK_{v\tau}}[R^{new}_{v_i}, N_1, N_2]$, $\varphi_1 = HMAC(PTK_{vu}, TID_{v_i}\|\Theta_1\|sid_j\|N_1\|T_1)$. Finally, V_i sends the message $Msg_2 = (TID_{v_i}, \Theta_1, \varphi_1, sid_j, T_1)$ to the $\mathcal{RT}$, namely RSU_i, over a public channel.

Step 3: (Vehicle Authentication). Upon receiving Msg_2, $\mathcal{RT}$ first checks the freshness of timestamp T_1 using ΔT and continues the session only if it's valid. Then, $\mathcal{RT}$ recovers V_i's identifier $v_i = h(h(K_{TA}\|T_1) \oplus TID_{v_i})$. Similar steps to V_i's steps will be performed here up to obtaining $p(u_i,y) = p(u_i,y) + K_j \pmod q$, then $\mathcal{RT}$ performs *revocation status checking* by HMAC REV_{status}. If V_i is unrevoked, $\mathcal{RT}$ proceeds and computes $PTK_{vu} = p(u_i, v_i) \pmod q$, $PTK_{vu} = h(PTK_{vu}\|REV_{status})$, and the first verification is performed on φ_1. If it is successful, then Θ_1 is decrypted by $\mathcal{RT}$, which is TA in this step, $[R^{new}_{v_i}, N_1, N_2] = \mathcal{D}_{PTK_{v\tau}}(\Theta_1)$.

After that, the second verification is performed by TA using checking N_1. If the verification holds, TA computes $(K^{new}_{v_i}, hd^{new}_{v_i}) = FE.Gen(R^{new}_{v_i})$, $\sigma^{new}_{v_i} = (p(v_i,y) + q(v_i,y))(K^{new}_{v_i})$, $\xi^{new}_{v_i} = p(v_i,0) \oplus K^{new}_{v_i}$, $\beta^{new}_{v_i} = \mathcal{E}_{K^{new}_{v_i}}[v_i, h(K_{TA})]$, $\Theta_2 = \mathcal{E}_{PSKv\tau}[hd^{new}_{v_i}, \sigma^{new}_{v_i}, \xi^{new}_{v_i}, \beta^{new}_{v_i}]$, and $\varphi_2 = HMAC(N_2, \Theta_2\|T_2)$. At last, $\mathcal{RT}$, that is RSU_i, sends $Msg_3 = (\Theta_2, \varphi_2, \beta^{new}_{v_i}, T_2)$ to V_i over a public channel.

Step 4: (Mutual Authentication and Secrets Updating Accomplished). After receiving Msg_3, V_i checks T_2 by ΔT and rejects if it is not fresh. V_i verifies $\mathcal{RT}$ by checking the condition φ_2. If the condition holds true, V_i authenticates the RSU_i and then it decrypts Θ_2 to get the new updates of its secrets, $[hd^{new}_{v_i}, \sigma^{new}_{v_i}, \xi^{new}_{v_i}, \beta^{new}] = \mathcal{D}_{PTK_{v\tau}}(\Theta_2)$. Finally, only V_i updates its secrets by the received ones, $C_{v_i} = C^{new}_{v_i}$, $hd_{v_i} = hd^{new}_{v_i}$, $\sigma_{v_i} = \sigma^{new}_{v_i}$, and $\xi_{v_i} = \xi^{new}_{v_i}$, $\beta_{v_i} = \beta^{new}_{v_i}$. Remarkably, the proposed scheme is free from possible de-synchronization attacks during secret keys updating due to deploying our *one-side secret updating* mechanism. That is, $\mathcal{RT}$ only sends (not stores) the updates to V_i, however, they can still recognize each other (also see Section V-C).

G. V2V Authenticated Secure Message Communication Phase

1) V2V Unicast Expeditious Message Authentication: Upon entering RSU_i's jurisdiction, V_i initially receives information from RSU_i about other vehicles within the transmission coverage range of this RSU_i. Thus, V_i utilizes this information, which will be being broadcast frequently by RSU_i, to initiate V2V unicast immediate secure message transmission in some urgent situations such as accident-related information transmission. The following steps, which are also illustrated in Table III, are required for V2V message authentication:

Step 1: (Authenticated Secure Message Transmission). Similar steps to those performed by V_i (which are also described in Table II) are executed here and thus V_i obtains its $p(v_i,y)$ with K_j added to it and computes its $REV_{status} = HMAC(K_j, TID_i\|T)$, where T is the current timestamp. V_i then computes a pairwise temporal key with the intended vehicle V_j using its v_j as $PTK_{ij} = p(v_i, v_j) \pmod q$ and $PTK_{ij} = h(PTK_{ij}\|T)$. Afterward, V_i encrypts a message m, which could be BSM, $\mathcal{M} = \mathcal{E}_{PTK_{ij}}[m]$. V_i also calculates $\delta = HMAC(PTK_{ij}, TID_{v_i}\|\mathcal{M}\|sid\|T)$ and lastly sends $Msg = (TID_{v_i}, \mathcal{M}, sid, \delta, T)$ to vehicle V_j publicly.

Step 2: (Message Verification). Upon receiving Msg, V_j first checks the freshness of timestamp T by ΔT. If it is not fresh, the current session is terminated; otherwise, V_j recovers the sender's identifier $v_i = h(h(K_{TA})\|sid\|T) \oplus TID_{v_i}$ and follows the same steps as V_i to retrieve its $p(v_j,y)$, adding K_j to it. Now, V_j verifies V_i's revocation status by checking the condition $REV_{status} \stackrel{?}{=} HMAC(K_j, TID_i\|T)$ and further proceeds (only if V_i is unrevoked) with computing $PTK_{ij} = p(v_j, v_i) \pmod q$, $PTK_{ij} = h(PTK_{ij}\|REV_{status})$, and $\delta \stackrel{?}{=} HMAC(PTK_{ij}, TID_{v_i}\|REV_{status}\|\mathcal{M}\|T)$. If the δ holds true, V_j accepts the received message and decrypts $\mathcal{M}$; obtaining the message m (i.e., BMS), $m = \mathcal{D}_{PTK_{ij}}(\mathcal{M})$.

Significantly, our scheme eliminates the need for third-party involvement in V2V communication even with employing only symmetric cryptography (HMAC, XOR, and PUF). Consequently, a vehicle expedites sending authenticated, secure, and confidential instant messages, specifically in transferring traffic-related information directly to a particular vehicle.

TABLE II
V2I MUTUAL AUTHENTICATION AND SECRET KEYS RENEWAL

Vehicle V_i	**RSU-TA Unit** $\mathcal{RT}$	
	Generate nonce N_1	
$[v_i, p(v_i, y)] \leftarrow Retrieval(\beta_{v_i}, C_{v_i}, q(v_i, y), \xi_{v_i}, \sigma_{v_i})$	$\xleftarrow[\text{via public channel}]{\boldsymbol{Msg_1} = (\boldsymbol{N_1})}$	
$TID_{v_i} = h(h(K_{TA})\|T_1) \oplus v_i$		
If $(sid_j > 0)$ do	Check T_1; terminate if not valid	*By* RSU_i
$K_j = K_{v_i} \oplus MK_j$	$v_i = h(h(K_{TA})\|T_1) \oplus TID_{v_i}$	
$p(v_i, y) = p(v_i, y) + K_j \pmod q$	$[u_i, p(u_i, y)] \leftarrow Retrieval(\beta_{u_i}, C_{u_i}, q(u_i, y), \xi_{u_i}, \sigma_{u_i})$	
$REV_{status} = HMAC(K_j, TID_{v_i}\|T_1)$	If $(sid_j > 0)$ do	
$PTK_{vu} = p(v_i, u_i) \pmod q$	$K_j = K_{u_i} \oplus MK_j$	
$PTK_{vu} = h(PTK_{vu}\|REV_{status})$	$p(u_i, y) = p(u_i, y) + K_j \pmod q$	
$PTK_{v\tau} = p(v_i, \tau) \pmod q$	Verify $REV_{status}? = HMAC(K_j, TID_{v_i}\|T_1)$	
$PTK_{v\tau} = h(PTK_{v\tau}\|REV_{status})$	$PTK_{vu} = p(u_i, v_i) \pmod q$	
$C_{v_i}^{new} = h(C_{v_i}\|K_{v_i})$	Verify φ_1; 1^{st} Verification	
$R_{v_i}^{new} = PUF(C_{v_i}^{new})$	$[R_{v_i}^{new}, N_1, N_2] = \mathcal{D}_{PTK_{v\tau}}(\Theta_1)$	*By TA*
Generate nonce N_2	Check N_1;2^{nd} Verification	
$\Theta_1 = \mathcal{E}_{PTK_{v\tau}}[R_{v_i}^{new}, N_1, N_2]$	$(K_{v_i}^{new}, hd_{v_i}^{new}) = FE.Gen(R_{v_i}^{new})$	
$\varphi_1 = HMAC(PTK_{vu}, TID_{v_i}\|\Theta_1\|REV_{status}\|T_1)$	$\sigma_{v_i}^{new} = (p(v_i, y) + q(v_i, y))(K_{v_i}^{new})$	
$\xrightarrow[\text{via public channel}]{\boldsymbol{Msg_2} = (\boldsymbol{TID_{v_i}}, \boldsymbol{\Theta_1}, \boldsymbol{\varphi_1}, \boldsymbol{REV_{status}}, \boldsymbol{T_1})}$	$\xi_{v_i}^{new} = p(v_i, 0) \oplus K_{v_i}^{new}$	
	$\beta_{v_i}^{new} = \mathcal{E}_{K_{v_i}^{new}}[v_i, g, h(K_{TA})]$	
Check T_2, φ_2	$\Theta_2 = \mathcal{E}_{PSKv\tau}[hd_{v_i}^{new}, K_{v_i}^{new}, \sigma_{v_i}^{new}, \xi_{v_i}^{new}]$	
$[hd_{v_i}^{new}, K_{v_i}^{new}, \sigma_{v_i}^{new}, \xi_{v_i}^{new}] = \mathcal{D}_{PTK_{v\tau}}(\Theta_2)$	$\varphi_2 = HMAC(N_2, \Theta_2\|\beta_{v_i}^{new}\|T_2)$	
Update $C_{v_i} = C_{v_i}^{new}, MK_j^{new} = K_j \oplus K_{v_i}^{new}$,	$\xleftarrow[\text{via public channel}]{\boldsymbol{Msg_3} = (\boldsymbol{\Theta_2}, \boldsymbol{\varphi_2}, \boldsymbol{\beta_{v_i}^{new}}, \boldsymbol{T_2})}$	
$hd_{v_i} = hd_{v_i}^{new}, \sigma_{v_i} = \sigma_{v_i}^{new}, \xi_{v_i} = \xi_{v_i}^{new}, \beta_{v_i} = \beta_{v_i}^{new}$		

TABLE III
V2V UNICAST EXPEDITIOUS MESSAGE AUTHENTICATION

Vehicle V_i	**Vehicle** V_j
$[v_i, p(v_i, y)] \leftarrow Retrieval(\beta_{v_i}, C_{v_i}, q(v_i, y), \xi_{v_i}, \sigma_{v_i})$	
$TID_{v_i} = h(h(K_{TA})\|T) \oplus v_i$	Check T; terminate if not valid
If $(sid_j > 0)$ do	$v_i = h(h(K_{TA})\|T) \oplus TID_{v_i}$
$K_j = K_{v_i} \oplus MK_j$	$[v_j, p(v_j, y)] \leftarrow Retrieval(\beta_{v_j}, C_{v_j}, q(v_j, y), \xi_{v_j}, \sigma_{v_j})$
$p(v_i, y) = p(v_i, y) + K_j \pmod q$	If $(sid_j > 0)$ do
$REV_{status} = HMAC(K_j, TID_{v_i}\|T)$	$K_j = K_{v_j} \oplus MK_j$
$PTK_{ij} = p(v_i, v_j) \pmod q$	$p(v_j, y) = p(v_j, y) + K_j \pmod q$
$PTK_{ij} = h(PTK_{ij}\|REV_{status})$	Verify $REV_{status}? = HMAC(K_j, TID_{v_i}\|T)$
$\mathcal{M} = \mathcal{E}_{PTK_{ij}}(m)$	$PTK_{ij} = p(v_j, v_i) \pmod q$
$\delta = HMAC(PTK_{ij}, TID_{v_i}\|REV_{status}\|\mathcal{M}\|T)$	$PTK_{ij} = h(PTK_{ij}\|REV_{status})$
$\xrightarrow[\text{via public channel}]{< \boldsymbol{Msg} = (\boldsymbol{TID_{v_i}}, \boldsymbol{REV_{status}}, \mathcal{M}, \boldsymbol{\delta}, \boldsymbol{T}) >}$	Verify $\delta? = HMAC(PTK_{ij}, TID_{v_i}\|\mathcal{M}\|sid\|T)$
	$m = \mathcal{D}_{PTK_{ij}}(\mathcal{M})$;

2) V2V Broadcast Secure Message Transmission: Consider a vehicle V_i with an identifier v_i has a message m, i.e., BSM, to broadcast. It first invokes Algorithm 1 to recover its $p(v_i, y)$. Afterward, V_i executes Algorithm 2 and eventually broadcasts $\mathcal{B}_i = \{TID_{v_i}, REV_{status}, \omega(y), \mathcal{M}, T, \eta\}$ to all vehicles and the RSU in its vicinity. A receiver of the broadcast $\mathcal{B}(y)$ initially calls Algorithm 1 to recover its $p(v_j, y)$ and then executes Algorithm 3. Obviously, only a receiver with a valid $p(v_j, y)$, i.e., unrevoked vehicles, can establish pairwise secret keys ($p(v_j, v_i)$, $q(v_j, v_i)$) with the sender and therefore can recover the broadcast encryption key g^r. HMAC function is generally used to verify the authenticity of a sender and the integrity of a message. If HMAC η' matches η, Algorithm 3 returns the decrypted m from $\mathcal{M}$; otherwise, it returns $\perp$.

H. Revocation Mechanism Phase

The proposed scheme introduces a revocation mechanism whose revocation checking process in the main protocol has constant computation complexity $\mathcal{O}(1)$. The revocation phase is triggered by TA when there exists at least one malicious vehicle to be revoked from the system. In our scheme, TA needs not publishing CRLs. Alternatively, TA securely sends a tuple of revocation data, including an offset/update key K_j, to all RSUs

Algorithm 2: *Broadcast Encryption* (invoked by V_i).

Input: $p(v_i, y) \leftarrow p(v_i, y) + K_j \pmod q$, $q(v_i, y)$, m
Output: $\mathcal{B}_i$
Picks randomly $r \in \mathbb{F}_q$
$Q(y) = p(v_i, y) + q(v_i, y) + r \pmod q$
$\omega(y) = g^{Q(y)} \pmod q$
$\mathcal{M} = \mathcal{E}_{g^r}[m]$
$TID_{v_i} = h(h(K_{TA})\|T) \oplus v_i$
$REV_{status} = HMAC(K_j, TID_{v_i}\|T)$
$\eta = HMAC(g^r, TID_{v_i}\|REV_{status}\|\omega(y)\|\mathcal{M}\|T)$
$\mathcal{B}_i = \{TID_{v_i}, REV_{status}, \omega(y), \mathcal{M}, T, \eta\}$
return $\mathcal{B}_i$

Algorithm 3: *Broadcast Decryption* (executed by V_j).

Input: $p(v_j, y) \leftarrow p(v_j, y) + K_j \pmod q$, $q(v_j, y)$, $\mathcal{B}_i$
Output: m or $\perp$
Check the validity of timestamp T
Recover sender's $v_i = h(h(K_{TA})\|T) \oplus TID_{v_i}$
Verify $REV_{status} \stackrel{?}{=} HMAC(K_j, TID_{v_i}\|T)$
Establish the pairwise secret keys $p(v_j, v_i)$, $q(v_j, v_i)$
Evaluate $\omega(y)|_{y=v_j} = g^{Q(v_j)} \pmod q = g^{p(v_i,v_j)+q(v_i,v_j)+r}$
Divide step 5 by step 4 $\frac{g^{p(v_i,v_j)+q(v_i,v_j)+r}}{g^{p(v_j,v_i)+q(v_j,v_i)}} = g^{((p(v_i,v_j)+q(v_i,v_j)+r)-p(v_j,v_i)+q(v_j,v_i))} = g^r$
$\eta' = HMAC(g^r, TID_{v_i}\|REV_{status}\|\omega(y)\|\mathcal{M}\|T)$
if $(\eta' == \eta))$ **then**
  Decrypt to obtain $m = \mathcal{D}_{g^r}(\mathcal{M})$
  return m
return $\perp$

in the network. Each RSU utilizes the Secret Sharing-based Session Group Key Distribution (SGKD) below to distribute the offset key (namely a revocation or update key) K_j by which a revoked vehicle cannot get authenticated/their messages accepted by others any longer.

Secret Sharing-based SGKD: Broadcast with Revocation Capability using Secret Sharing. In this approach, our novel idea is that we separate the whole system into several groups based on RSUs coverage areas. Subsequently, $\{RSU_l\}_{l=1,2,\ldots,m}$, after receiving the revocation information for a session j from TA, constructs its own threshold secret sharing-based access polynomial $\phi_j^l(x)$ according to the legitimate vehicles in its vicinity. That is to say, $\phi_j^l(x)$ only passes the pairwise secret keys established with non-revoked vehicles in its coverage area. Therefore, our protocol ensures an efficient revocation technique with no communication overhead. The following steps, which are also illustrated in Fig 3, are required for the revocation mechanism:

1) *Broadcast*. TA picks a random $k_j \in \mathbb{F}_q$, where the j-th session offset key (revocation key) is $K_j = g^{k_j} \in \mathbb{F}_q$. Then, TA sends the tuple $(K_j, sid_j, \mathcal{R}_j)$ to each $\{RSU_l\}_{l=1,2,\ldots,m}$ securely.
   First, let $\mathcal{G}_j^l = \{v_{i_1}, v_{i_2}, \ldots, v_{i_{t_j}}\}$ denotes a set of all non-revoked vehicles' identifiers in the coverage area of RSU_l for session j. Upon receiving $(K_j, sid_j, \mathcal{R}_j)$ from TA, RSU_l invokes Algorithm 1 to recover its $p(u_l, y)$ and then constructs a random t-th degree polynomial $\phi_j^l(x) \in \mathbb{F}_q[x]$ to pass through $(t+1)$ coordinates, $(0, K_j)$ and $(v_i, g^{p(u_l,v_i)})$; for all $v_i \in \mathcal{G}_j^l$ and where $t = |\mathcal{G}_j^l|$. RSU_l also computes t additional points P_d on $\phi_j^l(x)$ and $\alpha = h(K_j\|h(K_{TA}))$, where $\{P_d\}_{d=1,2,\ldots,t}$. Lastly, RSU_l broadcasts $\mathcal{B}_j^l = \{\{P_d\}_{d=1,2,\ldots,t}\|sid_j\|HMAC(\alpha, sid_j\|\{P_d\}_{d=1,2,\ldots,t})\}$.
2) *j-th session offset key recovery*. When $V_i \in \mathcal{G}_j^l$ receives the broadcast of its nearby RSU_l, it initially recovers its $p(v_i, y)$ by invoking Algorithm 1. After that, V_i establishes a pairwise key with RSU_l as $PTK_{vu} = p(v_i, u_l)$. If $V_i \in \mathcal{G}_j^l$, it then knows the point $(v_i, g^{p(v_i,u_l)})$ plus the t additional public points, $\{P_d\}_{d=1,2,\ldots,t}$, on $\phi_j^l(x)$. Thus, only vehicles in $\mathcal{G}_j^l$ can interpolate $\phi_j^l(x)$ and recover $K_j = \phi_j^l(0)$ for a session j. V_i proceeds with checking the authenticity and integrity of the received broadcast by computing $\alpha = h(K_j\|h(K_{TA}))$ and then verifying $HMAC(\alpha, sid_j\|\{P_d\}_{d=1,\ldots,t})$. If it matches true, V_i authenticates that K_j is sent from the nearby RSU_l.

Consequently, V_i keeps using K_j as a secure shift/update value added to its private $p(v_i, y)$ in all successive authentication sessions, proving to others it is a legitimate entity and it continues as such up to the time wherein a new offset key is being broadcast by RSU_l in the vicinity. It's worth mentioning that in traditional CRLs-based revocation or other revocation approaches, a revoked vehicle V_i can never rejoin the system whereas in our scheme a revoked V_i can rejoin the system again if TA permits it by removing its identifier from the revocation list $\mathcal{R}_j$ in the next sessions.

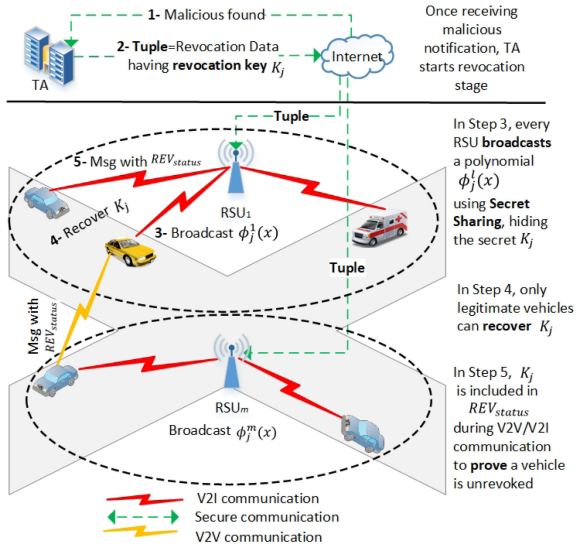

Fig. 3. Revocation mechanism of the proposed protocol.

V. Security Analysis

In this section, we provide the formal and informal security analyses of the proposed protocol. Additionally, we provide both

the simulation details of our protocol using AVISPA tool and the mutual authentication validation using BAN-logic in the supplemental materials.

A. Formal Security Analysis of the Proposed Scheme

We follow a similar game transformations-based security proof as that described in [30].

Theorem 1 (Security): Consider a $(d, n, l, \lambda, \epsilon)$-secure PUF and let *FE* be a (d, λ, ϵ)-secure fuzzy extractor, $h(.)$ be a secure pseudorandom function, and the symmetric-key encryption scheme, Ω, be IND-CPA secure. Then, the proposed protocol is secure against MITM attacks with memory leakage.

Proof: The goal of adversary $\mathcal{A}$ is to break the security game and deceive V_i and $\mathcal{RT}$ to accept the session while there is no *matching session*, i.e., the communication is modified by $\mathcal{A}$. We consider the following game transformations. Let χ_i denotes the advantage that the adversary wins the game in Game i.

Game 0. It represents the original game between the challenger $\mathcal{C}$ and the adversary $\mathcal{A}$.

Game 1. The challenger $\mathcal{C}$ randomly guesses the vehicle $V^* \xleftarrow{U} \{V_1, V_2, \ldots, V_n\}$. $\mathcal{C}$ aborts the game if $\mathcal{A}$ cannot impersonate V^* to $\mathcal{RT}$ or has a different sid^*.

Game 2. Let ℓ be the upper bound of the number of sessions that $\mathcal{A}$ can establish in the game. For $1 \leq j \leq \ell$, we evaluate or modify the parameters related to the session between $\mathcal{RT}$ and V^* up to the ℓ-th session as follows.

- *Game 2-j-1.* Challenger $\mathcal{C}$ evaluates the output from the PUF embedded in V^* at the j-th session. $\mathcal{C}$ terminates the game if the output of the PUF doesn't produce enough entropy m or does not satisfy the requirements for PUF inter and intra distances.
- *Game 2-j-2.* The output from the fuzzy extractor ($K_{v_i}^{new}$, $hd_{v_i}^{new}$) is turned into a random variable.
- *Game 2-j-3.* The output from the encryption scheme Ω $\Theta_1 = \mathcal{E}_{PTK_{v\tau}}[.]$ is derived from a truly random function.
- *Game 2-j-4.* Similarly, the output from the Ω $\Theta_2 = \mathcal{E}_{N_2}[.]$ is derived from a truly random function.
- *Game 2-j-5.* The output from the pseudorandom function (PRF) $HMAC(PTK_{uv}, .)$ is derived from a truly random function in this game.
- *Game 2-j-6.* The output from the PRF $HMAC(N_2, .)$ is obtained from a truly random function.

The basic strategy of the security proof is to alter the exchanging messages corresponding to the target vehicle V^* to random strings such that an attacker cannot distinguish the arbitrary strings from real messages. The game transformation starts from the first invocation of the vehicle V^* and proceeds from Game 2-j-1 to Game 2-j-6. Upon finishing these game transformations, we can move to the next session. This strategy can be recursively implemented up to the upper bound ℓ. *Note* if the PUF embedded in the vehicle outputs enough entropy, the fuzzy extractor can generate variables that are statistically close to random. Vehicle V^* then utilizes this output to retrieve its secret $p(v_i, y)$ using Algorithm 1; establishing a pair-wise temporal key with $\mathcal{RT}$ which can also be used for symmetric encryption thereafter. *Therefore, an adversary's advantage against our authentication protocol is negligible as shown in the following lemmas.*

Lemma 1: $\chi_0 = n\chi_1$ if the number of vehicles is n.

Proof: The adversary $\mathcal{A}$ wins the game if there is at least one session accepted by $\mathcal{RT}$ or V^* when the communication is modified by the adversary. Since we assume there are n vehicles and the challenger $\mathcal{C}$ randomly selects a vehicle, the probability that $\mathcal{C}$ correctly guessed the vehicle impersonated by $\mathcal{A}$ is at least $1/n$. ■

Lemma 2: $|\chi_1 - \chi_{2-1-1}| \leq \epsilon$ and $|\chi_{2-j-1} - \chi_{2-(j-1)-6}| \leq \epsilon$ hold for any $2 \leq j \leq \ell$ if the PUF embedded in the vehicle is (d, n, l, λ, ϵ)-secure PUF.

Proof: There is no difference in these games since the output of PUF has enough min-entropy and is independent of other outputs, namely, (d, n, l, λ, ϵ)-secure PUF has enough min-entropy larger than λ, intra-distance smaller than d, and inter-distance larger than d. Additionally, PUF is assumed to have a desirable property (described in Def. 1) that even if the input to the PUF is disclosed, the output derived from the input fulfills the requirement of sufficient min-entropy, and thus each output is uncorrelated.

Since the games χ_1, χ_{2-1-1}, $\chi_{2-(j-1)-6}$, and χ_{2-j-1} assume the aforementioned PUF conditions, the gap between them is bounded by the negligible probability ϵ. Therefore, the game transformation can proceed further without any negative effect even if $\mathcal{A}$ invokes the $Reveal$ query and obtains the secrets stored in the OBU of V^*. ■

Lemma 3: $|\chi_{2-j-1} - \chi_{2-j-2}| \leq \epsilon$ holds for any $(1 \leq j \leq \ell)$ if the Fuzzy Extractor(FE) is a (d, λ)-FE.

Proof: As explained in the proof of Lemma 2 about the assumption of (d, n, l, λ, ϵ)-secure PUF, then (d, λ, ϵ)-fuzzy extractor (*FE*), which is based on PUF outputs, guarantees an output that is statistically close to random. As a result, no adversary can distinguish these two games χ_{2-j-1} and χ_{2-j-2} due to the randomization property of the *FE*. ■

Lemma 4: $|\chi_{2-j-2} - \chi_{2-j-3}| \leq Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k)$ for all $(1 \leq j \leq l)$, where $Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k)$ denotes an advantage of $\mathcal{B}$ to break the security of *IND-CPA* Ω.

Proof: We construct an algorithm $\mathcal{B}$ which breaks the security of our encryption scheme Ω. $\mathcal{B}$ can access the real encryption/decryption or truly random function algorithms $\mathcal{E}_k(.)/\mathcal{D}_k(.)$ or **RF**, respectively. $\mathcal{B}$ sets up all secret parameters, simulating the proposed protocol except the *n*-th session. Once invoking the *n*-th session by the adversary $\mathcal{A}$, $\mathcal{B}$ sends $N_1 \xleftarrow{U} \{0,1\}^k$ as the output of $\mathcal{RT}$. When $\mathcal{A}$ sends $N_1^{\#}$ to the vehicle V^*, $\mathcal{B}$ proceeds the computation as per the protocol specification and issues $N_1^{\#}$ to the oracle rather than the normal computation of $\Theta_1 = \mathcal{E}_{PTK_{v\tau}}[.]$. Upon receiving Θ_1, $\mathcal{B}$ outputs $\{TID_{vi}, \varphi_1, \Theta_1, REV_{status}, T_1\}$ as the V^*'s response. When $\mathcal{A}$ sends $\{TID_{vi}^{\#}, \varphi_1{}^{\#}, \Theta_1^{\#}, REV_{status}^{\#}, T_1^{\#}\}$ to the $\mathcal{RT}$, $\mathcal{B}$ issues N_1 to the oracle and obtains Θ_1.

If $\mathcal{B}$ accesses the real $\mathcal{E}_k(.)/\mathcal{D}_k(.)$, this simulation is equivalent to Game 2-j-2. Otherwise, the oracle query invoked by $\mathcal{B}$ is completely random and thus this distribution is similar to Game 2-j-3. Hence we have $|\chi_{2-j-2} - \chi_{2-j-3}| \leq Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k)$ ■

Lemma 5: $|\chi_{2-j-3} - \chi_{2-j-4}| \leq Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k)$ for all $(1 \leq j \leq l)$, where $Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k)$ denotes an advantage of $\mathcal{B}$ to break the security *IND-CPA* Ω.

Proof: Similarly, this lemma can be proven as the proof for Lemma 4 ■

Lemma 6: $|\chi_{2-j-4} - \chi_{2-j-5}| \leq Adv_{HMAC(.),\mathcal{B}'}^{PRF}(k)$ for all $(1 \leq j \leq \ell)$, where $Adv_{HMAC(.),\mathcal{B}'}^{PRF}(k)$ denotes an advantage of $\mathcal{B}'$ to break the security $PRF\ HMAC(.)$.

Proof: Obviously, the input to the PRF $HMAC(PTK_{vu}, .)$ is turned into random from the previous games. Yet, if there is a difference between these games, we construct an algorithm $\mathcal{B}$, as in the proof for Lemma 4, which breaks the security of PRF $HMAC(.)$. $\mathcal{B}$ can interact with the real PRF $HMAC(PTK_{vu}, .)$ or a truly random function **RF**. $\mathcal{B}$ generates (φ_1, Θ_1) and issues $\varphi_1 \| \Theta_1$ to the oracle. Like Game 5, $\mathcal{B}$ also generates the other variables and sends $(TID_{vi}, \varphi_1, \Theta_1, REV_{status}, T_1)$ as V^*'s output after obtaining φ_1 from the oracle. If $\mathcal{RT}$ receives $(TID_{vi}^{\#}, \varphi_1{}^{\#}, \Theta_1^{\#}, REV_{status}^{\#}, T_1^{\#})$, $\mathcal{B}$ inspects $(\Theta_1^{\#}, \varphi_1{}^{\#})$? $= (\Theta_1, \varphi_1)$. If so, $\mathcal{B}$ issues $TID_{vi}^{\#} \| \varphi_1{}^{\#} \| \Theta_1^{\#} \| REV_{status}^{\#} \| T_1^{\#}$ to the oracle, examining if its response is identical to $\varphi_1^{\#}$. ■

Lemma 7: $|\chi_{2-j-5} - \chi_{2-j-6}| \leq Adv_{HMAC(.),\mathcal{B}'}^{PRF}(k)$ for all $(1 \leq j \leq \ell)$.

Proof: We can prove this lemma as the proof for Lemma 6. ■

There is no advantage for the adversary to break the security when we transform Game 0 to Game 2-ℓ-6 since those games are bounded by assumptions, i.e., secure PUF and fuzzy extractor, IND-CPA symmetric encryption, and PRF HMAC. To mount a MITM attack, the adversary must modify at least one of these variables $(TID_{vi}, \Theta_1, \varphi_1, REV_{status}, T_1)$ or $(\Theta_2, \varphi_2, \beta_{v_i}^{new}, T_2)$. Modifying $\{T_i | i = 1, 2\}$ or REV_{status} results in instant rejection. When an adversary modifies Θ_1 and/or φ_1, the advantage of the adversary in breaking the security game is negligible since Θ_1, which is used as a variable in φ_1, is obtained from a truly random function. That is to say, Θ_1 is encrypted data by a fresh pairwise temporal key $PTK_{v\tau}$. Likewise, $\beta_{v_i}^{new}$ is encrypted by a freshly generated PUF-based key. Moreover, the symmetric encryption technique, Ω, adopted in the scheme is considered *IND-CPA* secure as defined in [33], [34] and the initialization vector (IV) of CBC is chosen at random. In addition, the seed of the *HMAC* is derived from a truly random function; therefore, modifying any variable of *HMAC* without knowing the seed will never be accepted by the other party. Even when $\mathcal{A}$ invokes the *Reveal* query and obtains the contents of V_i's OBU including the challenge C_{v_i}, $\mathcal{A}$ cannot predict the response R_{v_i} to compute the uniformly distributed key K_{v_i}. Consequently, our scheme is immune to the MITM attack. Finally, by adding up all previous lemmas' results, we have

$$Adv_{\Pi,\mathcal{A}}^{Sec}(k) \leq \frac{1}{2\ell n}\left(Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k) + Adv_{HMAC(.),\mathcal{B}'}^{PRF}(k)\right)$$

□

Theorem 2 (Privacy): Let *FE* be (d, λ, ϵ)-fuzzy extractor and $(d, n, l, \lambda, \epsilon)$-secure PUF. Assume *HMAC* be a secure pseudorandom function and the encryption scheme, Ω, be IND-CPA secure. Then our protocol satisfies the indistinguishability-based privacy property under memory leakage.

Proof: This proof is similar to the proof of Theorem 1. However, *note* in order for privacy to hold, it is important for our protocol to satisfy security first as shown in Theorem 1. This is due to the fact that if the security is broken, impersonation attacks, specifically $\mathcal{RT}$ impersonation, can be mounted successfully, leading to updating the secret key that is not derived by $\mathcal{RT}$. As a result, the $\mathcal{RT}$ can no longer accept this vehicle and then $\mathcal{A}$ can easily distinguish the challenge vehicle V_b^* in the privacy game. In view of the game transformation described in the proof of Theorem 1, it is clear that the whole transcripts will be identical to random variables inasmuch as we continuously modify the communication messages for the two vehicles V_0^* and V_1^*. Game 1 is modified here such that $\mathcal{C}$ guesses two vehicles, which will be selected by $\mathcal{A}$ in the privacy game. In this case, the probability of the random guess will be at least $1/n^2$. Upon proceeding, the game transformation described in Game 2 is applied to the sessions related to the vehicles V_0^* and V_1^*. Thus, the transmitting messages $(TID_i, \Theta_1, \varphi_1, REV_{status}, T_1)$ and $(\Theta_2, \varphi_2, \beta_{v_i}^{new}, T_2)$ are turned into arbitrary strings and no information will be revealed about the challenger's coin. Even though it is assumed that $\mathcal{A}$ is capable of extracting the secret keys from the vehicle's OBU in the privacy game, these secret credentials will not expose any information about the real identity of the device because they are updated from random sources. Therefore, no adversary can distinguish the challenge vehicle with a probability higher than $1/n^2$ and we get

$$Adv_{\Pi,\mathcal{A}}^{IND^*}(k) \leq Adv_{\Pi,\mathcal{A}'}^{Sec}(k) + \frac{1}{4\ell n^2}\left(Adv_{\Omega,\mathcal{B}}^{IND-CPA}(k) + Adv_{HMAC(.),\mathcal{B}'}^{PRF}(k)\right)$$

■

Theorem 3 (Collusion Attack): The proposed Secret Sharing-based revocation phase is a secure privacy-preserving key distribution mechanism with revocation capability and realizes unbounded collusion resistance capability even with compromising all personal secrets of all vehicles in $\mathcal{R}_j$.

Proof: For proof of Theorem 3 see Appendix A ■

B. Formal Security Verification Using AVISPA

In this section, the proposed protocol is evaluated for the formal security verification using the widely accepted Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. AVISPA comprises four back-ends and abstraction-based roles, which are specified through the High Level Protocol Specific Language (HLPSL). The detailed description and functionality of the four back-ends of AVISPA are available in [35], [36]. The simulation result of the proposed protocol is *SAFE* under the two backends OFMC and CL-AtSe as shown in Fig. 4. Therefore, our scheme is secure against

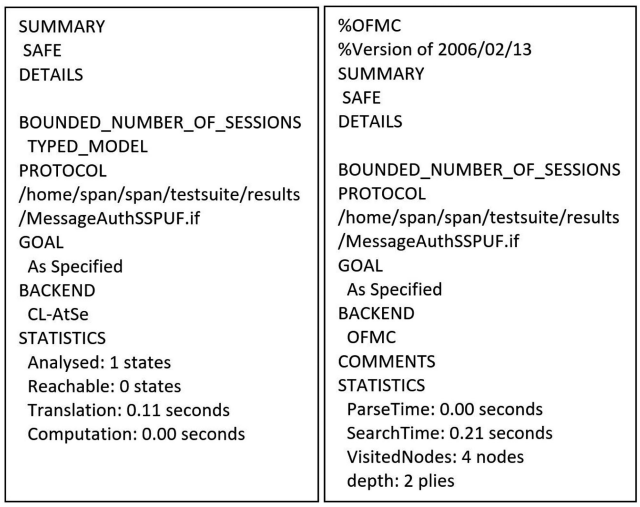

Fig. 4. Simulation results of AVISPA of the proposed scheme using the backends OFMC and CL-AtSe backends.

replay and MITM attacks. The full details of the implementation process of our protocol are provided in the supplementary materials.

C. Informal Security Analysis

In this subsection, we provide informal security analysis to show that our scheme is secure against well-known attacks.

1) *Resiliency against De-synchronization Attacks*. Suppose Msg_3, which contains secret keys updating for V_i, got lost during transmission or attacked by $\mathcal{A}$, V_i undoubtedly wouldn't update its secrets. However, $\mathcal{RT}$ can still recognize V_i in later sessions. That is because: 1) our mechanism requires only one side (which is V_i) to store the received updates and thus the attack cannot cause inconsistent shared data. 2) the parties usually recognize each other based on (PUF and $P(x, y)$)-based pairwise temporal keys which will always be valid as long as V_i is unrevoked. As a result, the proposed scheme is fully immune to this attack.
2) *Protection against Cloning and Physical Attacks*. Suppose an adversary $\mathcal{A}$ attempts to physically tamper with the V_i's OBU. However, any such attempt to tamper with PUF_i (i.e., PUF separation from its OBU) destroys and renders it useless. As a result, PUF_i cannot reproduce the desired response $R_i = PUF_i(C_i)$. Hence $p(v_i, y)$ cannot be reconstructed successfully, resulting in an inability to establish pairwise keys with the other parties. Additionally, PUFs are safeguarded against cloning and can not be recreated [23]. Consequently, the proposed scheme is resilient against cloning and physical attacks.
3) *Protection against Impersonation Attacks*. Suppose $\mathcal{A}$ intercepts Msg_1 during the authentication phase. $\mathcal{A}$ attempts to create a valid authentication response Msg_2, say $Msg_2^{\#} = \{TID_{v_i}^{\#}, \Theta_{v_i}^{\#}, REV_{status}^{\#}, T_1^{\#}\}$, on behalf of V_i using its current timestamp $T_1^{\#}$. Since $\mathcal{A}$ doesn't know any of these v_i, $h(K_{TA})$, and $p(v_i, y)$, s/he can never establish valid $PTK_{vu}/PTK_{v\tau}$ with $\mathcal{RT}$ and thus cannot compose valid Msg_2. Therefore, $\mathcal{RT}$ will reject this response. Similarly, if $\mathcal{A}$ tries to impersonate *GWN* to deceive V_i, s/he cannot compose valid Msg_3 due to not knowing of $PTK_{v\tau}$. As a result, the proposed scheme is immune to vehicle/RSU impersonation attacks. Moreover, due to employing PUF in the proposed scheme, $\mathcal{A}$ cannot succeed in mounting impersonation attacks even under the aforementioned physical attacks.

TABLE IV
SECURITY FEATURES COMPARISON

Security feature	[5]	[12]	[37]	Our
Secure against replay attack	✓	✓	✓	✓
Immune to Man-in-the-Middle attacks	✓	✗	✓	✓
Secure against impersonation attacks	✓	✓	✓	✓
Protection against physical attacks	✗	✗	✗	✓
Resiliency against de-synchronization	✓	✓	✓	✓
Protection against cloning attacks	✗	✗	✗	✓
Immune to plaintext BSMs-based traceability	✗	✗	✗	✓
Polynomial-based encrypted broadcast	✗	✗	✗	✓

TABLE V
EXECUTATION TIME OF CRYPTOGRAPHIC OPERATIONS [39]

Cryptographic Operation	Notation	IoT Device	Server
Multiplication point	T_{mp}	$5.9ms$	$2.6ms$
Multiplication	T_m	$22.93ms$	$14.5ms$
Bilinear pairing	T_b	$9.23ms$	$3.78ms$
Certificate generation	$T_{cert_{gen}}$	$57.63ms$	–
Certificate verification	$T_{cert_{ver}}$	–	$17.24ms$
Modular exponential	T_e	$7.86ms$	$2.34ms$
Symmetric en(de)cryption	T_s	$0.079ms$	$0.041ms$
One-way hash	T_h	$0.026ms$	$0.011ms$
PUF (128-bit Arbiter)	T_p	$0.12ms$	–
Fuzzy extractor generation	$T_{f.g}$	–	$1.17ms$
Fuzzy extractor reproduction	$T_{f.r}$	$3.28ms$	–
Message authentication code	T_{mac}	$2.9ms$	$1.23ms$

VI. PERFORMANCE EVALUATION

In this section, we conduct a comparative performance analysis of the proposed scheme against the related schemes. Table IV shows security and functionality features comparisons with the related schemes [5], [12], [37]. From this table, it is obvious that still none of the compared schemes are completely free from security flaws, while the proposed scheme achieves all these features simultaneously. As also shown in Table IV, traceability, physical, and cloning attacks are still successful when applying to all compared schemes.

Prior to showing the efficiency of our proposed protocol in terms of computation and communication overhead, it is worth mentioning that establishing PUF-based PTKs is computationally lightweight due to that it is only evaluation of $p(v_i, y)$, which is one-degree polynomial, and adding K_j to its result. In addition, the execution time of recovering K_j from PUF is not computationally expensive and still lightweight.

A. Comparison With Existing Authentication Protocols for Vehicular Communications

1) Computation Cost Analysis: The estimated execution time values of the cryptographic operations used in the computation comparison are listed in Table V. In the proposed scheme, the 128-bit Arbiter PUF is considered as the PUF

TABLE VI
COMPUTATION COST COMPARISON

Communication Type	Scheme	Vehicle	RSU($\mathcal{RT}$)	Cloud server	Total Cost
V2I	[12]	$3T_{bp} + 5T_h + T_{mp}$	$T_{bp} + 2T_h + T_{mp}$	$3T_h + T_{mp}$	$\approx 42.8ms$
	[37]	$4T_h + 3T_{mp}$	$4T_h + 4T_{mp}$	$11T_h + 10T_{mp}$	$\approx 246.8ms$
	Our	$T_{Ret} + 5T_h + T_p + T_s + 2T_{mac}$	$T_{Ret} + 3T_h + T_{f.g} + 2T_s + 2T_{mac}$	NA	$\approx 16.84ms$
V2V	[9]	$6T_h + 2T_{bp} + 4T_{mp} + T_e + 2T_{ad}$	NA	NA	$\approx 50.1ms$
	[19]	$4T_{ecc} + 2T_{cert_{gen}} + 2T_{cert_{ver}}$	NA	NA	$\approx 152.9ms$
	Our	$2T_{Ret} + 6T_h + 2T_{mac} + 2T_s$	NA	NA	$\approx 13.1ms$
Broadcast	[5]	$T_{mac} + 2(2T_{cert_{ver}})$	NA	NA	$\approx 71.9ms$
	Our	$T_{Ret} + T_h + 3T_e + 2T_{mac} + T_s$	NA	NA	$\approx 32.8ms$

embedded in the vehicles and the code offset mechanism using BCH is adopted [38]. Table VI shows computation comparisons of our proposed scheme with various related schemes based on different approaches, i.e., Vehicle-to-Infrastructure (V2I) communication, Vehicle-to-Vehicle (V2V) instant message communication, and broadcast transmission. Considering the V2I approach, the proposed scheme is compared with [12], [37]. It's shown in Table VI that the total computation of the proposed scheme requires $2T_{Ret} + 8T_h + T_p + 3T_s + T_{f.g} + 4T_{mac} \approx 16.84ms$, where T_{Ret} is the computation cost of the *Retrieval* algorithm (see Algorithm 1). On the other hand, the overall computational cost required in schemes [12] and [37] are $42.8ms$ and $246.8ms$, respectively. Additionally, our scheme is compared with schemes [9] and [19] with respect to the V2V communication-based immediate transmission. Whereas our scheme only requires $2T_{Ret} + 6T_h + 2T_{mac} + 2T_s \approx 13.1ms$ for sending an expeditious BSMs (i.e., accident-related) to an adjacent particular vehicle, schemes [9] and [19] need $6T_h + 2T_{bp} + 4T_{mp} + T_e + 2T_{ad} \approx 50.1ms$ and $4T_{ecc} + 2T_{cert_{gen}} + 2T_{cert_{ver}} \approx 152.9ms$, respectively. On the basis of the broadcast message transmission, we further compared our protocol with the protocol in [5]. The computation cost required to verify a broadcast received message in our scheme is $T_{Ret} + T_h + 3T_e + 2T_{mac} + T_s \approx 32.8ms$, whereas the verification process of the scheme [5] requires $T_{mac} + 2(2T_{cert_{ver}}) \approx 71.9ms$. Consequently, it is obvious from Table VI that the proposed scheme outperforms all compared schemes with respect to the three comparative approaches since it realizes the lowest computation cost as compared to the other schemes. Moreover, it is worth mentioning that a traffic-related message, say BSM, is also confidential in our scheme besides having authenticity and integrity which is not provided in the compared schemes.

Fig. 5 illustrates the computation cost needed by an RSU in V2I communication to process only one BSM received from multiple vehicles (i.e., up to 40 vehicles) in RSU's coverage area. The computation cost at RSU side to process only one BSM received from only one vehicle in schemes [12] and [37] are $T_{bp} + 5T_h + 2T_{mp} = 3.78 + 5 \times 0.011 + 2 \times 14.5 \cong .32.84ms$ and $15T_h + 14T_{mp} = 15 \times 0.011 + 14 \times 14.5 \cong .203.2ms$, respectively. On the other hand, in our V2I approach, the computation cost at the RSU side is $\approx 12.3ms$. Therefore, it is clear from Fig. 5 that an RSU in schemes [12] and [37] requires much time to process *only one message* for a number of vehicles (e.g., ranging from 1 to 40 vehicles) in its coverage. For example, to process one message for 40 vehicles, an RSU in [12] and [37] consumes computation

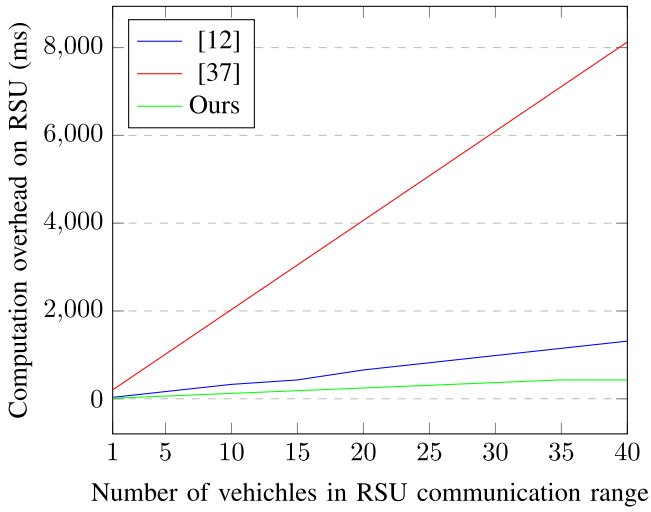

Fig. 5. Overall Communication Overhead on RSU.

overhead of 1.32 and 8.13 seconds, respectively. In contrast, our protocol requires only less than 0.5 s.

2) Communication Cost and Storage Overhead Analysis: In this subsection, the communication and storage costs of our scheme are compared with the related schemes based on different approaches in a similar manner to that described in Section VI-A1. Let the length of values in $\mathbb{F}_q$, $\mathbb{F}_p^*$, and the cyclic additive group $\mathbb{G}$ whose points on the elliptic curve be donated as $|\mathbb{F}_q|$, $|\mathbb{F}_p^*|$, and $|\mathbb{G}|$, respectively. We consider $|\mathbb{F}_q|$, $|\mathbb{F}_p^*|$, and $|\mathbb{G}|$ are 160 bits, 512 bits, and 1024 bits, respectively. We further assume that plaintext/ciphertext block symmetric encryption/decryption (using AES-CBC algorithm), hash digests (SHA-1), nonces, and timestamps are 128 bits, 160 bits, 128 bits, and 32 bits in length, respectively. According to [40] the normal sizes of signature (generation/verification), certificate, and message (traffic-related) are 344 bits, 504 bits, and 800 bits, respectively. Helper data hd length size is considered 1264 bits as described in [41]. Table VII presents the communication costs of our proposed scheme compared with different related schemes. Considering the V2I communication, our protocol needs to send the following three messages: $Msg_1 = N_1$, $Msg_2 = (TID_{v_i} + \Theta_1 + \varphi_1 + REV_{status} + T_1)$, and $Msg_3 = (\Theta_2 + \varphi_2 + \beta_{v_i}^{new} + T_2)$, which require 128 bits, $160 + \lceil (160 + 128 + 128)/128 \rceil \times 128 + 160 + 160 + 32 \approx 896$ bits, and $(\lceil (1264 + 128 + 160 + 160)/128 \rceil \times 128 + 160 + \lceil (160 + 160 + 160)/128 \rceil) \times 128 + 32 \approx 2336$ bits, respectively. Thus, the total transmission incurred to send the aforementioned messages is $128 + 896 + 2336 = 3360$

TABLE VII
COMMUNICATION AND STORAGE OVERHEAD COMPARISON

Communication Type	Scheme	Communication overhead [bits]	Storage overhead (bits) [bits]
V2I	[12]	$9\|\mathbb{G}\| + 13\|\mathbb{F}_q\| + 5\|T\| \approx 11040$	$AID_i + R_i + Z_i + IM_i + sIM_i \approx 1408$
	[37]	$10\|\mathbb{G}\| + 14\|\mathbb{F}_q\| + 4\|T\| \approx 12192$	$ID_{U_i} + D_{ID_i} \approx 1152$
	Our	$5\|\mathbb{F}_q\| + \|\Theta_{1,2}\| + 2\|T\| + \|\beta_{v_i}^{new}\| \approx 3360$	$q(v_i, y) + \beta_{v_i} + C_{v_i} + hd_{v_i} + \sigma_{v_i} + \xi_{v_i} + MK_j \approx 2640^*$
V2V	[9]	$3\|\mathbb{F}_q\| + \|\mathbb{F}_p^*\| + \|T\| \approx 1024$	
	[19]	$Msg_{BA} + Msg_{AB} \approx 1008$	
	Our	$3\|\mathbb{F}_q\| + \|T\| \approx 512$	Same as $*$
Broadcast	[5]	$\|\mathbb{F}_q\| + cert_u + sig_u + \|T\| \approx 1608$	$\ell \times (PID_u + Cert_u + RS_u + RP_u) + K_g + P_\circ \approx \ell \times (820) + 288$
	Our	$5\|\mathbb{F}_q\| + \|T\| \approx 832$	Same as $*$

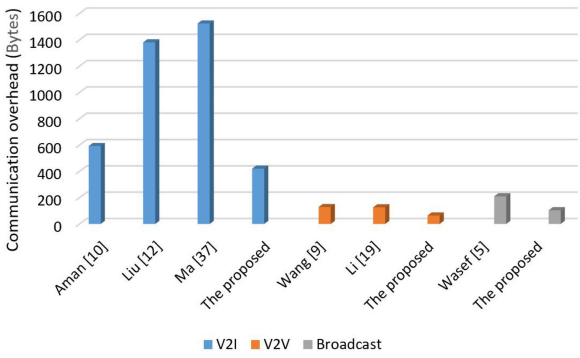

Fig. 6. Overall Communication Overhead.

bits. In contrast, the total transmission costs required for the schemes [12] and [37] are $9|\mathbb{G}| + 13|\mathbb{F}_q| + 5|T| \approx 11040$ bits and $10|\mathbb{G}| + 14|\mathbb{F}_q| + 4|T| \approx 12192$ bits, respectively, which obviously burden much communication overhead when compared to our scheme. In view of the communication cost incurred in the V2V instant message transmission, both compared schemes require much computation overhead for instance scheme [9] incurs $|\mathbb{F}_q| + \mathbb{F}_p^*| + |T| \approx 1024$ bits, and scheme [19] requires ≈ 1008 bits, whereas only $3|\mathbb{F}_q| + |T| \approx 512$ bits are incurred by our protocol. Lastly, a signed broadcast message in [5] requires $|\mathbb{F}_q| + |cert_u| + |sig_u| + |T| \approx 1608$ bits, whereas our scheme only requires $5|\mathbb{F}_q| + |T| \approx 832$ bits. Fig. 6 illustrates the communication overhead in our protocol compared with the related work in the case of the three different communication approaches: V2I, V2V, and broadcast. It is clear from Fig. 6 that our scheme achieved the lowest communication overhead when compared to the other schemes in the three vehicular communication approaches. Therefore, the communication overhead incurred by our protocol is far better and more efficient than the compared schemes and thus it's more suitable for vehicular communication.

For storage overhead comparisons shown in Table VII, a vehicle V_i in our protocol needs to store $(q(v_i, y), \beta_{v_i}, C_{v_i}, hd_{v_i}, \sigma_{v_i}, \xi_{v_i}, MK_i, sid_j) = 2\times 160 + \lceil(160 + 160 + 160)/128\rceil \times 128 + 128 + 1264 + 160 + 160 + 160 + 128 = 2640$ bits. On the other hand, the protocols proposed in [12] and [37] require total storage $AID_i + R_i + Z_i + IM_i + sIM_i \approx 1408$ bits, $ID_{U_i} + D_{ID_i} \approx 1152$ bits, respectively. Although these protocols require less storage than our protocol, they lack critical security features as described in Table IV and they burden much communication overhead as shown in Table VI. Furthermore, our proposed protocol achieves far less storage cost than that is incurred in [5], which requires each vehicle to store a set ℓ of each of the following variables (except their last two) and thus the storage overhead on V_i is $\ell \times (PID_u + Cert_u + RS_u + RP_u) + K_g + P_\circ = \ell \times (820) + 288$ bits. Hence the proposed protocol obviously outperforms the existing authentication protocols proposed for vehicular communications in terms of computation, communication, and storage cost. As a result, it is more feasible to VANETs as compared to the other schemes.

B. Comparison With Existing PUF Based Authentication Protocols

In this section, the proposed protocol is compared with some of the existing PUF-based authentication protocols [10], [23], [24] in terms of computation, communication, and security features. Table VIII shows that our protocol incurs far less communication overhead than the compared schemes. Therefore, the proposed work has efficient computation cost as opposed to the compared PUF-based protocols. We assumed using fuzzy extractor to extract uniformly distributed keys from PUF output in the compared schemes to have fair comparisons. Even though the compared protocols in Table VIII have more efficient computation cost than our protocol, it is obvious that our protocol achieves better security and privacy as none of the compared protocols realized the important security features of VANET: *SF2*, *SF3*, *SF4*, and *SF5*.

C. Computation Complexity of Revocation Checking Process

Let N_{rev} denote the total number of all revoked certificates in a CRL. Overall, performing a revocation status checking for a vehicle using the linear or binary search algorithm requires computation complexity $\mathcal{O}(N_{rev})$ or $\mathcal{O}(\log N_{rev})$, respectively [42]. In contrast, the revocation checking process in our scheme requires only one comparison between the calculated and received value of *REVstatus*, $REV_{status} = h(K_j\|v_i\|T)$ as it is independent of the number of revoked certificates. Consequently, the computation complexity of our proposed revocation mechanism is $\mathcal{O}(1)$ similar to that in the scheme [5]. However, our revocation is superior to that in [5] due to the following reasons: 1) the number of revocation sessions, performed by TA, is unlimited in our scheme while it is limited in [5] and also bounded by a hash chain value which is continuously used up and thus a mechanism to replace the current hash value with a new one is essential. 2) Once a malicious vehicle is discovered in [5],

TABLE VIII
COMPLEXITY COMPARISONS WITH PUF BASED AUTHENTICATION PROTOCOLS

Scheme	Communication Overhead [bits]	Computation Overhead [ms]	Security Features				
			SF1	*SF2*	*SF3*	*SF4*	*SF5*
[10]	$\|M_{RG}\|+\|M_{GS}\|+\|M_{SG}\|+\|M_{GR}\|+\|M_{SA1}\|+4\|h(.)\|+5\|ID\| \approx 4736$	$2T_p+(2T_{f.r})+9T_s+7T_h \approx 7.1ms$	Yes	No	No	No	No
[23]	$\|M_1\|+\|M_2\|+\|M_3\| \approx 2224$	$2T_p+T_{f.g}+T_{f.r}+10T_h \approx 6.5ms$	Yes	No	No	No	No
[24]	$\|M_1\|+\|M_2\|\|M_3\|+\|M_4\|+\|M_5\| \approx 1792$	$4T_p+(2T_{f.r})+13T_h \approx 7.5ms$	Yes	No	No	No	No
Ours	$3\|\mathbb{F}_q\|+\|T\| \approx 512$	$2T_{Ret}+6T_h+2T_{mac}+2T_s \approx 13.1ms$	Yes	Yes	Yes	Yes	Yes

SF1:Secure against physical attacks; *SF2*: Broadcast encryption; *SF3*: Message confidentiality (encrypted BSMs); *SF4*:Expeditious authenticated message transmission without third party involvement; *SF5*:Traceability attacks based on linking plaintext (BSMs) messages.

the global key update process begins, which is another form of CRL and is hard to execute [3]. 3) Unlike the scheme [5], our revocation can even be temporary, namely, TA can permit a previously revoked vehicle to rejoin the system again. Such a rejoining is possible as none of the vehicle's secrets was exposed when revoked formerly.

VII. CONCLUSION AND FUTURE WORK

In this paper, we proposed a message authentication protocol for vehicular communication (i.e., VANET) in smart cities based on PUF and Secret Sharing. First, the proposed protocol enables entities to utilize their PUF to establish pairwise keys (PTKs) with other entities for mutual authentication and encrypting BSMs as well. Second, our protocol employs Secret Sharing-based SGKD for the revocation mechanism in which the time complexity $\mathcal{O}(1)$. Third, a vehicle in our scheme expedites sending unicast and broadcast *polynomial-based encrypted BSMs* that only unrevoked vehicle(s) can decrypt. Therefore, BSMs linking-based traceability attacks can be thwarted and a higher level of vehicle privacy can be realized. Fourth, the proposed scheme guarantees provable security and privacy even under memory leakage. Overall, compared to the existing schemes, our scheme offers more security features, better computation and communication efficiency, a higher privacy level, and more suitability for vehicle communication.

A proposed direction for the future work based on the limitations of this work would be as below: In our V2V communications, a vehicle V_i with an identifier v_i must obtain the identifier v_j of a vehicle V_j so that they can both establish PTKs together. Although a vehicle uses a virtual identifier/identity TID each time it communicates with others, the vehicle originally owns only one identifier and even it is protected by a PUF-based key to preserve the vehicle's privacy, there is still a slim chance to disclose when a communicating party becomes malicious. In addition, the PUF is used only to provide protection against physical and cloning attacks, however, PUF is not used to generate vehicles' (pseudonym) identities. Therefore, in our future work, we aim to employ pseudonymous identities approach based on PUF. Furthermore, we will utilize Public Physical Unclonable Function (PPUF) to act as a public key for an entity.

APPENDIX A
PROOF OF THE THEOREM 3

Proof: Suppose a coalition of all vehicles in $\mathcal{R}_j$, which are revoked before and in the *j*-th session. In the proposed Secret Sharing-based SGKD, since each $\{RSU_l|_{l=1,2,\ldots,m}\}$ constructs its own *t*-th degree polynomial $\phi_j^i(x)$ (whose constant term is K_j) as described in Section IV-H, then the coalition of $\mathcal{R}_j$ has at most *t* public points on the $\phi_j^l(x)$ if they occurred to exist in the same RSU_l's coverage area and even fewer points if they are in different coverage areas. Hence the coalition $\mathcal{R}_j$ cannot interpolate any of $\{\phi_j^l(x)|_{l=1,2,\ldots,m}\}$. Thus, K_j is entirely safe and this security feature is computationally secure under memory leakage. Even with compromising all secrets from OBUs of all nodes in $\mathcal{R}_j$, the coalition of $\mathcal{R}_j$ still cannot obtain K_j. This is due to the masking protection provided for an offset session key K_j by a PUF-based key K_{v_i}, $MK_j = K_{v_i} \oplus K_j$. ■

REFERENCES

[1] A. Dua, N. Kumar, A. K. Das, and W. Susilo, "Secure message communication protocol among vehicles in smart city," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4359–4373, May 2018.

[2] J. Camenisch, M. Drijvers, A. Lehmann, G. Neven, and P. Towa, "Zone encryption with anonymous authentication for V2V communication," in *Proc. IEEE Symp. Secur. Privacy*, pp. 405–424, 2020.

[3] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular Ad Hoc networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 907–919, Feb. 2014.

[4] B. Ying and A. Nayak, "Anonymous and lightweight authentication for secure vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10 626–10 636, Dec. 2017.

[5] A. Wasef and X. Shen, "Emap: Expedite message authentication protocol for vehicular Ad Hoc networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 1, pp. 78–89, Jan. 2013.

[6] J. Cui, D. Wu, J. Zhang, Y. Xu, and H. Zhong, "An efficient authentication scheme based on semi-trusted authority in vanets," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2972–2986, Mar. 2019.

[7] P. Vijayakumar, V. Chang, L. J. Deborah, B. Balusamy, and P. Shynu, "Computationally efficient privacy preserving anonymous mutual and batch authentication schemes for vehicular Ad Hoc networks," *Future Gener. Comput. Syst.*, vol. 78, pp. 943–955, 2018.

[8] X. Cheng, L. Yang, and X. Shen, "D2D for intelligent transportation systems: A feasibility study," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1784–1793, Aug. 2015.

[9] C. Wang, L. Xiao, J. Shen, and R. Huang, "Neighborhood trustworthiness-based vehicle-to-vehicle authentication scheme for vehicular Ad Hoc networks," *Concurrency Comput.: Pract. Exper.*, vol. 31, no. 21, 2019, Art. no. e4643.

[10] M. N. Aman, U. Javaid, and B. Sikdar, "A privacy-preserving and scalable authentication protocol for the internet of vehicles," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1123–1139, Jan. 2021, doi: 10.1109/JIOT.2020.3010893.

[11] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[12] Y. Liu, Y. Wang, and G. Chang, "Efficient privacy-preserving dual authentication and key agreement scheme for secure V2V communications in an IoV paradigm," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2740–2749, Oct. 2017.

[13] V. Sureshkumar, S. Anandhi, R. Madhumathi, and N. Selvarajan, "Light weight authentication and key establishment protocol for smart vehicles communication in smart city," in *Proc. Int. Conf. Smart City Informatization*, Springer, 2019, pp. 349–362.

[14] L. Dang *et al.*, "Efficient identity-based authenticated key agreement protocol with provable security for vehicular ad hoc networks," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 4, 2018, Art. no. 1550147718772545.
[15] J. Liu, Q. Li, R. Sun, X. Du, and M. Guizani, "An efficient anonymous authentication scheme for internet of vehicles," in *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–6.
[16] H. Vasudev, V. Deshpande, D. Das, and S. K. Das, "A lightweight mutual authentication protocol for V2V communication in internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 6709–6717, Jun. 2020, doi: 10.1109/TVT.2020.2986585.
[17] S. Wang and N. Yao, "Liap: A local identity-based anonymous message authentication protocol in vanets," *Comput. Commun.*, vol. 112, pp. 154–164, 2017.
[18] I. Ali and F. Li, "An efficient conditional privacy-preserving authentication scheme for vehicle-to-infrastructure communication in vanets," *Veh. Commun.*, vol. 22, 2020, Art. no. 100228.
[19] L. Teng *et al.*, "Lightweight security authentication mechanism towards UAV networks," in *Proc. Int. Conf. Netw. Netw. Appl.*, 2019, pp. 379–384.
[20] Q. Jiang, N. Zhang, J. Ni, J. Ma, X. Ma, and K.-K. R. Choo, "Unified biometric privacy preserving three-factor authentication and key agreement for cloud-assisted autonomous vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 9390–9401, Sep. 2020.
[21] U. Chatterjee *et al.*, "Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database," *IEEE Trans. Dependable Secure Comput.*, vol. 16, no. 3, pp. 424–437, May/Jun. 2018.
[22] Q. Jiang, X. Zhang, N. Zhang, Y. Tian, X. Ma, and J. Ma, "Two-factor authentication protocol using physical unclonable function for IoV," in *Proc. IEEE/CIC Int. Conf. Commun. China*, 2019, pp. 195–200.
[23] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for IoT devices," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 580–589, Feb. 2019.
[24] P. Gope and B. Sikdar, "An efficient privacy-preserving authenticated key agreement scheme for edge-assisted internet of drones," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13 621–13 630, Nov. 2020.
[25] P. Gope, O. Millwood, and N. Saxena, "A provably secure authentication scheme for RFID-enabled UAV applications," *Comput. Commun.*, vol. 166, pp. 19–25, 2021.
[26] P. Gope, B. Sikdar, and O. Millwood, "A scalable protocol level approach to prevent machine learning attacks on PUF-based authentication mechanisms for internet-of-medical-things," *IEEE Trans. Ind. Inform.*, vol. 18, no. 3, pp. 1971–1980, Mar. 2022.
[27] M. Ebrahimabadi, M. Younis, and N. Karimi, "A PUF-based modeling-attack resilient authentication protocol for IoT devices," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2021.3098496.
[28] U. Chatterjee, R. S. Chakraborty, and D. Mukhopadhyay, "A PUF-based secure communication protocol for IoT," *ACM Trans. Embedded Comput. Syst.*, vol. 16, no. 3, pp. 1–25, 2017.
[29] M. N. Aman, M. H. Basheer, and B. Sikdar, "Two-factor authentication for IoT with location information," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3335–3351, Apr. 2019.
[30] D. Moriyama, S. Matsuo, and M. Yung, "PUF-based RFID authentication secure and private under memory leakage," *IACR Cryptol. ePrint Arch*, vol. 3, pp. 61–83, 2013.
[31] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 2004, pp. 523–540.
[32] G. R. Blakley *et al.*, "Safeguarding cryptographic keys," in *Proc. Nat. Comput. Conf.*, 1979, pp. 313–313.
[33] S. Wu and K. Chen, "An efficient key-management scheme for hierarchical access control in e-medicine system," *J. Med. Syst.*, vol. 36, no. 4, pp. 2325–2337, 2012.
[34] K.-K. R. Choo, "Key establishment: Proofs and refutations," Ph.D. dissertation, Queensland University of Technology, 2006.
[35] Automated Validation of Internet Security Protocols and Applications (AVISPA), Automated validation of internet security protocols and applications. [Online]. Available: http://www.avispa-project.org
[36] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, "Secure biometric-based authentication scheme using chebyshev chaotic map for multi-server environment," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 824–839, Sep./Oct. 2016.
[37] M. Ma, D. He, H. Wang, N. Kumar, and K.-K. R. Choo, "An efficient and provably secure authenticated key agreement protocol for fog-based vehicular ad-hoc networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8065–8075, Oct. 2019.
[38] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, 2008.
[39] P. Gope and B. Sikdar, "Privacy-aware authenticated key agreement scheme for secure smart grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3953–3962, Jul. 2019, doi: 10.1109/TSG.2018.2844403.
[40] H. Vasudev and D. Das, "An efficient authentication and secure vehicle-to-vehicle communications in an IoV," in *Proc. IEEE 89th Veh. Technol. Conf.*, 2019, pp. 1–5.
[41] A. Aysu, E. Gulcan, D. Moriyama, P. Schaumont, and M. Yung, "End-to-end design of a PUF-based privacy preserving authentication protocol," in *Int. Workshop Cryptographic Hardware Embedded Syst.*, Springer, 2015, pp. 556–576.
[42] T. H. Cormen *et al.*, Introduction to Algorithms. 2001.

Wajdy Othman received the master's degree from the School of Computer Science and Technology, Hunan University, China, in 2017. He is currently working toward the Ph.D. degree with the School of Computer Science and Technology, University of Science and Technology of China (USTC). His current research interests include network security, cryptography, authentication, key management, access control, and security and privacy in IoT.

Miao Fuyou received the M.S. degree in computer science and technology from the China University of Mining Technology, Beijing, China, in 1999, and the Ph.D. degree in computer science from the University of Science and Technology of China, China, in 2005. He is currently an Associate Professor with the School of Computer Science and Technology, University of Science and Technology of China, China. His research interests include applied cryptography, network security, and mobile computing.

Kaiping Xue (Senior Member, IEEE) received the Ph.D. degree from the Department of Electronic Engineering and Information Science (EEIS), USTC, in 2007. From May 2012 to May 2013, he was a Postdoctoral Researcher with the Department of Electrical and Computer Engineering, University of Florida. He is currently a Professor with the Department of Information Security and Department of EEIS, USTC. His research interests include next-generation Internet, distributed networks and network security. He is on the Editorial Board of several journals, including IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (TWC), IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT (TNSM), *Ad Hoc Networks*, IEEE ACCESS, and *China Communications*. He was a Guest Editor of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC) and a Lead Guest Editor of *IEEE Communications Magazine*. He is an IET Fellow.

Ammar Hawbani (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer software and theory from the University of Science and Technology of China (USTC), Hefei, China, in 2009, 2012 and 2016, respectively. He is an Associate Professor of networking and communication algorithms with the School of Computer Science and Technology, the University of Science and Technology of China, China. From 2016 to 2019, he worked as Postdoctoral Researcher with the School of Computer Science and Technology at USTC. His research interests include Internet of things, WSNs, WBANs, WMNs, VANETs, and SDN.