This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

LBRAKA: Lattice-Based Robust Authenticated Key Agreement for VANETs

Gao Liu, Weiyang Li, Chengsheng Yuan, Ning Wang, Chuan Ma, Nankun Mu, Zhiquan Liu, Yining Liu, *Senior Member, IEEE*, Tao Xiang, *Senior Member, IEEE*

***Abstract*—Authenticated key agreement between vehicles and roadside units (RSUs) is crucial for securing vehicular ad-hoc networks (VANETs). However, most of relative works cannot withstand quantum attacks due to the adoption of conventional cryptographic algorithms, such as those based on discrete logarithm and large integer factorization problems. Although some solutions generally employ lattice problems to resist against quantum attacks, they lack a privacy preserving mechanism to provide the unlinkability of vehicle public keys or certificates, as well as robust conditional traceability. In this study, we propose a lattice-based robust authenticated key agreement (LBRAKA) scheme for VANETs, which facilitates anonymous authentication and key negotiation between vehicles and RSUs and employs the ring learning with errors problem to withstand quantum attacks. Specifically, obfuscated expiration times are allocated to vehicle public keys and certificates for achieving the unlinkability of public keys. Vehicles and a central authority are required to create commitments to the vehicle's real identity and public key, in order to support robust conditional traceability. Security analysis demonstrates that LBRAKA not only ensures anonymity, conditional privacy, unlinkability, key escrow freedom, public verification of traceability, and tracing robustness, but also effectively thwarts most known attacks. Comparative experimental results show the promising usability of LBRAKA.**

***Index Terms*—VANETs, lattice-based authenticated key agreement, robustness, unlinkability.**

I. Introduction

Copyright (c) 20xx IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This work was supported by the National Key R&D Program of China under Grant 2022YFB3103500, the National Natural Science Foundation of China under Grants 62302070, 62101079, 62272073, 62402202 and 62202071, and the Fundamental Research Funds for the Central Universities under Grant 2023CDJXY-039, the Natural Science Foundation of Chongqing, China, under Grants cstc2021jcyj-msxm0465 and cstc2021jcyj-msxmX0273, the Venture and Innovation Support Program for Chongqing Overseas Returnees under Grant cx2021012, the China Postdoctoral Science Foundation under Grants 2023M740399, 2022M710520 and 2022M710518, the Chengdu Science and Technology Program, Sichuan, China, under Grant 2023-YF11-00020-HZ, and the Joint Fund of Ministry of Education of China for Equipment Pre-Research under Grant 8091B032127. (Corresponding author: Ning Wang)

Gao Liu, Weiyang Li, Ning Wang, Chuan Ma, Nankun Mu, and Tao Xiang are with the College of Computer Science, Chongqing University, Chongqing, 400044 China (Email: {gaoliu, nwang5, chuan.ma, nankun.mu, txiang}@cqu.edu.cn, weiyangli@stu.cqu.edu.cn).

Chengsheng Yuan is with the School of Computer Science, Nanjing University of Information Science and Technology, Nanjing, 210044 China (Email: yuancs@nuist.edu.cn).

Zhiquan Liu is with the College of Cyber Security, Jinan University, Guangzhou, 510632 China (Email: zqliu@vip.qq.com).

Yining Liu is with the School of Data Science and Artificial Intelligence, Wenzhou University of Technology, Wenzhou, 325000 China (Email: lyn7311@sina.com).

VEHICULAR ad-hoc networks (VANETs) have moved from pilot demonstrations to commercial applications, and will become an indispensable part of people's daily lives. In VANETs, vehicles equipped with on-board units (OBUs) can communicate wirelessly with roadside units (RSUs) directly via the dedicated short range communication (DSRC) protocol, which is defined as vehicle-to-infrastructure (V2I) communication. During V2I communication, RSUs collect traffic information from vehicles and forward it to a traffic center, which allows the center to make real-time decisions and improve traffic conditions. Additionally, vehicles obtain traffic information from RSUs, e.g., traffic conditions, accidents, and road speed limits. Therefore, V2I can enhance driving safety, navigation efficiency, and traffic flow management.

Although VANETs offer many conveniences, they still face several challenges related to security and privacy [1]–[3]. (i) In terms of communication security challenges, communications between vehicles and RSUs are susceptible to eavesdropping and tampering attacks. Moreover, quantum attacks can break traditional cryptographic algorithms (e.g., the algorithm based on discrete logarithm and large integer factorization problems). Therefore, how to achieve authentication and confidentiality in communications that are also resistant to quantum attacks is an important and unresolved issue [4], [5]. (ii) With respect to privacy challenges, it is necessary to ensure the unlinkability of vehicle public keys or pseudonyms [3]. When vehicles violate rules or engage in malicious behaviors, it is essential to identify and publicly disclose their real identities (i.e., support conditional privacy). Additionally, when tracing malicious vehicles, it is crucial to ensure that the tracing is publicly verifiable and robust to prevent malicious tracers.

Although there exist many authentication schemes [6]–[8] for VANETs, the Internet of Vehicles (IoV), Internet of Things (IoT), they generally rely on traditional cryptographic algorithms, such as those based on discrete logarithm and large integer factorization problems. These algorithms are vulnerable to quantum attacks, as attackers with quantum computers can exploit Grover and Shor algorithms [9] to break them. There are some quantum-resistant authenticated key agreement (AKA) solutions, which primarily adopt lattice problems to withstand quantum attacks, such as learning with errors (LWE) and ring learning with errors (RLWE) [10]–[13]. However, most of them are unable to simultaneously address the aforementioned security and privacy challenges. Although most existing quantum-resistant AKA schemes have addressed communication security issues (i.e., eavesdropping and tampering attacks), they still only provide basic support

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

TABLE I
COMPARISON OF LBRAKA WITH RELATED WORKS

	An	Un	CP	PVT	TR	KEF	RI	RMM	RR	RKS	PI	FA
[10]	●	○	○	○	○	●	●	●	●	●	●	●
[11]	●	○	○	○	○	●	●	●	●	●	●	●
[12]	○	○	○	○	○	○	●	●	●	●	●	●
[13]	●	○	○	○	○	○	●	●	●	●	●	●
[14]	●	○	○	○	○	●	●	●	●	●	●	●
[15]	●	○	○	○	○	●	●	●	●	●	●	●
[16], [17]	●	○	○	○	○	●	●	●	●	●	●	●
[18]	●	○	○	○	○	●	●	●	●	●	●	●
[19]	●	○	○	○	○	●	●	●	●	●	●	●
[20]	●	○	○	○	○	●	●	●	●	●	●	●
[21]	●	○	○	○	○	●	●	●	●	●	●	●
[22]	●	○	○	○	○	●	●	●	●	●	●	●
[23]	●	●	●	○	○	○	●	●	●	●	●	●
[24]	○	○	○	○	○	○	●	●	●	●	●	●
[25]	○	○	○	○	○	○	●	●	●	●	●	●
Ours	●	●	●	●	●	●	●	●	●	●	●	●

An: anonymity; Un: unlinkability; CP: conditional privacy; PVT: public verification of tracing; TR: Tracing robustness; KEF: key escrow freeness; RI: resistance against impersonation attacks; RMM: resistance against man-in-the-middle attacks; RR: resistance against replay attacks; RKS: resistance against known session key attacks; PI: resistance against privileged-insider attacks; FA: formal analysis; ●: supported; ○: not supported or considered.

for anonymity in terms of privacy preservation. Firstly, these schemes do not support the update and unlinkability of vehicle public keys or pseudonyms, which are however required to prevent attackers from linking vehicle behaviors. Secondly, these schemes do not consider the tracking of malicious vehicles or users. Lastly, they do not account for the possibility of tracers maliciously colluding with rogue vehicles to falsely implicate honest vehicles, hence they do not support the public verification and robustness of tracing.

In order to fill this gap, we propose LBRAKA, a lattice-based robust authenticated key agreement scheme for VANETs, which supports anonymous authentication and key negotiation between RSUs and vehicles. To support unlinkability, vehicle certificates and public keys are assigned obfuscated expiration time, making attackers difficult to track a vehicle through publicly available network information. During registration, vehicles need to commit to their real identities through signatures, thus the real identities can be revealed from the commitment to support public verification during tracing. With respect to tracing robustness, the central authority should commit to the real identity and public key of vehicles during vehicle registration. Thus, if the central authority attempts to frame an honest vehicle during tracing, the vehicle can disclose the central authority's commitment with its real identity and public key, proving that the authority issued two valid credentials and it is framed. We summarize the main contributions of this paper as follows:

(1) We provide a lattice-based robust authenticated key agreement scheme for VANETs, which supports anonymous authentication and key negotiation between vehicles and RSUs.

(2) We set obfuscated expiration time for vehicle public keys and credentials, which enables their timely updates and ensures the unlinkability of new and old public keys or credentials.

(3) We require vehicles to create commitments to their real identities, and the central authority to generate commitments to vehicles' public keys and real identities, thereby supporting publicly verifiable traceability and detection of collusion attacks (i.e., framing) for tracing robustness.

(4) We conduct both formal and informal analyses of LBRAKA. The results demonstrate that LBRAKA not only meets the design requirements for anonymity, conditional privacy, unlinkability, key escrow freedom, public verification of tracing, and tracing robustness, but also effectively protects against impersonation attacks, man-in-the-middle attacks, replay attacks, known session key attacks, and quantum attacks.

(5) We evaluate LBRAKA's performance. The results indicate that, compared to other related schemes, LBRAKA has potential usability.

The rest of this paper is structured as follows. Section II offers a review of related work. Section III specifies the problem statements by describing LBRAKA's preliminaries, system model, security and privacy requirements, and threat model. Section IV details the LBRAKA design, followed by performance analysis and evaluation in Section V. Finally, Section VI concludes this paper.

II. RELATED WORK

Researchers have suggested numerous authenticated key agreement (AKA) schemes for VANETs. However, these schemes largely rely on traditional hard problems like large integer factorization, making them vulnerable to quantum attacks [26]. Therefore, we review quantum-resistant AKA schemes for VANETs, and highlight their limitations. A comparison of these schemes with LBRAKA is provided in TABLE I to demonstrate LBRAKA's superiority, and the schemes are summarized in terms of used techniques, advantages and limitations in TABLE II.

Feng et al. proposed an ideal lattice based anonymous authentication scheme for mobile devices [10], which adopts the ring learning with errors (RLWE) problem to achieve mutual authentication and key exchange between users and a server. Dabra et al. analyzed that Feng et al.'s scheme [10] suffers from signal leakage attacks, and presented a lattice

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

TABLE II
COMPARATIVE SUMMARY OF RELATED WORKS

	Techniques	Advantages	Limitations
[10], [11], [14], [16]–[19]	Hash function and RLWE	Supporting anonymity, key escrow freeness and formal analysis	Failing to support handover authentication, unlinkability, conditional privacy, public verification of tracing, and tracing robustness
[12]	Hash function and module LWE	Supporting formal analysis	Lack of anonymity, unlinkability, conditional privacy, public verification of tracing, tracing robustness, and key escrow freeness
[13]	Hash function and NTRU	Supporting anonymity, and formal analysis	Lack of unlinkability, conditional privacy, public verification of tracing, tracing robustness, and key escrow freeness
[15]	Hash function, RLWE, and fuzzy extractor	Supporting anonymity, key escrow freeness and formal analysis	Failing to support handover authentication, unlinkability, conditional privacy, public verification of tracing, and tracing robustness
[20]	Hash function and module learning with rounding	Supporting anonymity, key escrow freeness and formal analysis	Lack of unlinkability, conditional privacy, public verification of tracing, and tracing robustness
[21]	Hash function and RLWE	Supporting anonymity, key escrow freeness and formal analysis	Lack of unlinkability, conditional privacy, public verification of tracing, and tracing robustness
[22]	Hash function, LWE, and ISIS	Supporting anonymity, key escrow freeness and formal analysis	Lack of unlinkability, conditional privacy, public verification of tracing, and tracing robustness
[23]	Hash function and RLWE	Supporting anonymity, unlinkability, conditional privacy, and formal analysis	Failing to support key agreement, public verification of tracing, tracing robustness, and key escrow freeness
[24], [25]	Hash function and RLWE	Supporting formal analysis	Failing to support key agreement, anonymity, unlinkability, conditional privacy, public verification of tracing, tracing robustness, and key escrow freeness

based anonymous password AKA scheme [11]. However, it still fails to resist against denial-of-service attacks. After that, Ding et al. further analyzed and concluded that Dabra et al.'s scheme [11] still suffers from signal leakage attacks. Thus, they presented an improved version [14]. Dharminder et al. presented a three factor AKA scheme [15], which still suffers from signal leakage attacks. Islam et al. presented two quantum-resistant two-party AKA schemes [16], [17], but Dabra et al. found that signal leakage attacks exist in the two schemes [27]. Wang et al. proposed a quantum resistant two factor authentication scheme for mobile devices, which enables the user and server to authenticate each other and share a session key [18]. In 2022, a lattice based reconciliation enabling key exchange scheme was proposed for internet of things environments [19]. However, these schemes fail to consider unlinkability, conditional privacy, public verification of tracing, and tracing robustness. In addition, they focus on the mutual AKA between the user and server, thus failing to be applied into handover authentication between RSUs and vehicles directly.

Basu et al. presented a module learning with rounding based AKA scheme for two party communications, in order to ensure efficiency [20]. Islam and Basu proposed a password-based three-party AKA (PB-3PAKA) scheme for mobile devices, which enables two users to authenticate each other and establish a common session key [21]. Wei et al. introduced a lattice based certificateless anonymous AKA scheme for internet of things (IoTs) [22], which relies on LWE [28] and inhomogeneous small integer solution (ISIS) problems. However, these three schemes cannot support unlinkability, conditional privacy, public verification of tracing, and tracing robustness. Xue et al. introduced an efficient lattice-based authenticated key exchange scheme that incorporates key encapsulation mechanisms and signatures [12]. However, this scheme does not address several critical aspects such as anonymity, unlinkability, conditional privacy, public verification of tracing, tracing robustness, and key escrow problem. Consequently, it may not be suitable for use in VANETs. Zhang et al. introduced a quantum-resistant handover authentication scheme [13], which adopts the number theory research unit (NTRU) encryption algorithm [29] and can guarantee the anonymity of mobile devices. Nevertheless, this scheme fails to consider unlinkability, conditional privacy, public verification of tracing, tracing robustness, and key escrow freeness.

Prajapat et al. proposed a lattice based aggregate signature scheme for VANETs [23], which aggregates multiple users' signatures as a single signature. This work can offer anonymity, unlinkability and conditional privacy, but cannot provide key agreement, public verification of tracing, tracing robustness, and key escrow freeness. The reason key escrow freeness cannot be satisfied is that the vehicle's private and public keys are entirely generated by a trusted agency. Bagchi et al. presented an aggregate signature method based on RLWE, and applied this method with blockchain for the real-time Internet of drones applications [24]. In addition, a lattice based security framework was designed by using aggregate signature in an ambient intelligence-assisted private blockchain-based IoT environment [25], where the signatures on medical devices' secret message ciphertexts are aggregated and verified. However, these two works mainly focus

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

on authentication rather than authenticated key agreement. Additionally, they do not consider anonymity, unlinkability, conditional privacy, public verification of tracing, tracing robustness, and key escrow freeness.

In summary, there are rarely quantum resistant AKA schemes, which enable mutual authentication and key exchange between vehicles and RSUs in VANETs. In TABLE I, we compare LBRAKA with the aforementioned related works with respect to various evaluation criteria, such as anonymity and unlinkability. TABLE I demonstrates that the majority of existing quantum-resistant schemes fail to support robust and conditional privacy-preserving authenticated key agreement with unlinkability for VANETs.

III. Problem Statements

A. Preliminaries

1) Secure Hash Function: A secure hash function $H(*)$ can map the input data x of arbitrary length to an l-length output, $H:\{0,1\}^* \rightarrow \{0,1\}^l$. $H(*)$ is one-way, which implies that it is infeasible to extract x from $H(x)$. Meanwhile, given $H(x)$, it is impossible to obtain x' satisfying $H(x)=H(x')$.

Let $\mathbb{N}$ and $\mathbb{Z}$ be the set of all natural numbers and integers, respectively. Given $f \in \mathbb{N}$, we define $n=2^f \in \mathbb{N}$. Suppose q is a large prime, which satisfies $q \mod 2n=1$, and the finite field $\mathbb{Z}/q\mathbb{Z}$ is denoted as $\mathbb{Z}_q$, where $\mathbb{Z}_q=\{0,1,\cdots,q-1\}$. Now, we define $R=\mathbb{Z}[x]/<x^n+1>$, $R_q=\mathbb{Z}_q[x]/<x^n+1>$, $R_{q,k} \subset R_q$, so that $R_{q,k}=\{g(x) \in R_q : g(x)$'s degree is at most $n-1$ and $g(x)$'s all coefficients belong to $[-k,k]\}$ with $0<k \leq (q-1)/2$, and $D_{32}^n, n \geq 512$ contains all polynomials of degree at most $n-1$, which have all 0 coefficients except at most 32 coefficients belonging to $\{+1,-1\}$. Given $\boldsymbol{a}=a_0+a_1x+\cdots+a_{n-1}x^{n-1} \in R$, L_2 and L_∞ norms are represented as $||\boldsymbol{a}||=\sqrt{a_0^2+\cdots+a_{n-1}^2}$ and $||\boldsymbol{a}||_\infty = max\{a_i\}_{i=1}^{n-1}$. Suppose χ_β represents the discrete Gaussian distribution on R_q, where β is the standard deviation of the distribution χ_β.

2) RLWE Distribution: Suppose ψ is a discrete Gaussian distribution over R_q with a small standard deviation β. A distribution $\boldsymbol{A}_{\boldsymbol{s},\psi}$ is generated by sampling pairs $(\boldsymbol{a}_i,\boldsymbol{b}_i)$ from $R_q \times R_q$, where $\boldsymbol{a}_i$ is sampled from R_q and $\boldsymbol{b}_i=\boldsymbol{a}_i\boldsymbol{s}+\boldsymbol{e}$ with $\boldsymbol{s},\boldsymbol{e} \leftarrow Sample(\psi)$.

3) RLWE Problem: Given a fixed $\boldsymbol{s}$ sampled from ψ, and a polynomial number of samples, $\boldsymbol{A}_{\boldsymbol{s},\psi}$ is indistinguishable from the uniform distribution over $R_q \times R_q$.

4) Search RLWE Problem: Let $\{\boldsymbol{s},\boldsymbol{a}_i,\boldsymbol{e}\}$ be as defined in the RLWE distribution. It is infeasible to recover $\boldsymbol{s}$ from $T, T>1$ samples of $(\boldsymbol{a}_i, \boldsymbol{a}_i\boldsymbol{s}+\boldsymbol{e}_i \mod q)$ [30].

Lemma 1: Given $\boldsymbol{a},\boldsymbol{b} \in R$, $||\boldsymbol{ab}|| \leq \sqrt{n}||\boldsymbol{a}||||\boldsymbol{b}||$ and $||\boldsymbol{ab}||_\infty \leq n||\boldsymbol{a}||_\infty||\boldsymbol{b}||_\infty$ hold [31].

Lemma 2: For a positive real number $\beta=\omega\sqrt{\log n}$, $Pr_{\boldsymbol{a}\leftarrow\chi_\beta}[||\boldsymbol{a}|| > \beta\sqrt{n}] \leq 2^{-n+1}$ holds [32].

Assume $\mathbb{Z}_q=\{-\frac{q-1}{2},\cdots,\frac{q-1}{2}\}$ and its middle subset $E=\{-\lfloor\frac{q}{4}\rceil,\cdots,\lfloor\frac{q}{4}\rceil\}$. There is a characteristic or signal function, $Cha(*)$, which is the complement of E and satisfies $Cha(x)=0$ if $x \in E$ and $Cha(x)=1$ otherwise for given $x \in \mathbb{Z}_q$. In addition, the $Mod_2(*)$ function is employed to create the session key in our scheme, and the $Mod_2(*)$ function $Mod_2 : \mathbb{Z}_q \times \{0,1\} \rightarrow \{0,1\}$ is represented as $Mod_2(u,b)=(u+b\frac{q-1}{2}) \mod q \mod 2$, where $u \in \mathbb{Z}_q$ and $b=Cha(u)$ [33].

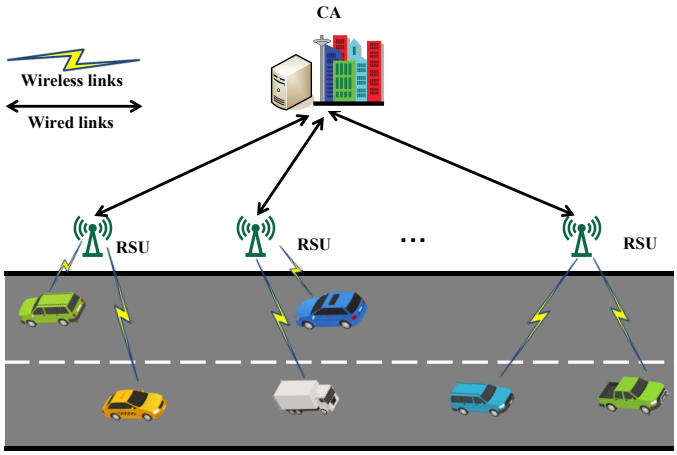


Fig. 1. System model

Lemma 3: Given an odd prime q, $\boldsymbol{u},\boldsymbol{b} \in R_q$ satisfying $||\boldsymbol{e}|| < \frac{q}{8}$, $Mod_2(\boldsymbol{b},Cha(\boldsymbol{b}))=Mod_2(\boldsymbol{u},Cha(\boldsymbol{b}))$ holds [34], where $\boldsymbol{u}=\boldsymbol{b}+2\boldsymbol{e}$.

We can extend $Cha(*)$ to elements in the ring R_q, $Cha(\boldsymbol{b})=(Cha(b_0),\cdots,Cha(b_{n-1}))$, where $\boldsymbol{b}=b_0+b_1x+\cdots+b_{n-1}x^{n-1}$. Meanwhile, we can also extend $Mod_2(*)$ to elements in the ring R_q.

5) Generalized Lattice Based Signature (GLS) Scheme: This scheme was presented in [35], which contains three algorithms: (1) KeyGen, (2) Sign, and (3) Verification. A secure hash function $H:\{0,1\}^* \rightarrow D_{32}^n$ is adopted.

(1) $KeyGen$: This algorithm is to generate the public/private key pair of signer.

- Randomly select $\boldsymbol{a} \in R_q$ and $(\boldsymbol{s}_1,\boldsymbol{s}_2) \in R_{q,1} \times R_{q,1}$.
- Calculate $\boldsymbol{p}=\boldsymbol{a}\boldsymbol{s}_1+\boldsymbol{s}_2$.
- Set the signer's private key $sk=(\boldsymbol{s}_1,\boldsymbol{s}_2)$ and public key $pk=(\boldsymbol{a},\boldsymbol{p})$.

(2) $Sign(sk,m)$: This algorithm aims to generate a signature on the message m by adopting the private key sk.

- Select $(\boldsymbol{y}_1,\boldsymbol{y}_2) \in R_{q,k} \times R_{q,k}$.
- Calculate $\boldsymbol{c}=H(\boldsymbol{a}\boldsymbol{y}_1+\boldsymbol{y}_2,m)$, $\boldsymbol{z}_1=\boldsymbol{s}_1\boldsymbol{c}+\boldsymbol{y}_1$, and $\boldsymbol{z}_2=\boldsymbol{s}_2\boldsymbol{c}+\boldsymbol{y}_2$.
- When $\boldsymbol{z}_1$ or $\boldsymbol{z}_2 \notin R_{q,k-32}$, restart the signature algorithm.
- Set the signature as $sig=(\boldsymbol{z}_1,\boldsymbol{z}_2,\boldsymbol{c})$.

(3) $Verification(m,pk,sig)$: This algorithm is to verify the signature.

- Check $\boldsymbol{z}_1$ and $\boldsymbol{z}_2 \in R_{q,k-32}$.
- Check $\boldsymbol{c}=H(\boldsymbol{a}\boldsymbol{z}_1+\boldsymbol{z}_2-\boldsymbol{p}\boldsymbol{c},m)$.

B. System Model

Fig. 1 illustrates the system model of LBRAKA, which primarily comprises three types of entities: the vehicles, roadside units and central authority. Additionally, the responsibilities of each entity are described as follows:

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

Vehicles (V) have onboard units (OBUs). These OBUs are capable of communicating with roadside units (RSUs) by using dedicated short-range communications (DSRC) protocols. In this operational scenario, OBUs seek to access services or send messages by interacting with RSUs. Vehicles are not supposed to be honest, because they might launch attacks on VANETs by interacting with roadside units without being held accountable.

Roadside units (RSU) generally deployed and managed by the central authority (e.g., transport management center and VANET operator), serve as access points that authenticate vehicles and facilitate the forwarding of messages to and from vehicles. RSUs are generally considered to be honest but curious about sensitive information, such as the real identities of vehicles.

Central authority (CA) is a trusted entity (e.g., transport management center) within the entire system, equipped with enough computational and storage resources. The CA's primary responsibilities are to help initialization, registration, update, and robust tracing. We suppose that CA can defend against intrusions (e.g., by using intrusion detection systems [36], firewalls [37] and intrusion prevention systems [38]), which prevents adversaries from compromising CA to obtain the list of vehicle <public key, real identity> to link the behaviors of the vehicle. In addition, CA can adopt the physically unclonable function (PUF) or trusted execution environments (TEE) to protect the list locally [6]. However, ensuring that the list cannot be compromised within CA is beyond the scope of this paper. Although we assume CA is trusted, we also consider that CA could be corrupted to collude with malicious vehicles in order to accuse honest vehicles.

C. Security and Privacy Requirements

The security and privacy requirements of LBRAKA are described as follows.

(1) *Mutual authentication:* A vehicle and a RSU should authenticate with each other for determining each other's legitimacy.

(2) *Key agreement:* A vehicle and a RSU should establish a session key for future secure communications.

(3) *Anonymity:* Except for CA, no one can reveal a vehicle's real identity from messages transmitted by the vehicle over an open channel.

(4) *Conditional privacy:* When a vehicle's misbehaviors are detected under an access request, CA has the capability to uncover the vehicle's real identity based on the request sent by this vehicle.

(5) *Unlinkability:* An attacker cannot link the public keys or credentials of a vehicle through observing the public information of this vehicle, in order to link the vehicle's behaviors.

(6) *Key escrow freeness:* The public/private keys of a vehicle should not be fully created by other entities.

(7) *Public verification of tracing:* The process by which CA traces malicious vehicles should be subject to public verification.

(8) *Tracing robustness:* When tracing a malicious vehicle, LBRAKA should safeguard against the possibility of a corrupted CA framing honest vehicles.

(9) *Resistance against attacks:* LBRAKA should be resilient against most known attacks in VANETs, such as impersonation attacks, man-in-the-middle attacks, replay attacks, known session key attacks, and privileged-insider attacks. Additionally, LBRAKA should be designed to withstand quantum attacks.

D. Threat Model

We mainly identify three types of adversaries within VANETs, namely vehicles, RSUs, and external entities other than vehicles and RSUs. Malicious vehicles and external entities may engage in various malicious activities, such as eavesdropping on communication channels to gather traffic data, attempting to decrypt ciphertexts to access secret information, modifying parts of intercepted valid messages and forwarding them to target receivers, or impersonating legitimate vehicles or RSUs. RSUs are assumed to be honest but curious, thus they could collect traffic data via them, and try to decrypt ciphertexts to access secret information. In addition, although CA is supposed to be trusted, CA could be corrupted to accuse honest vehicles.

We assume that vehicles can establish secure communication channels with CA for purposes (i.e., registration and update) [6]. Additionally, we suppose that vehicles, RSUs and CA can employ handshake protocols to synchronize their time [39].

IV. LBRAKA Design

We introduce the LBRAKA design in this section. The flowchart of this design is shown in Fig. 2. In the initialization, CA configures system parameters. During vehicle registration, a vehicle registers with CA. In detail, the vehicle sends CA its identity and public key with a commitment to its identity. Then, CA sends the vehicle a credential with a commitment to the vehicle's identity and public key. In the RSU registration, RSU sends CA its identity and public key, and obtains a credential from CA. During mutual authentication and key agreement, the vehicle authenticates with RSU, and they create a common session key. In the update, once the expiration date of the vehicle's credential approaches, the vehicle and CA should update the credential of the vehicle and their commitments. With respect to robust tracing, if the misbehaviors of a vehicle are detected publicly, CA should release a proof and the identity of the vehicle to trace this malicious vehicle. Once the vehicle discovers that it is falsely accused by CA, it publishes a proof to prove its innocence. For the simplicity of presentation, the notations we mainly use in the rest of this paper are shown in TABLE III.

A. Initialization

In this phase, CA generates system parameters and announces public parameters.

(1) CA selects a security parameter n, so that n is the power of 2. Then, it selects an odd prime number q satisfying $q \mod 2n = 1$, a discrete Gaussian distribution χ_β over $R_{q,1}$ with a standard deviation β and $\boldsymbol{a} \in R_q$.

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

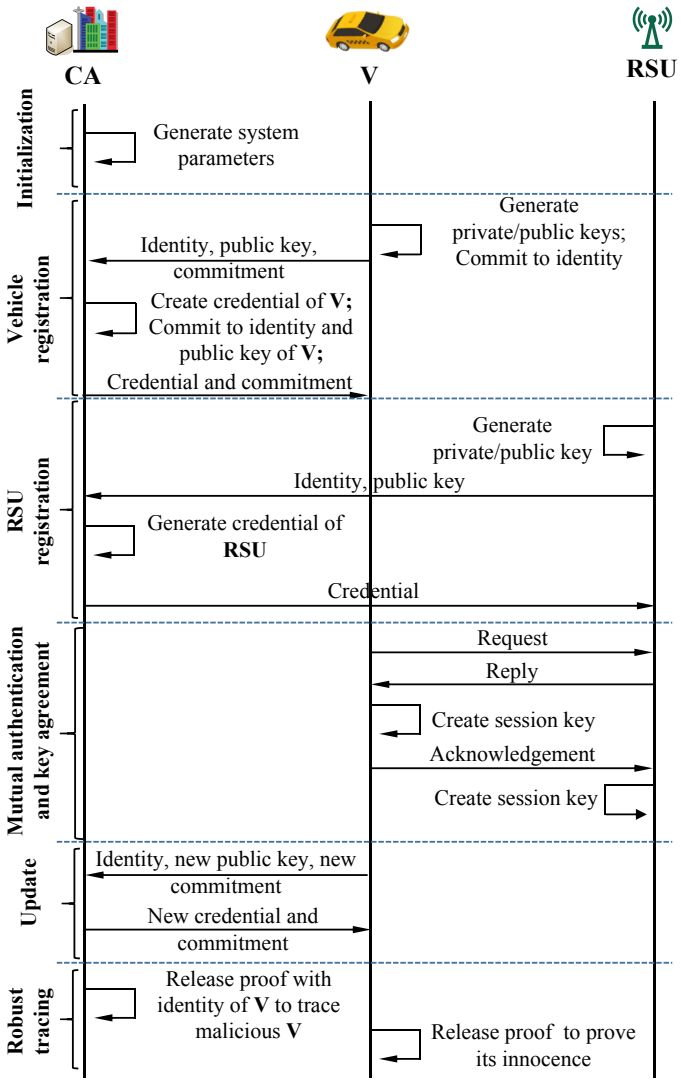


Fig. 2. Flowchart of LBRAKA

TABLE III
NOTATIONS

Symbols	Descriptions
q	An odd prime.
n	A security parameter $n = 2^f$, so that $q \mod 2n = 1$.
R	$\mathbb{Z}[x]/ < x^n + 1 >$, the quotient ring of the polynomial ring.
R_q	$\mathbb{Z}_q[x]/ < x^n + 1 >$, where $\mathbb{Z}_q = \{0, \cdots, q-1\}$.
$R_{q,1}$	The subset of R_q, so that all the coefficients of each polynomial of this set belong to $[-1, 1]$.
χ_β	The Gaussian distribution over $R_{q,1}$ with a standard deviation β.
$H(*)$	A secure hash function $H : \{0,1\}^* \to \{0,1\}^l$.
ID_X	The real identity of the entity X.
$(\boldsymbol{p}_X, \boldsymbol{a}_X)$	The public key of X.
$(\boldsymbol{s}_X, \boldsymbol{e}_X)$	The private key of X.
$Cred_X$	The credential issued to X.
δ_v	The vehicle's signature (i.e., commitment) on its real identity.
η_v	The central authority's signature (i.e., commitment) on the real identity and public key of vehicle.

(2) CA randomly samples $\boldsymbol{s}, \boldsymbol{e} \leftarrow \chi_\beta$, and computes $\boldsymbol{p} = \boldsymbol{a}\boldsymbol{s} + 2\boldsymbol{e}$, where $(\boldsymbol{s}, \boldsymbol{e})$ are CA's master private key, and $(\boldsymbol{a}, \boldsymbol{p})$ are CA's master public key.

(3) CA selects a secure hash function, $H(*)$, which is described as $H : \{0,1\}^* \to \{0,1\}^l$.

(4) CA publishes public parameters $\{n, q, \chi_\beta, \boldsymbol{a}, \boldsymbol{p}, H(*)\}$.

B. Registration

As depicted in Fig. 3 and Fig. 4, vehicles and RSUs should register with CA through secure channels, in order to obtain authorization from CA.

Vehicle Registration

(1) A vehicle V selects its public/private keys, $(\boldsymbol{a}_v, \boldsymbol{p}_v)$ and $(\boldsymbol{s}_v, \boldsymbol{e}_v)$, which satisfy $\boldsymbol{p}_v = \boldsymbol{a}_v\boldsymbol{s}_v + 2\boldsymbol{e}_v$.

(2) In order to support robust tracing, V uses its private key $(\boldsymbol{s}_v, \boldsymbol{e}_v)$ to create a signature or commitment δ_v on its real identity ID_v through the Sign algorithm of GLS scheme. At last, it sends $\{ID_v, \boldsymbol{a}_v, \boldsymbol{p}_v, \delta_v\}$ to CA for registration.

(3) CA checks ID_v, and verifies the validity of δ_v through the Verification algorithm of GLS scheme. When all verifications hold, CA employs its private key $(\boldsymbol{s}, \boldsymbol{e})$ to create a credential $Cred_v$ on V's public key $(\boldsymbol{a}_v, \boldsymbol{p}_v)$ and an obfuscated expiration date T_v. Note that after mutual authentication (similar to network access authentication) between V and RSU, V could use its public key to access different services through RSUs (similar to network access points) in open network environments, and an attacker could link V's new and old public keys to track V's behaviors by observing the appearance of new public key after the expiration of old ones. Thus, expiration time obfuscation should be offered. To be specific, CA can organize registered vehicles as a group or various subgroups with the same size [40], and assign the same expiration time for credentials issued to members of the same group or subgroup, in order to ensure unlinkability. Meanwhile, CA should create a signature η_v on $\{ID_v, \boldsymbol{a}_v, \boldsymbol{p}_v, T_v\}$ to support robust tracing. Finally, CA stores $\{ID_v, \boldsymbol{a}_v, \boldsymbol{p}_v, \delta_v\}$ for tracing, and issues $\{T_v, Cred_v, \eta_v\}$ to V.

(4) After receiving $\{T_v, Cred_v, \eta_v\}$ from CA, V checks their validity through the Verification algorithm. If the verification holds, V stores $\{ID_v, T_v, \boldsymbol{a}_v, \boldsymbol{p}_v, \boldsymbol{s}_v, \boldsymbol{e}_v, Cred_v, \eta_v\}$ for future authentication and key agreement with RSUs.

RSU Registration

(1) RSU chooses its public/private keys, $(\boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu})$ and $(\boldsymbol{s}_{rsu}, \boldsymbol{e}_{rsu})$, which satisfy $\boldsymbol{p}_{rsu} = \boldsymbol{a}_{rsu}\boldsymbol{s}_{rsu} + 2\boldsymbol{e}_{rsu}$. Then, it sends $\{ID_{rsu}, \boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu}\}$ to CA.

(2) After receiving $\{ID_{rsu}, \boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu}\}$ from RSU, CA checks ID_{rsu}, and then generates a credential $Cred_{rsu}$ on RSU's public key $(\boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu})$. At last, CA sends $Cred_{rsu}$ to RSU.

(3) After receiving the information from CA, RSU verifies $Cred_{rsu}$. If the verification holds, RSU stores $\{ID_{rsu}, \boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu}, \boldsymbol{s}_{rsu}, \boldsymbol{e}_{rsu}, Cred_{rsu}\}$ for the future authentication and key agreement with vehicles.

C. Mutual Authentication and Key Agreement

If V moves into the area of RSU, V and RSU perform mutual authentication and key agreement, in order to check

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

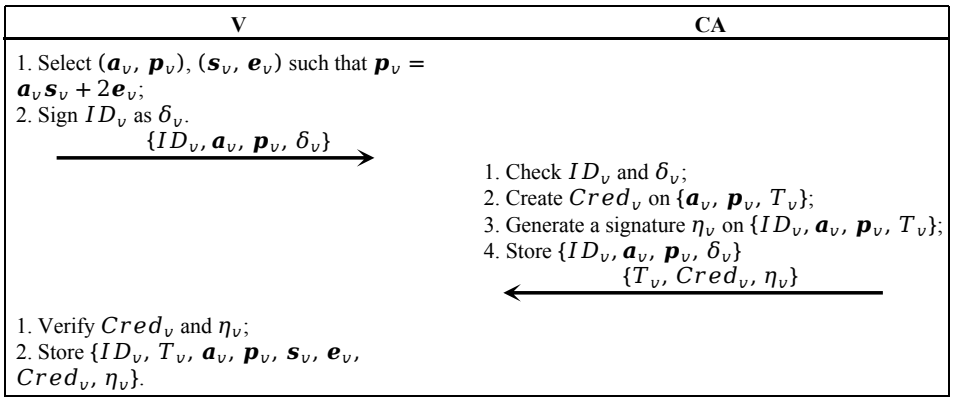


Fig. 3. The registration of vehicle

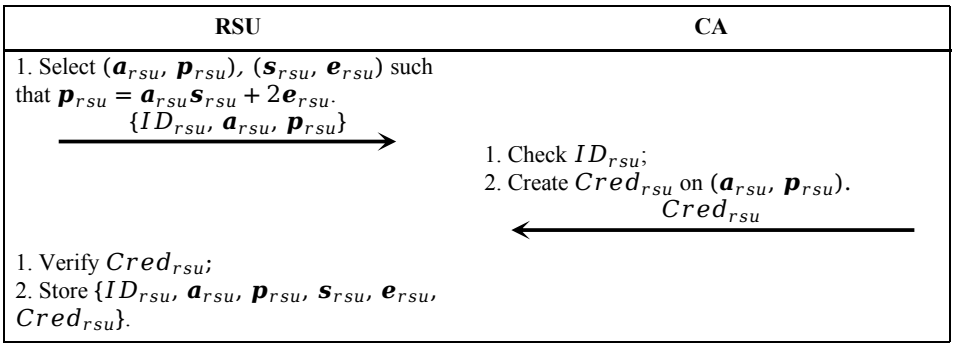


Fig. 4. The registration of RSU

the legitimacy of each other and generate a session key. The procedure is shown in Fig. 5. Note that V can broadcast its public key and credential $\{\boldsymbol{a}_v, \boldsymbol{p}_v, T_v, Cred_v\}$ to RSU's neighboring RSUs, and these RSUs can verify the received messages in advance for preparation. Meanwhile, V can obtain and verify a neighboring RSU' public key and credential $\{\boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu}, Cred_{rsu}\}$ through the cooperation of RSUs in advance for preparation [41].

(1) V randomly samples $\boldsymbol{r}_v, \boldsymbol{f}_v \in \chi_\beta$, and generates $\boldsymbol{x}_v = \boldsymbol{a}\boldsymbol{r}_v + 2\boldsymbol{f}_v$. Then, it employs the Sign algorithm to create a signature σ_v on $\{\boldsymbol{x}_v, t_0\}$ by using its private key, where t_0 is a timestamp. At last, V sends a request $Q_v = \{\boldsymbol{a}_v, \boldsymbol{p}_v, \boldsymbol{x}_v, t_0, \sigma_v\}$ to RSU.

(2) After receiving Q_v from V, RSU checks the freshness of t_0 and $t_0 \leq T_v$, and verifies σ_v. If all the verifications hold, it samples $\boldsymbol{r}_{rsu}, \boldsymbol{f}_{rsu} \in \chi_\beta$, and generates

$$\boldsymbol{x}_{rsu} = \boldsymbol{a}\boldsymbol{r}_{rsu} + 2\boldsymbol{f}_{rsu},$$

$$\boldsymbol{k}_{rsu} = \boldsymbol{x}_v\boldsymbol{r}_{rsu},$$

$$\boldsymbol{w}_{rsu} = Cha(\boldsymbol{k}_{rsu}),$$

$$\boldsymbol{\gamma}_{rsu} = Mod_2(\boldsymbol{k}_{rsu}, \boldsymbol{w}_{rsu}),$$

$$\boldsymbol{\kappa}_{rsu} = H(\boldsymbol{x}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}_{rsu}).$$

Similar to V, RSU adopts its private key $(\boldsymbol{s}_{rsu}, \boldsymbol{e}_{rsu})$ to create a signature σ_{rsu} on $\{\boldsymbol{x}_{rsu}, \boldsymbol{w}_{rsu}, \boldsymbol{\kappa}_{rsu}, t_1\}$, where t_1 is a timestamp. At last, RSU sends a reply $R_{rsu} = \{\boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu}, \boldsymbol{x}_{rsu}, \boldsymbol{w}_{rsu}, \boldsymbol{\kappa}_{rsu}, t_1, \sigma_{rsu}\}$ to V.

(3) After receiving R_{rsu} from RSU, V checks the freshness of t_1, and verifies σ_{rsu}. If all the verifications hold, V computes

$$\boldsymbol{k}_v = \boldsymbol{x}_{rsu}\boldsymbol{r}_v,$$

$$\boldsymbol{\gamma}'_{rsu} = Mod_2(\boldsymbol{k}_v, \boldsymbol{w}_{rsu}),$$

and checks

$$\boldsymbol{\kappa}_{rsu} = H(\boldsymbol{x}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}'_{rsu}).$$

Furthermore, V computes

$$\boldsymbol{w}_v = Cha(\boldsymbol{k}_v),$$

$$\boldsymbol{\gamma}_v = Mod_2(\boldsymbol{k}_v, \boldsymbol{w}_v),$$

$$\boldsymbol{\kappa}_v = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}'_{rsu}||\boldsymbol{\kappa}_{rsu}),$$

$$\boldsymbol{sk}_v = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}_v||\boldsymbol{\kappa}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}'_{rsu}||\boldsymbol{\kappa}_{rsu}),$$

where $\boldsymbol{sk}_v$ is considered as the session key. At last, V sends an acknowledgement $\{\boldsymbol{w}_v, \boldsymbol{\kappa}_v\}$ to RSU.

(4) Upon receiving $\{\boldsymbol{w}_v, \boldsymbol{\kappa}_v\}$, RSU computes

$$\boldsymbol{\gamma}'_v = Mod_2(\boldsymbol{k}_{rsu}, \boldsymbol{w}_v),$$

$$\boldsymbol{\kappa}'_v = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}'_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}_{rsu}||\boldsymbol{\kappa}_{rsu}).$$

If $\boldsymbol{\kappa}'_v = \boldsymbol{\kappa}_v$, RSU can obtain a session key

$$\boldsymbol{sk}_{rsu} = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}'_v||\boldsymbol{\kappa}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}_{rsu}||\boldsymbol{\kappa}_{rsu}).$$

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

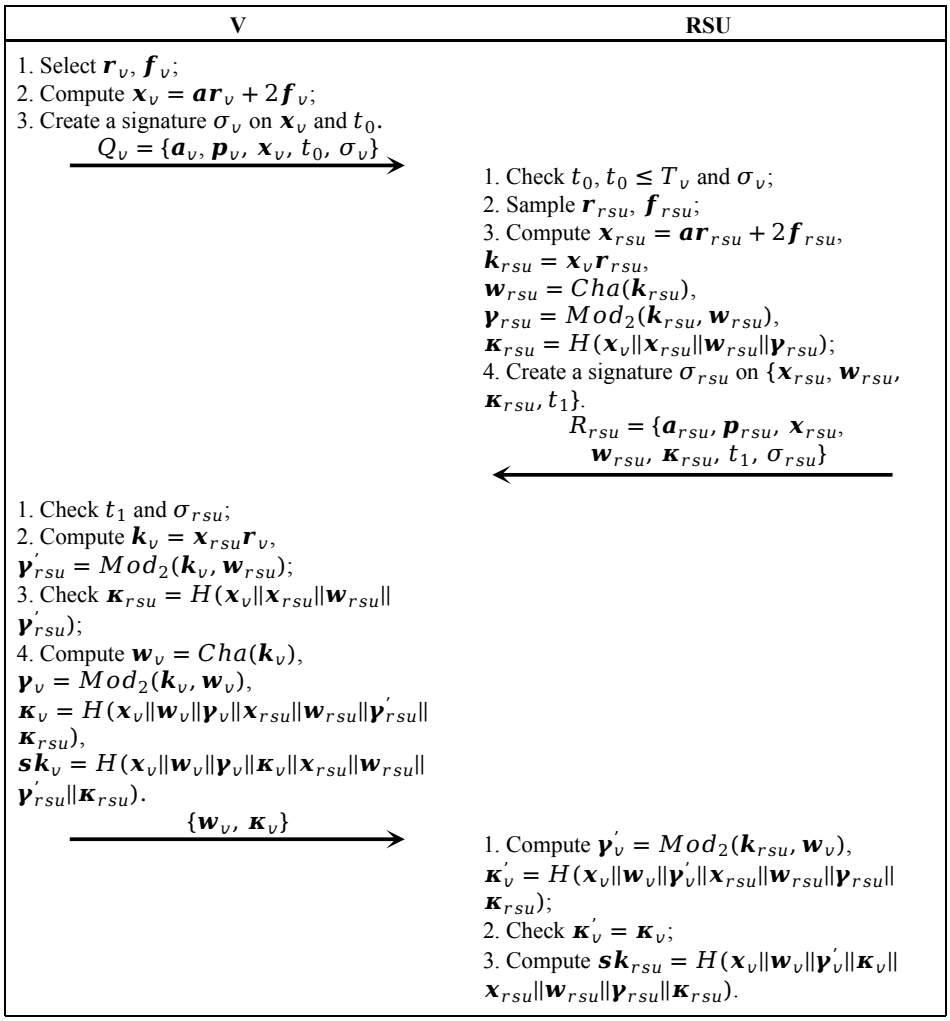


Fig. 5. Mutual authentication and key negotiation

Correctness condition: When V and RSU authenticate with each other, γ_{rsu} should be equal to γ'_{rsu} for authentication. Similarly, γ_v and γ'_v should be the same for establishing a common session key. Thus, we give the condition for the correctness of LBRAKA by considering γ_{rsu} and γ'_{rsu}, and the condition for γ_v and γ'_v can be derived by the same way.

γ_{rsu} and γ'_{rsu} are derived from $\boldsymbol{k}_{rsu}$ and $\boldsymbol{k}_v$ respectively. Thus, we can have

$$\begin{aligned}\boldsymbol{k}_{rsu} &= \boldsymbol{x}_v \boldsymbol{r}_{rsu} \\ &= (a\boldsymbol{r}_v + 2\boldsymbol{f}_v)\boldsymbol{r}_{rsu} \\ &= a\boldsymbol{r}_v \boldsymbol{r}_{rsu} + 2\boldsymbol{f}_v \boldsymbol{r}_{rsu}, \\ \boldsymbol{k}'_{rsu} &= \boldsymbol{x}_{rsu} \boldsymbol{r}_v \\ &= (a\boldsymbol{r}_{rsu} + 2\boldsymbol{f}_{rsu})\boldsymbol{r}_v \\ &= a\boldsymbol{r}_{rsu} \boldsymbol{r}_v + 2\boldsymbol{f}_{rsu} \boldsymbol{r}_v.\end{aligned}$$

Therefore, we further get

$$||\boldsymbol{k}_{rsu} - \boldsymbol{k}'_{rsu}|| \leq 2(||\boldsymbol{f}_{rsu} \boldsymbol{r}_v|| + ||\boldsymbol{f}_v \boldsymbol{r}_{rsu}||).$$

Based on *Lemma 1* and *Lemma 2*, we can obtain

$$||\boldsymbol{k}_{rsu} - \boldsymbol{k}'_{rsu}|| \leq 4\beta^2 n^{3/2}.$$

In addition, we can derive $||\boldsymbol{u} - \boldsymbol{b}|| < ||2\boldsymbol{e}|| < \frac{q}{4}$ from *Lemma 3*, and further get $||\boldsymbol{e}|| < \frac{q}{8}$. Thus, we can obtain the correctness condition of LBRAKA as follows.

$$||\boldsymbol{k}_{rsu} - \boldsymbol{k}'_{rsu}|| \leq 4\beta^2 n^{3/2} < \frac{q}{8},$$

$$q > 32\beta^2 n^{3/2}.$$

D. Update

In order to ensure unlinkability, V should update its public key when the obfuscated expiration date approaches. Similar to the registration of V, V should generate its new private/public keys, and create a signature on its real identity in order to support robust tracing. Then, it sends the real identity and public key with the signature to CA for registration. At last, CA issues a credential, obfuscated expiration time and a commitment to V, where the commitment is produced by CA based on the identity and public key of V and the time.

E. Robust Tracing

If V's misbehaviors are detected publicly under $Q_v = \{\boldsymbol{a}_v, \boldsymbol{p}_v, \boldsymbol{x}_v, t_0, \sigma_v\}$ with $\{Cred_v, T_v\}$, CA needs to trace V.

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

(1) The honest CA releases V's commitment $\{ID_v, \delta_v\}$ with $\{Q_v, Cred_v, T_v\}$ to trace V.

(2) Any verifiers can verify the validity of $\{Q_v, Cred_v, T_v\}$, which is similar to the process that RSU authenticates V.

(3) The verifier can check σ_v is V's signature on ID_v based on $\{\boldsymbol{a}_v, \boldsymbol{p}_v\}$.

When all the verifications hold, the tracing is confirmed by the verifier.

However, the corrupted CA could collude with the malicious V^* to frame the honest V. Through colluding, CA stores V^*'s signature δ_{v^*} on ID_V and issues $\{Cred_{v^*}, T_{v^*}, \eta_{v^*}\}$ to V^*.

(1) After V^*'s misbehaviors are detected under $Q_{v^*} = \{\boldsymbol{a}_{v^*}, \boldsymbol{p}_{v^*}, \boldsymbol{x}_{v^*}, t_0^*, \sigma_{v^*}\}$ and $\{Cred_{v^*}, T_{v^*}\}$, the corrupted CA attempts to release V^*'s commitment $\{ID_v, \delta_{v^*}\}$, Q_{v^*} and $\{Cred_{v^*}, T_{v^*}\}$ to frame V, where δ_{v^*} is created on ID_v by V^*.

(2) Any verifiers verify the validity of V^*'s request Q_{v^*}, $\{cred_{v^*}, T_{v^*}\}$ and δ_{v^*} through credential and signature verification.

(3) When V knows it is framed by the corrupted CA, it can publish CA's commitment $\{ID_v, \boldsymbol{a}_v, \boldsymbol{p}_v, T_v, \eta_v\}$ to prove the framing, where T_v is embedded into η_v.

(4) The verifier can check $t_0^* \leq T_v$ and $(\boldsymbol{a}_v, \boldsymbol{p}_v) \neq (\boldsymbol{a}_{v^*}, \boldsymbol{p}_{v^*})$, and verify η_v on $\{ID_v, \boldsymbol{a}_v, \boldsymbol{p}_v, T_v\}$. If all verifications hold, the verifier is convinced that CA has distributed two valid credentials on the same identity ID_v, thus it considers CA is corrupted.

V. Performance Analysis and Evaluation

We analyze the proposed scheme with respect to various security properties, and evaluate its performance through simulations.

A. Performance Analysis

We prove LBRAKA formally, and informally analyze it in terms of anonymity, conditional privacy, unlinkability, key escrow freeness, public verification of tracing, tracing robustness, and the resistance against impersonation attacks, man-in-the-middle attacks, replay attacks, known session key attacks, and quantum attacks.

1) Formal Analysis: The security of LBRAKA is proved based on the real-or-random (ROR) model [42], and some definitions are involved in the ROR model as follows.

Participants: Let $\prod_V^e$ and $\prod_{RSU}^f$ be the instance e and f of V and RSU, respectively.

Accepted states: When all messages between RSU and V are transmitted in order and the last message is received, $\prod_V^e$ and $\prod_{RSU}^f$ are under accepted states.

Partnering: When $\prod_V^e$ and $\prod_{RSU}^f$ share the same session and are in accepted states, they are regarded as partners.

Freshness: When the session key $\boldsymbol{sk}$ between $\prod_V^e$ and $\prod_{RSU}^f$ is not acquired by an adversary $\mathcal{A}$, $\prod_V^e$ and $\prod_{RSU}^f$ are considered fresh.

Adversary: $\mathcal{A}$ can engage in interactions between $\prod_V^e$ and $\prod_{RSU}^f$ by utilizing the following oracle queries, with the capability to read, modify, and replay messages within the interaction.

1) ***Execute***$(\prod_V^e, \prod_{RSU}^f)$***:*** This query indicates that $\mathcal{A}$ has the capability to intercept messages exchanged between $\prod_V^e$ and $\prod_{RSU}^f$.
2) ***Send***$(\prod_V^e, \prod_{RSU}^f, m)$***:*** This query represents a form of active attack. $\mathcal{A}$ can create, intercept, modify, and replay a message m to either $\prod_V^e$ or $\prod_{RSU}^f$. Upon receiving m, $\prod_V^e$ or $\prod_{RSU}^f$ sends a response back to $\mathcal{A}$.
3) ***CorruptV***$(\prod_V^e)$***:*** $\mathcal{A}$ can use this query to access the secret of $\prod_V^e$.
4) ***CorruptRSU***$(\prod_{RSU}^f)$***:*** $\mathcal{A}$ can employ this query to extract the secret of $\prod_{RSU}^f$.
5) ***Test***$(\prod_V^e, \prod_{RSU}^f)$***:*** When $\prod_V^e$ and $\prod_{RSU}^f$ have the same session key $\boldsymbol{sk}$ and $\mathcal{A}$ utilizes this query, $\mathcal{A}$ acquires $\boldsymbol{sk}$ if $b = 1$ or a random number if $b = 0$, with the value of b remaining unknown to $\mathcal{A}$. If $\prod_V^e$ and $\prod_{RSU}^f$ do not have the same session key and $\mathcal{A}$ employs this query, $\mathcal{A}$ receives an invalid symbol $\perp$.

Semantic security of session key: When LBRAKA is indistinguishable under the ROR model, we deduce that the creation of $\boldsymbol{sk}$ is sufficiently random to prevent information disclosure. $\mathcal{A}$ can execute ***Test***$(\prod_V^e, \prod_{RSU}^f)$, and attempt to guess the value of $b' \in \{0, 1\}$. If $b' = b$, $\mathcal{A}$ is considered successful. We designate LBRAKA as $\mathcal{P}$, and the advantage of $\mathcal{A}$ breaking the semantic security of ROR model within the polynomial time t is expressed as $Adv_{\mathcal{A}}^{\mathcal{P}} = |2 \times Pr[b' = b] - 1|$, where $Pr[EV]$ denotes the probability of an event EV occurring. If $Adv_{\mathcal{A}}^{\mathcal{P}}$ is negligible, $\mathcal{P}$ is regarded as secure under the ROR model.

Theorem 1: Let q_H be the number of hash queries, q_e the number of ***Execute***$(\prod_V^e, \prod_{RSU}^f)$ queries, q_s the number of ***Send***$(\prod_V^e, \prod_{RSU}^f, m)$ queries, $|Hash|$ the space range of hash function, and $Adv_{\mathcal{A}}^{RLWE}$ the advantage of $\mathcal{A}$ to address the RLWE problem. The advantage of $\mathcal{A}$ in breaking the semantic security of LBRAKA within the polynomial time t can be calculated as $Adv_{\mathcal{A}}^{\mathcal{P}} \leq \frac{{q_H}^2}{|Hash|} + \frac{(q_e+q_s)^2}{|SP_{\chi_\beta}|} + 2Adv_{\mathcal{A}}^{RLWE}$.

Proof: We employ the proof approach outlined in [43]. To calculate $Adv_{\mathcal{A}}^{\mathcal{P}}$, we define a series of games $Game_i, i = 0, \cdots, 3$, and designate $succ_i$ as the event where $\mathcal{A}$ correctly guesses $b' = b$ in $Game_i$.

$Game_0$: This game permits $\mathcal{A}$ to conduct actual attacks on the protocol $\mathcal{P}$. Given a random number b before the start of this game, and according to the semantic security of the session key, we have

$$Adv_{\mathcal{A}}^{\mathcal{P}} = |2Pr[succ_0] - 1|.$$

$Game_1$: $\mathcal{A}$ is permitted to intercept the messages exchanged between $\prod_V^e$ and $\prod_{RSU}^f$ through ***Execute***$(\prod_V^e, \prod_{RSU}^f)$. Finally, $\mathcal{A}$ performs ***Test***$(\prod_V^e, \prod_{RSU}^f)$ to differentiate the actual session key $\boldsymbol{sk}$ from a random number. Recall that $\boldsymbol{sk}_v = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}_v||\boldsymbol{\kappa}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}'_{rsu}||\ \boldsymbol{\kappa}_{rsu}) = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}'_v||\boldsymbol{\kappa}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}_{rsu}||\boldsymbol{\kappa}_{rsu}) = \boldsymbol{sk}_{rsu}$. However, $\mathcal{A}$ is unable to acquire $\{\boldsymbol{r}_{rsu}, \boldsymbol{r}_v\}$ through eavesdropping, and therefore cannot calculate the actual session key $\boldsymbol{sk}$. Thus, $Game_0$ is equivalent to $Game_1$, and we can obtain

$$Pr[succ_0] = Pr[succ_1].$$

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

$Game_2$: Based on $Game_1$, $\mathcal{A}$ is permitted to execute ***Send***$(\prod_V^e, \prod_{RSU}^f, m)$ and perform hash oracle queries. In this game, $\mathcal{A}$ initiates spoofing attacks by generating, editing, and sending a request to both $\prod_V^e$ and $\prod_{RSU}^f$. In LBRAKA, $\mathcal{A}$ can modify Q_v, R_{rsu} and $\{\boldsymbol{w}_v, \boldsymbol{\kappa}_v\}$. However, random elements and independent timestamps are embedded in these messages. Therefore, $\mathcal{A}$ should execute ***Send***$(\prod_V^e, \prod_{RSU}^f, m)$ without encountering any collisions. We should consider some collision could occur on $\{\boldsymbol{x}_v, \boldsymbol{x}_{rsu}\}$ and $\{\boldsymbol{\kappa}_{rsu}, \boldsymbol{\kappa}_v\}$. Based on the birthday paradox, we can get

$$|Pr[succ_1] - Pr[succ_2]| \leq \frac{{q_H}^2}{2|Hash|} + \frac{(q_s + q_e)^2}{2|SP_{\chi_\beta}|}.$$

$Game_3$: $\mathcal{A}$ is allowed to compromise the session key $\boldsymbol{sk}$ through ***CorruptV***$(\prod_V^e)$ or ***CorruptRSU***$(\prod_{RSU}^f)$. Note that the creation of the session key $\boldsymbol{sk} = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}_v||\boldsymbol{\kappa}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}_{rsu}||\ \boldsymbol{\kappa}_{rsu})$ depends on random $\{\boldsymbol{r}_{rsu}, \boldsymbol{r}_v, \boldsymbol{f}_{rsu}, \boldsymbol{f}_v\}$, which are only selected by $\prod_V^e$ and $\prod_{RSU}^f$ during mutual AKA. In addition, $\prod_V^e$ and $\prod_V^f$ do not store $\{\boldsymbol{r}_{rsu}, \boldsymbol{r}_v, \boldsymbol{f}_{rsu}, \boldsymbol{f}_v\}$ after using them. To compromise $\boldsymbol{sk}$, $\mathcal{A}$ should address the RLWE problem by calculating $\{\boldsymbol{k}_v, \boldsymbol{k}_{rsu}\}$ according to $\{\boldsymbol{x}_v, \boldsymbol{x}_{rsu}\}$, thus we get

$$|Pr[succ_2] - Pr[succ_3]| \leq Adv_{\mathcal{A}}^{RLWE}.$$

In the final game, $\mathcal{A}$ has utilized all available oracles to challenge the semantic security of $\mathcal{P}$ and attempts to win the game solely by guessing b. Therefore, we obtain

$$Pr[succ_3] = \frac{1}{2}.$$

By referring to the previous inequalities, we can get

$$\begin{aligned}
\frac{1}{2}Adv_{\mathcal{A}}^{\mathcal{P}} &= |Pr[succ_0] - \frac{1}{2}| \\
&= |Pr[succ_0] - Pr[succ_3]| \\
&\leq |Pr[succ_0] - Pr[succ_1]| + |Pr[succ_1] \\
&- Pr[succ_2]| + |Pr[succ_2] - Pr[succ_3]| \\
&\leq \frac{{q_H}^2}{2|Hash|} + \frac{(q_s + q_e)^2}{2|SP_{\chi_\beta}|} + Adv_{\mathcal{A}}^{RLWE}.
\end{aligned}$$

Thus, we deduce $Adv_{\mathcal{A}}^{\mathcal{P}} \leq \frac{{q_H}^2}{|Hash|} + \frac{(q_s+q_e)^2}{|SP_{\chi_\beta}|} + 2Adv_{\mathcal{A}}^{RLWE}$. $|Hash|$ and $|SP_{\chi_\beta}|$ are typically sufficiently large, and $Adv_{\mathcal{A}}^{RLWE}$ is adequately small due to the hardness assumption of the RLWE problem. As a result, the advantage $Adv_{\mathcal{A}}^{\mathcal{P}}$ of $\mathcal{A}$ in breaking the semantic security of LBRAKA is negligible, indicating that LBRAKA is semantically secure under the ROR model. ■

2) Informal Analysis: We discuss that LBRAKA meets the following security properties and resists against most know attacks.

Anonymity: In LBRAKA, no one (except CA) can reveal V's real identity ID_v from public information, since ID_v is not used publicly except in cases where V is traced due to its misbehaviors. Thus, LBRAKA can provide anonymity.

Conditional privacy: When V's misbehaviors are detected under $\{Q_v, Cred_v, T_v\}$, CA can uncover V's real identity ID_v. Specifically, CA can search ID_v locally in its database for releasing ID_v according to $\{\boldsymbol{p}_v, \boldsymbol{a}_v\}$ that are included in Q_v. Thus, LBRAKA can ensure conditional privacy.

Unlinkability: During update, V should interact with CA to renew its public/private key pair when its public key or credential expires. CA assigns an obfuscated expiration date for V's new credential, and issues the new credential with the obfuscated expiration time to V. Thus, any attackers (except CA) cannot link V's behaviors based on public information (i.e., its public key, credential and expiration time). As a result, LBRAKA can ensure unlinkability.

Key escrow freeness: During registration, vehicles autonomously select their own keys, thereby ensuring that LBRAKA achieves key escrow freeness.

Public verification of tracing: During tracing, CA publishes $\{ID_v, \delta_v\}$ with $\{Q_v, Cred_v, T_v\}$ to trace V. Anyone can act as a verifier to confirm $\{Q_v, Cred_v, T_v\}$ and $\{ID_v, \delta_v\}$, thereby enabling public verification.

Tracing robustness: As shown in *Robust Tracing*, if the corrupted CA tries to collude with the malicious vehicle V^* for framing the honest V, V can release $\{ID_v, T_v, \boldsymbol{p}_v, \boldsymbol{a}_v, \eta_v\}$ to prove CA has issued two valid credentials for the same ID_v, which implies CA is corrupted. Therefore, LBRAKA ensures tracing robustness.

Resistance against impersonation attacks: In order to launch impersonation attacks, an attacker should be able to forge the signatures in $\{Q_v, R_{rsu}\}$. In order to forge the signature, the attacker should know the private keys of V and RSU according to their public keys. Thus, the attacker's advantage to forge is negligible due to the hardness of the RLWE problem. Therefore, LBRAKA can withstand the impersonation attacks.

Resistance against man-in-the-middle attacks: LBRAKA facilitates mutual authentication between V and RSU by using signatures and credentials to verify each other's legitimacy. Thus, if an attacker aims to successfully execute man-in-the-middle attacks, they must be capable of impersonating one of the vehicle and RSU. However, our analysis has shown that LBRAKA can resist against impersonation attacks, thereby allowing LBRAKA to withstand the man-in-the-middle attacks effectively.

Resistance against replay attacks: In LBRAKA, $\{\boldsymbol{r}_{rsu}, \boldsymbol{r}_v, \boldsymbol{f}_{rsu}, \boldsymbol{f}_v\}$ are selected randomly and different in various sessions, and timestamps are embedded in signatures. Thus, V and RSU can detect replay attacks through verifying received messages. As a result, LBRAKA can withstand the replay attacks.

Resistance against known session key attacks: LBRAKA generates a session key between V and RSU as $\boldsymbol{sk} = H(\boldsymbol{x}_v||\boldsymbol{w}_v||\boldsymbol{\gamma}_v||\boldsymbol{\kappa}_v||\boldsymbol{x}_{rsu}||\boldsymbol{w}_{rsu}||\boldsymbol{\gamma}_{rsu}||\ \boldsymbol{\kappa}_{rsu})$, whose creation depends on random $\{\boldsymbol{r}_{rsu}, \boldsymbol{r}_v, \boldsymbol{f}_{rsu}, \boldsymbol{f}_v\}$. Thus, even if an adversary manages to compromise a session key $\boldsymbol{sk}$, it cannot compromise other session keys created either before or after this session key. Because compromising other session keys between V and RSU requires knowledge of $\{\boldsymbol{r}_{rsu}, \boldsymbol{r}_v, \boldsymbol{f}_{rsu}, \boldsymbol{f}_v\}$ or the ability to compute $\{\boldsymbol{k}_v, \boldsymbol{k}_{rsu}\}$ based on public information. However, achieving this requires solving the RLWE problem [10]. Thus, LBRAKA can resist against known session key attacks.

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

Resistance against privileged-insider attacks: The attack enables a trusted user of CA, referred to as a privileged-insider attacker, to misuse the credentials of vehicles or RSUs to initiate other types of attacks, such as impersonation and man-in-the-middle attacks [25]. However, CA is to help initialization, registration, update and robust tracing, and does not participate in mutual authentication and key agreement. In addition, although the privileged-insider attacker could know the credential of vehicle or RSU, it cannot misuse the credential to launch other attacks or engage in misbehaviors (e.g., impersonation attacks). Because the private key of vehicle or RSU should be employed with the credential to effectively carry out network activities (e.g., mutual authentication and key agreement), but the attacker has no idea about their private key. Therefore, the possibility of privileged-insider attacks is effectively eliminated.

Resistance against quantum attacks: Hybrid lattice-reduction attacks necessitate that an adversary solve the shortest vector problem (SVP). Specially, given a basis of lattice vectors, where the vectors are integers' fixed-length tuples, the goal is to determine a non-zero vector whose length is that of the shortest vector. A hybrid attack can be composed of a combination of hybrid lattice-reduction attacks and meet-in-the-middle (MiTM) attacks, which is crucial to assess the security of lattice based cryptographic protocols. The generic attacks require an adversary to recover a secret or private key according to the decryption errors' generation. The quantum MiTM attack allows an adversary to block all calibration signals, and transmit forged calibration signals, in order to interfere with the detector's activation timing calibration.

LBRAKA adopts the RLWE problem to construct an authenticated key agreement scheme, where lattice based quantum keys are employed. This enables LBRAKA to withstand various attacks (e.g., hybrid lattice-reduction attacks, generic attacks and quantum MiTM attacks) from classical and quantum computers [24].

TABLE IV
EXECUTION TIME OF CRYPTOGRAPHIC OPERATIONS

Description	Notation	Run time (ms)
The execution time of the multiplication of two polynomial elements in R_q	T_{mul0}	0.259
The execution time of the addition of two polynomial elements in R_q	T_{add0}	0.015
The execution time of H_0 where $H_0 : \{0,1\}^* \rightarrow D_{32}^n$	T_{h0}	0.004
The execution time of H_1 (i.e., sha3-256(*)) where $H_1 : \{0,1\}^* \rightarrow \{0,1\}^l$ and $l = 256$	T_{h1}	0.001
The execution time of H_2 where $H_2 : \{0,1\}^l \rightarrow R_q$	T_{h2}	0.006
The execution time of H_3 where $H_3 : \{0,1\}^* \rightarrow \chi_\beta$	T_{h3}	0.007
The execution time of $Cha(*)$ in R_q	T_{cha}	0.103
The execution time of $Mod_2(*)$ in R_q	T_{mod}	0.143

B. Performance Evaluation

In this part, we analyze LBRAKA's performance through simulations, and we compare LBRAKA with related works including [11], [21] in terms of computation overhead, communication overhead, storage overhead and unlinkability. As per the estimation done in [11], we set the system parameters, $n = 512$, and $q = 7557773$. As for the Gaussian sampling distribution, we fixed $\beta = 3.192$, and we applied the SHA3 function with an output of 256 bits for simulating hash operations.

1) Computation Overhead: In this part, we test the computation overhead of LBRAKA with related schemes [11], [21]. We simulate primitive cryptographic operations, which are implemented on a host machine equipped with Intel Xeon CPU E5-2678 v3 @2.50GHz and RAM @128GB by using the Openssl library, and NTL library with the option NTL_GMP_LIP = on (i.e., building NTL using the GNU Multi-Precision package) in C++. In particular, the OpenSSL library is a robust, commercial-grade, and full-featured toolkit for the transport layer security (TLS) and secure sockets layer (SSL) protocols. It also serves as a general-purpose cryptography library that supports a wide range of cryptographic algorithms and operations. The NTL library, on the other hand, is a high-performance, portable, and flexible C++ library for performing number theory operations, especially those concerning multi-precision arithmetics and polynomial arithmetics. Both libraries were utilized in our simulations to perform and analyze the primitive cryptographic operations, with the details of each operation's execution time documented in Table IV.

We mainly consider the time cost of LBRAKA during mutual authentication and key agreement, and compare it with that of Dabra et al.'s scheme [11], and Islam and Basu's scheme [21] in TABLE V and Fig. 6. Specifically, Fig. 6(a) depicts the time cost of vehicle-side computation while Fig. 6(b) shows the time cost of RSU-side computation. Note that we have not accounted for the time spent on pre-computations that can be performed in advance, in order to optimize the AKA execution time of LBRAKA and related schemes.

In the AKA phase of Dabra et al.'s scheme, the vehicle needs to perform three multiplication operations of two polynomial elements in R_q, five addition operations of two polynomial elements in R_q, eight hash operations, one characteristic function operation in R_q, and two Mod_2 function operations, so the time cost of the vehicle is $3T_{mul0} + 5T_{add0} + 6T_{h1} + 1T_{h2} + T_{h3} + T_{cha} + 2T_{mod} = 1.26$ms. The RSU needs to perform two multiplication operations of two polynomial elements in R_q, six addition operations of two polynomial elements in R_q, six hash operations, one characteristic function operation in R_q, and two Mod_2 function operations, so the time cost of the RSU is $2T_{mul0} + 6T_{add0} + 5T_{h1} + T_{h3} + T_{cha} + 2T_{mod} = 1.009$ms.

In the AKA phase of Islam and Basu's scheme, the vehicle/RSU needs to perform one multiplication operation of two polynomial elements in R_q, seven hash operations, one characteristic function operation in R_q, and two Mod_2 function operations, so the time cost of the vehicle/RSU is $T_{mul0} + 7T_{h1} + T_{cha} + 2T_{mod} = 0.655$ms.

In the AKA phase of our scheme, the vehicle/RSU needs to perform six multiplication operations of two polynomial elements in R_q, five addition operations of two polynomial elements in R_q, five hash operations, one characteristic function operation in R_q, and two Mod_2 function operations, so the time cost of the vehicle/RSU is $6T_{mul0} + 5T_{add0} + 2T_{h0} +$

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

TABLE V
COMPARISON OF COMPUTATION OVERHEAD

Scheme	V's Time Cost (ms)	RSU/CA's Time Cost (ms)	Total Cost (ms)
[11]	$3T_{mul0}+5T_{add0}+6T_{h1}+T_{h2}+T_{h3}+T_{cha}+2T_{mod}=1.26$	$2T_{mul0}+6T_{add0}+5T_{h1}+T_{h3}+T_{cha}+2T_{mod}=1.009$	2.269
[21]	$T_{mul0}+7T_{h1}+T_{cha}+2T_{mod}=0.655$	$T_{mul0}+7T_{h1}+T_{cha}+2T_{mod}=0.655$	1.131
Ours	$6T_{mul0}+5T_{add0}+2T_{h0}+3T_{h1}+T_{cha}+2T_{mod}=2.029$	$6T_{mul0}+5T_{add0}+2T_{h0}+3T_{h1}+T_{cha}+2T_{mod}=2.029$	4.058

TABLE VI
COMPARISON OF COMMUNICATION OVERHEAD (WHILE $n=512$, $q=7557773$)

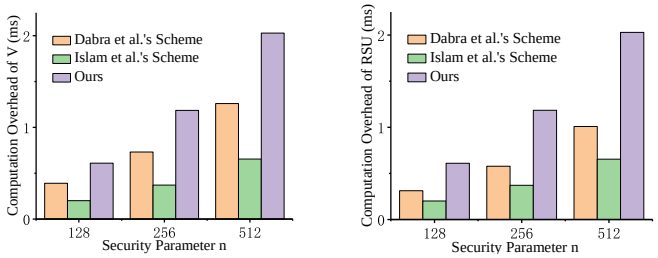
Scheme	Total Communication Overhead (bits)
[11]	$2l_p+3l_{h1}+2l_{cha}+l_{pid}=25376$
[21]	$4l_p+4l_{ts}+6l_{h1}+6l_{id}+2=49090$
Ours	$10l_p+2l_{ts}+2l_{h0}+2l_{h1}+2l_{cha}=120064$

TABLE VII
COMPARISON OF STORAGE OVERHEAD (WHILE $n=512$, $q=7557773$)

Scheme	V's Storage Cost (bits)	RSU's Storage Cost (bits)	CA's Storage Cost (bits)
[11]	$l_{h1}+l_{id}+l_{parameters}=321$	$2l_p+l_{id}+l_{parameters}=23617$	-
[21]	$2l_{h1}+l_q+l_{parameters}=568$	$2l_{h1}+l_q+l_{parameters}=568$	-
Ours	$8l_p+l_{ts}+2l_{h0}+l_{id}+l_{parameters}=94977$	$6l_p+l_{h0}+l_{id}+l_{parameters}=71041$	$4l_p+l_{h0}+l_{id}+l_{parameters}=47489$

$3T_{h1}+T_{cha}+2T_{mod}=2.029$ms.

As depicted in Fig. 6(a), the time cost of V in our scheme is higher than that in Dabra et al.'s scheme [11] and Islam and Basu's scheme [21], since V should perform additional signature verification on RSU for checking the legitimacy of RSU after pre-computation in our scheme. In 6(b), the time cost of RSU in our scheme exceeds that of other schemes, since RSU should perform additional signature creation for proving its legitimacy and additional signature verification on V for checking the legitimacy of V. In addition, RSUs generally have abundant computing resources in real scenarios. Although the time cost of V and RSU in our scheme is generally higher than that of other compared schemes, Dabra et al.'s scheme [11] only requires the AKA between users and a server, where no trust transfer from the server to RSU is involved and thus this scheme cannot be applied into VANETs. The AKA in Islam and Basu's scheme [21] involves an intermediate node (i.e., central authority), thus the authority suffers from high overheads and this scheme is not fit for VANETs. In addition, the schemes [11], [21] fail to provide the security properties of unlinkability, conditional privacy, public verification of tracing, and racing robustness, which can be offered by our scheme as shown in Table I. Consequently, although LBRAKA incurs a higher computation overhead, it offers more attractive security properties and is more fit for VANETs.



(a) Computation overhead of V (b) Computation overhead of RSU

Fig. 6. The comparison of computation overhead

2) Communication Overhead: In order to compare LBRAKA with related works [11], [21] in terms of communication overhead, we set the output size of hash function H_1 (SHA3-256) as $l_{h1}=256$ bits, the length of identity and pseudonym as $l_{id}=l_{pid}=32$ bits, the output size of hash function H_0 as $l_{h0}=32\log_2(n)+32=320$ bits, the length of timestamp as $l_{ts}=64$ bits, the length of each polynomial element as $l_p=n\cdot\log_2(q)=11776$ bits, and the output size of characteristic function $Cha(*)$ as $l_{cha}=n=512$ bits. In LBRAKA, the messages exchanged between the vehicle and RSU in three rounds of communication are $Q_v=\{\boldsymbol{a}_v,\boldsymbol{p}_v,\boldsymbol{x}_v,t_0,\boldsymbol{\sigma}_v\}$, $R_{rsu}=\{\boldsymbol{a}_{rsu},\boldsymbol{p}_{rsu},\boldsymbol{x}_{rsu},\boldsymbol{w}_{rsu},\boldsymbol{\kappa}_{rsu},t_1,\boldsymbol{\sigma}_{rsu}\}$, and $\{\boldsymbol{w}_v,\boldsymbol{\kappa}_v\}$, so the total communication overhead is $10l_p+2l_{ts}+2l_{h1}+2l_{h0}+2l_{cha}=120064$ bits. Table VI presents the comparative results of communication overhead. It is evident that the total communication overhead of LBRAKA exceeds that of Dabra et al.'s scheme [11] and Islam and Basu's scheme [21]. However, Dabra et al.'s scheme [11] and Islam and Basu's scheme [21] are not fit for VANETs, and fail to provide attractive properties. In summary, although the overhead of LBRAKA is higher than that of other schemes, it can provide many attractive security properties and is fit for VANETs, thus having potential usability.

3) Storage Overhead: We compare the storage cost of LBRAKA with related works [11], [21] in this part. According to the length assumption in communication overhead, we compute the size of the data stored by each entity (i.e., the vehicle, RSU, and CA).

In LBRAKA, the vehicle needs to store the public system parameters and $\{ID_v,T_v,\boldsymbol{a}_v,\boldsymbol{p}_v,\boldsymbol{s}_v,\boldsymbol{e}_v,Cred_v,\eta_v\}$, thus the

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

storage cost of the vehicle is $8l_p+l_{ts}+2l_{h0}+l_{id}+l_{parameters}$ = 94977 bits. In addition, RSU needs to store the public system parameters and $\{ID_{rsu}, \boldsymbol{a}_{rsu}, \boldsymbol{p}_{rsu}, \boldsymbol{s}_{rsu}, \boldsymbol{e}_{rsu}, Cred_{rsu}\}$, thus the storage cost of RSU is $6l_p + l_{h0} + l_{id} + l_{parameters}$ = 71041 bits. At last, CA needs to store the public system parameters and $\{ID_v, \boldsymbol{a}_v, \boldsymbol{p}_v, \delta_v\}$, thus the storage cost of CA is $4l_p + l_{h0} + l_{id} + l_{parameters}$ = 47489 bits.

The storage cost of related works can be calculated by using the same way. TABLE VII gives a comparison of LBRAKA and related works [11], [21] with respect to the storage cost. Although the storage overhead of LBRAKA is greater than that of the schemes proposed by Dabra et al. [11] and Islam and Basu [21], LBRAKA offers attractive security properties (e.g., unlinkability, conditional privacy, public verification of tracing, and tracing robustness), which are not ensured by the works [11], [21] and bring additional overheads.

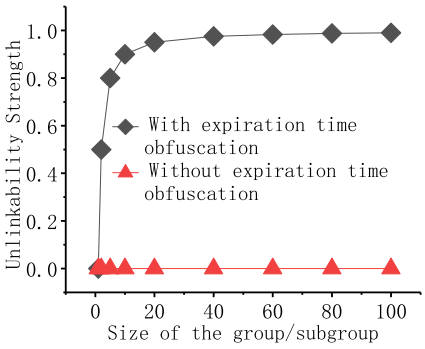


Fig. 7. Unlinkability of LBRAKA

4) Unlinkability: During registration, CA divides registered vehicles into a group or various subgroups with the same size, and assigns the same expiration time for credentials issued to each member of the same group or subgroup. When the credential of each vehicle within the group/subgroup is nearing its expiration, all the vehicles of the same group/subgroup update their public keys and credentials, thus the attacker is hard to link the new and old public keys and credentials of a vehicle according to the expiration time. However, the unlinkability strength is related to the size of the group/subgroup.

Fig. 7 depicts the relationship between the unlinkability strength and the size of the group/subgroup. When the expiration time obfuscation is not adopted, LBRAKA exhibits the weakest unlinkability of public key or credential. However, if the obfuscation is employed, the unlinkability strength increases as the size of the group/subgroup grows. Therefore, a proper size of the group/subgroup could be determined to achieve an expected unlinkability strength.

VI. Conclusion

In this paper, we proposed LBRAKA, a lattice-based robust authenticated key agreement scheme, which enables anonymous authentication and key negotiation between vehicles and RSUs. By assigning obfuscated expiration time for a vehicle's public key and certificate, their periodical updates and unlinkability are achieved. Through allowing the central authority and vehicles to commit to the vehicles' real identities and public keys during registration, public verification of tracing and tracing robustness are achieved. Performance analysis indicates that LBRAKA has more attractive security properties (e.g., public key unlinkability and robustness). Performance evaluation and comparison shows LBRAKA's efficacy and potential usability. However, LBRAKA allows a central authority to create and manage credentials, thus this scheme cannot be applied to heterogeneous vehicular networks, which are composed of different network/trust domains (e.g., mobile cellular network). Thus, in the future, quantum-resistant decentralized certificate management for authenticated key agreement needs to be studied in heterogeneous vehicular networks, which does not rely on a trusted third party.

References

[1] Z. Liu, L. Wan, J. Guo, F. Huang, X. Feng, L. Wang, and J. Ma, "Ppru: A privacy-preserving reputation updating scheme for cloud-assisted vehicular networks," *IEEE Transactions on Vehicular Technology*, 2023, doi: 10.1109/TVT.2023.3340723.

[2] J. Guo, X. Li, Z. Liu, J. Ma, C. Yang, J. Zhang, and D. Wu, "Trove: A context-awareness trust model for vanets using reinforcement learning," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6647–6662, 2020.

[3] S. Zhang, R. He, Y. Xiao, and Y. Liu, "A three-factor based trust model for anonymous bacon message in vanets," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 9, pp. 11 304–11 317, 2023.

[4] X. Feng, K. Cui, L. Wang, Z. Liu, and J. Ma, "Pbag: A privacy-preserving blockchain-based authentication protocol with global-updated commitment in iovs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 10, pp. 13 524–13 545, 2024.

[5] W. Jiang and X. Lv, "A distributed internet of vehicles data privacy protection method based on zero-knowledge proof and blockchain," *IEEE Transactions on Vehicular Technology*, 2023, doi: 10.1109/TVT.2023.3345272.

[6] Y. Liang, E. Luo, and Y. Liu, "Physically secure and conditional-privacy authenticated key agreement for vanets," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7914–7925, 2023.

[7] Q. Xie, Z. Ding, W. Tang, D. He, and X. Tan, "Provable secure and lightweight blockchain-based v2i handover authentication and v2v broadcast protocol for vanets," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 15 200–15 212, 2023.

[8] M. A. Saleem, X. Li, M. F. Ayub, S. Shamshad, F. Wu, and H. Abbas, "An efficient and physically secure privacy-preserving key-agreement protocol for vehicular ad-hoc network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 9, pp. 9940–9951, 2023.

[9] V. Nath, "Time and space complexies of shor's, grover's algorithms over classical algorithms," *VNSGU Journal of Science and Technology*, vol. 5, no. 1, pp. 13–20, 2016.

[10] Q. Feng, D. He, S. Zeadally, N. Kumar, and K. Liang, "Ideal lattice-based anonymous authentication protocol for mobile devices," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2775–2785, 2018.

[11] V. Dabra, A. Bala, and S. Kumari, "Lba-pake: lattice-based anonymous password authenticated key exchange for mobile devices," *IEEE Systems Journal*, vol. 15, no. 4, pp. 5067–5077, 2020.

[12] G. Xue, B. Wang, Q. Qu, and W. Zhang, "Efficient lattice-based authenticated key exchange based on key encapsulation mechanism and signature," *IET Information Security*, vol. 15, no. 1, pp. 107–116, 2021.

[13] S. Zhang, X. Du, and X. Liu, "A novel and quantum-resistant handover authentication protocol in iot environment," *Wireless Networks*, vol. 29, no. 6, pp. 2873–2890, 2023.

[14] R. Ding, C. Cheng, and Y. Qin, "Further analysis and improvements of a lattice-based anonymous pake scheme," *IEEE Systems Journal*, vol. 16, no. 3, pp. 5035–5043, 2022.

[15] D. Dharminder and K. P. Chandran, "Lwesm: learning with error based secure communication in mobile devices using fuzzy extractor," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 10, pp. 4089–4100, 2020.

[16] S. H. Islam, "Provably secure two-party authenticated key agreement protocol for post-quantum environments," *Journal of Information Security and Applications*, vol. 52, p. 102468, 2020.

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

[17] S. H. Islam and S. Zeadally, "Provably secure identity-based two-party authenticated key agreement protocol based on cbi-isis and bi-isis problems on lattices," *Journal of Information Security and Applications*, vol. 54, p. 102540, 2020.
[18] Q. Wang, D. Wang, C. Cheng, and D. He, "Quantum2fa: Efficient quantum-resistant two-factor authentication scheme for mobile devices," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 193–208, 2021.
[19] D. Dharminder, C. B. Reddy, A. K. Das, Y. Park, and S. S. Jamal, "Post-quantum lattice-based secure reconciliation enabled key agreement protocol for iot," *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2680–2692, 2022.
[20] S. Basu, K. Seyhan, S. H. Islam, and S. Akleylek, "Mlwr-2paka: a hybrid module learning with rounding-based authenticated key agreement protocol for two-party communication," *IEEE Systems Journal*, vol. 17, no. 4, pp. 6093–6103, 2023.
[21] S. H. Islam and S. Basu, "Pb-3paka: password-based three-party authenticated key agreement protocol for mobile devices in post-quantum environments," *Journal of Information Security and Applications*, vol. 63, p. 103026, 2021.
[22] G. Wei, K. Fan, K. Zhang, H. Wang, H. Li, and Y. Yang, "Quantum-safe lattice-based certificateless anonymous authenticated key agreement for internet of things," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 9213–9225, 2024.
[23] S. Prajapat, D. Gautam, P. Kumar, S. Jangirala, A. K. Das, Y. Park, and P. Lorenz, "Secure lattice-based aggregate signature scheme for vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 12 370–12 384, 2024.
[24] P. Bagchi, R. Maheshwari, B. Bera, A. K. Das, Y. Park, P. Lorenz, and D. K. Yau, "Public blockchain-envisioned security scheme using post quantum lattice-based aggregate signature for internet of drones applications," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 8, pp. 10 393–10 408, 2023.
[25] P. Bagchi, B. Bera, A. K. Das, S. Shetty, P. Vijayakumar, and M. Karuppiah, "Post quantum lattice-based secure framework using aggregate signature for ambient intelligence assisted blockchain-based iot applications," *IEEE Internet of Things Magazine*, vol. 6, no. 1, pp. 52–58, 2023.
[26] S. Mukherjee, D. S. Gupta, and G. Biswas, "An efficient and batch verifiable conditional privacy-preserving authentication scheme for vanets using lattice," *Computing*, vol. 101, no. 12, pp. 1763–1788, 2019.
[27] V. Dabra, A. Bala, and S. Kumari, "Flaw and amendment of a two-party authenticated key agreement protocol for post-quantum environments," *Journal of Information Security and Applications*, vol. 61, p. 102889, 2021.
[28] D. Dharminder, A. K. Das, S. Saha, B. Bera, and A. V. Vasilakos, "Post-quantum secure identity-based encryption scheme using random integer lattices for iot-enabled ai applications," *Security and Communication Networks*, vol. 2022, no. 1, p. 5498058, 2022.
[29] J. Hoffstein, "Ntru: A ring based public key cryptosystem," in *Proc. of ANTS*, 1998, pp. 267–288.
[30] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," in *Proc. of EUROCRYPT*, 2010, pp. 1–23.
[31] J. Ding, S. Alsayigh, J. Lancrenon, S. Rv, and M. Snook, "Provably secure password authenticated key exchange based on rlwe for the post-quantum world," in *Proc. of CT-RSA*, 2017, pp. 183–204.
[32] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on gaussian measures," *SIAM Journal on Computing*, vol. 37, no. 1, pp. 267–302, 2007.
[33] J. Zhang, Z. Zhang, J. Ding, M. Snook, and Ö. Dagdelen, "Authenticated key exchange from ideal lattices," in *Proc. of EUROCRYPT*, 2015, pp. 719–751.
[34] J. Ding, X. Xie, and X. Lin, "A simple provably secure key exchange scheme based on the learning with errors problem," *Cryptology ePrint Archive*, 2012.
[35] T. Güneysu, V. Lyubashevsky, and T. Pöppelmann, "Practical lattice-based cryptography: A signature scheme for embedded systems," in *Proc. of CHES*, 2012, pp. 530–547.
[36] G. Liu, Z. Yan, W. Feng, X. Jing, Y. Chen, and M. Atiquzzaman, "Sedid: An sgx-enabled decentralized intrusion detection framework for network trust evaluation," *Information Fusion*, vol. 70, pp. 100–114, 2021.
[37] J. Liang and Y. Kim, "Evolution of firewalls: Toward securer network using next generation firewall," in *Proc. of IEEE CCWC*, 2022.
[38] S. E. Quincozes, C. Albuquerque, D. Passos, and D. Mossé, "A survey on intrusion detection and prevention systems in digital substations," *Computer Networks*, vol. 184, p. 107679, 2021.
[39] S. Li, K. Xue, D. S. Wei, H. Yue, N. Yu, and P. Hong, "Secgrid: A secure and efficient sgx-enabled smart grid system with rich functionalities," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1318–1330, 2019.
[40] G. Liu, Z. Yan, D. Wang, H. Wang, and T. Li, "Deptvm: Decentralized pseudonym and trust value management for integrated networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 1, pp. 110–124, 2023.
[41] Y. Zhang, R. H. Deng, E. Bertino, and D. Zheng, "Robust and universal seamless handover authentication in 5g hetnets," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 858–874, 2019.
[42] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. of PKC*, 2005, pp. 65–84.
[43] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 357–366, 2015.

Gao Liu received the B.S. degree from the Department of Mathematics, Yibin University, Yibin, China, in 2013, the M.S. degree from the School of Mathematics and Computing Science, Guilin University Of Electronic Technology, Guilin, China, in 2016, and the Ph.D. degree from the School of Cyber Engineering, Xidian University, Xi'an, China, in 2022. He is currently an assistant professor of the College of Computer Science, Chongqing University, Chongqing, China. His research interest includes network security/trust measurement, anonymous authentication, data collection, blockchain, intrusion detection, federated learning and data aggregation.

Weiyang Li received the B.S. degree in data science and big data technology from Huazhong Agricultural University, Wuhan, China, in 2022. He is currently pursuing an M.S. degree in computer technology at Chongqing University, Chongqing, China. His research interests include physical layer security, intelligent autonomous systems, and UAV security.

Chengsheng Yuan received his B.S. degree from Nanjing University of Information Science and Technology, China, in 2014, and the Ph.D. degree from Nanjing University of Information Science and Technology, China, in 2019. From 2017 to 2019, he was a visiting student in the Department of Electrical and Computer Engineering at University of Winsor, Canada. From July 2019 to July 2020, he was also a research fellow in the Department of Electrical and Computer Engineering at University of Winsor, Canada. Currently, he is an associate professor in the College of Computer Science, Nanjing University of Information Science and Technology, China. His research interests include biometric liveness detection, information hiding, machine learning and AI security.

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2024.3515072

Ning Wang received the Ph.D. degree in information and communication engineering from Beijing University of Posts and Telecommunication in 2017. He was an Engineer at Huaxin Post and Telecommunications Consulting Design Company Ltd. from 2012 to 2013. He was with the Department of Electrical and Computer Engineering, George Mason University, as a PostDoctoral Scholar from 2017 to 2020. He is currently a Professor with the College of Computer Science, Chongqing University. His current research interests are in physical layer security, machine learning, RF fingerprinting, and cyber-physical systems security and privacy.

Chuan Ma received the B.S. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2013 and Ph.D. degree from the University of Sydney, Australia, in 2018. From 2018 to 2022, he worked as a lecturer in the Nanjing University of Science and Technology, and now he is a principal investigator at Zhejiang Lab. He has published more than 40 journal and conference papers, including a best paper in WCNC 2018, and a best paper award in IEEE Signal Processing Society 2022. His research interests include stochastic geometry, wireless caching networks and distributed machine learning, and now focuses on the big data analysis and privacy-preserving.

Nankun Mu received the B.S. degree from the School of Big Data & Software Engineering, Chongqing University, Chongqing, China, in 2011, and the Ph.D. degree from the College of Computer Science, Chongqing University, Chongqing, China, in 2015. He is currently an associate professor of the College of Computer Science, Chongqing University, Chongqing, China. His research interest includes AI security, data trading, privacy protection and intelligent optimization.

Zhiquan Liu received the B.S. degree from the School of Science, Xidian University, Xi'an, China, in 2012, and the Ph.D. degree from the School of Computer Science and Technology, Xidian University, Xi'an, China, in 2017. He is currently a full professor with the College of Cyber Security, Jinan University, Guangzhou, China. His current research focuses on security, trust, privacy, and intelligence in vehicular networks. He currently serves as the associate editors of IEEE Internet of Things Journal, IEEE Network, Computer Networks, etc. His homepage is https://www.zqliu.com.

Yining Liu received the B.S. degree in applied mathematics from Information Engineering University, Zhengzhou, China, in 1995, the M. S. degree in computer software and theory from the Huazhong University of Science and Technology, Wuhan, China, in 2003, and the Ph. D. degree in mathematics from Hubei University, Wuhan, in 2007. He is currently a professor with school of Computer and Information Security, Guilin University of Electronic Technology, Guilin, China. His research interests include the information security protocol and data privacy.

Tao Xiang received the BEng, MS and PhD degrees in computer science from Chongqing University, China, in 2003, 2005, and 2008, respectively. He is currently a professor with the College of Computer Science, Chongqing University. His research interests include multimedia security, cloud security, data privacy and cryptography. He has published more than 150 papers on international journals and conferences. He also served as a referee for numerous international journals and conferences.

Authorized licensed use limited to: CHONGQING UNIVERSITY. Downloaded on April 06,2025 at 08:09:52 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.