scientific reports

Check for updates

OPEN

Vehicular ad hoc networks verification scheme based on bilinear pairings and networks reverse fuzzy extraction

Zaid Ameen Abduljabbar[1,2,3]✉, Vincent Omollo Nyangaresi[4,5], Ahmed Ali Ahmed[6], Junchao Ma[2]✉, Mustafa A. Al Sibahee[6,7], Mohammed Abdulridha Hussain[1], Zaid Alaa Hussien[8], Ali Hasan Ali[9,10,11], Abdulla J. Y. Aldarwish[1] & Husam A. Neamah[12,13]

Vehicular Ad-Hoc Networks (VANETs) have facilitated the massive exchange of real-time traffic and weather conditions, which have helped prevent collisions, reduce accidents, and road congestions. This can effectively enhance driving safety and efficiency in technology-driven transportation systems. However, the transmission of massive and sensitive information across public wireless communication channels exposes the transmitted data to a myriad of privacy as well as security threats. Although past researches has developed many vehicular ad-hoc networks security preservation schemes, several of them are inefficient or susceptible to attacks. This work, introduces an approach that leverages reverse fuzzy extraction, bilinear pairing, and Physically Unclonable Function (PUF) to design an efficient and anonymity-preserving authentication scheme. We conduct an elaborate formal security analysis to demonstrate that the derived session key is secure. The semantic security analyses also demonstrate its resilience against typical VANET attacks such as impersonations, denial of service, and de-synchronization, instilling confidence in its effectiveness. Moreover, our approach incurs the lowest computational overheads at relatively low communication costs. Specifically, our protocol attains a 66.696% reduction in computation costs, and a 70% increment in the supported security functionalities.

Keywords Attacks, Bilinear pairing, Fuzzy extraction, Privacy, PUF, Security, VANET

The continued developments in communication and networking technologies have influenced the massive deployments of VANETs. This has seen enhancements in both transportation efficiency and safety. In VANETs, each vehicle acts as a node whose capabilities include data sensing, communication with other network entities as well as processing. This communication is accomplished using the Dedicated Short-Range Communication (DSRC) protocol[1]. In this protocol, messages related to safety are sent out by the vehicles after every 100 to 300 milliseconds. The transmitted messages may be about bad weather or traffic congestions, which can help in traffic navigation, management services, and collision avoidance. By exchanging real-time traffic density information and prevailing weather conditions, VANETs can help prevent collisions and road congestions,

[1]Department of Computer Science, College of Education for Pure Sciences, University of Basrah, Basrah 61004, Iraq. [2]College of Big Data and Internet, Shenzhen Technology University, Shenzhen 518118, China. [3]Department of Business Management, Al-imam University College, Balad 34011, Iraq. [4]Department of Computer Science and Software Engineering, Jaramogi Oginga Odinga University of Science & Technology, Bondo 40601, Kenya. [5]Department of Applied Electronics, Saveetha School of Engineering, SIMATS, Chennai 602105, Tamil Nadu, India. [6]Department of Management and Marketing, College of Industrial Management for Oil and Gas, Basrah University for Oil and Gas, Basrah 61004, Iraq. [7]National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen 518060, China. [8]Information Technology Department, Management Technical College, Southern Technical University, Basrah 61005, Iraq. [9]Department of Mathematics, College of Education for Pure Sciences, University of Basrah, Basrah 61004, Iraq. [10]Technical Engineering College, Al-Ayen University, Thi-Qar 64001, Iraq. [11]Institute of Mathematics, University of Debrecen, Pf. 400, Debrecen 4002, Hungary. [12]Department of Electrical Engineering and Mechatronics, Faculty of Engineering, University of Debrecen, Otemeto u.4-5, Debrecen 4028, Hungary. [13]College of Engineering, National University of Science and Technology, Dhi Qar 64001, Iraq. ✉email: zaid.ameen@uobasrah.edu.iq; majunchao@sztu.edu.cn

minimize accidents, and boost efficiency and road safety[2]. Therefore, these networks have found applications in traffic vigilance, trajectory predictions, and automated accident notifications, which enhance comfort and safety for drivers and passengers[3]. As explained in[4], a typical VANET comprises of Roadside Units (RSUs), Trusted or Certificate Authority (TA/CA), and vehicles fitted with Onboard Units (OBUs). As such, vehicle communication in these networks can be done with RSUs, nearby vehicles, or other supporting infrastructures.

Although VANETs offer reliable transmissions that boost safety and autonomy, establishing resilient and secure authentication at low overheads remains a mirage. For instance, during VANET communication, open and public wireless communication channels are utilized. This introduces numerous security and privacy threats[2,5]. Therefore, the transmitted messages can be modified, replayed, or deleted. In addition, the frequent transmission of safety-related messages can expose user's historical route, location, identity, and other sensitive information such as license plate information. Moreover, adversaries can launch attacks such as Man-in-the-Middle (MitM), sniffing, bogus message injection, fabrication, Advanced Persistent Threats (APTs), packet interception, replays and modifications[2]. This can potentially result in severe consequences such as fatal traffic accidents and unsafe driving, highlighting the urgent need for immediate action to address these threats. It is also possible for VANETs to experience sensor failures[6] which can result in unavailability. To mitigate this, fault[7] detection systems must be incorporated in VANETs.

All the above security and privacy issues point to the need for stringent authentication and privacy preservation. Therefore, message integrity, authenticity, and anonymity must be guaranteed. To this end, many security solutions have been developed based on techniques such as Elliptic Curve Cryptography (ECC), blockchain, and Public Key Cryptography (PKC). However, the majority of these schemes fail to support high functionality at low latencies and overheads[8]. Therefore, these computational complexities[9] must be minimized. Consequently, the development of an efficient authentication protocol is needed to support many security and privacy functionalities at low overheads.

Motivation

Many security techniques have been developed for safe message exchanges in VANETs. The majority of the protocols in this environment are based on bilinear pairings, Public Key Infrastructure (PKI), identity, and group signatures. However, some of these approaches require the maintenance of Certificate Revocation Lists (CRLs). Although this list facilitates the effective elimination of malevolent vehicles from the network, its distribution results in high communication costs. The network terminals must check through this list for each received signature, which results in high computation costs. Considering high mobility terminals centralization of certification services on servers can result in Denial of Service (DoS). Since some of the current security techniques require digital certificates and key pairs storage, they incur high storage space, among other values. Although identity-based schemes address these challenges, they have key escrow problems. In addition, some of them are based on pairings and map-to-point hashing operations. This renders them computationally extensive. Group signature-based techniques have been developed to offer conditional privacy so that members can anonymously sign the messages. Here, actual vehicle identity tracking can only be accomplished by the group manager. However, entire group reconstruction must occur whenever vehicles leave or join the group so as to maintain backward and forward key secrecy. Considering high-speed vehicles, these procedures are infeasible due to high overheads. In light of these challenges, it is clear that an efficient and secure authentication protocol is not just a solution, but a necessity to support VANET communications.

Contribution

In the wake of numerous insecurities in vehicular ad hoc networks and the inability of the current schemes to efficiently and effectively address these threats, our contributions include the following:

- Bilinear pairing operations are deployed to introduce stochasticity in generating intermediary security tokens. This helps thwart typical VANETs attacks such as privileged insider and impersonation.
- Physically unclonable functions are deployed to generate device fingerprints, which are then combined with reverse fuzzy extractors. Due to the unique nature of the generated challenge-response pairs, this combination is shown to help prevent attacks such as cloning.
- Our protocol's resilience is not just a claim but a result of rigorous formal security analysis. This analysis demonstrates the derived session key's robustness. In addition, semantic security analysis proves that our scheme effectively mitigates common threats in a VANET environment, including eavesdropping and packet replays.
- Our protocol outperforms others in performance. A comprehensive comparison with state-of-the-art schemes reveals that our protocol minimizes computation costs and maintains moderately low communication costs, making it a highly efficient choice. In specific, the proposed protocols attains a 66.696% reduction in computation costs, and a 70% increment in the supported security functionalities.

The rest of this paper is structured as follows: Part 2 discusses related works, while Part 3 describes the operation of our protocol. Conversely, Part 4 discusses our protocol's security analyses, while its performance evaluations are detailed in Part 5. Finally, Part 6 presents the conclusion and future research scopes in this domain.

Related work

The requirement for enhanced security in VANETs has led to the development of numerous security solutions over the recent past. For instance, a PKI-based scheme is presented in[10]. However, protocols based on PKI present some challenges in CRL management and have extensive storage, communication, and computation overheads. Therefore, several identity-based schemes have been developed in[11–20] to address these issues. Unfortunately,

identity-based protocols have key escrow issues[21], and hence, the compromise of the Key Generation Center (KGC) can result in the leakage of vehicle secret keys. In addition, the scheme in[20] preloads the TA's master key into each vehicle's Tamper Proof Device (TPD), exposing it to side-channeling attacks. To solve this challenge, the scheme in[11] utilizes the secure channel between the RSUs and TA to buffer dynamically refreshed master key into the RSUs' TPDs. However, this scheme incurs extensive computation overheads due to large numbers of map-to-point hashing operations. Motivated by this challenge, a pairing-free protocol is presented in[12], which updates the TPD's secret values in an online mode. However, its communication overheads are still high.

To provide anonymity, decentralization and immutability, numerous blockchain-based schemes have been developed in[22–37]. However, the on-chain operations render these schemes computationally extensive and hence inefficient[38]. In addition, these techniques may not offer support for key revocation[2]. Particularly, the schemes in[22,23,28] involve frequent vehicle interactions with the CA to obtain anonymous certificates, leading to increased computation and communication overheads. Similarly, the protocol in[31] offers unlinkability but is inefficient. On the other hand, the scheme in[29] supports strong privacy but ignores the traceability requisite of TA[2]. On its part, the scheme[33] is vulnerable to location-tracking attacks[32].

In order to offer mutual authentication, an authentication technique is developed in[39]. However, its bilinear pairings are time-consuming. To address this challenge, a full aggregation approach is adopted in[40] to reduce computation and bandwidth overheads. Similarly, a lightweight and anonymous authentication protocol is developed in[41] based on Elliptic Curve Cryptography (ECC). To prevent forgery attacks, certificate-less aggregate signature schemes are presented in[42–46]. However, the security analyses of the schemes in[40–46] fail to consider critical attacks such as DoS and de-synchronization. In order to minimize the number of secret tokens buffered in vehicles, a key derivation function is developed in[22]. However, with the surge in the number of vehicles, there is a corresponding increase in public keys search time, making the cost of tracing malicious vehicles enormous.

It is evident that insecurities in VANETs are serious issues requiring immediate attention from academia and industry. However, the majority of the techniques proposed so far either have security and privacy issues or are inefficient. For instance, it has been shown that most of the current PKC, bilinear pairing, and blockchain based techniques incur huge overheads. On the other hand, group signature-based techniques are susceptible to secret key disclosure and cannot withstand quantum computing attacks[47–49]. We show that our proposed approach effectively solves some of these challenges and holds great promise for the future of VANET security.

Proposed protocol

In this section, we present the mathematical principles critical to the design of our protocol. We also detail the key design principles of the proposed technique, attack model, network architecture, as well as different phases of our protocol. The notations we use throughout this work are detailed in Table 1.

Mathematical preliminaries

The proposed protocol is majorly based on one-way hashing functions, bilinear pairings, Physically Unclonable Function (PUF), and Reverse Fuzzy Extractor (RFE). The cryptographic hash function is characterized by the non-linearity, avalanche effect and non-reversibilty. As such, it is computationally infeasible to reverse the hash function. This implies that adversaries are unable to reconstruct the original input message from the output hash value. On the other hand, bilinear pairing's suitablity for cryptography is based on the difficulty of solving its Discrete Logarithm (DL), Computational Bilinear Diffie-Hellman (CBDH), and Decisional Bilinear Diffie-Hellman (DBDH) problems. The computational hardness of solving these problems implies that no polynomial-time algorithm can solve them and hence the security tokens protected this way can never be extracted by the

Symbol	Description
RA	Registration authority
V_i	Vehicle i
VID_i	Unique identity of V_i
RSU_j	Roadside Unit j
U_i	User i
δ, φ	RA's master secret and private keys respectively
P_k	RA's public key
$h(.)$	One-way hashing function
RID_j, PID_j	Unique and pseudo-identities for RSU_j respectively.
r_i, T_i	Random nonce i and timestamp i respectively
ΔT	Maximum permissible transmission delay
SR_j, PR_j	RSU_j's secret key and public key respectively
SV_i, PV_i	V_i's secret and public key respectively
HD	Helper data
l_j, l_i	Session keys derived at the RSU_j and V_i respectively
$\|\|$	Concatenation operation

Table 1. Notations.

adversaries. For instance, the difficulty of solving DBDH ensures that adversaries cannot discern the actual shared key from some random data.

Regaridng the PUFs, the physical variations in manufacturing is exploited to generate unique and unpredictable responses to challenges. The computational difficulty of PUFs lie in the practical unfeasibility of cloning or simulating their behavior. This is because the PUF responses (outputs) are hinged on minute, uncontrollable variations in physical devices during the manufacturing process. As such, it is impossible to algorithmically predict these response. Since the PUF properties are based on arbitrary, submicron imperfections, cloning, invasive probing, side-channeling and machine learning threats are mitigated. Due to the extremely large output space and unpredictable responses, PUFs are ideal in generating device-specific cryptographic keys that are extremely hard to guess or compute. On the other hand, reverse fuzzy extraction is deployed for secure extraction of keys from noisy data (such as biometrics or PUFs). Therefore, RFE ensures that only genuine noisy sources (original PUF or biometric) can recreate the secret keys. However, the adversaries are unable to reconstruct these secret keys (even when they possess partial information). The following sub-sections presents the mathematical formulations of these cryptographic techniques.

One-way hashing function

This function $h(.)$ takes as input information of arbitrary length l and computes a fixed-length hash value as output. For different input messages, $h(.)$ computes different hash values. Suppose that a polynomial time t adversary Ω is interested in establishing a hash collision. Therefore, Ω stochastically chooses messages $\{m_1, m_2\}$ in t. In essence, the adversarial objective is to use these messages to obtain $h(m_1) = h(m_2)$. The adversarial advantage of establishing this hash collision is represented as $Adv_{\Omega}^{hash}(t) = \Pr[(m_1, m_2) \leftarrow \Omega : (m_1 \neq m_2), (h(m_1) = h(m_2))]$. For this collision resistance, $Adv_{\Omega}^{hash}(t) < \varepsilon$.

Bilinear pairing operations

Suppose that ψ_1 is an additive cyclic group and ψ_2 is a multiplicative cyclic group. The order of both ψ_1 and ψ_2 is some large prime number p. In addition, we treat μ_1 and μ_2 as the generators of ψ_1 and ψ_2 respectively. The various definitions of bilinear pairing operations are mathematically expressed as follows:

(i) **Computability**: a proficient algorithm for deriving bilinear map b_m: $\psi_1 \times \psi_2 \rightarrow \psi_T$
(ii) **Computable isomorphism**: υ is a computable isomorphism from ψ_2 to ψ_1 given that $\psi_1 = \upsilon(\psi_2)$.
(iii) **Computable map**: b_m is a computable map b_m: $\psi_1 \times \psi_2 \rightarrow \psi_T$ and fulfills the following characteristics:

Bilinearity for all $a \in \psi_1, b \in \psi_2$, and $c, d \in \mathbb{Z}$ such that $b_m(a^c, b^d) = b_m(a, b)^{cd}$.
Non-degeneracy There exists μ_1 and μ_2 such that $b_m(\mu_1, \mu_2) \neq 1$.
Symmetry For every $a \in \psi_1$ and $b \in \psi_2$, $b_m(a, b) = b_m(b, a)$.

Physically unclonable function (PUF)

In PUFs, excitation by the stimulus results in unpredictable, unique, and reproducible output, representing a specific PUF's actual fingerprint. This serves to distinguish this PUF from the rest of the PUFs. In security, this stimulus and output denote the challenge and response, respectively. As such, the set of all feasible challenges and their resultant responses form the Challenge-Response Pairs (CRPs). During the manufacturing process, the unique and distinct assigned fingerprint is assigned to semiconductor pieces through some delay features in transistors and wires. Therefore, physical cloning of these semiconductors is very difficult; this renders PUFs a low-cost and yet cumbersome technology for integrated circuit tagging compared to bar codes, hard-printed serial numbers, and holograms, which are easily reproducible and hence insecure.

Suppose that C and R are sets of challenges and equivalent responses, respectively. Mathematically, a PUF is a function f:C→R. This function $f(.)$ should be reliable, unique, one-way, simple to assess, and cumbersome to replicate and predict. Mathematically, these features are defined as follows:

i) *One-way*: Taking κ as an identity function on C, function $f(.)$ is not invertible and hence $f \circ f^{-1} \neq \kappa$.
ii) *Reliability*: The same challenge should yield the same response at any given time. That is, $\forall c_i \in C, f_t(c_i) \cong f_{t^*}(c_i)$, where $t^* > t$.
iii) *Simple to assess*: This means that the temporal intricacy for assessing a PUF is constant. For any $c_i \in C$, function $f(.)$ is straightforward to assess. That is, $\forall c_i \in C, \ O(f(c_i)) \leq O(1)$.
iv) *Uniqueness*: The *PUF* function $f(.)$ should not have an equivalent function. That is, if $\forall c_i \in C, f(c_i) = f^*(c_i)$, then $f(.) = f^*(.)$.
v) *Cumbersome to replicate*: The temporal difficulty of building a PUF clone tends to be immeasurable. Therefore, its is not possible to clone or copy function $f(.)$. That is, $\forall c_i \in C, f(.) = f^*(.)$, then $O(f^*(.)) \approx \infty$.
vi) *Unpredictable*: In PUFs, the temporal difficulty of predicting a response based on some known set of CRPs tends to be infinite. Suppose that $\Gamma_{k \geq 0}$ is a set of CRPs and $L(.)$ is the prediction function. As such, the responses of $f(.)$ should be infeasible and cumbersome to predict or guess. That is, $\forall \Gamma_{k \geq 0} = \{(c_0, f(c_0)), \ldots, (c_k, f(c_k))\}$, $O(L\Gamma_k((c_{n>k}, f(c_{n>k})))) \approx \infty$.

Reverse fuzzy extractor (RFE)

In the proposed protocol, *RFE* utilizes the *FEGen* (.) method to produce helper data *HD* proficiently. In addition, the reproduction method *FERep* (.) is deployed to retrieve the real responses from the collected responses. The works in[50–52] give detailed descriptions on the fuzzy concept.

Suppose that C is the received challenge. Then, the response b is computed as $b = PUF(C)$. After that, *RFE* derives the helper data *HD* as $HD = FEGen(b)$. In addition, the verifier entity extracts the actual response from b and *HD* as $b^* = FERep(b, HD)$.

Key design principles

A practical and efficient authentication scheme should uphold the requirements below in the face of numerous security threats in VANETs.

Message authentication All transmitted messages must be mutually certified to determine their legitimacy and freshness. In this way, any network or device tampering can be detected.

Session key negotiation Upon the successful authentication of all network entities, session keys should be negotiated to encipher the collected data transmitted over the public Internet.

Integrity It should be cumbersome for the adversary to intercept and modify any message transmitted in VANETs.

Perfect key secrecy The attacker should be unable to use the captured current session key to compute session keys for the preceding and successive communications.

Confidentiality Attackers should be unable to discern the contents of all messages exchanged among the VANET entities.

Availability It should be difficult for the adversary to compromise the network or participating entities to the extent that the legitimate entities are denied access to the required services or data.

Non-repudiation and revocability All network entities should be incapable of denying having sent some messages in a VANET. Consequently, the registration authority should be capable of revoking the security tokens of any misbehaving entity.

Anonymity Based on the publicly exchanged messages in VANETs, adversaries should be incapable of discerning the actual identities of network entities.

Conditional traceability The registration authority should easily identify any malicious vehicle and eliminate it from the VANET environment.

Robustness against threats Threats such as de-synchronization, physical capture, MitM, DoS, replays, and eavesdropping can wreak havoc in VANETs. As such, an authentication scheme should be capable of resisting these attacks.

Lightweight The execution time of the authentication scheme should be very short to prevent long latencies during the authentication procedures.

Attack model

High volumes of sensitive data are exchanged over VANETs across public channels. Therefore, adversaries can launch a series of attacks towards these networks. Consequently, we assume that both passive and active attacks can be perpetrated in VANETs. As such, we deploy the widely implemented Canetti-Krawczyk (CK) model for adversary Ω. Specifically, Ω can intercept, delete, eavesdrop, modify, and insert malicious information into the transmission channel. In addition, Ω can capture the VANET network elements and retrieve the stored credentials. Moreover, all the intermediary session status and parameters can be compromised by Ω.

Network model

The major entities in our network architecture include vehicle i (V_i), Registration Authority (*RA*), and Roadside Unit j (RSU_j), as shown in Fig. 1. The *RA* is responsible for the system setup, generation of public values for other network entities, registration and secret keys generation for all users.

On the other hand, each V_i is equipped with an Onboard Unit (OBU_i) for data collection. The collected data is then shared with the RSU_j and other vehicles. The RSU_j, on its part, receives and forwards data to a number of vehicles. In terms of actual execution, the proposed protocol comprises of the system setup, registration, authentication and key agreement phases. Since the *RA* has enough computational and communication capabilities, it is designated to perform much of the intensive operations such as bilinear and fuzzy extractions.

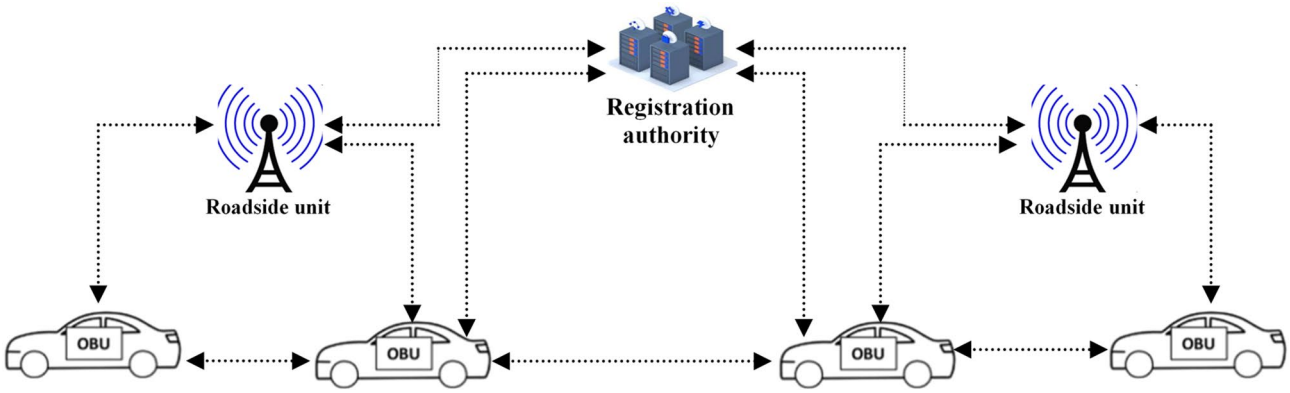

Fig. 1. Network model.

In addition, the system setup and registration phases are characterized with computationally intensive operations because these phases are executed only once. However, for the authentication and key setup phase which are performed frequently, few fuzzy extractions are performed. Therefore, the authentication and key agreement phase is charactrized by mostly lightweight one-way hashing and PUF operations.

System setup phase

The *RA* selects some random value $\delta \in Z_q^*$ as its master secret and $\varphi \in Z_q^*$ as its private key. Next, the registration authority computes its public key as $P_k = \mu_1^{\varphi}$ in addition to selecting h: $\{0,1\}^* \rightarrow Z_q^*$ as its cryptographic hash function. Finally, it publishes parameter set $\{\mu_1, \mu_2, \psi_1, \psi_2, \psi_T, q, b_m, P_k, h(.)\}$ as shown in Fig. 2.

Roadside unit registration

Prior to joining the network, it is required that RSU_j register itself to the *RA*. This registration is accomplished by executing the following procedures over highly secure communication channels, as evidenced in Fig. 2.

Step 1 The roadside unit RSU_j generates RID_j and designates it as its unique identity. This is followed by constructing the registration message $Req_i = \{RID_j\}$. Finally, Req_i is transmitted to *RA* across secured channels.

Step 2 After getting Req_i, the registration authority *RA* generates random nonce $r_1 \in Z_q^*$. Next, it computes RSU_j private key $SR_j = \mu_1^{(r_1+\phi)^{-1}}$ and its equivalent public key as $PR_j = \mu_1^{r_1}$.

Step 3 The registration authority *RA* generates some random challenge CR_j to authenticate the RSU_j. To prevent denial of service attacks, the *RA* generates challenge set $CR_j^{\complement} = CR_1, CR_2, CR_3, \ldots, CR_n$. Next, the *RA* composes the registration response message $Res_i = \{CR_j, CR_j^{\complement}\}$. At the end, Res_i is passed over to the roadside unit RSU_j.

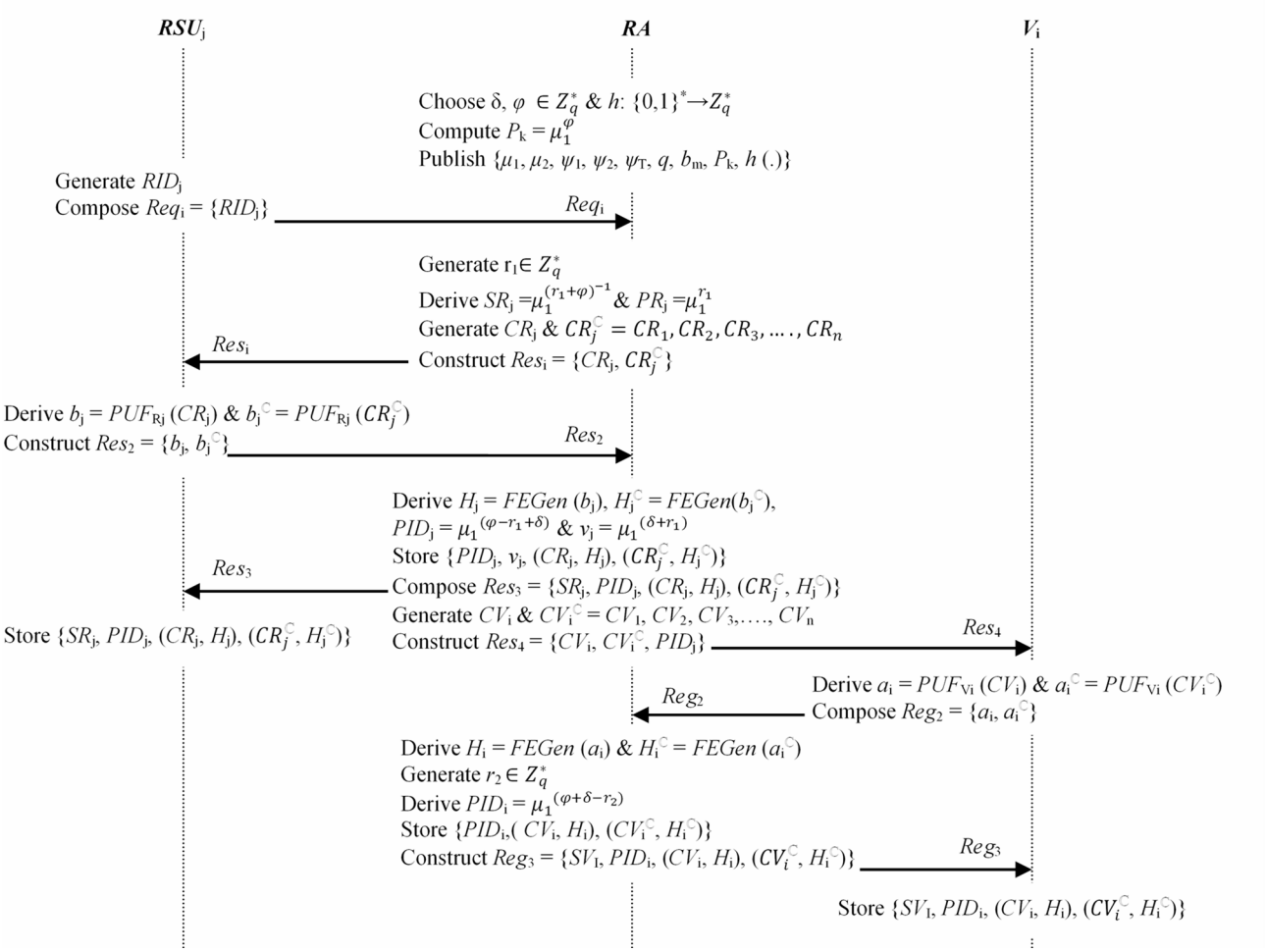

Fig. 2. System setup and registration.

Step 4 Upon receiving Res_i, RSU_j deploys the challenge set $\{CR_j, CR_j^{\mathbb{C}}\}$ as input to its PUF_{Rj} so as to generate PUF responses $b_j = PUF_{Rj}(CR_j)$ and $b_j^C = PUF_{Rj}(CR_j^{\mathbb{C}})$. Finally, it constructs response message $Res_2 = \{b_j, b_j^C\}$ and transmits it to the registration authority RA.

Step 5 After getting Res_2, the registration authority RA retrieves helper data as $H_j = FEGen(b_j)$ and $H_j^C = FEGen(b_j^C)$. Next, it computes pseudo-identity $PID_j = \mu_1^{(\phi - r_1 + \delta)}$ and $v_j = \mu_1^{(\delta + r_1)}$ as the parameter to track any misbehaving RSU_j. The RA then proceeds to store $\{PID_j, v_j, (CR_j, H_j), (CR_j^{\mathbb{C}}, H_j^C)\}$ in its database. Finally, it constructs message $Res_3 = \{SR_j, PID_j, (CR_j, H_j), (CR_j^{\mathbb{C}}, H_j^C)\}$, which is transmitted over to the RSU_j, where they are stored in memory.

Vehicle registration

The objective of this phase is to have all vehicles registered at the RSU_j in readiness for their actual deployment. This is facilitated by the execution of the following 3 steps which are summarized in Fig. 2.

Step 1 The roadside unit RSU_j generates random challenge CV_i as well as another set of challenges $CV_i^C = CV_1, CV_2, CV_3, \ldots, CV_n$ so as to thwart denial of service attacks. Next, it composes registration message $Res_4 = \{CV_i, CV_i^C, PID_j\}$. Finally, Res_4 is transmitted towards V_i across secured communication media.

Step 2 On getting Res_4, the challenge set $\{CV_i, CV_i^C\}$ is utilized as input to its PUF_{Vi} so as to generate responses $a_i = PUF_{Vi}(CV_i)$ and $a_i^C = PUF_{Vi}(CV_i^C)$. Next, it constructs message $Reg_2 = \{a_i, a_i^C\}$ and sends it over to the RSU_j.

Step 3 After getting the message Reg_2, the RSU_j derives helper data as $H_i = FEGen(a_i)$ and $H_i^C = FEGen(a_i^C)$. Next, it chooses some random nonce $r_2 \in Z_q^*$ that it utilizes to compute pseudo-identity $PID_i = \mu_1^{(\phi + \delta - r_2)}$ for V_i. At the end, it stores value set $\{PID_i, (CV_i, H_i), (CV_i^C, H_i^C)\}$ in its repository. Meanwhile, it composes $Reg_3 = \{SV_I, PID_i, (CV_i, H_i), (CV_i^{\mathbb{C}}, H_i^C)\}$ that is forwarded to V_i for storage in its memory.

Authentication and key agreement

Prior to exchanging data, the RSU_j and V_i must authenticate each other's identity. In addition, they must agree on a common session key to protect all the messages exchanged across insecure public communication media. This is accomplished through the following procedures, which are summarized in Fig. 3.

Step 1 The V_i chooses some random nonce r_3 and secret key SV_i that it uses to derive parameters $A_1 = \mu_1^{h(SV_i)+r_3)}$, $A_1^* = (\mu_1^{-r_3})^{-1}$, $A_2 = h(A_1||PID_j)$ and $A_3 = (PID_i||PID_j||A_1^*||A_2||T_1)$. After that, it composes the authentication message $Auth_1 = \{A_3\}$, which is transmitted to RA.

Step 2 After getting $Auth_1$, RA extracts timestamp T_1 and validates it against the current timestamp T_2. This is accomplished by confirming if $|T_2 - T_1| \le \Delta T$ so that session termination is executed if this condition does not hold. If not, RA calculates $A_1^{**} = A_2^* \times \mu_1^{h(SV_i)}$ and $A_2^* = h(A_1^{**}||PID_j)$. Next, it retrieves PID_j^* from its database and checks if $A_2^* = A_2$ as well as $PID_j^* = PID_j$. Basically, session termination must happen if this verification flops. If not, RA chooses (CV_i, a_i) and (CR_j, b_j).

Step 3 The RA generates some random nonces r_4 and r_5. Next, it derives $A_4 = \mu_1^{r_4}$, $A_4^* = \mu_1^{\phi + r_4}$, $A_5 = (\mu_1^{r_5})^{-1}$ and $A_5^* = \mu_1^{h(SR_j)+r_5)}$. Next, the registration authority calculates $B_1 = h(A_4||A_1||H_i)$, $B_2 = h(A_5||H_j)$, $B_3 = (CV_i||A_4^*||B_1||T_2)$ and $B_4 = (PID_j||CR_j||A_5^*||B_2||T_2)$. Finally, it constructs an authentication message $Auth_2 = \{B_3\}$ that is forwarded to the V_i. Similarly, it composes an authentication message $Auth_3 = \{B_4\}$ that is transmitted over to RSU_j.

Step 4 After getting the message $Auth_2$, the V_i extracts and verifies its timestamp by confirming whether $|T_3 - T_2| \le \Delta T$, where T_3 is the current timestamp. The authentication session is basically aborted when this freshness check flops. Otherwise, it computes $a_i = PUF_{Vi}(CV_i)$ and $H_i = FEGen(a_i)$. This is followed by the computation of $A_4^{**} = A_4^* \times \mu_1^{-\phi}$ and $B_1^* = h(A_4^{**}||A_1||H_i)$. Next, it verifies whether $B_1^* = B_1$, terminating the session if this check flops. Otherwise, it derives $CV_i^{new} = h(CV_i||a_i||A_4)$, $a_i^{new} = PUF_{Vi}(CV_i^{new})$, $H_i^{new} = FEGen(a_i^{new})$ and $PID_i^{new} = h(PID_i||H_i)$.

Step 5 The V_i stores value set $\{PID_i^{new}, (a_i^{new}, H_i^{new})\}$ in its memory. Afterwards, it derives $H_i^* = H_i^{new} \times \mu_1^{h(SV_i)}$, $B_5 = h(H_i^*||a_i||A_4)$ and $C_1 = (H_i||H_i^*||B_5||T_3)$. Ultimately, it constructs an authentication message $Auth_4 = \{C_1\}$, which is forwarded to RA. Similarly, upon getting the message $Auth_3$, the RSU_j extracts and validates its timestamp by checking if $|T_3 - T_2| \le \Delta T$, where T_3 is the current timestamp. Provided that this confirmation is unsuccessful, the session is halted. Otherwise, RSU_j derives the response for challenge CR_j as $b_j = PUF_{Rj}(CR_j)$ and computes helper data as $H_j = FEGen(b_j)$. Next, it computes $A_5^{**} = A_5^* \times (\mu_1^{h(SR_j)+r_5^*)})^{-1}$ and $B_2^* = h(A_5^{**}||H_j)$ before checking if $B_2^* = B_2$. In essence, session termination is activated whenever this authentication flops. Otherwise, it derives $CR_j^{new} = h(CR_j||b_j||A_5)$, $b_j^{new} = PUF_{Rj}(CR_j^{new})$ and $H_j^{new} = FEGen(b_j^{new})$. This is succeeded by the calculation of $PID_j^{new} = h(PID_j||H_j)$ before storing value set $\{PID_j^{new}, (CR_j^{new}, H_j^{new})\}$. In addition, the RSU_j calculates $H_j^* = H_j^{new} \times \mu_1^{h(SR_i)}$, $C_2 = (H_j^*||b_j||A_5)$ and $C_3 = (H_j||H_j^*||C_2||T_3)$ which is sent to the RA in authentication message $Auth_5 = \{C_3\}$.

Step 6 On receiving message $Auth_4$, the RA validates its timestamp as $|T_4 - T_3| \le \Delta T$, where T_4 is the present timestamp. On condition that T_3 is invalid, the session is aborted. Otherwise, it derives $a_i^* = FERep(a_i, H_i)$ and $B_5^* = h(H_i^*||a_i^*||A_4)$. This is followed by confirming whether $B_5^* = B_5$ so that the session is halted if this verification flops. Otherwise, the RA calculates $CV_i^{new} = h(CV_i||a_i||A_4)$ and $H_i^{new} = H_i^*(\mu_1^{h(SV_i)})^{-1}$ before storing parameter set $\{CV_i^{new}, H_i^{new}\}$ in its database. Similarly, RA verifies its timestamp T_3 in message $Auth_5$ as $|T_4 - T_3| \le \Delta T$ and aborts the session upon validation failure. Otherwise, it derives $b_j^* = FERep(b_j, H_j)$ and $C_2^* = (H_j^*||b_j^*||A_5)$ before verifying whether $C_2^* = C_2$. Fundamentally, session abortion takes place when this substantiation is unsuccessful. Otherwise, it derives $CR_j^{new} = h(CR_j||b_j||A_5)$ and $H_j^{new} = H_j^* \times (\mu_1^{h(SR_i)})^{-1}$ before storing parameter set $\{CR_j^{new}, H_j^{new}\}$ in its database.

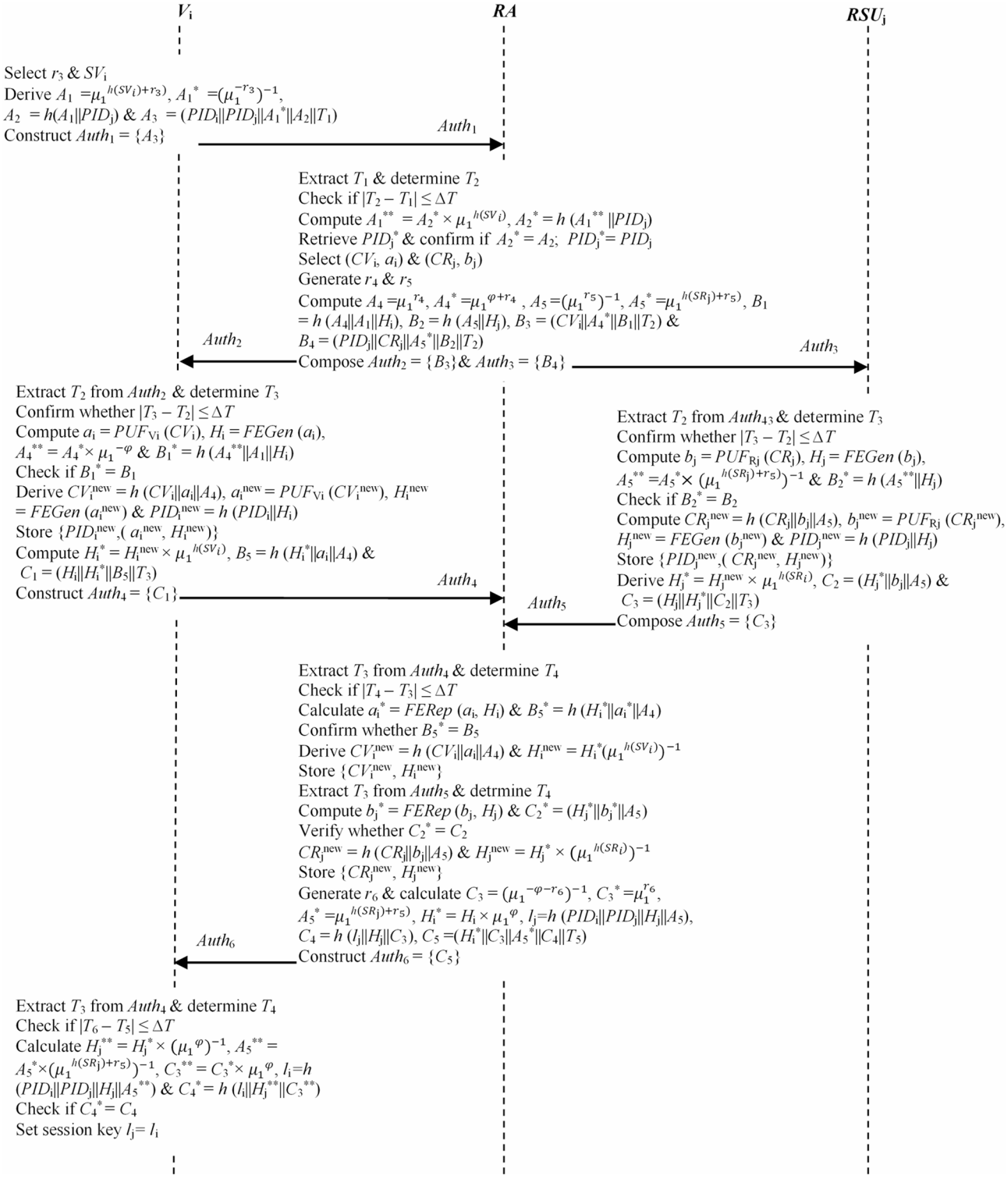

Fig. 3. Authentication and key agreement.

Step 7 The RSU_j chooses some random nonce r_6 and derives $C_3 = (\mu_1^{-\phi-r_6})^{-1}$, $C_3^* = \mu_1^{r_6}$, $A_5^* = \mu_1^{h(SR_j)+r_5)}$, $H_i^* = H_i \times \mu_1^{\phi}$, session key $l_j = h(PID_i||PID_j||H_j||A_5)$, $C_4 = h(l_j||H_j||C_3)$ and $C_5 = (H_i^*||C_3||A_5^*||C_4||T_5)$. Finally, it composes the authentication message $Auth_6 = \{C_5\}$, which is communicated over to V_i.

Step 8 After getting $Auth_6$, vehicle V_i authenticates its timestamp as $|T_6 - T_5| \leq \Delta T$, where T_6 is the current timestamp. On condition that T_5 is not fresh, the session is halted. Otherwise, it derives $H_j^{**} = H_j^* \times (\mu_1^{\phi})^{-1}$, $A_5^{**} = A_5^* \times (\mu_1^{h(SR_j)+r_5)})^{-1}$, $C_3^{**} = C_3^* \times \mu_1^{\phi}$, session key $l_i = h(PID_i||PID_j||H_j||A_5^{**})$ and $C_4^* = h(l_i||H_j^{**}||C_3^{**})$.

This is succeeded by the checking of whether $C_4^* = C_4$. Principally, authentication procedures are halted when these values are dissimilar. Otherwise, the RSU_j and V_i have effectively authenticated one another and can deploy the agreed session key $l_j = l_i$ to encipher the data exchanged over public channels.

Security analysis

This section's objective is to carry out our formal and semantic security analyses of our scheme. The detailed descriptions of the procedures involved are discussed below.

Formal security

The Real-Or-Random (ROR) model is frequently deployed to analyze the security and privacy posture of numerous authentication protocols. Therefore, we deploy this model to evaluate the provable security of the proposed protocol. Suppose that $\prod^{i}_{V_i}$, $\prod^{j}_{RSU_j}$, and $\prod^{k}_{RA}$ are the i^{th}, j^{th} and k^{th} instances of vehicle V_i, roadside unit RSU_j, and registration authority RA. In this scenario, $\prod^{i}_{V_i}$ is said to be partnered with $\prod^{j}_{RSU_j}$ if a section of the data exchanged between V_i and RSU_j is distinctive. Additionally, all the data exchanged are identified by a session identity (SID). Taking Ω as a polynomial time adversary, then instances $\prod^{i}_{V_i}$ and $\prod^{j}_{RSU_j}$ are said to be fresh on condition that the shared session key l setup between V_i and RSU_j has not been disclosed to Ω. The ROR model assumes that Ω can not only read but also alter the communicated data. Additionally, the adversary is capable of launching *Execute* (.), *Send* (.), *Corrupt_*V_i(.) and *Corrupt_*RSU_j (.) queries. The detailed explanation of these queries is as follows:

Execute ($\prod^{i}$, $\prod^{j}$): The adversary Ω utilizes this query to interrupt the communication process between vehicle V_i and roadside unit RSU_j. Therefore, this is a typical passive attack.

Send ($\prod^{j}$,Ē): Through this query, adversary Ω is able to communicate some information to the participating instance $\prod^{j}$. It is, therefore, a typical active attack.

*Corrupt_*V_i($\prod^{i}$): Using this query, adversary Ω can extract stored secrets from V_i's memory. Therefore, it represents a stolen or lost smart device attack.

*Corrupt_*RSU_j ($\prod^{j}$): Through this query, attacker Ω can compromise the RSU_j and retrieve the private parameters stored in its repository.

Test ($\prod^{i}$, $\prod^{j}$): Based on the ROR model's indistinguishability, the provable security of the negotiated key l between vehicle V_i and roadside unit RSU_j is tested. To accomplish this, unbiased coin c is tossed before the commencement of the game. Basically, the result of this query is determined by the output obtained upon the tossing of a fair coin. Suppose that l is fresh when this query is executed. Therefore, $\prod^{i}$ or $\prod^{j}$ can establish l when c is equal to 1 or a stochastic number satisfies $c = 0$. If not, a null value $\perp$ is returned.

Semantic security of l According to the ROR model, adversary Ω makes an attempt to distinguish between some random key and an instance's actual l. Therefore, Ω executes *Test* (.) queries on $\prod^{i}$ or $\prod^{j}$ with the results being checked against bit β. At the end of the game, adversary Ω has the capability of reviewing the envisaged bit β^* so as to triumph the game. Basically, adversary Ω wins the game on the condition that $\beta = \beta^*$. Let us denote the proposed scheme as θ and an event that Ω wins the game as Suc_{Ω}. Therefore, in polynomial time t, the adversarial advantage of Ω in breaking the security of θ is expressed as.

$$Adv_{\theta}^{\Omega}(t) = |2.\Pr[Suc_{\Omega}] - 1| \tag{1}$$

In this proof, Ω, RSU_j, V_i, and RA have access to both h (.) and PUF (.), which are deployed as random oracles. To demonstrate the security of θ, the features of Zipf's law on passwords, h (.) and PUF (.), are utilized.

Hypothesis 1 Let λ_1 and λ_2 denote RSU_j's and V_i's unique identities RID_j and VID_i, respectively. In addition, let Ω run against the proposed protocol θ. Specifically, Ω may attempt to compromise the provable security of θ by deriving key l as follows:

$$Adv_{\theta}^{\Omega}(t) \leq \frac{hsh^2}{|\mathbb{h}|} + \frac{puf^2}{|\mathbb{p}|} + 2\max(z_1.s^{z_2}, \frac{s}{2^{\lambda_1}}, \frac{s}{2^{\lambda_2}}) \tag{2}$$

where *hsh* and *puf* denote *hash* and *PUF* queries respectively, || and || are the range spaces for h(.) and *PUF* (.) functions respectively, s denote *Send* (.) queries while z_1 and z_2 are the Zipf's law parameters.

Proof The provable security of our approach involves the simulation of 5 games, $\mathbb{G}_0$ to $\mathbb{G}_4$. Here, $Suc_{\Omega i}$ represents the incident that Ω has correctly guessed bit β in these 5 games. The detailed descriptions of each of these games are given below.

$\mathbb{G}_0$: This is the actual attack against θ by Ω where the bit is chosen at the onset of this game. Therefore,

$$Adv_{\theta}^{\Omega}(t) = |2.\Pr[Suc_{\Omega 0}] - 1| \tag{3}$$

$\mathbb{G}_1$: This is an eavesdropping attack where Ω invokes *Execute* ($\prod^{i}$, $\prod^{j}$) to intercept the messages $Auth_1 = \{A_3\}$, $Auth_2 = \{B_3\}$, $Auth_3 = \{B_4\}$, $Auth_4 = \{C_1\}$, $Auth_5 = \{C_3\}$ and $Auth_6 = \{C_5\}$ exchanged over the public channels. After that, the *Test* (.) query is performed by Ω to establish if l is a genuine session key or an arbitrary key. Here, $A_3 = (PID_i||PID_j||A_1^*||A_2||T_1)$, $B_3 = (CV_i||A_4^*||B_1||T_2)$, $B_4 = (PID_j||CR_j||A_5^*||B_2||T_2)$, $C_1 = (H_i||H_i^*||B_5||T_3)$, $C_3 = (H_j||H_j^*||C_2||T_3)$ and $C_5 = (H_i^*||C_3||A_5^*||C_4||T_5)$. In our scheme, session keys are calculated as $l_j = h$

($PID_i||PID_j||H_j||A_5$) and l_i=h ($PID_i||PID_j||H_j||A_5^{**}$). Due to the deployed encapsulations, values PID_i, PID_j, and H_j are unknown to Ω. Since it is only the legitimate network entities that have access to these parameters, eavesdropping on the transmission channel does not increase the probability of Ω winning game $\mathbb{G}_1$. As such,

$$\Pr[Suc_{\Omega 1}] = \Pr[Suc_{\Omega 0}] \tag{4}$$

$\mathbb{G}_2$: In this game, *Send* (.) and *hash* (.) queries are simulated; hence, this is an active attack. Let us assume that Ω wants to modify the communicated messages before forwarding them to unsuspecting network entities. During the process of mutual verification and key setup, messages $Auth_1 = \{A_3\}$, $Auth_2 = \{B_3\}$, $Auth_3 = \{B_4\}$, $Auth_4 = \{C_1\}$, $Auth_5 = \{C_3\}$ and $Auth_6 = \{C_5\}$ are exchanged. Here, $A_3 = (PID_i||PID_j||A_1^*||A_2||T_1)$, $B_3 = (CV_i||A_4^*||B_1||T_2)$, $B_4 = (PID_j||CR_j||A_5^*||B_2||T_2)$, $C_1 = (H_i||H_i^*||B_5||T_3)$, $C_3 = (H_j||H_j^*||C_2||T_3)$, $C_5 = (H_i^*||C_3||A_5^*||C_4||T_5)$, $A_1^* = (\mu_1^{-r_3})^{-1}$, $A_2 = h(A_1||PID_j)$, $A_4^* = \mu_1^{\phi+r_4}$, $B_1 = h(A_4||A_1||H_i)$, $A_1 = \mu_1^{h(SV_i)+r_3)}$, $A_5^* = \mu_1^{h(SR_j)+r_5)}$,), $B_2 = h(A_5||H_j)$, $H_i = FEGen(a_i)$, $H_i^* = H_i^{new} \times \mu_1^{h(SV_i)}$, $B_5 = h(H_i^*||a_i||A_4)$, $A_4 = \mu_1^{r_4}$, $H_j = FEGen(b_j)$, $H_j^* = H_j^{new} \times \mu_1^{h(SR_i)}$, $C_2 = (H_j^*||b_j||A_5)$, $A_5 = (\mu_1^{r_5})^{-1}$ and $B_1 = h(A_4||A_1||H_i)$. Evidently, all the exchanged messages incorporate random secrets such as nonces and timestamps. Since the deployed one-way hashing function is collision-resistant, any attempt by Ω to perform *Send* (.) queries with the help of h (.) cannot yield any hash collisions. Based on the birthday paradox,

$$|\Pr[Suc_{\Omega 2}] - \Pr[Suc_{\Omega 1}] \leq \frac{hsh^2}{(2|\mathbb{h}|)} \tag{5}$$

$\mathbb{G}_3$: This is an extension of $\mathbb{G}_2$ in that both *Send* (.) and *PUF* (.) queries are executed in this game. Based on the arguments in $\mathbb{G}_2$ and the secure *PUF* properties,

$$|\Pr[Suc_{\Omega 3}] - \Pr[Suc_{\Omega 2}] \leq \frac{puf^2}{(2|\mathbb{p}|)} \tag{6}$$

$\mathbb{G}_4$: This game entails both *Corrupt_V*$_i$($\prod^i$) and *Corrupt_RSU*$_j$ ($\prod^j$) queries through which parameters {SR_j, PID_j, (CR_j, H_j), ($CR_j^{\mathbb{C}}$, $H_j^{\mathbb{C}}$)} and {PID_i,(CV_i, H_i), ($CV_i^{\mathbb{C}}$, $H_i^{\mathbb{C}}$)} stored in RSU_j as well as {SV_I, PID_i, (CV_i, H_i), ($CV_i^{\mathbb{C}}$, $H_i^{\mathbb{C}}$)} stored in V_i are extracted. Using the *PUF*, the probability of Ω correctly guessing λ_1 and λ_2 is given by $(2^{\lambda_1})^{-1}$ and $(2^{\lambda_2})^{-1}$, respectively. To predict low entropy passwords, adversary Ω may utilize Zipf's law on passwords. Based on arbitrary guessing attacks, the probability of Ω obtaining a correct guess is more than 0.5 when $s = 10^7$ or 10^8[53]. However, this probability is more than 0.5 when $s \leq 10^6$ in targeted guessing attacks utilizing user data. This puts some constraints on the password inputs into the proposed protocol. Devoid of these guessing attacks, $\mathbb{G}_3$ and $\mathbb{G}_4$ are indistinguishable and hence,

$$|\Pr[Suc_{\Omega 4}] - \Pr[Suc_{\Omega 3}] \leq \max(z_1.s^{z_2}, \frac{s}{2^{\lambda_1}}, \frac{s}{2^{\lambda_2}}) \tag{7}$$

To win $\mathbb{G}_4$, adversary Ω needs to guess bit β and execute the *Test* (.) query to establish whether it is the correct guess. As such,

$$\Pr[Suc_{\Omega 4}] = \frac{1}{2} \tag{8}$$

Combining Eqs. (3), (4), and (8) yield the following relation.

$$\frac{1}{2}Adv_{\theta}^{\Omega}(t) = \left|\Pr[Suc_{\Omega 0}] - \frac{1}{2}\right| = |\Pr[Suc_{\Omega 1}] - \frac{1}{2}| = |\Pr[Suc_{\Omega 1}] - \Pr[Suc_{\Omega 4}]| \tag{9}$$

The application of the triangular inequality in Eq. (5) to (7) yields the following:

$$\begin{aligned} |\Pr[Suc_{\Omega 1}] - \Pr[Suc_{\Omega 4}]| \leq\ & |\Pr[Suc_{\Omega 1}] - \Pr[Suc_{\Omega 3}]| + |\Pr[Suc_{\Omega 3}] - \Pr[Suc_{\Omega 4}]| \leq |\Pr[Suc_{\Omega 1}] - \Pr[Suc_{\Omega 2}] \\ & + |\Pr[Suc_{\Omega 2}] - \Pr[Suc_{\Omega 3}]| + |\Pr[Suc_{\Omega 3}] - \Pr[Suc_{\Omega 4}]| \leq \frac{hsh^2}{(2|\mathbb{h}|)} \\ & + \frac{puf^2}{(2|\mathbb{p}|)} + \max(z_1.s^{z_2}, \frac{s}{2^{\lambda_1}}, \frac{s}{2^{\lambda_2}}) \end{aligned} \tag{10}$$

In the end, Eqs. (9) and (10) are solved to obtain the following:

$$Adv_{\theta}^{\Omega}(t) \leq \frac{hsh^2}{|\mathbb{h}|} + \frac{puf^2}{|\mathbb{p}|} + 2\max(z_1.s^{z_2}, \frac{s}{2^{\lambda_1}}, \frac{s}{2^{\lambda_2}}) \tag{11}$$

This effectively completes the proof; hence, the proposed protocol is semantically secure.

Semantic security analysis

In an effort to show that our scheme resists typical VANET threats, we formulate and proof a number of theorems. The specific details of these theorems are described below.

Theorem 1 *Conditional traceability, non-repudiation, and revocation are provided.*

Proof Suppose that a dispute has arisen regarding some data, and the *RA* is interested in tracking and eliminating such V_i and RSU_j from the network. For instance, if *RA* receives bogus $Auth_5 = \{C_3\}$ from RSU_j, it can decrypt it to obtain value set $\{H_j, H_j^*, C_2, T_3\}$. During registration, *RA* stores $\{PID_j, v_j, (CR_j, H_j), (CR_j^C, H_j^C)\}$ in its repository, while V_i stores $\{SV_I, PID_i, (CV_i, H_i), (CV_i^C, H_i^C)\}$ in its memory. As such, *RA* can easily associates H_j to a particular RSU_j pseudo-identity PID_j. Similarly, upon receiving a bogus message $Auth_4 = \{C_1\}$ from V_i, *RA* can decipher it to obtain parameter set $\{H_i, H_i^*, B_5, T_3\}$. Afterwards, *RA* can easily associate H_i to a particular V_i pseudo-identity received in message $Auth_1 = \{A_3\}$, where $A_3 = (PID_i||PID_j||A_1^*||A_2||T_1)$. This association is then followed by the revocation of all secret parameters linked to the malicious V_i and RSU_j from the network.

Theorem 2 *Mutual authentication is successfully attained.*

Proof In this scheme, all the network elements are authenticated prior to sharing the collected data. For instance, upon getting the message $Auth_1 = \{A_3\}$ from V_i, the *RA* derives $A_2^* = h(A_1^{**}||PID_j)$ and retrieves PID_j^* from its database. It then checks if $A_2^* = A_2$ and $PID_j^* = PID_j$. Similarly, upon receiving $Auth_2 = \{B_3\}$ from *RA*, the V_i computes $B_1^* = h(A_4^{**}||A_1||H_i)$ and verifies whether $B_1^* = B_1$. On the other hand, after receiving message $Auth_3 = \{B_4\}$ from *RA*, RSU_j derives $B_2^* = h(A_5^{**}||H_j)$ before confirming whether $B_2^* = B_2$. Similarly, upon getting message $Auth_4 = \{C_1\}$ from V_i, the *RA* calculates $B_5^* = h(H_i^*||a_i^*||A_4)$ and checks if $B_5^* = B_5$. Conversely, after getting $Auth_5 = \{C_3\}$ sent by RSU_j, the registration authority *RA* derives $C_2^* = (H_j^*||b_j^*||A_5)$ before verifying whether $C_2^* = C_2$. Additionally, message $Auth_6 = \{C_5\}$ sent from RSU_j is verified by calculating $C_4^* = h(l_i||H_j^{**}||C_3^{**})$ and confirming whether $C_4^* = C_4$. For all these verification instances, session termination is invoked upon validation flop.

Theorem 3 *This protocol preserves confidentiality and sensor physical security.*

Proof To confirm the physical security of V_i's *OBU*, the RSU_j chooses some random nonce r_7 and derives values $D_1 = (\mu_1^{h(SR_j)-r7)})^{-1}$ and $D_1^* = \mu_1^{-h(SR_j)}$. Next, it selects pair $\{CV_i, a_i\}$ from its memory and derives values $D_2 = h(a_i||D_1)$ and $D_3 = (CV_i||D_1^*||D_2||T_7)$. It then sends D_3 towards the V_i. After receiving D_3, the V_i validates timestamp by checking if $|T_8 - T_7| \le \Delta T$, where T_8 is the current timestamp at V_i. Provided that T_7 is fresh, the V_i derives $a_i = PUF_{Vi}(CV_i)$ and $H_i = FEGen(a_i)$, $D_1^{**} = D_1^* \times \mu_1^{r7}$ and $D_2^* = h(a_i||D_1^{**})$. Next, it checks if $D_2^* = D_2$, and provided this verification is successful, V_i chooses random nonce r_8. Afterwards, it derives $D_4 = \mu_1^{r_8 - h(SR_j)}$, $D_4^* = (\mu_1^{h(SR_j)})^{-1}$, $CV_i^{new} = h(CV_i||D_4||a_i)$, $a_i^{new} = PUF_{Vi}(CV_i^{new})$, $H_i^{new} = FEGen(a_i^{new})$ and $H_i^* = H_i^{new} \times \mu_1^{h(SR_j)}$. This is succeeded by the calculation of $D_5 = h(H_i^*||D_4||a_i)$, transient session key $l_T = h(D_1||D_4||H_i)$ and value $E_1 = (H_i^*||D_4^*||D_5||H_i||T_8)$. After that, E_1 is forwarded to the RSU_j, which then proceeds to validate its timestamp T_8 by confirming whether $|T_9 - T_8| \le \Delta T$, where T_9 is the current timestamp at RSU_j. Provided that T_8 is fresh, RSU_j computes $a_i^* = FERep(a_i, H_i)$, $D_4^{**} = D_4^* \times \mu_1^{r8}$ and $D_5^* = h(H_i^*||D_4^{**}||a_i)$. Next, it confirms whether $D_5^* = D_5$ and provided that this condition holds, the RSU_j calculates $CV_i^{new} = h(CV_i||D_4^{**}||a_i)$, $H_i^{new} = H_i^* \times (\mu_1^{h(SR_j)})^{-1}$ and transient session key $l_T^* = h(D_1||D_4||H_i)$. In the end, the RSU_j stores pair $\{CV_i^{new}, H_i^{new}\}$ in its memory for subsequent communication. The deployment of *PUF* challenges and responses instead of buffering user and device-specific secret parameters in memory protects against cloning, physical attacks, and subsequent side-channeling through power analysis. On the other hand, the negotiated transient session key enciphers exchanged messages so as to preserve their confidentiality.

Theorem 4 *Session key for data protection is negotiated.*

Proof In our proposed approach, the roadside unit RSU_j and vehicle V_i compute session keys that are deployed for payload protection. For instance, RSU_j calculates the session key as $l_j = h(PID_j||PID_j||H_j||A_5)$, where $H_j = FEGen(b_j)$ and $A_5 = (\mu_1^{r5})^{-1}$. Similarly, V_i computes the session key as $l_i = h(PID_i||PID_j||H_j||A_5^{**})$, where $A_5^{**} = A_5^* \times (\mu_1^{h(SR_j)+r5)})^{-1}$. During the actual message exchanges, transient session keys are also derived. Whereas V_i computes $l_T = h(D_1||D_4||H_i)$, the RSU_j calculates $l_T^* = h(D_1||D_4||H_i)$, where $D_1 = (\mu_1^{h(SR_j)-r7)})^{-1}$, $D_4 = \mu_1^{r_8 - h(SR_j)}$ and $H_i = FEGen(a_i)$. These session keys are then deployed to uphold communication confidentiality.

Theorem 5 *This protocol attains perfect key secrecy.*

Proof In accordance with *Theorem 4*, the session keys are derived as $l_j = h(PID_j||PID_j||H_j||A_5)$, $l_i = h(PID_i||PID_j||H_j||A_5^{**})$, $l_T = h(D_1||D_4||H_i)$ and $l_T^* = h(D_1||D_4||H_i)$. Here, PID_i and PID_j are the pseudo-identities for V_i and RSU_j, respectively. On the other hand, $H_j = FEGen(b_j)$, $A_5 = (\mu_1^{r5})^{-1}$, $A_5^{**} = A_5^* \times (\mu_1^{h(SR_j)+r5)})^{-1}$, $D_1 = (\mu_1^{h(SR_j)-r7)})^{-1}$, $D_4 = \mu_1^{r_8 - h(SR_j)}$ and $H_i = FEGen(a_i)$. Incorporating random nonces such as r_5, r_7 and r_8 renders the derived keys one-time. In addition, RSU_j's private key SR_j is required to successfully derive some of these keys. Therefore, it is difficult for adversaries to determine past and subsequent session keys hinged on the current session keys.

Theorem 6 *Anonymous communication is preserved.*

Proof In the process of mutually verifying each other and setting up the session key, messages $Auth_1 = \{A_3\}$, $Auth_2 = \{B_3\}$, $Auth_3 = \{B_4\}$, $Auth_4 = \{C_1\}$, $Auth_5 = \{C_3\}$ and $Auth_6 = \{C_5\}$ are exchanged. Here, $A_3 = (PID_i||PID_j||A_1^*||A_2||T_1)$, $B_3 = (CV_i||A_4^*||B_1||T_2)$, $B_4 = (PID_j||CR_j||A_5^*||B_2||T_2)$, $C_1 = (H_i||H_i^*||B_5||T_3)$, $C_3 = (H_j||H_j^*||C_2||T_3)$ and $C_5 = (H_i^*||C_3||A_5^*||C_4||T_5)$. Evidently, the distinctive device identities, such as RID_j, are never incorporated into these messages. Therefore, an adversary cannot discern a unique device identity based on the exchanged messages.

Theorem 7 *This scheme thwarts replay attacks.*

Proof Timestamps are incorporated in all publicly transmitted messages to prevent packet replays. For instance, message $Auth_1$, $Auth_2$, $Auth_3$, $Auth_4$, $Auth_5$ and $Auth_6$ incorporate timestamps T_1, T_2, T_2, T_3, T_3 and T_5 respectively. At the receiver side, these timestamps are validated to thwart any message replays. For example, when receiving $Auth_1$, RA extracts the timestamp T_1 and validates it against the current timestamp T_2. This is accomplished by confirming if $|T_2 - T_1| \leq \Delta T$. Similarly, after getting message $Auth_2$, the V_i extracts and verifies its timestamp by confirming whether $|T_3 - T_2| \leq \Delta T$. On the other hand, upon getting the message $Auth_3$, the RSU_j extracts and validates its timestamp by checking if $|T_3 - T_2| \leq \Delta T$. Similarly, the RA validates $Auth_4$'s timestamp T_3 as $|T_4 - T_3| \leq \Delta T$ while it verifies $Auth_5$'s timestamp T_3 in message $Auth_5$ as $|T_4 - T_3| \leq \Delta T$. Finally, upon receiving message $Auth_6$, the V_i authenticates its timestamp T_5 as $|T_6 - T_5| \leq \Delta T$. In all these instances, the authentication sessions are terminated for invalid timestamps.

Theorem 8 *Denial of service and de-synchronization attacks are mitigated.*

Proof Let us assume that an adversary wants to de-synchronize roadside unit RSU_j using challenge CR_j. Similarly, let us assume that an attacker wants to de-synchronize V_i through challenge CV_i. To address these two cases, the RA generates a challenge set $CR_j^{\mathbb{C}} = CR_1, CR_2, CR_3, \ldots ., CR_n$ that can be deployable upon compromise of CR_j. Similarly, RSU_j generates a set of challenges $CV_i^{\mathrm{C}} = CV_1, CV_2, CV_3, \ldots, CV_n$ that are deployable upon CV_i compromise. As such, the authentication procedures will still be executed even when challenges CR_j and CV_i have been compromised. This effectively thwarts any denial of service and de-synchronization attacks that may use these two challenges as vectors.

Theorem 9 *This protocol can withstand eavesdropping and MitM threats.*

Proof Let us assume that an adversary can potentially intercept all the data exchanged over public channels. After that, an attempt is made to modify these messages so as to mislead unsuspecting receivers. These messages include $Auth_1 = \{A_3\}$, $Auth_2 = \{B_3\}$, $Auth_3 = \{B_4\}$, $Auth_4 = \{C_1\}$, $Auth_5 = \{C_3\}$ and $Auth_6 = \{C_5\}$ are exchanged. Here, $A_3 = (PID_i||PID_j||A_1^*||A_2||T_1)$, $B_3 = (CV_i||A_4^*||B_1||T_2)$, $B_4 = (PID_j||CR_j||A_5^*||B_2||T_2)$, $C_1 = (H_i||H_i^*||B_5||T_3)$, $C_3 = (H_j||H_j^*||C_2||T_3)$, $C_5 = (H_i^*||C_3||A_5^*||C_4||T_5)$, $A_1^* = (\mu_1^{-r_3})^{-1}$, $A_2 = h(A_1||PID_j)$, $A_4^* = \mu_1^{\phi + r_4}$, $B_1 = h(A_4||A_1||H_i)$, $A_1 = \mu_1^{h(SV_i)+r_3}$, $A_5^* = \mu_1^{h(SR_j)+r_5}$,), $B_2 = h(A_5||H_j)$, $H_i = FEGen(a_i)$, $H_i^* = H_i^{\text{new}} \times \mu_1^{h(SV_i)}$, $B_5 = h(H_i^*||a_i||A_4)$, $A_4 = \mu_1^{r_4}$, $H_j = FEGen(b_j)$, $H_j^* = H_j^{\text{new}} \times \mu_1^{h(SR_i)}$, $C_2 = (H_j^*||b_j||A_5)$, $A_5 = (\mu_1^{r_5})^{-1}$ and $B_1 = h(A_4||A_1||H_i)$. Clearly, any successful modification requires knowledge of random nonces (such as r_3, r_4, and r_5) and secret keys (such as SR_j and SV_i). Since these values cannot be eavesdropped across the public internet, they are unavailable to the adversary. As such, both eavesdropping and MitM attacks flop.

Theorem 10 *Our approach mitigates impersonation and privileged insider attacks.*

Proof Here, we assume a privileged user can access RA's database. Therefore, he/she can retrieve stored parameter set $\{PID_j, v_j, (CR_j, H_j), (CR_j^{\mathbb{C}}, H_j^{\mathrm{C}})\}$. After that, an attempt is made to impersonate all the exchanged messages $Auth_1 = \{A_3\}$, $Auth_2 = \{B_3\}$, $Auth_3 = \{B_4\}$, $Auth_4 = \{C_1\}$, $Auth_5 = \{C_3\}$ and $Auth_6 = \{C_5\}$. However, in accordance with *Theorem 9*, an adversary still needs random nonces and secret keys to construct valid messages. As such, the retrieved parameters cannot enable the adversary to impersonate the network entities.

Theorem 11 *The proposed protocol is highly scalable.*

Proof During the registration phase, the RA stores $\{PID_j, v_j, (CR_j, H_j), (CR_j^{\mathbb{C}}, H_j^{\mathrm{C}})\}$ in its database, while the RSU_j stores $\{SR_j, PID_j, (CR_j, H_j), (CR_j^{\mathbb{C}}, H_j^{\mathrm{C}})\}$ and $\{PID_i, (CV_i, H_i), (CV_i^{\mathrm{C}}, H_i^{\mathrm{C}})\}$. On its part, the V_i stores $\{SV_i, PID_i, (CV_i, H_i), (CV_i^{\mathbb{C}}, H_i^{\mathbb{C}})\}$. As such, all the three network entities share the responsibility of storing security tokens required for effective authentication and key agreement. This ensures that no network entity is solely overwhelmed with storage as the network expands to accommodate more vehicles or RSUs. Another major feature of the proposed protocol that renders it scalable is its extremely low computation costs, as shown in Table 3. This ensures that high numbers of vehicles and roadside units can be authenticated devoid of overwhelming the network with excessive computational overheads.

Comparative performance evaluation

In the performance evaluation of most of the VANETs verification schemes, computation costs, supported functionality, and communication overheads are frequently utilized. Whereas the computation costs gives the execution duration of the various cryptographic operations, the communication costs accounts for the number of bits exchanged (and hence the bandwidth consumed). On the other hand, the supported functionalities gives the basis for appraising the robustness of the authentication schemes against attacks. Consequently, these

Operation	Runtime (ms)
On-way hashing (T_h)	0.0003
Fuzzy extraction (T_{fe})	0.6548
PUF (T_{puf})	0.0035
EC scalar multiplication (T_{pm})	0.6548
EC point addition ($T_{p.a.}$)	0.0050
Bilinear pairing (BP) (T_{bp})	6.5364
BP scalar multiplication (T_{pmb})	2.6537
BP point addition (T_{pab})	0.0148
BP map-to-point hash function (T_{mh})	1.4265

Table 2. Cryptographic runtimes.

Protocol	Cryptographic operations	Total (ms)
Wang et al.[40]	$3T_h + 5T_{pmb} + 4T_{pab} + 2T_{bp}$	26.4014
Yang et al.[41]	$10T_h + 10T_{pm} + 6T_{p.a.}$	6.5810
Liang & Liu[42]	$10T_h + 6T_{pm} + 4T_{bp}$	30.0774
Mei et al.[43]	$4T_{mh} + 6T_{pmb} + 2T_{pab} + 4T_{bp}$	47.8034
Kamil et al.[44]	$3T_{mh} + 6T_{pmb} + 2T_{pab} + 3T_{bp}$	39.8405
Xu et al.[45]	$3T_{mh} + 5T_{pmb} + T_{pab} + 3T_{bp}$	37.1720
Altaf & Maity[46]	$3T_{mh} + 3T_{pmb} + T_{pab} + 3T_{bp}$	31.8646
Proposed	$17T_h + 6T_{fe} + 4T_{puf}$	3.9479

Table 3. Computation costs comparisons.

metrics are utilized to evaluate our protocol. Thereafter, the obtained values are compared with those of other state-of-the-art approaches.

Computation costs

To derive the computational overhead for our protocol, we implemented it in an Intel Core i5-13600 K laptop with 4-GB RAM, installed with the Ubuntu 16.04.3 LTS operating system, and having a clock frequency of 3.6 GHz. The cryptographic library deployed was the Pairing-Based Cryptography (PBC). In this environment, the durations taken by the various cryptographic operations are detailed in Table 2 below.

In the process of verifying the network elements and setting up the session key, the roadside unit RSU_j carries out $\{6T_h + 2T_{fe} + 2T_{puf}\}$ operations while the registration authority RA performs $\{5T_h + 2T_{fe}\}$ operations. In contrast, vehicle V_i carries out $\{6T_h + 2T_{fe} + 2T_{puf}\}$ cryptographic operations. Consequently, the cumulative computational overheads for our scheme is $\{17T_h + 6T_{fe} + 4T_{puf}\}$ as evidenced in Table 3.

As shown in Fig. 4, the scheme in[43] incurs the highest computation overheads. This is followed by the protocols in[40–46] respectively. Conversely, our protocol requires the least computation overheads of only 3.9479 ms. The computationally extensive bilinear pairing operations executed in[40,42–46] account for the exhibited high computation overheads.

Similarly, the scheme in[41] executes a relatively high number of elliptic curve scalar point multiplications, which also renders it computationally extensive. In contrast, the proposed protocol mostly executes efficient cryptographic operations such as one-way hashing and PUF operations. This is the rationale behind the least computation costs incurred in the proposed protocol. Since the VANETs components such as OBUs have limited computation power, it is recommended that authentication schemes be extremely computationally proficient. Since our scheme incurs the least computation overheads, it perfectly fulfills this important requirement.

Communication overheads

In this part, we derive the communication cost for our scheme and compare the obtained value with those of other related protocols. To accomplish this, we take into consideration the sizes of the message exchanged during the verification and key setup phase. These messages include $Auth_1 = \{A_3\}$, $Auth_2 = \{B_3\}$, $Auth_3 = \{B_4\}$, $Auth_4 = \{C_1\}$, $Auth_5 = \{C_3\}$ and $Auth_6 = \{C_5\}$. We take the PUF challenge - response pairs, identities, timestamps, one-way hashing function, and random nonces to be 128 bits, 64 bits, 32 bits, 160 bits, and 64 bits correspondingly. With this, the derivation of communication cost proceeds as follows:

$\boldsymbol{V_i \rightarrow RA}$:

$Auth_1 = \{PID_i || PID_j || A_1^* || A_2 || T_1\} = \{64 + 64 + 128 + 160 + 32\} = 448$ bits.

$\boldsymbol{RA \rightarrow V_i}$:

$Auth_2 = \{CV_i || A_4^* || B_1 || T_2\} = \{128 + 128 + 160 + 32\} = 448$ bits.

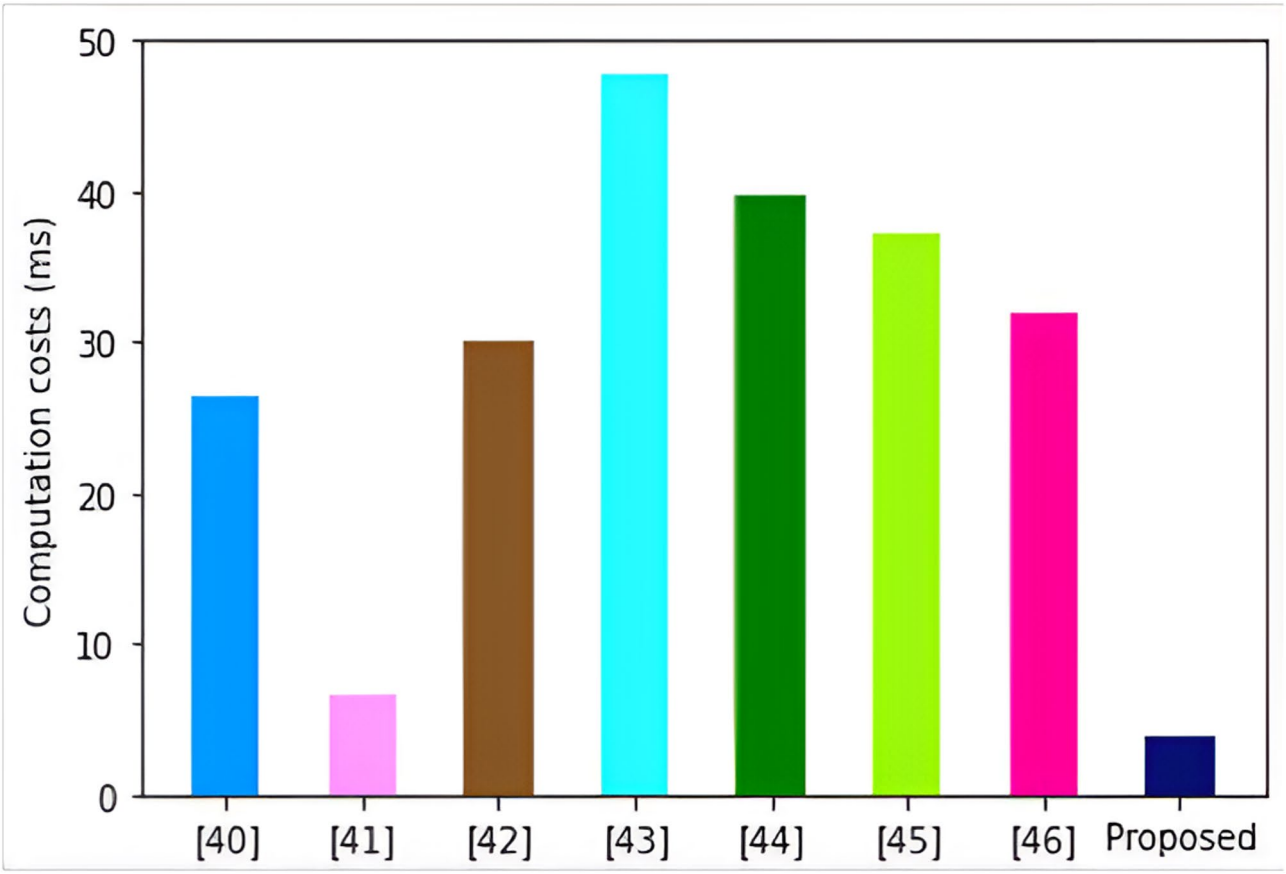

Fig. 4. Execution durations.

Protocol	Total (bits)
Wang et al.[40]	3104
Yang et al.[41]	3456
Liang & Liu[42]	2048
Mei et al.[43]	2080
Kamil et al.[44]	2080
Xu et al.[45]	2048
Altaf & Maity[46]	2080
Proposed	2752

Table 4. Communication costs comparisons.

$\boldsymbol{RA} \rightarrow \boldsymbol{RSU_j}$:

$Auth_3 = \{ PID_j||CR_j||A_5^*||B_2||T_2\}=\{64+128+128+160+32\} = 512$ bits.

$\boldsymbol{V_i} \rightarrow \boldsymbol{RA}$:

$Auth_4 = \{H_i||H_i^*||B_5||T_3\} =\{64+64+160+32\} = 320$ bits.

$\boldsymbol{RSU_j} \rightarrow \boldsymbol{RA}$:

$Auth_5 = \{ H_j||H_j^*||C_2||T_3\} = \{64+64+160+32\} = 320$ bits.

$\boldsymbol{RSU_j} \rightarrow \boldsymbol{V_i}$:

$Auth_6 = \{ H_i^*||C_3||A_5^*||C_4||T_5\} =\{64+320+128+160+32\} = 704$ bits.

Based on the above computations, our protocol's cumulative communication cost is 2752 bits. The communication overheads incurred by other related techniques are detailed in Table 4.

As shown in Fig. 5, the scheme in[41] exchanges the highest number of bits. This is followed by the protocol in[40] and the proposed scheme respectively.

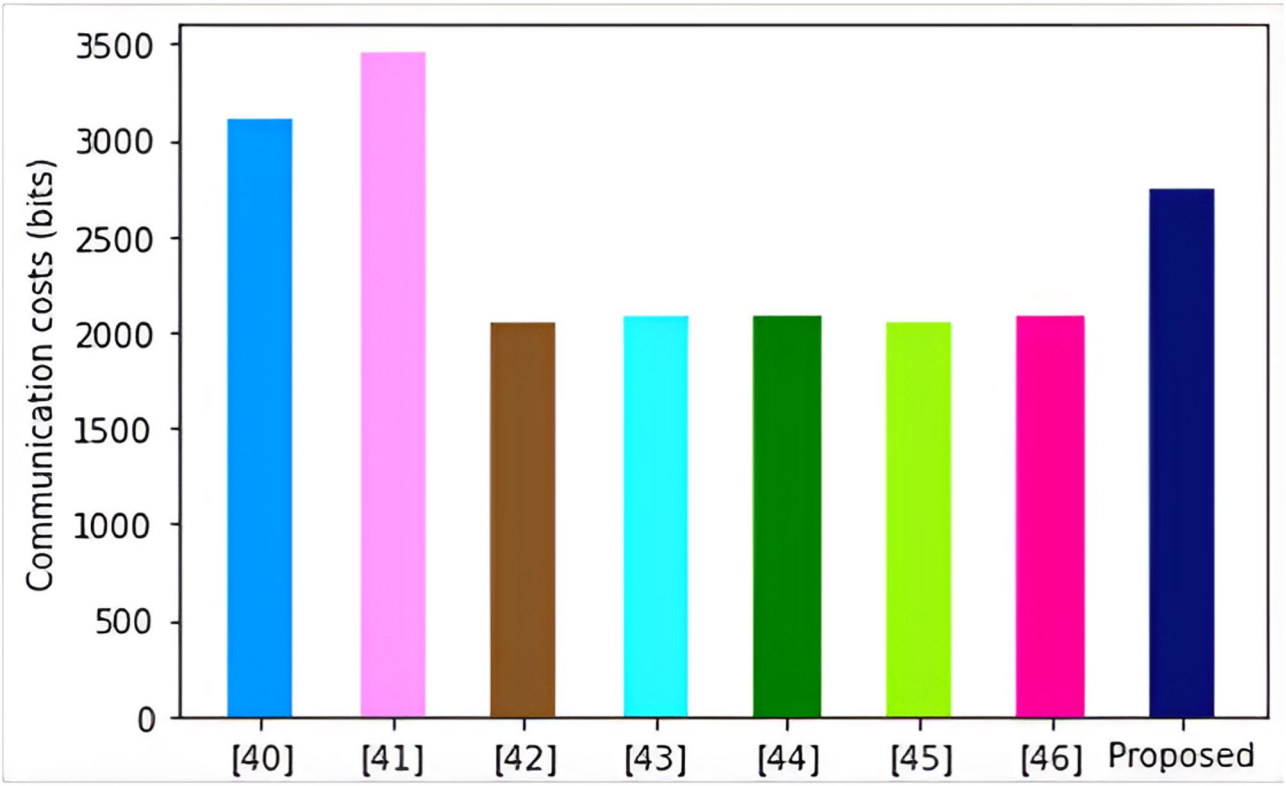

Fig. 5. Number of exchanged bits.

	[41]	[46]	[45]	[43]	[42]	[40]	[44]	Our scheme
Security features								
Joint verification	√	√	√	√	√	√	√	√
Session key setup	√	×	×	×	×	×	×	√
Anonymity	√	√	√	√	√	√	√	√
Conditional traceability	√	√	√	√	√	√	√	√
Key secrecy	×	×	×	×	×	×	×	√
Revocability	×	√	×	×	×	×	×	√
Non-repudiation	×	×	×	×	×	×	×	√
Confidentiality	×	√	√	√	×	√	√	√
Formal verification	√	√	√	√	√	√	√	√
Robust against								
Privileged insider	×	×	×	×	√	×	√	√
Impersonation	√	√	√	√	√	√	√	√
DoS	×	×	×	×	×	×	×	√
Physical attacks	√	√	√	√	×	×	×	√
De-synchronization	×	×	×	×	×	×	×	√
MitM	×	×	√	√	√	×	√	√
Eavesdropping	×	√	√	×	×	×	√	√
Replays	√	√	√	√	√	√	√	√

Table 5. Supported security features. √ Supported; × Not supported or not considered.

Although the schemes in[42,45] incur the lowest communication overheads, they fail to offer some critical security functionalities as evidenced in Table 5. Similarly, the protocols in[43] and [46] are susceptible to numerous attacks, as shown in Table 5. The six messages exchanged in our scheme ensure its robustness against numerous security threats (as shown in Table 5), but yields slightly high communication costs. The effect of this is the slightly increased network bandwidth consumption during the authentication phase.

Supported functionalities

In this part, the functionalities offered by the proposed protocol are presented, including its resilience to various attacks. Thereafter, these functionalities and robustness are compared with other state-of-the-art approaches, as shown in Table 5. These features include key secrecy, joint verification, session key setup, anonymity, conditional traceability, revocability, confidentiality, and formal verification.

In contrast, attacks resisted include privileged insider, impersonation, DoS, physical attacks, de-synchronization, MitM, eavesdropping, and packet replays. As shown in Table 5, the schemes in[40–46] support 8, 7, 10, 8, 10, 9, and 10 functionalities, respectively. Conversely, our protocol supports all the 17 functionalities. Therefore, the developed protocol offers the most salient security protection when compared with its peers.

The performance evaluation above has unequivocally demonstrated the efficiency of our scheme, since it requires the least computational costs overheads and slightly lower communication overheads. Using the scheme in[41] as the benchmark, our protocol attains a 66.696% reduction in computation costs. Its functionalities comparisons have shown that it supports the highest number of privacy and security functionalities. Using the techniques developed in[42,44,46] as the baselines, then the proposed scheme achieves a 70% increment in the supported security functionalities. Despite a slightly higher communication overhead, this is more than compensated by the enhanced security it provides. Given that most devices in vehicular ad hoc networks, such as OBUs, are resource-constrained, our scheme's lightweight nature makes it the most suitable for this environment.

Conclusion

Numerous vulnerabilities, threats, and attacks in VANETs have led to many security and privacy preservation schemes being developed by both the industry and academia. Most of these security solutions deploy cryptographic primitives such as public key infrastructure, blockchains, identity verification, bilinear pairing, elliptic curve scalar multiplication, and group signatures. Unfortunately, these techniques render the resulting security protocols inefficient or susceptible to attacks. In contrast, our protocol is demonstrated to be verifiably secure under the ROR model. In addition, the elaborate informal security analysis performed shows that the proposed protocol is robust against numerous VANET threats such as DoS, replays, eavesdropping, and physical capture. Moreover, our scheme is relatively lightweight in terms of performance, requiring the least computational overheads. However, during the authentication phase, the six messages exchanged results in slightly high communication overheads. As such, future research works should be directed towards approaches that could lessen the number and size of the exchanged messages so as to considerably minimize bandwidth consumption.

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 29 September 2024; Accepted: 5 August 2025
Published online: 09 August 2025

References

1. Tavasoli, M. et al. Data communication challenges of connected and automated vehicles in rural areas. *IEEE Access.* **13**, 29220–29251 (2025).
2. Zhou, X., He, D., Khan, M. K., Wu, W. & Choo, K. K. R. An efficient blockchain-based conditional privacy-preserving authentication protocol for VANETs. *IEEE Trans. Veh. Technol.* **72** (1), 81–92 (2022).
3. Parmar, K. et al. Privacy-preserving authentication scheme for VANETs using blockchain technology. *Procedia Comput. Sci.* **220**, 40–47 (2023).
4. Shawky, M. A., Abbasi, Q. H., Imran, M. A., Ansari, S. & Taha, A. Cross-layer authentication based on physical-layer signatures for secure vehicular communication. In *2022 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1315–1320). IEEE. (2022), June.
5. Nyangaresi, V. O. Provably secure authentication protocol for traffic exchanges in unmanned aerial vehicles. *High-Confidence Comput.* **3** (4), 100154, 1–13 (2023).
6. Song, X., Peng, Z., Song, S. & Stojanovic, V. Interval observer design for unobservable switched nonlinear partial differential equation systems and its application. *Int. J. Robust Nonlinear Control.* **34** (16), 10990–11009 (2024).
7. Peng, Z., Song, X., Song, S. & Stojanovic, V. Spatiotemporal fault Estimation for switched nonlinear reaction–diffusion systems via adaptive iterative learning. *Int. J. Adapt. Control Signal Process.* **38** (10), 3473–3483 (2024).
8. Shawky, M. A. et al. Adaptive chaotic map-based key extraction for efficient cross-layer authentication in VANETs. *Veh. Commun.* **39** (100547), 1–14 (2023).
9. Gao, L., Zhuang, Z., Tao, H., Chen, Y. & Stojanovic, V. Non-lifted norm optimal iterative learning control for networked dynamical systems: A computationally efficient approach. *J. Franklin Inst.* **361** (15), 107112, 1–18 (2024).
10. Zhang, R. & Zhou, W. Shared group session key-based conditional privacy-preserving authentication protocol for VANETs. *Veh. Commun.* **47**, 100782. https://doi.org/10.1016/j.vehcom.2024.100782 (2024).
11. Bayat, M., Pournaghi, M., Rahimi, M. & Barmshoory, M. NERA: A new and efficient RSU based authentication scheme for VANETs. *Wireless Netw.* **26**, 3083–3098 (2020).
12. Al-shareeda, M. A., Anbar, M., Manickam, S. & Hasbullah, I. H. An efficient identity-based conditional privacy-preserving authentication scheme for secure communication in a vehicular ad hoc network. *Symmetry* **12** (10), 1687, 1–25 (2020).
13. Zhu, D. & Guan, Y. Secure and lightweight conditional Privacy-Preserving identity authentication scheme for VANET. *IEEE Sens. J.* **24** (21), 35743–35756 (2024).
14. Zhang, G., Liao, Y., Fan, Y. & Liang, Y. Security analysis of an identity-based signature from factorization problem. *IEEE Access.* **8**, 23277–23283 (2020).
15. Naresh, V. S. & Reddi, S. An identity-based secure VANET communication system. *Secur. Priv.* **7** (2), 1–18 (2024). e349.
16. Qiao, Z. et al. An anonymous and efficient certificate-based identity authentication protocol for VANET. *IEEE Internet Things J.* **11** (7), 11232–11245 (2023).

17. Vangujar, A. K., Umrani, A. & Palmieri, P. Identity-based cluster authentication and key exchange (id-cake) message broadcasting and batch verification in vanets. In International Conference on Applied Cryptography and Network Security (pp. 162–179). Cham: Springer Nature Switzerland. (2024), March.
18. Ali, I., Chen, Y., Ullah, N., Kumar, R. & He, W. An efficient and provably secure ECC-based conditional privacy-preserving authentication for vehicle-to-vehicle communication in VANETs. *IEEE Trans. Veh. Technol.* **70** (2), 1278–1291 (2021).
19. Zhou, Y. et al. An efficient identity authentication scheme with dynamic anonymity for VANETs. *IEEE Internet Things J.* **10** (11), 10052–10065 (2023).
20. Asaar, M. R., Salmasizadeh, M., Susilo, W. & Majidi, A. A secure and efficient authentication technique for vehicular ad-hoc networks. *IEEE Trans. Veh. Technol.* **67** (6), 5409–5423 (2018).
21. Nyangaresi, V. O. Privacy preserving three-factor authentication protocol for secure message forwarding in wireless body area networks. *Ad Hoc Netw.* **142** (103117), 1–15 (2023).
22. Lin, C., He, D., Huang, X., Kumar, N. & Choo, K. K. R. BCPPA: A blockchain-based conditional privacy-preserving authentication protocol for vehicular ad hoc networks. *IEEE Trans. Intell. Transp. Syst.* **22** (12), 7408–7420 (2020).
23. Feng, Q., He, D., Zeadally, S. & Liang, K. BPAS: Blockchain-assisted privacy-preserving authentication system for vehicular ad hoc networks. *IEEE Trans. Industr. Inf.* **16** (6), 4146–4155 (2019).
24. Alzaidi, Z. S., Yassin, A. A., Abduljabbar, Z. A. & Nyangaresi, V. O. A Fog Computing and Blockchain-based Anonymous Authentication Scheme to Enhance Security in VANET Environments. *Eng. Technol Appl. Sci. Res.* **15**(1), 19143–19153 (2025).
25. Srivastava, S., Agarwal, D., Chaurasia, B. K. & Adhikari, M. Blockchain-based trust management for data exchange in internet of vehicle network. *Multimedia Tools Appl.* **84** (8), 4837–4855 (2025).
26. Kalidoss, L., Thouti, S., Arunachalam, R. & Ramamurthy, P. An efficient model of enhanced optimization and dilated-GRU based secured multi-access edge computing with blockchain for VANET sector. *Expert Syst. Appl.* **260**, 125275. https://doi.org/10.1016/j.eswa.2024.125275 (2025).
27. Juárez, R. & Bordel, B. Augmenting vehicular ad hoc network security and efficiency with blockchain: A probabilistic identification and malicious node mitigation strategy. *Electronics* **12** (23), 4794, 1–35 (2023).
28. Lu, Z., Wang, Q., Qu, G., Zhang, H. & Liu, Z. A blockchain-based privacy-preserving authentication scheme for VANETs. *IEEE Trans. Very Large Scale Integr. VLSI Syst.* **27** (12), 2792–2801 (2019).
29. Gabay, D., Akkaya, K. & Cebe, M. Privacy-preserving authentication scheme for connected electric vehicles using blockchain and zero knowledge proofs. *IEEE Trans. Veh. Technol.* **69** (6), 5760–5772 (2020).
30. Liu, X., Liu, Q., Luo, M., Yang, X. & Luo, Q. Blockchain-based efficient and traceable data sharing scheme for vehicular networks. *IEEE Trans. Veh. Technol.* 1–16. https://doi.org/10.1109/TVT.2025.3555108 (2025).
31. He, X. et al. A hierarchical blockchain-assisted conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *Sensors* **22** (6), 2299, 1–18 (2022).
32. Shawky, M. A. et al. Blockchain-based secret key extraction for efficient and secure authentication in VANETs. *J. Inform. Secur. Appl.* **74**, 103476, 1–18 (2023).
33. Lu, Z., Liu, W., Wang, Q., Qu, G. & Liu, Z. A privacy-preserving trust model based on blockchain for VANETs. *Ieee Access.* **6**, 45655–45664 (2018).
34. Beijia, H. & Yi, L. Blockchain-Based key management and security decisions in the internet of vehicles. *IEEE Internet Things J.* **14** (8), 1–18 (2025).
35. Surapaneni, P., Bojjagani, S. & Khan, M. K. DYNAMIC-TRUST: Blockchain-Enhanced trust for secure vehicle transitions in intelligent transport systems. *IEEE Trans. Intell. Transp. Syst.* 1–15. https://doi.org/10.1109/TITS.2025.3545755 (2025).
36. Mukathe, D., Di, W., Ahmed, W. & Worku, T. Blockchain-Powered authenticated key agreement scheme with Reputation-Incentive mechanism for Vehicle-to-Vehicle communication in IoV. *IEEE Internet Things J.* **14** (8), 1–16 (2025).
37. Rawat, G. S. et al. BTC2PA: A Blockchain-Assisted trust computation with conditional Privacy-Preserving authentication for connected vehicles. *IEEE Trans. Intell. Transp. Syst.* **26** (1), 1134–1148 (2024).
38. Nyangaresi, V. O. et al. A biometric and physically unclonable function–Based authentication protocol for payload exchanges in internet of drones. *e-Prime-Advances Electr. Eng. Electron. Energy*. **7**, 100471, 1–14 (2024).
39. Almazroi, A. A. et al. A bilinear Pairing-Based anonymous authentication scheme for 5G-Assisted vehicular fog computing. *Arab. J. Sci. Eng.* 1–22. https://doi.org/10.1007/s13369-024-09617-y (2024). https://link.springer.com/
40. Wang, H., Wang, L., Zhang, K., Li, J. & Luo, Y. A conditional privacy-preserving certificateless aggregate signature scheme in the standard model for VANETs. *IEEE Access.* **10**, 15605–15618 (2022).
41. Yang, Q., Zhu, X., Wang, X., Zheng, J. & Liu, Y. A novel authentication and key agreement scheme for internet of vehicles. *Future Generation Comput. Syst.* **145**, 415–428 (2023).
42. Liang, Y. & Liu, Y. Analysis and improvement of an efficient certificateless aggregate signature with conditional privacy preservation in VANETs. *IEEE Syst. J.* **17** (1), 664–672 (2022).
43. Mei, Q. et al. Efficient certificateless aggregate signature with conditional privacy preservation in IoV. *IEEE Syst. J.* **15** (1), 245–256 (2020).
44. Kamil, I. A. & Ogundoyin, S. O. On the security of privacy-preserving authentication scheme with full aggregation in vehicular ad hoc network. *Secur. Priv.* **3** (3), 1–20 (2020). e104.
45. Xu, Z., He, D., Kumar, N. & Choo, K. K. R. Efficient certificateless aggregate signature scheme for performing secure routing in VANETs. *Secur. Communication Networks*. **2020(1)** (5276813), 1–20 (2020).
46. Altaf, F. & Maity, S. PLHAS: Privacy-preserving localized hybrid authentication scheme for large scale vehicular ad hoc networks. *Veh. Commun.* **30**, 100347. https://doi.org/10.1016/j.vehcom.2021.100347 (2021).
47. Cao, Y., Xu, S., Chen, X., He, Y. & Jiang, S. A forward-secure and efficient authentication protocol through lattice-based group signature in VANETs scenarios. *Comput. Netw.* **214** (109149), 1–13 (2022).
48. Hussien, Z. A. et al. Lightweight integrity preserving scheme for secure data exchange in cloud-based IoT systems. *Appl. Sci.* **13** (2), 691, 1–20 (2023).
49. Abduljabbar, Z. A. et al. Session-dependent token-based payload enciphering scheme for integrity enhancements in wireless networks. *J. Sens. Actuator Networks*. **11** (3), 55, 1–16 (2022).
50. Du, Z., Xie, X., Qu, Z., Hu, Y. & Stojanovic, V. Dynamic event-triggered consensus control for interval type-2 fuzzy multi-agent systems. *IEEE Trans. Circuits Syst. I Regul. Pap.* **71** (8), 3857–3866 (2024).
51. You, W., Xie, X., Wang, H., Xia, J. & Stojanovic, V. Relaxed model predictive control of TS fuzzy systems via a new switching-type homogeneous polynomial technique. *IEEE Trans. Fuzzy Syst.* **32** (8), 4583–4594 (2024).
52. Xie, B. et al. PID-fuzzy switching-based strategy to heading control for remote operated vehicle. *Neural Comput. Appl.* 1–17. https://doi.org/10.1007/s00521-024-10911-x (2024).
53. Wang, D., Cheng, H., Wang, P., Huang, X. & Jian, G. Zipf's law in passwords. *IEEE Trans. Inf. Forensics Secur.* **12** (11), 2776–2791 (2017).

Author contributions

Author contributions "Conceptualization: Z.A.A., V.O.N., J.M.; Methodology: Z.A.A., V.O.N, M.A.A.; Investigation: J.M., M.A.A., A.A.A., H.A.N.; Validation: V.O.N, M.A.A., A.J.Y.A.; Formal analysis: Z.A.A., V.O.N., Z.A.H.; Writing—original draft preparation: V.O.N., A.J.Y.A.; Supervision: J.M., V.O.N.; Writing—review and

editing: M.A.A., H.A.N.; Project Administration: Z.A.A, J.M.; Visualization: A.A.A., M.A.H., A.H.A.; Resources: M.A.H., Z.A.H., A.H.A.; Funding acquisition: Z.A.A., M.A.A. All authors have read and agreed to the published version of the manuscript. ".Note: All authors reviewed the manuscript.

Funding

Scientific Research Capacity Enhancement Program for Key Construction Disciplines in Guangdong Province under Grant 2024ZDJS063.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Z.A.A. or J.M.