ZKP-based Lightweight Authentication protocol during handovers in Vehicular Networks

Indukuri Mani Varma
Computer Science and Engineering
Indian Institute of Technology, Roorkee, India
Roorkee, India
im_varma@cs.iitr.ac.in

Neetesh Kumar
Computer Science and Engineering
Indian Institute of Technology, Roorkee
Roorkee, India
neetesh@cs.iitr.ac.in

GLOBECOM 2023 - 2023 IEEE Global Communications Conference | 979-8-3503-1090-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/GLOBECOM54140.2023.10437483

***Abstract*—Internet of Vehicles (IoV), as an emerging technology, has attracted much research over the years due to rapid advancements in computing paradigms and vehicular and wireless technologies. These advancements enable vehicle-to-everything (V2X) communication to offer various services such as traffic management, data exchange, and route scheduling. However, the increase in density and the malicious behaviour of vehicle users have seriously threatened security and privacy concerns in the network. These concerns are related to anonymity, privacy, and verification of the identity of vehicle users. It is crucial to preserve users' privacy to prevent traceability and linkability, besides authentication to track malicious activities in the network. Therefore, in this paper, a novel privacy-preserving lightweight zk-SNARK of polynomial-based authentication protocol is presented. The vehicles are initially registered with a trusted authority (TA) in this protocol. After that, they are authenticated by RSUs, followed by verification of authentication during vehicle handover between RSUs. The proposed protocol is implemented using the Mininet-WiFi tool, and its performance is analyzed by comparing communication latency and computation time for variable vehicular density. An informal security analysis is also done to prove that the proposed protocol provides anonymity, privacy, user verifiability, and untraceability features.**

***Index Terms*—VANET, Authentication, Zero-knowledge Proofs, zk-SNARKs, Bilinear Pairing**

I. Introduction

Due to the recent advancements and developments in V2I and V2V communication, smart vehicles can get services such as lane change warnings, parking assistance, route planning, road safety assurance etc [1]. These services have raised hope for the evolution of many potential V2X applications in vehicular networks (VNs). However, these services do not guarantee protection against cyber attacks [2]. These attacks can lead to the broadcast of false information in the network, violation of traffic rules, disclosure of sensitive information of vehicle users, traffic congestion and accidents, etc. These hazardous consequences can pose a severe threat to human lives. Therefore, it is important to secure VNs with a focus on confidentiality and integrity of data, verification of the identity of vehicles in the network, availability of safety services, and trust. Besides these security services, utmost importance should be given to authentication, privacy, and anonymity of vehicle users [3]. The anonymity of a user refers to the state of not having an identity associated with him. To preserve the anonymity of vehicle users while using services in the network, pseudonyms are widely used [4]. Anonymous identities are assigned to autonomous vehicles as per IEEE 1609.2 standard [5]. However, pseudonyms can be changed frequently, which makes tracking malicious vehicles difficult. Therefore, a trusted authority (TA) must possess mapping information between the pseudonym and the vehicle's real identity. This mechanism assists the TA in revoking or removing malicious vehicles from the network. Privacy refers to a state of being able to control who has access to personal information. During the authentication process, the vehicle must provide its identity information to third parties to prove itself to be authentic. However, this mechanism cannot be trusted completely as third parties are again prone to attacks, or they themselves are untrustworthy.

Much research has been done on anonymous and privacy-preserving authentication schemes [6]. These schemes use cryptographic primitives and concepts such as digital signatures, anonymous public key cryptography, group signatures, hash algorithms, and zero-knowledge proofs (ZKPs). The security of these primitives is based on various computational provable hardness assumptions such as discrete-log problem, factorization problem (as in RSA), point multiplication (as in ECC) etc [7]. The protocol design flaws permit attackers to break the goal of authentication without necessarily breaking cryptographic provable hardness assumptions [8]. Therefore, it is essential to design robust authentication protocols utilizing cryptographic primitives. In this context, several shortcomings have been identified in authentication protocols discussed in [9]. The existing protocols either cannot resolve the unlinkability, traceability, and privacy issues or are inefficient in terms of computational time and lightweight security [10]. Therefore, we propose a privacy-preserving lightweight authentication protocol in VN utilizing Zero-knowledge Succinct Non-interactive ARgument of Knowledge (zk-SNARK) [11]. Zk-SNARK, an efficient variant of zero-knowledge proof (ZKP), allows an entity (prover) to prove the correctness of a statement/solution without revealing its contents to another entity (verifier).

As per the authors' knowledge, there has been no handover authentication mechanism based on ZKP discussed in the literature. The main contributions of this paper are mentioned as follows.

1) A novel zk-SNARK of polynomial-based lightweight authentication protocol using bilinear mapping for VNs is presented in three phases, i.e., registration, authentication, and handover.
2) The proposed protocol is implemented in Mininet-WiFi, and simulation results are presented. We analyzed the authentication latency and computational time during initial and handover authentication of vehicles.
3) An informal security analysis is presented to prove that the proposed authentication protocol provides anonymity, untraceability, privacy, and user verifiability.

II. Related Work

As per the authors' knowledge, a few works have been done based on ZKP-based authentication in VANETs. As part of these works, authors in [5] propose a ZKP-based authentication scheme using blockchain. The scheme uses pederson commitment scheme and zk-SNARK proof to verify the identity of the vehicle anonymously and employs a token-based approach to provide random pseudonyms to the vehicles. To reduce the utilization of high computational resources involved during the generation of zk-SNARK proofs, Khor *et al.* propose a blockchain-based lightweight anonymous authentication protocol for IoT devices [4]. The protocol uses zokrates, a framework for zk-SNARK proofs in the Ethereum blockchain, to verify proofs generated by IoT devices. A novel efficient anonymous authentication approach for the IoV based on ZKP and elliptic curve cryptography (ECC) [10]. The Fujisaki–Okamoto Commitment algorithm is used to achieve the user's authenticity and anonymity. In the scheme, the trusted authority can track users using the verification keys. A fast reconnection procedure based on the security context from the last access has also been proposed to reduce the computation overhead effectively. However, in this scheme, public-key cryptography is used to secure data during transit. To overcome issues in certificate management, a token-based approach is proposed to ensure mutual authentication in V2I communication [12]. This scheme employs timed efficient stream loss-tolerant authentication protocol as an underlying broadcast authentication scheme to achieve various security goals. In all the above protocols, vehicle handover between RSUs is not handled. Since RSUs maintain transmission coverage areas, it is crucial to ensure the authenticity of vehicles across these areas. To facilitate handover of vehicles between RSUs, a V2I handover authentication protocol is discussed in [13]. This protocol uses hash and XOR operations to compute intermediate parameters and store the results in the blockchain. In our proposed protocol, along with the authentication of vehicles using ZKP, handover authentication is also handled. The data exchange during handover is secured with symmetric key cryptography. The proposed protocol employs bilinear pairing and polynomial commitment-based ZKP to anonymously verify vehicles during RSU handovers.

III. System Architecture

This section discusses the network and threat models in which the proposed protocol is employed and cryptographic concepts used as part of the protocol design.

A. Network Model

The network model is intended to be a VN with entities such as a trusted authority, an authentication server, multiple RSUs, and vehicles. The architecture and communication flow among these entities is depicted in Fig. 1. As part of the model, it is assumed that all the channels between the entities are secure. The functionalities of each entity in the model are illustrated as follows.

1) *Authentication Server*: The authentication server (AS) stores the identifying parameters, prover, and verifier keys in its database. RSUs can retrieve these parameters and keys during the authentication of the vehicles.

2) *TA*: The TA functions as a trusted party to facilitate the registration of vehicles and generate cryptographic parameters. During the registration process, the TA registers each vehicle, and the corresponding proving and verification keys are computed and sent to the vehicle and AS, respectively.

3) *RSU*: The RSU is responsible for authenticating the vehicle when it enters the transmission range. The authentication process is initiated by the vehicle as per the protocol. The RSU generates a new vehicle identity, a session key and broadcasts them to the vehicle and its neighboring RSUs after successful authentication.

4) *On-board Unit of Vehicle*: A vehicle is generally equipped with an on-board unit (OBU) as a transceiver. The OBU monitors and processes data and communicates with RSUs and OBUs of other vehicles [14].

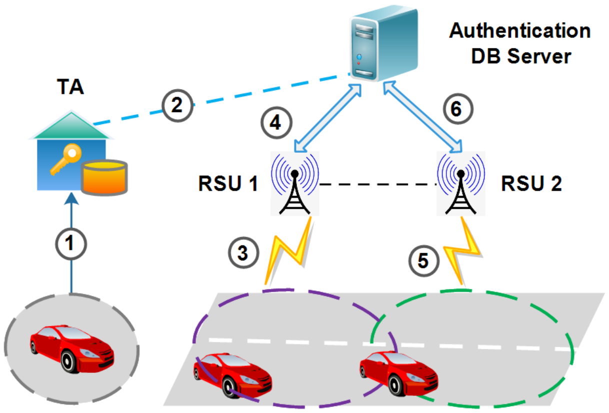


Fig. 1. Network Model with step ①. Vehicle registration with TA, ②. TA stores vehicle registration details in server, ③. Vehicle initial authentication with RSU1, ④. RSU1 stores authentication details in server, ⑤. Vehicle handover authentication with RSU2, ⑥. RSU2 retrieves initial authentication details from server

B. Threat Model

In the threat model, the work focusses on the following security and privacy goals.

1) *User Anonymity*: During the authentication process, the real identity is never disclosed by the vehicle or TA. Pseudonyms are used, but adversaries may infer relevant information on the vehicle's real identity based on the pseudonyms.

2) *Untraceability*: During handover authentication, the location of a vehicle can be tracked by the RSUs, and violates the privacy of location.

3) *Unforgeability*: Adversaries may identify or link genuine vehicles if an RSU is compromised or the wireless channel is controlled. An adversary can issue fake proof to be authenticated successfully.

4) *Proof-based user verifiability*: Unlike conventional authentication schemes, different proofs can be sent by the vehicles based on the challenges issued by verifiers.

C. Bilinear Mapping

A bilinear map $e : G{\times}G- > G_T$ is a mathematical function which maps multiplication of two points in group G to a point in group G_T [15]. The groups G, G_T are finite cyclic groups of some prime order. The bilinear mapping should satisfy the following constraints.

1) $e(g^a, g^b) = e(g, g)^{ab} \forall a, b \in Z, g \in G$ where Z is cyclic group of some prime order.
2) A polynomial-time computable and non-degenerate, g generates $G \implies e(g, g)$ generates G_T.

D. zk-SNARK

Zero-knowledge Succinct Non-interactive ARgument of Knowledge (zk-SNARK) is a variant of ZKP and can be distinguished by the properties as follows:

1) *Non-interactive*: a prover can generate proofs without interaction with the verifier.
2) *Succinct*: proofs are short and can be verified quickly.
3) *Constant size*: the generated proofs are of same size irrespective of the complexity of problem being proven.
4) *Publicly verifiable*: anyone can verify without sharing any secret key.
5) *Private inputs*: the ability to prove using private inputs not revealed to the verifier.
6) *Unforgeability*: computationally infeasible to create false proofs that will be successfully verified.
7) *Dynamic computation*: the idea of verified computing is conceived for dynamic computation.

Zk-SNARK of polynomials is a variant of the ZKP-based technique to prove and verify the correctness of a statement without disclosing any knowledge to the verifier. The discussed protocol uses zk-SNARK and homomorphic encryption of polynomials. It consists of three phases.

1) Setup Phase: In this pre-processing phase, both the prover and verifier agree upon the generator g from the finite field $\mathbb{Z}_p^*$ and a target polynomial $t(x)$. A trusted computing device (trusted setup) is used to perform the following steps.

1) The trusted setup device chooses random values s, α
2) Calculates encryptions: g^{α}, $\{g^{s^i}\}_{i \in [d]}$, $\{g^{\alpha s^i}\}_{i \in \{0......d\}}$
3) Calculate the proving key: $(\{g^{s^i}\}_{i \in [d]}, \{g^{\alpha s^i}\}_{i \in \{0......d\}})$
4) Calculate the verification key: $(g^{\alpha}, g^{t(s)})$

2) Proving Phase:

1) Assign the coefficients $\{c_i\}_{i \in \{0......d\}}$ (i.e., witness/knowledge), $p(x) = c_d x^d + ... + c_1 x^1 + c_0 x^0$
2) Calculate the polynomial $h(x) = p(x)/t(x)$
3) Evaluate the encrypted polynomials $g^{p(s)}$ and $g^{h(s)}$ using $\{g^{s^i}\}_{i \in [d]}$
4) Evaluate the encrypted shift polynomials $g^{\alpha . p(s)}$ using $\{g^{\alpha . s^i}\}_{i \in 0...d\}}$
5) Choose a random sample δ
6) Set the randomized proof $\pi = (g^{\delta p(s)}, g^{\delta h(s)}, g^{\delta \alpha p(s)})$

3) Verification Phase:

1) Parse the proof π as $(g^p, g^h, g^{p'})$
2) Compare polynomial restriction: $e(g^{p'}, g) = e(g^p, g^{\alpha})$
3) Compare polynomial cofactors: $e(g^p, g) = e(g^h, g^{t(s)})$

If polynomial restrictions and cofactors are compared successfully, the prover is verified without disclosure of the polynomials $p(x)$ and $h(x)$.

IV. The Proposed Protocol

The protocol involves three phases i.e., vehicle registration, initial authentication and handover authentication and are illustrated as shown in Fig. 2, 3 and 4 respectively.

A. Vehicle Registration Phase

The vehicle V_i initiates the registration process. Initially, V_i chooses its ID as VID_i and password PWD_i. The vehicle's AU computes $IPW_i = h(VID_i, PWD_i)$, where $h()$ is a hash function. The AU sends VID_i, IPW_i, and the registration request Reg_req to TA across the secure channel. After TA receives the registration request, it generates two random numbers x, y and calculates $NVID_i = IPW_i \oplus h(x \| T_c) \oplus h(y \| T_c)$ where T_c is TA's current timestamp. TA sends VID_i, $NVID_i$ to V_i. If received VID_i matches, V_i generates two univariate polynomials, i.e., target polynomial $t(x)$, $h(x)$, and multiply them to get prover's (V_i) polynomial $p(x)$ of degree d. V_i sends $NVID_i$, $t(x)$, $p(x)$'s degree d to TA. TA checks if $NVID_i$ matches, generates a random number k and computes $NVIDnew_i = NVID_i \oplus h(k \| T_c) \oplus h(x \| T_c)$. TA sends $NVIDnew_i$, IPW_i and registration status of V_i. V_i compares IPW_i and knows about its registration status. After the registration process is successful, TA chooses two random numbers s, α, calculates (g^{α}, $\{g^{s^i}\}$, $\{g^{\alpha . s^i}\}$ for i in $0, 1, ..., d$) and the values s, α are deleted from its memory. The TA then stores proving key $\{g^{s^i}, g^{\alpha . s^i}\}$, verification key $\{g^{\alpha}, g^{t(s)}\}$, $NVIDnew_i$, IPW_i in an external database server.

B. Vehicle Authentication Phase

The vehicle V_i generates a random number $auth\_r$ and initiates the authentication process by sending { IPW_i, $NVID_i$, and $auth\_r$ } to RSU. The RSU accesses the AS database to ensure V_i has been registered using $NVID_i$. If $NVID_i$ matches with $NVIDnew_i$ from the database, it acquires other parameters. These parameters include encryptions (g^{α}, $\{g^{s^i}\}$, $\{g^{\alpha s^i}\}$ for i in $0, 1, ..., d$), r, and $g^{t(s)}$. The RSU sends these encryptions to V_i i.e., $\{g^{s^i}\}$, $\{g^{\alpha s^i}\}$ for i in $(0, 1, ..., d)$ along with $auth\_r$.

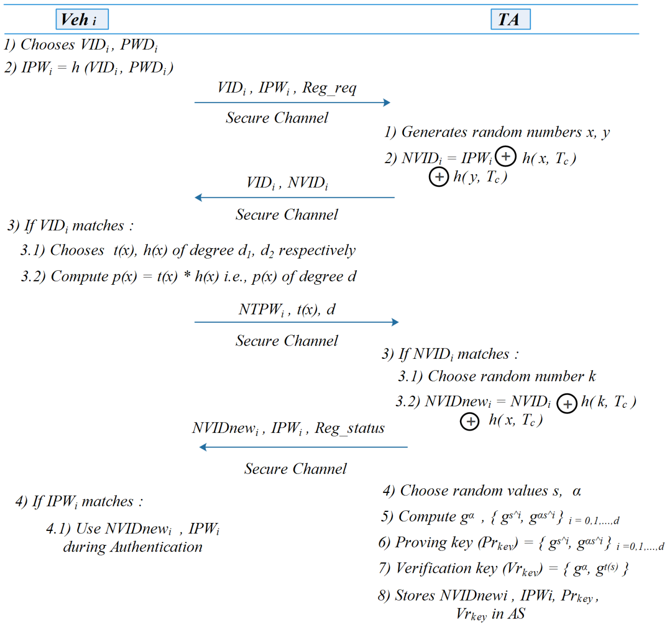

Veh i
TA
1) Chooses VID_i, PWD_i
2) $IPW_i = h(VID_i, PWD_i)$
VID_i, IPW_i, Reg_req
Secure Channel
1) Generates random numbers x, y
2) $NVID_i = IPW_i \oplus h(x, T_c) \oplus h(y, T_c)$
VID_i, $NVID_i$
Secure Channel
3) If VID_i matches :
3.1) Chooses t(x), h(x) of degree d_1, d_2 respectively
3.2) Compute p(x) = t(x) * h(x) i.e., p(x) of degree d
$NTPW_i$, t(x), d
Secure Channel
3) If $NVID_i$ matches :
3.1) Choose random number k
3.2) $NVIDnew_i = NVID_i \oplus h(k, T_c) \oplus h(x, T_c)$
$NVIDnew_i$, IPW_i, Reg_status
Secure Channel
4) If IPW_i matches :
4.1) Use $NVIDnew_i$, IPW_i during Authentication
4) Choose random values s, α
5) Compute g^{α}, $\{g^{s^\wedge i}, g^{\alpha s^\wedge i}\}_{i=0,1,\ldots,d}$
6) Proving key (Pr_{key}) = $\{g^{s^\wedge i}, g^{\alpha s^\wedge i}\}_{i=0,1,\ldots,d}$
7) Verification key (Vr_{key}) = $\{g^{\alpha}, g^{t(s)}\}$
8) Stores NVIDnewi , IPWi, Pr_{key}, Vr_{key} in AS

Fig. 2. Vehicle Registration Phase

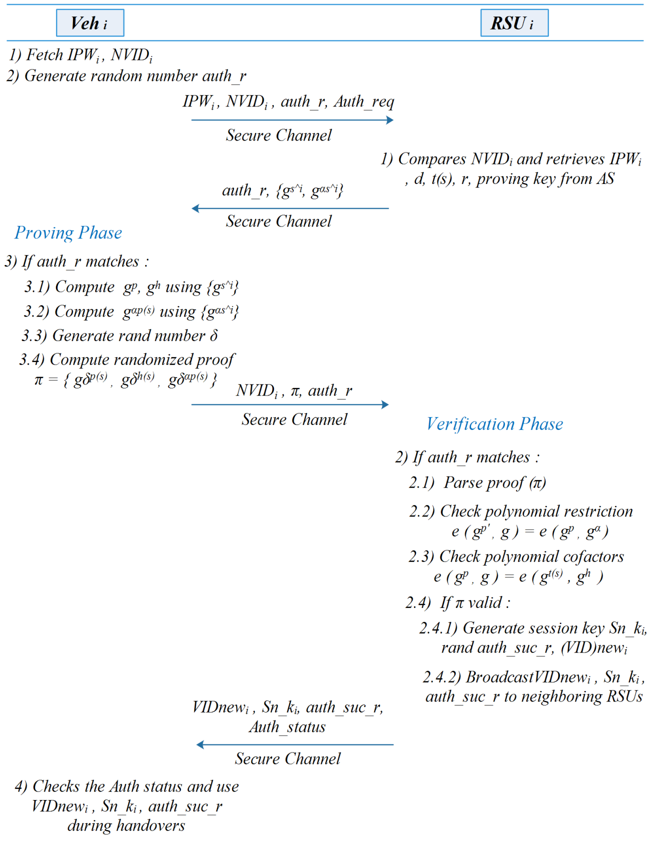

Veh i
RSU i
1) Fetch IPW_i, $NVID_i$
2) Generate random number auth_r
IPW_i, $NVID_i$, auth_r, Auth_req
Secure Channel
1) Compares $NVID_i$ and retrieves IPW_i, d, t(s), r, proving key from AS
auth_r, $\{g^{s^\wedge i}, g^{\alpha s^\wedge i}\}$
Secure Channel
Proving Phase
3) If auth_r matches :
3.1) Compute g^p, g^h using $\{g^{s^\wedge i}\}$
3.2) Compute $g^{\alpha p(s)}$ using $\{g^{\alpha s^\wedge i}\}$
3.3) Generate rand number δ
3.4) Compute randomized proof $\pi = \{g^{\delta p(s)}, g^{\delta h(s)}, g^{\delta \alpha p(s)}\}$
$NVID_i$, π, auth_r
Secure Channel
Verification Phase
2) If auth_r matches :
2.1) Parse proof (π)
2.2) Check polynomial restriction $e(g^{p'}, g) = e(g^p, g^{\alpha})$
2.3) Check polynomial cofactors $e(g^p, g) = e(g^{t(s)}, g^h)$
2.4) If π valid :
2.4.1) Generate session key Sn_k_i, rand auth_suc_r, $(VID)new_i$
2.4.2) Broadcast$VIDnew_i$, Sn_k_i, auth_suc_r to neighboring RSUs
$VIDnew_i$, Sn_k_i, auth_suc_r, Auth_status
Secure Channel
4) Checks the Auth status and use $VIDnew_i$, Sn_k_i, auth_suc_r during handovers

Fig. 3. Vehicle Authentication Phase

a) Proving Phase: If received $auth\_r$ matches, V_i evaluates encrypted polynomials $g^{h(s)}$ and $g^{p(s)}$ using $\{g^{s^i}\}_{i\in[d]}$ and encrypted shift polynomial $g^{\alpha.p(s)}$ using $\{g^{\alpha s^i}\}_{i\in\{0......d\}}$. V_i generates a random number δ and formulates the randomized proof as $\pi = \{\mathrm{g}^{\delta p(s)}, g^{\delta h(s)}, g^{\delta\alpha p(s)}\}$. After computing proof π, V_i sends $NVID_i$, $auth\_r$, π to RSU.

b) Verification Phase: RSU parses the received proof π as $(g^p, g^h, g^{p'})$ and verifies polynomial restriction $e(g^{p'}, g) = e(g^p, g^{\alpha})$ and polynomial cofactors $e(g^p, g) = e(g^{t(s)}, g^h)$. After successfully verifying the proof using these conditions, RSU generates a random number $auth\_suc\_r$, a session key Sn_k_i, a new $VIDnew_i$, and sends them along with V_i's authentication status to both V_i and its neighboring RSUs. These parameters $VIDnew_i$, session key Sn_k_i, and $auth\_suc\_r$ can be utilized to authenticate V_i during RSU handovers.

C. RSU Handover Phase

After successful authentication of vehicle V_i, it can use the session key Sn_k_i to secure communications with other RSUs in the network. During RSU handover, V_i encrypts Enc(VID_i, $auth\_suc\_r$) using key Sn_k_i and sends it along with VID_i. Since RSU_j already has the authenticated V_i's session key (received from neighboring RSU_i) in its database, it decrypts the received data, compares VID_i and $auth\_suc\_r$ respectively. If both values match, RSU_j generates new ID $VIDnew_i$ for V_i and a random number $hand\_suc\_r$. It encrypts Enc($VIDnew_i$, $hand\_suc\_r$) using key Sn_k_i and sends to V_i and its neighboring RSU_k. After receiving this data, V_i decrypts and uses $VIDnew_i$, Sn_k_i, and $hand\_suc\_r$ during the next RSU handover.

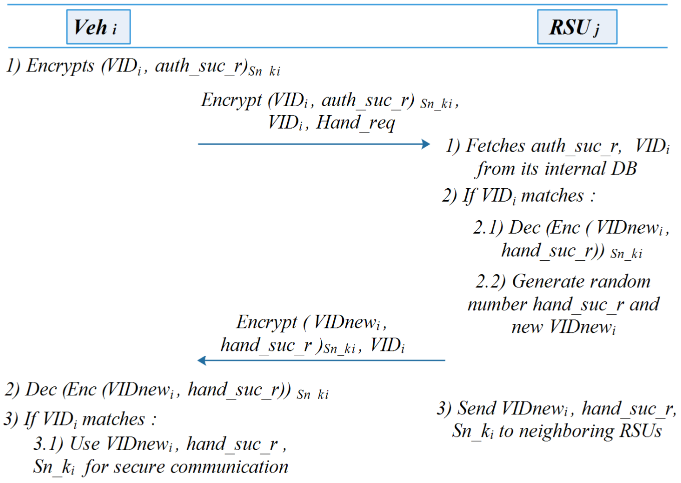
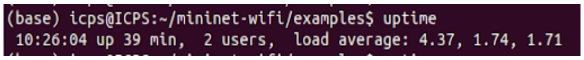

Veh i
RSU j
1) Encrypts $(VID_i, auth\_suc\_r)_{Sn\_ki}$
Encrypt $(VID_i, auth\_suc\_r)_{Sn\_ki}$, VID_i, Hand_req
1) Fetches auth_suc_r, VID_i from its internal DB
2) If VID_i matches :
2.1) Dec (Enc ($VIDnew_i$, hand_suc_r)) $_{Sn\_ki}$
2.2) Generate random number hand_suc_r and new $VIDnew_i$
Encrypt ($VIDnew_i$, hand_suc_r $)_{Sn\_ki}$, VID_i
2) Dec (Enc ($VIDnew_i$, hand_suc_r)) $_{Sn\_ki}$
3) If VID_i matches :
3.1) Use $VIDnew_i$, hand_suc_r , Sn_k_i for secure communication
3) Send $VIDnew_i$, hand_suc_r, Sn_k_i to neighboring RSUs

Fig. 4. Vehicle Handover Phase

V. Security Analysis and Results

A. Experimentation and Evaluation Results

The protocol simulation is done using Mininet-WiFi [16] in Ubuntu 22.04 (64-bit) with 32 GB RAM and 16 CPU cores. The simulation employs a random direction mobility model, log-distance loss model as a propogation model and is operated in 802.11g wireless mode. For convenience, the communication range of RSU is set to 30 meters. The working code is implemented in Python and is open-source at [17] along with simulation videos. Since mininet-wifi does not support 1609.2 WAVE communication standard [18], TCP/IP protocol is employed for communication between the vehicle and RSU. During the simulation, a new thread is created for each vehicle node, and communication persists with RSU exclusively. The average computational load on the Ubuntu system is analyzed using the Linux command *uptime*. The load average before simulation is 0.13, 0.73, and 1.48 for 1, 5, and 15 minutes, respectively. The load average during or at the end of the simulation with 60 vehicles is shown in Fig. 5. The output shows that the load is very low with respect to the number of CPU cores.



```
(base) icps@ICPS:~/mininet-wifi/examples$ uptime
 10:26:04 up 39 min,  2 users,  load average: 4.37, 1.74, 1.71
```

Fig. 5. Load on the system (16 CPU cores) during the simulation

The simulation has two RSU networks with RSU1 network: 192.168.0.0/24, and RSU2 network: 192.168.1.0/24. Initially, a server is run as a TA, and registration of vehicles is done manually. The latency and computation time for 60 vehicles during the registration phase are 0.05285 and 0.0494 seconds, respectively. As part of the simulation, the efficiency of the protocol is analyzed by running a variable number of vehicles at different speeds. We generated authentication requests from vehicles at the same time and plotted the results. This experiment has been performed to simulate how an RSU handles a load of multiple requests at an instance.

It has been observed that latency and computation time during initial authentication increase as the density of vehicles increases with an increase in speed, as shown in Fig. 6 and 7, respectively. The handover authentication latency and computation time are initially low for vehicles moving at higher speeds. However, their values gradually increase for vehicles moving at high speeds with an increase in vehicular density. Since the same operations are performed with variable polynomials and their degrees during the proving and verification phases, computation time increases linearly with vehicle density. During simulation, after initial authentication, an RSU handles multiple vehicles' handover requests, as all vehicles are expected to be associated simultaneously. In this context, it is observed that handover authentication is slow. During the RSU handover, each vehicle gets a new internet protocol (IP) address after associating with a new RSU. This operation consumes some time, and as per the protocol, the data is encrypted and decrypted before communication between both parties. The computation time increases with vehicle density and is almost the same for vehicles moving at different speeds. The communication latency and computation time at variable vehicle density for handover authentication are plotted in Fig. 8 and 9, respectively.

B. Security Goals

The following assumptions/proofs have been made to prove the robustness of the proposed protocol.

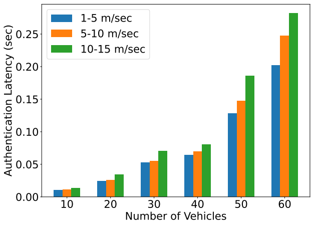


Fig. 6. Authentication Latency

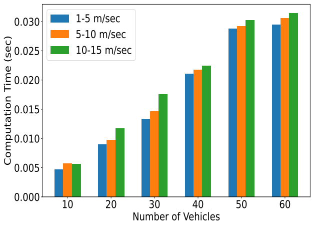


Fig. 7. Computation Time during Authentication phase

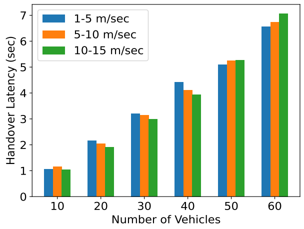


Fig. 8. Communication Latency during Handover phase

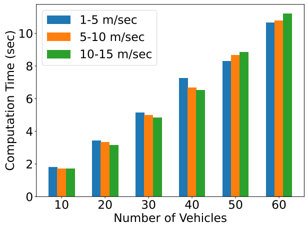


Fig. 9. Computation Time during Handover phase

Theorem 1: It is computationally hard for the prover to generate non-legitimate yet matching polynomial evaluations.

Proof: During the registration phase, the TA generates a random number s and α, computes encryption of s and homomorphically multiplies by α as $(g^s)^\alpha$ and sends to the vehicle (prover). Since the prover does not know the values of α and s, it is infeasible for him to generate non-legitimate yet matching evaluations due to the cryptographic hardness assumptions in homomorphic encryption.

Theorem 2: A prover cannot generate fake proofs during initial authentication.

Proof: In the protocol, the prover uses the encryptions $\{g^{s^i}\}_{i\in[d]}$ and $\{g^{\alpha s^i}\}_{i\in\{0......d\}}$ to evaluate $g^{p(s)}$, $g^{h(s)}$ and encrypted shift polynomial $g^{\alpha.p(s)}$ respectively. The secret values s, α created in the setup phase make it infeasible for the prover to generate fake proofs. It is also computationally infeasible to generate fake proofs if the prover does not know the actual polynomials $p(x)$ and $h(x)$.

Theorem 3: The verifier (RSU) extracts zero knowledge about the polynomial from the proof sent by the verifier (vehicle).

Proof: To ensure that the verifier gains zero knowledge from the proof, the encrypted polynomials in the proof are shifted by random number δ in the protocol. The values in the proof are exponentiated with δ as π = $(g^{\delta p(s)}, g^{\delta h(s)}, g^{\delta\alpha p(s)})$. With these exponentiations, the equations in the verification phase remain balanced and it is computationally infeasible for the verifier to find the value of δ and polynomials $p(x)$ and $h(x)$.

C. Informal Security Analysis of the protocol

In this subsection, we provide an informal analysis of how the proposed protocol provides security and privacy services.

TABLE I
SECURITY COMPARISION

Scheme	Anonymity	Untraceability	Unforgeability	Proof-based User Verifiability	User Privacy
[20]	×	✓	✓	✓	✓
[10]	✓	×	✓	✓	✓
[19]	✓	×	✓	✓	✓
Our scheme	✓	✓	✓	✓	✓

1) *Vehicle User Privacy and Anonymity*: During the authentication and handover phases, the vehicle sends pseudo identities, i.e., $NVID_i$ and VID_i, generated during the registration and authentication phases respectively. These parameters are used to verify and authenticate the registered vehicles. Since the actual identity of the vehicle is not shared with any RSU, the vehicle users remain anonymous to RSUs and other entities. The original identity of vehicle is not shared by TA with any other entities in the VN, thereby providing privacy and anonymity.

2) *Untraceability*: As per the protocol, the session key Sn_k_i, VID_i and random number $hand\_suc\_r$ shared during handovers are updated over a time period. The frequent change in keys and random numbers makes it impossible for RSUs to track the location of vehicles, thereby providing untraceability.

3) *Proof-based User Verifiability*: The inherent proving and verification steps in zk-SNARKs guarantee the verifiability and authenticity of the vehicle. Since the vehicle maintains the polynomials $p(x)$ and $h(x)$ securely, it is computationally infeasible for adversaries to impersonate a genuine vehicle.

4) *Unforgeability*: Since RSUs share the proving keys securely based on $NVID_i$, it is computationally infeasible for an attacker/vehicle to send false proof, yet the verification process is successful. Besides this, it is also infeasible for a vehicle to generate proofs without the knowledge of polynomials $p(x)$ and $h(x)$.

Based on the above security features, we compare our work with existing ZKP-based authentication protocols in VNs [10] [19]. As shown in Table I, ✓means the corresponding security feature is provided, × means security feature is not satisfied.

VI. CONCLUSION AND FUTURE WORK

In this paper, a zk-SNARK of polynomial-based authentication protocol in vehicular networks is discussed. As part of this protocol, privacy, anonymity and verification services using polynomial-based ZKPs and bilinear mapping are provided. An informal security analysis is presented to ensure that the proposed protocol provides untraceability and unforgeability of vehicle users. Meanwhile, the handover authentication slightly consumes affordable latency for additional security features and network constraints. The experiment results depict that communication latency for registration and initial is very low and affordable for RSUs. Our future work involves security in V2V communications.

ACKNOWLEDGEMENT

This work is supported by *Qualcomm Inc.* as part of the project *Software Defined Networking enabled Secure Vehicular Communication* under the programme *Qualcomm 6G University Research India Program* with grant number *IND-516251*.

REFERENCES

[1] T. Yoshizawa, D. Singelée, J. T. Muehlberg, S. Delbruel, A. Taherkordi, D. Hughes, and B. Preneel, "A survey of security and privacy issues in v2x communication systems," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–36, 2023.

[2] Y. Aref and A. Ouda, "Autonomous vehicle cyber-attacks classification framework," in *2023 15th International Conference on COMmunication Systems NETworkS (COMSNETS)*, 2023, pp. 373–377.

[3] M. A. Al-Shareeda, M. Anbar, I. H. Hasbullah, and S. Manickam, "Survey of authentication and privacy schemes in vehicular ad hoc networks," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 2422–2433, 2021.

[4] J. H. Khor, M. Sidorov, N. T. M. Ho, and T. H. Chia, "Public blockchain-based lightweight anonymous authentication platform using zk-snarks for low-power iot devices," in *2022 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2022, pp. 370–375.

[5] D. Gabay, K. Akkaya, and M. Cebe, "Privacy-preserving authentication scheme for connected electric vehicles using blockchain and zero knowledge proofs," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5760–5772, 2020.

[6] S. A. Jan, N. U. Amin, M. Othman, M. Ali, A. I. Umar, and A. Basir, "A survey on privacy-preserving authentication schemes in vanets: attacks, challenges and open issues," *IEEE Access*, vol. 9, pp. 153 701–153 726, 2021.

[7] J. Katz and Y. Lindell, *Introduction to modern cryptography*. CRC press, 2020.

[8] W. Mao, *Modern Cryptography: Theory and Practice*. Prentice Hall, 2003.

[9] F. Azam, S. K. Yadav, N. Priyadarshi, S. Padmanaban, and R. C. Bansal, "A comprehensive review of authentication schemes in vehicular ad-hoc network," *IEEE access*, vol. 9, pp. 31 309–31 321, 2021.

[10] N. Xi, W. Li, L. Jing, and J. Ma, "Zama: A zkp-based anonymous mutual authentication scheme for the iov," *IEEE Internet of Things Journal*, vol. 9, no. 22, pp. 22 903–22 913, 2022.

[11] M. Petkus, "Why and how zk-snark works," *arXiv preprint arXiv:1906.07221*, 2019.

[12] W. Hathal, H. Cruickshank, Z. Sun, and C. Maple, "Certificateless and lightweight authentication scheme for vehicular communication networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 16 110–16 125, 2020.

[13] S. Son, J. Lee, Y. Park, Y. Park, and A. K. Das, "Design of blockchain-based lightweight v2i handover authentication protocol for vanet," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 3, pp. 1346–1358, 2022.

[14] Z. Lu, G. Qu, and Z. Liu, "A survey on recent advances in vehicular network security, trust, and privacy," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 760–776, 2018.

[15] D. Meffert, "Bilinear pairings in cryptography," *Radboud Universiteit Nijmegen, Computing Science Department, the Netherlands*, pp. 22–82, 2009.

[16] ramonfontes, "Mininet-WiF," https://mininet-wifi.github.io/, 2023.

[17] 14vv1A0516, "ZKP codes," https://github.com/14vv1A0516/ZKP-based-Authentication-using-Mininet-wifi/, 2023.

[18] R. A. Uzcátegui, A. J. De Sucre, and G. Acosta-Marum, "Wave: A tutorial," *IEEE Communications magazine*, vol. 47, no. 5, pp. 126–133, 2009.

[19] A. A. Rasheed, R. N. Mahapatra, and F. G. Hamza-Lup, "Adaptive group-based zero knowledge proof-authentication protocol in vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 867–881, 2020.

[20] X. Zeng, G. Xu, X. Zheng, Y. Xiang, and W. Zhou, "E-aua: An efficient anonymous user authentication protocol for mobile iot," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1506–1519, 2018.