Design of Blockchain and ECC-Based Robust and Efficient Batch Authentication Protocol for Vehicular Ad-Hoc Networks

Sanjeev Kumar Dwivedi, *Member, IEEE*, Ruhul Amin, *Senior Member, IEEE*, Satyanarayana Vollala, *Member, IEEE*, and Ashok Kumar Das, *Senior Member, IEEE*

***Abstract*— The intelligent vehicles collect and distribute the data to other vehicles and Roadside Units (RSU), which ultimately strengthens the vehicular services in the Vehicular Ad-hoc Network (VANET). In order to protect against a variety of potential security threats, the VANET system needs a proper authentication mechanism, which requires more computational overheads and cannot perform better, when a cluster of vehicles sends the messages simultaneously. This paper aims to design a decentralized blockchain-based batch authentication protocol using Elliptic Curve Cryptography, where RSU authenticates the group of vehicles together. Moreover. our protocol also supports the verification of both individual messages (signature) generated by the vehicle and batch signature. We have used the Scyther security tool to verify our protocol and found that the protocol is safe and secure. A detailed comparative analysis reveals that the proposed scheme achieves more functionality and security compared to relevant schemes. The security analysis confirms that the proposed scheme is secure against all applicable attacks. Moreover, the ethereum platform simulates the proposed scheme, showing its effectiveness and confirming that it is feasible to deploy and execute transactions in real networks.**

***Index Terms*— VANET security, blockchain, batch authentication, Scyther simulation.**

I. Introduction

IN RECENT years, various new intelligent applications have evolved to simplify people's lives. Thanks to the fast expansion and inclusion of emerging technologies that provide the groundwork for intelligent applications. The smart Vehicular Ad-hoc Network (VANET)-based systems are one of the most promising new areas of study. The VANET system is constantly being improved and enriched by the academic community and the private sector. The most prominent network model VANET system [1], is a type of Wireless Sensor Network (WSN), which has restricted functionality. Nodes in VANET are mobility-in-nature, resulting in frequent topology changes. Through the use of an On-Board Unit (OBU), vehicles in the VANET system gather crucial data (for instance, traffic and accident information), sign the information using its private key, and broadcast it to the nearby intelligent vehicles and Roadside Units (RSU). RSU, with the assistance of a trusted authority, performs the authentication of vehicles and takes the necessary action accordingly. Moreover, the trusted authority and centralized cloud server store the crucial parameters required for vehicle authentication and information, respectively [2].

To secure the communication system in VANET, various authentication schemes have been proposed in the past. A few techniques utilize the Public Key Infrastructure (PKI) mechanism. The certificate authority (or) trusted authority provides the certificate consisting of the public key, and vehicles use the corresponding private key for signing the messages. Moreover, RSU requires more processing power and storage capacity to verify the certificates [3]. On the other side, a few techniques employ identity information to generate the public key, and use the private key generator to create the user's private key. However, the private key generator in such systems may fabricate the user's private key, known as the escrow issue [4]. The number of intelligent vehicles is continuously increasing in the VANET system. These vehicles frequently learn about road conditions and traffic situations from the other vehicles. Currently, most systems use RSU to authenticate each vehicle individually. When the batch of vehicles (for instance, vehicles from the same cluster) sends the messages to RSU, the RSU sequentially processes each vehicle. As a result, substantial computing power is required for vehicle authentication.

To minimize the computational overhead during message interchange between the vehicles and RSU, many batch verification schemes have been proposed by researchers in the past. Many solutions adopt the centralized architecture, where the centralized cloud servers store the traffic-related information, and trusted authority maintains the parameters required for mutual authentication. These types of solutions cannot overcome the limitations such as Single-Point-of-Failure (SPoF) and scalability, and the system also suffers from trust issues.

Manuscript received 25 June 2022; revised 11 January 2023 and 29 June 2023; accepted 15 August 2023. Date of publication 13 October 2023; date of current version 17 January 2024. This work was supported in part by the "Design and Development of a Unified Blockchain Framework for offering National Blockchain Service" Project, through the Ministry of Electronics and Information Technology, New Delhi, Government of India, under Grant 4(4)/2021-ITEA; and in part by the "Design of Blockchain-Based Authentication and Access Control Protocols in Internet of Vehicles and Internet of Things Deployment" Project through the Ripple Centre of Excellence (CoE) Scheme, CoE in Blockchain, IIIT Hyderabad, India, under Grant IIIT/R&D Office/Internal Projects/001/2019. The Associate Editor for this article was X. Cheng. *(Corresponding authors: Ruhul Amin; Ashok Kumar Das.)*

Sanjeev Kumar Dwivedi is with the VIT-AP School of Computer Science and Engineering (SCOPE), VIT-AP University, Inavolu, Amaravati, Andhra Pradesh 522237, India (e-mail: sanjeevdwivedi131988@gmail.com).

Ruhul Amin and Satyanarayana Vollala are with the Department of Computer Science and Engineering, Dr. Shyama Prasad Mukherjee International Institute of Information Technology, Naya Raipur, Chhattisgarh 493661, India (e-mail: amin_ruhul@live.com; satya4nitt@gmail.com).

Ashok Kumar Das is with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India (e-mail: iitkgp.akdas@gmail.com).

Digital Object Identifier 10.1109/TITS.2023.3310514

In addition, it is not applicable for the scenario where nodes do not trust each other [5]. Therefore, it is an urgent need to develop and implement a decentralized VANET, where the information is distributed across a number of nodes and created an environment where the system provides the inherent trust for disseminating the information. With the adoption of blockchain technology in VANET, it is possible to solve the current issues. It combines encryption, smart contracts, consensus mechanisms, and distributed data storage (such as Interplanetary File System (IPFS)) to produce immutable records that can't be altered and are easily traced by the system's administrator. Blockchain's properties provide data auditability, and nodes are held responsible for their actions. As a consequence, it ensures data security and offers built-in trust for blockchain-enabled applications, such as trusted delivery of drugs from manufacturer to end customer, in the pharmaceutical supply-chain system [6].

A. Motivation

The VANET-based intelligent smart transportation systems are rapidly evolving, providing assurance of road safety and avoiding congestion messages. The intelligent vehicles receive traffic-related messages from other vehicles and send the next advanced capability component like RSU. The authentication of vehicles plays a crucial role in the VANET because it involves the safety of the driver and passengers of the vehicle. In the centralized VANET, the RSU performs the authentication of vehicles with the supervision of the TA and does not admit the messages if vehicles are found to be suspicious.

Initially, the research community proposed schemes that authenticate a single vehicle at a time. Furthermore, these schemes fail to execute the authentication of vehicles in batches when a cluster of vehicles delivers the messages to RSU. Later, they proposed centralized batch authentication schemes where RSU executes the batch operations with the support of TA. In many of these schemes, TA is responsible for generating the batch keys [7], [8], used to perform vehicle batch authentication. Whenever RSU needs to execute batch authentication, it first contacts TA to issue the batch key. If TA is offline, RSU fails to perform the batch authentication of vehicles. Furthermore, the batch key is static. If somehow the batch key has leaked, the functioning of the entire system has been compromised. Therefore, centralized architecture-based batch authentication schemes have trust issues in their system and suffer from SPoF problems.

Thanks to blockchain technology which provides trust and eliminates the SPoF from the system. Only a few researchers utilized blockchain technology [9] to perform vehicle authentication and to achieve a secure system. But, these schemes are prone to various known attacks, including privileged-insider attack, impersonation attack, physical vehicle capture attack, and sybil attack with the amalgamation of high computation costs. Therefore, a novel solution is required to achieve the authentication of the vehicle in batches, which requires minimal energy consumption, and provides an adequate level of security in the VANET. To address the aforementioned issues, we present an efficient batch authentication scheme in VANET using ECC and blockchain technology. In the proposed scheme, RSU executes the batch operations and verifies the group of vehicles which significantly reduces the computation overhead.

B. Our Research Contributions

The main contributions of this paper include the following points.

(1) The numerous phases of the proposed protocol outline the potential of blockchain technology. More specifically, the initial stage includes the development of the blockchain-assisted vehicle pre-authentication protocol. Following that, a protocol for verifying individual messages and batches of messages is established by utilizing the blockchain, ECC, and one-way hash functions.

(2) In the proposed protocol, RSU authenticates the group of vehicles, which is both cost-effective (in terms of computation cost) and efficient. As a consequence, the overall verification time is significantly reduced. Furthermore, blockchain technology assists the proposed protocol in providing a trusted environment for exchanging vital information with other system infrastructures.

(3) The Delov-Yao threat model is used to examine the security of the proposed scheme, and after that, it is independently cross-verified by using both informal and formal means. The results obtained from these techniques demonstrate that the suggested scheme is secure enough against the many different assaults that are currently known, such as man-in-the-middle attacks, impersonation attacks, privileged-insider attacks, and so on.

(4) The proposed scheme is deployed in the Ethereum-based JavaScript VM and Ropsten test-net to determine the actual transaction cost and its deployment time (in terms of consumed GAS, the amount required in USD, and in sec respectively). Further, the computational cost, communication overhead, energy consumption, fundamental functioning features, and security attributes of the proposed scheme are studied.

C. Roadmap of the Paper

The remainder of the paper is organized as follows. In Section II, we present the state-of-the-art schemes for batch verification of vehicles which is further classified into two major categories; centralized and decentralized vehicle's batch verification protocol. In Section III, we provide the proposed scheme, which combines the system model, and various phases of the proposed model, including the pre-authentication phase, message signature generation phase, single and batch message signature verification phase. The informal and formal security analysis of the proposed scheme is presented in Section IV. Whereas, the performance analysis and comparison with the state-of-the-art scheme are discussed in Section V followed by the conclusion and future scope of the paper presented in Section VI.

II. Related Work

This section presents the various existing solutions designed for batch verification of vehicles. The state-of-the-art work is

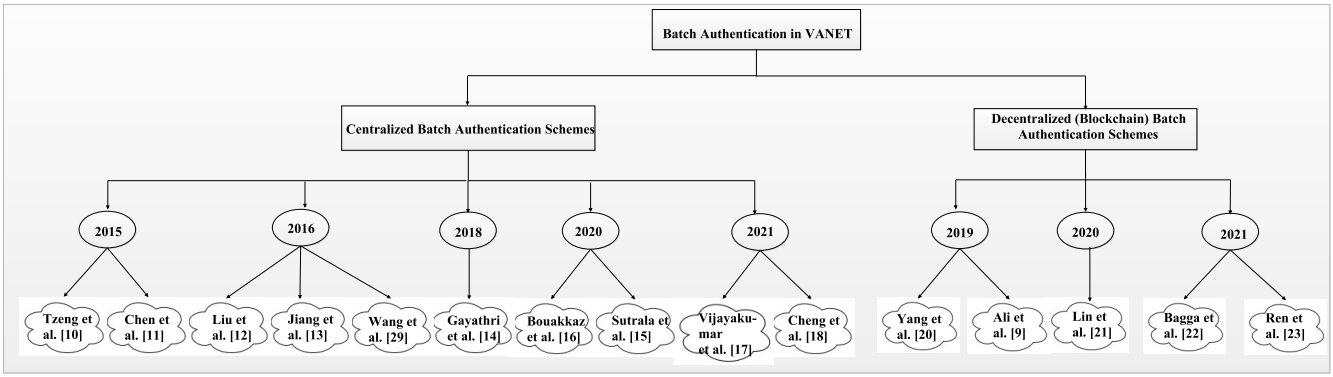

Fig. 1. Classification on batch authentication in the VANET system.

classified into two significant Subsections, based on the nature of the architecture considered by the researchers (See Fig. 1).

A. Centralized Architecture Based Vehicle's Batch Verification Protocol

Initially, researchers proposed a centralized server-assisted system to verify the signatures of both vehicles and RSU. In that system, the centralized server is primarily responsible for generating the key pairs, verifying the signatures (individual (or) batch), and performing the authentication. Various techniques, like identity-based schemes, have been advised to execute the verification of signatures in batches. Regarding this in 2015, Tzeng et al. [10] proposed an identity-based batch verification scheme for VANET, which is further based on the bilinear pairing and point multiplication operations. This model does not consider famous and known attacks such as vehicle physical capturing attacks and impersonation attacks. Furthermore, Chen et al. [11] utilizes the identity-based group signature scheme to reduce the signature verification overhead and to achieve the vehicle's batch verification very efficiently. In 2016, Liu *et al.* [12] suggested the identity-based signature scheme with bilinear pairing for vehicle batch verification. This scheme can retrieve the real identity of vehicles in case of dispute in the message signature (or) message is not valid with the assistance of a trusted authority. A certificate revocation list is one of the solutions for verifying the signatures. This process requires large storage space, and more checking time is needed. In order to address the issue generated by the certificate revocation list, Jiang et al. [13] proposed a solution that is based on the hash message authentication code, and the identity-based group signature scheme is used to achieve the batch verification.

In 2018, Gayathri et al. [14] proposed a bilinear pairing free enabled certificateless batch verification protocol for VANET. This solution does not consider the formal security analysis model AVISPA (or) Scyther to prove the proposed protocol is secure against different types of attacks. In 2020, Sutrala et al. [15] designed the ECC-based conditional privacy-preserving vehicle's batch authentication protocol for the Internet-of-Vehicles (IoV) environment. This protocol employs a centralized structure to store information. As a result, it has trust and SPoF problems. Bouakkaz et al. [16] proposed the certificateless ring signature-based privacy-preserving batch authentication protocol for VANET. The proposed scheme achieves anonymity and traceability and delivers average performance. In 2021, to reduce the authentication overheads on RSU in a high-density traffic area, Vijayakumar et al. [17] proposed a bilinear pairing-based anonymous batch verification and key management scheme for VANET. Although, the formal security model is not considered while designing the scheme. Furthermore, the authors use homomorphic encryption and data aggregation techniques in [18] to protect the privacy of vehicles in the cloud-assisted VANET. They claim that RSU handles 1999 messages in every 300 msec. In 2023, Maurya et al. [19] proposed the anonymous batch authentication scheme for the IoV application by using the concept of bilinear pairing. Further, random oracle model and AVISPA tool are used to verify the security of this scheme.

B. Decentralized Architecture Based Vehicle's Batch Verification Protocol

Although scholars have developed a few batch signature verification schemes by utilizing the centralized framework, all these solutions are undergoing from SPoF problem and having the trust issue. To mitigate it, the research community has been looking at the usage of decentralized frameworks, like blockchain, in the VANET system for the last few years. In 2019, Yang et al. [20] proposed the elliptic curve cryptography (ECC) and blockchain-enabled batch verification scheme for VANET. The trusted authority is a special node in the blockchain that provides the digest of public keys and vehicle identity to the blockchain network. Authors in [9] proposed the certificateless public-key signature scheme with the bilinear pairing in the VANET to achieve conditional privacy and vehicle batch authentication. In this work, the identities of vehicles are verified with the help of proof-of-presence and proof-of-absence protocols. In 2020, the ethereum blockchain-enabled public-key infrastructure-based conditional privacy-preserving authentication protocol for VANET is designed by Lin et al. [21]. They believe that the modified ECDSA method may be utilized for batch verification of vehicles with minimum overhead. In 2021, Bagga et al. [22] proposed the blockchain-based V2V authentication and batch authentication protocol for the IoV-enabled smart city. In this protocol, RSU authenticates

TABLE I
CONTEMPORARY WORKS IN THE VEHICLE'S BATCH VERIFICATION SCHEMES

Works, Year	Problem Statement	Proposed Solution	Cryptographic Features	Limitations
Sutrala *et al.* [15], 2020	Minimize the energy consumption required to verify the messages	Certificate-less conditional privacy-preserving batch verification protocol	Utilized Elliptic curve cryptosystem, hash functions, partial private key generator	Does not provide full transparency and trust in the system
Cheng *et al.* [18], 2021	Focus on privacy protection of feedback vehicles in VANET	Cloud-assisted feedback privacy protection and batch authentication protocol	Utilized paillier encryption, one-way hash function, ECC	The system suffers from SPoF issue.
Yang *et al.* [20], 2019	Trustworthy authority is required to issue the certificate in the certificate-based key agreement protocol	Blockchain-enabled certificateless batch authentication protocol	Utilized blockchain, ECC, one-way hash function	Does not support node accountability and blockchain security simulation solution
Bagga *et al.*[22], 2021	Focus on the scalability, limited computing power, and centralized storage of messages in the IoV	Blockchain and fog-computing enabled batch authentication protocol	Utilized blockchain, ECC, one-way hash function, symmetric encryption	Increases extra communication cost
Ren *et al.* [23], 2021	Focus on the issue of data exchange in an unreliable environment	Blockchain-enabled certificateless privacy-preserving batch verification scheme	Utilized bilinear pairing, hash functions, partial key generator	It is computationally expensive protocol

the cluster of vehicles, and the group key establishes between RSU and vehicles for sharing crucial messages. Ren et al. [23] proposed the privacy-preserving and certificateless-based batch verification scheme for the blockchain-envisioned VANET system. This method uses two blockchains to secure the user's identity and a bilinear-pairing cryptosystem for the authentication of the vehicle. A blockchain-based authentication scheme has been proposed by authors [24]. This scheme allows the RSU to authenticate a foreign vehicle in case of roaming. In this scheme, TA and RSU are interconnected, maintain the blockchain, and finally provide trust for the real-time applications of VANET. A brief summary of a few existing batch authentication schemes, along with their cryptographic features and limitations, is presented in Table I.

III. PROPOSED SYSTEM

This section presents a batch verification model for VANET by utilizing the ECC and hash function. This section first discusses a brief mathematical background of ECC, then presents the system model and various phases involved in the proposed scheme, including the signature generation and verification phase. The various notations utilized in the discussion of the proposed scheme are illustrated in Table II.

A. Mathematical Foundation

Public-key cryptography schemes can be implemented quickly and effectively using Elliptic Curve Cryptography (ECC). It has the potential to provide security, similar to RSA while using a much smaller key size. Elliptic curves are the fundamental building blocks of ECC and are defined as follows: The curve obtained by $y^2 \bmod q = (x^3+ax+b) \bmod q$ is called a non-singular elliptic curve with $4a^3 + 27b^2 \bmod q \neq 0$, where $x, y, a, b \in F_q$ and F_q is a set of points over a prime field. A point P with coordinates (x, y) is represented as $P(x, y)$, and it is an elliptic curve point if it satisfies the equation mentioned above. The negative of $P(x, y)$ is $Q(x, -y)$. Let $P(x_1, y_1)$ and $Q(x_2, y_2)$ are two different points on the elliptic curve. The addition of these two points is another point R on the elliptic curve with $P+Q = R$. Joining point P and point Q intersect the elliptic curve at point $(-R)$, and the reflection of $(-R)$ concerning the x-axis is point R. The scalar point multiplication of point P with scalar value k, i.e., $k \cdot P$ is also a point S on the curve with k times the addition of point P. With this notion, given the values of k and P, it is easy to find the value of $S = k \cdot P$. But, given the values of S and P, it is computationally hard to determine the value of k since there is no technique for solving this problem in polynomial time. This computationally hard problem is known as the elliptic curve discrete logarithm problem (ECDLP). Researchers are thus taking advantage of the ECDLP's potential to propose secure protocols.

B. System Model

The proposed network model for blockchain-enabled batch authentication for the VANET system consists of registration authority RA, roadside units RS_i, $(\forall i = 1, 2, \ldots, n_{rsu})$; and vehicles V_k, $(\forall k = 1, 2, \ldots, n_v)$, as shown in Fig. 2. The RA is responsible for registering RS_i and V_k. It has the sufficient computing power to perform computations and maintain track of transactions in its ledger. The RS_i acts as an intermediary between RA and V_k, and they are deployed alongside the road. The RS_i are responsible for monitoring the suspicious messages gathered by the V_k and verifying the authenticity of V_k. It authenticates V_k individually and in batches, depending on the requirements of the VANET system. The V_k plays a crucial role in the VANET system for sending the information (for instance, traffic-related messages) to the nearest RSU, which finally enhances the safety of passengers and the efficiency of VANET. Every V_k has an OBU, which is responsible for storing sensitive information. Furthermore, RA, RS_i, and V_k are believed to have synchronized their clocks in the VANET system.

In the proposed system, initially, RA broadcasts an enquiry message related to the registration of each deployed RS_i and each new V_k. When the registration is over, RA creates $root_{mht}$ based on the computed parameters and then finally computes the new block. Once this computation is complete, RA locally keeps the index of the block and securely handovers it to corresponding V_k, and each deployed RS_i. RA also constructs a cluster of V_k, which is based on its locality. As a

TABLE II
NOTATIONS AND ITS MEANING

Symbol	Description
RA	Registration Authority
V_k	k^{th} Vehicle
RS_i	i^{th} Roadside unit
(PU_{RA}, s)	Public and private key pair of RA
ID_{RS_i}	Identity of RS_i
(PU_{RS_i}, PR_{RS_i})	Public and private key pair of RS_i
BI_{RS_i}	Block index of RS_i
ID_{V_k}	Identity of V_k
PIN	Pin number (or) secret information of V_k
(r_k, v_k)	random numbers selected by V_k
(e_k, d_k)	Public and private key pair of V_k
$E(\cdot)$	Symmetric encryption
BI_{V_k}	Block index of V_k
B_x	Current block
B_{x-1}	Previous block
$hash_{cur}$	Hash value of current block B_x
$hash_{pre}$	Hash value of previous block B_{x-1}
$root_{mht}$	Merkle root value of current block B_x
N_{B_x}	Nonce value selected for B_x
BT_{B_x}	Time-stamp for B_x
BCN	Blockchain network
m_k	Sensitive information (or) message
$\sigma_k = \{S_{k_1}, S_{k_2}\}$	Signature on m_k
T_k	Time-stamp on m_k
$h(\cdot)$	hash function $h(\cdot):\{0,1\}^* \rightarrow Z_q^*$
$H(\cdot)$	hash function $H(\cdot): G \rightarrow Z_q^*$
G	An additive group G of elliptic curve points with order q
(p, q)	Large prime numbers
P	Generator (or) base point of G
$k \cdot P$	Elliptic curve scalar multiplication $k \cdot P = P + P + ... + P$ (k times)
$\|$	Concatenation operator

results, RS_i can use the BI_{V_k} for individually authenticating V_k. It is worth noting that, in the proposed scheme, RA, V_k, $(\forall k = 1, 2, \ldots, n_v)$ and RS_i, $(\forall i = 1, 2, \ldots, n_{rsu})$; maintains the distributed ledger (i.e., blockchain).

C. System Initialization Phase

The RA is responsible for the initialization of the system. In this phase, RA initializes the essential system parameters and its public and private key pairs. RA selects two large prime numbers p and q, and an additive elliptic cyclic group G with order q. Then RA selects a generator (base point) P of G and a random number $s \in Z_q^*$ as a private key and finally computes the corresponding public key $PU_{RA} = s \cdot P$. RA also selects the one-way cryptographic secure hash functions $H(\cdot) : G \rightarrow Z_q^*$ and $h(\cdot) : \{0,1\}^* \rightarrow Z_q^*$. Next, RA publishes $\langle G, q, P, PU_{RA}, H(\cdot), h(\cdot)\rangle$ as public system parameters and keeps s secret. Note that, the proposed system utilizes different one-way cryptographic secure hash functions based on the different inputs.

D. RSU Registration Phase

The RA performs the registration of RSU, offline through a secure channel. To make the RSU registration process clear, we consider here the registration of i^{th} RSU i.e., RS_i. Initially, RS_i chooses its unique identity ID_{RS_i} and sends $\langle ID_{RS_i}\rangle$ to RA using any confidential channel. Once RA receives $\langle ID_{RS_i}\rangle$, it checks whether RS_i is already registered in the system or not. After this confirmation, RA then computes $PU_{RS_i} = (ID_{RS_i} \cdot s) \cdot P$ and $PR_{RS_i} = H(PU_{RS_i} \cdot s)$ and creates a new block based on the parameters $\langle PU_{RS_i}, ID_{RS_i}\rangle$, and adds this new block to the blockchain network[1] (See Subsection III-F). Once this computation is complete, RA sends $\langle PU_{RS_i}, PR_{RS_i}, BI_{RS_i}\rangle$ to RS_i securely. After getting the $\langle PU_{RS_i}, PR_{RS_i}, BI_{RS_i}\rangle$, RS_i publicly announces PU_{RS_i}, and it securely keeps $\langle PR_{RS_i}, BI_{RS_i}\rangle$. The same procedure is repeated for the registration of all deployed RS_i, $(\forall i = 1, 2, \ldots, n_{rsu})$.

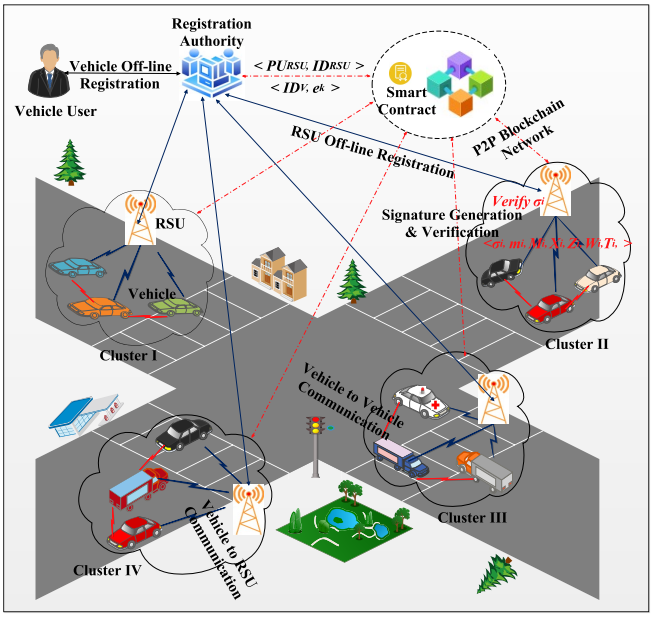

Fig. 2. Systematic architecture of proposed scheme.

E. Vehicle Registration Phase

The vehicle registration phase is performed between vehicles and RA offline through a secure channel. To make the vehicle registration process clear, we consider the registration of k^{th} vehicle, i.e., V_k. Initially, V_k chooses its unique identity ID_{V_k}, PIN number, used to activate OBU, and sends ID_{V_k} to RA. RA then checks ID_{V_k} using some offline document and provides its confirmation regarding the genuine identity ID_{V_k} to V_k. V_k then computes $P_k = r_k \oplus h(PIN)$ and $Q_k = h(ID_{V_k} \| PIN \| r_k)$, where r_k and PIN are random information and secret information of V_k respectively. V_k then randomly selects an integer $d_k \in Z_q^*$ as private key and computes public key $e_k = (d_k \cdot P)$. Once this computation is over, V_k stores $\langle P_k, Q_k, E_{PIN}(d_k)\rangle$ into the memory of OBU, and sends the registration confirmation with the message $\langle ID_{V_k}, e_k\rangle$ to RA. After this, RA creates a new block by considering the parameters $\langle ID_{V_k}, e_k\rangle$ as transaction, adds it to the blockchain network[1] (See Subsection III-F), and securely handovers BI_{V_k} to V_k. Finally, the OBU V_k stores $\langle P_k, Q_k, E_{PIN}(d_k), BI_{V_k}\rangle$ parameters in its memory.

[1]In this work, we have created a block for each non-deployed RSU and vehicle at the time of registration. In the proposed system, every entity maintains the blockchain. As a result, anyone can verify the presence of deployed RSU (or) vehicles. Second, it also prevents the Sybil attack because one entity can acquire almost to one-identity only. The blockchain ensures that no entity has received multiple identities.

F. Block Creation Phase

In this phase, RA creates a block B_x based on the parameters $\langle hash_{pre}, root_{mht}, N_{B_x}, BT_{B_x}\rangle$ as shown in equation 1 [1]. The proof-of-work consensus mechanism is utilized to verify a new block. Moreover, RA utilizes parameters $\langle PU_{RS_i}, ID_{RS_i}\rangle$ (for RS_i) and $\langle ID_{V_k}, e_k\rangle$ (for V_k) for computation of $root_{mht}$ which is further stored in the block header.

$$hash_{cur} = h(hash_{pre} \parallel root_{mht} \parallel N_{B_x} \parallel BT_{B_x}) \quad (1)$$

where $root_{mht}$ is computed as:
$root_{mht} = h(h(PU_{RS_i}) \parallel h(ID_{RS_i}))$ (for RS_i)
& $root_{mht} = h(h(ID_{V_k}) \parallel h(e_k))$ (for V_k)

Initially, N_{B_x} is set to zero and incremented by one at each iteration. The block header values $\langle hash_{pre}, root_{mht}, BT_{B_x}\rangle$ are repeatedly hashed with different values of N_{B_x} for the computation of $hash_{cur}$. Once the creation of a new block is successful, RA hands over the index of new blocks BI_{RS_i} and BI_{V_k} to RS_i and V_k respectively. The same procedure is repeated by RA to all RSU and the vehicles. In our proposed system, Both RA and RSU maintain the distributed ledger blockchain. Therefore, RSU utilizes BI_{V_k} to authenticate the single vehicle (See Subsection III-G).

G. Pre-Authentication Phase

Once all vehicles have registered in the network, they can collect the crucial information m_k and send this information to the nearby RSU. Therefore, before sending the information to RSU, it is required to check the legitimacy of the vehicle's user. The OBU of V_k stores parameters $\langle P_k, Q_k, E_{PIN}(d_k), BI_{V_k}\rangle$ in its memory. The user of the vehicle submits ID'_{V_k} PIN' to OBU. After this, OBU of V_k calculates $r'_k = P_k \oplus h(PIN')$, $Q'_k = h(ID'_{V_k} \parallel PIN' \parallel r'_k)$ and verifies whether $Q'_k = Q_k$ holds or not. If it is incorrect, OBU confirms that the user is not legitimate and rejects the message m_k. Otherwise, it selects $v_k \in Z^*_q$ as a random number, and then computes $G_k = v_k \cdot PU_{RS_i} = (G_{k_x}, G_{k_y})$, $H_k = v_k \cdot P$, $I_k = h(G_{k_x} \parallel ID_{RS_i} \parallel T_1) \oplus v_k$, $J_k = BI_{V_k} \oplus h(v_k)$, $TID_{V_k} = ID_{V_k} \oplus h(G_{k_y} \parallel v_k)$, $U_k = h(BI_{V_k} \parallel G_{k_y} \parallel J_k \parallel T_1)$. Finally, OBU sends parameters $\langle H_k, I_k, J_k, U_k, m_k, TID_{V_k}, e_k, T_1\rangle$ to near-by RSU. It is our assumption here that RS_i is the nearest RSU. Now, RS_i utilizes the blockchain network to complete the partial authentication phase. Algorithm 1 illustrates the steps taken by the RS_i to authenticate the V_k through the blockchain.

H. Message Signature Generation Phase

It is necessary for V_k to sign the messages in order to ensure the integrity of the sensitive information m_k is transmitted over the VANET system. It is explained in Algorithm 2, how to generate the message signature $\sigma_k = \{S_{k_1}, S_{k_2}\}$ using the V_k private key and m_k. Finally, the m_k, together with σ_k, will be broadcasted to RS_i over the public channel.

Algorithm 1 Algorithm for Vehicle's Authentication Through the Blockchain

Require: $\langle H_k, I_k, J_k, U_k, m_k, TID_{V_k}, e_k, T_1\rangle$
Ensure: $\langle yes, no\rangle$

```
begin
calculate G'_k = H_k · PR_RSi = (G'_kx, G'_ky)
calculate v'_k = h(G'_kx ‖ ID_RSi ‖ T_1) ⊕ I_k
calculate BI'_Vk = J_k ⊕ h(v'_k)
calculate ID'_Vk = TID_Vk ⊕ h(G'_ky ‖ v'_k)
calculate U'_k = h(BI'_Vk ‖ G'_ky ‖ J_k ‖ T_1)
if (U'_k,U_k = True) then
    consider the BI'_Vk for the pre-authentication of V_k
end if
if (BI'_Vk exists in BCN = false) then
    abort and stop
else
    root_mht ← fetch(BI'_Vk,BCN)
    compute root'_mht = h(h(ID'_Vk) ‖ h(e_k))
    if (root'_mht,root_mht = True) then
        V_k is legitimate vehicle and authentication of V_k is
        successful
    else
        the authentication of V_k is unsuccessful
    end if
end if
end begin
```

I. Message Signature Verification Phase

In the VANET system, all vehicles are continuously sending messages with their signature to RSU. Therefore, it is necessary to verify the signature before accepting the messages. As explained in Algorithm 3, Once RS_i receives the tuple $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k, N_k, T_k, W_k, Z_k\rangle$, RS_i computes X'_k and W'_k and ensures that m_k has not altered during the transmission in the public channel. After the successful verification, m_k has been stored in the blockchain and can be used for different purposes (for instance, detecting suspicious activity on roads, etc.).

J. Batch Verification Phase

Batch authentication allows a single RSU to verify numerous vehicles simultaneously, saving computational time and enhancing V2I communication performance. Algorithm 4 illustrates the steps involved in the batch verification of messages generated by $\{V_k\}_{k=1,..,n_v}$ corresponding to their signatures $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}\rangle_{k=1,..,n_v}$. To execute the batch verification, initially, each $\{V_k\}_{k=1,..,n_v}$ uploads their signatures to the blockchain to begin the batch verification process. Since RSU are part of the blockchain, they can retrieve the uploaded parameters for Δt time period, check the validity of the time-stamp, and then call the smart contract. The smart contract computes the pre-programmed functions and computed parameters automatically stored in the blockchain. Finally, RSU computes the parameters $\langle A, B, X'\rangle$ based on the aggregated parameters, and accordingly accept (or reject) all messages $\{m_k\}_{k=1,..,n_v}$.

Algorithm 2 Algorithm for Message Signature Generation

Require: $\langle P,\ e_k,\ m_k,\ BI_{V_k},\ T_k\rangle$
Ensure: $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}\rangle$
1: begin
2: randomly selects $r_k \in Z_q^*$
3: calculate $X_k = (BI_{V_k} \cdot r_k) \cdot e_k$, $Y_k = r_k \cdot P$, $Z_k = d_k \cdot e_k$
4: calculate $M_k = h(m_k \parallel BI_{V_k})$
5: calculate $L_k = r_k \oplus h(BI_{V_k} \parallel T_k)$
6: calculate $N_k = h(M_k \parallel BI_{V_k} \parallel ID_{V_k} \parallel T_k)$
7: calculate $S_{k_1} = (X_{k_x} + Y_{k_x}) \bmod q$
8: calculate $S_{k_2} = (M_k + d_k \cdot S_{k_1}) r_k^{-1} \bmod q$
9: calculate $W_k = h(M_k \parallel X_k \parallel Y_k \parallel ID_{RS_i} \parallel T_k)$
10: broadcast $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k,\ N_k,\ T_k,\ W_k,\ Z_k\rangle$ to RS_i
11: end begin

Remarks: (X_{k_x}, X_{k_y}) and (Y_{k_x}, Y_{k_y}) are the x and y coordinates of the elliptic curve point X_k and Y_k respectively.

Algorithm 3 Algorithm for Message Signature Verification

Require: $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k,\ N_k,\ T_k,\ W_k,\ Z_k\rangle$
Ensure: $\langle success,\ failure\rangle$
1: begin
2: **if** (validity of T_k is expired) **then**
3: discard m_k and abort the operation
4: **else**
5: calculate $M'_k = h(m_k \parallel BI'_{V_k})$
6: calculate $N'_k = h(M'_k \parallel BI'_{V_k} \parallel ID_{V_k} \parallel T_k)$
7: **if** (N'_k,N_k = True) **then**
8: calculate $r'_k = L_k \oplus h(BI'_{V_k} \parallel T_k)$
9: calculate $X_k = (BI'_{V_k} \cdot r'_k) \cdot e_k$
10: calculate $Y'_k = r'_k \cdot P$
11: **else**
12: discard m_k and abort the operation
13: **end if**
14: calculate $A_k = M'_k \cdot S_{k_2}^{-1} \bmod q$
15: calculate $B_k = S_{k_2}^{-1} \cdot S_{k_1} \bmod q$
16: calculate $X'_k = BI'_{V_k}(A_k \cdot e_k + B_k \cdot Z_k)$
17: **if** ($X'_k = X_k$) **then**
18: calculate $W'_k = h(M_k \parallel X'_k \parallel Y'_k \parallel ID_{RS_i} \parallel T_k)$
19: **if** ($W'_k = W_k$) **then**
20: Successful verification of V_k and store m_k in blockchain
21: **else**
22: signature verification fails and stop further processing
23: **end if**
24: **end if**
25: return error
26: **end if**
27: end begin

Remarks: RS_i computes BI'_{V_k} in the pre-authentication phase (Algorithm 1) of V_k.

K. Correctness Proof

The correctness proof of single message signature verification and batch message signature verification, which are illustrated in the Section III-I and III-J are presented here.

1) Correctness Proof of Single Message Signature Verification: We have $A_k = M_k \cdot S_{k_2}^{-1}$ and $B_k = S_{k_2}^{-1} \cdot S_{k_1}$. We then have $BI_{V_k}(A_k \cdot e_k + B_k \cdot Z_k) = BI_{V_k}(M_k \cdot S_{k_2}^{-1} \cdot e_k + S_{k_2}^{-1} \cdot S_{k_1} \cdot d_k \cdot e_k) = BI_{V_k} \cdot S_{k_2}^{-1} \cdot e_k(M_k + S_{k_1} \cdot d_k) = (M_k + S_{k_1} \cdot d_k)^{-1} \cdot BI_{V_k} \cdot r_k \cdot e_k(M_k + S_{k_1} \cdot d_k) = (BI_{V_k} \cdot r_k) \cdot e_k = X_k$. Hence it follows that $X_k = BI_{V_k}(A_k \cdot e_k + B_k \cdot Z_k)$.

2) Correctness Proof of Batch Message Signature Verification: We have $A = M \cdot S_2^{-1}$ and $B = S_2^{-1} \cdot S_1$. We then have $BI_V(A \cdot e + B \cdot Z) = BI_V(M \cdot S_2^{-1} \cdot e + S_2^{-1} \cdot S_1 \cdot Z) = \sum_{k=1}^{n}((BI_{V_k} \cdot r_k) \cdot e_k(M_k + S_{k_1} \cdot d_k)^{-1}(M_k + S_{k_1} \cdot d_k)) = \sum_{k=1}^{n}((BI_{V_k} \cdot r_k) \cdot e_k) = \sum_{k=1}^{n} X_k$. Hence it follows that $\sum_{k=1}^{n} X_k = \sum_{k=1}^{n} BI_{V_k}(A_k \cdot e_k + B_k \cdot Z_k)$.

IV. Security Analysis and Verification

This Section presents the informal security analysis of the proposed scheme, which shows that the method is robust and resilient against the different types of known security attacks. In addition to it, the formal security verification of the proposed model using the Scyther tool is also presented in this Section.

A. Adversary Model

The proposed scheme employs the Delov-Yao (DY) threat model. According to this threat model, our valid assumption is that all the participating entities of the system utilize unsecured public channels for their communication. However, an adversary has the following capabilities to launch various attacks.

- The adversary can eavesdrop on all transmitted messages over the public channel, which are exchanged between the vehicles and RSU. However, messages transmitted over secure channels cannot be intercepted by the attacker.
- The adversary has the capability to modify (or insert false information) and deletes the eavesdropped messages.
- The adversary can seize the vehicles physically and control them to get the private information stored in them.
- The adversary cannot extract the private key from the known public key.
- It is very likely for the adversary to guess one information at a time. But, it is computationally infeasible to guess the two secret information in polynomial time.
- The adversary knows the protocols used in the communication between the vehicles and RSU.

B. Informal Security Analysis of Known Attacks

Proposition 1: The proposed protocol is secure against the replay attack.

Proof: During the signature verification phase between the vehicles V_k and RS_i, the messages $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k, N_k,\ T_k,\ W_k,\ Z_k\rangle$ is sent over the public channel by V_k, is attached with time-stamp T_k. When RS_i receives $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k,\ N_k,\ T_k,\ W_k,\ Z_k\rangle$, it checks the freshness of the message by validating T_k, as presented in

Algorithm 4 Algorithm for Batch Message Signature Verification

Require: $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k, N_k, T_k, W_k, Z_k \rangle_{(k=1,..,n_v)}$
Ensure: $\langle success, failure \rangle$

```
begin
for (each m_k) do
    if (validity of T_k is expired) then
        discard m_k with ⟨σ_k = {S_k1, S_k2}⟩ and abort the operation
    else
        compute M_k, X_k, BI_Vk and continue to collect the messages with signatures for Δt time-period
    end if
    SC aggregates signatures and computes S_1 = Σ_{k=1}^n S_k1, S_2 = Σ_{k=1}^n S_k2, X = Σ_{k=1}^n X_k, M = Σ_{k=1}^n M_k, e = Σ_{k=1}^n e_k, BI_V = Σ_{k=1}^n BI_Vk, Z = Σ_{k=1}^n Z_k
    SC uploads ⟨S_1, S_2, X, M, e, Z⟩ to blockchain
    RSU downloads ⟨S_1, S_2, X, M, e, Z⟩ from blockchain
    compute A = M · S_2^{-1} mod q
    compute B = S_2^{-1} · S_1 mod q
    compute X' = BI_V(A · e + B · Z)
    if ( X' = X) then
        accept all signatures ⟨σ_k = {S_k1, S_k2}⟩_{k=1,..,n_v} and store each {m_k}_{k=1,..,n_v} in blockchain
    else
        signature verification fails and drop all messages {m_k}_{k=1,..,n_v}
    end if
end for
end begin
```

Section III-I & III-J. Therefore, the presence of T_k prevents V_k for transmitting the same message in the realm of RSU. Hence, the proposed scheme is resilient against the replay attack.

Proposition 2: The proposed protocol is secure against the man-in-the-middle attack.

Proof: Assuming that an adversary A eavesdrops the signature message $\langle \sigma_k = \{S_{k_1}, S_{k_2}\}, m_k, L_k, N_k, T_k, W_k, Z_k \rangle$ generated by V_k, and then it tries to make another legitimate signature σ'_k message on the top of it such that RSU can not detect the modified message. In order to do this, A may create parameter X'_k and picks time-stamp T'_k. But without the knowledge of pre-selected values d_k and r_k, it is infeasible for A to compute the valid $\langle X'_k, Y'_k, Z'_k \rangle$ due to the hardness problem of elliptic curve, and finally A can not produce a valid signature σ'_k. Hence, the proposed scheme is resilient against the man-in-the-middle attack.

Proposition 3: The proposed protocol is secure against the impersonation attack.

Proof: Suppose adversary A tries to behave as a legitimate vehicle V_k, and then it tries to produce a valid signature σ'_k with message parameters $\langle M'_k, X'_k, T'_k, Z'_k \rangle$. To acquire its goal, A picks a random number $r'_k \in Z_q^*$ and time-stamp T'_k, and then A computes $X'_k = r'_k \cdot e_k$, $Y'_k = r'_k \cdot P$. But without the knowledge of secret credentials d_k, it is computationally infeasible task to create a valid S'_{k_2} and Z'_k under the ECDLP assumption. Moreover, block index BI_{V_k} has very securely stored in the OBU of V_k. As a result, without the knowledge of BI_{V_k}, A cannot produce a valid M'_k. Hence, the proposed scheme is resilient against the impersonation attack.

Proposition 4: The proposed protocol is secure against the privileged-insider attack.

Proof: In the vehicle enrollment phase, vehicle V_k does not send any kind of secret credentials (for example, private key and pin number) to registration authority RA, except $\langle ID_{V_k}, e_k \rangle$. Based on the parameters $\langle ID_{V_k}, e_k \rangle$, RA creates a new block B_{V_k} and handovers the index of B_{V_k} (i.e., BI_{V_k}) to V_k, and then RA deletes BI_{V_k} from its database or memory. It is noted that the execution of the new block is done only once by RA, prior to the deployment of all vehicles as illustrated in Section III-E. In the proposed system, all the entities of the network maintain the same copy of the ledger. Therefore, as a privileged-insider user, if the RA tries to modify the payload information of any block, it must modify the entire ledger of blockchain, starting from block index BI_{V_k} up to the last block, which is computationally not feasible for RA. Hence, we conclude that the proposed scheme is resilient against the privileged-insider attack.

Proposition 5: The proposed protocol is secure against the physical vehicle capture attack.

Proof: In this attack, the adversary A may physically seize the vehicles because of the antagonistic environment. A utilizes the power analysis attacks [25] for extracting the stored parameters $\langle P_k, Q_k, E_{PIN}(d_k), BI_{V_k} \rangle$ from the OBU of the compromised vehicle. Since, all stored credentials $\langle P_k, Q_k, E_{PIN}(d_k), BI_{V_k} \rangle$ $(\forall k = 1, 2, \ldots, n_v)$ in the OBU of every vehicle is unique, and distinct from non-compromised vehicles. As a result, V_k secret credentials are not helpful for generating the signatures $\langle \sigma_i = \{S_{i_1}, S_{i_2}\} \rangle$ of other V_i. Hence, the proposed scheme is resilient against the physical vehicle capture attack.

Proposition 6: The proposed protocol is secure against the off-line PIN guessing attack.

Proof: The PIN information enter in the OBU by the user of vehicle is a sensitive information. Therefore, it must be kept very securely so that an adversary cannot extract the PIN information during the execution of the protocol. In the proposed protocol, an OBU stores parameters $\langle P_k, Q_k, E_{PIN}(d_k), BI_{V_k} \rangle$ in its memory, where $P_k = r_k \oplus h(PIN)$ and $Q_k = h(ID_{V_k} \parallel PIN \parallel r_k)$. It is clear that the obtained parameters $\langle P_k, Q_k \rangle$ are non-invertible because of the inclusion of one-way hash functions. Moreover, if an adversary tries to guess PIN information off-line, s/he has to guess two unknown parameters $\langle ID_{V_k}, r_k \rangle$ at the same time which is not feasible in polynomial time, as the probability is $\frac{1}{2^{6+n}}$ for guessing the parameters $\langle ID_{V_k}, r_k \rangle$ where $n-bits$ is the length of r_k. Hence, we conclude that the proposed scheme is secure against the off-line PIN guessing attack.

Proposition 7: The proposed protocol is secure against the sybil attack.

Proof: Sybil attack allows an adversary A to create numerous fraudulent identities, and based on this, A tries to control the decentralized network and can conduct the different attacks

(for instance, refuse the transactions, etc.). In the vehicle enrollment phase, initially RA checks the identity of V_k (i.e. ID_{V_k}) using its own database, and then RA provides confirmation to V_k for its genuine identity ID_{V_k}. Afterwards, RA creates a new block B_x by considering $\langle ID_{V_k}, e_k \rangle$ as a new transactions, and permanently stores BI_{V_k} to the blockchain network. The BI_{V_k} is further utilized by RSU in the vehicle's partial authentication phase, which is illustrated in Section III-G. Therefore, the probability of getting multiple identities is negligible. Hence, we conclude that the proposed scheme is secure against the sybil attack.

Proposition 8: The proposed protocol is secure against the blockify attack.

Proof: In this attack, adversary A tries to enroll in the private blockchain, captures the existing blocks from the blockchain network, and then modifies (or) deletes the block information. All the transactions in the block payload are encrypted and signed. Updation in a single block parameter requires modification in all the subsequent blocks, which requires a lot of computation power and time. In the limited time-period and resources, it is impossible to modify the ledger of a single node and then synchronize with all other peer nodes. Hence, there is a negligible probability of a blockify attack in the proposed scheme.

Proposition 9: The proposed scheme achieves the verification of a block.

Proof: In the proposed scheme, the verification of block requires to recompute the merkle root $root'_{mht}$ based on all encrypted transactions from block payload. If $root'_{mht} = root_{mht}$ is correct, the verifier further recomputes the $hash'_{cur}$ based on the block header information and block payload. The verifier checks the correctness of $hash'_{cur}$ with $hash_{cur}$. Once this correctness is completed, the verifier finally validates the block signature using the ECDSA signature verification algorithm [26]. If all validation results are positive, the verifier considers this block as a legitimate block.

C. Formal Security Verification Under Scyther Tool

Before deploying the security protocols in real networks, it is necessary to examine the strength of the security provided by the protocols. To achieve it, the proposed protocol is simulated by using the Scyther simulator, which formally proves that the protocol is secure from all types of possible attacks. Scyther has recently gained prominence in checking and analyzing security protocols, and it is noted for its enhanced features, and excellent performance [27]. Figure 2 demonstrates the communication process among vehicles, RSU and RA are secure, and the secret parameters are not revealed during their communication. The suggested protocol is simulated several times in various environments, and each simulation shows that there are no attacks within the specified bounds. The simulation finding reveals that the noninjective agreement (Ni-Agree) and noninjective synchronization (Ni-Synch) are met. The Ni-Agree asserts that the communication parties agree on variable values that are transferred between them, and the results of the analysis confirm that this assertion is accurate. The Ni-Synch property necessitates the execution of the relevant sending and receiving events by the runs specified by the cast function, and it is implemented in the proper sequence.

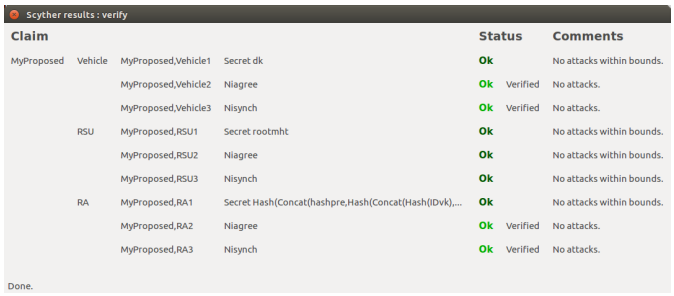

Fig. 3. Scyther output for vehicle, roadside unit and registration authority communication.

V. Performance Analysis and Ethereum Implementation

This section compares the proposed scheme's performance with state-of-the-art research in terms of computation and communication costs.

A. Functionality Features and Security Attributes Comparison

In Table III, the proposed scheme is compared with the state-of-the-art research based on the functionality attributes, including ledger distribution, decentralization, accountability, anonymity, block verification, and various security attacks such as a privileged-insider attack, impersonation attack, sybil attack, etc. In this work, ledger distribution means that every peer node maintains the same ledger. The anonymity means that the identity of the vehicles is kept anonymous while sharing the information. Accountability means that the nodes are held accountable for their operations. In Yang et al. [20], Bagga et al. [22], Ren et al. [23], and Ali et al. [9] schemes utilize the blockchain in their solutions, but in all these schemes sybil attack, blockify attack, and physical vehicle capturing attack are not provided. In contrast to these schemes, the proposed solution leverages the blockchain to provide the decentralized batch authentication of vehicles and resolves all kinds of possible attacks. Overall, this comparison table provides the insightful functionality of the proposed scheme and achieves the required security features compared to the existing state-of-the-art research.

B. Communication Cost Comparison

For the calculation of communication cost, a message signature and the other relevant parameters along with the message are transmitted from the vehicle to RSU. The proposed scheme utilizes the one-way cryptographic hash function that takes an input of arbitrary size $\{0,1\}^*$ and produces an output of a fixed length of size 256 bits. We also consider the 160 bit ECC because the 160 bit ECC provides the same level of security as that of the 1024 bit RSA cryptosystem. Therefore, the communication overhead for sending a elliptic curve point $P = (P_x, P_y)$ is (160 bits + 160 bits)= 320 bits. Furthermore,

TABLE III
COMPARATIVE STUDY ON FUNCTIONALITY FEATURES AND SECURITY ATTRIBUTES

Authors	BC/NBC	FS_1	FS_2	FS_3	FS_4	FS_5	FS_6	FS_7	FS_8	FS_9	FS_{10}	FS_{11}	FS_{12}	FS_{13}	FS_{14}	FS_{15}
Sutrala *et al.* [15]	NBC	×	×	×	✓	✓	✓	×	✓	✓	×	×	×	×	×	✓
Cheng *et al.* [18]	NBC	×	×	×	✓	✓	✓	×	✓	✓	×	×	×	×	×	×
Bouakkaz *et al.* [16]	NBC	×	×	×	✓	✓	✓	×	×	×	×	×	×	×	×	×
Yang *et al.* [20]	BC	✓	✓	✓	✓	✓	✓	×	✓	✓	×	✓	×	×	✓	×
Bagga *et al.* [22]	BC	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	✓	✓
Ren *et al.* [23]	BC	✓	✓	✓	✓	✓	✓	×	✓	✓	×	×	×	×	✓	×
Ali *et al.* [9]	BC	✓	✓	✓	✓	✓	✓	×	✓	✓	×	×	×	×	×	×
Lin *et al.* [21]	BC	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	×	×	×	✓	×
Gayathri *et al.* [14]	NBC	×	×	×	✓	✓	✓	×	×	×	×	×	×	×	×	×
Vijayakumar *et al.* [17]	NBC	×	×	×	✓	✓	✓	✓	✓	✓	×	✓	×	×	×	×
Tzeng *et al.*[10]	NBC	×	×	×	✓	✓	✓	×	✓	×	×	✓	×	×	×	×
Chen *et al.* [11]	NBC	×	×	×	✓	✓	×	×	×	×	×	×	×	×	×	×
Cui et al. [28]	NBC	×	×	×	✓	✓	✓	×	×	×	×	✓	×	×	×	×
Liu *et al.* [12]	NBC	×	×	×	✓	✓	✓	×	✓	✓	×	✓	×	×	×	×
Jiang *et al.* [13]	NBC	×	×	×	✓	✓	✓	×	✓	×	×	×	×	×	×	×
Wang *et al.*[29]	NBC	×	×	×	✓	✓	✓	×	✓	×	×	×	×	×	×	×
Zhou *et al.*[30]	NBC	×	×	×	×	✓	✓	×	×	×	×	×	×	×	×	×
Alazzawi *et al.* [31]	NBC	×	×	×	✓	✓	✓	×	✓	✓	×	✓	×	×	×	×
Wang *et al.* [32]	NBC	×	×	×	✓	✓	✓	×	✓	×	×	✓	×	×	×	×
Maria *et al.* [24]	BC	✓	✓	✓	✓	✓	✓	×	✓	✓	×	×	×	×	×	×
Proposed scheme	BC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: FS_1 : ledger distribution; FS_2 : decentralized; FS_3 : transparency; FS_4 : accountability; FS_5 : node message authentication; FS_6 : replay attack; FS_7 : privileged-insider attack; FS_8 : main-in-the-middle attack; FS_9 : impersonation attack; FS_{10} : physical vehicle capture attack; FS_{11} : off-line PIN guessing attack. FS_{12} : sybil attack; FS_{13} : blockify attack; FS_{14} : block verification in blockchain; FS_{15} : formal security verification using Scyther tool; ✓ : the scheme supports the feature or secure against attack; × : the scheme does not support the feature (or) security solution against attack is not presented in the state-of-the-art research. BC: The scheme has adopted blockchain based security solution; NBC: The scheme has adopted non-blockchain based security solution.

we also consider the length of vehicle identity, a random nonce, time-stamp, block index, and other relevant parameters such as $\langle A, B\rangle$ takes 160 bit, 160 bit, 32 bit, 256 bit, 256 bit, 256 bit Respectively. In order to provide the comparison with the existing schemes, we also assumed that the message m_k transmitted by vehicles is a length of 160 bits. The V_k transmits the m_k along with the tuple $\langle \sigma_k, L_k, N_k, T_k, W_k, Z_k\rangle$ to RS_i where $\sigma_k = \{S_{k_1}, S_{k_2}\}$ is the signature of the message as illustrated in Section III-J. The total communication cost incurred to send these parameters is (160 bits + 160 bits + 160 bits + 256 bits + 256 bits + 32 bits + 256 bits + 320 bits) = 1600 bits. In the proposed scheme, 1600 bits are required to transmit a single message from V_k to RS_i. Therefore, the total communication cost to send the n number of messages from V_k $(\forall k = 1, 2, \ldots, n_v)$ to RS_i is $1600n$.

The communication overhead of Gayathri et al. [14] scheme, Zhou et al. [30] scheme, Sutrala et al. [15] scheme, and Bagga et al. [22] are 1984 bits, 2656 bits, 1984 bits, 2912 bits respectively. The solution provided by the schemes in [14], [15], and [30] requires more communication cost than the proposed scheme, whereas the scheme in [22] utilizes the blockchain mechanism to offer the batch authentication of vehicles. However, this scheme requires more communication cost than the scheme proposed in this work and [30]. The overall comparison of the communication cost with different schemes is presented in Table IV. From Table IV, It is very clear that only a few of the schemes utilize the blockchain to achieve the transparent and decentralized batch authentication of vehicles, while other schemes adopted the centralized system architecture for vehicle batch authentication and offer high communication costs. On the other hand, the proposed scheme utilizes the blockchain to achieve the decentralized batch authentication of vehicles and offers a comparable communication cost compared with the state-of-the-art research, which is shown in Fig. 4.

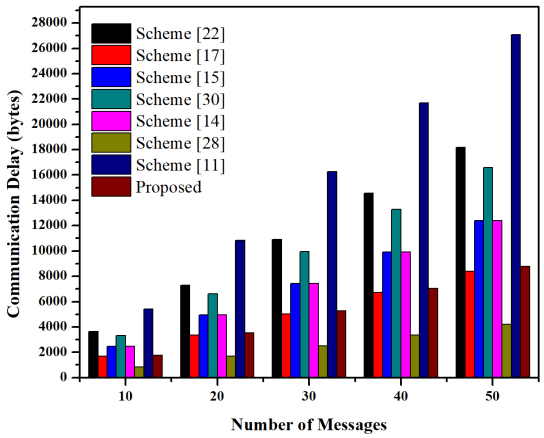

Fig. 4. Comparison of communication delay with the relevant schemes.

C. Computation Cost Comparison

In order to compute and compare with the existing schemes, we considers the existing results mentioned in [3] and [15] for the approximated time required by the various cryptographic primitives. In this work, we denote T_h, T_{ecm}, T_{eca}, T_{bp}, T_{ex}, T_{mi} and T_{mtp} as the time required for one-way hash function, elliptic curve scalar multiplication, elliptic curve point addition, bilinear pairing operation, exponentiation operation, modular inverse and map-to-point operation respectively. We have $T_h \approx 0.32$ ms, $T_{ecm} \approx 17.10$ ms, $T_{eca} \approx 4.4$ ms, $T_{bp} \approx 42.11$ ms, $T_{ex} \approx 19.20$ ms, $T_{mi} \approx 2.64$ ms, and $T_{mtp} \approx 44.06$ ms. The proposed scheme takes three scalar point multiplication and four hash computation for generating a signature on message m_k. Therefore, the computation time required for this is $3\ T_{ecm} + 4\ T_h \approx 52.58$ ms. On the other hand, for signature verification, this scheme requires four scalar point multiplication, one elliptic curve point addition, and four hash computation which requires $4\ T_{ecm} + T_{eca} + 4$

TABLE IV
COMMUNICATION COST (BITS) ANALYSIS OF PROPOSED WORK AND STATE-OF-THE-ART RESEARCH

Ref.	BC/NBC	Sending single message	sending n messages
Sutrala *et al.* [15]	NBC	1984	$1984n$
Bouakkaz *et al.* [16]	NBC	408	$408n$
Bagga *et al.* [22]	BC	2912	$2912n$
Ali *et al.* [9]	BC	536	$536n$
Gayathri *et al.* [14]	NBC	1984	$1984n$
Vijayakumar *et al.* [17]	NBC	1344	$1344n$
Tzeng *et al.* [10]	NBC	536	$536n$
Chen *et al.* [11]	NBC	4336	$4336n$
Cui *et al.* [28]	NBC	672	$672n$
Liu *et al.* [12]	NBC	760	$760n$
Jiang *et al.* [13]	NBC	736	$736n$
Wang *et al.* [29]	NBC	832	$832n$
Zhou *et al.* [30]	NBC	2656	$2656n$
Alazzawi *et al.* [31]	NBC	864	$864n$
Wang *et al.* [32]	NBC	1216	$1216n$
Maurya *et al.* [19]	NBC	1472	$1472n$
Maria *et al.* [24]	BC	1632	$1632n$
Feng *et al.* [33]	NBC	6144	$6144n$
Proposed	BC	1600	$1600n$

BC: The scheme has adopted blockchain based security solution; NBC: The scheme has adopted non-blockchain based security solution.

$T_h \approx 74.08$ ms. As a result, it requires $7\ T_{ecm} + T_{eca} + 8\ T_h \approx 126.66$ ms to generate a signature on a single message and its further verification. Moreover, to verify the signature of n number of messages in batch, the proposed scheme requires $2\ T_{ecm} + (3n+1)\ T_{eca} \approx 38.6 + 13.2n$ computational time.

In the Table V, we have compared the computation delay in branch verification of the proposed scheme with the existing state-of-the-art research. Bagga et al.'s scheme [22] offers the vehicle's batch authentication protocol with the consideration of blockchain. This scheme utilizes bilinear pairing and ECC cryptosystem to achieve it, and the total computational delay in batch verification is $[3\ T_{bp} + 5n\ T_{ecm} + (3n+1)\ T_{eca} + (2n+1)\ T_h] \approx 131.05 + 99.34n$. In Alazzawi et al. [31] and Wang et al. [32] scheme, batch verification delay is $2(n+1)\ T_{ecm} + (n+1)\ T_{eca} + n\ T_h \approx 38.6 + 38.92n$ and $T_{ecm} + T_{mtp} + n\ T_h + 3\ T_{bp} \approx 187.49 + 0.32n$ respectively. These schemes have not utilized the blockchain in their solution, and offers a more computational time than the our proposed scheme. With the close analysis and comparison with the existing schemes as shown in Fig. 5, it is worth noticing that our scheme achieves a less (or) comparable computational time with the consideration of decentralized blockchain-based system for vehicle's batch verification.

D. Energy Consumption Comparison

In order to conduct all of the computations while exchanging the messages from vehicles to RSU, some energy is needed. The efficiency of the proposed solutions is also affected by their power usage. Hence, equation 2 is utilized to compare the existing and proposed systems in terms of energy usage. The energy consumption is calculated by [34]:

$$EC = T_{delay} * max_{processing} \tag{2}$$

where EC is the total energy consumption (in MJ) required by the RSU to verify the vehicle's in batches, T_{delay} is the time required for the vehicle's batch verification (in ms), and $max_{processing}$ is the maximum processing power required in wireless transmission, which is considered as 10.88 W [35]. The value of T_{delay} is taken from Table V. Table VI illustrates the total energy consumption required by the proposed scheme, and the same is compared with the state-of-the-art research. From Fig. 6, it is evident that the proposed scheme requires comparable energy and can be used in the real-time implementation of the vehicle's batch verification.

TABLE V
COMPUTATION COST ANALYSIS OF PROPOSED WORK AND STATE-OF-THE-ART RESEARCH

Schemes	BC/NBC	Computational delay in batch verification	Verification time (miliseconds)
Sutrala *et al.* [15]	NBC	$(2n+1)\ T_{eca} + 5\ T_{ecm} + 3\ T_h$	$90.86+8.8n$
Bouakkaz *et al.* [16]	NBC	$2\ T_{bp} + T_{ecm} + (2n+1)\ T_{eca}$	$105.72 + 8.8n$
Bagga *et al.* [22]	BC	$[3\ T_{bp} + 5n\ T_{ecm} + (3n+1)\ T_{eca} + (2n+1)\ T_h]$	$131.05 + 99.34n$
Ali *et al.* [9]	BC	$T_{bp} + n\ T_{ecm} + n\ T_{eca}$	$42.11 + 21.5n$
Gayathri *et al.* [14]	NBC	$3(n-1)\ T_{eca} + n\ T_{ecm} + n\ T_h + 3\ T_{bp}$	$121.93 + 30.62n$
Vijayakumar *et al.* [17]	NBC	$n\ (T_{ecm} + T_{bp})$	$59.21n$
Tzeng *et al.* [10]	NBC	$n+1\ (T_{ecm} + 2\ T_{bp})$	$101.32(n+1)$
Chen *et al.* [11]	NBC	$12n\ T_{ecm} + 3\ T_{bp} + 16n\ T_{ex}$	$126.33 + 512.4n$
Cui *et al.* [28]	NBC	$(n+2)\ T_{ecm} + n\ T_{eca} + n\ T_h$	$672 \cdot n$
Liu *et al.* [12]	NBC	$(n+1)\ T_{ecm} + 2\ T_{bp}$	$34.2 + 21.82n$
Jiang *et al.* [13]	NBC	$(n+1)\ T_{ecm} + 3\ T_{bp}$	$143.43 + 17.10n$
Wang *et al.* [29]	NBC	$2\ T_{bp} + (2n+1)\ T_{ecm} + n\ T_{mtp}$	$101.32 + 78.26n$
Zhou *et al.* [30]	NBC	$2(n-1)\ T_{bp} + (n+1)\ T_{ecm} + n\ T_h$	$101.64n - 67.12$
Alazzawi *et al.* [31]	NBC	$2(n+1)\ T_{ecm} + (n+1)\ T_{eca} + n\ T_h$	$38.6 + 38.92n$
Wang *et al.* [32]	NBC	$T_{ecm} + T_{mtp} + n\ T_h + 3\ T_{bp}$	$187.49 + 0.32n$
Maurya *et al.* [19]	NBC	$3\ T_{bp} + (3n+2)\ T_{ecm} + 2n\ T_h + n\ T_{mi}$	$160.53 + 54.58n$
Maria *et al.* [24]	BC	$4n\ T_{ecm} + n\ T_{eca} + n\ T_{bp}$	$114.91 \cdot n$
Feng *et al.* [33]	NBC	$4\ T_{bp} + (6n-1)\ T_{ecm} + 2n\ T_h + (4n+2)\ T_{ex}$	$189.74 + 180.04n$
Proposed scheme	BC	$2\ T_{ecm} + (3n+1)\ T_{eca}$	$38.6+13.2n$

BC: The scheme has adopted blockchain based security solution; NBC: The scheme has adopted non-blockchain based security solution.

E. Implementation on Ethereum and Analysis of Gas Cost

Remix IDE[2], an open-source Ethereum test network, is utilized to implement the suggested method. Solidity[3], a Java scripting language intended specifically for creating contract code between peer nodes, is supported by the Remix IDE. Contract code is executed with the help of Solidity Compiler $0.5.17+commit.d19bba13$. The transactions are deployed and operated using JavaScript VM and injected Web3 (such as the Ropsten test-net) to demonstrate the efficacy and viability of our technique. Using the Ropsten test-net, developers may

[2]https://remix.ethereum.org/
[3]https://docs.soliditylang.org/

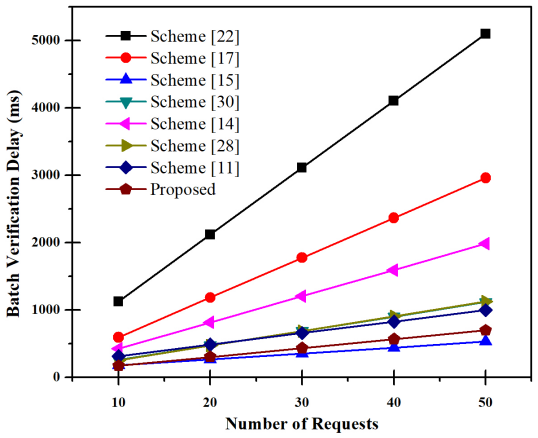

Fig. 5. Comparison of batch verification delay with the relevant schemes.

TABLE VI
ENERGY CONSUMPTION (IN MJ) OF PROPOSED WORK AND STATE-OF-THE-ART RESEARCH

Ref.	BC/NBC	Energy consumption
Sutrala *et al.* [15]	NBC	99.66
Bouakkaz *et al.* [16]	NBC	114.52
Bagga *et al.* [22]	BC	230.39
Ali *et al.* [9]	BC	63.61
Gayathri *et al.* [14]	NBC	152.55
Vijayakumar *et al.* [17]	NBC	59.21
Tzeng *et al.* [10]	NBC	202.64
Chen *et al.* [11]	NBC	638.73
Cui *et al.* [28]	NBC	56.02
Liu *et al.* [12]	NBC	118.42
Jiang *et al.* [13]	NBC	160.53
Wang *et al.* [29]	NBC	179.58
Zhou *et al.* [30]	NBC	34.52
Alazzawi *et al.* [31]	NBC	77.52
Wang *et al.* [32]	NBC	187.81
Proposed scheme	BC	51.8

BC: The scheme has adopted blockchain based security solution; NBC: The scheme has adopted non-blockchain based security solution.

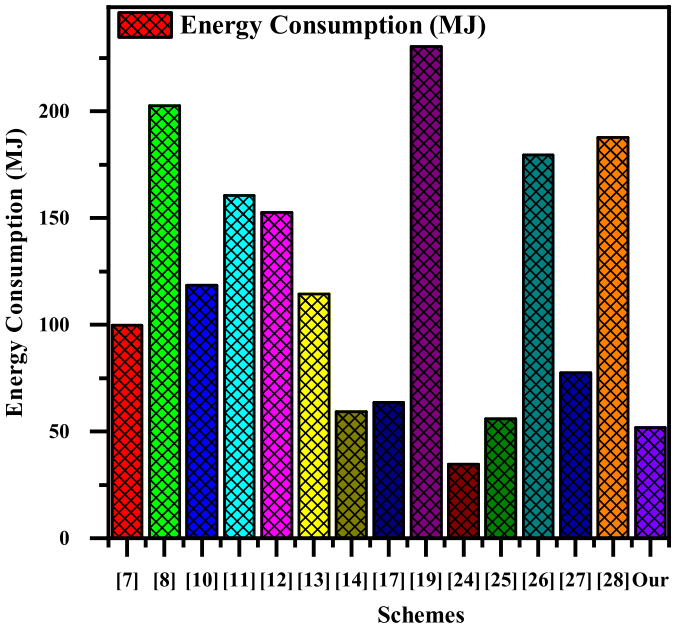

Fig. 6. Comparison of energy consumption with the relevant schemes.

simulate real-world ethereum network transactions without spending real money (ETH). To generate the fictitious ethers for the test-net account address, this simulation makes use

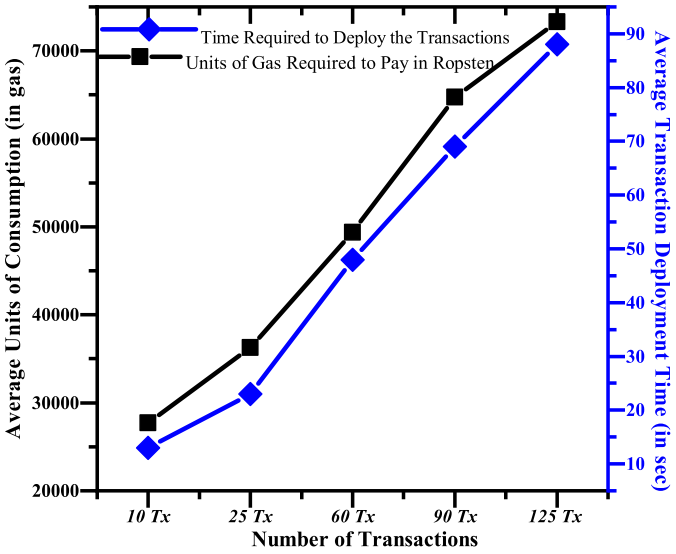

Fig. 7. Average gas consumption versus deployment time.

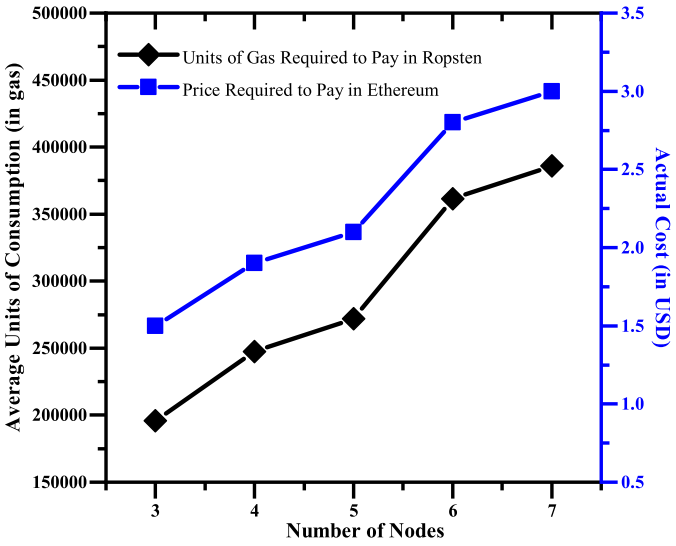

Fig. 8. Average gas consumption versus transaction cost.

of the Ropsten Ethereum Faucet[4]. Additionally, the Ropsten test-net explorer[5] offers a user-friendly online interface that exposes critical data about the mined transactions (such as Gas used, block reward, etc.). Ropsten test-net and Remix IDE are linked through the Google Chrome plug-in MetaMask[6], allowing the contract's code to be deployed and invoked.

The quantity of gas consumed illustrates the computational overhead of the proposed approach. Every step of the execution process in the ethereum ecosystem costs gas. Two forms of gas costs were detected throughout the simulations: transaction costs and execution costs. The amount of money is necessary to pay in order to put a contract code (or transaction) into the Ethereum network is the transaction cost. On the other hand, execution cost refers to the amount of money needed to carry out a specific task. Fig. 7 shows the average gas consumption and deployment time for different

[4]https://faucet.ropsten.be/
[5]https://ropsten.etherscan.io/
[6]https://metamask.io/

numbers of transactions. In contrast, Fig. 8 shows the average gas consumption and transaction cost in USD[7] for various numbers of nodes. The result obtained from the simulation result concludes that the gas price, deployment time, and USD cost increase proportionally to the varying number of transactions and nodes.

VI. Concluding Remarks & Future Scope

This article presents a batch authentication protocol that utilizes smart contracts, blockchain technology, and ECC crypto-operations. This protocol utilizes smart contracts during the registration procedure and provides a partial authentication phase to make the protocol robust. We have to do formal verification using the Scyther security tool, which indicates no attacks found in the proposed protocol within the specified bound. The informal study of the scheme proves that the proposed scheme is secure against well-known attacks. The detailed comparative analysis exhibits that the proposed scheme is superior in terms of computational and communication overheads and efficiently achieving the decentralized properties of the blockchain. In addition, this scheme also supports more features and security attributes than the existing schemes. The Remix IDE and Rosten test-net is used to simulate the Ethereum-enabled blockchain, which exhibits the associated cost and deployment time for the different number of transactions.

Negotiation of group keys among the static and dynamic entities of a network is one of the challenging tasks. In VANET, vehicles are frequently migrating from one vehicular network to other. The vehicles, with the assistance of RSU, want to create a group key for different purposes, like achieving a consensus for an event. Moreover, communication between vehicles is done through a public channel. As a result, the security and confidentiality of transmitted messages constitute a significant concern in VANET. In our continuous research, we will work on the construction of group keys among the vehicles (for both static and dynamic cases) in VANET using blockchain technology.

References

[1] S. K. Dwivedi, R. Amin, and S. Vollala, "Blockchain-based secured IPFS-enable event storage technique with authentication protocol in VANET," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 12, pp. 1913–1922, Dec. 2021.

[2] P. Bagga et al., "On the design of mutual authentication and key agreement protocol in Internet of Vehicles-enabled intelligent transportation system," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1736–1751, Feb. 2021.

[3] A. Vangala, B. Bera, S. Saha, A. K. Das, N. Kumar, and Y. H. Park, "Blockchain-enabled certificate-based authentication for vehicle accident detection and notification in intelligent transportation systems," *IEEE Sensors J.*, vol. 21, no. 14, pp. 15824–15838, Jul. 2021.

[4] D. Chattaraj, B. Bera, A. K. Das, S. Saha, P. Lorenz, and Y. Park, "Block-CLAP: Blockchain-assisted certificateless key agreement protocol for Internet of Vehicles in smart transportation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 8092–8107, Aug. 2021.

[5] S. K. Dwivedi, R. Amin, S. Vollala, and M. K. Khan, "B-HAS: Blockchain-assisted efficient handover authentication and secure communication protocol in VANETs," *IEEE Trans. Netw. Sci. Eng.*, early access, Apr. 24, 2023, doi: 10.1109/TNSE.2023.3264829.

[7]The price of ether is cited from CoinMarketCap in $2022-01-04$.

[6] S. K. Dwivedi, R. Amin, and S. Vollala, "Blockchain based secured information sharing protocol in supply chain management system with key distribution mechanism," *J. Inf. Secur. Appl.*, vol. 54, Oct. 2020, Art. no. 102554.

[7] M. Azees and P. Vijayakumar, "Cekd: Computationally efficient key distribution scheme for vehicular ad-hoc networks," *Austral. J. Basic Appl. Sci.*, vol. 10, no. 2, pp. 171–175, 2016.

[8] A. B. S. Ahamed, N. Kanagaraj, and M. Azees, "EMBA: An efficient anonymous mutual and batch authentication schemes for vanets," in *Proc. 2nd Int. Conf. Inventive Commun. Comput. Technol. (ICICCT)*, Apr. 2018, pp. 1320–1326.

[9] I. Ali, M. Gervais, E. Ahene, and F. Li, "A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in VANETs," *J. Syst. Archit.*, vol. 99, Oct. 2019, Art. no. 101636.

[10] S. F. Tzeng, S. J. Horng, T. Li, X. Wang, P. H. Huang, and M. K. Khan, "Enhancing security and privacy for identity-based batch verification scheme in VANETs," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3235–3248, Apr. 2017.

[11] J. Chen, M. S. I. Mamun, and A. Miyaji, "An efficient batch verification system and its effect in a real time VANET environment," *Secur. Commun. Netw.*, vol. 8, no. 2, pp. 298–310, Jan. 2015.

[12] Y. Liu, Z. He, S. Zhao, and L. Wang, "An efficient anonymous authentication protocol using batch operations for VANETs," *Multimedia Tools Appl.*, vol. 75, no. 24, pp. 17689–17709, Dec. 2016.

[13] S. Jiang, X. Zhu, and L. Wang, "An efficient anonymous batch authentication scheme based on HMAC for VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2193–2204, Aug. 2016.

[14] N. Gayathri, G. Thumbur, P. V. Reddy, and M. Z. U. Rahman, "Efficient pairing-free certificateless authentication scheme with batch verification for vehicular ad-hoc networks," *IEEE Access*, vol. 6, pp. 31808–31819, 2018.

[15] A. K. Sutrala, P. Bagga, A. K. Das, N. Kumar, J. J. P. C. Rodrigues, and P. Lorenz, "On the design of conditional privacy preserving batch verification-based authentication scheme for Internet of Vehicles deployment," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5535–5548, May 2020.

[16] S. Bouakkaz and F. Semchedine, "A certificateless ring signature scheme with batch verification for applications in VANET," *J. Inf. Secur. Appl.*, vol. 55, Dec. 2020, Art. no. 102669.

[17] P. Vijayakumar, M. Azees, S. A. Kozlov, and J. J. P. C. Rodrigues, "An anonymous batch authentication and key exchange protocols for 6G enabled VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1630–1638, Feb. 2022.

[18] H. Cheng, M. Shojafar, M. Alazab, R. Tafazolli, and Y. Liu, "PPVF: Privacy-preserving protocol for vehicle feedback in cloud-assisted VANET," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 9391–9403, Jul. 2022.

[19] C. Maurya and V. K. Chaurasiya, "Efficient anonymous batch authentication scheme with conditional privacy in the Internet of Vehicles (IoV) applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9670–9683, Sep. 2023.

[20] Y. Yang, D. He, H. Wang, and L. Zhou, "An efficient blockchain-based batch verification scheme for vehicular ad hoc networks," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 5, May 2022, Art. no. e3857.

[21] C. Lin, D. He, X. Huang, N. Kumar, and K.-K. R. Choo, "BCPPA: A blockchain-based conditional privacy-preserving authentication protocol for vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 12, pp. 7408–7420, Dec. 2021.

[22] P. Bagga, A. K. Sutrala, A. K. Das, and P. Vijayakumar, "Blockchain-based batch authentication protocol for Internet of Vehicles," *J. Syst. Archit.*, vol. 113, Feb. 2021, Art. no. 101877.

[23] Y. Ren, X. Li, S.-F. Sun, X. Yuan, and X. Zhang, "Privacy-preserving batch verification signature scheme based on blockchain for vehicular ad-hoc networks," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102698.

[24] A. Maria, A. S. Rajasekaran, F. Al-Turjman, C. Altrjman, and L. Mostarda, "BAIV: An efficient blockchain-based anonymous authentication and integrity preservation scheme for secure communication in VANETs," *Electronics*, vol. 11, no. 3, p. 488, Feb. 2022.

[25] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, May 2002.

[26] B. Bera, S. Saha, A. K. Das, N. Kumar, P. Lorenz, and M. Alazab, "Blockchain-envisioned secure data delivery and collection scheme for 5G-based IoT-enabled Internet of Drones environment," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9097–9111, Aug. 2020.

[27] P. Mall and R. Amin, "EuDaimon: PUF-based robust and lightweight authenticated session key establishment protocol for IoT-enabled smart society," *IEEE Syst. J.*, vol. 16, no. 2, pp. 2891–2898, Jun. 2022.

[28] J. Cui, J. Zhang, H. Zhong, and Y. Xu, "SPACF: A secure privacy-preserving authentication scheme for VANET with cuckoo filter," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10283–10295, Nov. 2017.

[29] Y. Wang, H. Zhong, Y. Xu, J. Cui, and F. Guo, "Efficient extensible conditional privacy-preserving authentication scheme supporting batch verification for VANETs," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 5460–5471, Dec. 2016.

[30] J.-x. Zhou and J.-H. Yan, "Secure and efficient identity-based batch verification signature scheme for ADS-B system," *KSII Trans. Internet Inf. Syst. (TIIS)*, vol. 13, no. 12, pp. 6243–6259, 2019.

[31] M. A. Alazzawi, H. Lu, A. A. Yassin, and K. Chen, "Efficient conditional anonymity with message integrity and authentication in a vehicular ad-hoc network," *IEEE Access*, vol. 7, pp. 71424–71435, 2019.

[32] Y. Wang, H. Zhong, Y. Xu, J. Cui, and F. Guo, "Efficient privacy-preserving authentication scheme with fine-grained error location for cloud-based VANET," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10436–10449, Oct. 2021.

[33] X. Feng, Q. Shi, Q. Xie, and L. Wang, "P2BA: A privacy-preserving protocol with batch authentication against semi-trusted RSUs in vehicular ad hoc networks," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3888–3899, 2021.

[34] H. Sikarwar and D. Das, "Towards lightweight authentication and batch verification scheme in IoV," *IEEE Trans. Depend. Secure Comput.*, vol. 19, no. 5, pp. 3244–3256, Sep. 2022.

[35] T. Limbasiya and D. Das, "Lightweight secure message broadcasting protocol for vehicle-to-vehicle communication," *IEEE Syst. J.*, vol. 14, no. 1, pp. 520–529, Mar. 2020.

Sanjeev Kumar Dwivedi (Member, IEEE) received the M.Tech. degree in distributed computing systems from the Department of Computer Science, Pondicherry University, Puducherry, in 2013, and the Ph.D. degree in computer science and engineering from the Dr. Shyama Prasad Mukherjee International Institute of Information Technology, Naya Raipur, Chhattisgarh, in 2023. He is currently an Assistant Professor with the VIT-AP School of Computer Science and Engineering (SCOPE), VIT-AP University, Amaravati, AP, India. He has a total academic experience of 4.5 years. He has published a few research papers in journals and conference proceedings of international repute. Some of his research papers are published in reputed journals, such as IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE/CAA JOURNAL OF AUTOMATICA SINICA, and *Journal of Information Security and Applications* (Elsevier). His current research interests include authentication protocols, cryptography, and blockchain technology and its applications.

Ruhul Amin (Senior Member, IEEE) received the B.Tech. and M.Tech. degrees in computer science and engineering from the Maulana Abul Kalam Azad University of Technology, West Bengal, India, in 2009 and 2013, respectively, and the Ph.D. degree in computer science and engineering from the Indian Institute of Technology (ISM) Dhanbad, Jharkhand, India, in 2017. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Dr. Shyama Prasad Mukherjee International Institute of Information Technology, Naya Raipur, India. He has authored many technical research papers published in leading international conferences and peer-reviewed international journals, such as IEEE, Elsevier, Springer, and John Wiley. Some of his research findings are published in top-cited journals, such as IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTION ON CONSUMER ELECTRONICS, IEEE INTERNET OF THINGS, IEEE SYSTEM JOURNAL, *Ad-Hoc Networks*, *Computer Networks*, *Future Generation Computer Science*, *Journal of Network and Computer Application* (Elsevier). His research interests include cryptography and network security, authentication protocol, and blockchain technology. He is also serving as an Associate Editor of *Security and Privacy* (John Wiley). He has been included in the subject-wise ranking of the top 2% of scientists from India (all fields) in the area of networking and telecommunications.

Satyanarayana Vollala (Member, IEEE) received the M.Tech. degree in computer science and engineering from Jawaharlal Nehru Technological University Hyderabad, Andhra Pradesh, and the Ph.D. degree from the National Institute of Technology, Tiruchirappalli. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Dr. Shyama Prasad Mukherjee International Institute of Information Technology, Naya Raipur. His research interests include security protocols, number systems, hardware implementations public-key cryptography, and modular exponential algorithms.

Ashok Kumar Das (Senior Member, IEEE) received the M.Sc. degree in mathematics, the M.Tech. degree in computer science and data processing, and the Ph.D. degree in computer science and engineering from IIT Kharagpur, India. He has been working as a Visiting Faculty with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA, USA, since 2022. He is currently a Full Professor with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad, India. His current research interests include cryptography, network security, security in vehicular ad hoc networks, smart grids, smart homes, the Internet of Things (IoT), the Internet of Drones, the Internet of Vehicles, cyber-physical systems (CPS), cloud computing, intrusion detection, blockchain, and AI/ML security. He has authored over 375 papers in international journals and conferences in the above areas, including over 315 reputed journal articles. He has served as a program committee member for many international conferences. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He also served as one of the Technical Program Committee Chairs of the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, in June 2019, the International Conference on Applied Soft Computing and Communication Networks (ACN'20), Chennai, India, in October 2020, and the second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, in October 2020. He is/was on the editorial board of IEEE SYSTEMS JOURNAL, *Journal of Network and Computer Applications* (Elsevier), *Computer Communications* (Elsevier), *Journal of Cloud Computing* (Springer), *Cyber Security and Applications* (Elsevier), *IET Communications*, *KSII Transactions on Internet and Information Systems*, and *International Journal of Internet Technology and Secured Transactions* (Inderscience).