Secure and Scalable Communication in IoV using Blockchain

Sandeep Srivastava, Deepshikha Agarwal, *Member, IEEE*, Brijesh Kumar Chaurasia, *Senior Member, IEEE*

***Abstract*—Recently with the prime importance of intelligent technology in new generation Internet of Vehicles (IoV) become an attractive field for researchers and engineers both. With major advancements authentication of devices and privacy of data transfer in IoV also becomes one of the important issue from the prospective of security, various kind of attacks, integrity, confidentiality and identity management are major issues that must be addressed under the authentication properties. Securing IoV networks requires secure key management. The proliferation of roadside devices makes it impossible for wireless connections to be both secure and efficient in vehicles with real-time information transmission limits. To address issues, We propose a blockchain-based key management system with improved efficiency and identity security, achieved by employing multi-tier BLS signature based key generation, enhancement and verification. The proposed three-layer identity-based key distribution mechanism to ward off attackers, and secures system using a pre-generated aggregate signature technique. Based on the results of the security analysis and performance simulation, our method is more secure, scalable and does not exceed the communication latency limit.**
Keywords-IoV, Authentication, Signature, Function, Data integrity, Verification

I. Introduction

The large amount of remote data which are processed continuously is difficult to store and in this a virtual base cloud helps to provide secure storage resources to IoV network [1]. IoV generates large amount of data that is difficult to store and mange. Cloud computing provide a platform which is an easy travel method from big data [2]. Cloud is the location where big data is stored accessed, processed and modelled. Cloud is the utility of IoV that available on net. So it has come up with many security issues also like privacy, eavesdropping, authentication data theft, confidentiality and integrity. Today, the demand of secure data transfer is a big problem that we take as a problem statement and propose a solution for trusted IoV environment where data authentication and confidentiality can be maintained. For inducing trust in the computing environment, there is a need of a system that performs authentication, verification and encryption that helps maintain the data confidentiality. There are various kind of attacks that hinders the safe data transfer between two devices. Some of them are –

- Tempering- In this attack the attacker modify the data saved on network or in a local device.is attack the attacker modify the data saved on network or in a local device.
- Eavesdropping - In this attack the attacker gain the information about routing data and enters in data path and access the data.
- Repudiation- in this situation the sender may deny about the data validity which is sent by him.
- Man in the middle attack- In this type of attack the attacker interfere in the communication channel or data path and try to alter the information.
- Replay attack- When the original valid data is delayed by the attacker with the malicious purpose.
- Identity spoofing- When the attacker replace his identity with the original sender then this type of attack called as identity spoofing.
- Differential analysis threat- Difference between the code of new released version and old published version.
- Viruses and worms- The program codes that attacks on the original files by attaching itself with the valid codes.

The advent of internet of vehicles makes life easier and smart but with this enhancement of technology, the security of private data is become one of the biggest concern [3]. Due to need of unlimited storage and computational resources IoV has increases its cloud resources in various ways. IoV starts storing data by using storage resource provided by cloud service providers. Cloud computing greatly help the IoV by leveraging the overhead of storage, computation and efficiency [4]. Cloud storage providers used centralized storage system that is the combination of hardware and software events and if any malicious attack will damage any of the middle event the whole IoV system will seriously threatened. Therefore to check integrity of data is important for its availability on time. In order to resolve the above problem and increase the safety of IoV a short signature based model i.e. BLS signature is presented in this paper for data integrity and verification. BLS signature scheme uses bilinear pairing for mapping the signature. A special hash function is required in BLS signature scheme which is a specific function called as Map to point which is used in many conventional cryptography scheme. The signature scheme can be summarized as below-

- By introducing trusted third party this scheme can auditing the data of public so that additional overhead on users can be easily reduced about holding the data.
- This scheme uses masking technique to maintain the privacy of data.
- By computing the hash function the integrity of data is improved.
- This scheme can hold out against the chosen adaptive message attack and provide high security in oracle model.

Corresponding author: Sandeep Srivastava.

S. Srivastava and D. Agarwal are with the Department of Computer Science, Indian Institute of Information Technology Lucknow, India and and B. Chaurasia is with Department of Computer Science, PSIT Kanpur. (e-mail: sandeepsriv20@gmail.com, deepshikha.agarwal@iiitl.ac.in,bkc.iiita@gmail.com).

Paper is structured with the explanation of related work in this field and description of the standard definition and syntax of BLS signature. At the end concludes with the results that shows of achieving properties of authentication and data privacy.

A. Related Work

Today IoV and cloud is an intensively important for each other and this integration of cloud and IoV is become a widely used network. Neagu e.t al. present an integrated IoV cloud architecture in their health monitoring model [5]. In this paper many benefits are counted of cloud-IoV as high data storage, efficient performance, recovery of data and efficiency of sensors data [6]. Liu et. al. present an opinion where they say that most IoV applications using cloud for data storage which is also not much safe [7]. So they propose a block chain based data security architecture which was a decentralized framework which verifies data integrity. Shah et. al. propose authentication scheme for the verification of data integrity that is code based to verify the integrity of real time data. Venkatesh et. al. propose a PDP mechanism with RSA signature to verify the integrity of remote data [8]. Liu et al propose a varication mechanism for batch integrity [7]. This scheme supports full dynamic data, performs authentication of each block and verifies the data updates. Digital signature scheme was proposed by Diffie and Hellman [9] in which a scheme was proposed where secret signing key was generated for message signing and a public verification key for receiver to verify and recover original message. Goldwasser et.al. [10] gave the another definition of signature scheme in which they define a scheme which is unforgeable means any adversary in probabilistic polynomial time with negligible probability. Then Boyle et.al. [10] proposed the functional signature scheme in which a master key is used to sign any message. Johnson et.al. [11] proposed the homomorphic signature which is used to authenticate the data and code. In this signature scheme any sender can construct a signature for function without having signing key. Catalina. et.al. [12] extended this work and present the model of homomorphic signature for polynomial functions .

Functional encryption was first presented by Sahai and waters [13] and simulation based concept was proposed by Boneh et.al. [14]. They present the scheme for cloud service and restrict the number of client for security and indistinguishable obfuscation. Fan and tang et.al. [15] proposed a model in which verification key is not one but for function (f) several splited key sk_i^f.

Security on cloud service for many users and authority inspired by public key encryption scheme presented in Chandra et.al.[16]. Where they gave the general encryption and general functional policy for multi authorities. They show the functional encryption for arbitrary polynomial size circuits, denotes as $[F\{U_{id}\}_{id\in S}, m]$.

In Okamoto et.al. explained the signature scheme for multi authority and a MA-ABS model was presented in which multiple authorities share a secret key to each authority. But in this model a centralised certified trust is needed. And if central trust will crashed then the whole signature model will be failed. Again Okamoto et.al. [17] presented this model with non-central authority which is more general and here no global co-ordination is required. They also proposed DMA-FS model which is more secure under the standard assumption i.e. DLIN assumptions rather than MA-ABE model.

Liang and Mitro kotsa et.al. [6] gave the more generalized definition of distributed many authority functional signature. Data et.al. [18] presented the model which is having qualities of functional signature and functional encryption both .

In summary most of available data integrity verification techniques are somehow based on RSA and in RSA the computational overhead become so heavy to process the data. And in IoV signature efficiency is remain improvable and privacy needs always strengthening. Recently short signature scheme is identified which saves most of the storage, reduces the computational overhead, improves the efficiency of signature and provide data security. BLS signature known as Boneh Lynn Shacham is a cryptographic technique which uses bilinear mapping to map signature with message.

II. Preliminaries

Secure communication with authenticated devices is required for the IoV network. Various tools for developing cryptography algorithms suggest bilinear pairing as a suitable method.

Definition 2.1: Let G_1 be a cyclic additive group and G_2 be a cyclic multiplicative group of the prime order q. Let P be an arbitrary generator of group G_1. aP denotes P added to itself a times. A bilinear map is a map $\hat{e} : G_1 \times G_1 \to G_2$ with the following properties.

- **Bilinearity:** $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in G_1$ and $a, b \in Z_q^*$. Here $Z_q^* = \{\rho | 1 \leq \rho \leq q-1\}$
- **Non-degeneracy:** $\hat{e}(P, P) \neq 1$
- **Computable:** There exist an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in G_1$.

We also assume G_1 is a Gap Differ-Hellman(GDH) group. As such, Computational Diffie-Hellman Problem (CDHP) is hard in G_1. That is, consider a cyclic group G of order q, give (P, aP, bP) for a randomly chosen generator P and random $a, b \in 0, ..., q-1$, compute the value of abP is hard.

III. Problem Statement and System Model

A. Problem Statement

The objective of our proposed scheme is to propose and design a secure and proficient key management protocol. We, first and foremost, carry out decentralized and anonymous authentication without a confided in outsider. Besides, it should meet the security prerequisites, such as validation and integrity, non-contradiction and tamper-evident, traceability, and protection from different attacks. At long last, to ensure the reasonable application impact, the above of the authentication protocol ought to be lightweight, and the framework ought to accomplish high scalability.

Considering dishonest vehicles and lack of quality of network communication, We summarise the likely threats of malicious

TABLE I
VARIOUS TECHNIQUES FOR SECURITY AND AUTHENTICATION

Paper	Year Published	Focus areas(s) of the paper	Method(s) used	Issues
Lin et al.[19]	2007	Privacy, security and authentication	Group signatures	• RSUs have to save all third party certificates from the neighboring vehicles which have associated overheads. • If the group leader is malicious and reveals the key, the entire group is compromised. • Group-signatures generally have high signature verification and revocation costs.
Lin et al.[18]	2008	Privacy, security and authentication	Bilinear pairing based cryptography	• Receives RSU compromise attack. • Generation of pseudonym key by RSU increases latency. • Requires frequent interaction with RSU. • No revocation mechanism specified.
Huang et al. [20]	2011	Authentication, security, and privacy	Bilinear pairing ID-based encryption	• Uses ECC which has associated overhead. • Can handle only replay and modification attacks.
Zhang et al. [21]	2013	Privacy, security and authentication	Self- certified public keys and bilinear pairing	• Preserves conditional privacy under random oracle track.
Azees et al.[22]	2017	Privacy, security and authentication	Bilinear pairing based cryptography	• Verification of certificates and signatures is faster, however signing and authentication is delayed.

nodes and uncertainity factors in the network. The harmful nodes would attack the network by Denial-of-Service, DDoS, Man-in-the-middle, Spoofing, Eavesdropping and Phishing type of attacks. This would result in a compromised network with malicious vehicles communicating within the network which degrades network performance and increases network latency.

B. System Model

The scheme is composed of four algorithms called **Key generation, Key enhancement, Message Sign,** and **Verification** as shown in the figure 1. Let γ denote the security parameter. Let $H(\cdot)$ denote the hash function on the elliptic curve. Let $\{v_1, v_2, v_3, \ldots, v_n\}$ denote the vehicles to join signing. The identity of v_i is denoted as ID_{v_i} and the message to be signed by v_i is m_i.

- **Key Generation (KeyGen):** takes 1 γ as input and generates master key pair (pk, pub). Where *pk* is the private key, *pub* is the public key.
- **Key Enhancement(KeyEnh):** takes *pk* and an identity ID_{v_i} as input, and generates identity-related key pair (pk_{v_i}, pub_{v_i}).
- **Signing the message (SignM):** takes pk_{v_i} and m_i as input, and generates identity-based signature σ_i $(i \in 1, 2, 3, \ldots, n)$. Let's consider B = $\sum_{i=1}^{n} B_{v_i}$, where $B_{v_i} = b_{v_i} P$, b_{v_i} is randomly chosen from Z_q^* . Let $h_{v_i} = H(m_i, B)$, $L_{v_i} = b_{v_i pub} + h_{v_i} pk_{v_i}$. The signature of m_i signed by vehicle v_i is σ_i = (B_{v_i}, L_{v_i}). Let L= $\sum_{i=1}^{n} L_{v_i}$, the aggregate signature of $m_1, m_2, m_3, \ldots, m_n$ is $\sigma = (B, L)$.
- **Verification (Verify):** takes $m_1, m_2, m_3, \ldots, m_n$ and σ as input, the signature is confirmed if $\hat{e}(P, L)$ = $\hat{e}(pub, N)$, $N = B + \sum_{i=1}^{n} h_{v_i} pub_{v_i}$.

IV. PROPOSED BLOCKCHAIN-ENABLED SECURE BLS SIGNATURE BASED PRIVACY PRESERVATION

Let $ct_1, ct_2, ct_3, \ldots, ct_n$ denote all the CTAs connected with NTA. Let $v_1, v_2, v_3, \ldots, v_n$ denote all the vehicles connected with $ct_i = (i = 1, 2, \ldots, n)$. There are three phases in the proposed scheme named System Initialization, Group join, and verification.

A. System initialization

In this phase NTA, CTA, RSUs, and vehicles run the following step to initialize the system parameters. Note that Blockchain data should be backed upto NTA and eliminated by CTA and RSUs after closing the toll to restore storage burden. To enhance key distribution confidentiality initialization needs to be processed in batch (Algorithm 1).

1) Let G_1 and G_2 indicate two groups of the same prime order q. Let G_1 be an additive group and G_2 as a multiplicative group. Let P be an arbitrary generator of G_1. NTA chooses $\hat{e}$ as a bilinear map that satisfies $\hat{e} : G_1 \times G_1 \rightarrow G_2$. Then NTA choose $H : \{0, 1\}^* \times$

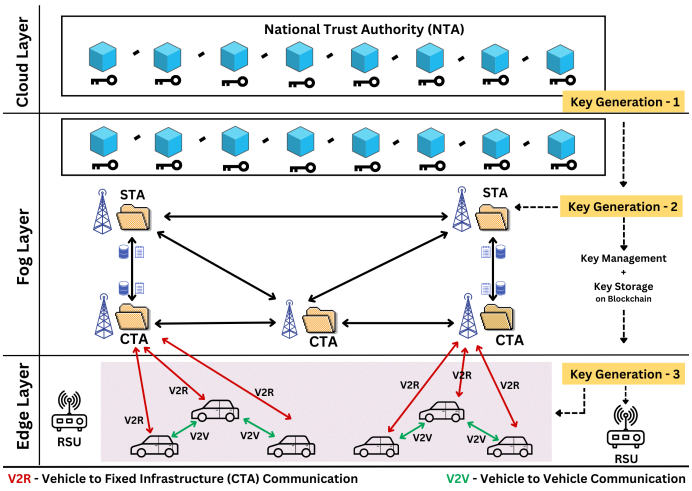

Fig. 1. Proposed three tier key management framework

$G_1{}^* \rightarrow Z_q^*$ and $h : \{0,1\}^* \rightarrow G_1{}^*$ as two collision-resistant hash functions. Finally, NTA publish the system parameter. $Param = \{G_1, G_2, \hat{e}, q, P, H, h\}$ to CTA's and RSU's.

2) NTA performs key function to generate master key pair (pk_{NTA}, pub_{NTA}), where $pk_{NTA} \in Z_q^*$ as a private key. And computes public key $pub_{NTA} = pk_{NTA} * P$, where pk_{NTA} and pub_{NTA} belongs to Z_q^*.
3) The ith CTA, ct_i performs key enhancement by sending key enhancement request with their id ct_{ID_i} to NTA.
4) NTA takes input pk_{NTA} and ct_{ID_i} to generate identity based public private key pairs (pk_{ct_i}, pub_{ct_i}) and returns into CTA by calculating $pub_{ct_i} = h(ct_{ID_i})$ and $pk_{ct_i} = pk_{NTA} * pub_{ct_i}$.
5) Let jth static entity RSU_i^j performs key enhancement by sending a key enhancement request with their identity RSU_{ID}^j to the nearest CTA.
6) CTA takes private key ct_i and RSU_i^j to generate identity based public private key pairs $(pkr_i^j, pubr_i^j)$. To the static entity by calculating $pk_{r_i^j} = h(RSU_i^j)$ and $pub_{r_i^j} = pk_{ct_i} pub_{r_i^j}$.
7) The master key pub_{NTA} system parameter $param$, CTA_{ID_i} and RSU_i^j are recorded in initial block. $ID_{ct_i} = H * CTA_{ID_i}$ and $ID_{RSU_i^j}$ for security. CTAs can verify master key by checking if $\hat{e}(P_{NTA}, pk_{ct_i}) = \hat{e}(pub_{ct_i}, pub_{NTA})$. Similarly RSUs can verify the master keys by checking if $\hat{e}(P, p_{r_i^j}) = \hat{e}(pub_{ct_i} pub_{r_i^j}, pub_{NTA})$.

B. Communication Phase

In this phase CTA allocates a dynamic key to high speed vehicle which is entering current section of toll. Vehicle interactions are confirmed and computed trust values are written to blockchain by a certain group of static entities (Algorithm 2).

1) At the toll the high speed vehicle request NTA for key generation using its identity which is vehicle v_i.

Algorithm 1: System Inititalization

Input: $G_1, G_2, \hat{e}, q, P$
Output: $pk_{NTA}, pub_{NTA}, pk_{ct_i}, pub_{ct_i}$, Key Status

1: NTA generates the master public and private key pair (pk_{NTA}, pub_{NTA}).
2: Private key $(pk_{NTA} = Random(Z_q^*))$
3: Public Key $(pub_{NTA} = pk_{NTA} * P)$
4: Record the key pair generated by NTA to the blockchain.
5: **for** $i = 1$ **to** n **do**
6: Store parameter $G_1, G_2, \hat{e}, q, P$ in ct_i
7: CTA requests NTA for key enhancement with their id ct_{ID_i}
8: NTA takes input and calculates $pub_{ct_i} = h(ct_{ID_i})$ and $pk_{ct_i} = pk_{NTA} * pub_{ct_i}$.
9: NTA returns the pair (pub_{ct_i}, pk_{ct_i}) and records it to the blockchain.
10: **end for**
11: **for** $i = 1$ **to** m **do**
12: Store parameter $G_1, G_2, \hat{e}, q, P$ in RSU_i .
13: RSU_i^j requests the nearest CTA for key enhancement with their id RSU_{ID}^j.
14: CTA takes private key ct_i and RSU_i^j and calculates $pk_{r_i^j} = h(RSU_i^j)$ and $pub_{r_i^j} = pk_{ct_i} pub_{r_i^j}$.
15: CTA returns the key pair $(pkr_i^j, pubr_i^j)$ and records it to the blockchain.
16: CTA then verifies the master keys and returns Key Status.
17: **if** $\hat{e}(P, p_{r_i^j}) = \hat{e}(pub_{ct_i} pub_{r_i^j}, pub_{NTA})$ **then**
18: Key Status = 1
19: **else**
20: Key Status = 0
21: **end if**
22: **end for**

2) NTA uses pk_{NTA} and v_i as inputs and returns identity base key pairs (pk_{v_i}, pub_{v_i}) to the Vehicle by calculating $pub_{v_i} = h(v_i)$ and $pk_{v_i} = pk_{NTA} pub_{v_i}$. Then NTA also broadcast ID_{v_i} and pub_{v_i} to blockchain network with signature $\sigma_{v_i} = (B_{v_i}, L_{v_i})$. Now here $ID_{v_i} = H(v_i)$ for security. NTA selects arbitrary number $b_{v_i} \in Z_q^*$ and calculates $B_{v_i} = b_{v_i} P$. Let $h_{v_i} = H((ID_{v_i} || pub_{v_i}), B_{v_i})$, L_{v_i} can be calculated as $L_{v_i} = b_i pub_{NTA} + h_{v_i} pk_{NTA}$.
3) σ_{v_i} and pub_{NTA} can verify the correctness of ID_{v_i} and pub_{v_i} by confirming if $\hat{e}(P, L_{v_i}) = \hat{e}(pub_{NTA}, N_{v_i})$, where $N_{v_i} = B_{v_i} + h_{v_i} pub_{NTA}$.
4) After entering the toll vehicle broadcast its state with signature $\sigma_{state} = (B_{state}, L_{state})$ to RSU, CTA, and other vehicles. Here $state = \{ID_{v_i} || Directions || Speed || Location || Time || pub_{v_i}\}$, where Direction is the running direction, Speed is the current speed, Location is the current location, Time is the time stamp. To sign the state vehicle uses random number $b_{state} \in Z_q^*$ and computes $B_{state} = b_{state} P$. Let $h_{state} = H(ID_{v_i} || pub_{v_i}), B_{state})$ and L_{state} can be calculated as $L_{state} = b_{state} pub_{v_i} + h_{state} pk_{v_i}$.
5) CTA and RSUs first confirmed the validity of ID_{v_i}

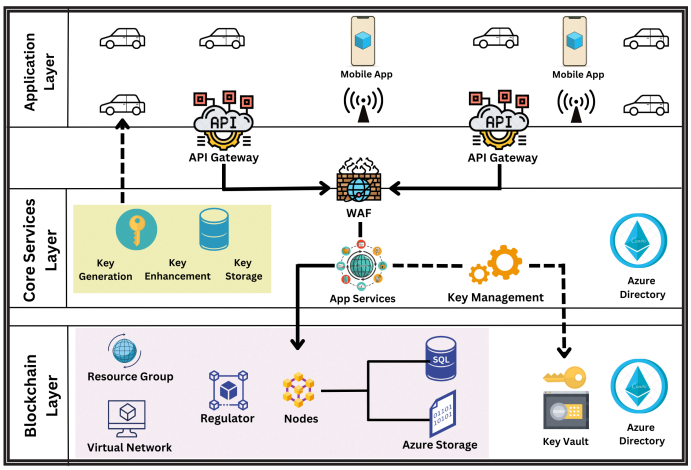

Fig. 2. Reference deployment architecture of proposed model

in the state by checking if there is already ID'_{v_i} on blockchain, i.e. $ID_{v_i} = ID'_{v_i}$. If ID'_{v_i} exists then RSU confirms vehicle as authenticated vehicle.

6) State, σ_{state} and pub_{v_i} as input. Let $h_{state} = H((ID_{v_i}||pub_{v_i}), B_{state})$. Now, CTA and static entity verifies the correctness of state by confirming if $\hat{e}(P, L_{state}) = \hat{e}(pub_{v_i}, N_{state})$, where $N_{state} = B_{state} + h_{state}pub_{v_i}$. Each static entity signs ID_{v_i} and state publishes the signature $\sigma_{v_i}(B_{v_i}, L_{v_i})$. CTA performs aggregate of $\sigma = \sigma_{v_1}, \sigma_{v_2}, \sigma_{v_3}, \ldots, \sigma_{v_n}$.
7) Verification: CTA takes σ, $(ID_{v_i}, state)$, and ID_{v_i} as input to verify the correctness of ID_{v_i}, state. If the aggregated signature is confirmed then trust value will be computed and recorded on blockchain.

V. Empirical Evaluation

A. Experimental Setup:

The Tendermint framework is used to develop permissioned blockchains in order to evaluate the efficacy of our proposed multi-tiered key management approach using the Blockchain Network. The network was evaluated using 20 docker nodes that use Optimised PBFT Consensus at the fog and cloud layers. The IoV simulation was performed using NS3 ver. 3.26, which is installed on a system with a 3.0-GHz Ryzen 5 processor, 16 GB 3200-MHz DDR4 RAM, and Ubuntu-14.04 LTS (64 bit). The road length used for the simulations is 1000 m. In this simulation, the transmission range is set at 50 m. The execution duration of the authentication operations was calculated utilizing the public pairing-based cryptography (PBC) library [23] and [24].

B. Security discussion

Theorem 1: The proposed model can achieve credible identities and state of storage under single point of failure.

Proof: In the proposed model, Blockchain is used to store the vehicle states and the information related to the identities. Using blockchain, all participants securely store states and identities, rather than storing it on a single server. At least

Algorithm 2: Communication Phase

Input: $v_i, pk_{NTA}, P, state, \sigma_{state}, pub_{v_i}$
Output: $pk_{v_i}, pub_{v_i}, ID_{v_i}, ID_{v_i}$ and pub_{v_i} correctness, Vehicle Authentication, State Correctness

High Speed Vehicle requests NTA for key generation using it's identity v_i.
for $i = 1$ **to** n **do**
  NTA uses pk_{NTA} and v_i as inputs and calculates $pub_{v_i} = h(v_i)$ and $pk_{v_i} = pk_{NTA}pub_{v_i}$.
  NTA returns identity base key pairs (pk_{v_i}, pub_{v_i}) to the Vehicle.
  $ID_{v_i} = H(v_i)$ for security.
  NTA selects arbitrary number $b_{v_i} \in Z_q^*$ to calculate $B_{v_i} = b_{v_i}P$.
  NTA calculates $h_{v_i} = H((ID_{v_i}||pub_{v_i}), B_{v_i})$, so that L_{v_i} can be calculated as $L_{v_i} = b_{v_i}pub_{NTA} + h_{v_i}pk_{NTA}$.
  NTA broadcasts ID_{v_i} and pub_{v_i} to blockchain network with signature $\sigma_{v_i} = (B_{v_i}, L_{v_i})$.
end for
for $i = 1$ **to** n **do**
  σ_{v_i} and pub_{NTA} can verify the correctness of ID_{v_i} and pub_{v_i} by calculating $N_{v_i} = B_{v_i} + h_{v_i}pub_{NTA}$.
  if $\hat{e}(P, L_{v_i}) = \hat{e}(pub_{NTA}, N_{v_i})$ **then**
    ID_{v_i} and pub_{v_i} correctness = 1
  else
    ID_{v_i} and pub_{v_i} correctness = 0
  end if
  $B_{state} = b_{state}P$. {Where $b_{state} \in Z_q^*$ to sign vehicle state}
  Compute $h_{state} = H(ID_{v_i}||pub_{v_i}), B_{state})$.
  $L_{state} = b_{state}pub_{v_i} + h_{state}pk_{v_i}$.
  Broadcast $\sigma_{state} = (B_{state}, L_{state})$ to other nodes.
  if $ID_{v_i} = ID'_{v_i}$ **then**
    Vehicle Authentication = 1
  else
    Vehicle Authentication = 0
  end if
end for
CTA and RSU verify state correctness by computing $N_{state} = B_{state} + h_{state}pub_{v_i}$.
for $i = 1$ **to** n **do**
  if $\hat{e}(P, L_{state}) = \hat{e}(pub_{v_i}, N_{state})$ **then**
    State Correctness = 1
  else
    State Correctness = 0
  end if
  Each static entity signs ID_{v_i} and publishes the signature $\sigma_{v_i}(B_{v_i}, L_{v_i})$.
end for
CTA performs aggregate of $\sigma = \sigma_{v_1}, \sigma_{v_2}, \sigma_{v_3}, \ldots, \sigma_{v_n}$.
for $i = 1$ **to** n **do**
  Verify = $f(\sigma, (ID_{v_i}, state))$
  if σ is confirmed **then**
    Compute and record trust.
  end if
end for

$N/2+1$ of the available nodes in subnet are required to implement verification and assume control of blockchain data storage. Accordingly, the proposed approach can offer $N/2-1$ fault tolerance for the storage of information in each subnet. Undoubtedly, the dependability of blockchain data in this scenario is not impacted by the failure of a small number of devices. As per the reports, the communication system availability is more than 99.44%, which signifies that our model's function is available most of the time while using. Moreover, since the blockchain is append-only, the identification information is preserved permanently. Hence, this signifies that our model is capable of dependable state and identity storage even in the presence of a single point of failure.

Theorem 2: The proposed model can achieve a credible and lightweight verification of identity under high mobility.

Proof: In order to achieve credible and lightweight verification of identity, the authorized blockchain and ID-based composite signature techniques are utilized. The proposed model employs a PoA-like consensus method rather than a mining technique that requires a lot of processing power to lower the storage need and time commitment of rode side devices. Hence in a situation with high mobility, the lightweight consensus may be used to swiftly check identification information and vehicle states. Besides, an ID-based composite signature technique can help in decreasing the storage burden. Each transaction is traditionally recorded by a block generator, and blockchain nodes are needed to validate each transaction individually. By employing composite signatures, participants can collect multiple signatures in one transaction. Participants may check the validity of all messages by using the signature $\sigma = (R, L)$ in conjunction with the messages $m_1, m_2, m_3, \ldots, m_n$. The ID-based composite signature technique's accuracy may be shown in the following ways:

$$\hat{e}(P, L)$$

$$= \hat{e}(P, \sum_{i=1}^{n} L_{v_i}) \tag{1}$$

$$= \sum_{i=1}^{n} \hat{e}(pkP, b_{v_i}P + h_{v_i} pub_{v_i}) \tag{2}$$

$$= \hat{e}(pub, \sum_{i=1}^{n} (b_{v_i}P + h_{v_i} pub_{v_i})) \tag{3}$$

$$= \hat{e}(pub, \sum_{i=1}^{n} (B_{v_i} + h_{v_i} pub_{v_i})) \tag{4}$$

Hence, this signifies that our model can achieve a credible and lightweight verification of identity under high mobility.

VI. COMPARISONS AND PERFORMANCE ANALYSIS

We start with thorough comparison of our proposed algorithm with four other related validation schemes. Following that, broad trials are performed to exhibit the outcomes of our proposed algorithm.

TABLE II
COMPARISON OF AUTHENTICATION FEATURES

Algorithm	Decentralization	Anonymity	Traceability	Lightweight
[25]	×	✓	×	×
[26]	×	×	✓	×
[27]	×	✓	✓	×
[28]	✓	✓	✓	✓
Proposed	✓	✓	✓	✓

A. Comparisons of Our Proposed Algorithm

We begin by comparing our proposed algorithm with the other References with the help of Table II. We mostly talk about the plans from the parts of decentralization, anonymity, traceability and lightweight. The validation plans in works [25], [26] and [27] all require the third-party KGC to create partial private keys for the nodes as per their identity data for confirmation. Therefore, they are centralized, with a weak link, key administration, and identity stockpiling issues. To conceal the genuine identity of members, Refs. [25], [27] and [28] all execute mysterious interaction of members. In any case, in Ref. [25], it is hard to follow and rebuff malignant way of behaving. Then again, every one of these four examination plans execute identity confirmation through bilinear pairing. Taking into account the high computational intricacy of bilinear pairing, these plans are computationally costly. Additionally, Refs. [26] and [27] use blockchains to record public key. Our proposal in light of blockchain and gatherer, doesn't need a trusted third party to create keys for nodes. Nodes create their own public-private key pair and add (pub_i, pk_i) to the gatherer, which is kept up with by the blockchain advisory group hub. Therefore, decentralized hub enrollment and authentication are accomplished. Our proposed algorithm understands the unknown preparation of participating nodes and uncovers their actual characters in case of malignant way of behaving. Furthermore, our proposed signature calculation doesn't include time consuming bilinear pairing and exponential operation, and it supports clump confirmation to accomplish lightweight authentication. While querying the public key (pub_i, pk_i) on the DAG blockchain.

B. Comparison of Computational Costs

In the proposed methodology that the Exponential operations (T_e) takes 0.141 ms, One-Way Hashing operation (T_h) takes 0.058 ms, Multiplication operation (T_m) takes 0.049 ms and Bilinear Pairing operation (T_p) takes 4.28 ms which will be used to compare system performance against other schemes.

TABLE III
COMPARISON OF COMPUTATIONAL TIME CONSUMPTION

Scheme	Signature Time (ms) ≈	Verification Time of a message (ms) ≈
[26]	4.808	9.058
[27]	2.69	13.13
[28]	0.237	0.42
[29]	2.692	21.04
[30]	0.285	5.045
Proposed	0.165	0.263

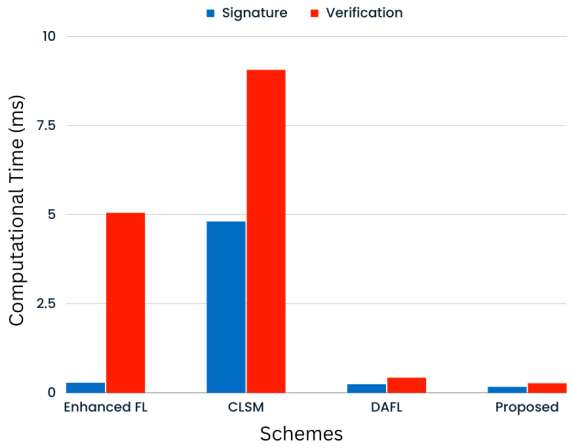

Fig. 3. Comparison of computation cost for signing and verifying a message

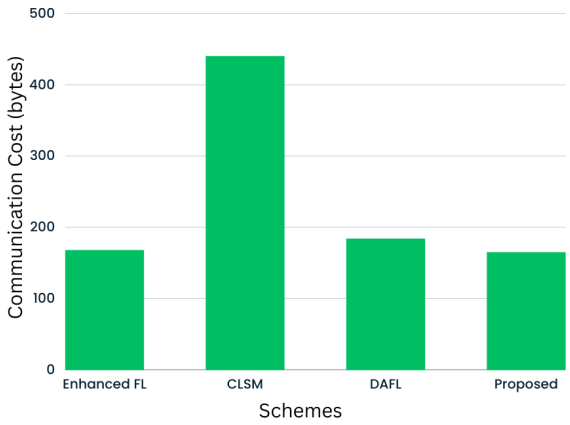

Fig. 5. Comparison of communication cost

TABLE IV
COMPARISON OF EXECUTION TIME FOR BATCH MESSAGE VERIFICATION

Scheme	Verification Time of n messages (ms) $\approx$
[26]	$9.058n$
[27]	$0.0022n + 13.128$
[28]	$0.071(2n + 1) + 0.166n$
[29]	$1.3472n + 19.692$
[30]	$5.045n$
Proposed	$2n(0.107) + 0.049$

This part exhibits the computation of proposed calculation and compares it with Refs. [28] and [26] under a similar experimental setup. Table III records different cryptographic computation operations and their execution times.

We can see the computational costs at various stages in the Table IV. While computing a message's signature, Ref. [26] requires a bilinear pairing T_p, an exponential T_e operation, a multiplication T_m operation on G and a hash T_h operations, while our scheme just require a multiplication T_m operation and two hash T_h operations. Ref. [30] requires one exponential T_e operation and one hash T_h operation worth time for signature and two multiplication T_m operation, three hash T_h operations, one bilinear pairing T_p operation and one exponential T_e operation for verification of a message. To check a single signature, Ref. [26] includes two bilinear pairing T_p operations, one multiplicative T_m operation, and one hash T_h operation.

To look at the computational cost of Refs. [26], [28] also, our scheme all the more plainly, we describe the computational cost in terms of computing a signature and carrying out an identity authentication, as shown in Fig. 3. It tends to be seen that while working out a signature, our proposed algorithm has the lowest cost, and the computational cost of Ref. [26] is a lot higher than that of [28] and ours. While checking a signature, Ref. [26] has the largest cost because there are two T_p operations, and proposed algorithm is even faster than the smaller Ref. [28] algorithm. From Fig. [graph], proposed algorithm has even lower computations in terms of signature and single check than the Ref. [28]. At the point when 100 messages are confirmed in bunch, our scheme just requires 21.45 ms of computation cost, while Refs. [28] requires 31 ms, Ref. [26] and [30] require 905.8 ms and 504.5 ms respectively.

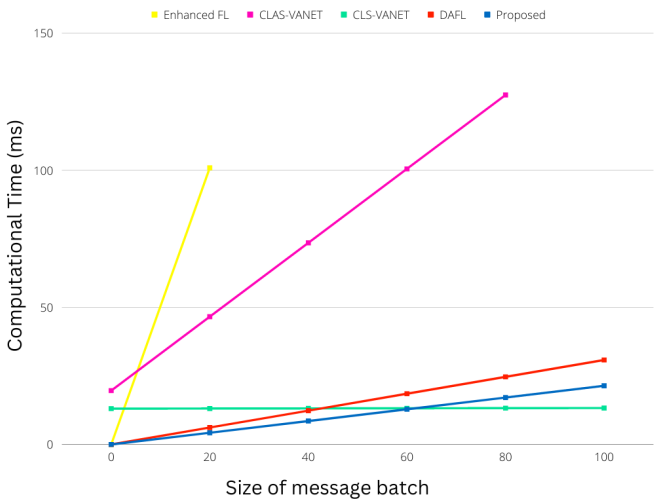

Fig. 4. Execution time for batch message verifications

C. Comparison of Communication Costs

In this section, we compare the communication costs of our proposed scheme for sending the authentication message in contrast of the other schemes. To begin with, we compared the communication cost of Ref. [28] with that of Refs. [30] and [26] under the same settings. Consider that the additive group G and the multiplicative group GT produce 40 and 128 bytes of output, respectively, and for uniformity, we assumed that worker's identity, hash function's output, a random nonce, the message, and the timestamp are 20 bytes, 20 bytes, 20 bytes, 20 bytes, and 4 bytes, respectively. In Ref. [30], the communication overhead of Ref. [30] is 168 bytes. In Ref. [26], the communication cost is 440 bytes. In Ref. [28] scheme and our scheme, it takes 184 bytes for a worker to send a tuple to a blockchain committee node. Figure 5, shows that

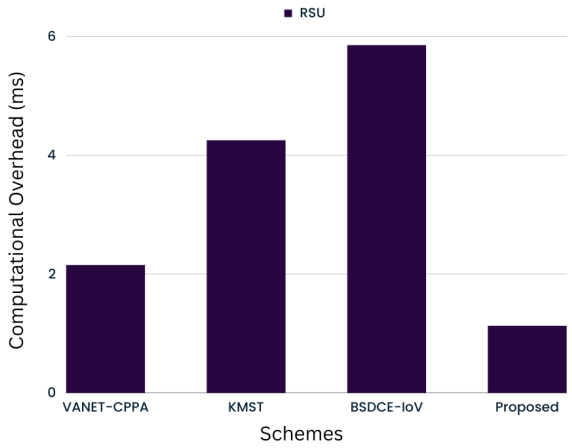

Fig. 6. Comparison of computational overhead

our communication overhead during message authentication is significantly lower when compared to Ref. [28] and Ref. [26]. Furthermore, our scheme has even lower communication overhead, since both our proposed scheme and Ref. [26] utilize blockchain for authentication. Our analysis has not considered the communication overhead of blockchain consensus to avoid taking propagation delays and possible errors into account. Our scheme only submits local model upload transactions to a committee node, while Ref. [26] broadcasts transactions across the blockchain network. So our proposed scheme has advantages in communication overhead.

D. Comparison of computational overhead

We now compare the computational overhead taken by RSU using the the cost function of a single authentication cycle $10T_h + 2T_r + 6T_m + 3T_a$ (point addition). While, Fig. 6 provides the execution time for the proposed, we take Refs. [31], [32], [33] and our proposed scheme in account for comparison. The proposed algorithm outperforms all the other schemes in terms of total execution time, added to its improved security.

VII. Conclusion

The effective IoV authentication scheme is an important feature of security. There are multiple security mechanisms were defined to defend the attacks. IoV uses multiple devices to send data over network and reliable communication is possible only through authentication. For authentication, devices should be verified by any means. In this paper signature scheme is used to solve this purpose. Here we used the BLS digital signature scheme and verified the data integrity of remote data. This scheme can properly addresses the depicted attacks in cloud network and secure against delivery of operation.

We created a framework for secure communication which leverages environmental factors to change timestamp, location-stamp, and range-stamp in order to give a precise key management method and handle security, scalability, and authentication challenges. In order to protect against attacks, the system proposes a two-layer identity-based key distribution mechanism and a bilinear pairing-based authentication method of the IOV. To boost efficiency, a pre-generated aggregate signature system and a graph-based blockchain structure are proposed. Our approach outperforms existing methods in terms of security and communication costs, according to the security analysis and performance simulation. The performance simulation also shows that our approach is expandable, and provides a foundation for the future creation of a Blockchain-based safe Key management system.

References

[1] Honggang Wang, Shui Yu, Sherali Zeadally, Danda B Rawat, and Yue Gao. Introduction to the special section on network science for internet of things (iot). *IEEE Transactions on Network Science and Engineering*, 7(1): 237–238, 2020.

[2] Chaowei Yang, Qunying Huang, Zhenlong Li, Kai Liu, and Fei Hu. Big data and cloud computing: innovation opportunities and challenges. *International Journal of Digital Earth*, 10(1):13–53, 2017.

[3] Fadele Ayotunde Alaba, Mazliza Othman, Ibrahim Abaker Targio Hashem, and Faiz Alotaibi. Internet of things security: A survey. *Journal of Network and Computer Applications*, 88:10–28, 2017.

[4] Debiao He, Neeraj Kumar, Sherali Zeadally, and Huaqun Wang. Certificateless provable data possession scheme for cloud-based smart grid data management systems. *IEEE Transactions on Industrial Informatics*, 14(3): 1232–1241, 2017.

[5] Gabriel Neagu, Ştefan Preda, Alexandru Stanciu, and Vladimir Florian. A cloud-iot based sensing service for health monitoring. In *2017 E-Health and Bioengineering Conference (EHB)*, pages 53–56. IEEE, 2017.

[6] Bei Liang and Aikaterini Mitrokotsa. Decentralised functional signatures. *Mobile Networks and Applications*, 24:934–946, 2019.

[7] Keyan Cao, Yefan Liu, Gongjie Meng, and Qimeng Sun. An overview on edge computing research. *IEEE access*, 8:85714–85728, 2020.

[8] M Venkatesh, MR Sumalatha, and C SelvaKumar. Improving public auditability, data possession in data storage security for cloud computing. In *2012 International Conference on Recent Trends in Information Technology*, pages 463–467. IEEE, 2012.

[9] Eric Rescorla. Diffie-hellman key agreement method. Technical report, 1999.

[10] Elette Boyle, Shafi Goldwasser, and Ioana Ivan. Functional signatures and pseudorandom functions. In *Public-Key Cryptography–PKC 2014: 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings 17*, pages 501–519. Springer, 2014.

[11] Robert Johnson, David Molnar, Dawn Song, and David Wagner. Homomorphic signature schemes. In *Topics in Cryptology—CT-RSA 2002: The Cryptographers' Track at the RSA Conference 2002 San Jose, CA, USA, Febru-*

ary 18–22, 2002 Proceedings, pages 244–262. Springer, 2002.

[12] Mikhail Stepanov, Sergey Bezzateev, and Tae-Chul Jung. Privacy homomorphism for delegation of the computations. In *Next Generation Teletraffic and Wired/Wireless Advanced Networking: 6th International Conference, NEW2AN 2006, St. Petersburg, Russia, May 29-June 2, 2006. Proceedings 6*, pages 474–480. Springer, 2006.

[13] Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: a new vision for public-key cryptography. *Communications of the ACM*, 55(11):56–64, 2012.

[14] Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: Definitions and challenges. In *Theory of Cryptography: 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings 8*, pages 253–273. Springer, 2011.

[15] Xiong Fan and Qiang Tang. Making public key functional encryption function private, distributively. In *Public-Key Cryptography–PKC 2018: 21st IACR International Conference on Practice and Theory of Public-Key Cryptography, Rio de Janeiro, Brazil, March 25-29, 2018, Proceedings, Part II 21*, pages 218–244. Springer, 2018.

[16] Nishanth Chandran, Vipul Goyal, Aayush Jain, and Amit Sahai. Functional encryption: Decentralised and delegatable. *Cryptology ePrint Archive*, 2015.

[17] Tatsuaki Okamoto and Katsuyuki Takashima. Efficient attribute-based signatures for non-monotone predicates in the standard model. *IEEE Transactions on Cloud Computing*, 2(4):409–421, 2014.

[18] Man-Ho Au and Atsuko Miyaji. *Provable Security: 9th International Conference, ProvSec 2015, Kanazawa, Japan, November 24-26, 2015, Proceedings*, volume 9451. Springer, 2015.

[19] Xiaodong Lin, Xiaoting Sun, Xiaoyu Wang, Chenxi Zhang, Pin-Han Ho, and Xuemin Shen. Tsvc: Timed efficient and secure vehicular communications with privacy preserving. *IEEE transactions on wireless communications*, 7(12):4987–4998, 2008.

[20] Debiao He, Sherali Zeadally, Baowen Xu, and Xinyi Huang. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Transactions on Information Forensics and Security*, 10(12):2681–2691, 2015.

[21] Xiaoyu Zhang, Hong Zhong, Jie Cui, Irina Bolodurina, and Lu Liu. Lbvp: a lightweight batch verification protocol for fog-based vehicular networks using self-certified public key cryptography. *IEEE Transactions on Vehicular Technology*, 71(5):5519–5533, 2022.

[22] Maria Azees, Pandi Vijayakumar, and Lazarus Jegatha Deboarh. Eaap: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, 18(9):2467–2476, 2017.

[23] Helder Eijs. Python package of low-level cryptographic primitives. *Available online at https://pypi.org/project/pycryptodome/*, 2023.

[24] George Danezis. Library implementing support for computations on groups supporting bilinear pairings. *Available online at https://pypi.org/project/bplib/*, 2019.

[25] Pengcheng Zhao, Yuanhao Huang, Jianping Gao, Ling Xing, Honghai Wu, and Huahong Ma. Federated learning-based collaborative authentication protocol for shared data in social iov. *IEEE Sensors Journal*, 22(7): 7385–7398, 2022.

[26] Guangxia Xu, Jingnan Dong, Chuang Ma, Jun Liu, and Uchani Gutierrez Omar Cliff. A certificateless signcryption mechanism based on blockchain for edge computing. *IEEE Internet of Things Journal*, 2022.

[27] Yanli Ren, Xiangyu Li, Shi-Feng Sun, Xingliang Yuan, and Xinpeng Zhang. Privacy-preserving batch verification signature scheme based on blockchain for vehicular ad-hoc networks. *Journal of Information Security and Applications*, 58:102698, 2021.

[28] Mochan Fan, Zhipeng Zhang, Zonghang Li, Gang Sun, Hongfang Yu, and Mohsen Guizani. Blockchain-based decentralized and lightweight anonymous authentication for federated learning. *IEEE Transactions on Vehicular Technology*, 2023.

[29] Shi-Jinn Horng, Shiang-Feng Tzeng, Po-Hsian Huang, Xian Wang, Tianrui Li, and Muhammad Khurram Khan. An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks. *Information Sciences*, 317:48–66, 2015.

[30] Weizheng Wang, Memon Hussain Fida, Zhuotao Lian, Zhimeng Yin, Quoc-Viet Pham, Thippa Reddy Gadekallu, Kapal Dev, and Chunhua Su. Secure-enhanced federated learning for ai-empowered electric vehicle energy prediction. *IEEE Consumer Electronics Magazine*, 2021.

[31] Leyan Shen, Liangliang Wang, Kai Zhang, Jinguo Li, and Kefei Chen. An efficient conditional privacy-preserving authentication scheme with scalable revocation for vanets. *Journal of Systems Architecture*, 133: 102764, 2022.

[32] Jing Wang, Libing Wu, Kim-Kwang Raymond Choo, and Debiao He. Blockchain-based anonymous authentication with key management for smart grid edge computing infrastructure. *IEEE Transactions on Industrial Informatics*, 16(3):1984–1992, 2019.

[33] Sulaiman M Karim, Adib Habbal, Shehzad Ashraf Chaudhry, and Azeem Irshad. Bsdce-iov: Blockchain-based secure data collection and exchange scheme for iov in 5g environment. *IEEE Access*, 2023.