Ad Hoc Networks

A Privacy-Preserving Authentication Scheme based on Quotient Filter and Elliptic Curve Cryptography in VANET

--Manuscript Draft--

Manuscript Number:	ADHOC-D-20-00227
Article Type:	Research paper
Keywords:	Authentication; Privacy; Quotient Filter; Fog Computing; Big Data; VANET
Corresponding Author:	Shidrokh Goudarzi UKM MALAYSIA
First Author:	Ahmad Soleymani
Order of Authors:	Ahmad Soleymani
	Shidrokh Goudarzi
	Muhammad khuram Khan
	Mohammad Hossein Anisi
Abstract:	"> Vehicular ad hoc network (VANET), as an important network infrastructure in the Industrial Internet of Thing (IIoT), creates an intelligent space for vehicular communications. However, security and privacy are the main issues related to VANET since it is an openaccess environment. To this end, a secure and impressive privacy-preserving authentication scheme can improve safety in VANET. In this paper, a node and message authentication with privacy-preserving is designed wherein the node authentication is based on the quotient filter (QF) which is well-known for the quick querying on big datasets generated in the vehicular network, and the proposed message authentication is established on the elliptic curve cryptography (ECC) since it reduces the computation overhead. To meet the privacypreserving, mapping each vehicle is performed to a different pseudo-identity. Since the security models are latency-sensitive, this work extends the fog computing to the VANET. To this end, fog nodes (FN) are distributed along the road-side. This is mainly because fog nodes with much better processing power than road-side units (RSUs), reduce latency, and thereby enhances system efficiency and throughput. In this work, security analysis indicates that our scheme meets the VANETs' security requirements as well as performance analysis proves the validity of the proposed security scheme to identify illegitimacy vehicle nodes and invalid messages when the fog-enabled VANET is exposed to the attacks.
Suggested Reviewers:	Mazdak Zamani zamani.mazdak@gmail.com
	Mahdi Zareei m.zareei@tec.mx
Opposed Reviewers:	

LP-FTSM-2020-008

A Privacy-Preserving Authentication Scheme based on Quotient Filter and Elliptic Curve Cryptography in VANET

S.A. Soleymani[a], Sh. Goudarzi[b,*], M. H. Anisi[c] and M. Khurram Khan[d]

[a]School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, Johor, Malaysia.
[b]Centre for Artificial Intelligent (CAIT), Universiti Kebangsaan Malaysia, 43600 Bangi, Malaysia.
[c]School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, United Kingdom.
[d]Center of Excellence in Information Assurance (CoEIA), College of Computer & Information Sciences, Building 31, King Saud University, P.O. Box 92144, Riyadh 11653, Kingdom of Saudi Arabia.

ARTICLE INFO

Keywords:
Authentication
Privacy
Quotient Filter
Fog Computing
Big Data
VANET

ABSTRACT

Vehicular ad hoc network (VANET), as an important network infrastructure in the Industrial Internet of Thing (IIoT), creates an intelligent space for vehicular communications. However, security and privacy are the main issues related to VANET since it is an open-access environment. To this end, a secure and impressive privacy-preserving authentication scheme can improve safety in VANET. In this paper, a node and message authentication with privacy-preserving is designed wherein the node authentication is based on the quotient filter (QF) which is well-known for the quick querying on big datasets generated in the vehicular network, and the proposed message authentication is established on the elliptic curve cryptography (ECC) since it reduces the computation overhead. To meet the privacy-preserving, mapping each vehicle is performed to a different pseudo-identity. Since the security models are latency-sensitive, this work extends the fog computing to the VANET. To this end, fog nodes (FN) are distributed along the road-side. This is mainly because fog nodes with much better processing power than road-side units (RSUs), reduce latency, and thereby enhances system efficiency and throughput. In this work, security analysis indicates that our scheme meets the VANETs' security requirements as well as performance analysis proves the validity of the proposed security scheme to identify illegitimacy vehicle nodes and invalid messages when the fog-enabled VANET is exposed to the attacks.

1. Introduction

The smart city's purpose is to enhance the life quality of people by empowering and utilizing technologies leading to smart outcomes. In smart cities, the Internet of things (IoT) characterizes a cyber-physical paradigm, where a wide range of real physical elements are associated and are capable to autonomously interact with each other. This type of consistent network is the empowering agent for intelligent transportation systems (ITS) [1].

ITS is an advanced application that aims to improve safety, mobility, and efficiency to ground transportation. VANET, as a key part of ITS technology, has obtained incrementing attention from both the industry and research communities. As an important network infrastructure in the industrial Internet of Thing (IIoT) [2], it creates an intelligent space for vehicular communications. It is expected to VANET presents new ideas to improve road safety, and infotainment dissemination [3]. However, with the lack of efficient security and privacy, not only private information such as identity, tracing, and preference be compromised by attackers [4] but also an attacker can easily forge the message exchanged among vehicles and RSUs. Therefore, it is required to design secure and efficient authentication with privacy-preserving schemes [5] wherein authentication certifies the legitimacy of vehicle nodes and integrity of the message, and privacy keeps the information protected and private [6].

In this path, one of the most important challenges is the short communication range while the speed of the vehicle is high. As a result, it limits the communication time among RSUs and vehicles. The big data generated on the edge of the network is another challenge in VANET. This is because of the growth of connected nodes in vehicular environments as well as RSUs and sensors deployed in the VANET that lead to generating a large volume of data [7]. In this situation, the need to minimize communication and computation overhead as well as latency are the basic requirements of the security scheme.

Probabilistic data structure (PDS) and fog computing are two concepts that can be used to deal with these challenges. PDS, as a kind of data structure, is particularly suitable for large data because it reduces latency and analytical procedure. Fog computing also decreases the delay and latency by moving the part of the computational power to the edge of the network. To deal with big data issues, fog computing is also able to present elastic resources to large scale data procedure system without the disadvantage of cloud, high latency [3].

In this study, to cope with the security and privacy

*Corresponding author
shidrokh@ukm.edu.my (Sh. Goudarzi)

concerns related to VANET, a privacy-preserving authentication scheme using fog computing for big data analytics is designed. In the proposed scheme, fog computing is integrated into the node and message authentication process, wherein fog nodes are distributed along the roadside. In this work, before initiating any communication, it first needs to check the legitimacy of the node. Verification of node authentication is through a query on the fog node's QF. After start communication and data sharing, the receiver of a signed message has to check the integrity of the message through signature verification.

The key contributions of this work are as follows:

1) We proposed a fog computing-based VANET architecture to reduce latency and in result increase throughput of the proposed security and privacy scheme.

2) We proposed a QF-based node authentication scheme to check the legitimacy of the vehicle node. This scheme aims to deal with illegal nodes who try to join the network. Before any data sharing and communication with other nodes in the network, node authentication verification is required.

3) We proposed a message authentication scheme to guarantee the event message's integrity. This model is established on ECC. The message's signing and single/batch signature verification are the main tasks of this model. We also used the pseudonym to realize privacy-preserving of vehicle nodes.

4) We provided the NS-3 simulation-based practical demonstration of the proposed scheme to measure the impact on transmission delay under different density and velocity with the different percent of malicious nodes distributed in the network.

The remaining of this article is structured as follows. Related works on security models for VANET are clarified in Section 2. In Section 3, information and background on the designed scheme is provided in detail. Section 4 presents the methodology of the proposed scheme. Section 5 presents the analysis of security proof for the suggested scheme. The performance among state-of-the-art methods is compared in Section 6. Ultimately, the conclusion and future work are provided in Section 7.

2. RELATED WORK

Security and privacy are the most important issues related to the vehicular network. On the security issues for VANET, many studies on privacy-preserving authentication have been reported.

Hubaux and Raya [8] suggested a system for signature authentication oriented by public key infrastructure (PKI). The traffic-related data shared in VANETs should be checked in this network before trusting the data. Based on checking authentication and integrity, PKI-based systems are well-selected options. However, in the PKI-based systems, the RSU 's transmission overhead rise with the growth in vehicle numbers since vehicles need to store many pseudonym certificates.

To tackle issues concerning the PKI-based schemes, an effective batch message signature verifying system for the vehicle to infrastructure (V2I) communications is presented in [9]. In the proposed scheme, multiple received messages are simultaneously verified by the RSUs. Compared to the schemes that each message is verified by RSU separately, the total authentication overhead significantly is reduced and hence the VANETs' operational efficiency is enhanced. In addition, as the scheme suggested in [9] is identity dependent, thereby the certificate is not required. Although, this scheme enhances efficiency, however, it fails for example when the number of vehicles is much.

Chim et al. [10] developed a scheme, in which RSU helps neighboring vehicles to authenticate their messages received. In other words, the vehicle node is just responsible for transferring the messages to the RSU, and message verification is the task of RSU. In this scheme, RSU has the role of the cloud for the vehicle. In general, multiple messages are authenticated by the RSU utilizing the batch confirmation method. The messages in a batch are valid when batch verification is carried out successfully. In contrast, when at least one invalid message exists in the batch, it will be discovered by a binary search. The RSU assigns two positive and negative bloom filters, respectively, to store the hash value for valid and invalid messages. Then, the negative and positive filters will be distributed by the RSU at a particular frequency to neighboring vehicles. Therefore, vehicles just need to investigate the two filters for the authentication of messages. Authentication is reduced considerably by this scheme and the entire system's efficiency is improved. However, a large number of vehicles will result in the RSU's decreased computation performance causing considerable delay.

To address this problem, [11] stated the possibility of sharing the computational load on the RSU with adjacent vehicles. In this work, proxy vehicles are elected by the system based on the calculation power. The proxy vehicles will share the verification of the messages performed by the RSU and then the verification results will be sent to the RSU. Next, the accuracy of the results will be evaluated by RSU. Although the RSU's verification performance is significantly improved by the suggested scheme, however, the scheme performance is not enough since the basic operation includes

map-to-point operation and bilinear pairing with large overhead.

A batch verification scheme based on identity (IBV) scheme is designed by Zhang et al. in [12] for VANETs. This scheme decreases the whole confirmation delay of batch message signatures. It also is faster compared to the PKI-based systems. However, this scheme with a huge overhead would lead to performance issues as it is based on bilinear pairing. This is a common problem among all proposed authentication schemes based on bilinear pairing [13, 14].

In order to reduce the computation overhead created by the bilinear pairing method and map-to-point hash function, He et al. [15] proposed a scheme based on ECC. In this scheme, the process of signature generation has been simplified which improves efficiency. However, this scheme is incapable to meet all security requirements.

An ECC-based anonymous privacy-preserving authentication scheme is proposed for VANET in [16]. In this scheme, each message transmitted by a vehicle needs the verification of RSUs. However, the aggregate signature verification has a leak by which a malicious user can construct bogus signatures and muddle throughput the aggregate verification. Also, to meet privacy, each vehicle has a group of pseudo-IDs which increases the memory usage. They also proposed a authentication scheme for the Internet of vehicles in [17]. This scheme is certificate-less scheme that satisfy privacy. In this scheme, each traffic message needs to verified by RSUs. Because of the big data generated in vehicular network, however it increases RSU overhead communications and in result reduce operational efficiency.

In [18], a scheme based on ECC is proposed to message authentication. In this work, for improving message authentication efficiency, a few vehicles are selected as edge nodes to support the RSUs with the message's authentication. It is supposed that RSUs act as the cloud of the vehicles. However, given the very dynamic topology of the network that is related to the high velocity of vehicles, considering vehicle as the edge node cannot be suitable. Also, vehicles are more threatened by destructive nodes, and the selection of reliable vehicles, as the edge nodes, is an important issue. In contrast, RSUs have a high ability in computation than vehicles. Also, since it is difficult for RSU to be threatened by destructive nodes, hence they are more trustable and reliable than vehicle nodes.

Based on available knowledge, there is a lack of a proper security and privacy scheme with the lowest computation overhead, communication overhead, and latency in VANET wherein the number of vehicle nodes and data generated are huge and vehicles also moving fast. In this network, efficient security and privacy scheme are required that not only needs to ensure the legitimacy of vehicle nodes, the integrity of the message, and meet privacy-preserving but also deal with concerns related to big data.

3. Background

3.1. Network Model

In this work, a fog-enabled VANET architecture is proposed. As shown in Figure 1, this architecture includes two layers: upper and lower. The upper layer comprises of cloud servers (CS) and root trusted authority (TA), whereas the lower layer consists of fog nodes, RSUs, and vehicle nodes.

Upper Layer: Cloud servers are employed in this layer to offer high computing power and reliable and permanent data storage, whereas the master secret and global system parameters and issues credentials for the vehicles and fog nodes are generated by the TA. Moreover, TA is responsible for recovering the vehicles' real identities signing and disseminating bogus messages. Trace authority (TRA), as a part of TA, creates pseudonyms for vehicles and it is also capable to track the real identity from the pseudonyms used by the vehicle.

Lower Layer: This layer comprises of fog and vehicular layer. In the fog layer, fog nodes and RSUs are deployed along the roadside. RSUs are supposed to host the fog nodes and connect to cloud through a secure manner using wired communication technologies such as Ethernet. RSUs are equipped with persistent links to a service provider hosted on the cloud. They also communicate with vehicles with short-range communication capabilities. RSUs continuously monitor various parameters and transmit the required aggregated data to the FNs. RSUs are also able to generate a notification for vehicle nodes when it is necessary. FNs are equipped with communication capabilities, processing power, and storage space. FN interacts with vehicle nodes who are within its communication area via the open wireless technologies such as 4G/LTE/5G. It is worth noting that FN with much better processing power than RSUs, reduce latency, and thereby increasing throughput.

In the vehicular layer, to enhance the operational efficiency of traffic security and regional traffic, vehicle nodes periodically broadcast the traffic-related data to the local area by using IEEE 802.11p protocol. Vehicles are equipped with a range of internal sensors that able to detect events within the transmission range. Vehicles are also equipped with a realistic tamper-proof device (TPD) for storing the secure substances received from the group key and TA. The medium utilized for communications between vehicles and fog nodes is 5.9-GHz DSRC recognized as IEEE 802.11p.

A simplified view of how RSU and FN are used in the fog layer to assist vehicles during mobility from one geographic location to another domain is shown

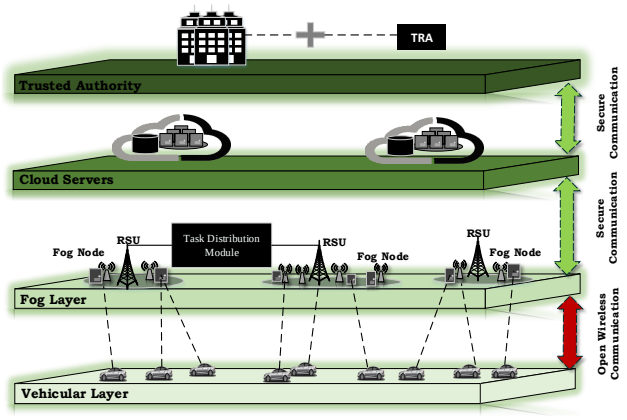

Figure 1: Fog-based VANET architecture.

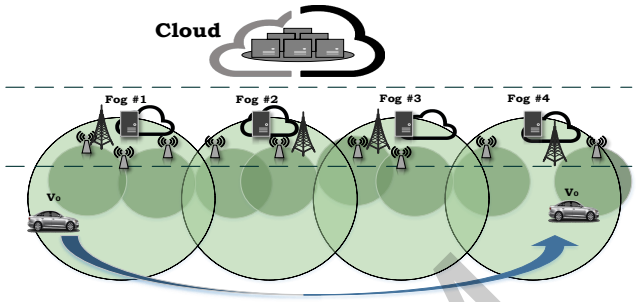

Figure 2: Vehicle node mobility in fog computing.

in Figure 2. It is supposed that RSUs cover the whole area in the network and FN's communication range covers a region of the vehicular environment and can involve several intersections [19]. When a vehicle node physically located within the communication range of the fog nodes, it can send and receive data to and from the fog nodes. For example, when a vehicle enters a region covered by fog node, it will send its speed, current location, and road conditions to the specific node frequently until it leaves this region. Based on this assumption, a vehicle continuously will be supported by fog nodes. Whenever a vehicle node is under the coverage of multiple access fog nodes or RSU, it needs to select the most suitable RSU/FN to send and receive data. To this end, the vehicle node first calculates the link quality between itself and nearby FNs, if exist. Otherwise, it computes quality of link with the existing RSUs. According to [20], the link quality can be measured based on some parameters such as bandwidth, signal the noise ratio (SNR), and bit error rate (BER) that is out of the scope of this paper.

Additionally, due to the large number of tasks created by vehicles for processing using FNs, it needs to monitor fog nodes in terms of computational power, memory availability, and CPU availability, as well as tasks loaded. To this purpose, a module on the RSUs is developed to collect information on distributed fog nodes. Then, it computes tasks locally and offload them to fog nodes for processing. The task distribution mechanism greatly reduces delay for the latency-sensitive applications and enhances the overall system scalability.

3.2. Security Requirements

According to [18], a well-designed privacy-preserving message and node verification scheme need to meet the security requirements:

1) **Message Verification and Integrity:** An FN verifies the signed message has not been forged or modified by malicious nodes once receives the message from the authorized vehicles.

2) **Resistance to Unauthorized Nodes:** An illegal and unregistered node cannot join the network and start any communication with existing nodes in the network.

3) **Identity Preserving Privacy:** The vehicle's real identity should endure anonymously and no third party extracts the real identity and private information from the vehicle's pseudo-identity.

4) **Resistance to Replay Attack:** A malicious node is unable to store the gathered signed messages and disseminate it when the validity of the message is expired.

5) **Traceability:** TRA can trace a vehicle's real identity by analyzing the pseudo-identity extracted from the message.

3.3. Elliptic Curve Cryptography (ECC)

This is a public-key cryptography approach focused on the elliptic curves over finite fields. Consider a set of the elliptic curve point $E_P(a, b)$ is defined as follows:

- $E_p\,(a, b) : y^2 = x^3 + ax + b\,(mod\ \ p)\,; a, b \in F_p$
- F_p is a finite field that is defined by p as a prime number
- $\left(4a^3 + 27b^2\right)\ mod\ \ p \neq 0$

In the elliptic curve cryptography, some of the computational problems are hard to be solved such as discrete logarithm problem (DLP) [21]. Let G be an additive elliptic curve group of order q and $P, Q \in G$ as two random numbers on E where $Q = x.P$. Based on the DLP, it is not easy to compute x from Q.

3.4. Fog Computing

The cloud computing services are extended by fog computing to the network edge [4]. It is a greatly virtualized platform providing storage, computation, and networking services between traditional cloud servers and end tools. The combination of VANET with fog

computing will provide numerous advantages such as local data processing, local resource pooling, cache data management, load balancing, and delay decrease [22]. In fog computing-based VANET, the time-critical data locally are analyzed through the fog node tools leading to the lower latency. It is worth stating that by fog computing, the interactions between vehicle nodes are facilitated and very effective collaboration of nodes with each other becomes possible [23].

3.5. Quotient Filter (QF)

QF, as a cache-friendly and space-efficient probabilistic data structure, is useful for large data because it reduces latency and analytical procedure. It representing a multiset of elements $S \subseteq U$ by storing a p bit fingerprint for each element. QF stores the multiset $F = h(S) = \{h(x) \mid x \in S\}$, where $h : U \to \{0, \cdots, 2p-1\}$ is a hash function.

Conceptually, it is assumed that F has been stored in an open hash table T with $m = 2^q$ buckets utilizing a method known as the quotient, proposed by Knuth [24]. In this method, a fingerprint f is divided into its r least significant bits, $f_r = f \ mod \ 2^r$ (the remainder), and its $q = p - r$ most significant bits, $f_q = \lfloor f/2^r \rfloor$ (the quotient). For inserting a fingerprint f into F, we store f_r in bucket $T\left[f_q\right]$. Considering a remainder f_r in bucket f_q, the full fingerprint can be exclusively reconstructed as $f = f_q 2^r + f_r$ [25].

4. Proposed Scheme

Security and privacy are real significant in VANET since it is an open-access environment [6]. Building on this, we designed a secure and efficient message and node authentication scheme with privacy-preserving. In the proposed scheme, node authentication is based on QF whereas message authentication is established on ECC. In order to meet the privacy-preserving, mapping each vehicle is also performed to a different pseudo-identity. In this section, we describe our scheme in the following phases: system initialization, registration, and authentication.

4.1. Initialization Phase

In this phase, TA produces the required parameters of system, first. Then, it preloads parameters into the TPD of vehicles and memory of fog nodes. To this end, considering two primes p, q; group G of order q; and let two distinct generators $P, Q \in G$. TA randomly chooses an at least 160 bits number $s \in Z_q^*$ as the master private key. Using the master private key, it also computes the corresponding public key $P_{pub} = s.P$. Then, the TA selects a secure SHA-256 hash function $h : \{0,1\}^* \to G$. This is mainly because it is difficult to reconstruct the initial data from the hash value generated by SHA-256. Also, it is impossible that SHA-256 creates the same hash value for different messages.

Table 1
Definition of Notations in the Proposed Scheme

Model	Method
$\oplus$	XOR operation
$\|\|$	Concatenation operation
TA	Trusted authority
TPD	Tamper-proof device
TRA	Trace authority
CS	Cloud server
RSU	Roadside unit
FN	Fog node
V	Vehicle node
h	Secure hash function
PID	Pseudo-identity
RID	Real identity
P_{pub}	System public key
G	Cycle additive group
s	System private key
$params$	System public parameters
P, Q	Distinct generators of G
$\tau, \tau^{'}$	Signature generated by vehicle and RSU/FN, resp.
t	Timestamp of message
VP	Timestamp of pseudo-identity

Next, TA sets the system public parameters $params = \left\{p, q, a, b, G, P, P_{pub}, h\right\}$ and publishes $params$ to the cloud servers, RSUs, fog nodes, and vehicles where a and b are the parameters of the elliptic curve function $E_P(a, b)$. The notations utilized throughout this work are illustrated in Table I.

4.2. Registration Phase

In this phase, TA accomplishes the registration of vehicles, RSUs, fog nodes, and cloud servers as follows:

1) Registration of Fog Node
Let $\mathfrak{F}_{FN} = \{FN_1, FN_2, \cdots, FN_M\}$ be a set of authorized FNs that have been registered in the network. TA chooses a unique identity RID_{FN_k} for each $FN_k \in \mathfrak{F}_{FN}$. It also randomly selects a number $s_{fn} \in Z_q^*$ as the private secret key of FN and then computes the FN's public key $PUB_{fn} = s_{fn}.P$.

2) Registration of Vehicle Node
Consider a set of authorized vehicle nodes that have been registered in the network $\mathfrak{V}_v = \{V_1, V_2, \cdots, V_N\}$. For each vehicle $V_l \in \mathfrak{V}_v$, the TA chooses a unique identity RID_{V_l}. Each vehicle maintains its own real identity RID_{V_l} and password PWD_{V_l} in the TPD. TA also sends securely system private key s for authorized vehicles and vehicle stores it in TPD. To meet the privacy-preserving, each vehicle uses the generated pseudo-identity $PID_V = \{PID_{V,1}, PID_{V,2}\}$ by TPD and TRA that we explain more next.

3) Registration of Roadside Unit
Consider a set of RSUs that have been registered in the network $\mathfrak{R}_{rsu} = \{RSU_1, RSU_2, \cdots, RSU_L\}$. For each $RSU_j \in \mathfrak{R}_{rsu}$ to be deployed, the TA selects a unique real identity RID_{rsu_j}. It picks a random number $s_{rsu} \in Z_q^*$ as the private secret key of RSU and then computes the public key $PUB_{rsu} = s_{rsu}.P$.

4) Registration of Cloud Server
Let $\mathfrak{C}_{CS} = \{CS_1, CS_2, \cdots, CS_P\}$ be a set of authorized cloud servers that have been registered in the network. For each cloud server $CS_i \in \mathfrak{C}_{CS}$, the TA chooses a unique identity RID_{CS_i}. TA also randomly selects a random number $s_{cs} \in Z_q^*$ as the master private key of the cloud server. Then, it calculates the CS's public key using $PUB_{cs} = s_{cs}.P$.

It is worth noting that, because the privacy is not an important issue and a requirement for the fog nodes and cloud servers, hence they use the real identity to sign the message.

4.3. Authentication Phase

In this section, we explain both node and message authentication and verification procedures as follows:

4.3.1. QF-based Node Authentication Scheme

In the vehicular network, exchange data among nodes is the basis of the network. To ensure security, before initiating any communication and data sharing, the receiver of data needs to check the legitimacy of the sender. Due to the big data generated in the network and a large number of vehicle nodes, we proposed a node authentication scheme based on QF. As described above, QF is a probabilistic data structure for query of massive dataset that used to decrease processing overhead and improve security [26].

In this scheme, each vehicle V is equipped with a quotient filter QF_V to maintain the information of all authorized and unauthorized vehicle nodes. Depending legitimacy and or illegitimacy of vehicle nodes belonging to the RSU_k, they will be stored in the relevant quotient filter of the vehicle using the fingerprint of pseudo vehicle identity (PID_W), a public key (PUB_{rsu}) provided by the related RSU, and (A : *authorized* or U : *unauthorized*) as follows:

$$(QF_V) \leftarrow h\left(fingerprint\left(PID_W\right) \oplus PUB_{rsu}\right) \parallel A/U \tag{1}$$

RSU updates the QF_V of the legitimate vehicle nodes who are under its transmission range immediately after a change in the list of authorized and unauthorized vehicle nodes. It will be performed by broadcasting the new list to the nearby vehicle nodes.

As the same way, each FN maintains own quotient filter QF_{FN} of all genuine and fake vehicle nodes. Like the QF_V, all QF_{FN} continuously upgraded by the relevant RSU.

In a vehicle-to-vehicle communication, before data sharing and communication, the destination node V_j performs $Query(V_i)$ on its QF_{V_j}. If the query returns $TRUE$ with A, it means the V_i is a genuine node, and If the query returns $TRUE$ with U, it means the V_i is an unauthorized node otherwise, if the query returns $FALSE$, it means the V_i is not a member of the QF_{V_j}, hence V_j immediately sends a request to the FN_k. When the FN_k receives the request, it will check the legitimacy of node V_i by performing a query on QF_{RSU_k}. If the query on QF_{FN_k} returns $TRUE$ with A, V_j start data sharing with V_i and updates QF_{V_j} and if the query returns $TRUE$ with U, V_j stop any communication with V_i and updates QF_{V_j}. Otherwise, if the query on QF_{FS_k} returns $FALSE$, it means that V_i has not been registered in the network and so it is a fake vehicle node has entered the network. Additionally, if V_j does not receive a reply after a certain time from the RSU_k, it just rejects the request of communication and data sharing with V_i.

In a vehicle-to-fog node communication, FN_k performs the query on its QF_{FN_k}. If the query returns $TRUE$ with A, the link between vehicle and FN will be established because the vehicle node is authorized. Otherwise, if the query returns $TRUE$ with U or it returns $FALSE$, the link request will be rejected by FN.

4.3.2. Message Authentication Scheme

In fog computing-based VANET, raw data can be gathered by sensors installed on the vehicle node, and stored in on-board storage. Because of the redundancy of the raw data, the processing of data is conducted to extract valuable information. Then, for further processing in terms of integrity and reliability of data, the vehicle node signs the extracted information and sends it to the relevant FN/RSU. After verifies the vehicle's signature and checking the data reliability, FN/RSU also signs the message and broadcasts in the vehicular network. Once a vehicle node received a signed message from FN/RSU, it checks the signature first and then signs the message for broadcasting to the neighbour vehicle nodes and nearby FNs/RSUs.

Based on the defined architecture in this study (see Figure 1), the following communications in the fog-based VANET are conceivable: vehicle and vehicle (V-V), vehicle and FN (V-FN), vehicle and RSU (V-RSU), FN and RSU (FN-RSU), and RSU and CS (RSU-CS). It supposes that both FN-RSU and RSU-CS communication are via a secure manner. Therefore, we focus on other communications and explain the message authentication in the following.

2.1) V-FN COMMUNICATION

[Message signing by Vehicle]: To ensure authentication and message integrity, each message should be signed by the vehicle before sending to neighbour nodes. Also, to satisfy privacy-preserving, each vehicle node has to use its pseudo-identity. Each message will be signed by a vehicle generating a pseudo-identity and related signing key.

To generate pseudo-identity, TPD randomly selects a number $r_i \in Z_q^*$ and calculates $PID_{i,1} = r_i.P$. When a vehicle node enters the VANET, TPD securely sends $SIG_{VSK_i}(RID_i, PWD_i, PID_{i,1})$ to TRA for verifying $\{RID_i, PWD_i\}$. After verifying the signature using the vehicle's public key, TRA calculates the pseudo-identity $PID_i = \{PID_{i,1}, PID_{i,2}, VP_i\}$ by choosing a random number $r_i \in Z_q^*$, where $PID_{i,2} = RID_i \oplus h\left(r_i.P_{pub}, VP_i\right)$, and VP_i defines the valid period of the PID_i. The generated pseudo-identities are valid within VP_i. This frequent change is mainly because when a vehicle uses a pseudo-identity constantly within the vehicular communication, the vehicle movement trajectory can be traced by an adversary.

Then, vehicle V_i has to sign the $M_i = PID_{i,2} \parallel m_i \parallel t_i$ where M_i is combining of $PID_{i,2}$ as a part of pseudo-identity, m_i as message and t_i as the timestamp that gives the freshness of the signed message against a replay attack. To sign the M_i, TPD selects $r_i \in Z_q^*$. Next, it computes the corresponding signature $\tau_i = r_i + s.h(M_i)$ on M_i for PID_i. Then, the vehicle sends $\{PID_i, M_i, \tau_i\}$ to the relevant RSU/FN.

[Message verification by FN]: Once an FN receives the signed message from vehicles, not only it has to verify the vehicle node authentication, but also it needs to verify the signature of the message. It ensures the vehicle is not attempting to impersonate legitimate vehicles or spread false message. If the vehicle node is genuine (see Section 4.3.1), it verifies the signed message as follows:

- Single Message Verification: Once a fog node $FN_j \in \mathfrak{F}_{FN}$ receives a signed message $\{PID_i, M_i, \tau_i\}$, after checking the freshness of $t_i - t_c \leq \Delta t$ and VP_i, if the message and pseudo-identity have not expired, it calculates $h(PID_{i,2} \parallel m_i \parallel t_i)$ and verifies whether

$$\tau_i.P = PID_{i,1} + P_{pub}.h(PID_{i,2} \parallel m_i \parallel t_i) \tag{2}$$

hold or not. If so, the message will be verified; otherwise, FN_j discards the message and recommend the vehicle with PID_i as an illegal vehicle node to the relevant RSU.

- Batch Message Verification: Once the fog node FN_j receives multiple signed messages from vehicles in a time interval, it uses the batch message verification method as follows:

Consider n distinct vehicles $\mathfrak{V}_V = \{V_1, \cdots, V_n\}$ and corresponding message-signature tuples

$$SML = \{\{PID_1, M_1, \tau_1\}, \cdots, \{PID_n, M_n, \tau_n\}\}$$

To sign verification, the fog node FN_j computes $h(PID_{i,2}||m_i||t_i)$ for $i = 1, \cdots, n$ and then checks whether

$$\left(\sum_{i=1}^{n} v_i.\tau_i\right).P = \left(\sum_{i=1}^{n} v_i.PID_{i,1}\right) + \left(\sum_{i=1}^{n} v_i.h\left(PID_{i,2}||m_i||t_i\right)\right).P_{pub} \tag{3}$$

holds or not. If holds, it means the checking was successfully and hence accept the signatures, otherwise, it indicates there is at least one invalid message in the batch. In the following, due to the $P_{pub} = s.P$, $PID_{i,1} = r_i.P$, $PID_{i,2} = RID_i \oplus h\left(r_i.P_{pub}\right)$, $M_i = PID_{i,2} \parallel m_i \parallel t_i$ and $\tau_i = r_i + s.h(M_i)$, we prove the validation of the batch message verification.

$$\begin{aligned}
\left(\sum_{i=1}^{n} v_i.\tau_i\right).P &= \left(\sum_{i=1}^{n} v_i.\left(r_i + s.h(M_i)\right)\right).P \\
&= \left(\sum_{i=1}^{n} v_i.r_i\right).P + \left(\sum_{i=1}^{n} v_i.s.h(M_i)\right).P \\
&= \left(\sum_{i=1}^{n} v_i.r_i.P\right) + \left(\sum_{i=1}^{n} v_i.s.P.h(M_i)\right) \\
&= \left(\sum_{i=1}^{n} v_i.PID_{i,1}\right) + \left(\sum_{i=1}^{n} v_i.h(M_i)\right).P_{pub} \\
&= \left(\sum_{i=1}^{n} v_i.PID_{i,1}\right) + \left(\sum_{i=1}^{n} v_i.h(PID_{i,2} \parallel m_i \parallel t_i)\right).P_{pub}
\end{aligned}$$

After performing the batch message verification by the fog node FN_j, a recursive algorithm based on the binary search is performed to detect the invalid messages contained in the batch.

In the proposed algorithm, we have considered a batch segmentation and both single message verification and batch message verification. The desired batch contained signed event messages from *Lindex* till *Hindex* will be divided into two separate batches by this algorithm. The first batch is from *Lindex* till *Mindex* = $(Lindex + Hindex)/2$ and the later one is from *Mindex* + 1 till *Hindex*. After each segmentation, the batch message verification will be used to verify the new batches. If each new batch holds Equation (3), existing messages in the batch will be inserted to the *vML* and algorithm immediately will be stopped for this batch. Otherwise, segmentation will be continued until finding invalid message(s). When there is two or one message in the batch, the single message verification by Equation (2) will be used to check validity of the message. If Equation (2) is established, this message will be inserted into the *vML*, otherwise, it goes to *ivML*.

The output of the algorithm is two lists namely vML and $ivML$. Finally, the fog node FN_j signs the $List = \{vML, ivML\}$ and sends to the related RSU. When the RSU receives the list from a fog node, it verifies the signature using the fog node public key PUB_{fn}, and then upgrade its quotient filter QF_{RSU} and nearby fog nodes QF_{FN} .

2.2) V-RSU Communication

Due to the expansion of the transportation network, the distributed fog nodes cannot cover all locations in the vehicular environment. Therefore, a vehicle sometimes is not under communication coverage of fog node. In this situation, it needs to communicate with the related RSU, and hence sends the signed messages to the RSU.

To verification the signed message, the RSU verifies whether the batch authentication Equation 3 holds or not. If the equation is established, it indicates that the batch of the message passes the check. Otherwise, it means that the message contains at least one invalid message. In order to determine the invalid messages in the batch, the RSU uses the binary search algorithm.

2.3) RSU-V / FN-V Communication

[Message signing by RSU]: To ensure secure communication, each RSU/FN also has to sign the event message and then broadcast it to the nearby vehicle nodes. To this end, the $RSU_i \in \mathfrak{R}_{rsu}$ signs $M_i = RID_{rsu_i}||m_i||t_i$ with private key s_{rsu}. Because privacy is not an important issue and a requirement for the RSUs, hence its real identity (RID_{rsu_i}) is used to sign the message. The corresponding signature on M_i is $\tau'_{RSU_i} = s_{rsu}.h(M_i)$ and the RSU broadcasts $\left\{RID_{RSU_i}, M_i, \tau'_{RSU_i}\right\}$ to vehicles and the relevant FNs.

[Message verification by Vehicle]: Once a vehicle receives the signed message from the RSU, it has to verify the signature of the message to ensure that the RSU is not attempting to impersonate any other legitimate RSUs or disseminate false messages. To this end, when a vehicle (V_j) receives a signed message $\left\{RID_{RSU_i}, M_i, \tau'_{RSU_i}\right\}$, after checking the freshness of t_i, it verifies whether

$$\tau'_{RSU_i}.P = PUB_{rsu_i}.h\left(RID_{RSU_i}||m_i||t_i\right) \quad (4)$$

hold or not. If it does not hold, Vehicle discards the message and marked the RSU as an intruder and broadcast an alert to authorized RSU in its communication range. Otherwise, if the equation is established, accept the message.

2.4) V-V Communication

Once a vehicle V_l receives a signed message from another vehicle V_k, it firstly needs to check the legitimacy of V_k (see Section 4.3.1). If V_k is valid, it checks the integrity of the message $M_k = PID_k||m_k||t_k$. In a V-V communication, to check the integrity of the message, V_l sends a request $Req = \langle Req_{id}, PID_l, M_k, PID_k\rangle$ to the related fog node and wait for a reply. As mentioned above, if no fog node is in its communication range, it sends the request to the related RSU.

When the fog node FN_j receives a request from the vehicle, it checks to determine whether $\langle M_k, PID_k\rangle$ is within the vML or not. If exist, to verify the message M_k , FN_j sends a reply $Rep(verified)$ to the V_l. Otherwise, FN_j checks message in $ivML$. If exist, it responds $Rep(ignored)$ to V_l. But, if the message is not existing in both vML and $ivML$, it means V_k did not send the message to the FN_j and or the fog node FN_j received the message after the request of V_l. In this regard, FN_j waits for a certain time. If it received the message during this time, FN_j checks the authentication of the message using a single authentication method and responds the result to the V_l. Otherwise, it sends $Rep(ignored)$ to V_l.

In the vehicle side, if V_l receives a reply of FN_j, the vehicle verified/ignored the message based on the type of reply. Otherwise, if V_l not received a reply after a certain time, the message will be discarded.

The pseudo-identity generation process, signature generation and signature verification between vehicle and fog node (V-FN) and between RSU and vehicle/fog node (RSU-V/FN) can also be found in Figure 3.

5. Security Analysis

In this section, we prove that our scheme meets the security requirements mentioned in subsection 3.2 and resist attacks. Firstly, we give a proof that our scheme is secure with the random oracle model. This is because proof in the random oracle model ensures the security of the overall design of a signature scheme [27]. To this purpose, Theorem 1 gives a formal proof of the proposed signature scheme against an alternatively chosen message attack using a game between challenger and an adversary as follows:

Theorem 1: Our system is secure and efficient with random oracles for VANET.

Proof: Let in our system, the security model is established by a challenger $\mathcal{CH}$ and an adversary $\mathcal{ADV}$, in which $\mathcal{ADV}$ is able to forge the message $\{PID_i, M_i, \tau_i\}$. Consider a game between $\mathcal{CH}$ and $\mathcal{ADV}$, which can solve the discrete logarithm problem (DLP) by running $\mathcal{ADV}$ with a non-negligible probability. To that end, it is assumed that $\mathcal{CH}$ maintains three hash lists $List_{H_1}$, $List_{H_2}$ and $List_{H_3}$ which are initialized to empty.

Setup: $\mathcal{CH}$ selects a random number s as the private key of the system and compute the public key using $P_{pub} = s.P$. Then, $\mathcal{CH}$ sends the generated system parameters $params = \left\{p, q, P, Q, P_{pub}, H_1, H_2, H_3\right\}$ to $\mathcal{ADV}$.

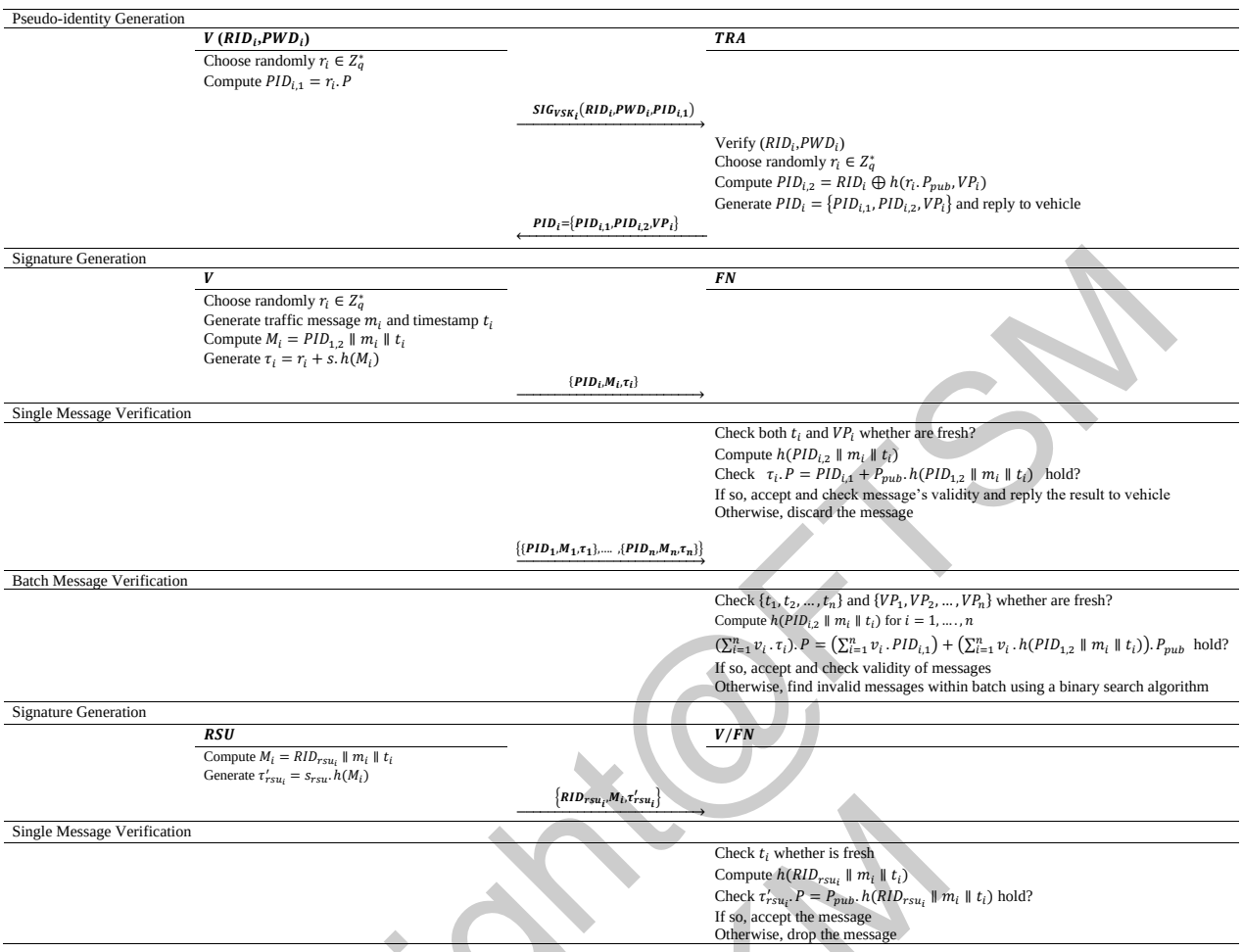

Figure 3: Pseudo-identity generation and authentication processes of our scheme.

H_1**-Oracle**: $\mathcal{CH}$ keeps a list $(List_{H_1})$ with the form of $\langle m, \tau \rangle$. When $\mathcal{ADV}$ creates a H_1 query with message m, $\mathcal{CH}$ checks whether the tuple $\langle m, \tau \rangle$ is already in the $List_{H_1}$ or not. If so, $\mathcal{CH}$ sends $\tau = H_1(m)$ to $\mathcal{ADV}$; if not, $\mathcal{CH}$ selects a random $\tau \in Z_q^*$ and adds $\langle m, \tau \rangle$ into the $List_{H_1}$. Finally, $\mathcal{CH}$ sends $\tau = H_1(m)$ to $\mathcal{ADV}$.

H_2**-Oracle**: $\mathcal{CH}$ keeps a list $(List_{H_2})$ with the form of $\langle PID_i, m, \tau \rangle$ When $\mathcal{ADV}$ creates a H_2 query with the message $\langle PID_i, m, \tau \rangle$, $\mathcal{CH}$ exams whether the tuple $\langle PID_i, m, \tau \rangle$ is already in the $List_{H_2}$ or not. If so, $\mathcal{CH}$ sends $\tau = H_2(PID_i||m)$ to $\mathcal{ADV}$. Otherwise, $\mathcal{CH}$ selects a random $\tau \in Z_q^*$ and then adds $\{PID_i, m, \tau\}$ into the $List_{H_2}$. In the end, $\mathcal{CH}$ sends $\tau = H_2(PID_i||m)$ to $\mathcal{ADV}$.

H_3**-Oracle**: $\mathcal{CH}$ keeps a list $(List_{H_3})$ with the form of $\langle PID_i, M_i, \tau \rangle$ in which $M_i = m_i||t$. Once $\mathcal{CH}$ receives a query of $\mathcal{ADV}$ creates with the message $\langle PID_i, M_i, \tau \rangle$, it checks whether the tuple $\langle PID_i, M_i, \tau \rangle$ is already in the $List_{H_3}$ or not. If so, $\mathcal{CH}$ sends $\tau = H_3(PID_i||M_i)$ to $\mathcal{ADV}$. Otherwise, $\mathcal{CH}$ selects a random $\tau \in Z_q^*$ and then adds $\{PID_i, M_i, \tau\}$ into the $List_{H_3}$. In the end, $\mathcal{CH}$ sends $\tau = H_3(PID_i||M_i)$ to $\mathcal{ADV}$.

Sign-Oracle: Upon receive a query of $\mathcal{ADV}$ with the message m, $\mathcal{CH}$ generates three random numbers $\alpha_i, \beta_i, \tau_i \in Z_q^*$ and chooses a random point $PID_{i,2}$ and computes $PID_{i,1} = \tau_i.P - P_{pub}.h(PID_{i,2}||m_i||t_i)$. Then, $\mathcal{CH}$ adds $\langle PID_i, m_i, \alpha_i \rangle$ and $\langle PID_i, M_i, \beta_i \rangle$, respectively, into the $List_{H_2}$ and $List_{H_3}$ in which $PID_i = \{PID_{i,1}, PID_{i,2}\}$. Next, $\mathcal{CH}$ sends $\langle PID_i, M_i, \tau_i \rangle$ to $\mathcal{ADV}$. It is easy to verify the equation $\tau_i.P = PID_{i,1} + P_{pub}.h(PID_{i,2}||m_i||t_i)$ holds. Therefore, all signatures generated by $\mathcal{CH}$ are indistinguishable from those generated by legal vehicles. Finally, $\mathcal{ADV}$ outputs a message $\langle PID_i, M_i, \tau_i \rangle$ and $\mathcal{CH}$ checks whether $\tau_i.P = PID_{i,1} + P_{pub}.h(PID_{i,2}||m_i||t_i)$ is established or not. If no, $\mathcal{CH}$ aborts the process.

By using Forking Lemma [27], $\mathcal{ADV}$ produces another valid message $\left\langle PID_i, M_i, \tau_i^{'} \right\rangle$. In the valid messages $\langle PID_i, M_i, \tau_i \rangle$ and $\left\langle PID_i, M_i, \tau_i^{'} \right\rangle$, the signatures $\tau_i = r_i + s.h(M_i)$ and $\tau_i^{'} = r_i + s.h^{'}(M_i)$ where $h \neq h^{'}$ are produced by $\mathcal{CH}$ within polynomial running time. According to these two signatures, $\mathcal{CH}$ achieves the

value of $x = \left(\tau_i - \tau_i^{'}/h - h^{'}\right) \ mod \ \ q$ as the answer of the DLP. To prove this, with substituting $r_i = \tau_i - s.h$ in the $r_i = \tau_i^{'} - s.h^{'}$, it gives the following result:

$$\tau_i - s.h = \tau_i^{'} - s.h^{'} \Rightarrow \tau_i - \tau_i^{'} = s\left(h - h^{'}\right) \Rightarrow$$
$$s = \left(\tau_i - \tau_i^{'}/h - h^{'}\right) mod \ \ q$$

The ability to solve the DL problem contradicts the hardness of this problem. Therefore, the proposed scheme is secure against forgery under adaptive chosen message attack in the random oracle model, hence it provides message authentication for VANETs.

Theorem 2: (Verification and Integrity of Message) the message's integrity is ensured by the signature of the message.

Proof: Proof in the random oracle model ensures the security of the signature scheme. As discussed in Theorem 1, our proposed signature is secure against an alternatively chosen message attack under the random oracle model, and as a result, a malicious attacker cannot forge valid signatures.

Theorem 3: (Resistance to Unauthorized Nodes) it guarantees an unauthorized and fake node cannot enter the network and initiating data sharing with authorized nodes.

Proof: Each vehicle has a filter namely QF_V containing the pseudo-identity of genuine and fake nodes. Before starting any communication, the vehicle node which has a request for communicating checks the validity and legitimacy of another vehicle node using the query on the filter. If the query returns $FALSE$, the vehicle node immediately sends a request to the related FN. If the query on the filter QF_{FN} returns $FALSE$, it indicates that the vehicle node did not register in TA before joining the VANET and hence it marks the vehicle node as a fake node. Also, if the query on QF_{FS} returns $TRUE$, it means that the FN has been detected the vehicle as an illegitimate node, previously. Consequently, a fake and unauthorized vehicle node cannot join the network and initiate any communication with other vehicles and fog nodes.

Theorem 4: (Privacy-Preserving) during the communication, no adversary can extract the vehicle's real identity from its pseudonym.

Proof: The vehicle V_i transmits message $\{PID_i, M_i, \tau_i\}$ to other nodes, where $PID_i = \{PID_{i,1}, PID_{i,2}, VP_i\}$, $PID_{i,1} = r_i.P$, and $PID_{i,2} = RID_i \oplus h\left(r_i.P_{pub}, VP_i\right)$. The real identity RID_i of the vehicle is perfectly concealed since PID_i is an unknown identity with a random number r_i. Based on the DLP, it is hard to compute the private key r_i of the vehicle through $PID_{i,1}$ and P. Hence, the adversary is unable to extract RID_i and as a result, the proposed scheme satisfies privacy-preserving. Furthermore, in our scheme, a vehicle node changes pseudo-identity PID_i after a valid period of time VP_i. This frequent change is mainly because when a vehicle uses a pseudo-identity constantly within the vehicular communication, the vehicle movement trajectory can be traced by an adversary [9]. We prove that the relation between the pseudo-identities can be revealed only by TRA. To this end, consider two pseudo-IDs $PID_{i,2}$ and $PID_{i+1,2}$ related to the vehicle node RID_i where $PID_{i,2} = RID_i \oplus h\left(r_i.P_{pub}, VP_i\right)$ and $PID_{i+1,2} = RID_i \oplus h\left(r_{i+1}.P_{pub}, VP_{i+1}\right)$. Assuming that attacker knows P_{pub}, VP_i, and VP_{i+1}. To verify relation of $PID_{i,2}$ and $PID_{i+1,2}$ with RID_i, the attacker should computes both $h^{-1}\left(r_i.P_{pub}, VP_i\right)$ and $h^{-1}\left(r_{i+1}.P_{pub}, VP_{i+1}\right)$. These computations are performed until the relation verification is confirmed. As described in [5], for a n-bit one-way hash function, the complexity of solving h^{-1} is $O(2^{n-1})$. Suppose $PID_{i,2}$ an d $PID_{i+1,2}$ belong to RID_i, hence for each $h^{-1}\left(r_i.P_{pub}, VP_i\right)$, 2^{n-1} times of $h^{-1}\left(r_{i+1}.P_{pub}, VP_{i+1}\right)$ operation needs to confirm. So, the total complexity is $O(2^{2n-2})$. Since the hash function used in our scheme is a SHA-256, hence the relationship verification between two pseudo identities is not easy computational problem.

Theorem 5: (Resistance to Replay Attack) an adversary is unable to broadcast the received signed message if it is expired.

Proof: Signature of the message includes the timestamp capable of resisting replayed attacks. The timestamp t_i is attached with the message m_i and all vehicles preserve time synchronization. The current timestamp is employed for all communicating entities. In each exchanged message, the highest transmission delay is typically a small value. Hence, even if the intercepted messages are replayed by an adversary, they are simply discovered in our scheme owing to timestamp validation by the receiving participants. Consider an adversary $\mathcal{ADV}$ intercepts a message $\{PID_i, M_i, \tau_i\}$ where $M_i = m_i||t_i$ and it presents a replay attack at the time t_j. Due to the $t_j - t_i > \Delta t$, the receiver will reject the message in which Δt is a conjointly agreed to transmission delay. Therefore, this scheme protects against a replay attack.

Theorem 6: (Traceability) TRA is able to track the real identity from the pseudonym of the vehicle.

Definition 1: It is possible to encrypt the string of text by employing the XOR operation $(\oplus)$ to every character utilizing a given key. For decryption the output, the cipher will be removed only by reapplying the XOR function with the key as:

$$If \quad \mathbb{X} \oplus \mathbb{Y} = \mathbb{Z} \quad then \quad \mathbb{X} \oplus \mathbb{Z} = \mathbb{Y}$$

Proof: In case the TRA should trace the vehicle's real identity, it can get a real identity by the equivalent pseudo-identity. Considering a pseudo-identity $PID_i = \{PID_{i,1}, PID_{i,2}, VP_i\}$ in a signed message and $PID_{i,2} =$

$RID_i \oplus h\left(r_i.P_{pub}, VP_i\right)$, the TRA is able to trace the vehicle's real identity using definition 1:

$$RID_i = PID_{i,2} \oplus h\left(r_i.P_{pub}, VP_i\right)$$

Consequently, when a signature is in dispute, the TRA assigning the pseudo identities to the vehicles' real identity can trace the vehicle from the disputed message.

Vehicle

Type ID	Message ID	Payload	Timestamp	Signature	Pseudo ID
2 Bytes	2 Bytes	100 Bytes	4 Bytes	20 Bytes	64 Bytes

Fog Node

Type ID	Message ID	Payload	Timestamp	Signature	Real ID
2 Bytes	2 Bytes	100 Bytes	4 Bytes	20 Bytes	10 Bytes

Figure 4: Format of signed message for vehicle and fog node.

6. Performance Evaluation

Here, a comparison is made between our scheme and related works CPAS [13], PPAS [14], CL-CPPA [17], and EMAS [18] in terms of both communication and computation cost. It is worth noting that, the first two models are based on the bilinear pairing method, whereas CL-CPPA, EMAS, and our scheme are established on elliptic curve cryptography.

We used simulations in NS-3 to assess the performance. The simulation area is 5 km $\times$ 5 km and the highest node density on the simulation area is 500 nodes. We consider 5 RSUs and 15 fog nodes along the roadside for serving the vehicle nodes. RSUs and fog nodes are mounted at appropriate distances to provide sufficient coverage to take advantage of a fog computing-based VANET. Each RUS can serve 500 demands at the same time. In order to model the wireless channel, the two-ray ground reflection model is utilized as the radio propagation model. IEEE 802.11p is utilized in the MAC-layer. Moreover, the vehicles' transmission range is adjusted at 300 m. The channel bandwidth utilized in our simulation is 6 Mbps. The total simulation time is 360 seconds in each simulation run. The setting time is set to 30 seconds at the start of simulation for removing the impact of transient performance over the results. The overall simulation time also involved 30 seconds of stop sending packets from the simulation end. For simplicity, we assume that both vehicles and fog nodes have the same equipment and the experiment is executed in a machine equipped with a 3.4GHZ i7-2600 CPU.

6.1. Communication Overhead

Communication overhead is a key element in assessing the scheme's performance. To verify a message sender and ensure the message integrity, vehicles or fog nodes need to sign messages before sending it. For analyzing the communication overhead of the presented system, we follow the safety messages format between vehicles and fog nodes as in [13] (see Figure 4). In this format, the signature is considered as cryptography overhead. Obviously, to reduce communication costs, it needs to decrease the size of the signature. According to [13], to decrease the signature length, it is appropriate to utilize a 159-bit subgroup of the MNT curve with an embedding degree of 6.

In our scheme, the overall packet size can be decreased by 192 bytes where the signature is 20 bytes and 64 bytes is for pseudo-identity.

According to [28], the size of the element in group G such $\{PID \in G\}$, timestamp $\{VP\}$, the output of the hash function such as $\left\{\tau \in Z_q^*\right\}$, and real identity $\{RID\}$ are respectively 40 bytes, 4 bytes, 20 bytes, and 10 bytes. So, given $\{PID_V, M_V, \tau_V\}$ the total signature size of our scheme excluding message size and pseudo-ID is 20 bytes where the total pseudo identity's size $\{PID_{V,1}, PID_{V,2}, VP_V\}$ is 64 bytes. Additionally, our scheme uses a real identity, instead of pseudo-identity, for sending message from RSU/FN to vehicle. Therefore, due to the size of the message, Type-ID, Message-ID, signature, and pseudo-identity, the total packet size from vehicle to RSU/FN in our scheme is 192 bytes and it is 138 bytes for fog node to vehicle.

Due to the size of each element, the signature size for CPAS is 20+20+20 = 60 bytes. And, it is 20+20 = 40 bytes for PPAS, 40+20 = 60 bytes for CL-CPPA, whereas the signature's size of EMAS is 20 bytes. The pseudo-ID size of CPAS, PPAS, and our scheme is 40+20+4 = 64 bytes. It is 40+40 = 80 bytes for CL-CPPA, whereas, size of pseudo-ID of EMAS is 40+20 = 60 bytes.

The communication cost of our scheme and other comparable schemes are illustrated in Table 2. As illustrated in this table, EMAS has the lowest cost of communication. This is because the our scheme and other comparable schemes use the timestamp as an element in pseudo-ID generation that it increases the communication cost by 4 bytes. In EMAS and CL-CPPA, each vehicle uses only one pseudo-ID when communicating with other entities during movement; whereas our scheme, CPAS, and PPAS change the pseudo-ID of the vehicle nodes over a period of time. According to [9], the vehicle movement trajectory can be traced by an adversary, if a vehicle uses one pseudo-ID during all communication. Building on this, EMAS and CL-CPPA cannot meet privacy-preserving requirements.

In this study, to reflect our scheme performance efficiency, we also utilized the transmission delay for quantifying the communication overhead. We compare the transmission delay of our scheme and compa-

Table 2
Comparison of Communication Cost

Model	Type ID	Msg. ID	Payload	Timestamp	Signature	Pseudo-ID	Total
CPAS	2	2	100	4	60	64	232 Bytes
PPAS	2	2	100	4	40	64	212 Bytes
CL-CPPA	2	2	100	4	60	80	248 Bytes
EMAS	2	2	100	4	20	60	188 Bytes
Our Scheme	2	2	100	4	20	64	192 Bytes

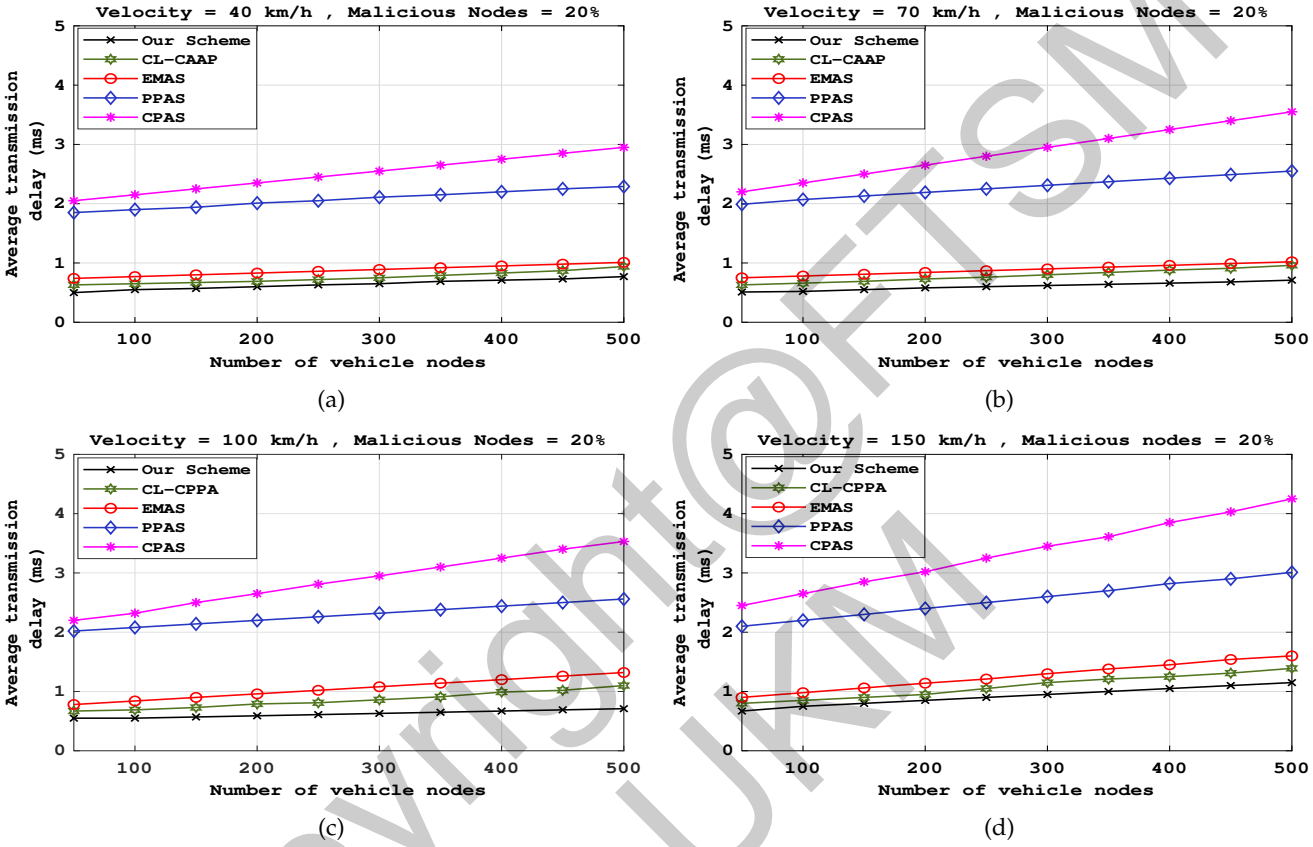

Figure 5: Average transmission delay on different number of vehicles (a) velocity = 40 km/h, malicious node = 20% (b) velocity = 70 km/h, malicious node = 20% (c) velocity = 100 km/h, malicious node = 20% (d) velocity = 150 km/h, malicious node = 20%.

rable models with different speeds (40 km/h, 70 km/h, 100 km/h and 150 km/h) in different density of vehicle nodes (50, 100, 200, 300, 400, and 500 nodes) when 20% of participated vehicles in the network are malicious node who generate invalid signatures (see Figure 5).

We acknowledge that the average transmission delay increases with the increasing of vehicle node. Also, velocity influences the transmission delay. It is obvious that the transmission delay increases with the number of malicious nodes increasing. To prove this, we measured transmission delay when 50% of vehicle nodes in the network are malicious nodes. We observed that when the speed of vehicle nodes is 100 (km/h), with the number of malicious nodes increasing from 20% to 50% in the network, the transmission delay of our scheme, CPAS, PPAS, CL-CPPA and EMAS respectively increase nearly 25%, 42%, 37%, 36% and 38%.

6.2. Computation Overhead

Here, we compare our scheme, CPAS, PPAS, CL-CPPA, and EMAS in terms of computation overhead. To this end, by inspiring the computation evaluation method for VANET in [29], we construct an additive group generated by a point P on a non-singular elliptic curve $E : y^2 = x^3 + ax + b \ (mod \quad p)$, and its order is q, where $a, b \in Z_p^*$, and p,q are two 160-bit prime numbers.

Regarding convenience, we get the cryptographic implementation time by using the MIRACL library [30]. Some notations for execution time are explained as fol-

low:

1) T_{bp}: A bilinear pairing operation's execution time $\bar{e}(P, Q)$, where $\bar{P}$,$\bar{Q} \in G_1$ and $T_{bp} \cong 4.2110(ms)$.
2) $T_{bp.m}$: A scalar multiplication operation's execution time $x.\bar{P}$ associated with the bilinear pairing, in which $\bar{P} \in G_1$ and $x \in Z_q^*$ and $T_{bp.m} \cong 1.7090(ms)$.
3) $T_{bp.sm}$: The execution time of a small scalar multiplication operation $v_i.\bar{P}$ associated with the bilinear pairing utilized in the small exponent test, wherein, $P \in G_1$, $v_i \in [1, 2^t]$ is a small random integer, t is a small integer and $T_{bp.sm} \cong 0.0535(ms)$.
4) $T_{bp.a}$: A point addition operation's execution time $P + Q$ associated with the bilinear pairing, where $P, Q \in G_1$ and $T_{bp.a} \cong 0.0071(ms)$.
5) T_{mtp}: The execution time of a Map-To-Point hash operation associated with the bilinear pairing; $T_{mtp} \cong 4.4060(ms)$.
6) $T_{e.m}$: A scale multiplication operation's execution time $x.P$ associated with the ECC, where $P \in G$ and $x \in Z_q^*$ and $T_{e.m} \cong 0.4420(ms)$.
7) $T_{e.sm}$: The execution time of a small scalar multiplication operation $v_i.P$ utilized in the small exponent test technology, in which, $P \in G$,$v_i \in [1, 2^t]$ is a small random integer, t is a small integer and $T_{e.sm} \cong 0.0138(ms)$.
8) $T_{e.a}$: A point addition operation's execution time $P + Q$ associated with the ECC, where $P, Q \in G$ and $T_{e.a} \cong 0.0018(ms)$.
9) T_h: The execution time of a One-way hash function operation. $T_h \cong 0.0001(ms)$.

Here, we calculate the computation time of pseudo-identity generation, single message verification, message signing, and batch message authentication for our scheme and related works, separately.

[To pseudo-identity generation]: our scheme comprises of two scalar multiplication processes, and only one one-way hash function operation. Therefore, the whole procedure's overall computation time is $2T_{e.m} + T_h \cong 2 * 0.4420 + 0.0001 = 0.8841(ms)$. For PPAS, it includes one one-way hash function operation and two scalar multiplication processes. Therefore, the whole procedure's overall calculation time is $2T_{e.m} + T_h \cong 2 * 0.4420 + 0.0001 = 0.8841(ms)$. For CPAS, this includes one one-way hash function operation and three scalar multiplication processes. Thus, the total computation time of the whole procedure is $3T_{e.m} + T_h \cong 3 * 0.4420 + 0.0001 = 1.3261(ms)$. CL-CPPA consists one scalar multiplication processes and two point addition operations. So, the overall calculation time is $(T_{e.m} + 2T_{e.a}) \times z \cong 0.4420 + 2 * 0.0018 = 0.4456(ms)$. And, for EMAS, pseudo-identity generation includes only one one-way hash function operation and two scalar multiplication processes. Therefore, the whole procedure's overall calculation time is $2T_{e.m} + T_h \cong 2 * 0.4420 + 0.0001 = 0.8841(ms)$.

[To message signing]: to do this, our scheme includes one one-way hash function operation, and two scalar multiplication processes. Hence, the overall calculation time of the entire procedure is $2T_{e.m} + T_h \cong 2 * 0.4420 + 0.0001 = 0.8841(ms)$. Whereas, PPAS includes tree scalar multiplication processes, one map-to-point hash function, and two one-way hash function processes. Therefore, the overall computation time of the entire procedure is $3T_{e.m} + T_{mtp} + 2T_h \cong 3 * 0.4420 + 4.4060 + 2 * 0.0001 = 5.7302(ms)$. CPAS signs a message with five scalar multiplication processes, one one-way hash function operation, and one map-to-point hash function. Consequently, the whole procedure's overall calculation time is $5T_{e.m} + T_{mtp} + T_h \cong 5 * 0.4420 + 4.4060 + 0.0001 = 6.7161(ms)$. For CL-CPPA, it includes three scalar multiplication processes, two point addition operations and only one one-way hash function processes. So, the overall computation time of the entire procedure for message signing is $3T_{e.m} + 2T_{e.a} + 1T_h \cong 3 * 0.4420 + 2 * 0.0018 + 1 * 0.0001 = 1.3297(ms)$. And, EMAS includes four scalar multiplication processes and two one-way hash function processes. Therefore, the overall computation time of the entire procedure is $4T_{e.m} + 2T_h \cong 4 * 0.4420 + 2 * 0.0001 = 1.7682(ms)$.

[To single message verification]: our scheme involves only one one-way hash function operation, and three scalar multiplication processes. Hence, the entire procedure's overall computation time is $3T_{e.m} + T_h \cong 3 * 0.4420 + 0.0001 = 1.3261(ms)$. PPAS comprises two bilinear pairing processes, three one-way hash function operation, one map-to-point hash function operation, and three scalar multiplication processes. Therefore, the entire procedure's overall calculation time is $2T_{bp} + 3T_h + T_{mtp} + 3T_{e.m} \cong 2 * 4.2110 + 3 * 0.0001 + 4.4060 + 3 * 0.4420 = 14.1543(ms)$. CPAS comprises two bilinear pairing processes, one map-to-point hash function operation, three scalar multiplication processes, and only one one-way hash function operation. Thus, the overall calculation time of the entire procedure is $2T_{bp} + T_h + T_{mtp} + 3T_{e.m} \cong 2 * 4.2110 + 0.0001 + 4.4060 + 3 * 0.4420 = 14.1541(ms)$. CL-CPPA includes tree scalar multiplication processes, three point addition operations and one one-way hash function operation. So, the entire procedure's overall calculation time is $3T_{e.m} + 3T_{e.a} + T_h \cong 3 * 0.4420 + 3 * 0.0018 + 0.0001 = 1.3315(ms)$. And, EMAS comprises two one-way hash function operation and five scalar multiplication processes. Therefore, the entire procedure's overall calculation time is $5T_{e.m} + 2T_h \cong 5 * 0.4420 + 2 * 0.0001 = 2.2102(ms)$.

[To batch message verification]: our scheme is made up of includes $(n + 1)$ scalar multiplication processes, and (n) one-way hash function processes. Therefore,

Table 3
Comparison of Computation Cost

Model	Pseudo-id Generation	Message Signing	Single Verification	Batch Verification
CPAS	1.3261	6.7162	14.1541	$9.2541n + 8.8640$
PPAS	0.8841	5.7302	14.1543	$9.2542n + 8.8641$
CL-CPPA	0.4456	1.3297	1.33150	$1.3315n + 0.8840$
EMAS	0.8841	1.7682	2.21020	$1.7682n + 0.4420$
Our Scheme	0.8841	0.8841	1.32610	$0.4421n + 0.4420$

the overall calculation time of the entire procedure is $(n+1)T_{e.m} + nT_h \cong (n+1) * 0.4420 + n * 0.0001 = 0.4421n + 0.4420(ms)$. PPAS contains comprises two bilinear pairing processes, $(n+1)$ scalar multiplication processes, $(2n)$ map-to-point hash function process, and $(2n+1)$ one-way hash function processes. Accordingly, the entire procedure's overall calculation time is $2T_{bp} + (2n+1)T_h + 2nT_{mtp} + (n+1)T_{e.m} \cong 2 * 4.2110 + (2n+1) * 0.0001 + 2n * 4.4060 + (n+1) * 0.4420 = 9.2542n + 8.8641(ms)$. CPAS includes two bilinear pairing processes, $(n+1)$ scalar multiplication processes, $(2n)$ map-to-point hash function processes, and (n) one-way hash function processes. Therefore, the overall calculation time of the entire procedure is $2T_{bp} + nT_h + 2nT_{mtp} + (n+1)T_{e.m} \cong 2 * 4.2110 + n * 0.0001 + 2n * 4.4060 + (n+1) * 0.4420 = 9.2541n + 8.8640(ms)$. CL-CPPA comprises $(3n+2)$ scalar multiplication processes, $(3n)$ point addition operations and (n) one-way hash function processes. So, the entire procedure's overall calculation time is $(3n+2)T_{e.m} + (3n)T_{e.a} + (n)T_h + \cong (3n+2) * 0.4420 + (3n) * 0.0018 + (n) * 0.0001 = 1.3315n + 0.8840(ms)$. EMAS contains $(4n+1)$ scalar multiplication processes and $(2n)$ one-way hash function processes. Accordingly, the entire procedure's overall calculation time is $(2n)T_h + (4n+1)T_{e.m} \cong (2n) * 0.0001 + (4n+1) * 0.4420 = 1.7682n + 0.4420(ms)$.

A comparison of these schemes in terms of pseudo-identity generation, single message authentication, message signing, and batch authentication as shown in Table 3. The batch verification cost of our scheme, CPAS, PPAS, CL-CPPA and EMAS for 100 messages respectively is 44.6520, 934.2740, 934.2841, 134.0340 and 177.2620 milliseconds. It indicates that batch verification in our scheme has an improvement higher than CPAS, PPAS, CL-CPPA, and EMAS. In this phase, the percentage improvement of the total operation time of the proposed scheme is approximately $\frac{934.2740-44.6520}{934.2740} \times 100 \cong 95.22\tilde{}$, $\frac{934.2841-44.6520}{934.2841} \times 100 \cong 95.22\tilde{}$, $\frac{134.0340-44.6520}{134.0340} \times 100 \cong 66.68\tilde{}$, and $\frac{177.2620-44.6520}{177.2620} \times 100 \cong 74.81\tilde{}$.

Moreover, we show the impact of fog node in the designed framework for proposed scheme. Since delay is an important issue in VANET, we evaluate and compare our scheme in terms of network delay in four different scenarios: (i) our scheme with only RSU; (ii) our scheme with only Cloud; (ii) our scheme with cloud-edge; (iv) our scheme with cloud-RSU-fog. Figure 6 shows the average network delay is high when we use only RSU or cloud. In contrast, the average network delay when we use of edge or fog is low. For the last two scenarios, the network delay is almost the same until the number of vehicles is below 400, but, with increasing the density, it is more when the cloud-edge is utilized for the proposed scheme. It is mainly because the lower processing and storage capabilities in edge nodes than fog nodes. Hence, edge node needs to send the data to cloud for more processing.

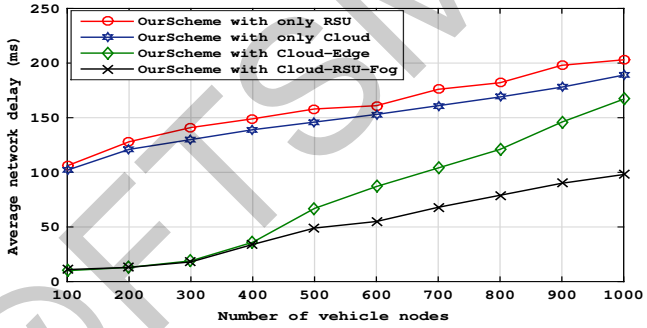

Figure 6: Comparison of network delay of proposed scheme in four different scenarios.

6.3. Batch Message Verification Analysis

As mentioned earlier, in the batch message verification, when there is at least one invalid message in the batch, it needs to find the invalid message(s). To this end, we proposed a recursive algorithm based on the binary search. In this algorithm, the desired batch will be broken into two separate batches, first. This segmentation will be continued until finding all invalid message(s). Figure 7 shows the segmentation by this algorithm when the desired batch contained 10 event messages.

As shown in this figure, the batch message verification will be performed on the initial batch contained 10 messages, first. If Equation (3) is not established, there is at least one invalid message in the batch. Therefore, the initial batch divides into two separate batches contained 5 messages. The batch message verification performs on these two batches, separately. If each batch holds Equation (3), it means all messages exist in the batch are valid and hence the

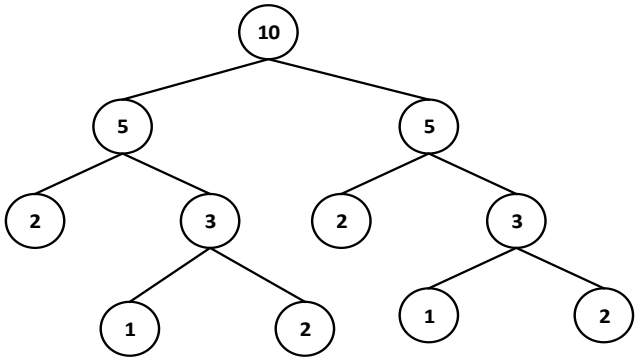

Figure 7: Segmentation of the batch contained 10 messages by the proposed algorithm.

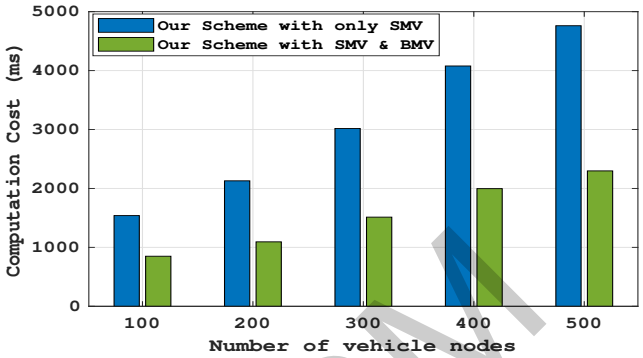

Figure 8: Comparison of computation cost of proposed scheme with SMV and with SMV&BMV.

algorithm will be stopped for this batch. Otherwise, the batch segmentation will be continued until each batch contained two or one messages. In this situation, the proposed single message verification will be performed to check the validity of the message.

For this example, in the worst case when all existing messages in the batch are invalid, we use one BMV(10), two BMV(5), two BMV(3), and ten SMV where BMV(x) is a batch message verification on a batch contained x messages and SMV is a single message verification. Based on Table 3, the computation cost of batch message verification and in addition finding invalid messages is $BMV(10)+2*BMV(5)+2*BMV(3)+10*SMV = (0.4421*10+0.4420)+2*(0.4421*5+0.4420)+2*(0.4421*3+0.4420)+10*1.32610 = 26.9656(ms)$ and the total overhead cost for only finding invalid messages is $26.9656 - BMV(10) = 26.9656 - (0.4421*10+0.4420) = 22.1026(ms)$. Whereas, the total cost of computation for 10 messages using the single message verification is $10*1.32610 = 13.2610(ms)$.

The mathematical proof shows that it is better to use the single message verification instead of batch message verification in the proposed scheme, but we experimentally found that the proposed scheme with both batch and single message verification is much better than the scheme with only single verification. To this purpose, we have separately simulated the proposed scheme with only SMV and with SMV & BMV under the different density when 20% of participated vehicles in the network are malicious nodes. The comparison of obtained computation cost shows that the performance of proposed scheme with SMV & BMV about 49% is better than the scheme with only SMV (see Figure 8).

6.4. Quotient Filter Analysis

The probabilistic data structure is extremely useful for big data generated in VANET. It usually uses to enhance lookup performance and reduce memory consumption. Here, we analyze the quotient filter utilized in the proposed node authentication scheme. To evaluate efficiency of the proposed QF-based scheme, it has been compared to an approach based on bloom filter (BF), counting bloom filter (CBF), hash table (HT), and B+ tree (B+).

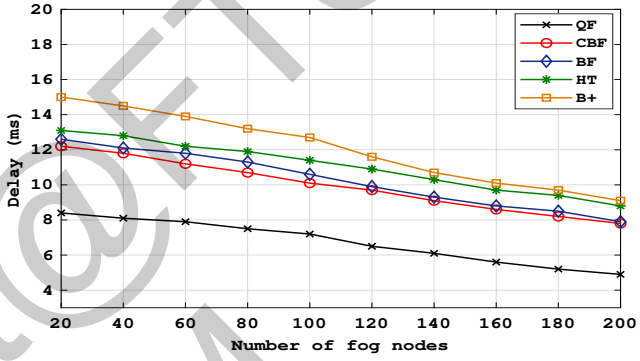

Figure 9: Comparative delay evaluation of QF with BF, CBF, HT, and B+.

The obtained results in the proposed fog-enabled VANET are shown in Figure 9. In this figure, the delay has been reflected in comparison among the above approaches. It is clear that the proposed scheme has comparatively less delay relative to other schemes. An average of overall improvement of 32.51, 34.70, 39.40, and 44.18 percent has been observed in this figure.

In terms of execution time and throughput, QF has better performance than BF. According to [31], using a quotient filter, 0.3 sec is needed to extract 10000 packets from a standard database and load into memory, whereas it takes 0.6 sec using BF, where the size of each packet is 1166 bytes. Building on this, the throughput of QF is about 310 Mbits/sec and it is 155 Mbits/sec for BF.

7. Conclusion

In this paper, we proposed a security and privacy scheme based on node and message authentication. In this scheme, to reduce latency and enhance security, fog nodes are distributed to the edge of vehicular network while the RSUs host the existing fog nodes. Due

to the large number of vehicle nodes as well as the big data generated in the VANET, quotient filter is used to keep the information of authorized and unauthorized vehicles. The proposed QF-based node authentication scheme ensures the legitimacy of nodes entered the network. In fact, the authenticity of the vehicle node is checked Before initiating data sharing. Additionally, the message authentication technique proposed based on elliptic curve-cryptography guarantees the integrity of the event message by signing the messages and verifying the signatures. To meet privacy-preserving, we used the pseudonym for vehicle nodes. As shown in the security analysis, our scheme meets the security requirement of VANET appropriately and is suitable to be in in real VANET scenarios. Furthermore, the performance analysis shows that the proposed scheme outperforms the related works.

References

[1] Akyildiz, I. F., Pierobon, M., Balasubramaniam, S., and Koucheryavy, Y. "The internet of bio-nano things," IEEE Communications Magazine, 53(3), 32-40, 2015.

[2] Ye, L., Kong, L., Ghafoor, K. Z., Chen, G., and Mumtaz, S. "LAB: Lightweight adaptive broadcast control in DSRC vehicular networks," Wireless Communications and Mobile Computing, 2018.

[3] Yi, S., Li, C., and Li, Q. "A survey of fog computing: concepts, applications, and issues," In Proceedings of the 2015 workshop on mobile big data (pp. 37-42), 2015.

[4] Lyu, L., Jin, J., Rajasegarar, S., He, X., and Palaniswami, M. "Fog-empowered anomaly detection in IoT using hyperellipsoidal clustering," IEEE Internet of Things Journal, 4(5), 1174-1184, 2017.

[5] Jiang, S., Zhu, X., and Wang, L. "An efficient anonymous batch authentication scheme based on HMAC for VANETs," IEEE Transactions on Intelligent Transportation Systems, 17(8), 2193-2204, 2016.

[6] Engoulou, R. G., Bellaïche, M., Pierre, S., and Quintero, A. "VANET security surveys," Computer Communications, 44, 1-13, 2014.

[7] Cheng, N., Lyu, F., Chen, J., Xu, W., Zhou, H., Zhang, S., and Shen, X. S. "Big data driven vehicular networks," IEEE Network, 32(6), 160-167, 2018.

[8] Raya, M., & Hubaux, J. P. "Securing vehicular ad hoc networks," Journal of computer security, 15(1), 39-68, 2007.

[9] Zhang, C., Lin, X., Lu, R., Ho, P. H., and Shen, X. "An efficient message authentication scheme for vehicular communications," IEEE transactions on vehicular technology, 57(6), 3357-3368, 2008.

[10] Chim, T. W., Yiu, S. M., Hui, L. C., and Li, V. O. "SPECS: Secure and privacy enhancing communications schemes for VANETs," Ad Hoc Networks, 9(2), 189-203, 2011.

[11] Liu, Y., Wang, L., and Chen, H. H. "Message authentication using proxy vehicles in vehicular ad hoc networks," IEEE Transactions on Vehicular Technology, 64(8), 3697-3710, 2014.

[12] Zhang, C., Lu, R., Lin, X., Ho, P. H., and Shen, X. "An efficient identity-based batch verification scheme for vehicular sensor networks," In IEEE INFOCOM 2008-The 27th Conference on Computer Communications, pp. 246-250, 2008.

[13] Shim, K. A. "CPAS: An Efficient Conditional Privacy-Preserving Authentication Scheme for Vehicular Sensor Networks," IEEE Transactions on Vehicular Technology, 61(4), 1874-1883, 2012.

[14] Zhong, H., Han, S., Cui, J., Zhang, J., & Xu, Y. "Privacy-preserving authentication scheme with full aggregation in VANET," Information Sciences, 476, 211-221, 2019.

[15] He, D., Zeadally, S., Xu, B., and Huang, X. "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.

[16] Li, JiLiang, Kim-Kwang Raymond Choo, WeiGuo Zhang, Saru Kumari, Joel JPC Rodrigues, Muhammad Khurram Khan, and Dieter Hogrefe. "EPA-CPPA: An efficient, provably-secure and anonymous conditional privacy-preserving authentication scheme for vehicular ad hoc networks." Vehicular Communications 13, 104-113, 2018.

[17] Li, Jiliang, Yusheng Ji, Kim-Kwang Raymond Choo, and Dieter Hogrefe. "CL-CPPA: certificate-less conditional privacy-preserving authentication protocol for the Internet of Vehicles." IEEE Internet of Things Journal 6, no. 6, 10332-10343, 2019.

[18] Cui, J., Wei, L., Zhang, J., Xu, Y., and Zhong, H. "An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks," IEEE Transactions on Intelligent Transportation Systems, 20(5), 1621-1632, 2018.

[19] Huang, C., Lu, R., Choo, KK. "Vehicular fog computing: architecture, use case, and security and forensic challenges," IEEE Communications Magazine, 55(11):105-11, 2017.

[20] Goudarzi, S., Anisi, MH., Abdullah, AH., Lloret, J., Soleymani, S.A., Hassan, WH. "A hybrid intelligent model for network selection in the industrial Internet of Things," Applied Soft Computing, 1,74:529-46, 2019.

[21] Koblitz, N. "Elliptic curve cryptosystems," Mathematics of computation, 48(177), 203-209, 1978.

[22] Khattak, H. A., Islam, S. U., Din, I. U., and Guizani, M. "Integrating fog computing with VANETs: A consumer perspective," IEEE Communications Standards Magazine, 3(1), 19-25, 2019.

[23] Moghaddam, M. H. Y., & Leon-Garcia, A. "A fog-based internet of energy architecture for transactive energy management systems" IEEE Internet of Things Journal, 5(2), 1055-1069, 2018.

[24] D. E. Knuth. "The Art of Computer Programming: Sorting and Searching," volume 3. Addison Wesley, 1973.

[25] Bender, M. A., Farach-Colton, M., Johnson, R., Kraner, R., Kuszmaul, B. C., Medjedovic, D., and Zadok, E. "Don't Thrash: How to Cache Your Hash on Flash," PVLDB, 5(11), 1627-1637, 2012.

[26] Garg, Sahil, Amritpal Singh, Kuljeet Kaur, Gagangeet Singh Aujla, Shalini Batra, Neeraj Kumar, and Mohammad S. Obaidat. "Edge computing-based security framework for big data analytics in VANETs." IEEE Network 33, no. 2,72-81, 2019.

[27] Pointcheval, D., and Stern, J. "Security arguments for digital signatures and blind signatures," Journal of cryptology, 13(3), 361-396, 2000.

[28] Xie, Y., Wu, L., Shen, J., & Alelaiwi, A. "EIAS-CP: new efficient identity-based authentication scheme with conditional privacy-preserving for VANETs," Telecommunication Systems, 65(2), 229-240, 2017.

[29] Cui, J., Zhang, J., Zhong, H., & Xu, Y. "SPACF: A secure privacy-preserving authentication scheme for VANET with cuckoo filter," IEEE Transactions on Vehicular Technology, 66(11), 10283-10295, 2017.

[30] Shamus Software Ltd. MIRACL Library. Accessed: May 1, 2015. [Online]. Available: http://www.shamus.ie/index.php?page=home

[31] Al-Hisnawi, M., and Ahmadi, M. "Deep packet inspection using quotient filter," IEEE Communications Letters, 20(11), 2217-2220, 2016.

Shidrokh Goudarzi received her Ph.D. degree in communication system and wireless network from Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia (UTM). In 2014, She received three-year full scholarship to study Ph.D. at (UTM).Then, She joined the Department of Advanced Informatics School at Universiti Teknologi Malaysia as a Postdoctoral Fellow from 2018 to 2019. Currently, she is a senior lecturer at Universiti Kebangsaan Malaysia (UKM). She also serves as reviewer for Canadian Journal of Electrical and Computer Engineering, KSII Transactions on Internet and Information Systems Journal,Journal of Engineering and Technological Sciences, Mathematical Problems in Engineering and IEEE Access. Her research interests are in wireless networks, artificial intelligence, machine learning, next generation networks, Internet of Things (Iot) and Mobile/distributed/Cloud Computing.

Mohammad Hossein Anisi is an Assistant Professor at the School of Computer Science and Electronic Engineering, University of Essex and head of Internet of Everything Laboratory. Prior to that, he worked as a Senior Research Associate at University of East Anglia, UK and Senior Lecturer at University of Malaya, Malaysia where he received 'Excellent Service Award' for his achievements. His research has focused specifically on real world application domains such as energy management, transportation, healthcare and other potential life domains. As a computer scientist, he has designed and developed novel architectures and routing protocols for Internet of Things (IoT) enabling technologies including wireless sensor and actuator networks, vehicular networks, heterogeneous networks, body area networks and his research results have directly contributed to the technology industry. He has strong collaboration with industry and been working with several companies in the UK with the focus on monitoring and automation systems based on IoT concept capable of reliable and seamless generation, transmission, processing and demonstration of data. He has published more than 90 articles in high quality journals and several conference papers and won two medals for his innovations from PECIPTA 2015 and IIDEX 2016 expositions. He has received several International and national funding awards for his fundamental and practical research as PI and Co-I. Dr Anisi is an associate editor of a number of journals including 'IEEE Access', 'Ad Hoc and Sensor Wireless Networks', 'IET Wireless Sensor Systems', 'International Journal of Distributed Sensor Networks'. 'KSII Transactions on Internet and Information Systems journals' and 'Journal of Sensor and Actuator Networks'. He has been guest editor of special issues of the journals and Lead organizer of special sessions and workshops at IEEE conferences such as IEEE CAMAD, IEEE PIMRC, IEEE VTC and etc. He has been also serving as executive/technical committee member of several conferences. Hossein is Fellow of Higher Education Academy and Senior Member of IEEE. He is also a technical committee member of Finnish-Russian University Cooperation in Telecommunications (FRUCT), Senior Member of Institute of Research Engineers and Doctors (the IRED), Member of ACM, IEEE Council on RFID, IEEE Sensors Council, IEEE Systems Council and International Association of Engineers (IAENG).

Seyed Ahmad Soleymani received his Ph.D. degree in computer science from faculty of engineering, Universiti Teknologi Malaysia (UTM). He received his M.S degree from the Department of Computer Engineering, Islamic Azad University, Iran and B.S. degree from the Department of Computer Engineering, Sadjad University, Iran. His research interests are in the area of Wireless Sensor Network (WSN), Mobile Ad Hoc Network (MANET), Vehicular Ad Hoc Network (VANET), Internet of Things and Nano Things (IoT and IoNT), Visible Light Communication (VLS), Intelligent Algorithms (IAs), Big Data and Machine Learning.

Muhammad Khurram Khan is currently working as a Professor of Cybersecurity at the Center of Excellence in Information Assurance (CoEIA), King Saud University, Kingdom of Saudi Arabia. He is one of the founding members of CoEIA and has served as Manager R&D from March 2009 until March 2012. He, along with his team, developed and successfully managed Cybersecurity research program at CoEIA, which turned the center as one of the best centers of excellence in Saudi Arabia and in the region. He is founder and CEO of the 'Global Foundation for Cyber Studies and Research' (http://www.gfcyber.org), an independent, non-profit, and non-partisan cybersecurity think-tank in Washington D.C. USA. Prof. Khurram is the Editor-in-Chief of a well-reputed International journal 'Telecommunication Systems' published by Springer-Nature with its recent impact factor of 1.707 (JCR 2019). Furthermore, he is the editor of several international journals, including, IEEE Communications Surveys & Tutorials, IEEE Communications Magazine, IEEE Internet of Things Journal, IEEE Transactions on Consumer Electronics, Journal of Network & Computer Applications (Elsevier), IEEE Access, IEEE Consumer Electronics Magazine, PLOS ONE, Electronic Commerce Research, IET Wireless Sensor Systems, Journal of Information Hiding and Multimedia Signal Processing, and International Journal of Biometrics, etc. He has also played role of the guest editor of several international journals of IEEE, Springer, Wiley, Elsevier Science, and Hindawi. Moreover, he is one of the organizing chairs of more than 5 dozen international conferences and member of technical committees of more than 10 dozen international conferences. In addition, he is an active reviewer of many international journals as well as research foundations of Switzerland, Italy, Saudi Arabia and Czech Republic. Prof. Khurram is an honorary Professor at IIIRC, Shenzhen Graduate School, China and an adjunct professor at Fujian University of Technology, China. He has secured an outstanding leadership award at IEEE international conference on Networks and Systems Security 2009, Australia. He has been included in the Marquis Who's Who in the World 2010 edition. Besides, he has received certificate of appreciation for outstanding contributions in 'Biometrics & Information Security Research' at AIT international Conference, June 2010 at Japan. He has been awarded a Gold Medal for the 'Best Invention & Innovation Award' at 10th Malaysian Technology Expo 2011, Malaysia. Moreover, in April 2013, his invention has got a Bronze Medal at '41st International Exhibition of Inventions' at Geneva, Switzerland. In addition, he was awarded best paper award from the Journal of Network & Computer Applications (Elsevier) in Dec. 2015. Prof. Khurram is the recipient of King Saud University Award for Scientific Excellence (Research Productivity) in May 2015. He is also a recipient of King Saud University Award for Scientific Excellence (Inventions, Innovations, and Technology Licensing) in May 2016.He has published more than 360 papers in the journals and conferences of international repute. In addition, he is an inventor of 10 US/PCT patents. He has edited 9 books/proceedings published by Springer-Verlag, Taylor & Francis and IEEE. He has secured several national and international competitive research grants

with an amount of over USD 3 Million in the domain of Cybersecurity. Prof. Khurram has played a leading role in developing 'BS Cybersecurity Degree Program' and 'Higher Diploma in Cybersecurity' at King Saud University. In 2019, he has played an instrumental role as a cybersecurity subject expert for a USD 6 Million series B investment in a South Korean startup 'SecuLetter', which has received a corporate valuation of USD 38 Million.His research areas of interest are Cybersecurity, digital authentication, IoT security, biometrics, multimedia security, cloud computing security, cyber policy, and technological innovation management. He is a fellow of the IET (UK), fellow of the BCS (UK), fellow of the FTRA (Korea), senior member of the IEEE (USA), senior member of the IACSIT (Singapore), member of the IEEE Consumer Electronics Society, member of the IEEE Communications Society, member of the IEEE Computers Society, member of the IEEE Technical Committee on Security & Privacy, member of the IEEE IoT Community, member of the IEEE Smart Cities Community, and member of the IEEE Cybersecurity Community. He is also the Vice Chair of IEEE Communications Society Saudi Chapter. He is a distinguished Lecturer of the IEEE. His detailed profile can be visited at http://www.professorkhurram.com

Author Photo

Shidrokh Goudarzi

Mohammad Hossein Anisi

Seyed Ahmad Soleymani

Muhammad Khurram Khan

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☒The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

None