Vehicular Communications

A Security and Privacy Scheme based on Node and Message Authentication and Trust in Fog-enabled VANET

--Manuscript Draft--

Manuscript Number:	VEHCOM-D-20-00059R2
Article Type:	Discussion
Keywords:	Authentication; Trust; Privacy; Quotient Filter; Fog Node
Corresponding Author:	Shidrokh Goudarzi MALAYSIA
First Author:	Seyed Ahmad Soleymani
Order of Authors:	Seyed Ahmad Soleymani
	Shidrokh Goudarzi, Dr.
	Mohammad Hossien Anisi, Dr.
	Mahdi Zareei, Dr.
	Abdul Hanan, Prof.
	Nazri kama, Dr.
Abstract:	Security and privacy are the most important concerns related to vehicular ad hoc network (VANET), as it is an open-access and self-organized network. The presence of 'selfish' nodes distributed in the network are taken into account as an important challenge and as a security threat in VANET. A selfish node is a legitimate vehicle node which tries to achieve the most benefit from the network by broadcasting wrong information. An efficient and proper security model can be useful to tackle advances from attackers, as well as selfish nodes. In this study, a privacy-preserving node and message authentication scheme, along with a trust model was developed. The proposed node authentication ensures the legitimacy of the vehicle nodes, whereas the message authentication was developed to ensure the message's integrity. To deal with selfish nodes, an experience-based trust model was also designed. Additionally, to fulfill the privacy-preserving aspect, the mapping of each vehicle was performed using a different pseudo-identity. In this paper, fog nodes instead of road-side units (RSUs), were distributed along the roadside. This was mainly because of the fact that fog computing reduces latency, and results in increased throughput. Security analysis indicated that our scheme met the VANETs' security requirements. In addition, the performance analysis showed that the proposed scheme had a lower communication and computation overhead, compared to the other related works. Monte-Carlo simulation results were applied to estimate the false-positive rates (FPR), which also proved the validity of the proposed security scheme
Response to Reviewers:	

LP-FTSM-2020-011

Editor-in-Chief

Vehicular Communications

November 16, 2020

Dear Dr. M. Atiquzzaman

We would like to submit the revised version of the research article entitled "**A Security and Privacy Scheme based on Node and Message Authentication and Trust in Fog-enabled VANET**" for publication in the Vehicular Communications journal.

We have revised the paper based on the reviewers' comments. This manuscript has not been published elsewhere and it is not under consideration for publication by another journal. We declare that have no conflicts of interest to disclose if you feel our manuscript is appropriate for your journal.

Thank you for your time and consideration.

Sincerely,

Shidrokh Goudarzi

Title: A Security and Privacy Scheme based on Node and Message Authentication and Trust in Fog-enabled VANET

Manuscript number: VEHCOM-D-20-00059R1

Revision Comments

We appreciate the time and efforts of the editor and reviewers in reviewing our manuscript. Their attention to the details is of great benefit to our research. We carefully considered the comments and revised the paper accordingly. Compared to the previous version, we think this submission appears more rigorous and complete. We really appreciate the Editor and reviewers' help. Here are our responses to the comments.

Reviewer #3: In this paper, an authentication scheme has been proposed. Overall, the topic itself is interesting.

1. But the paper is not well written and difficult to follow up. Poor organization.

Response: We have reorganized the paper. The new organization is as follows:

1. Introduction		
2. Related Works		
	2.1. Authentication and Privacy	
	2.2. Trust	
3. Preliminaries		
	3.1. Network Model	
	3.2. Security Requirements	
	3.3. Fog Computing	
	3.4. Probabilistic Data Structure	
	3.5. Quotient Filter	
4. Proposed Scheme		
	4.1. Initialization Phase	
	4.2. Registration Phase	
	4.3. Node Authentication Phase	
	4.4. Message Authentication Phase	
		A. V-FEN Communication
		B. V-FS Communication
		C. FEN-V Communication

		D. V-V Communication
	4.5. Trust Measurement Phase	
5. Security Analysis		
6. Performance Evaluation		
	6.1. Communication Overhead	
	6.2. Computation Overhead	
	6.3. Quotient Filter Analysis	
	6.4. Trust Model Analysis	
7. Simulation with NS-2		
	7.1. Basis of Scheme Simulation	
	7.2. Result of Simulation	
		FPR
		Transmission Delay
		Packet Loss Ratio
8. Conclusion		
Appendix A		
Appendix B		
References		

2. Authors need to compare their scheme with respect to other existing works.

Response: We compared our scheme with three related works CPAS [34], ASBV [35], and PPAS [36].

34. Shim, K. A. CPAS: An Efficient Conditional Privacy-Preserving Authentication Scheme for Vehicular Sensor Networks. IEEE Transactions on Vehicular Technology, **2012**, 61(4), 1874-1883.
35. Bayat, M., Barmshoory, M., Rahimi, M. and Aref, M.R. A secure authentication scheme for VANETs with batch verification. Wireless networks, **2015**, 21(5), pp.1733-1743.
36. Zhong, H., Han, S., Cui, J., Zhang, J., & Xu, Y. Privacy-preserving authentication scheme with full aggregation in VANET. Information Sciences, **2019**, 476, 211-221.

3. English also needs improvement.

Response: The paper has been proofread.

Reviewer #4: There are some minor concerns have in this paper.

1. Needs to be included limitation and future challenges of this work.

Response: The computation cost of our scheme is still high, as it is based on bilinear pairing, and this matter leads to performance issues. In the future, we plan to develop a message authentication scheme based on an elliptic curve cryptography, to help reduce the communication and computation costs. We also intend to incorporate a cloud-fog computing in a 5G-VANET environment. **(Page: 23)**

2. Some errors occur in this paper. Proofread this paper.

Response: The paper has been proofread.

3. Some Figs are not clear.

Response: Quality of the figures have improved.

Reviewer #5: Several of the concerns have been addressed by the authors but some key points are still not clear. Some answers are not adequate, and some are not consistent. Specifically, the novelty of the work is still questionable, given the significant amount of similar works in the past. Some of the technical issues such as the cluster formation, have still not been addressed. Given the insufficiency of the progress made in the revision of the paper.

1. Some of the technical issues such as the cluster formation, have still not been addressed

Response: In terms of cluster formation, we considered different sections for each fog edge node. The received messages from each section within a time interval will be joined to the corresponding batch. Once the fog edge node FEN received multiple messages from a group of vehicles in a time interval, it first categorized the messages into the different batches and then verify the messages using the batch message verification method. As shown in Figure 3, FEN searches in its own message list to extract the message related to each section and create the corresponding batch. **(Page: 10,11)**

2. The protocols need for VANETs should be further elaborated motivation and why such a protocol is needed should be added.

Response: The most of existing privacy-preserving authentication protocols have focused on authorized and unauthorized vehicle nodes, message authentication and privacy. However, the presence of selfish nodes is also an important challenge in VANET. A selfish node is an authorized vehicle node which tries to achieve the most benefit of the network by broadcasting the wrong information. To tackle these nodes, it is essential to identify them, first. To this end, a proper trust model can be effective. In this work, to consider mentioned security concerns, we developed a security and privacy model based on node and message authentication and trust. **(Page: 2)**

3. The organization of the statement in this paper needs to be modified to make it more logical and focused. Such as in the abstract part, the authors spent most of space to introduce the

background of the paper but did not show what's the advantage of their protocol over other related protocols.

Response: We have revised the abstract based on your helpful comment. We also reorganized the paper.

Abstract: Security and privacy are the most important concerns related to vehicular ad hoc network (VANET), as it is an open-access and self-organized network. The presence of 'selfish' nodes distributed in the network are taken into account as an important challenge and as a security threat in VANET. A selfish node is a legitimate vehicle node which tries to achieve the most benefit from the network by broadcasting wrong information. An efficient and proper security model can be useful to tackle advances from attackers, as well as selfish nodes. In this study, a privacy-preserving node and message authentication scheme, along with a trust model was developed. The proposed node authentication ensures the legitimacy of the vehicle nodes, whereas the message authentication was developed to ensure the message's integrity. To deal with selfish nodes, an experience-based trust model was also designed. Additionally, to fulfill the privacy-preserving aspect, the mapping of each vehicle was performed using a different pseudo-identity. In this paper, fog nodes instead of road-side units (RSUs), were distributed along the roadside. This was mainly because of the fact that fog computing reduces latency, and results in increased throughput. Security analysis indicated that our scheme met the VANETs' security requirements. In addition, the performance analysis showed that the proposed scheme had a lower communication and computation overhead, compared to the other related works. Monte-Carlo simulation results were applied to estimate the false-positive rates (FPR), which also proved the validity of the proposed security scheme. **(Page: 1)**

I refer you to the above table that we created for responding to reviewer #3. This table shows the new organization of the paper, clearly.

4. What are some of the hard research challenges and what are new problems to be tackled? Please Explain in detail.

Response: High mobility of vehicle nodes, rapid topology changing, and frequent disconnection are the research challenges in VANET. It needs to decrease the computation cost as much as possible. The computation cost of our scheme is high for vehicular environment as it is based on bilinear pairing. This matter leads to performance issues. We plan to design a message authentication scheme based on elliptic curve cryptography to reduce more communication and computation costs.

5. Justify the results (analytical and experimental) except architectural ideas?

Response: We analyzed and discussed on the results obtained from the simulation of proposed scheme by NS-2. We used three indexes false-positive rate (FPR), transmission delay, and packet loss ratio to evaluate our scheme and in addition comparison with other related works:

FPR: We proved the validity of our scheme using the Monte-Carlo simulation. To this end, we performed 1000 Monte-Carlo simulations for a large-scale network to estimate FPR. **(Page: 20)**

Transmission Delay: In order to show our scheme's efficiency, we utilized the transmission delay for quantifying the communication overhead. Obviously, a vehicle/FEN/FS has to sign the message before broadcasting it over the network. This process increases the size of exchanged message and in result caused transmission delay between vehicle-to-vehicle and or between vehicle-to-fog node. We compared the average transmission delay of our scheme with CPAS, ASBV, and PPAS under different density when the velocity of all vehicles is 20 km/h. We also evaluated our scheme under different density and velocity. As shown in Figure 8, the average transmission delay is respectively 1.11 ms, 1.52 ms, 1.77 ms, and 0.77 ms for CPAS, ASBV, PPAS, and our proposed scheme. From Figure 8, we found that the average transmission delay increases by incrementing the number of vehicles from 50 to 500. This is because the size of messages exchanged in the network will be increased as the number of vehicles increased. Figure 9 shows the impact of velocity on transmission delay related to our scheme under different density. As we can see, velocity has a slightly effect on the transmission delay. The obtained results were conceivable as our scheme has lowest message size and in result lowest communication overhead in comparison with PPAS, ASBV, and CPAS. **(Page: 21)**

Packet Loss Ratio: The size of signed messages and the number of messages transferred over the network have impact on packet loss ratio. Hence, the ratio of packet loss can be a useful metric to reflect efficiency of our scheme. We presented the equivalent packet loss ratio for our scheme, CPAS, and PPAS in Figure 10. As depicted in this figure, it is observed that by increasing the number of vehicles in the communication range, the transmission loss ratio increases. This is mainly because of the increasing number of messages transferred over the network as the vehicle density roses. We also examined the effect of velocity on packet loss ratio. As observed in Figure 11, the packet loss rises with the increase in the velocity of vehicles. But this effect is not significant. This is because the propagation speed of radio waves is much higher than the moving speed of the vehicles. As we can see in this figure, there are not many differences in packet loss for our scheme when velocity reaches 150 km/h from 40 km/h. To improve the ratio of packet loss, it is better to focus on communication cost and decrease size of signed messages. **(Page: 22)**

6. Are the references and related works comprehensive? The related works section appears monolithic. Is it possible to more categorize them based on some themes or criteria?

Response: Based on the security issues considered in our study (authentication, privacy, and trust), we categorized the related work section into two subsections:

2.1. Authentication and Privacy **(Page: 3)**

2.2. Trust **(Page: 4)**

In this section, the proposed solutions to two pivotal problems, namely privacy-preserving authentication scheme, and trust model, are highlighted.

2.1. Authentication and Privacy:

In order to achieve broadcast authentication in VANETs, the use of public key infrastructure (PKI) is commonly adopted, including the IEEE1609.2 [5]. A PKI uses a public and a private cryptographic key pair to secure the exchange data in the network. Hubaux and Raya [6] suggested a scheme for signature authentication oriented using PKI. In this scheme, all traffic-

related data exchanged in the VANETs should be verified before trusting the data. As pointed out in [7], based on verification of the authentication and integrity, PKI-based systems are well-known choices. However, in the PKI-based systems, vehicles need to store many pseudonym certificates, and the transmission overhead of the RSUs will increase with the number of vehicles. Also, conventional PKI cannot satisfy the requirements of VANETs, as it cannot preserve the conditional privacy of the drivers, and the verification time is too long.

To address the PKI-based scheme problems, an effective batch message signature-verifying scheme for a vehicle to infrastructure (V2I) communication was proposed in [8]. In this scheme, multiple received messages were simultaneously verified by the RSUs. As a result, the total authentication overhead was significantly reduced, and the VANETs' operational efficiency was enhanced. Moreover, since this scheme was based on the driver's identity, a certificate was not needed. This scheme improved the efficiency, however, it failed when the number of vehicles increased.

Zhang et al. [9] proposed an identity-based batch verification (IBV) scheme for VANETs. It was based on bilinear pairing for secure communication from vehicles, to RSUs. It decreased the confirmation delay of batch message signatures and was also faster compared to the PKI-based systems.

To verify multiple requests sent from various vehicles and create various session keys for various vehicles simultaneously, an anonymous batch authentication and key agreement (ABAKA) system was proposed by Huang et al. [10] for value-added services in VANETs. They suggested a discovery algorithm to cope with the invalid request problem.

A scheme is proposed in [11], in which, the RSU supports adjacent vehicles to authenticate their received messages was explored. Hence, there is no need to authenticate messages individually by vehicles. In other words, the vehicle is responsible for transferring the message to the RSUs for verification. In this scheme, RSUs have the role of the cloud system for the vehicles. In general, multiple messages are authenticated by the RSUs, utilizing the batch confirmation technology. The existing messages in a batch are valid when the batch verification process is successful.

Otherwise, if the batch verification is unsuccessful, there is at least one invalid message in the batch, hence, a binary search will be implemented to recover the invalid messages. The RSU would then adjust two bloom filters for storing the verification results, followed by identification of the validity of the vehicles-sent messages. The RSU is utilized specifically to substitute a valid message's hash value in a positive filter, and the hash value of an invalid message in a negative filter. The negative and positive filters would then be broadcasted by the RSU to adjacent vehicles in a specific frequency. Therefore, the vehicles only need to examine the two filters for verifying these messages. This process significantly reduces redundancy, and the entire system's efficiency is improved. Nevertheless, a large number of vehicles will result in the RSU's decreased computation performance, causing considerable delay.

To tackle this issue, Liu et al. [12] mentioned that the calculation load on the RSU could be mutual amongst adjacent vehicles. In this outline, proxy vehicles are elected by the system based on the calculation power. Nearby vehicles should share the work performed by the RSUs in the verification of the messages and send the verified data back to the RSUs. The RSUs will examine the accuracy of the result. Though the RSU's verification performance is significantly improved through the suggested scheme, the scheme's performance itself is not sufficient, since the basic operation includes map-to-point operation and bilinear pairing with a large overhead cost. Furthermore, in the case a batch of messages which includes invalid messages, the message signature will not be valid, and the RSU will fail to endorse it if the original signature is not valid, or if the proxy vehicle interfered with the legitimate signature.

An identity-based signature scheme, KIBS, was proposed by Shim [13] using the random oracle model under the Computational Diffie-Hellman (CDH) assumption. Based on the KIBS supported batch authentication procedure, a secure conditional privacy authentication scheme is constructed quickly on the RSUs. The main goal of the pseudonym-based batch verification is to arrange an effective batch authentication scheme. However, it cannot take into account the communication and storage overheads, and further verification delay results in invalid requests.

A message verification scheme was also presented in [14] for secure communication in the VANET. In this scheme, the redundancy of the authentication was eliminated to attempt on the same message across various vehicles. It reduced the verification overhead and delay.

2.2. Trust

Trust, as an element of security [6], has a vital role to cope with untrustworthy nodes in the vehicular network, [15]. A comprehensive and systematic review of existing trust models was proposed in our previous research, [16].

To deal with selfish vehicle nodes, a framework was proposed in [17] to model the reliability of the agents of nearby vehicles. The proposed trust model used a multi-layered trust modelling approach and takes into account the role, experience, priority and majority-based trust as main factors to evaluate trust levels.

In [18] an infrastructure-based trust model was proposed to identify malicious nodes, which disseminate false information. In this model, the trust level is based on recommendations given by other vehicles, and road-side infrastructure units (RSUs). However, since the mobility of the vehicles is considerably high, the model failed to harvest sufficient information from nearby vehicles.

To identify malicious nodes, a trust model was proposed for VANETs using a robust algorithm [19]. The proposed model followed the game theory approach for implementing the Nash equilibrium, to calculate the best strategy against the attacker and defend, through the use of a payoff matrix. It verified the information and messages to identify trusted nodes for reliable communication.

The authors in [20] extensively discussed the fact that vehicle data might become partially or fully compromised by attackers, which will then require their rights to be revoked. They proposed a data-centric trust model that computed trust in each individual piece of data. However, the model suffered from latency and data loss, since the trust model required measurement of the

trustworthiness of received event messages, and the data might be duplicated, which caused a heavy traffic density in the network.

However, to the best of our knowledge, no practical approaches have been proposed to build proper and comprehensive security and privacy schemes that deal with attackers and malicious nodes, as well as selfish nodes. Since the number of vehicles and data generated in the network on a daily basis is steadily increasing, there is still a lack of a suitable security scheme with a lower computation and latency, as well as acceptable communication cost.

LP-FTSM-2020-011

A Security and Privacy Scheme based on Node and Message Authentication and Trust in Fog-enabled VANET

Seyed Ahmad Soleymani [1], Shidrokh Goudarzi [2,*], Mohammad Hossein Anisi [3], Mahdi Zareei [4], Abdul Hanan Abdullah [1], and Nazri Kama [5]

[1] School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, 81310, UTM, Johor, Bahru Johor, Malaysia; asseyed4@live.utm.my, hanan@utm.my.
[2] Faculty of Information Science and Technology, Center for Artificial Intelligence Technology (CAIT), Universiti Kebangsaan Malaysia; shidrokh@ukm.edu.my.
[3] School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, U.K; m.anisi@essex.ac.uk.
[4] Escuelade Ingenieriay Ciencias, Tecnologico de Monterrey, Monterrey 64849, Mexico; m.zareei@tec.mx.
[5] Fakulti Teknologi Dan Informatik Razak, Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, Malaysia; mdnazri@utm.my.

* Correspondence: shidrokh@ukm.edu.my.

Abstract: Security and privacy are the most important concerns related to vehicular ad hoc network (VANET), as it is an open-access and self-organized network. The presence of 'selfish' nodes distributed in the network are taken into account as an important challenge and as a security threat in VANET. A selfish node is a legitimate vehicle node which tries to achieve the most benefit from the network by broadcasting wrong information. An efficient and proper security model can be useful to tackle advances from attackers, as well as selfish nodes. In this study, a privacy-preserving node and message authentication scheme, along with a trust model was developed. The proposed node authentication ensures the legitimacy of the vehicle nodes, whereas the message authentication was developed to ensure the message's integrity. To deal with selfish nodes, an experience-based trust model was also designed. Additionally, to fulfill the privacy-preserving aspect, the mapping of each vehicle was performed using a different pseudo-identity. In this paper, fog nodes instead of road-side units (RSUs), were distributed along the roadside. This was mainly because of the fact that fog computing reduces latency, and results in increased throughput. Security analysis indicated that our scheme met the VANETs' security requirements. In addition, the performance analysis showed that the proposed scheme had a lower communication and computation overhead, compared to the other related works. Monte-Carlo simulation results were applied to estimate the false-positive rates (FPR), which also proved the validity of the proposed security scheme.

Keywords: Authentication; Trust; Privacy; Quotient Filter; Fog Node; VANET.

1. Introduction

The purpose of the smart cities is to provide economic growth and enhance the life quality of the people of the land, by empowering and utilizing technologies which lead to smart outcomes. In the smart city concept, Intelligent Transportation System (ITS) is a key factor towards achieving traffic efficiency, by reducing traffic problems. VANET, as a prospective ITS technology, has developed an attention from both the industry, and research communities. VANETs are identified as an important component of the ITS, for creating an intelligent space for vehicular communications. Technologies such as cloud computing and cellular networks have helped vehicular networks and related applications to develop at much quicker rates.

Empowering VANETs with data handling abilities requires effective data processing methods capable of decreasing the computational delay, and considerably minimizing the cost of data storage, as well as transmission. Cloud-based data processing is an attractive approach, as it promotes a dynamic topology, unlimited storage with vehicular nodes, and variable network density. Although central processing and data storage is essential in some cases, nevertheless, it is unsuitable when a minor delay in data processing can result in dangerous effects.

The growth of connected nodes in vehicular environments leads to generation of a large amount of data on the edge of the network. In such a situation, the need to minimize latency becomes the core focus area. To fulfill the needs of emerging communication applications, a geographically distributed computing architecture is required. Fog computing was introduced to support various services like computation, networking between the traditional cloud systems, and end nodes, as well as data storage [1]. To deal with big data issues, fog computing also presents resources for large scale data procedure systems, without the disadvantage of cloud, or high latencies [2]. However, this technology provides some benefits such as reducing bandwidth and latency. However, security, privacy and trust are major concerns in fog-enabled VANETs. Without the guarantee of security and privacy, attackers not only can steal private information [3] but they also can easily forge the message exchanges amongst the vehicles. Attackers broadcast wrong information and may even introduce themselves as vehicles when there is a lack of proper security model for the VANETs. In addition to the attackers and malicious nodes, the presence of authorized vehicle nodes that attempt to achieve the most benefits from the network for personal use can also be an issue. This is done through the creation and dissemination of inaccurate information on the network, which is also considered as a serious security threat for VANET. These nodes are called "selfish nodes". In the previous works, some authors focused only on the message authentication aspect, and some studies considered both the messages and node authentication outcomes. However, the presence of selfish nodes in VANET is a concern.

Privacy is also a major concern in VANETs, where a vehicular message includes data on the location, speed, and direction of the vehicle. Since these messages carry a huge deal of private data regarding the driver, it is vital to maintain privacy. In general, the lack of a proper security model may result in service abuse, and malevolent attacks toward the drivers.

To cope with the security concerns related to VANET, a privacy-preserving authentication scheme along with a lightweight trust model for big data analytics was designed. In the proposed scheme, fog computing was integrated into the node and message authentication process, wherein, because of the much better processing power, the fog nodes instead of the RSUs were distributed along the roadside. In the security scheme for VANETs, employing fog nodes in the network can enhance the communication and computation abilities [4]. Moreover, because of the big data generation as well as a large number of vehicles in the network, quotient filter (QF), as a space-efficient probabilistic data structure (PDS), was extended in the proposed scheme. Before initiating any communication, the authors first needed to check the legitimacy of the node. It was based on both the authentication and trustworthiness of the vehicle node. To this end, a query was performed on the vehicle and fog node's QF. After starting the communication and data sharing, the receiver of a signed message needed to check the integrity of the message through signature verification.

The main contribution of this work are as follows:

1. We proposed a node authentication scheme based on the probabilistic data structure to deal with illegal nodes, which try to join the network. Before any data sharing and communication with other nodes in the network are to take place, the node authentication verification was required.
2. We proposed a message authentication scheme to ensure the message's integrity. This scheme was established on bilinear pairing. The message's signing and single/batch signature verification are the main attributes of the proposed scheme. We also used a pseudonym to meet privacy-preserving requirements for vehicle nodes.
3. We proposed a trust model based on experience to tackle selfish nodes. These nodes attempted to gain benefits from the network for personal use only through broadcasting wrong information to the network. In the proposed trust model, each vehicle computes the trust score of the neighbor nodes based on past direct communications. Since each vehicle needs to have a predefined minimum trust score for starting any communication, the proposed trust model can be helpful to cope with selfish nodes.
4. We simulated the proposed scheme with NS-2, and the obtained results showed that our scheme was practical with a suitable and acceptable communication efficiency score.

The remaining sections are organized as follows: In section 2, the relevant works are reviewed. Section 3 describes the background knowledge utilized in this paper. The suggested system is explained in detail

in section 4. In section 5, the security concept and analysis of the suggested outline are presented. Section 6 assesses our scheme's performance. Ultimately, the conclusion is provided in section 7.

2. Related Works

Security, privacy and trust are critical issues in vehicular networks, as VANET is an open-access, distributed, and self-organized environment. Authentication, as a primitive security requirement, is a cryptographic process that is not only used to determine that the message has not been modified during transmission, but to also utilize necessary means to determine the source of the message [5]. Privacy-preserving of authorized nodes is another aspect that needs to be considered along with security issues. Untrustworthy vehicles should also be taken into account due to security concerns. Due to these concerns in the vehicular network, many studies have been conducted to deal with these issues. In this section, the proposed solutions for two pivotal problems, namely the privacy-preserving authentication scheme, and trust model, are highlighted.

2.1. Authentication and Privacy:

In order to achieve broadcast authentication in VANETs, the use of public key infrastructure (PKI) is commonly adopted, including the IEEE1609.2 [5]. A PKI uses a public and a private cryptographic key pair to secure the exchange data in the network. Hubaux and Raya [6] suggested a scheme for signature authentication oriented using PKI. In this scheme, all traffic-related data exchanged in the VANETs should be verified before trusting the data. As pointed out in [7], based on verification of the authentication and integrity, PKI-based systems are well-known choices. However, in the PKI-based systems, vehicles need to store many pseudonym certificates, and the transmission overhead of the RSUs will increase with the number of vehicles. Also, conventional PKI cannot satisfy the requirements of VANETs, as it cannot preserve the conditional privacy of the drivers, and the verification time is too long.

To address the PKI-based scheme problems, an effective batch message signature-verifying scheme for a vehicle to infrastructure (V2I) communication was proposed in [8]. In this scheme, multiple received messages were simultaneously verified by the RSUs. As a result, the total authentication overhead was significantly reduced, and the VANETs' operational efficiency was enhanced. Moreover, since this scheme was based on the driver's identity, a certificate was not needed. This scheme improved the efficiency, however, it failed when the number of vehicles increased.

Zhang et al. [9] proposed an identity-based batch verification (IBV) scheme for VANETs. It was based on bilinear pairing for secure communication from vehicles, to RSUs. It decreased the confirmation delay of batch message signatures and was also faster compared to the PKI-based systems.

To verify multiple requests sent from various vehicles and create various session keys for various vehicles simultaneously, an anonymous batch authentication and key agreement (ABAKA) system was proposed by Huang et al. [10] for value-added services in VANETs. They suggested a discovery algorithm to cope with the invalid request problem.

A scheme is proposed in [11], in which, the RSU supports adjacent vehicles to authenticate their received messages was explored. Hence, there is no need to authenticate messages individually by vehicles. In other words, the vehicle is responsible for transferring the message to the RSUs for verification. In this scheme, RSUs have the role of the cloud system for the vehicles. In general, multiple messages are authenticated by the RSUs, utilizing the batch confirmation technology. The existing messages in a batch are valid when the batch verification process is successful. Otherwise, if the batch verification is unsuccessful, there is at least one invalid message in the batch, hence, a binary search will be implemented to recover the invalid messages. The RSU would then adjust two bloom filters for storing the verification results, followed by identification of the validity of the vehicles-sent messages. The RSU is utilized specifically to substitute a valid message's hash value in a positive filter, and the hash value of an invalid message in a negative filter. The negative and positive filters would then be broadcasted by the RSU to adjacent vehicles in a specific frequency. Therefore, the vehicles only need to examine the two filters for verifying these messages. This process significantly reduces redundancy, and the entire system's efficiency is improved. Nevertheless, a large number of vehicles will result in the RSU's decreased computation performance, causing considerable delay.

To tackle this issue, Liu et al. [12] mentioned that the calculation load on the RSU could be mutual amongst adjacent vehicles. In this outline, proxy vehicles are elected by the system based on the calculation power. Nearby vehicles should share the work performed by the RSUs in the verification of the messages and send the verified data back to the RSUs. The RSUs will examine the accuracy of the result. Though the RSU's verification performance is significantly improved through the suggested scheme, the scheme's performance itself is not sufficient, since the basic operation includes map-to-point operation and bilinear pairing with a large overhead cost. Furthermore, in the case a batch of messages which includes invalid messages, the message signature will not be valid, and the RSU will fail to endorse it if the original signature is not valid, or if the proxy vehicle interfered with the legitimate signature.
An identity-based signature scheme, KIBS, was proposed by Shim [13] using the random oracle model under the Computational Diffie-Hellman (CDH) assumption. Based on the KIBS supported batch authentication procedure, a secure conditional privacy authentication scheme is constructed quickly on the RSUs. The main goal of the pseudonym-based batch verification is to arrange an effective batch authentication scheme. However, it cannot take into account the communication and storage overheads, and the further verification delay results in invalid requests.
A message verification scheme was also presented in [14] for secure communication in the VANET. In this scheme, the redundancy of the authentication was eliminated to attempt on the same message across various vehicles. It reduced the verification overhead and delay.

2.2. Trust

Trust, as an element of security [6], has a vital role to cope with untrustworthy nodes in the vehicular network, [15]. A comprehensive and systematic review of existing trust models was proposed in our previous research, [16].
To deal with selfish vehicle nodes, a framework was proposed in [17] to model the reliability of the agents of nearby vehicles. The proposed trust model used a multi-layered trust modelling approach, and takes into account the role, experience, priority and majority-based trust as main factors to evaluate trust levels. In [18] an infrastructure-based trust model was proposed to identify malicious nodes, which disseminate false information. In this model, the trust level is based on recommendations given by other vehicles, and road-side infrastructure units (RSUs). However, since the mobility of the vehicles is considerably high, the model failed to harvest sufficient information from nearby vehicles.
To identify malicious nodes, a trust model was proposed for VANETs using a robust algorithm [19]. The proposed model followed the game theory approach for implementing the Nash equilibrium, to calculate the best strategy against the attacker and defend, through the use of a payoff matrix. It verified the information and messages to identify trusted nodes for reliable communication.
The authors in [20] extensively discussed the fact that vehicle data might become partially or fully compromised by attackers, which will then require their rights to be revoked. They proposed a data-centric trust model that computed trust in each individual piece of data. However, the model suffered from latency and data loss, since the trust model required measurement of the trustworthiness of received event messages, and the data might be duplicated, which caused a heavy traffic density in the network.
However, to the best of our knowledge, no practical approaches have been proposed to build proper and comprehensive security and privacy schemes that deals with attackers and malicious nodes, as well as selfish nodes. Since the number of vehicles and data generated in the network on a daily basis is steadily increasing, there is still a lack of a suitable security scheme with a lower computation and latency, as well as acceptable communication cost.

3. Preliminaries

3.1. Network Model

According to Figure 1, the proposed system includes the lower and upper layers. Cloud servers (CS) and root-trusted authority (TA) are included in the upper layer, whereas the lower layer consists of fog nodes and vehicles.

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

Lower Layer: This layer comprises of fog nodes and the vehicles. In this study, fog nodes, instead of RSUs, are distributed along the roadside. This is mainly because the fog nodes contain much better processing power than the RSUs to reduce latency and increase throughput. Also, the existing RSU solution is far from perfect, because it is highly dependent on a centralized architecture, and bears the cost of additional infrastructure deployment [4]. According to [3], fog nodes can act as both fog servers (FS) and fog edge nodes (FEN). Fog servers, have higher processing ability and have storage that is more powerful, are able to host various management systems, coordination, and drive required collaboration services between FENs and cloud database systems. It stores a huge amount of data for supporting local FENs. FS connects to the cloud if needed, to recover the benefits provided by the cloud. It also communicates with vehicle nodes when there is no FEN available within the communication range of vehicles. FEN interacts with vehicle nodes which are within its communication range. Briefly, FEN concentrates on local processing of outgoing and incoming vehicle node dataflow. It is believed that CS, FS, and FEN can be fully trusted, and that communication between CS and FS, as well as FS and FEN, is through secure wired communication, such as the Ethernet.

Vehicle nodes broadcast the traffic-related data periodically to enhance the operational efficiency of traffic security and regional traffic. Vehicles with a range of internal sensors are able to detect events which take place within the communication range. Each vehicle has a realistic tamper-proof device (TPD) for storing the secure substances received from TA. We also assumed that each vehicle is equipped with both the Dedicated Short-range Communication (DSRC) module and LTE. The medium used for communication between the vehicles and fog nodes was LTE, whereas communication between the vehicles was through IEEE 802.11p DSRC. In the other words, the vehicles collected traffic information and broadcast it to the local area using the IEEE 802.11p and LTE.

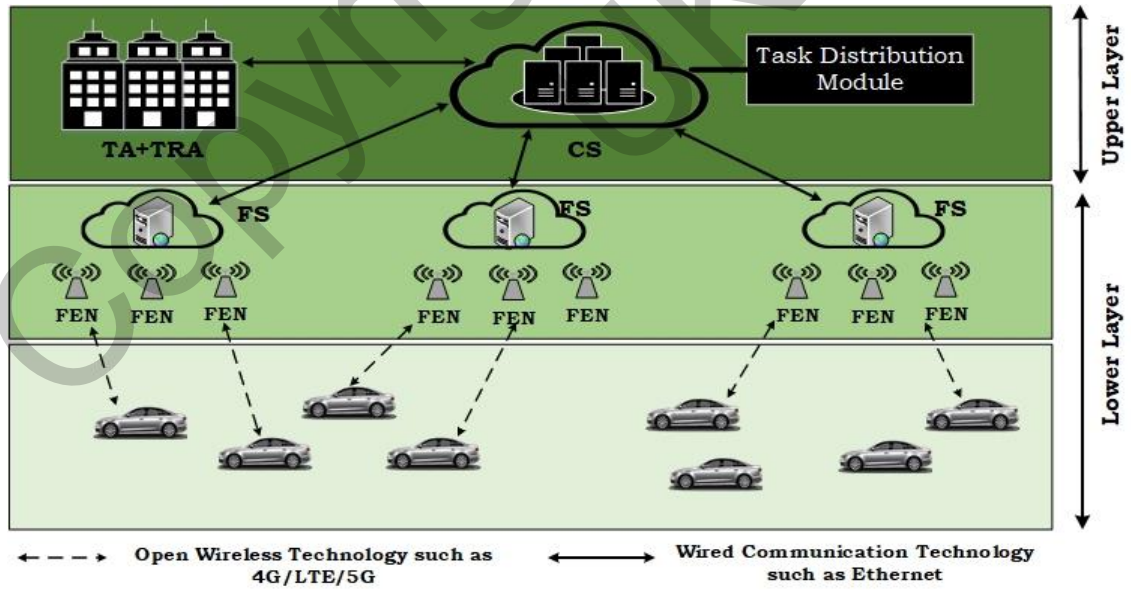

Figure 1. System architecture

Figure 2 shows a simplified view of how the FS and FEN are used in the fog layer to assist vehicles during mobility from one geographic location to another. It is believed that FS covers the whole area in the network, and the FEN's communication range covers a region of the city, which can involve several intersections [21]. When a vehicle node is 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 the 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 will be continuously supported by fog nodes. Whenever a vehicle node is under the coverage of multiple access to fog nodes, it needs to select the most suitable FS/FEN to send and receive data. To this end, the vehicle node calculates the link quality between itself, and nearby FS/FENs. According to [22], the quality of the link can be measured based on some parameters, such as bandwidth, signal to noise ratio (SNR), and bit error rate (BER), which is out of the scope of this paper.

Due to the large number of tasks created by vehicles for processing using the FSs/FENs, there is a need to monitor fog nodes in terms of computational power, memory availability, and CPU availability, as well as loading tasks. For this purpose, a module on the cloud server was developed to collect information on the distributed FSs, and then compute tasks locally, and offload them to FSs for processing. The task distribution mechanism greatly reduced the delay of the latency-sensitive applications and enhanced the overall system's scalability.

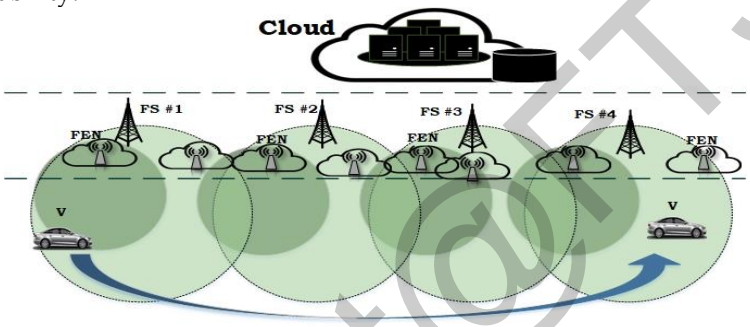

Figure 2. Vehicle node mobility in fog computing.

3.2. Security Requirements

According to [7], a well-designed privacy-preserving message and node verification outline should meet the following security goals:

1. **Resistance to Unauthorized Nodes:** An illegal and unregistered node cannot join the network and start any communication with existing nodes in the network.
2. **Node Authentication and Message Verification and Integrity:** The receiver of a message not only has to check the legitimacy of the sender's message, but also needs to assess the integrity and reliability of the incoming message.
3. **Identity Preserving Privacy:** The vehicle's real identity should endure anonymously, and no third party should be able to extract the real identity of the vehicle's pseudo-identity.
4. **Traceability:** TRA can trace the vehicle's real identity by analyzing its pseudo identity, which is extracted from its message.
5. **Resistance to Replay Attack:** A malevolent vehicle cannot gather and store a signed message, and try to send it later, in case of the original message expiring.
6. **Resistance to Selfish Node:** Selfish vehicles are considered as a serious security threat through the creation and dissemination of incorrect information in the network, which intend to derive the most benefits from the network for personal use.

3.3. Fog Computing

Cloud computing services are extended through fog computing to the edge of the network [3,14]. It is a greatly virtualized platform for providing storage, computation, and networking services, between traditional cloud servers and end tools.

Fog computing provides numerous benefits over cloud computing like load balancing, further bandwidth use, interconnectivity, minimal downtime, low latency, and improved quality of services (QoS). Combining VANETs with fog computing will provide numerous advantages, such as local data

processing, local resource pooling, cache data management, load balancing, and increased delay. In fog computing-based VANETs, the time-critical local data is analyzed through the fog edge node tools, leading to lower latency. It is worth stating that through fog computing, the interactions between vehicle nodes are facilitated, and are very effective for the collaboration of nodes [23].
In fog computing, infrastructures, or facilities, are capable of providing resources for services at the edge of the network and are termed as fog nodes [2]. Fog nodes can act as fog servers and fog edge nodes. Fog servers include more powerful processing and storage capability, whereas fog edge nodes can interact with heterogeneous tools, including various kinds of end tools requiring various protocols [3].

3.4. Probabilistic Data Structure

Due to the growth of connected vehicles with other entities in the vehicular network, this results in a generation of a large amount of data, of which, when using the traditional data structures, is not sustainable. This is due to the large memory and high latency issues for processing queries using these traditional data structures. The probabilistic data structure, as a kind of data structure, is particularly advantageous for large data, because it reduces latency, and analytical procedures [24]. They are tremendously handy data systems for reducing the space and time trade-off, and to a great extent, equivalent to retrieval and storage for querying of data [25]. They use various probability-based methods, accompanied by estimate principles, and hashing approaches. In comparison to error-free methods, less memory is used by these algorithms with constant query times. Furthermore, they usually support intersection and union processes, therefore, they can be easily parallelized. Some key probabilistic data structures comprising of Bloom filters (BF), and Quotient filters (QF) for massive dataset's membership query, count-min sketch for calculating the times for reaching the data item in the huge datasets, and hyper-log-log for cardinality approximations, have been suggested [26].

3.5. Quotient Filter

This is a cache-friendly, and space-efficient probabilistic data structure representing a multi-set of elements $S \subseteq U$ for storing a p bit fingerprint for each element. Precisely, the QF stores the multiset $F = h(S) = \{h(x) \mid x \in S\}$, where $h : U \to \{0, \ldots, 2p - 1\}$ is a hash function.

- We insert $h(x)$ into F to insert an element x into S.
- We examine whether $h(x) \in F$ to test whether an element $x \in S$.
- We eliminate $h(x)$ from F to remove an element x from S.

Theoretically, we can consider F as being stored in an open hash table T with $m = 2^q$ buckets, utilizing a method known as the quotient, which was proposed by Knuth [27]. In this method, a fingerprint, f is divided into its r least significant bits, $f_r = f \bmod 2^r$ (the remainder), and its $q = p - r$ most significant bits, $f_q = \lfloor f/2^r \rfloor$ (the quotient). For inserting the fingerprint f into F, we then store f_r in a bucket $T[f_q]$. Considering the remainder f_r in the bucket f_q, the full fingerprint can then be exclusively reconstructed as $f = f_q 2^r + f_r$ [28,29].

4. Proposed Scheme

In this study, a security and privacy scheme based on message authentication, node authentication, and trust is proposed. Both authentication and trust, as key elements of security, have a vital role to enhance safety in VANET [30]. In the proposed security scheme, message authentication ensures the integrity of the message using signature and verification, whereas node authentication ensures the legitimacy of nodes before initiating communication with other nodes. To deal with selfish nodes, a lightweight trust model based on past direct communication is also developed. To this end, the authorized vehicle nodes with a specific level of trust score or more, are able to communicate with other nodes, otherwise, it prevents data sharing. In this section, we describe our scheme in the following sections: system initialization phase, registration phase, node authentication phase, message authentication phase, and trust measurement phase.

4.1. Initialization Phase

At this phase, TA generates the necessary system parameters and preloads these parameters into TPD of vehicles and fog nodes memory. To that end, considering two prime p, q; two groups G_1 and G_2 of order q; three distinct generators P, Q and Q' in G_1; and let e be a bilinear pairing $e: G_1 \times G_1 \rightarrow G_2$.
TA randomly chooses a number $s \in Z_q^*$ as it's master private key which is at least 160 bits number. A key should be large enough that an attack is infeasible. For prime fields, a popular size is 160 bits both for the field and subgroup size [31]. Using the master private key, it also computes the corresponding public key $P_{pub} = s.P$. Then, TA chooses two secure hash functions $h_1: \{0,1\}^* \rightarrow Z_q^*$, $h_2: \{0,1\}^* \times \{0,1\}^* \rightarrow Z_q^*$, and $h_3: \{0,1\}^* \times G_1 \rightarrow Z_q^*$ which for better security, SHA-256 can be used. This is mainly because it is difficult to reconstruct the initial data from the hash value generated by SHA-256. Although SHA-512 has higher cryptography strength than SHA-256 [32], however, SHA-512 increases the length of the hash value and thereby the communication cost will be increased. Next, TA sets the system public parameters $params = \{p, q, a, b, G_1, G_2, e, P, Q, Q', P_{pub}, h_1, h_2, h_3\}$ and publishes $params$ to all cloud servers, fog servers, fog edge nodes, and vehicles where a and b are the parameters of the elliptic curve function $E_P(a, b)$. The notations used throughout this paper are listed in Table 1.

Table 1. Definition of notations in the proposed model

Notation	Description
$\oplus$	XOR
$\|\|$	Concatenation
TA	Trusted authority
TPD	Tamper-proof device
TRA	Trace authority
CS	Cloud server
FS	Fog server
FEN	Fog edge node
$\mathcal{V}$	Vehicle node
h_1, h_2, h_3	Secure hash functions
PID	Pseudo identity
RID	Real identity
P_{pub}	System public key
s	System private key
$params$	System public parameters
P, Q, Q'	Distinct generators
μ	A signature signed by the vehicle
μ^F	A signature signed by the fog edge node
t	The timestamp
e	Bilinear pairing

4.2. Registration Phase

In the registration phase, the TA performs the registration of vehicles, fog servers, fog edge nodes, and cloud servers. The following sections provide various registration processes.

1. Registration of Fog Edge Node: Let $\Re_{FEN} = \{FEN_1, FEN_2, \ldots, FEN_M\}$ be a set of authorized FENs that have been registered in the network. TA chooses a unique identity RID_{FEN_k} for each $FEN_k \in \Re_{FEN}$. It also randomly selects a number $s_{fen} \in Z_q^*$ as the private secret key of FEN and then computes the FEN's public key $PUB_{fen} = s_{fen}.P$.
2. Registration of Vehicle Nodes: Consider a set of authorized vehicle nodes that have been registered in the network $\mathfrak{V}_{\mathcal{V}} = \{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_N\}$. For each vehicle $\mathcal{V}_l \in \mathfrak{V}_{\mathcal{V}}$, the TA chooses a unique identity $RID_{\mathcal{V}_l}$. Each vehicle maintains its own real identity $RID_{\mathcal{V}_l}$ and password $PWD_{\mathcal{V}_l}$ in the TPD. TA also sends securely system private key s to the authorized vehicle and it will be

stored to TPD. For the privacy issue, vehicles do not use real identity, to be known by others. To this end, each vehicle uses the generated pseudo-identity $PID_{\mathcal{V}} = \{PID_{\mathcal{V},1}, PID_{\mathcal{V},2}\}$ by TPD and TRA that we explain more next.

3. Registration of Fog Server: Consider a set of fog servers that have been registered in the network $\mathfrak{F}_{FS} = \{FS_1, FS_2, \dots, FS_L\}$. For each fog server $FS_j \in \mathfrak{F}_{FS}$ to be deployed, the TA selects a unique real identity RID_{FS_j}. It chooses a random number $s_{fs} \in Z_q^*$ as the private secret key of fog server and then computes the FS's public key $PUB_{fs} = s_{fs}.P$.
4. Registration of Cloud Server: Let $\mathfrak{C}_{CS} = \{CS_1, CS_2, \dots, CS_P\}$ 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. Node Authentication Phase

Every vehicle $\mathcal{V}$ can send or receive data from any other vehicle $\mathcal{W}$ in the network. To ensure the security of the communication, before initiating any communication for sending and receiving the message, it needs to ensure the legitimacy of the sender by the receiver. To this end, each vehicle $\mathcal{V}$ is equipped with the two quotient filters $genuQF_{\mathcal{V}}$ and $fakeQF_{\mathcal{V}}$ to maintain all authorized and unauthorized nodes under the (FS_k), respectively. Depending legitimacy and or illegitimacy of vehicle nodes belonging to the FS_k, they will be stored in the relevant quotient filter of the vehicle using the fingerprint of pseudo vehicle identity $(PID_{\mathcal{W}})$, a public key (PUB_{fs}) provided by the related FS and trust score $(TrusT_{\mathcal{W}})$ (see Section 4.4) as follows:

$$(QF_{\mathcal{V}}) \leftarrow h_2\left(fingerprint(PID_{\mathcal{W}}) \oplus PUB_{fs} || TrusT_{\mathcal{W}}\right) \quad (1)$$

As the same way, fog servers also have two quotient filters $genuQF_{FS}$ and $fakeQF_{FS}$ to maintain genuine and fake vehicles. Whenever the vehicle enters the fog server's range, it upgrades its quotient filters, which includes the list of all the nodes registered with the fog server at that time.

In a vehicle-to-vehicle communication, before data sharing and communication, the destination node $\mathcal{V}_j$ performs $Query(\mathcal{V}_i)$ on its $genuQF_{\mathcal{V}_j}$. If the query returns TRUE as well as trust score is more than a threshold $(TrusT > TrusT_{min})$, it means the $\mathcal{V}_i$ is a genuine node, otherwise, the query will be performed on $fakeQF_{\mathcal{V}_j}$. If the query returns TRUE, it indicates that $\mathcal{V}_i$ is a fake vehicle node. If both queries return FALSE, $\mathcal{V}_j$ immediately sends a request to the FS_k. When the fog server FS_k receives the request, it will check the legitimacy of node $\mathcal{V}_i$ by performing a query on $genuQF_{FS_k}$and $fakeQF_{FS_k}$. If the query on $genuQF_{FS_k}$ returns TRUE along with a proper trust score, $\mathcal{V}_j$ start data sharing with $\mathcal{V}_i$ and updates $genuQF_{\mathcal{V}_j}$. Otherwise, if the query on $fakeQF_{FS_k}$ returns TRUE or FALSE, it means $\mathcal{V}_i$ as a fake vehicle node has entered the network. Additionally, if $\mathcal{V}_j$ does not receive a reply after a certain time from the FS_k, it just ignores the request of communication and data sharing with $\mathcal{V}_i$.

In a vehicle-to-fog edge node communication, FEN_l performs the query on its $genuQF_{FEN_k}$ and $fakeQF_{FEN_k}$. If both queries return FALSE, it needs to send a request to the relevant fog server. To reduce the cost of communication and computation time, FEN_l prepares a list of vehicles that had requested to data sharing. After preparation, it sends a request along with list $PID = \{PID_{\mathcal{V}_1}, PID_{\mathcal{V}_2}, \cdots, PID_{\mathcal{V}_n}\}$ to the fog server FS_k. When FS_k receives the request, it verifies the request using the FEN_l public key PUB_{fen_l}. If the request is legal, a query performs on $genuQF_{FS_k}$. If the query returns TRUE, it indicates all vehicles in the list are genuine, hence FEN_l starts communication with all vehicles on the list. If return FALSE, FEN_l identifies the genuine and fake nodes one by one. If the request is illegal, FS_k rejects the request.

4.4. Message Authentication Phase

Whenever a vehicle/FEN/FS want to send and or broadcast a message to nearby legitimate entities within its communication transmission range, it needs to sign the message first. In the other side, the receiver of

the message has to verify the signature. Based on the type of entities participated in this communication, the process of signature and verification will be different. Based on the designed architecture in this study (see Figure 1), the following communications in the fog-enabled VANET are possible: vehicle and vehicle **(V-V)**, vehicle and FN **(V-FEN),** vehicle and FS **(V-FS),** FEN and FS (**FEN-FS**), FEN and CS (**FEN-CS**), and FS and CS (**FS-CS**). It is supposed that communications among FEN, FS and CS are via a secure manner. In the following, we respectively explain the process of message signing and verification of the message for **V-FEN, V-FS, FEN-V,** and **V-V** communications:

A. V-FEN Communication:

[Vehicle's Pseudo-Identity Generation]: Before start communication and send the message, in order to fulfill privacy-preserving, each vehicle needs to generate its pseudo-identity. To generate pseudo-identity, TPD randomly selects a number $r_i \in Z_q^*$ as vehicle secret key and calculates $PID_{\mathcal{V}_i,1} = r_i.P$. When a vehicle node enters the VANET, TPD securely sends $\{RID_{\mathcal{V}_i}, PWD_{\mathcal{V}_i}, PID_{\mathcal{V}_i,1}\}$ to TRA. After verifying $\{RID_{\mathcal{V}_i}, PWD_{\mathcal{V}_i}\}$ and checking the legitimacy of $\mathcal{V}_i$, TRA calculates the pseudo-identity $PID_{\mathcal{V}_i} = \{PID_{\mathcal{V}_i,1}, PID_{\mathcal{V}_i,2}, VPT_{\mathcal{V}_i}\}$ by choosing a random number $z_i \in Z_q^*$, where $PID_{\mathcal{V}_i,2} = RID_{\mathcal{V}_i} \oplus h_3(PID_{\mathcal{V}_i,1} \| z_i.P_{pub})$ and $VPT_{\mathcal{V}_i}$ defines the valid period of the $PID_{\mathcal{V}_i}$.

[Message Signing by Vehicle]: To ensure message integrity and authentication, each message sent by a vehicle needs to sign and then the signature should be verified when the message received on the other side. Each message will be signed by a vehicle generating a pseudo-identity and related signing key. When a vehicle $\mathcal{V}_i$ enter the communication area of a fog edge node (FEN_j), it computes the $S_{\mathcal{V}_i} = s.\mathcal{H}_j.Q$ where $\mathcal{H}_j = h_1\left(RID_{FEN_j}\right)$ and store it in the TPD. It is obvious that $S_{\mathcal{V}_i}$ will be changed when $\mathcal{V}_i$ join the new fog edge node.
Then, vehicle $\mathcal{V}_i$ has to sign the message $\mathcal{M}_{\mathcal{V}_i} = m_i \| t_i$ where $\mathcal{M}_{\mathcal{V}_i}$ is combining of m_i as original message and t_i as the timestamp. The timestamp t_i gives the freshness of the signed message against a replay attack. To sign the $\mathcal{M}_{\mathcal{V}_i}$, TPD selects $k_i \in Z_q^*$ and calculates $U_{\mathcal{V}_i} = k_i.P$. Next, it computes $T_{\mathcal{V}_i} = S_{\mathcal{V}_i}.\mathcal{H}_i + k_i.Q'$ for $\mathcal{H}_i = h_2(PID_{\mathcal{V}_i} \| \mathcal{M}_{\mathcal{V}_i})$. The corresponding signature on $\mathcal{M}_{\mathcal{V}_i}$ for $PID_{\mathcal{V}_i}$ is $\mu_{\mathcal{V}_i} = (T_{\mathcal{V}_i}, U_{\mathcal{V}_i})$. Finally, the vehicle sends $\{PID_{\mathcal{V}_i}, \mathcal{M}_{\mathcal{V}_i}, \mu_{\mathcal{V}_i}\}$ to the relevant FEN.

[Message Verification by FEN]: Once FEN received the signed message from vehicle(s), not only it has to check legitimacy of the vehicle node(s), but also it needs to verify the signature related to the message. This process is to ensure that the corresponding vehicle is not attempting to impersonate any other legitimate vehicle or disseminate false messages. The message verification is performed in two ways: (i) single message verification; (ii) batch message verification. The first one will be used when FEN received only one emergency message of the vehicle and the second one will be utilized when it received a batch of messages from the different vehicles. In the following, we separately describe the process for each one:

- **Single Message Verification:** Once edge node $FEN_j \in \Re_{FEN}$ received a signed message $\{PID_{\mathcal{V}_i}, \mathcal{M}_{\mathcal{V}_i}, \mu_{\mathcal{V}_i}\}$, after checking the freshness of $|t_i - t_c| \leq \Delta t$ and VPT_i, if both message and pseudo-identity are valid, it calculates $\mathcal{H}_j = h_1\left(RID_{FEN_j}\right)$ and $\mathcal{H}_i = h_2(PID_{\mathcal{V}_i} \| \mathcal{M}_{\mathcal{V}_i})$, then verifies whether

$$e(T_{\mathcal{V}_i}, P) = e(P_{pub}.\mathcal{H}_j.\mathcal{H}_i, Q).e(U_{\mathcal{V}_i}, Q') \quad (2)$$

  hold or not. If it does not hold, FEN_j discards the message, otherwise, the message will be verified. The validation of Equation 2 is proved in Appendix A.

- **Batch Message Verification:** Once the fog edge node FEN_j received multiple signed messages from a group of vehicles in a time interval, it first categorized the messages into the different batches and then verify the messages using the batch message verification method. In this study, we divided the area under transmission range of a fog edge node into different sections. The received messages from each section within a time interval will be joined to the corresponding

batch. As shown in Figure 3, FEN_j searches in its own message list to extract the message related to each section and create the corresponding batch.

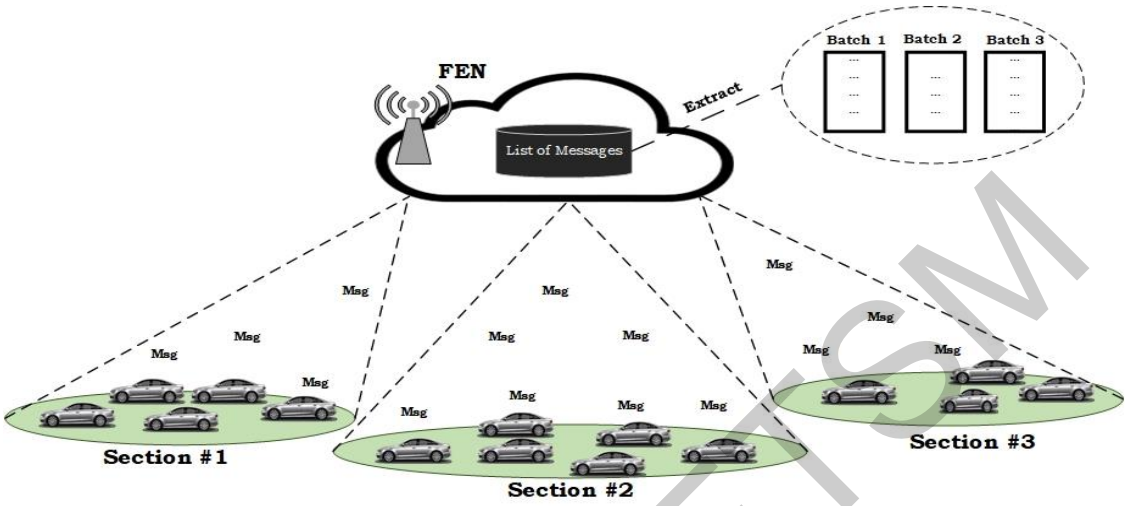

Figure 3. Batch message generation by FEN.

For example, consider n distinct vehicles $\mathfrak{V}_v = \{\mathcal{V}_1, \cdots, \mathcal{V}_n\}$ and corresponding message-signature tuples $signedMsgList = \left\{\{PID_{\mathcal{V}_1}, \mathcal{M}_{\mathcal{V}_1}, \mu_{\mathcal{V}_1}\}, \ldots, \{PID_{\mathcal{V}_n}, \mathcal{M}_{\mathcal{V}_n}, \mu_{\mathcal{V}_n}\}\right\}$. As explained above, it is assumed that these n vehicles were located in the same section when sending the message. To verify each batch, FEN_j computes $\mathcal{H}_i = h_1\left(RID_{FS_j}\right)$ and $\mathcal{H}_i = h_2\left(PID_{\mathcal{V}_i} \parallel \mathcal{M}_{\mathcal{V}_i}\right)$ for $i = 1, \ldots, n$ and then checks whether

$$e\left(\sum_{i=1}^{n} T_{\mathcal{V}_i}, P\right) = e\left(P_{pub}.\mathcal{H}_j.\sum_{i=1}^{n} \mathcal{H}_i, Q\right).e\left(\sum_{i=1}^{n} U_{\mathcal{V}_i}, Q'\right) \quad (3)$$

holds or not. If it is established, means the checking was successfully and hence accept the messages in the batch; otherwise, it means there is at least one invalid message in the batch. The validation of Equation 3 is in Appendix A. Whenever the Equation 3 is not established by FEN_j, an algorithm based on binary search will be used to detect the invalid messages contained in the batch (see Appendix B). The output of this algorithm is two lists for valid messages and invalid messages namely *validMsgList* and *invalidMsgList*. In this algorithm, *batchMsgVerification* and *singleMsgVerification* are batch and single message verification methods, respectively. Finally, the fog edge node FEN_j sends $List = \{validMsgList, invalidMsgList\}$ and the corresponding signature $SIGN\{List\}$ to the related fog server. When the fog server received the list from the FEN_j, it first verifies the message using the fog edge node public key PUB_{fen_j}. If so, fog server updates both filters $genuQF_{FEN_j}$ and $fakeQF_{FEN_j}$.

B. V-FS Communication:

Due to the vastness of the transportation network, the distributed fog edge nodes cannot cover all locations in the vehicular environment. Therefore, a vehicle sometimes is not under communication coverage of fog edge node. In this situation, it needs to communicate with the related fog server. When a vehicle detects no fog edge nodes are within its communication range, it sends the signed messages to the fog server. To verification the signed message, the fog server verifies whether the single/batch authentication holds or not. If it 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. To determine the invalid messages in the batch, the fog server performs Algorithm 1.

C. FEN-V Communication:

[Message Signing by FEN]: To ensure secure communication, each FEN also has to sign the event message and then broadcast to the vehicles within its transmission range as well as relevant FS. To this end, the $FEN_i \in \Re_{FEN}$ signs $\mathcal{M}_i = m_i \parallel t_i$ with private key $S_{FEN_i} = s_{fen}.\mathcal{H}_j.Q$ for $\mathcal{H}_j = h_1(RID_{FEN_i})$. Since privacy is not an important issue and a requirement for the FENs, hence its real identity (RID_{FEN_i}) is used to sign the message. The FEN computes $U^F_{FEN_i} = k_i.P$ where $k_i \in Z^*_q$ is a random number. Then, it computes $\mathcal{H}_i = h_2(RID_{FEN_i} \parallel \mathcal{M}_{\mathcal{V}_i})$. The corresponding signature on $\mathcal{M}_i$ is $\mu^F_{FEN_i} = (T^F_{FEN_i}, U^F_{FEN_i})$ where $T^F_{FEN_i} = S_{FEN_i}.\mathcal{H}_i + k.Q'$ and the FEN broadcasts $\{RID_{FEN_i}, \mathcal{M}_i, \mu^F_{FEN_i}\}$ to vehicles and the relevant FS.

[Message Verification by Vehicle]: Once $\mathcal{V}_j$ received a signed message $\{RID_{FEN_i}, \mathcal{M}_i, \mu^F_{FEN_i}\}$, after checking the freshness of t_i, it calculates $\mathcal{H}_j = h_1(RID_{FEN_i})$ and $\mathcal{H}_i = h_2(RID_{FEN_i} \parallel \mathcal{M}_{\mathcal{V}_i})$, then verifies whether

$$e(T^F_{FEN_i}, P) = e(PUB_{fen}.\mathcal{H}_j.\mathcal{H}_i, Q).e(U^F_{FEN_i}, Q') \quad (4)$$

hold or not. If it does not hold, $\mathcal{V}_j$ discards the message and marked the fog edge node as an intruder and report to the corresponding fog server. Otherwise, if the equation is established, accept the message. The validation of Equation 4 is explained in Appendix A.

D. V-V Communication:

Once vehicle $\mathcal{V}_l$ received a signed message from another vehicle $\mathcal{V}_k$, it firstly needs to check the legitimacy of $\mathcal{V}_k$ (see Section 4.3). If $\mathcal{V}_k$ is authorized and trustworthy, $\mathcal{V}_l$ checks the integrity of the message $\mathcal{M}_k = m_k \parallel t_k$. In a V2V communication, in order to check the integrity of the message, $\mathcal{V}_l$ sends a request $Req = \langle Req_{id}, PID_l, \mathcal{M}_k, PID_k \rangle$ to the related fog edge node and wait for a reply. As mentioned above, if there is no fog edge node in its communication range, it sends the request to the related fog server.

When the fog edge node FEN_j received a request from the vehicle, it checks to determine whether $\langle \mathcal{M}_k, PID_k \rangle$ is within the *validMsgList* or not. If exist, FEN_j sends a reply $Rep(verified)$ to the $\mathcal{V}_l$ to verify the message $\mathcal{M}_k$. Otherwise, FEN_j checks message in *invalidMsgList*. If exist, it responds $Rep(ignored)$ to $\mathcal{V}_l$. Otherwise, if the message is not existing in both *validMsgList* and in*validMsgList*, it means $\mathcal{V}_k$ didn't send the message to the FEN_j and or the fog edge node FEN_j received the message, after the request of $\mathcal{V}_l$. In this regard, FEN_j waits for a certain time. If it received the message during this time, FEN_j checks the authentication of the message using the single authentication method and responds back the result to the $\mathcal{V}_l$. Otherwise, it sends $Rep(ignored)$ to $\mathcal{V}_l$.

In the vehicle side, if $\mathcal{V}_l$ receives a reply of FEN_j, the vehicle *verified/ignored* the message based on the type of reply. Otherwise, if $\mathcal{V}_l$ not received a reply after a certain time, the message will be discarded.

4.5. Trust Measurement Phase

One of the other problems in the vehicular network is the presence of selfish nodes [15]. A selfish node is an authorized vehicle that wants to have the most benefits of the network for personal use only. To this end, these nodes build up trust first and then deceive. Selfish nodes change behavior over time. Therefore, selfish vehicles are considered a serious security threat by the creation and dissemination of incorrect information in the network. This behavior from nodes will result in a reduction of trust among vehicles.

To deal with selfish nodes, we proposed an experience-based trust model. To this end, each vehicle node has a trust score ($TrusT_{score} \in [0,1]$). It is based on three factors including the type of communication (direct-indirect), stability, and the previous score of trust. In the registration phase, $TrusT_{score} = 0.5$ assigns to each vehicle node when entering the network.

- NLOS: for delivering messages, nodes with direct communication are more reliable than nodes behind obstacles. The characteristic can be noted as $A_1 = \alpha_1(1 - NLOS)$, where $NLOS = 0\ or\ 1$.
- Stability: It specifies the time the node endured in the same state (line of sight or none line of sight). The characteristic is noted as $A_2 = \alpha_2\ Stab$.
- Previous Trust score: shows the previous trust score for the subject node. If there is no history of communication, it is equal to 0.5. We note it as $A_3 = \alpha_3\ TrusT_c$

where (α_i) is the normalization factor for each characteristic. We will utilize a weighted average method for computing trust score for each node, in which, weights (ω_i) are related to the values of the attribute such that:

$$TrusT_{score} = \frac{\omega_1 A_1 + \omega_2 A_2 + \omega_3 A_3}{\omega_1 + \omega_2 + \omega_3} \quad (5)$$

With nodes' reliability and reachability data, other services and applications can consider these values and attribute to their own trust assessment mechanism.

Consider $TrusT\ (\mathcal{V} \rightarrow \mathcal{W})\ in\ [0,1]$ as the trust value representing the range to which $\mathcal{V}$ trusts (or distrusts) node $\mathcal{W}$ based on $\mathcal{V}$'s personal experience in interacting with $\mathcal{W}$, in which 1 shows absolute trust and 0 indicates absolute distrust. When node $\mathcal{V}$ traces advice of $\mathcal{W}$, if the advice is assessed as reliable, the trust value will be increased using Equation 6, rather, if $\mathcal{W}$'s advice is assessed as unreliable, then trust value will be decreased by Equation 7 [17].

$$TrusT\ (\mathcal{V} \rightarrow \mathcal{W}) = TrusT\ + \gamma(1 - TrusT\ ); \quad 0 < \gamma < 1 \quad (6)$$

$$TrusT\ (\mathcal{V} \rightarrow \mathcal{W}) = TrusT\ + \delta(1 - TrusT\ ); \quad -1 < \delta < 0 \quad (7)$$

where γ and δ are positive increment and negative decrement factors, respectively.

The absolute values of γ and δ rely on various factors as a result of the environment dynamics including the data sparsity situation and the event/task-specific property. In this study, we set $|\delta| > |\gamma|$ by having $|\delta| = \alpha|\gamma|$ and $\alpha > 1$ to run the usual assumption that creating trust should be difficult, but easy tearing it down.

5. Security Analysis

In this section, we first show that our proposed signature scheme is secure against an alternatively chosen message attack under the random oracle model (**Theorem 1**). Then, we show that our proposed scheme satisfies several security requirements.

Considering the network model and the capability of the adversaries, the security model for our system is determined via a game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$. The adversary $\mathcal{A}$ could forge message $\{PID_{\mathrm{i}}, \mathcal{M}_{\mathrm{i}}, \mu_{\mathrm{i}}\}$ and it makes the following queries in the game.

Theorem 1: Our system is secure within the random oracle model for VANET.

Proof: Suppose that an adversary $\mathcal{A}$ is able to forge the message $\{PID_{\mathrm{i}}, \mathcal{M}_{\mathrm{i}}, \mu_{\mathrm{i}}\}$. We set-up a game between challenger $\mathcal{C}$ and $\mathcal{A}$ which can solve the CDH problem by running $\mathcal{A}$ as a subroutine with a non-negligible probability. It is also assumed that challenger $\mathcal{C}$ maintains two hash lists $List_{H_1}$, $List_{H_2}$.

Setup: $\mathcal{C}$ selects a random number s as the private key of the system and computes the public key using $P_{pub} = s.P$. It also chooses $\varphi \in Z_q^*$ and computes $Q' = \varphi P$. Then, $\mathcal{C}$ sends the system parameters $params = \{p, q, P, Q, P_{pub}, H_1, H_2\}$ to $\mathcal{A}$.

H_1-Oracle: When $\mathcal{A}$ creates an H_1 query with message $\mathfrak{m}$, $\mathcal{C}$ exams whether the tuple $\langle \mathfrak{m}, \mu_{H_1} \rangle$ is already in the hash $List_{H_1}$ or not. If so, $\mathcal{C}$ sends $\mu_{H_1} = H_1(\mathfrak{m})$ to $\mathcal{A}$. Then, $\mathcal{C}$ selects a random $\mu_{H_1} \in Z_q^*$ and then adds $\langle \mathfrak{m}, \mu_{H_1} \rangle$ into the $List_{H_1}$. At last, $\mathcal{C}$ sends $\mu_{H_1} = H_1(\mathfrak{m})$ to $\mathcal{A}$.

H_2-Oracle: When $\mathcal{A}$ creates an H_2 query with message $\{PID_{\mathrm{i}}, \mathcal{M}_{\mathrm{i}}, \mu_{H_2}\}$, $\mathcal{C}$ exams whether the tuple $\{PID_{\mathrm{i}}, \mathcal{M}_{\mathrm{i}}\}$ is already in the hash $List_{H_2}$. If so, $\mathcal{C}$ sends $\mu_{H_2} = H_2(PID_{\mathrm{i}} \parallel \mathcal{M}_{\mathrm{i}})$ to $\mathcal{A}$. Then, $\mathcal{C}$ selects a random $\mu_{H_2} \in Z_q^*$ and adds $\{PID_{\mathrm{i}}, \mathcal{M}_{\mathrm{i}}, \mu_{H_2}\}$ into the $List_{H_2}$. Finally, $\mathcal{C}$ sends $\mu_{H_1} = H_2(PID_{\mathrm{i}} \parallel \mathcal{M}_{\mathrm{i}})$ to $\mathcal{A}$.

Sign-Oracle: In case $\mathcal{A}$ demands a private key corresponding to PID_{i}, $\mathcal{C}$ requests a signature Y_i on PID_{i} to the signing oracle of our scheme. Then, $\mathcal{C}$ replies $\mathcal{A}$ with Y_i and stores (PID_{i}, Y_i) to the hash list. When $\mathcal{A}$ makes a sign query on $\mathcal{M}_{\mathrm{i}}$ for PID_{i}, $\mathcal{C}$ discovers the equivalent pair (PID_{i}, Y_i) from the hash list. If so, then $\mathcal{C}$ computes a signature μ_{i} by carrying out the signature algorithm. If not, $\mathcal{C}$ requests a query to obtain the corresponding private key Y_i. Then, $\mathcal{C}$ calculates a signature μ_{i} on $\mathcal{M}_{\mathrm{i}}$ for PID_{i} using Y_i, responds to $\mathcal{A}$ with μ_{i} and stores (PID_{i}, Y_i) to the hash list. It should be noted that $\mathcal{A}$'s view is equal to its view within the real attack.

Ultimately, $\mathcal{A}$ yields a forgery $\mu^* = (T^*, U^*)$ on $\mathcal{M}^*$ for PID^* such never requested by PID^* to the private key extraction oracle and the pair $(\mathcal{M}^*, PID^*)$ has never requested to the signing oracle, where $T^* = S^*.\mathcal{H} + k.Q'$ and $U^* = kP$. Then, $\mathcal{C}$ computes $S^* = (\mathcal{H})^{-1}(T^* - \varphi U^*)$ where S^* is a valid signature on PID^* of our scheme. Ultimately, $\mathcal{C}$ outputs S^* as a forgery of the proposed scheme. Therefore, the proposed scheme for VANETs includes security against forgery under the adaptively selected message attack in the random oracle model.

Theorem 2: (**Verification and Integrity of Message)** *the message's integrity and the node's legitimacy are ensured by the signature of the message.*

Proof: We proved that our scheme is secure against forgery under an adaptively selected message and an adaptively selected ID attack in the random oracle model under the CDH supposition. Consequently, pseudo-identity verification and message integrity are obtained.

Theorem 3: (**Resistance to Unauthorized Nodes)** *It guarantees an unauthorized node is unable to enter the network and initiating data sharing with authorized nodes.*

Proof: Each vehicle has two filters namely $genuQF_V$ and $fakeQF_V$ containing the pseudo-identity of genuine and fake nodes, respectively. 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 both filters. If both queries return FALSE, the vehicle node immediately sends a request to the related fog server. If the query on both filters $genuQF_{FS}$ and $genuQF_{FS}$ returns FALSE, it indicates that the vehicle node didn't register in TA before joining the VANET and hence it marks the vehicle node as a fake node. In addition, if the query on $fakeQF_{FS}$ returns TRUE, it means that the fog server 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 vehicle and fog edge nodes.

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

Definition 1: Computation Diffie–Hellman (CDH) problem can be explained as follow: Considering P, aP, bP for $a, b \in Z_p^*$, it is difficult to compute abP.

Proof: The vehicle $\mathcal{V}_i$ transmits message $\{PID_{\mathcal{V}_i}, \mathcal{M}_{\mathcal{V}_i}, \mu_{\mathcal{V}_i}\}$ to other nodes, where $PID_{\mathcal{V}_i} = \{PID_{\mathcal{V}_i,1}, PID_{\mathcal{V}_i,2}, VPT_{\mathcal{V}_i}\}$, $PID_{\mathcal{V}_i,1} = r_i.P$, and $PID_{\mathcal{V}_i,2} = RID_{\mathcal{V}_i} \oplus h_3(PID_{\mathcal{V}_i,1} \| z_i.P_{pub})$. The real identity $RID_{\mathcal{V}_i}$ of the vehicle is perfectly concealed since $PID_{\mathcal{V}_i}$ is an unknown identity with two random numbers r_i and s_{TRA}. Based on the CDH problem [33], it is hard to compute $z_i.r_i.P$ and hence the adversary cannot extract $RID_{\mathcal{V}_i}$. Hence, the suggested outline meets identity privacy needs.

In case a vehicle does not alter its pseudo-identity constantly within the connotation period, the vehicle movement trajectory can be traced by an adversary [8]. More precisely, when two messages m and m' more than Δt time apart are sent by the same vehicle, then an adversary should not be capable of determining whether m and m' originated from the same sender or not. In our system, considering signing all the messages with various pseudo-identity, in case the short expiration times $VPT_{\mathcal{V}_i}$ in the pseudo-identity satisfy $\Delta t > VPT_{\mathcal{V}_i}$, therefore, no message is linked to a vehicle.

Theorem 5: (Resistance to Selfish Nodes) *it is impossible for a selfish node to send the received signed message and attempt to send it if it is not valid.*

Proof: A selfish node, as security threats in the vehicular environment, is an authorized node that tries to maximize the car owner's utility by sending out false information to the neighbor nodes. In our scheme, the proposed trust model punishes the vehicle by decreasing the trust score whenever it is detected that a vehicle node is broadcasting incorrect information. Since the trust score $(TrusT)$ has an important role in start communicate with other nodes, it prevents broadcasting inaccurate data by vehicle nodes. Suppose a selfish node $\mathcal{S}$ with a trust score $TrusT_{\mathcal{S}} > min_{trust}$ start communicating with other nodes and broadcasts the wrong information to the neighbor nodes. Once vehicle nodes and or fog nodes (fog server and fog edge node) detect the wrong data, the related fog server will be decreased the $TrusT_{\mathcal{S}}$ and broadcast an alert to the nodes. After this, $\mathcal{S}$ cannot start communication with other vehicles until its trust score reaches a certain level over time. Hence, authorized nodes prevent sending inaccurate data to neighbor nodes. Consequently, this scheme provides protection against selfish attacks.

Theorem 6: (Resistance to Replay Attack) *it is not possible for an adversary to send the received signed message and attempt to send it if it is not valid.*

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 timestamps are 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 $\mathcal{A}$ intercepts a message$\{PID_i, \mathcal{M}_i, \mu_i\}$ where $\mathcal{M}_i = m_i \| t_i$ 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. Δt is a conjointly agreed to transmission delay. Therefore, this scheme provides protection against a replay attack.

Theorem 7: (Traceability) *TRA is able to track the real identity from the pseudonym of the vehicle.*

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

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

Proof: In order to assess the accountability of a malicious vehicle, the TRA should trace the vehicle's real identity. 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 $PID_{V_i} = \{PID_{V_i,1}, PID_{V_i,2}, VPT_{V_i}\}$ in a signed message and $PID_{V_i,2} = RID_{V_i} \oplus h_3(PID_{V_i,1} \| z_i.P_{pub})$, the TRA is able to trace the vehicle's real identity using definition 1.

$$RID_{V_i} = PID_{V_i,2} \oplus h_3(PID_{V_i,1} \| z_i.P_{pub})$$

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

6. Performance Evaluation

In this section, a comparison is made between our scheme and the related works CPAS [34], ASBV [35], and PPAS [36] in terms of communication and computation cost. We also have an analysis on the Quotient Filter and proposed trust model.

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 the message, before sending it. For analyzing the communication overhead of the presented system, we follow the safety message's format between vehicles and fog nodes as in [34] (see Figure 4). In this format, the signature is considered as cryptographic overhead. Obviously, to reduce communication costs, it needs to decrease the size of the signature. As explained in [34], to decrease the signature length, it is appropriate to utilize a 160-bit subgroup of the MNT curve with an embedding degree of 6.

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

In our scheme, the overall packet size can be decreased by 170 bytes where the signature is 20 bytes, and 42 bytes is for pseudo-identity.

According to [7,37], the size of each element $\{PID, U \in G_1\}$, timestamp $\{VPT\}$, the output of the hash function such as $\{\mu \in Z_q^*\}$, and real identity $\{RID\}$ is 40 bytes, 4 bytes, 20 bytes, and 10 bytes, respectively. So, given $\{PID_V, \mathcal{M}_V, \mu_V\}$ the total signature size of our scheme excluding message size $\mathcal{M}_V$ is 42 + 20 = 62 bytes where the total pseudo identity's size $\{PID_{V,1}, PID_{V,2}, VPT_V\}$ is 42 bytes. Additionally, our scheme uses a real identity, instead of pseudo-identity, for sending message form fog node to vehicle. Therefore, due to the size of the message, Type ID, Message-ID, timestamp, signature, and pseudo-identity, the total packet size from vehicle to fog node in our scheme is 170 bytes and it is 138 bytes for fog node to vehicle. Table 2 represents the communication cost comparison.

Table 2. Comparison of Communication Cost

Model	Type ID	Message ID	Payload	Timestamp	Signature	Pseudo ID	Total
CPAS	2 B	2 B	100 B	4 B	60 B	41 B	209 B
ASBV	2 B	2 B	100 B	4 B	344 B	40 B	492 B
PPAS	2 B	2 B	100 B	4 B	26 B	40 B	176 B
Our Scheme	2 B	2 B	100 B	4 B	20 B	42 B	170 B

6.2. Computation Overhead

Followed by receiving the messages, vehicles, and fog nodes should authenticate the messages' validity by proposed single or batch message verification before using them. Here, we assess our scheme's performance, CPAS, ASBV, and PPAS about computation overhead. These three schemes, as well as our scheme, are established on the bilinear pairings.

In this paper, by inspiring the computation evaluation method for VANET in [33], the bilinear pairing on the security level of 80 bits is made as e: $G_1 \times G_1 \rightarrow G_2$, where G_1 is an additive group created by a point P with the order q on the super singular elliptic curve $E: y^2 = x^3 + x \,(mod\, p)$ with embedding degree 2, specially p including a 512-bit prime number, q comprising of a 160-bit Solinas prime number. Regarding convenience, some notations for cryptographic execution time by using the MIRACL library are explained in [7,33]. MIRACL is a library for implementing number-theoretic based methods of cryptography.

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 scale multiplication operation's execution time $x.\bar{P}$ associated with the bilinear pairing, in which $\bar{P} \in G_1$ and $x \in Z_{\bar{q}}^*$ and $T_{bp.m} \cong 1.7090\ (ms)$
3. $T_{bp.sm}$: The execution time of a small scale multiplication operation $v_i.\bar{P}$ associated with the bilinear pairing utilized in the small exponent test, in which, $P \in G_1$, $v_i \in [1, 2^t]$ 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 MapToPoint hash operation associated with the bilinear pairing in which hash function maps a string $\{0,1\}^*$ to G_1 and $T_{mtp} \cong 4.4060\ (ms)$
6. $T_{e.m}$: A scale multiplication operation's execution time $x.P$ associated with the elliptic curve cryptography (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 scale 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. $T_h = 0.0001\ (ms)$

Here, we calculate the computation time of pseudo-identity generation (PIG), message signing (MS), single message verification (SMV), and batch message verification (BMV) for our scheme and related works, separately.

To pseudo-identity generation: our scheme comprises of two scalar multiplication processes, and one one-way hash function operation. Therefore, the whole procedure's overall computation time is $PIG(\text{Our Scheme}) = 2T_{bp.m} + T_h \cong 2 * 1.7090 + 0.0001 = 3.4181\ (ms)$. For ASBV, it also comprises of two scalar multiplication processes, and one one-way hash function operation. Therefore, the whole procedure's overall computation time is $PIG(\text{ASBV}) = 2T_{bp.m} + T_h \cong 2 * 1.7090 + 0.0001 = 3.4181\ (ms)$. For PPAS, it includes one map-to-point hash function and two scalar multiplication processes. Therefore, the whole procedure's overall calculation time is $PIG(\text{PPAS}) = 2T_{bp.m} + T_{mtp} \cong 2 * 1.7090 + 4.4060 = 7.8240\ (ms)$. And for CPAS, this includes one map-to-point hash function and three scalar multiplication processes. Thus, the total computation time of the whole procedure is $PIG(\text{CPAS}) = 3T_{bp.m} + T_h \cong 3 * 1.7090 + 4.4060 = 9.5330\ (ms)$

To message signing: to do this, our system includes one one-way hash function operation, three scalar multiplication processes, and one map-to-point hash function. Hence, the overall calculation time of the entire procedure is $MS(\text{Our Scheme}) = 3T_{bp.m} + T_h \cong 3 * 1.7090 + 0.0001 = 5.1271\ (ms)$. For ASBV, it also comprises of three scalar multiplication processes, one point-addition operation, one map-to-point hash function, and one one-way hash function operation. Therefore, the whole procedure's overall computation time is $MS(\text{ASBV}) = 3T_{bp.m} + T_{bp.a} + T_{mtp} + T_h \cong 3 * 1.7090 + 0.0071 + 4.4060 + 0.0001 = 9.5402\ (ms)$. And, 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 $MS(\text{PPAS}) = 3T_{bp.m} + T_{mtp} + 2T_h \cong 3 * 1.7090 + 4.4060 + 2 * 0.0001 = 9.5332\ (ms)$. 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 $MS(\text{CPAS}) = 5T_{bp.m} + T_h \cong 5 * 1.7090 + 2 * 0.0001 = 8.5452\ (ms)$.

To single message verification: our scheme involves one map-to-point hash function operation, two bilinear pairing processes, one one-way hash function operation, and two scalar multiplication processes. Hence, the entire procedure's overall computation time is $SMV(Our\ Scheme) = 2T_{bp} + T_h + 2T_{bp.m} \cong 2 * 4.2110 + 0.0001 + 2 * 1.7090 = 11.8401\ (ms)$. And, ASBV consists three bilinear pairing processes, one one-way hash function, one map-to-point hash function operation, and one scalar multiplication processes. Hence, the entire procedure's overall computation time is $SMV(ASBV) = 3T_{bp} + T_h + T_{mtp} + T_{bp.m} \cong 3 * 4.2110 + 0.0001 + 4.4060 + 1.7090 = 18.7841\ (ms)$. 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 $SMV(PPAS) = 2T_{bp} + 3T_h + T_{mtp} + 3T_{bp.m} \cong 2 * 4.2110 + 3 * 0.0001 + 4.4060 + 3 * 1.7090 = 17.9553\ (ms)$. At the end, CPAS comprises three bilinear pairing processes, three scalar multiplication processes, and one one-way hash function. Thus, the overall calculation time of the entire procedure is $SMV(CPAS) = 3T_{bp} + T_h + +3T_{bp.m} \cong 3 * 4.2110 + 0.0001 + 3 * 1.7090 = 17.7601\ (ms)$.

To batch message verification: our scheme is made up of includes two bilinear pairing processes, (n+1) scalar multiplication processes, (n) map-to-point hash function processes, and one one-way hash function processes. Therefore, the overall calculation time of the entire procedure is $BMV(Our\ Scheme) = 2T_{bp} + T_h + (n+1)T_{bp.m} \cong 2 * 4.2110 + 0.0001 + (n+1) * 1.7090 = 1.7090n + 10.1311\ (ms)$. ASBV involves three bilinear pairing processes, (n) scalar multiplication processes, (n) map-to-point hash functions, (n) one-way hash functions, and (3n-3) point-addition operations. Therefore, the overall calculation time of the entire procedure is $BMV(ASBV) = 3T_{bp} + nT_h + nT_{mtp} + nT_{bp.m} + (3n-3)T_{bp.a} \cong 3 * 4.2110 + n * 0.0001 + n * 4.4060 + (3n-3) * 0.0071 = 6.1364n + 12.6117\ (ms)$. PPAS contains two bilinear pairing processes, (n+1) scalar multiplication processes, (2n) map-to-point hash functions, and (2n+1) one-way hash functions. Accordingly, the entire procedure's overall calculation time is $BMV(PPAS) = 2T_{bp} + (2n+1)T_h + 2nT_{mtp} + (n+1)T_{bp.m} \cong 2 * 4.2110 + (2n+1) * 0.0001 + 2n * 4.4060 + (n+1) * 1.7090 = 10.5212n + 10.1311\ (ms)$. CPAS includes three bilinear pairing processes, (2n+1) scalar multiplication processes, and (n) one-way hash function. Therefore, the overall calculation time of the entire procedure

is $BMV(CPAS) = 3T_{bp} + nT_h + (2n+1)T_{bp.m} \cong 3 * 4.2110 + n * 0.0001 + (2n+1) * 1.7090 = 3.4181n + 14.3420\ (ms)$.
Table 3 illustrated the comparison of these schemes about pseudo-identity generation, single message authentication, message signing, and batch authentication.

Table 3. Comparison of Computation Cost

Model	PIG	MS	SMV	BMV
CPAS	9.5330	8.5452	17.7601	$3.4181n + 14.3420$
ASBV	3.4181	9.5402	18.7841	$6.1364n + 12.6117$
PPAS	7.8240	9.5332	17.9553	$10.5212n + 10.1311$
Our Scheme	3.4181	5.1271	11.8401	$1.7090n + 10.1311$

Based on this table, the computation cost of batch verification related to our scheme, CPAS, ASBV, and PPAS for 100 messages is 356.1520, 626.2517, 1062.2511, and 181.0311 milliseconds, respectively. It means that the batch verification phase of our scheme has higher improvement than CPAS, ASBV, and PPAS (see Figure 5). In this phase, the percentage improvement of the total operation time of the proposed scheme is approximately $\frac{356.1520-181.0311}{356.1520} \times 100 \cong 49.17\%$, $\frac{626.2517-181.0311}{626.2517} \times 100 \cong 71.09\%$, and $\frac{1062.2511-181.0311}{1062.2511} \times 100 \cong 82.95\%$.

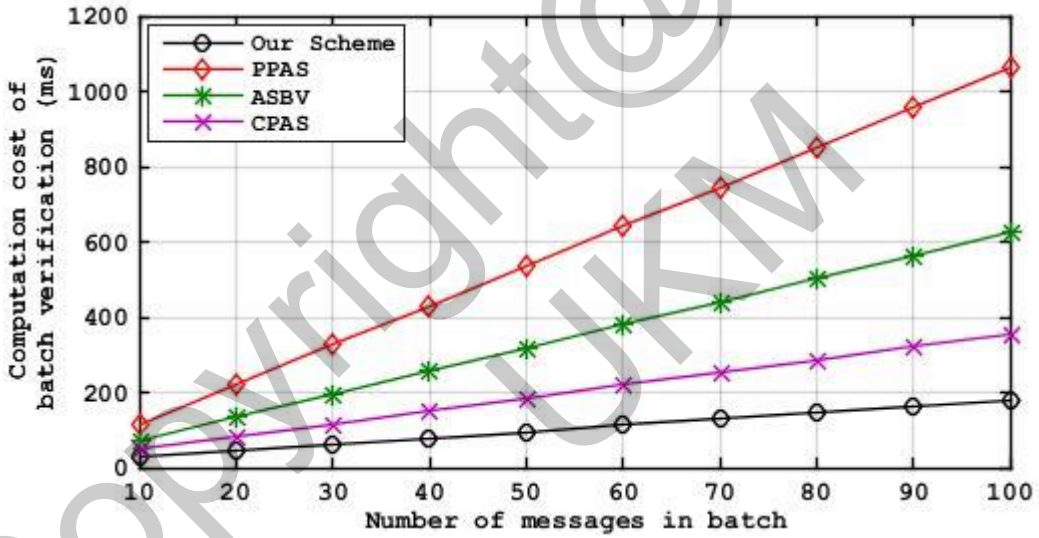

Figure 5. Comparison of computation time for batch message verification

6.3. Quotient Filter Analysis

A probabilistic data structure is used to boost up the lookup performance and to lower the memory consumption. Here, we analyze the quotient filter method utilized in the proposed scheme. We compare QF with BF based on false positive rate, memory usage, time spent, and throughput.
Regarding the false-positive rate, the less false positive rate for using quotient filter than standard bloom filter makes QF more accurate than BF. The false-positive formula for a BF is $\left(1 - e^{-kn/m}\right)^k$. In Quotient filter, the only case leading to a false positive is when two elements are mapped precisely to the same fingerprint. Regarding a good hash function, let the load factor of the hash table be $\alpha = n/m$, in which n shows the number of inserted elements, and $m = 2q$ shows the number of filter slots. Thus, the possibility of such a hard collision is almost $1 - e^{-\alpha/2^r} \leq 2^{-r}$ [28].
Regarding the memory, the space needed by a quotient filter can be compared to a bloom filter relying on the parameter selections. Generally, QF requires 10 - 25% more space than a BF, but it is faster than BF. This is because a BF has more hashing functions whereas a QF needs to utilize merely a single hash function for each access.

In terms of execution time and throughput, QF has better performance than BF. According to [38], 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. Base on this, the throughput of QF is about 310 *Mbits/sec* and it is 155 *Mbits/sec* for BF.

6.4. Trust Model Analysis

In this subsection, we analyze the proposed trust model under a different percentage of selfish nodes. The selfish nodes, as an attacker, can vigorously contribute to the network and disturb the integrity of the message by broadcasting wrong information. We compare the performance of the presented trust model with the Weighted Voting (WV) method [39]. To this end, we use the overall accuracy which represents the proportion of the total number of correct results. The following equation is used to computes the overall accuracy.

$$TrusT_{\ level} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{8}$$

where TP is the number of nodes properly found as selfish nodes, TN is the number of nodes correctly detected as unselfish nodes, FP is the number of nodes incorrectly found as selfish nodes and FN is the number of nodes incorrectly detected as unselfish nodes.
Figure 6 shows that the proposed trust model is more accurate than WV approach. The overall accuracy of our model is 93% when 10% of vehicle nodes in the network are the selfish node, whereas it is 87% for WV.

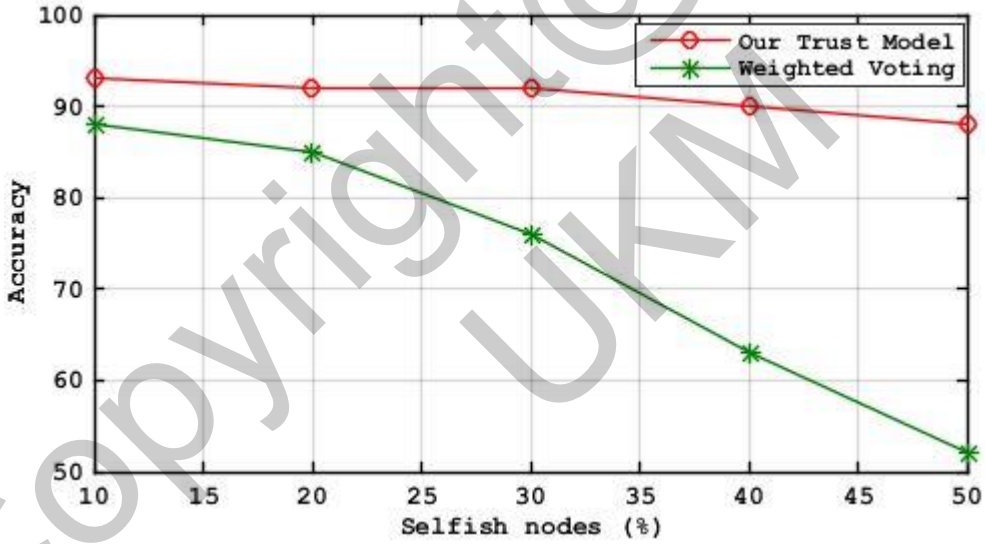

Figure 6. Comparison of the accuracy between the proposed trust model and WV

7. Simulation with NS-2

In this section, we analyze and discuss on the results obtained from the simulation of proposed scheme by NS-2. In this network simulator, many libraries work together to try to approach the real environment.

7.1. Basis of Scheme Simulation

Here, we use NS-2 with SUMO and MOVE for the urban environment in which the Open Street Map (OSM) file of Kuala Lumpur, from the database, is extracted. The simulation area is 5 km × 5 km and the highest node density on the simulation area is 500 nodes. We consider 5 FSs and 15 FENs along the roadside for serving the vehicle nodes. FSs and FENs are mounted at appropriate distances to provide sufficient coverage to take advantage of a fog computing based VANET. Each FS can serve 500 demands at the same time. To model the wireless channel, the two-ray ground reflection model is utilized as the radio propagation model. Moreover, the vehicles' transmission range is adjusted at 300 m. All vehicle

nodes are equipped with both DSRC module and LTE. DSRC is designed based on IEEE 802.11p. In our simulation, the medium utilized for communications between vehicles is IEEE 802.11p, whereas communication technology between vehicles and fog edge nodes is via LTE. Fog nodes also can connect to cloud through their LTE interface card. 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 s at the start of simulation for removing the impact of transient performance over the results. The overall simulation time also involved 30 s of stop sending packets from the simulation end. For simplicity, fog nodes and vehicles are assumed to be armed with 3.4GHZ i7-2600 machine.
Table 4 shows the parameters' values used in our tests. These parameters' values have been carefully selected after trying several ones, aiming to reflect a scenario as much realistic as possible.

Table 4. Experiment Parameters.

Parameter	Value
Radio Propagation	Two Ray Ground
Antenna Type	Omni-Antenna
MAC Layer	802.11p
Routing Protocol	AODV
Radio Range	300 m
Data Rate	6 Mbps
Packet Payload	152 bytes
Number of Vehicles	50 - 500
Number of FSs	5
Number of FENs	15
Velocity Limits	20 – 150 km/h
Road Length	5 km
Simulation Time	360 Second per each run

7.2. Result of Simulation

Here, we use three indexes false-positive rate (FPR), transmission delay, and packet loss ratio to evaluate our scheme and in addition comparison with other related works:

FPR: First, we prove the validity of our scheme using the Monte-Carlo simulation. According to [40], Monte-Carlo simulation investigates the validity and reliability of the model. As per Monte-Carlo's rule, the experiment is repeated a very large number of times. In this work, we perform 1000 Monte-Carlo simulations for a large-scale network to estimate FPR. It is measured as follow:

$$FPR = \frac{FP}{FP + TN} \tag{8}$$

where FP is the number of nodes incorrectly found as malicious nodes and TN is the number of nodes correctly detected as non-malicious nodes.
In order to understand the importance of modules used in our scheme, we evaluate the scheme in three different ways as follows: (i) with a message authentication module, node authentication module, and trust module (SchMNT), (ii) with a message authentication module, and node authentication module (SchMN), (iii) and finally, with just message authentication module (SchM).
In this work, FPR denotes the percentage of false messages that our scheme failed to reject. The result of the Monte-Carlo simulation is presented in Figure 7. It indicates that the false-positive rate would increase as the malicious node increases in the network, but this is not significant for the SchMNT. As shown in this figure, in the worst case where 90% of the vehicle nodes spread the false message, the FPR of the SchMNT is about 7.1%, whereas it is 13.4% and 17.4%, respectively, for the SchMN and SchM. Monte-

Carlo simulation results also validate SchMNT and show better performance than other comparable approaches.

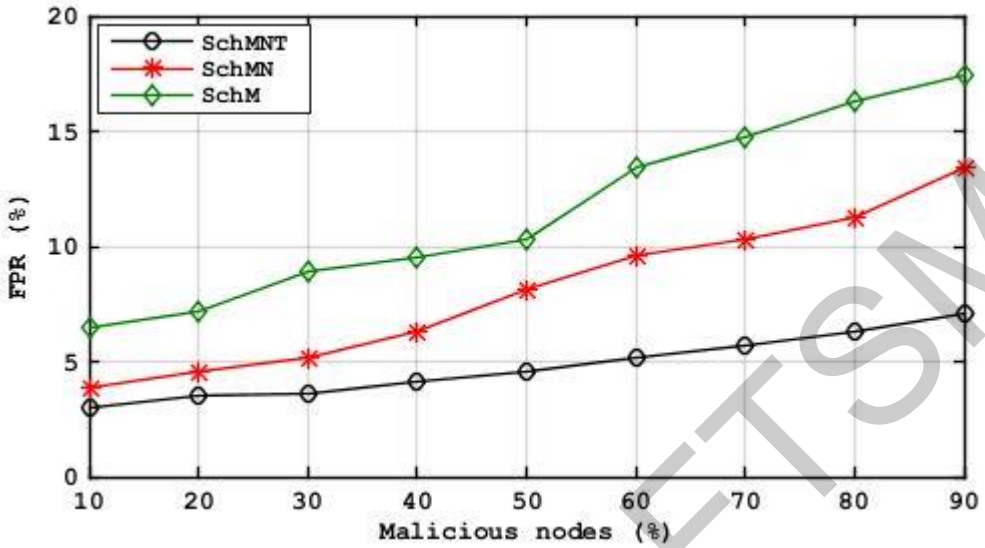

Figure 7. The false-positive rate of our scheme in a different way under the different % of malicious

Transmission Delay: In order to show our scheme's efficiency, we utilized the transmission delay for quantifying the communication overhead. As mentioned earlier, a vehicle/FEN/FS has to sign the message before broadcasting it over the network. Clearly, this process increases the size of exchanged message and in result caused transmission delay between vehicle-to-vehicle and or between vehicle-to-fog node. We compared the average transmission delay of our scheme with CPAS, ASBV, and PPAS under different density when the velocity of all vehicles is 20 km/h. We also evaluate our scheme under different density and velocity. As shown in Figure 8, the average transmission delay is respectively 1.11 *ms*, 1.52 *ms*, 1.77 *ms*, and 0.77 *ms* for CPAS, ASBV, PPAS, and our scheme. From Figure 8, we see the average transmission delay increases by incrementing the number of vehicles from 50 to 500. This is because the size of messages exchanged in the network will be increased as number of vehicles increased. Figure 9 shows the impact of velocity on transmission delay related to our scheme under different density. As we can see, velocity has slightly effect on the transmission delay. The obtained results were conceivable as our scheme has lowest message size and in result lowest communication overhead in comparison with PPAS, ASBV, and CPAS.

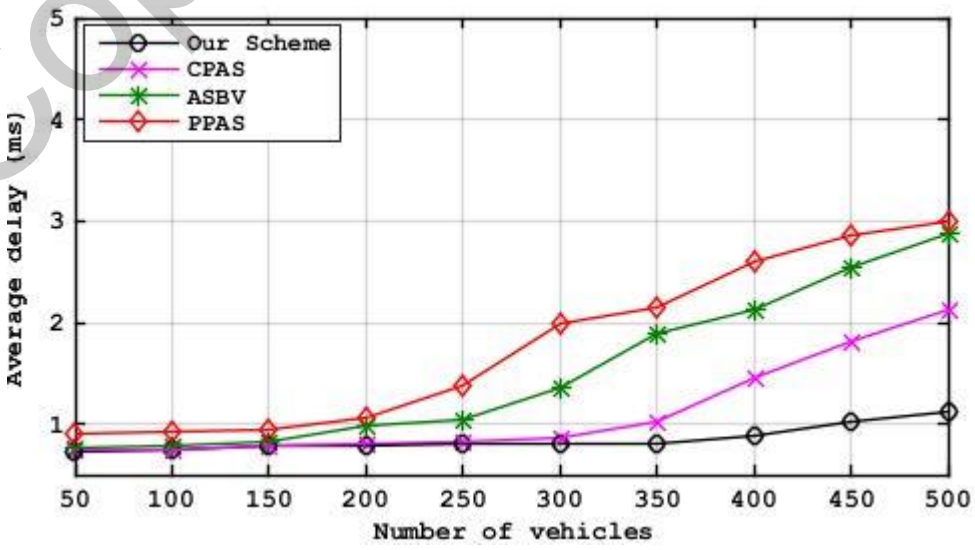

Figure 8. Comparison of average transmission delay under different density

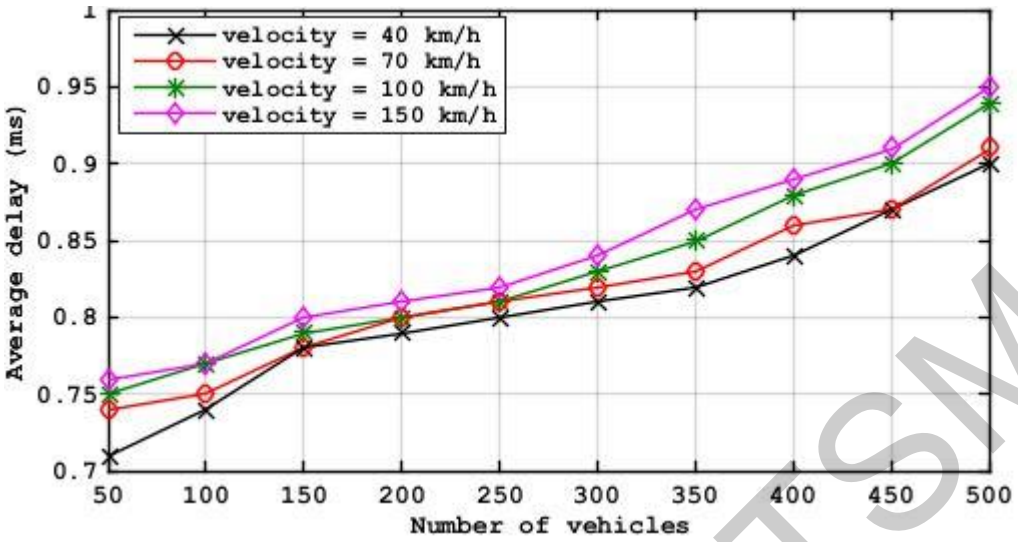

Figure 9. The impact of velocity on transmission delay under different density

Packet Loss Ratio: It is clear that size of signed messages and also the number of messages transferred over the network has impact on packet loss ratio. Hence, the ratio of packet loss can be a useful metric to reflect efficiency of our scheme. We presented the equivalent packet loss ratio for our scheme, CPAS, ASBV, and PPAS in Figure 10. As depicted in this figure, it is observed that by increasing the number of vehicles in the communication range, the transmission loss ratio increases. This is mainly because of the increasing the number of messages transferred over the network as the vehicle density roses. We also examined the effect of velocity on packet loss ratio. As observed in Figure 11, the packet loss rises with the increase in the velocity of vehicles. But this effect is not significant. This is because the propagation speed of radio waves is much higher than the moving speed of the vehicles. As we can see in this figure, there are not many differences of packet loss for our scheme when velocity reaches to 150 km/h from 40 km/h. To improve the ratio of packet loss, it is better to focus on communication cost and decrease size of signed messages.

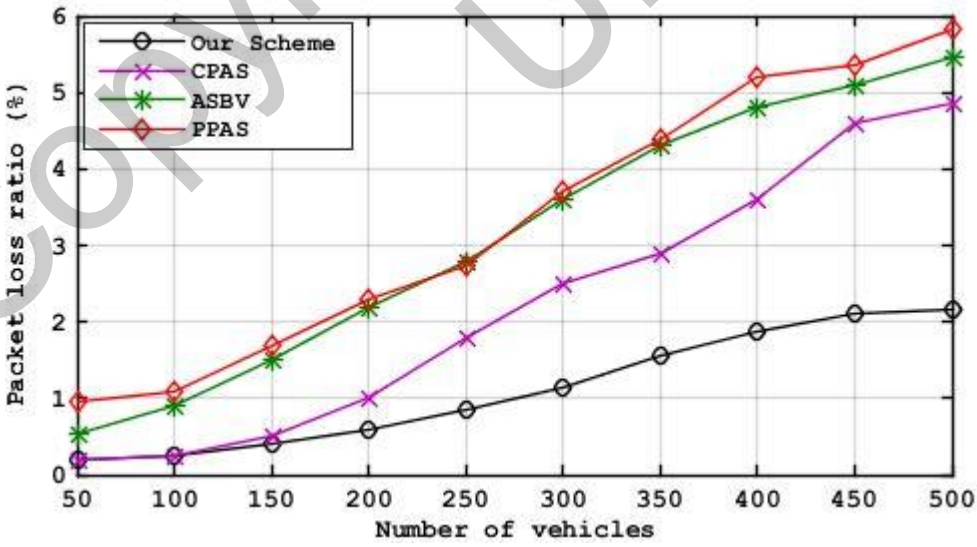

Figure 10. Comparison of packet loss ratio under different density

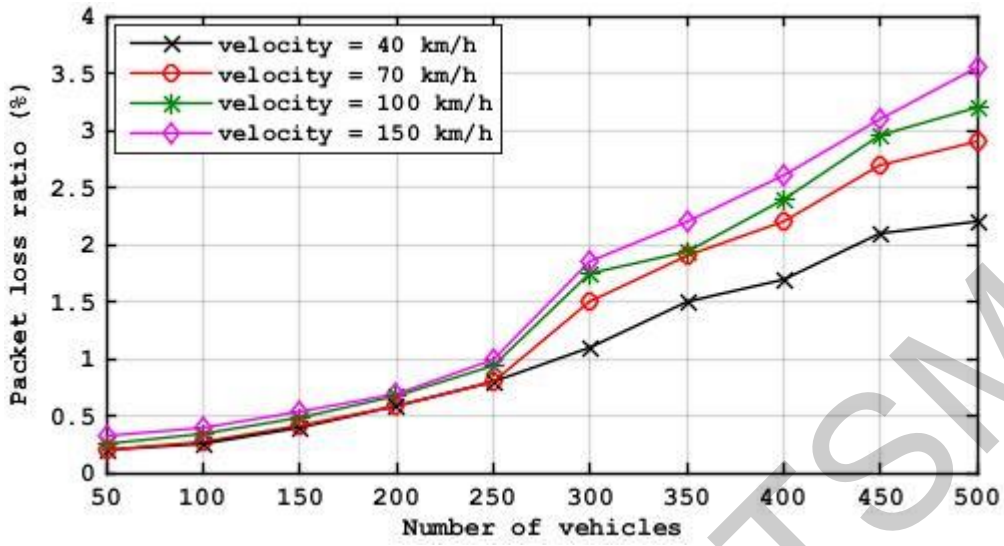

Figure 11. The impact of velocity on packet loss ratio under different density

8. Conclusion

VANET is an important component of ITS, which is crucial for creating an intelligent space, and has led to the development of many applications. However, its security is a major concern in the research community, since it in itself is an open-access and self-organized environment. To that end, we proposed a security model based on authentication, privacy, and trust. In this model, in order to reduce latency and enhance security, the fog nodes were distributed along the roadside. Additionally, due to the amount of data generated in the VANET, a quotient filter was used to maintain the required authorization for the vehicles. In the proposed security model, a node authentication scheme was proposed to ensure the legitimacy of the nodes which entered the network. Before initiating data sharing, the authentication of the vehicle node was checked using this scheme. It dramatically decreased the communication overhead. A message authentication scheme using a bilinear pairing was also developed to guarantee the integrity of the event of the messages, by signing the message, and running through a signature verification cycle. A lightweight trust scheme based on experience was proposed to cope with selfish nodes. To this end, vehicle nodes with a specific level of trustworthiness, are able to communicate with other nodes, otherwise, they are not allowed to join the network. In terms of preserving privacy, we used a pseudonym for the vehicle nodes. Our security model meets the security needs for the VANET as demonstrated by the conducted security analysis. Based on the performance analysis findings, it was shown that the model had a better performance compared to comparable systems, and it is more appropriate for deployment in real VANET settings. However, the computation cost of our scheme is still high, as it is based on bilinear pairing, and this matter leads to performance issues. In the future, we plan to develop a message authentication scheme based on an elliptic curve cryptography, to help reduce the communication and computation costs. We also intend to incorporate a cloud-fog computing in a 5G-VANET environment.

Appendix A

The proposed model is established on bilinear pairings. Let A, B, C be three generators in G_1 and e be a bilinear map. Based on the bilinear map's concepts, it satisfies the following properties:

Property 1. $e(A + b, C) = e(A, C).e(B, C)$

Property 2. $e(xyA, B) = e(A, xyB)$

Based on the properties and the equations that mentioned above:

Equation 2 Validation:

$$e(T_{V_i}, P) = e\left(S_{V_i}.\mathcal{H}_i + k_i.Q', P\right) = e(S_{V_i}.\mathcal{H}_i, P).e(k.Q', P)$$

$$= e\left(s.\mathcal{H}_j.Q.\mathcal{H}_i, P\right).e(k.Q', P) = \left(P_{pub}.\mathcal{H}_j.\mathcal{H}_i, Q\right).e\left(U_{V_i}, Q'\right)$$

Equation 3 Validation:

$$e\left(\sum_{i=1}^{n} T_{V_i}, P\right) = e\left(\sum_{i=1}^{n} S_{V_i}.\mathcal{H}'_{V_i} + k_i.Q', P\right) = e\left(\sum_{i=1}^{n} s.\mathcal{H}_j.Q.\mathcal{H}'_{V_i}, P\right).e\left(\sum_{i=1}^{n} k_i.Q', P\right)$$

$$= e\left(\sum_{i=1}^{n} s.P.\mathcal{H}_j.\mathcal{H}'_{\mathrm{i}}, Q\right).e\left(\sum_{i=1}^{n} k_i.P, Q'\right) = e\left(P_{pub}.\mathcal{H}_j.\sum_{i=1}^{n} \mathcal{H}'_{\mathrm{i}}, Q\right).e\left(\sum_{i=1}^{n} U_{V_i}, Q'\right)$$

Equation 4 Validation:

$$e\left(T^F_{FEN_i}, P\right) = e\left(S_{FEN_i}.\mathcal{H}_i + k.Q', P\right) = e\left(S_{FEN_i}.\mathcal{H}_i, P\right).e(k.Q', P)$$

$$= e\left(s_{fen}.\mathcal{H}_j.Q.\mathcal{H}_i, P\right).e(k.Q', P) = \left(PUB_{fen}.\mathcal{H}_j.\mathcal{H}_i, Q\right).e\left(U^F_{FEN_i}, Q'\right)$$

Appendix B

The following algorithm, using a binary search, try to identify the invalid messages exist in the batch.

Algorithm 1. Detecting valid and invalid messages in the batch using binary search

```
if batchMsgVerification (signedMsgList, Lindex, Hindex) == true then
    validMsgList.Insert (signedMsgList [Lindex, ..., Hindex])
    return 1
else
    if Lindex == Hindex then
        if singleMsgVerification (signedMsgList [Lindex]) == true then
            validMsgList.Insert (signedMsgList [Lindex])
            return 1
        else
            invalidMsgList.Insert (signedMsgList [Lindex])
            return 1
        endif
    else
        Mindex = (Lindex + Hindex)/2
        batchMsgVerification (signedMsgList, Lindex, Mindex)
        batchMsgVerification (signedMsgList, Mindex+1, Hindex)
    Endif
Endif
```

References

1. Khan, A. A., Abolhasan, M., & Ni, W. 5G next generation VANETs using SDN and fog computing framework. In 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), **2018,** (pp. 1-6). IEEE.
2. Yi, S., Li, C., & Li, Q. A survey of fog computing: concepts, applications and issues. In Proceedings of the 2015 workshop on mobile big data, **2015**, (pp. 37-42).

3. Lyu, L., Jin, J., Rajasegarar, S., He, X., & Palaniswami, M. Fog-empowered anomaly detection in IoT using hyperellipsoidal clustering. IEEE Internet of Things Journal, **2017**, 4(5), 1174-1184.
4. Garg S, Singh A, Kaur K, Aujla GS, Batra S, Kumar N, Obaidat MS. Edge computing-based security framework for big data analytics in VANETs. IEEE Network. **2019**, 27;33(2):72-81.
5. Qu F, Wu Z, Wang FY, Cho W. A security and privacy review of VANETs. IEEE Transactions on Intelligent Transportation Systems. **2015**;16(6):2985-96.
6. Raya, M., & Hubaux, J. P. Securing vehicular ad hoc networks. Journal of computer security, **2007**, 15(1), 39-68.
7. Cui, J., Wei, L., Zhang, J., Xu, Y., & Zhong, H. An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks. IEEE Transactions on Intelligent Transportation Systems, **2018**, 20(5), 1621-1632.
8. Zhang, C., Lin, X., Lu, R., Ho, P. H., & Shen, X. An efficient message authentication scheme for vehicular communications. IEEE transactions on vehicular technology, **2008**, 57(6), 3357-3368.
9. Zhang, C., Lu, R., Lin, X., Ho, P. H., & Shen, X. An efficient identity-based batch verification scheme for vehicular sensor networks. In IEEE INFOCOM 2008-The 27th Conference on Computer Communications, **2008**, (pp. 246-250). IEEE.
10. Huang, J. L., Yeh, L. Y., & Chien, H. Y. ABAKA: An anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks. IEEE Transactions on vehicular technology, **2010**, 60(1), 248-262.
11. Chim, T. W., Yiu, S. M., Hui, L. C., & Li, V. O. SPECS: Secure and privacy enhancing communications schemes for VANETs. Ad Hoc Networks, **2011**, 9(2), 189-203.
12. Liu, Y., Wang, L., & Chen, H. H. Message authentication using proxy vehicles in vehicular ad hoc networks. IEEE Transactions on Vehicular Technology, **2014**, 64(8), 3697-3710.
13. Lo, N. W., & Tsai, J. L. An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings. IEEE Transactions on Intelligent Transportation Systems, **2015**, 17(5), 1319-1328.
14. Wazid, M., Bagga, P., Das, A. K., Shetty, S., Rodrigues, J. J., & Park, Y. H. AKM-IoV: authenticated key management protocol in fog computing-based internet of vehicles deployment. IEEE Internet of Things Journal, **2019**, 6(5), 8804-8817.
15. Soleymani S.A., Abdullah A.H., Zareei M., Anisi M.H., Vargas-Rosales C., Khan M.K., Goudarzi S. A secure trust model based on fuzzy logic in vehicular ad hoc networks with fog computing. IEEE Access, **2017**, 5:15619-29.
16. Soleymani, S.A., Abdullah, A.H., Hassan, W.H., Anisi, M.H., Goudarzi, S., Baee, M.A.R. and Mandala, S. Trust management in vehicular ad hoc network: a systematic review. EURASIP Journal on Wireless Communications and Networking, **2015**,1, p.146.
17. Minhas, U. F., Zhang, J., Tran, T., & Cohen, R. A multifaceted approach to modeling agent trust for effective communication in the application of mobile ad hoc vehicular networks. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), **2010**, 41(3), 407-420.
18. F. G. Mármol and G. M. Pérez. TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks. J. Netw. Comput. Appl., **2012**, vol. 35, no. 3, pp. 934–941.
19. M. M. Mehdi, I. Raza, and S. A. Hussain. A game theory-based trust model for Vehicular Ad hoc Networks (VANETs). Comput. Netw., **2017**, vol. 121, pp. 152–172.
20. M. Raya, P. Papadimitratos, V. D. Gligor, and J.-P. Hubaux. On datacentric trust establishment in ephemeral Ad hoc networks. in Proc. 27th Conf. Comput. Commun. (INFOCOM), Apr. **2008**, pp. 1238–1246.
21. Huang C, Lu R, Choo KK. Vehicular fog computing: architecture, use case, and security and forensic challenges. IEEE Communications Magazine. **2017**;55(11):105-11.
22. Goudarzi S, Anisi MH, Abdullah AH, Lloret J, Soleymani SA, Hassan WH. A hybrid intelligent model for network selection in the industrial Internet of Things. Applied Soft Computing. **2019**, 1;74:529-46.
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, **2018**, 5(2), 1055-1069.
24. Singh, A., Garg, S., Kaur, R., Batra, S., Kumar, N., & Zomaya, A. Y. Probabilistic data structures for big data analytics: A comprehensive review. Knowledge-Based Systems, **2019**, 104987.
25. Dutta, S., Narang, A., & Bera, S. K. Streaming quotient filter: a near optimal approximate duplicate detection approach for data streams. Proceedings of the VLDB Endowment, **2013**, 6(8), 589-600.

26. Singh, A., Garg, S., Batra, S., & Kumar, N. Probabilistic data structure-based community detection and storage scheme in online social networks. Future Generation Computer Systems, **2019**, 94, 173-184.
27. D. E. Knuth. The Art of Computer Programming: Sorting and Searching, **1973**, volume 3. Addison Wesley.
28. Bender, M. A., Farach-Colton, M., Johnson, R., Kraner, R., Kuszmaul, B. C., Medjedovic, D. & Zadok, E. Don't Thrash: How to Cache Your Hash on Flash. PVLDB, **2012**, 5(11), 1627-1637.
29. Pandey, P., Bender, M. A., Johnson, R., & Patro, R. A general-purpose counting filter: Making every bit count. In Proceedings of the 2017 ACM international conference on Management of Data, **2017**, (pp. 775-787).
30. Manvi, S. S., & Tangade, S. A survey on authentication schemes in VANETs for secured communication. Vehicular Communications, **2017**, 9, 19-30.
31. Lenstra AK, Verheul ER. Selecting cryptographic key sizes. Journal of cryptology. **2001**;14(4):255-93.
32. Smitha A, Pai MM, Ajam N, Mouzna J. An optimized adaptive algorithm for authentication of safety critical messages in VANET. In2013 8th International Conference on Communications and Networking in China (CHINACOM) **2013**, (pp. 149-154). IEEE.
33. 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, **2017**, 66(11), 10283-10295.
34. Shim, K. A. CPAS: An Efficient Conditional Privacy-Preserving Authentication Scheme for Vehicular Sensor Networks. IEEE Transactions on Vehicular Technology, **2012**, 61(4), 1874-1883.
35. Bayat, M., Barmshoory, M., Rahimi, M. and Aref, M.R. A secure authentication scheme for VANETs with batch verification. Wireless networks, **2015**, 21(5), pp.1733-1743.
36. Zhong, H., Han, S., Cui, J., Zhang, J., & Xu, Y. Privacy-preserving authentication scheme with full aggregation in VANET. Information Sciences, **2019**, 476, 211-221.
37. Xie, Y., Wu, L., Shen, J., & Alelaiwi, A. EIAS-CP: new efficient identity-based authentication scheme with conditional privacy-preserving for VANETs. Telecommunication Systems, **2017**, 65(2), 229-240.
38. Al-Hisnawi, M., & Ahmadi, M. Deep packet inspection using quotient filter. IEEE Communications Letters, **2016**, 20(11), 2217-2220.
39. Ahmed, S., Al-Rubeaai, S., & Tepe, K. Novel trust framework for vehicular networks. IEEE Transactions on Vehicular Technology, **2017**, 66(10), 9498-9511.
40. Soleymani SA, Goudarzi S, Anisi MH, Kama N, Adli Ismail S, Azmi A, Zareei M, Hanan Abdullah A. A Trust Model Using Edge Nodes and a Cuckoo Filter for Securing VANET under the NLoS Condition. Symmetry. **2020** ;12(4):609.

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