smart cities

MDPI

Article

A Decentralized Blockchain-Based Trust Management Framework for Vehicular Ad Hoc Networks

Tahani Gazdar *, Ohoud Alboqomi and Asmaa Munshi

College of Computer Science and Engineering, University of Jeddah, Jeddah 21959, Saudi Arabia; oalboqomi.stu@uj.edu.sa (O.A.); asmmunshi@uj.edu.sa (A.M.)
* Correspondence: taalgazdar@uj.edu.sa

Abstract: Vehicular Ad hoc Networks (VANETs) are one of the pillars of the Internet of Vehicles, they provide plenty of applications ranging from safety to entertainment. Safety applications largely depend on reliable and authentic traffic-related data. However, ensuring the data reliability and authenticity is facing many challenges due mainly to the scalability of VANETs such as the high speed, the long roads, and the open nature of VANETs. This paper addresses these challenges by proposing a decentralized Blockchain-based trust management framework (BC-TMF) aiming to compute trust metrics for vehicles. These trust metrics rely on the authenticity of the messages. Each vehicle assesses the authenticity of the received messages in real-time, calculates a local trust metric for the originator of such messages, then shares it with a miner. Periodically each miner aggregates the received trust metrics into global trust metrics, then packs them in a block. To investigate the efficiency and consistency of the proposed framework, extensive simulations are conducted. The obtained results show that the proposed BC-TMF has an excellent capability in computing accurate trust metrics for vehicles. Besides, it outperforms the existing ones in terms of the accuracy of computed trust metrics, particularly for malicious vehicles.

Keywords: VANETs; trust; vehicle; malicious; blockchain

Citation: Gazdar, T.; Alboqomi, O.; Munshi, A. A Decentralized Blockchain-Based Trust Management Framework for Vehicular Ad Hoc Networks. *Smart Cities* **2022**, *5*, 348–363. https://doi.org/10.3390/smartcities5010020

Received: 8 February 2022
Accepted: 9 March 2022
Published: 12 March 2022

1. Introduction

1.1. Problem Statement

VANETs are a fundamental component of the Internet of Vehicles (IoV) [1], which is one of the newest technology designed for Smart transportation in Smart Cities. In VANETs, a myriad of smart vehicles, smart devices, and roadside units (RSUs) communicate to provide plenty of applications to the passengers [2]. These applications range from safety to entertainment aiming to facilitate and improve the passengers' journey. Particularly, safety applications in VANETs are based on a frequent exchange of alert messages aiming to share information reporting the road state, the traffic crowding, and sudden events such as vehicle crashes. The ultimate aim of the safety applications is to enhance road safety via vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. Nevertheless, safety applications require a reliable, authentic, and appropriately-timed exchange of data to avoid severe damage [3]. The vehicles should not only authenticate their peers, originators of messages but also assess the reliability of the received data to react accordingly. According to the IEEE 1609.2 standard [4], a public key infrastructure (PKI) would be used to authenticate vehicles using certificates. However, PKIs cannot be used to ensure the reliability of the data. Besides, a PKI allows mitigating only outsider attackers, an authenticated vehicle holding a valid certificate may forge messages without being detected. The concept of trust is used to overcome this shortcoming of PKI. In VANETs, trust management schemes have the potential to mitigate messages alteration and spoofing attacks, also to detect forged messages and intruder attackers [5,6]. Furthermore, trust schemes allow vehicles to communicate with unknown peers. Plenty of approaches has

been put forward to establish and maintain trust communications in VANETs [6–10]. In our previous work [11], recent approaches have been reviewed. The existing approaches have been classified into two categories: centralized-based and distributed-based. In centralized-based trust management approaches, the trust values are processed and stored in a central server or authority. However, given the large scale of VANETs in terms of long distances and the high number of vehicles, besides the time-sensitivity of the safety applications, the centralized-based approaches are ill-suited for VANETs. In decentralized-based trust management approaches [12], trust computing and storing are performed locally by the vehicles and/or the roadside Unit (RSU). However, due to the different conditions and capabilities to perceive and assess the target vehicles, trust values computed by only one vehicle cannot be always reliable.

1.2. Literature Review

The number of publications that have focused on providing trust management schemes for IoV has greatly increased in recent years [7–9]. In [10] a comprehensive review of existing trust management methods is presented. The authors present an adversary-oriented survey where they have discussed the attacks that can bypass the trust management schemes and how to mitigate them. The authors in [8] discuss trust management in vehicular environments from different perspectives. Additionally, they have classified the existing trust management schemes into two categories. The first category is the artificial intelligence-based approaches based on clustering, reinforcement learning, fuzzy logic, and game theory techniques. The second category is emerging technologies-based approaches based on Cloud, Fog, Edge, Blockchain, and Software-defined networking (SDN). These new technologies improve classic trust computing approaches in terms of accuracy, aggregation, and sharing of the trust values.

Moreover, classic approaches (not based on emerging technologies) often rely on recommendation systems to enhance the accuracy and consistency of the trust metrics like in [13,14]. In [13] the authors designed a decentralized trust management framework based on fuzzy logic-based and Q-learning approaches. Each vehicle assesses the trustworthiness of its 1-hop neighbors based on a direct and an indirect trust. The direct trust is evaluated based on the direct experience with the target vehicle, particularly, the forwarding ratio, the ratio of legitimate forwarded messages, and the percentage of the detected incidents reported by the target vehicle. The indirect trust is evaluated based on recommendations received from other vehicles about the target vehicle. Xia et al. proposed in [14] a lightweight trust-aware multicast routing protocol using a feedback mechanism and Markov prediction algorithm. They combined subjective trust with recommendations to create an attack-resistant trust inference model for VANETs. In both approaches, the accuracy of the trust values depends on the number of recommendations received from other vehicles, which makes it non-suitable for rural scenarios which are usually not dense. In [15], a distributed trust computing framework named EDTCF to compute the trust metrics of the vehicles in a VANET. Each vehicle solely assesses the trust metrics of its neighbors based on the credibility of their broadcast messages. However, it does not share them with other vehicles, in so doing, each vehicle in the network will have many trust metrics at the same time. Thereby, the trust metrics of each vehicle may be non-consistent with each other. Like many classic models, EDTCP [15] suffer also from the cold start problem because the trust metrics are not shared between the vehicles mainly in distributed approaches.

Blockchain is a modern technology that appeared in the last decade and it has grown quickly to provide a lot of promises. It is a distributed ledger that provides a secure and transparent way to create, verify and record any type of transaction [16]. It solves the issues related to trust in any particular type of communication. Blockchain technology comprises several technologies, such as cryptographic hash, digital signatures, and consensus algorithms. Blockchain is an emerging technology used today to improve the trust management scheme [8]. Although tremendous efforts have been made to provide trust management schemes for VANETS, the research on blockchain-based trust management schemes is still

limited. In [16] a blockchain-based trust management scheme is proposed for vehicular networks. It is a decentralized and hybrid scheme where each vehicle calculates first a trust rating about the received messages, then uploads the results to the nearest RSU. Based on the trust rating received from the vehicles, each RSU calculates the trust values of its target vehicles, packs these data into a block, then solves the Proof-of-Work (PoW) and Proof-of-Stack (PoS) consensus to add the block into the blockchain. In the proposed scheme, the trust evaluation relies only on the geographic position of the vehicle, if it is near to the event location then it is trustworthy. Nevertheless, the location is not sufficient to decide on the credibility of the received data and the trustworthiness of the originator vehicle because the message may be generated by a malicious vehicle located near the event. In [17] Lu et al. proposed a Blockchain-based Anonymous Reputation System (BARS). To estimate the reputation of a vehicle both direct historical interactions and indirect opinions about the target vehicle are considered. In their entity-centric framework, three different blockchains are used. The first one is used to store valid certificates, the second one is used to store the revoked certificates and the third one is used to store the messages disseminated in the network. To authenticate the vehicles, their certificates are searched in the first and second blockchains. If the certificate is present in the former blockchain and absent in the latter one, the vehicle is authentic. Further, RSUs will act as validators of this framework using Proof of Work consensus. BARS is based also on a Law enforcement authority (LEA) responsible for recording the pairs of public keys and real identities of the vehicles. LEA is a centralized authority which leads to a lack of scalability. Kchaou et al. [18] proposed a distributed trust management scheme for VANETs named DTCMV based on blockchain. The scheme consists of three steps: the transmission of messages, the creation of blocks, and the validation of the block. In their scheme, the RSUs will serve as miners performing a PoW consensus. However, the authors did not explain the trust metrics calculation process and which factors are considered to assess the trustworthiness of the vehicle.

In [19] the reputation score is calculated using an indirect reciprocity principle, where each vehicle calculates its reputation score according to its cooperation with other peers. The limitation of this approach is the fact that each vehicle updates its reputation value. Javaid et al. [20], propose a blockchain-based protocol for computing trust management on the Internet of Vehicles using smart contracts, certificates, and a dynamic proof-of-work consensus algorithm. The proposed entity-centric protocol consists of two phases: a setup phase and a data transfer phase. In the first phase, the vehicles register to become trusted users and get a blockchain account and unique ID. In the second phase, secure communication links between trusted vehicles are established. However, the proposed protocol revokes malicious vehicles definitively from the network, also it does not assess the credibility of the message itself, so a trust vehicle can broadcast forged messages without being detected. A Blockchain-based collaborative intrusion detection networks (CIDNs) framework is proposed in [21]. It is a data-centric framework that enables the vehicles to form a consortium chain by verifying the received data messages and the alarm rankings, which uses blockchains. The RSUs serve as a validator, responsible for detecting advanced malicious vehicles. The RSUs compute the trust values by comparing the received messages with the expected content. Unfortunately, this framework does not detect malicious vehicles in real-time. In [22], the author proposed a trust management system using a multi-criteria decision-making model, where each vehicle assesses the reliability of the messages and computes the trust value of the sender. Afterward, each vehicle transmits the computed trust values to the nearest RSU, which in turn calculates the trust values using a multi-criteria decision-making model. After that, the RSU creates a block and tries to solve the consensus to add the newly created block to the blockchain. The limitation of this system is the use of the RSU as miners, which is not practical because of their expensive deployment cost.

1.3. Contribution

One of the key features of Blockchain is the fact that is easily applied to distributed systems where multiple entities maintain the same information without requiring a central authority. It has a massive potential to improve the performance of distributed systems. Particularly, it allows mitigating many types of attacks emanating from systems governed by a central authority and reduces communication overhead with central servers and authorities [8,23,24].

In the same context, a new Blockchain-based Trust Management framework baptized BC-TMF is proposed in the current study. It aims to compute and share the trust metrics of the vehicles in a distributed fashion. It is a context-dependent framework where the trust metric of the vehicles depends on the trustworthiness of their alert messages disseminated in the network. These alerts report hazardous events on the road. The proposed BC-TMF is built on Blockchain technology to take benefit of its features. On one hand, the distribution and availability facilitate the update of the trust metrics of the vehicles. On the other hand, the integrity that blockchain technology offers allows saving the computed trust metrics in a dependable ledger. In so doing, the proposed BC-TMF will avoid bad-mouthing attacks. Moreover, the availability of the Blockchain allows sharing the computed trust metrics between the vehicles, which mitigates the cold start problem when a vehicle enters a new network and has to timely authenticate alerts received from unknown peers. To enhance the distribution of the proposed framework, the miners will be a set of vehicles selected on the road instead of RSUs, which seems more efficient in rural environments and in the earlier stages of VANETs deployment where few RSUs are deployed in the network due to their high costs. Unlike existing approaches, where the trust metrics of the vehicles are computed in the RSU, in the proposed approach the trust metrics of the vehicles are computed in a fully distributed manner in the vehicles due to their high potential to perceive the events on the roads. Furthermore, contrary to existing frameworks, in the proposed approach, the trust metric is inferred based only on direct observations without considering recommendations from other vehicles.

The remainder of the paper is organized as follows. Section 2 presents the different details of the proposed BC-TMF. Section 3 is devoted to the performance evaluation and discussion. Finally, Section 4 concludes the paper and presents some future directions.

2. Methods

2.1. Overview of BC-TMF

The proposed framework aims to compute, store, and publish the trust metrics of the vehicles based on the credibility of their messages. Each vehicle will have only one global trust metric at a given time, shared between all the vehicles in the network. This will avoid the cold start problem and expedite the authentication of the reported events in real-time. Moreover, due to the integrity provided by blockchain technology, the trust metrics will be stored in a secure and reachable ledger.

In the context of safety applications in VANETs, each vehicle traveling on the road disseminate and forward periodic alerts about the encountered events like traffic crowds and accidents. These alerts will fluidize the traffic, make far vehicles aware of the road state and allow drivers to make timely decisions [2]. Nevertheless, some malicious vehicles may deliberately disseminate forged alerts about non-existing events or alter the content of alerts reporting the real events. In the current model, two types of vehicles are distinguished: malicious and non-malicious vehicles. The probability for vehicle v to be malicious is denoted PM(v). It is a value in the interval (0–1). All malicious vehicles have their PM strictly greater than 0: PM > 0. However, non-malicious vehicles have PM = 0. On one hand, if a malicious vehicle perceives an event on the road, it may alter the alerts reporting this real event with probability PM. Similarly, it may alter all the forwarded alerts with the same probability. On the other hand, a malicious vehicle may forge alerts about non-real events with probability PM.

The primary objective of the proposed framework is to compute the trust metrics of malicious and non-malicious vehicles. These trust metrics should reflect the real behavior of the vehicles. A malicious vehicle should never reach the highest trust level. In the same way, a non-malicious vehicle must have high trust metrics. In the proposed BC-TMF, two trust metric types are considered: a local trust metric and a global trust metric. The local trust metric is a subjective value computed by a monitor about a target vehicle based on its own experience with it. The global trust metric is an aggregated trust metric computed by the miner about each vehicle based on local trust metrics received from the monitors, it is stored in the blockchain.

The local trust metric of vehicle v, calculated by vehicle m, noted $Tm(v)$ is defined as a value in the interval (0–1). The local trust metric of v relies on the credibility of its messages sent to m. Besides, the trust metric evaluation is based on direct experiences with target vehicles. Each vehicle asses the trust metric of its neighbors based on the received alerts subjectively. It is worth noting that the local trust metrics are asymmetric: $Tm(v)$ is different and independent from $Tv(m)$.

In the proposed BC-TMF, blockchain technology is used to store the trust values of vehicles and make them reachable by all other vehicles in the network. Contrary to many existing frameworks where the RSUs serve as miners [18,22], in the proposed approach, a public blockchain is considered, where the miners are a set of vehicles elected randomly on the road. Actually, it is not very practical to build the framework on RSUs because it is very expensive to deploy many RSUs along the roads, particularly in rural scenarios. More importantly, the RSUs are vulnerable to physical attacks. As illustrated in Figure 1, the proposed BC-TMF goes through three main phases.

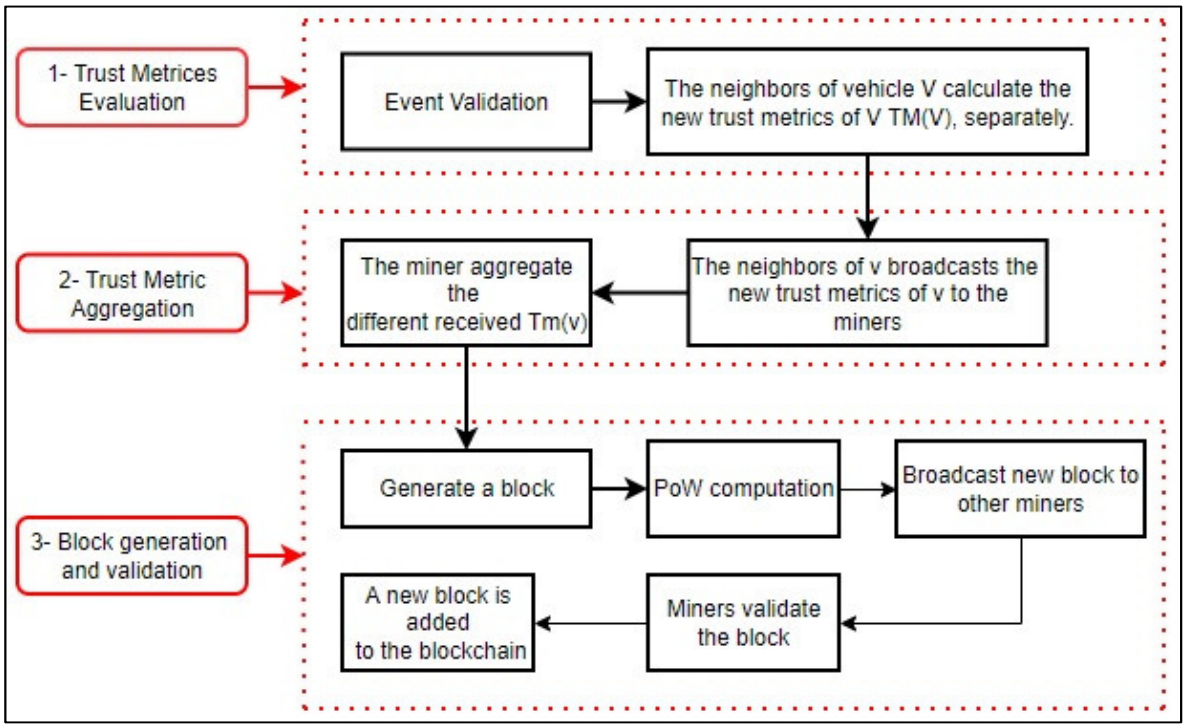

Figure 1. The different phases of the proposed BC-TMF.

In the first phase, the local trust metrics of the vehicles are evaluated by their neighbors. In the second phase, the obtained local trust metrics are shared with the miners, which are responsible for the aggregation to compute a global trust metric for each vehicle in its neighborhood. In the third phase, the miners generate and validate their blocks, then compete to add them to the blockchain. The following sections describe the different phases of the framework.

2.2. Trust Metric Evaluation

The first phase of the proposed framework is to compute local trust metrics for the vehicles saved locally in their monitors, these local trust metrics will be aggregated later into global trust metrics in the miners. To compute the local trust metrics of the vehicles in the proposed BC-TMF, protocol EDTCP proposed in [15] is used. It is a hybrid approach aiming to compute the trust metrics of the vehicles, and evaluate the reliability of the disseminated alerts. Each vehicle monitors its neighbors and updates their trust metrics accordingly. The local trust metric update is event-driven. Upon perceiving an event on the road, vehicle v periodically broadcasts alert messages that report the event in its neighborhood. Once vehicle m receives the alert message from vehicle v, it should, first, check the credibility of the event reported in the alerts. To this end, a tier-based data reliability assessment technique is used [11]. It aims to decide if the detected event is reliable or not. Afterward, vehicle m updates the trust metric of vehicle v accordingly: if the event is reliable, $Tm(v)$ increases by 0.1 otherwise it decreases by 0.1:

$$Tm(v) = \begin{cases} +0.1, \text{if the event is reliable} \\ -0.1, \text{otherwise} \end{cases} \quad (1)$$

Initially, vehicle v has $Tm(v) = 0.1$ in vehicle m. Each neighbor m of vehicle v separately computes a trust metric for it, this local trust metric will be stored in a local database named List_Local_Tm(m). Consequently, vehicle v will have more than one local trust metric evaluated by different vehicles at the same time. Besides, vehicle v has to cooperate with its neighbors along its journey to increase its trust metrics stored locally in its neighbors. Let us note that all the vehicles mutually and independently monitor and compute trust metrics about each other. The next phase of BC-TMF is trust metrics aggregation.

2.3. Trust Metrics Aggregation

It is not practical for each vehicle to have different trust metrics in different vehicles. This may lead to a non-consistency problem between different calculated local trust metrics because each monitor will update the trust metrics of its neighbors based on its own experience with them. Besides, this may facilitate the badmouthing attack where a vehicle overestimates or underestimates its neighbors. Therefore, each monitor m has to forward the trust metrics list of its monitored vehicles to the miners. As aforesaid, the miners are random vehicles selected on the road. Hence, periodically, each SendTransTimer period, monitor m transmits a copy of its database List_Local_Tm(m) to the nearest miner in its neighborhood. Afterward, the miner proceeds to the aggregation of the received local trust metrics into global trust metrics. The aggregation is a time-driven process. Periodically, the miner will go through the received databases and looks for trust metrics of each vehicle v, then it computes one global trust metric denoted $TM(v)$ for v as shown in Equation (2):

$$TM(v) = \frac{\left(TM(v)_{old} + \sum_{i=1}^{N} T_i(v)\right)}{N+1} \quad (2)$$

where N is the number of monitors of vehicle v and $TMold(v)$ is the old trust metric of v stored in the blockchain. Initially $TMold(v)$ is equal to 0.1. Similarly to the local trust metric, the aggregated TM is a value in the interval (0–1). If vehicle v gets $TM(v) = 1$, it is considered trustworthy. After aggregation, the miner records the new aggregated trust metric $TM(v)$ in a local database, to be packed later in a block. Afterward, the miner proceeds to the subsequent task which is the block generation and validation.

2.4. Block Generation and Validation

The main objective of the proposed blockchain-based framework is to compute for each vehicle only one accurate and up-to-date trust metric that reflects its current behavior and

is available to all the other vehicles in the network. Blockchain technology is a distributed ledger, it is exploited to record the trust metrics of the vehicles in it. Sharing the global trust metrics with all the vehicles in the network allows coping with the cold-start problem that arises when a new vehicle enters a new region without prior experience with the encountered vehicles. Consequently, it has to cooperate efficiently and for a long period to build a high trust metric. Unlike existing frameworks where the blocks are created continuously on every update of the local trust metrics [18,22], in the proposed BC-TMF the creation of the blocks is performed periodically every addBlockTimer period. The periodicity of the block generation has two main benefits. First, it allows the miners to collect more accurate trust metrics from the monitors because the more the monitor has a long experience with its neighbors, the more accurate the computed local trust metrics will be. Secondly, it economizes the computation resources of the miners upon the PoW calculation. More importantly, adding the blocks periodically to the Blockchain will reduce the number of blocks, consequently, it expedites the retrieve of the global trust metrics from the ledger.

Each block consists of two compartments: the header and the body. The block header contains the block version that indicates the position of this block in the blockchain, the previous block's hash, Merkle tree root hash, which is a hash value of all the trust metrics included in the current block, the nonce used in the PoW, as well as the timestamp, which is the block generation time. Besides, the header contains difficulty metric D used to solve the Proof-of-Work [23]. The body contains the list of trust metrics of the vehicles aggregated as explained in the previous section. The structure of the block is shown in Figure 2.

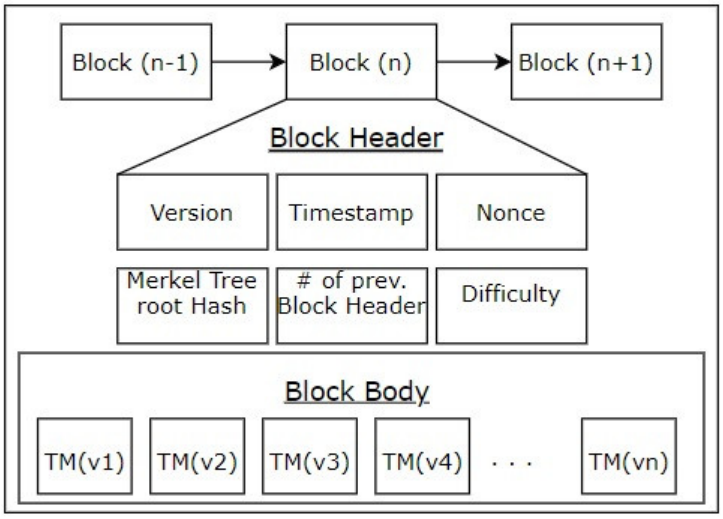

Figure 2. The structure of the block.

Since many miners may try to add their blocks at the same time to the blockchain, a technique to select which miner to add its block first must be expected. In Blockchain technology, the concept of PoW consensus is used to solve this problem [18]. The miner has to continuously hash the header of its block until getting a hash value H smaller than or equal to a threshold value named difficulty D: $H \leq D$, also D zeros must be padded in front of the hash. D is a hash threshold that can be tuned to control the difficulty of the PoW. The miner that solves the PoW first, adds its block first, to the blockchain. After the PoW computation, the miner signs the block using its private key and sends it to other miners. Other miners should validate the received block by checking if it fulfills the following condition $H \leq D$, also they check the signature of the miner originator of the block, if both conditions are fulfilled, the miners add the received block to their copies of the blockchain, also forward it to other vehicles in the network. Hereafter, every vehicle that receives an alert from a new vehicle can retrieve its global trust metric from the blockchain. The PoW Algorithm 1 is described below:

Algorithm 1: PoW

1. Start
2. Input D, nonce
3. Do
4. Calculate the block hash H
5. Increment nonce
6. While (H > D and (the number of zeros padded in the front of the block hash < D))
7. End

3. Simulation Results and Discussion

In this section, the performance of the proposed BC-TMF are investigated using simulations. We are interested in the efficiency of the framework and the consistency of its results.

3.1. Simulation Setup

An extensive set of simulations has been conducted using the vehicular network simulator Veins [25], conjointly with the road traffic simulator SUMO [26]. Both the proposed framework and EDTCP protocol [15] used in the trust metrics evaluation phase are implemented in Veins. The considered network map is shown in Figure 3.

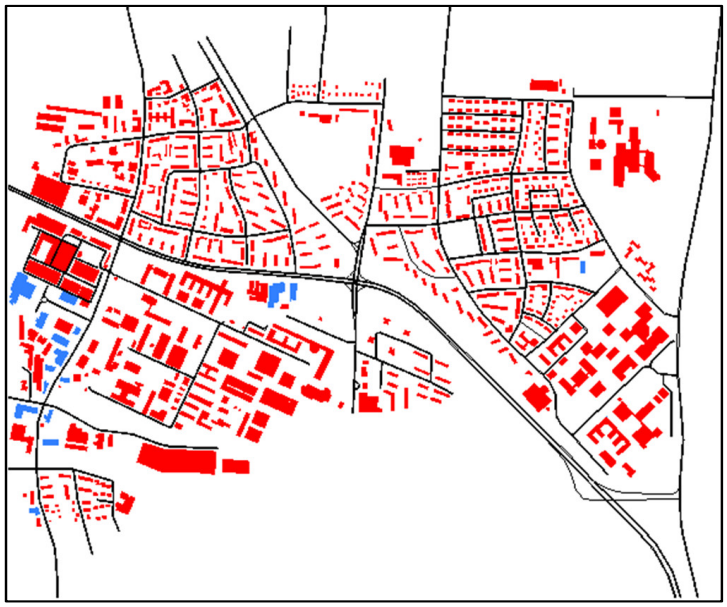

Figure 3. Network Map.

The vehicles enter the network from one entry point with a rate of 2 vehicles/s. All vehicles speed up to a maximum of 50 km/h. In the considered scenario, the vehicles will encounter congestion events on the road, in this case, they slow down or/and stop for a while till the road becomes unblocked. These congestions are reported in alerts broadcast periodically (every 1 s) in the network. Table 1. summarizes the simulation parameters.

3.2. Results

Initially, we are interested in the efficiency of the proposed BC-TMF. Figures 4 and 5 show the average global trust metric (TM) of non-malicious vehicles having PM = 0 as a function of addBlockTimer for an entry rate of 2 vehicles/s and 4 vehicles/s, respectively.

Table 1. Simulation Set up.

Parameters	Values
Simulation period	1200 s
MAC/PHY	IEEE 802.11p
PM	(0–1)
Number of Miners	50% of the total number of vehicles
Percentage of Malicious vehicles	15%
Maximum speed	50, 65 km/h
Transmission range	350 m
SendTransTimer	50, 100 s
AddBlockTimer	(150–350) s

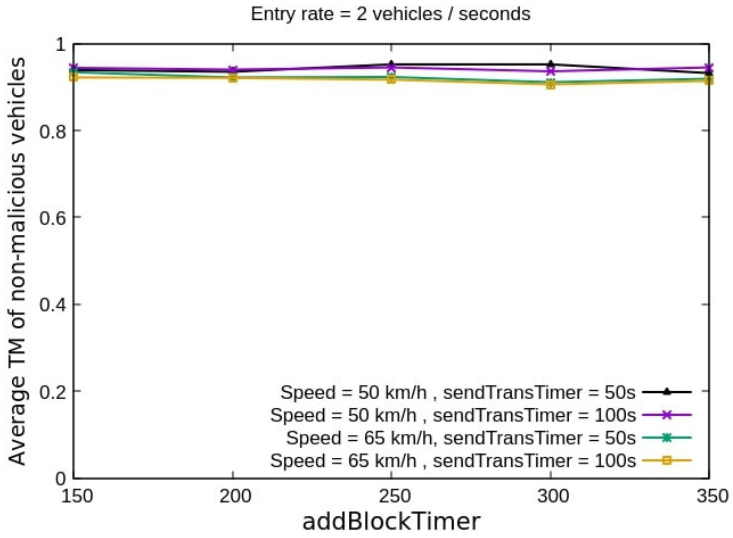

Figure 4. The average trust metric of non-malicious vehicles as a function of addBlockTimer(s), entry rate = 2 vehicles/s.

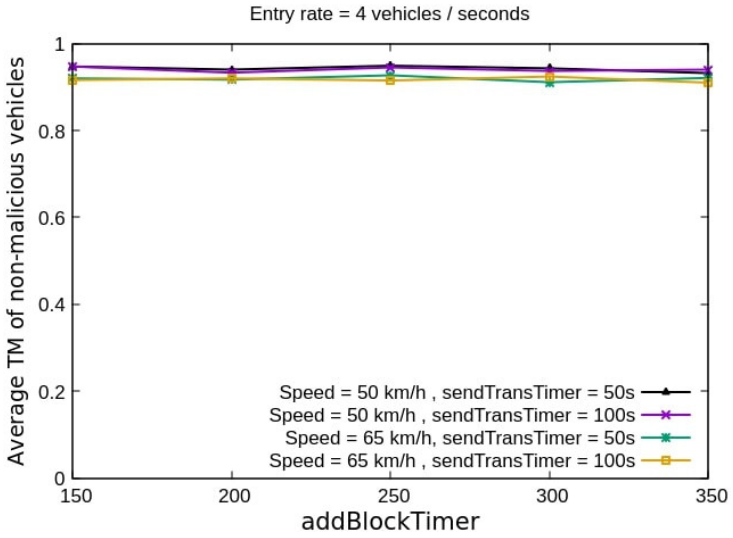

Figure 5. The average trust metric of non-malicious vehicles as a function of addBlockTimer(s), entry rate = 4 vehicles/s.

It is obvious from Figures 4 and 5 that the average trust metric of non-malicious vehicles varies between 0.92 and 0.95 for both scenarios. Besides, it is obvious that it is slightly higher in the case of 50 km/h compared to the case of 65 km/h. This result is due to the fact that the high speed causes an intermittent connection between the vehicles,

consequently, the monitors may not receive a sufficient number of alert messages from the target vehicles to increase their trust metrics. Furthermore, few vehicles did not reach a trust metric equal to 1 and this is maybe because they have recently joined the network, so they need more time to coordinate with their monitors and increase their trust metrics. Furthermore, we observe that addBlockTimer does not affect the trust metrics of non-malicious.

Figures 6 and 7 portray the average trust metric of malicious vehicles as a function of the PM for different speeds and addBlockTimer values. On one hand, we notice that in all the considered scenarios the average trust metric decreases inversely to the PM. This is an expected result because the PM impacts the behavior of the vehicles in terms of the percentage of non-authentic messages disseminated in the network. The more PM increases, the more the authenticity of the detected events is altered. Consequently, when the percentage of non-authentic messages increases, the trust metric of the source vehicle decreases. More importantly, although only 10% of the messages sent by malicious vehicles are non-authentic in the case of PM = 0.1, it is obvious that their average trust metric does not exceed 0.3 for all scenarios, which confirms the efficiency of the proposed BC-TMF in calculating the trust metrics of the vehicles. It does not allow malicious vehicles to be trustworthy (TM = 1).

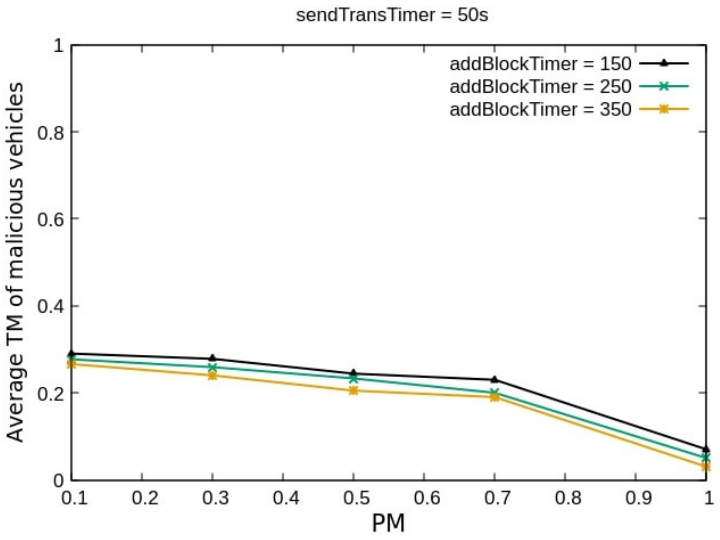

Figure 6. The average trust metric of malicious vehicles vs. PM, sendTransTimer = 50 s.

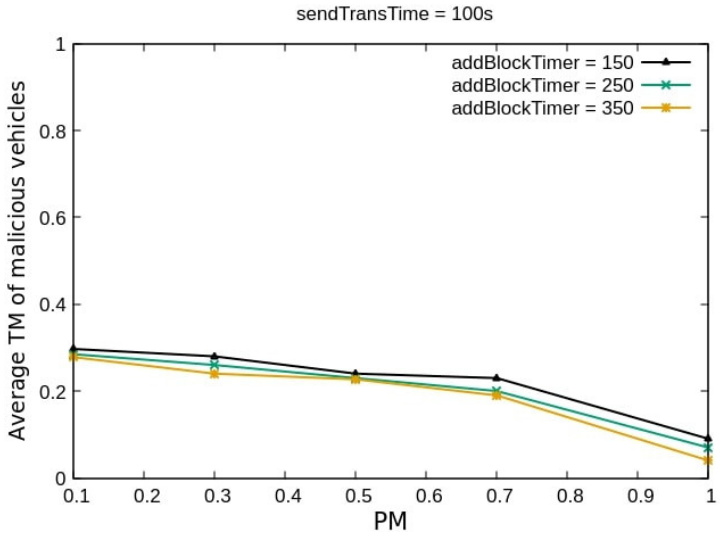

Figure 7. The average trust metric of malicious vehicles vs. PM, sendTransTimer = 100 s.

Besides Figures 6 and 7 point out the importance of the sendTansTimer parameter on the convergence of the global TM of the malicious vehicles to lower values. Particularly, we remark that the average TM slightly decreases when sendTransTimer increases mainly for high PM values (>0.5). We notice in Figures 6 and 7 that the average TM slightly decreases also when addBlockTimer increases for high PM values. It passes from 0.25 in case of addBlockTimer = 150 s to 0.19 in case of addBlockTimer = 350 s, for PM = 0.7 and sendTransTimer = 50 s. This result provides enough evidence that addBlockTimer and SendTransTimer must be accurately tuned to obtain exact trust metrics and ensure that the global trust metrics of malicious vehicles converge rapidly to lower values.

To emphasize the accuracy of the proposed BC-TMF compared to existing frameworks, we portray in Figure 8, the average global trust metric of malicious vehicles as a function of the percentage of malicious vehicles in the network. The average trust metric computed by the proposed BC-TMF is compared to that computed by NBC-TMV framework proposed in [22]. For both scenarios PM is equal to 0.7. As depicted in Figure 8 for both frameworks the average trust metric of malicious vehicles decreases when the percentage of malicious vehicles increases. This result is expected since the percentage of non-reliable alert messages increases. Nevertheless, the proposed BC-TMF outperforms NBC-TMV mainly in the case of a high percentage of malicious vehicles. It is obvious that it is about 0.1 for BC-TMF for 25% of malicious vehicles while it is 0.2 for NBC-TMV. This result is due to the aggregation process in BC-TMF that avoids overestimating the trust metric of malicious vehicles. Another reason behind the outperformance of the proposed BC-TMF is the update of the trust metrics performed in miner vehicles compared to the NBC-TMV framework where it is performed in the RSUs. However, RSUs are not deployed everywhere on the road, so the aggregation process and block creation may be delayed in NBC-TMV, consequently, the trust metrics are not updated timely.

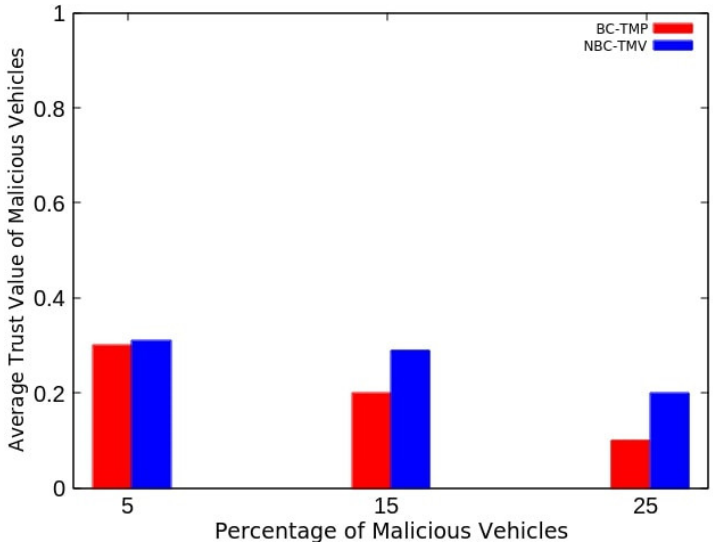

Figure 8. The average trust metric of malicious vehicles vs. the percentage of malicious vehicles in the network.

Now we focus on the consistency of the computed trust metrics stored in the blockchain. To this end, Figure 9 presents the variance of the global trust metrics for non-malicious vehicles (PM = 0) computed as in Equation (3) below:

$$variance_{nonmalicious} = \frac{\sum_{i=1}^{N}(TM(i) - Avg)^2}{N} \quad (3)$$

where N is the total number of non-malicious vehicles, Avg is the average global trust metric for non-malicious vehicles, and $TM(i)$ is the global trust metric of vehicle i.

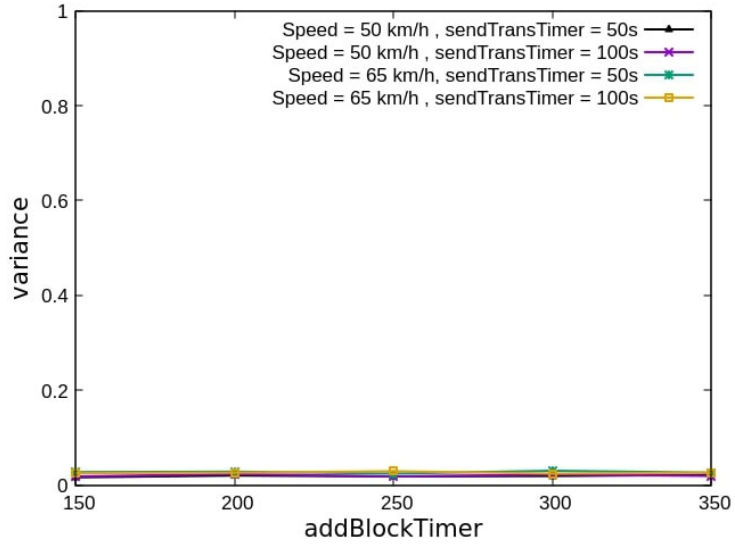

Figure 9. The variance of the trust metrics for non-malicious vehicles vs. addblockTimer (s).

Figure 9, portrays the variance of the global trust metric for the non-malicious vehicles as a function of addBlockTimer for different speed and sendTransTimer values.

It is obvious from Figure 9 that the variance does not exceed 0.01 in all scenarios, which confirms the consistency of the obtained trust metrics presented in Figures 4 and 5 shown above.

Similarly, Figure 10 portrays the variance of the global trust metric of malicious vehicles as a function of PM, it is computed as in Equation (4) below:

$$variance_{Malicious} = \frac{\sum_{i=1}^{N}(TM(i) - AvgM)^2}{M} \quad (4)$$

where M is the total number of malicious vehicles, $AvgM$ is the average global trust metric for malicious vehicles, and $TM(i)$ is the global trust metric of the malicious vehicle i.

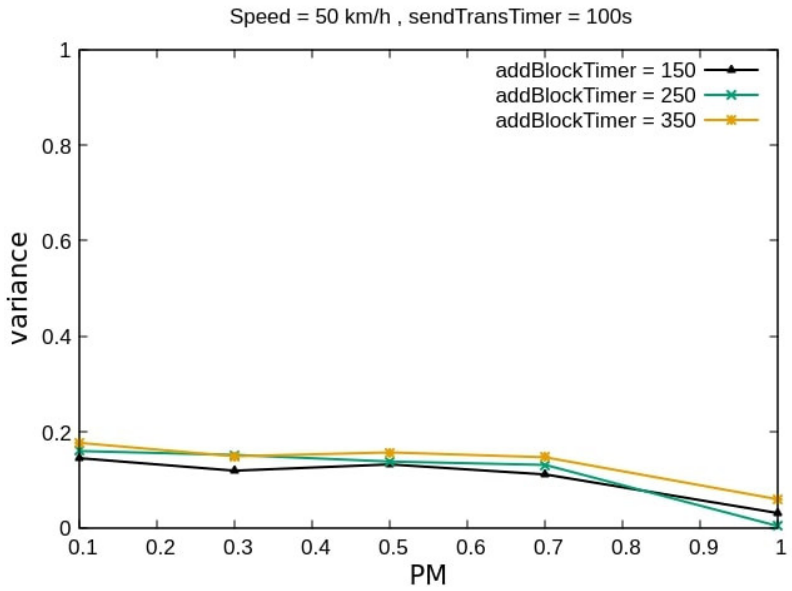

Figure 10. The variance of the trust metrics for malicious vehicles vs. PM, sendTransTimer = 100 s.

We observe in Figure 10 that the variance does not exceed 0.18. It passes from around 0.18 for PM = 0.1 to around 0 for PM = 1 and addBlockTimer = 350 s. Additionally, it is obvious that addBlockTimer slightly affects the variance. It passes from 0.18 for addBlockTimer

equal to 350 s to around 0.15 in the case of PM = 0.1. The value of addBlockTimer = 150 s seems to be the most appropriate to get lower trust metrics for malicious vehicles. This result confirms once again, the excellent capability of BC-TMF to compute exact and consistent trust metrics for malicious vehicles.

Let us now consider Figure 11 where we portray the fraction of malicious vehicles that reach a trust metric TM = 0.8 as a function of PM denoted $PS(8)$ and computed as shown in Equation (5) below:

$$PS(8) = \frac{\text{Number of vehicles (PM } > 0 \text{ and TM} \geq 0.8)}{\text{Total number of malicious vehicles}} \quad (5)$$

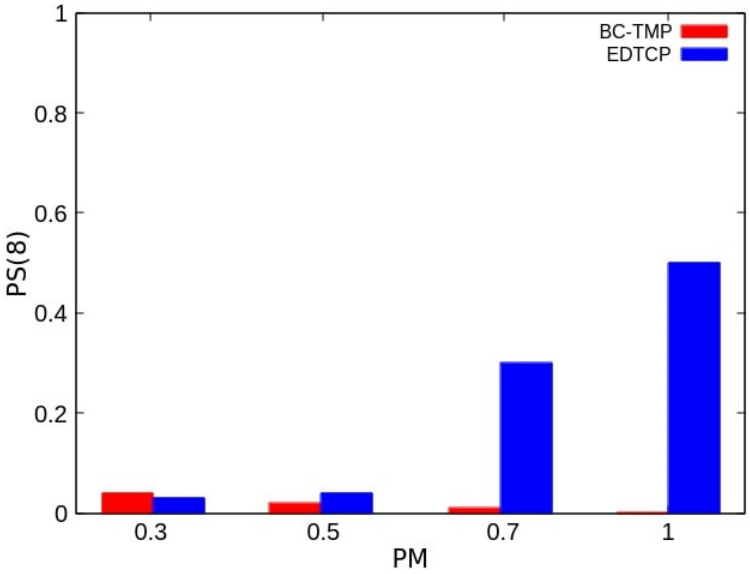

Figure 11. The fraction of malicious vehicles that reach trust metric greater than or equal to 0.8 vs. PM.

This metric ($PS(8)$) is interesting because it will be considered later as selection criteria of the miners. Let us recall that in the current work, the miners are a randomly selected set of vehicles. In Figure 11, the results computed using BC-TMF are compared to the results computed using EDTCP [15] where each vehicle has many local trust metrics saved in its monitors, no aggregation or sharing techniques are used.

We remark in Figure 11 the huge difference between the results obtained for the proposed BC-TMF compared to EDTCP mainly for PM $\geq$ 0.7. Particularly, $PS(8)$ is less than 0.02 for BC-TMF. However, it goes to 0.5 for EDTCP, which means that for EDTCP about half of the malicious vehicles having (PM = 1) reaches a trust metric higher than 0.8. Again BC-TMF computes accurate trust metrics, particularly for malicious vehicles.

3.3. Discussion

In this section, the most important features of the proposed framework are discussed and compared to the existing approaches. First of all, unlike classic approaches [13,14] where trust is based on direct observations and recommendations, the framework proposed in this study does not rely on a recommendation system, yet it allows calculating accurate trust metrics as shown in the previous section and this is due to the accurate tier-based event authentication approach used in the first phase of the framework. The proposed framework solves also, the cold-start problem which is a limitation of the classic approaches such as [14,15]. It arises when a vehicle enters a new network and has to cooperate with unknown vehicles. This problem is solved using Blockchain which provides a shared ledger used to securely store and share the global trust metrics of the vehicles. Additionally, unlike the existing blockchain-based frameworks [16–18,22], the proposed framework does not rely on RSUs to play the role of miners, instead, a set of vehicles randomly selected

in the network play the role of miners. Only the vehicles are involved in the different phases of the framework and this is for many reasons. First, the vehicles are supplied with sensors and many potential capabilities to adequately monitor their surroundings and perceive hazardous events timely and more accurately than the RSUs. Secondly, the RSUs are very expensive to deploy everywhere on the road mainly in the earlier deployment stages of vehicular networks. More importantly, as explained in the previous section, assigning the local trust metric assessment to the RSUs may delay the global trust metrics convergence and share. Hence, in the proposed framework, the local trust metric calculation is performed by the vehicles then shared with the miners. It is worth mentioning here that it will not cause a problem in terms of resources because the vehicles are equipped with onboard units characterized by sustainable computational resources. Yet, it expedites global trust metrics calculation in the miners since they will receive from the vehicles only the local trust metrics unlike the approaches proposed in [16–18] where the vehicles share with the miners the data messages, then it is up to these latter to assess and calculate the trust metrics. Moreover, in the proposed approach, global trust metrics update is time-driven contrary to existing approaches [16–18,22] where it is performed each time the vehicle transmits a data message. This feature allows the monitors to assess more accurately the local trust metrics of their neighbors, consequently, the global trust metrics converge rapidly to the adequate value mainly for malicious vehicles as explained in the previous section.

4. Conclusions

In this paper, a decentralized blockchain-based trust management framework is proposed. It aims to compute for each vehicle a global and accurate trust metric stored in the blockchain. The proposed BC-TMF is built upon three phases: trust metrics evaluation, trust metrics aggregation, and blocks generation and validation. Each vehicle assesses the trust metric of its neighbors from which it receives alerts. The trust metric calculation relies on the authenticity of the received messages. Then the miners aggregate the trust metrics received from other vehicles to compute an aggregated trust metric for each vehicle. The aggregated trust metric will be packed in a block signed by the miner, then added to the blockchain after solving the PoW. Sharing the aggregated trust metrics in a distributed ledger will solve the cold-start problem.

A set of simulations has been conducted to evaluate the efficiency and accuracy of the proposed BC-TMF in terms of the average trust metric for both malicious and non-malicious vehicles and the variance of the trust metrics as a function of many parameters. The obtained results show that the proposed BC-TMF has excellent capability in computing accurate trust metrics for malicious and non-malicious vehicles. The proposed BC-TMF outperforms not only classic schemes but also existing blockchain-based frameworks. In future work, it is expect to refine the miners' selection process. Besides, it is expected to conduct further simulations to tune the different parameters of the proposed framework.

Author Contributions: Methodology, T.G.; Resources, O.A.; Writing—review & editing, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the University of Jeddah, Saudi-Arabia, grant number UJ-20–124-DR.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Sharma, S.; Kaushik, B. A survey on internet of vehicles: Applications, security issues & solutions. *Veh. Commun.* **2019**, *20*, 100182. [CrossRef]
2. Hamdi, M.M.; Audah, L.; Rashid, S.A.; Mohammed, A.H.; Alani, S.; Mustafa, A.S. A review of applications, characteristics and challenges in vehicular ad hoc networks (vanets). In Proceedings of the International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 26–28 June 2020; pp. 1–7. [CrossRef]
3. Al-Ani, R.; Zhou, B.; Shi, Q.; Sagheer, A. A survey on secure safety applications in vanet. In Proceedings of the 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Exeter, UK, 28–30 June 2018; pp. 1485–1490. [CrossRef]
4. IEEE. *IEEE Standard for Wireless Access in Vehicular Environments—Security Services for Applications and Management Messages*; IEEE: Piscataway, NJ, USA, 2016; pp. 1–240. [CrossRef]
5. Rehman, A.; Hassan, M.F.; Yew, K.H.; Paputungan, I.; Tran, D.C. State-of-the-art IoV trust management a meta-synthesis systematic literature review (SLR). *PeerJ Comput. Sci.* **2020**, *6*, e334. [CrossRef] [PubMed]
6. Ahmad, F.; Adnane, A.; Kerrache, C.A.; Franqueira, V.N.L.; Kurugollu, F. Trust Management in Vehicular Ad-Hoc Networks and Internet-of-Vehicles: Current Trends and Future Research Directions. In *Global Advancements in Connected and Intelligent Mobility: Emerging Research and Opportunities*; IGI Global: Hershey, PA, USA, 2020; pp. 135–165, Chapter 4. Available online: https://www.igi-global.com/chapter/trust-management-in-vehicular-ad-hoc-networks-and-internet-of-vehicles/232026 (accessed on 23 January 2022).
7. Hussain, R.; Lee, J.; Zeadally, S. Trust in VANET: A Survey of Current Solutions and Future Research Opportunities. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 2553–2571. [CrossRef]
8. Iqbal, R.; Butt, T.A.; Afzaal, M.; Salah, K. Trust management in social Internet of vehicles: Factors, challenges, blockchain, and fog solutions. *Int. J. Distrib. Sens. Netw.* **2019**, *15*. [CrossRef]
9. Souissi, I.; Ben Azzouna, N.; Berradia, T. Trust management in vehicular ad hoc networks: A survey. *Int. J. Ad Hoc Ubiquitous Comput.* **2019**, *31*, 230. [CrossRef]
10. Kerrache, C.A.; Calafate, C.T.; Cano, J.-C.; Lagraa, N.; Manzoni, P. Trust Management for Vehicular Networks: An Adversary-Oriented Overview. *IEEE Access* **2016**, *4*, 9293–9307. [CrossRef]
11. Alboqomi, O.; Gazdar, T.; Munshi, A. A new blockchain-based trust management protocol for vehicular ad hoc networks. In Proceedings of the 4th International Conference on Future Networks and Distributed Systems ICFNDS, St. Petersburg, Russia, 26–27 November 2020; pp. 36:1–36:5. [CrossRef]
12. Guleng, S.; Wu, C.; Chen, X.; Wang, X.; Yoshinaga, T.; Ji, Y. Decentralized Trust Evaluation in Vehicular Internet of Things. *IEEE Access* **2019**, *7*, 15980–15988. [CrossRef]
13. Mahmood, A.; Butler, B.; Zhang, W.E.; Sheng, Q.Z.; Siddiqui, S.A. A hybrid trust management heuristic for vanets. In Proceedings of the 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Kyoto, Japan, 11–15 March 2019; pp. 748–752. [CrossRef]
14. Xia, H.; Zhang, S.-S.; Li, Y.; Pan, Z.-K.; Peng, X.; Cheng, X.-Z. An Attack-Resistant Trust Inference Model for Securing Routing in Vehicular Ad Hoc Networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 7108–7120. [CrossRef]
15. Gazdar, T.; Belghith, A.; Abutair, H. An Enhanced Distributed Trust Computing Protocol for VANETs. *IEEE Access* **2017**, *6*, 380–392. [CrossRef]
16. Yang, Z.; Yang, K.; Lei, L.; Zheng, K.; Leung, V.C.M. Blockchain-Based Decentralized Trust Management in Vehicular Networks. *IEEE Internet Things J.* **2018**, *6*, 1495–1505. [CrossRef]
17. Lu, Z.; Wang, Q.; Qu, G.; Liu, Z. Bars: A blockchain-based anonymous reputation system for trust management in vanets. In Proceedings of the 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), New York, NY, USA, 1–3 August 2018; pp. 98–103. [CrossRef]
18. Kchaou, A.; Abassi, R.; Guemara, S. Toward a distributed trust management scheme for vanet. In Proceedings of the 13th International Conference on Availability, Reliability and Security (ARES 2018), New York, NY, USA, 27–30 August 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 1–6. [CrossRef]
19. Dai, C.; Xiao, X.; Ding, Y.; Xiao, L.; Tang, Y.; Zhou, S. Learning based security for vanet with blockchain. In Proceedings of the 2018 IEEE International Conference on Communication Systems (ICCS), Chengdu, China, 19–21 December 2018; pp. 210–215. [CrossRef]
20. Javaid, U.; Aman, M.N.; Sikdar, B. A Scalable Protocol for Driving Trust Management in Internet of Vehicles With Blockchain. *IEEE Internet Things J.* **2020**, *7*, 11815–11829. [CrossRef]
21. Li, W.; Wang, Y.; Li, J.; Au, M.H. Toward a blockchain-based framework for challenge-based collaborative intrusion detection. *Int. J. Inf. Secur.* **2020**, *20*, 127–139. [CrossRef]
22. Pu, C. A novel blockchain-based trust management scheme for vehicular networks. In Proceedings of the 2021 Wireless Telecommunications Symposium (WTS), Virtual Event, CA, USA, 21–23 April 2021; pp. 1–6. [CrossRef]
23. Wang, C.; Cheng, X.; Li, J.; He, Y.; Xiao, K. A survey: Applications of blockchain in the Internet of vehicles. *EURASIP J. Wirel. Commun. Netw.* **2021**, *77*, 1–16. [CrossRef]

24. Mikavica, B.; Kostić-Ljubisavljević, A. Blockchain-based solutions for security, privacy, and trust management in vehicular networks: A survey. *J. Supercomput.* **2021**, *77*, 9520–9575. [CrossRef]
25. Sommer, C.; Eckhoff, D.; Brummer, A.; Buse, D.S.; Hagenauer, F.; Joerer, S.; Segata, M. Veins—The open source vehicular network simulation framework. In *Recent Advances in Network Simulation—The OMNeT++ Environment and Its Ecosystem*; Springer: Cham, Switzerland, 2019; pp. 215–252, Chapter 5. Available online: https://link.springer.com/book/10.1007/978-3-030-12842-5 (accessed on 20 November 2021).
26. Lopez, P.A.; Behrisch, M.; Bieker-Walz, L.; Erdmann, J.; Flotterod, Y.P.; Hilbrich, R.; Lucken, L.; Rummel, J.; Wagner, P.; Wiebner, E. Microscopic traffic simulation using SUMO. In Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2575–2582. [CrossRef]