2021

A secured message transmission protocol for vehicular ad hoc networks

A. F. M. Suaib Akhter

A. F. M. Shahen Shah

Mohiuddin Ahmed
Edith Cowan University

Nour Moustafa

Unal Çavuşoğlu

See next page for additional authors

10.32604/cmc.2021.015447
Suaib Akhter, A. F. M., Shahen Shah, A. F. M., Ahmed, M., Moustafa, N., Çavuşoğlu, U., & Zengin, A. (2021). A secured message transmission protocol for vehicular ad hoc networks. *Computers, Materials & Continua, 68*(1), 229-246. https://doi.org/10.32604/cmc.2021.015447

Computers, Materials & Continua
DOI:10.32604/cmc.2021.015447
Article

Tech Science Press

A Secured Message Transmission Protocol for Vehicular Ad Hoc Networks

A. F. M. Suaib Akhter[1,*], A. F. M. Shahen Shah[2], Mohiuddin Ahmed[3], Nour Moustafa[4], Unal Çavuşoğlu[1] and Ahmet Zengin[1]

[1]Department of Computer Engineering, Sakarya University, Sakarya, 54050, Turkey
[2]Department of Electrical and Electronics Engineering, Istanbul Gelisim University, Istanbul, Turkey
[3]School of Science, Edith Cowan University, Perth, WA6027, Australia
[4]School of Engineering and Information Technology, UNSW, Canberra BC, 2610, Australia
[*]Corresponding Author: A. F. M. Suaib Akhter. Email: suaib.akhter@ogr.sakarya.edu.tr
Received: 18 November 2020; Accepted: 21 December 2020

Abstract: Vehicular Ad hoc Networks (VANETs) become a very crucial addition in the Intelligent Transportation System (ITS). It is challenging for a VANET system to provide security services and parallelly maintain high throughput by utilizing limited resources. To overcome these challenges, we propose a blockchain-based Secured Cluster-based MAC (SCB-MAC) protocol. The nearby vehicles heading towards the same direction will form a cluster and each of the clusters has its blockchain to store and distribute the safety messages. The message which contains emergency information and requires Strict Delay Requirement (SDR) for transmission are called safety messages (SM). Cluster Members (CMs) sign SMs with their private keys while sending them to the blockchain to confirm authentication, integrity, and confidentiality of the message. A Certificate Authority (CA) is responsible for physical verification, key generation, and privacy preservation of the vehicles. We implemented a test scenario as proof of concept and tested the safety message transmission (SMT) protocol in a real-world platform. Computational and storage overhead analysis shows that the proposed protocol for SMT implements security, authentication, integrity, robustness, non-repudiation, etc. while maintaining the SDR. Messages that are less important compared to the SMs are called non-safety messages (NSM) and vehicles use RTS/CTS mechanism for NSM transmission. Numerical studies show that the proposed NSM transmission method maintains 6 times more throughput, 2 times less delay and 125% less Packet Dropping Rate (PDR) than traditional MAC protocols. These results prove that the proposed protocol outperforms the traditional MAC protocols.

Keywords: Ad hoc networks; data security; digital signatures; distributed storage; intelligent vehicles; vehicular ad hoc networks; wireless communication

1 Introduction

Vehicular Ad hoc Networks is an especial type of dynamic wireless network, designed to provide a communication infrastructure between vehicles. Inside a VANETs system each device needed to be well equipped to exchange information (send and receive) with other vehicle's drivers or vehicles within their reach. The IEEE 802.11-2016 standard [1] provides MAC and physical layer protocols for VANETs.

VANETs are targeted to provide safety, efficiency and Infotainment. Collision warning, safe-distance information, congested road notification, risky vehicle warning, road barrier/obstacles/block notification, signal/rule violation warning etc. are emergency notifications which are transmitted between vehicles to alert each other by using different VANET protocols. These are called safety messages. Moreover, infotainment can be delivered by incorporating commercial service information, gas station/parking/restaurants/hotel information, media content download, multiplayer games, etc. Those are categorized as the non-safety message. Safety messages always get priority for transmission as fail or delay distribution of those messages may result in severe accidents, traffic jam, etc. Processing all types of messages together will increase the traffic and harm the throughput and overall performance. Because during transmitting, it is required to follow Strict Delay Requirements (SDR) of 100 ms for safety message [2]. Contrastingly, NSM is comparatively less important than safety message and does not require SDR to follow.

To maximize the throughput and minimize transmission delay & PDR Cluster-based (CB) protocols could be a better solution for VANET [2]. The nearby vehicles heading towards the same direction could form a cluster to transfer information among themselves. Although CB systems are easier to manage and simultaneously appropriate for resource utilization and performance enhancement, traditional CB systems are suffering from some shortcomings like hidden node problems, traffic overloading, packet dropping, etc. CB-MAC protocol [2] has overcome the shortcomings of CB systems and proposed a complete solution for VANET. By using their Non-Safety Message Transmissions (NSMT) protocols, it improved the communication quality by increasing the throughput and decreasing transmission delay & Packet Dropping Rate (PDR). Adding the security attributes like confidentiality, authenticity, reliability, transparency, integrity etc. to the CB-MAC system could assure security with good performance.

However, safety messages are very crucial for VANET systems and should be protected from any kind of attacks. Attackers could modify message contents and generate false messages or can provide false replies by using a man-in-the-middle attack [3]. Those security leaks may result in fatal accidents.

Meanwhile, the popularity of blockchain is increasing because of its distributed features and the secured storage service for P2P communication. Blockchain is considered as immutable ledgers and ensures important security services [4,5]. All data and transactions are stored as chained blocks and it is not possible to edit or delete any information after being stored, which ensures the integrity, immutability and trustworthiness. These features motivate us to employ blockchain in the proposed system to store safety messages. A Public Key Infrastructure (PKI) based digital signature algorithm is used to ensure the authentication of the cluster members and also to provide communication security. Additionally, we propose a Certificate Authority (CA) for physical verification of the vehicles and to generate a public-private key pair for each of the vehicles. In this paper, to increase throughput by ensuring security services, we propose a Secure Cluster-based MAC (SCB-MAC) protocol for vehicular ad-hoc network. The target is to introduce the security features to the SMT of VANET systems. Handling safety and non-safety

messages separately, providing signature-based security during cluster joining and communication, blockchain-based decentralized and distributed storage of the messages and vehicles registration and physical verification are the novelty introduced in this paper. As a Proof of Concept (PoC), we implement an Ethereum blockchain in virtual machines with Cluster Members (CMs) and CH to simulate the SMT. The scenario is tested in a real-world Ethereum test network named *Rinkeby* test network [6].

The contributions of the paper are the followings:

- We propose a blockchain-based Secured Cluster-based MAC protocol (SCB-MAC) for VANETs. SCB-MAC defines the formation of the cluster, handshake methods, safety and non-safety message transmission in details. We have modified some of the control packets formats of IEEE 802.11 to allow blockchain and to support those methods.
- We propose blockchain to store and distributes the safety messages of clusters to provide a decentralized environment while ensuring robustness, temper resistance, immutability, fairness and transparency of the safety messages. The blockchain is hosted in the cloud and the corresponding CH and CMs will communicate with it by using high-speed internet. All the CMs including CH are considered as the full node and anyone can initiate a transaction on the blockchain to inform a safety message. Blockchain will generate block from each of the safety messages and broadcast it to all the CMs including CH.
- We have employed a PKI based digital signature method to ensure the authenticity of cluster members as well as to provide communication security. During cluster joining communicating with the blockchain server, digital signature is used to ensure user authentication and integrity, confidentiality, nonrepudiation of the message.
- We introduce CA to register and verify vehicles. Additionally, it is responsible to generate public-private key pair for each of the vehicles and to ensure the safety, security and preservation of their privacy.

We have discussed some cluster based VANET systems with their performance and security in the related work section (Section 2). Research paper where blockchain is employed for VANETs is also added in that section. The system structure is demonstrated in Section 3. The tools used for implementation and experimental setup details are discussed in Section 4. The performance analysis of the proposed SCB-MAC protocol is demonstrated in Section 5. Security analysis of the proposed method is presented in Section 6. In Section 7 we present the conclusion of the paper with some possible future works.

2 Related Works

Cluster-based systems are proved very useful for VANETs. The quality and performance improvement by using a cluster-based architecture in VANETs can be found in [7]. Yang et al. [8] proposed a cooperative Clustering-based Medium Access Control (CCB-MAC) protocol to enhance the trustworthiness of emergency message broadcasting by improving their reception rate. In [9], the researchers presented a multi-channel CCB-MAC which also improves the reliability with QoS support with the help of cooperation between the members. The authors in [10] also proposed a cluster-based multichannel MAC protocol where they developed an analytical model to find out suitable window size for the MAC protocol to balance between the delay and the successful delivery rate. A hybrid cluster-based protocol is proposed for safety message transmission by [11] which improves the network stability and increase channel utilization by selecting the cluster head according to the mobility factor of the vehicles. Due to lack of neighboring node,

TDMA protocols are not able to utilize all the time slots of a frame. The [8–11] do not have efficient resource utilization capability.

In [12], researchers proposed DMMAC, which is also a cluster-based MAC protocol by utilizing Fuzzy logic Inference System (FIS). But their method is applicable only for emergency/safety messages. A multihop-cluster based hybrid architecture is presented for the safety message transmission to minimize the connection overhead and PDR [13]. In [14], the researchers combine the clustering protocol and carry-and-forward schemes for highway VANETs. Reference [14] shows improvement in data download volume and throughput but information about network delay and packet dropping rate is not mentioned.

The strict delay constraint for the safety message transmission is 100ms, but the presented methods do not satisfy this credential. Additionally, [8,11–13] provide solution only for safety messages and does not concern about the general messages or non-safety messages. Thus, in this paper, we propose a cluster-based method where safety and non-safety messages are handled separately according to their importance. Rather than following the traditional MAC protocols, a blockchain-based method is proposed to ensure the SDR for safety message transmission.

Zhang et al. [15] presented a Data Security sharing and Storage system based on the Consortium Blockchain (DSSCB). They utilize the temper-proof and security features of blockchain to store authentication information like identity and keys with location, direction, current position and rule violation information of the vehicles. Similarly, Javaid et al. [16] use blockchain to store registration and status information of vehicles in their DrivMan system. To ensure the trust of vehicles, a video storage system was proposed by Xie et al. [17] where vehicles use their onboard camera to capture video of the surroundings and send it to a blockchain to store. The stored information is used to analyze the behavior of vehicles to find any unwanted or malicious behavior. Wagner et al. [18] proposed a method to ensure the integrity of the event messages. Blockchain is used to store the reputation score of the vehicles which is updated after each transaction. Zhang et al. [19] utilized blockchain to store important traffic event information like traffic violation and accidents. They use Mobile Edge Computing (MEC) for computational support, but because of MEC, the system is not fully decentralized. Another blockchain-based message dissemination service was proposed in [20,21]. Blockchain is used to store the verified event information to ensure the security and trust of the system. Another event validation mechanism is proposed by Yang et al. [22] where RSUs broadcast event messages to the vehicles and vehicles use PoW to verify the trustworthiness of that event. To implement a scalable system, they use local blockchain to provide quick response to the local vehicles and then all the local RSUs synchronized the data into the global blockchain. But the infrastructure cost for RSU and resources are high, thus in most of the cases vehicles have to pay a good amount of money for that [23]. To handle the huge workload of event data, Singh et al. [24] presented branch–based technology with blockchain in where blockchains are divided into branches and each branch is responsible to provide services in different geographical areas. To increase the scalability, some researchers use multiple blockchain to store different information separately [25–27].

In the above-mentioned research, different types of blockchain are used for different purposes. However, none of them differentiates between message or event types. If all the transmitted messages or event information are stored together in the blockchain with proper security encryption services, the time and storage overhead of the system must be high. It results to decrement of throughput and increment of delay. To minimize the difficulties of consensus and mining [18] minimize the difficulty to 4 leading zeros while [24] set it to 3. Blockchain could be the best solution to provide security, integrity, availability, transparency, robustness etc. But without proper

management, the performance could be very low. Thus, we propose a blockchain to store the safety messages. Non-safety messages are not stored in the blockchain as they are less important and consume too much storage.

3 System Structure

The nearby vehicles heading towards the same direction will form a cluster. All the vehicles are well equipped with necessary hardware and software resources to send and receive messages including OBU, Sensors, Global Positioning System (GPS) and high-speed internet connection. The vehicles are physically verified by a Certified Authority (CA). CA also generates and assigns a public-private key pair to each vehicle and all the vehicles will be known as their public key. CA is considered as secured enough to preserve the privacy of the vehicles. A graphical representation of clusters is presented in Fig. 1a. Among the vehicles, one will be elected as Cluster Head (CH) and others become Cluster Member (CM). By this way, a centralized system is formed where all the NSMTs between CMs will be handled by the CH as an access point. Every cluster owns a blockchain to store the safety messages. All the CMs including CH are considered as a full node and anyone can initiate a transaction in the allocated blockchain to inform about an emergency. Vehicles will sign the message with their private keys to confirm their identity and to ensure non-repudiation. The blockchain server will check the authentication and then generates block from the message and broadcasts it to all the members. Details of the system model are discussed in the following subsections.

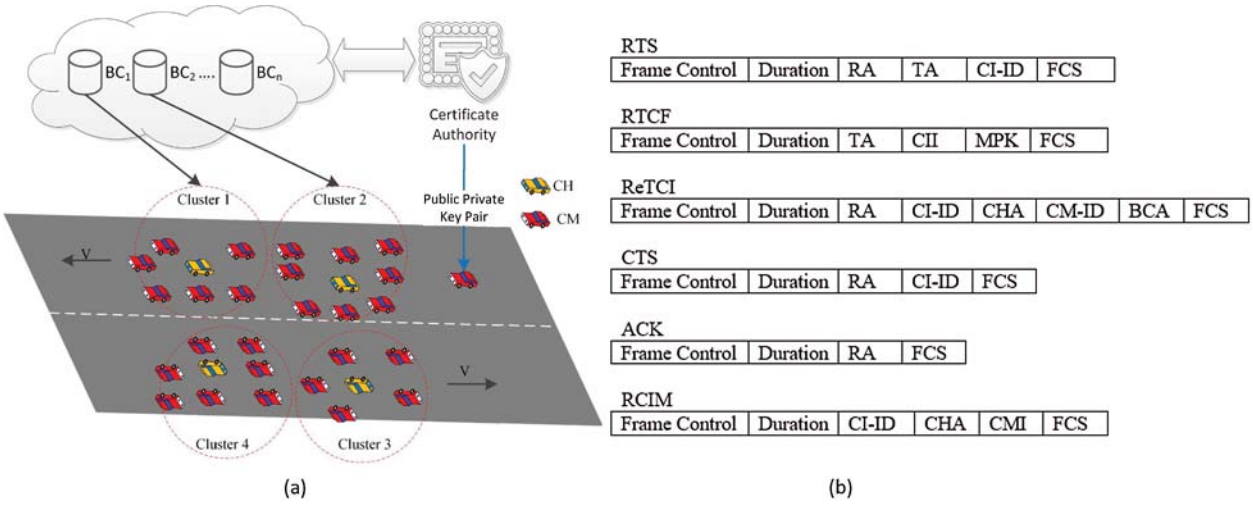


Figure 1: (a) Application scenario, (b) modified control packet format for SCB-MAC

3.1 Formation of SCB-MAC Cluster

SCB-MAC is a cluster-based system with some modification from the traditional IEEE802.11 standard (see Fig. 1b). In this section, we will discuss the details of cluster formation and related details.

3.1.1 Cluster Membership

To join a cluster, an isolated vehicle has to broadcast a control message called Request to Cluster Formation (RTCF) in the network. Cluster Information (ClI) and the vehicle's public key

i.e., the Member's Public Key (MPK) are included in the RTCF. Then, CH of the nearby cluster sends back a (Registration to Cluster) ReTCl packet to the isolated vehicle by informing about the cluster, Public key of CH and the Address of the Blockchain (BCA) assigned to that cluster. The new member id of the vehicle is also included in the ReTCl. To ensure authenticity, CH signs the BCA with its private key. The newly joined member has to decrypt it by using the public key of CH and then registered to the assigned blockchain. A vehicle can receive multiple ReTCl, in that case, the vehicle will calculate the time interval between sending and receiving of the control messages and join the cluster where the delay is minimum. If no cluster is present nearby and the vehicle considers itself as CH and starts a new cluster. Then it can apply to the server to allocate a blockchain for the newly formed cluster. The new CH will broadcast the ClI in the network and wait for some CMs to join. Unified Modelling Language (UML) is used to sketch the FSM of the proposed SCB-MAC protocol (see Fig. 2).

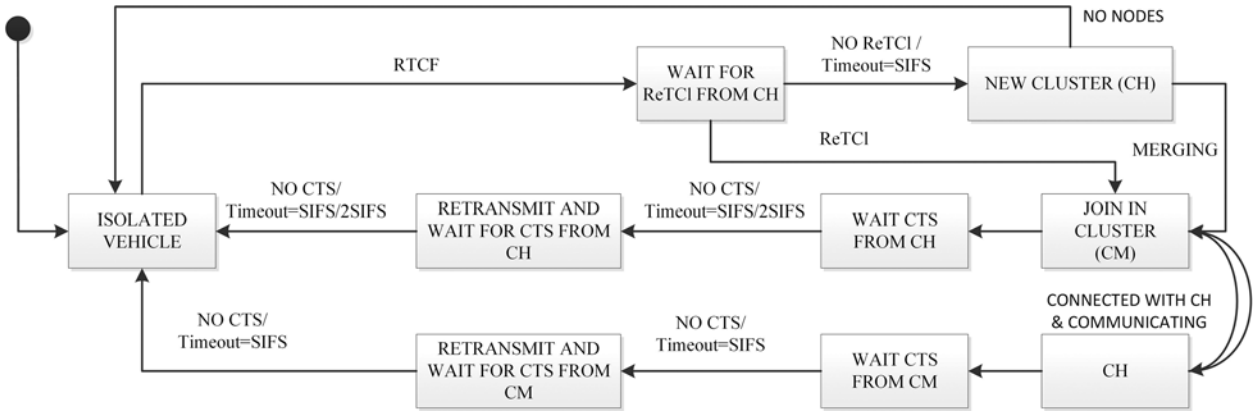


Figure 2: Finite state machine of the proposed SCB-MAC protocol

3.1.2 CH Election and Cluster Merging

An active but isolated vehicle will broadcast RTCF and wait for ReTCl to join an existing cluster as a CM. But if it does not receive any ReTCl and SIFS timeout occurs, the vehicle becomes CH to form a new cluster. If multiple CHs come very closer and start using the same channels, CHs will receive control messages from each other. Then all the CHs those who realize the existence of cluster(s) will broadcast a control message called Request to Cluster Merging (RCIM). Inside RCIM, CH includes Cluster's Member Information (CMI) to inform the number of CMs active under its cluster. After receiving the RCIM, Cluster(s) with a lower number of CMs will join to the cluster with the largest number of CMs. All the CMs including the CH(s) will join as new CM. Newly joined CMs will exchange RTCF and ReTCl with the CH to complete the merging process. CH of the previous cluster will initiate a transaction in the current blockchain to synchronize the valid safety messages from the previous blockchain.

3.1.3 Leaving a Cluster

For different circumstance, anyone can leave a cluster and then the CMs list is updated dynamically. Cluster leaving may be required in four situations and those are demonstrated in Fig. 3. While the CH sends RTS to a CM and does not receive any CTS even after retransmission, CM will be considered as out-of-reach (see Fig. 3a). Similarly, while the CH sends RTS on behalf of a sender CM to receiver CM and does not receive any CTS even after retransmission,

destination CM will be considered as out-of-reach (see Fig. 3d). If there is no ACK received from a CM after broadcasting and resending a message, that CM will be considered as out-of-reach (see Fig. 3b). If the CH is out of reach and a CM does not receive any CTS even after the retransmission the CM will initiate a cluster leaving process (see Fig. 3c).

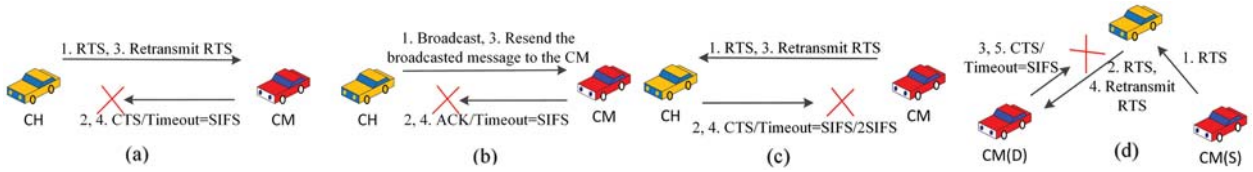


Figure 3: Cluster leaving processes while (a) no CTS is received from a CM, (b) no ACK is received from a CM, (c) no CTS is received from the CH and (d) no CTS is received from the CM(D)

3.2 Safety Message Transmission (SMT)

Collision warning, safe-distance information, congested road notification, risky vehicle warning, road barrier/obstacles/block notification, signal/rule violation warning etc. are considered as emergency or safety messages. These types of messages have strict delay requirement which is 100 ms [2]. The safety messages should come from a valid source and stored in such a way that, if one or multiple cluster members (including the CH) leave the cluster, the safety messages should not be lost. Traditional cluster-based systems are managed by a central node and thus the possibility of single point-of-failure is high. Blockchain is a perfect solution for these obligations as it provides data storage and management system in a distributed environment.

When any isolated vehicle become a CH, it will communicate with the server to get the network address of an available blockchain. The server will provide an address where the smart contract for the SMT was previously deployed. If the CH was a member of any previous SMT blockchain, it will copy the related and valid safety messages to the newly created blockchain as transactions. Whenever a CM wants to join the cluster and shares its public key, CH will send the sever credentials of the blockchain by signing it using the CH's private key. The CM will connect with the blockchain server and then it will synchronize to receive all the existing safety messages of the blockchain. All the CMs including CH are independent nodes in the blockchain and everyone can perform transactions in the blockchain to inform others about a safety message.

Each safety message is generated by a smart contract as a transaction and stored chronologically as a block in the blockchain. After any block is generated, all the CMs will get notification about the newly created safety message in block form. If any CM has validity expired information for a particular safety message, it will request for another transaction in the blockchain to mark the message as invalid. For example, whenever a vehicle changes a lane it will generate a transaction but when the vehicle will move to another lane the previous information become invalid. Thus, it will generate an invalid transaction and the block will be marked as invalid. As the messages consume very small storage, the block will not be removed from the blockchain. However, the information stored in the block could be used by the law enforcement authority to investigate different occurrence like an accident, traffic jam etc.

3.3 Non-Safety Message Transmission (NSMT)

The non-safety message transmission will be unicast to and fro a CM or a CH. There are three categories of unicast and corresponding transmission is briefly discussed here. From CH to CM, there is a direct unicast from CH to CM. CH verifies transmission using ACK. The data is routed via CH as a CM cannot send non-safety messages directly to another CM, rather they send it to the CH and CH will be responsible to broadcast messages to the destination CM. On the other hand, CH can transmit non-safety messages to neighbour clusters' CH by using the RTS/CTS mechanism. Fig. 4a–4c shows the handshaking between the members of the proposed NSMT protocol.

Like traditional MAC protocols, if all the CMs are transmitting messages to each other there will be duplicate message exchanges and hidden node problem is possible to occur. Moreover, with the increment of the number of vehicles a huge flow of messages will be generated which increase the chance of collisions and transmission delay [2]. Thus, in the proposed method, CM has the responsibility to handle the non-safety message communication and rather than broadcasting immediately CH sends to one CM at a time and waits until receiving an ACK from that CM. After the ACK is received it will send the message to another CM. It is possible to set the maximum number of retransmission limit for NSMT, and if an ACK does not receive by the CH within that time, it will retransmit the message until the limit. Fig. 4d shows the flowchart.

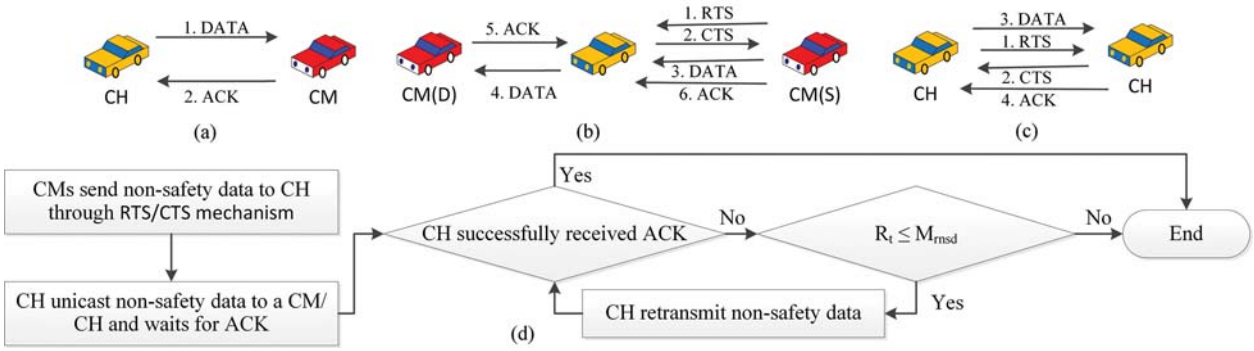


Figure 4: (a) Handshake between CH and CM during NSMT, (b) flowchart of the NSMT method

4 Implementation

For the proposed SMT of the SCB-MAC protocol, we present a Proof of Concept (PoC) implementation by using the Ethereum blockchain. Generally, the transactions are performed by miners who are also members of the blockchain. But in the proposed system, as the vehicles are the members of the blockchain and many of the vehicles do not have the capability to mine blocks, we have introduced a server which will perform the mining tasks on behalf of the vehicles. Moreover, online computing service providers also maintain a distributed service. Thus, our proposed system is decentralized and distributed as the data are not stored in the server or a specific location rather stored in all the vehicles storage. A Virtual Machine (VM) was configured with Ubuntu-18.04.4-desktop-amd64 to host the Ethereum blockchain and also act as a miner. Two other VM is considered as CH and CM. Registration to the blockchain and message transmission is tested with this setup. We are going to describe the details of the implementation in this section.

4.1 Tools

We implemented the SMT module of the SCB-MAC protocol by using the Truffle framework. It's a well-known testing framework for Ethereum blockchain which provides all the facilities to manage smart contracts, automated testing of the codes, deploy smart contracts in Ethereum blockchain [28]. To emulate and test the smart contracts into a blockchain, the truffle suite offers *Ganache* [29], a virtual private Ethereum blockchain. *Ganache* offers special features to examine the blocks and transactions, blockchain log to analyse the responses and debugging information in the popular platforms like Windows, Mac OS and Linux. The vehicles use *metamask wallet* [30] to connect with virtual private blockchains. It provides all the wallet facilities to access, control and pay to the blockchain-based applications. *Metamask* comes in the form of a browser extension and also available for iOS and Android as apps. Not only the main Ethereum network but *metamask* also provide the facility to connect with different test networks including custom RPC (Remote Procedure Call). A Node Packet Manager (NPM) is used to executes JavaScript in the proposed method [31]. To interact with the smart contract, we developed the client-side in HTML by using Lightweight NPM Server [32].

4.2 Experiment

In a typical VANET system, all the vehicles may not have the mining capabilities. Thus, in the proposed system, one or more servers are used to perform mining on behalf of the vehicles. It could be an external EDGE server (like [17,20,21]) or an existing blockchain server (like [6,33]) which is available online. To test our proposed system in both environment we present two different experimental setups. In the first setup, we are considering a dedicated EDGE server as miner (configured in a virtual machine). Moreover, In the second experiment, a real-world platform (*Rinkeby* test network) is considered as blockchain server to perform mining.

4.2.1 Ganache Test Server

To implement the SMT module we prepared a VM as blockchain server with Ubuntu-18.04.4-desktop-amd64 installed. First, we install *ganache* and consider it as a blockchain of a particular cluster. Then, we install NPM as it is a prerequisite to rum truffle framework and then install the other dependencies. In the SMT blockchain, there are two types of operations. First one is to store a safety message and the second one is to mark it as invalid when the impact/validity of the message is no longer valid. Thus, we write a smart contract which consists of three functions. One to view the existing blocks, the second one to add a safety message in the blockchain and the third one to mark a safety message as invalid. The SC is written in solidity and deployed into the blockchain by using truffle.

Next, in the CH and CM virtual machines, we install *metamask* Ethereum wallet extension in the Firefox web browser. In the *metamask*, we use the custom RPC option to connect the *ganache* blockchain server which is running in the server VM with a customisable port number. The CM and CH used their public keys to register with the blockchain. We considered that the CM and CH are verified by CA. Thus, the CM and CH have the permission to perform operations in the blockchain. *Ganache* provides 100 ethers to CM and CH to pay the fees i.e., the gas during a transaction. After testing it in the local VM, we found that all the functions are running fine and ready to deploy in a real-world platform.

4.2.2 Rinkeby Ethereum Testnet

To deploy smart contracts and execute transactions in a real-world platform, we have used *Rinkeby* [6], which is an Ethereum test network. It is one of the popular test networks used

by blockchain developers. By sharing the account information in the social network, we have earned some virtual currency i.e., ether which is usable only for *Rinkeby*. Although the earned ETHERs for *Rinkeby* are valueless in the real world, to perform any transaction and smart contract deployment we need those ETHERs to pay the gas price.

We have tested our smart contract in the Remix IDE (integrated Development Environment) which is a platform independent environment [34]. It's a web-based service which provides different compiler versions to run smart contracts and execute blockchain transaction. After deploying the smart contract and performing some operations in the *Rinkeby* testnet by using Remix IDE, details report about the blocks and transactions can be found in the *etherscan* web site [35]. The reports include the timestamp, transaction fee, gas limit, gas fee, block number, hash values of transactions etc.

5 Performance Analysis

The performance analysis of the proposed SCB-MAC protocol is divided into two parts. Firstly, we will demonstrate the performance of the SMT protocol which includes the computational overhead analysis of the digital signature and key generation algorithm. Storage overhead due to Ethereum blockchain is also presented in that section. Then we will discuss the performance of the NSMT protocol by comparing the throughput, PDR and delay with the traditional MAC protocol.

5.1 Performance Analysis of the SMT Protocol

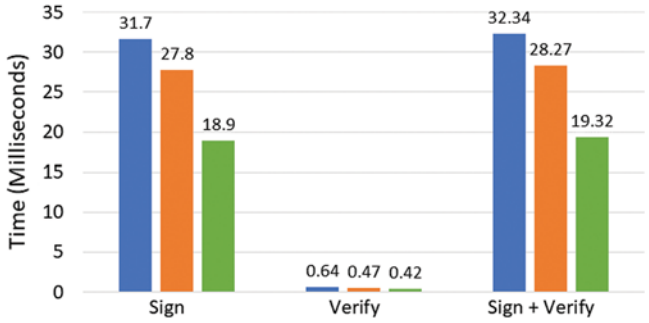
To ensure security, integrity and authenticity of the transferred message whenever any CM or CH wants to initiate a transaction in the blockchain, it signed the message with its private keys as a proof of authenticity. Similarly, during the first communication with a CM, CH sends BCA inside ReTCl by signing it with CH's private key. In both situations, the system uses RSA-1024 algorithm. The security strength i.e., the difficulty of breaking the key is measured in bits and according to NIST [36], the security strength of RSA-1024 is 80 bits. That means to break the key attacker have to perform at least 280 operations. According to some reports 80-bit security is considered as below standard, but for the system with lower computational power like VANET, IoT, etc. that would be considered secured enough. However, Singh et al. [37] presented RSA-1024 with the security level equal to the symmetric key size of 112-bit. In SCB-MAC, vehicles use high-speed internet connection to communicate with the blockchain and the propagation delay considered ignorable.

5.1.1 Computational Overhead

For a computer with more than 1.5 GHz clock speed, RSA-1024 with 80 bits security would take 1.48 ms for signing and 0.07 ms for verification [38]. So, it is possible to sign and verify a message within 1.55 ms. However, to calculate the signature and verification time for RSA-1024 with 112-bit security three intelligent vehicles are considered with different computational resources. Processing speed and RAM of the vehicles are presented with their time required to sign and verify a safety message of 24-bytes are presented in Fig. 5. For IV1, IV2 and IV3 it requires 32.34, 28.27 and 19.32 milliseconds respectively to complete sign and verification process.

As the strict delay constraint for the SMT is 100 ms, it is possible to sign and verify at least 64 messages by using RSA-1024 signature method (with 80-bit security). However, while the security strength is considered 112-bit, it is possible to complete 3 to 5 transaction. From the previous works we have found that the average delay for safety messages of [10] is 151 ms and in [39] it

is also more than 100 ms. So, the SMT time is good enough to maintain the SDR. However, it is possible to hire multiple EDGE servers to improve the scalability of the system. During cluster joining i.e., the registration processes the time required for signature and verification is 32.34 ms for a low configured vehicle (see Fig. 5). That means it is possible to register more than 30 low-configured vehicles per second. This is a minimum cost to ensure security, integrity and authenticity. The CA use a key generator to provide public and private key pairs for vehicles. The key generation time for RSA-1024 is 97 ms for a computer with a 3.1 GHz processor and 4.0 GB of RAM [37]. So, it can generate at least 10 keys per second.



Intelligent Vehicles	IV1	IV2	IV3
Processing Speed	2.2 GHz	2.7 GHz	3.1 GHz
RAM	1.0 GB	2.0 GB	4.0 GB

Figure 5: Signature and verification time required for various intelligent vehicles

5.1.2 Storage Overhead

Block header of the Ethereum blockchain is approximately 508 Bytes [40]. In Ethereum, every block consists of a single message. In the worst case, if a safety message block is generated in every 5 s (12 in a minute), the storage overhead is $508 \times 12 \times 60 \times 24 = 8.37$ MB/Day. Therefore, the proposed method requires a small amount of storage and possible to store them for a long period. However, when there remains no member in a cluster, the server reset the blockchain by archiving all the blocks in a cloud. Thus, too much storage support is not required for the proposed SMT protocol.

5.2 Performance of the NSMT Protocol

In SCB-MAC safety messages will be transmitted by using high-speed internet which will remove workloads from the internal network which results in an increment of throughput and decrease of PDR and delay during NSMT. In this section, we will present the performance analysis of the NSMT and compare it with the traditional MAC system. A numerical analysis is presented with arbitrarily distributed

A numerical analysis is presented with arbitrarily distributed n number of vehicles which are moving through a multi-lane road. Speed of the vehicles are considered as 100 km/h and the width of the road is 5 m. Vehicles are moving in almost the same speed and their transmission area is 500 m. If these parameters are changed, performance will be changed too. Details about their impacts are discussed in [41,42]. Tradition MAC protocols for VANETs are studied in [2,41,43]. We used these studies and data to compare our method with the traditional MAC protocols. However, in the context of this paper sensitivity test is not going to add any new value as the comparison will not be fair. More importantly, sensitivity test would have been apt if there were similar blockchain-based MAC protocol for VANET. The analysis is performed in MATLAB and the considered value of parameters are presented in [2].

5.2.1 Throughput Analysis of the NSMT Protocol

The normalised system throughput S for kth cluster is can be calculated as:

$$S_k = \frac{P_s P_{busy} L}{T_e} = \frac{P_s P_{busy} L}{P_i T_{slot} + P_{busy} P_s T_s + P_{busy}(1 - P_s) T_c} \tag{1}$$

Here, Ps = Probability of successful transmission, P_{busy} = At least one transmission is in progress, L = Transmitted packet length, P_i = Probability that the channel is idle, T_{slot} = Slot time, T_e = Expected time to spend in a state, T_{span} = Time span of slot, Ts = Time span for successful transmission and T_c = Time span if there is collision.

The throughput of the system would be:

$$S = \sum_{k=1}^{j} S_k \tag{2}$$

Fig. 6a shows that the throughput for SCB-MAC (NSMT) is comparatively higher than traditional MAC-based methods. In traditional MAC, CH broadcasts all the messages immediately which increase collisions and throughput decrease quickly. When the number of vehicles is small, cluster size will be small. A small cluster could not be able to utilise the available radio resources due to an inadequate number of vehicles in the cluster and low traffic demand generated in the cluster [2]. Therefore, throughput is lower than traditional MAC protocol.

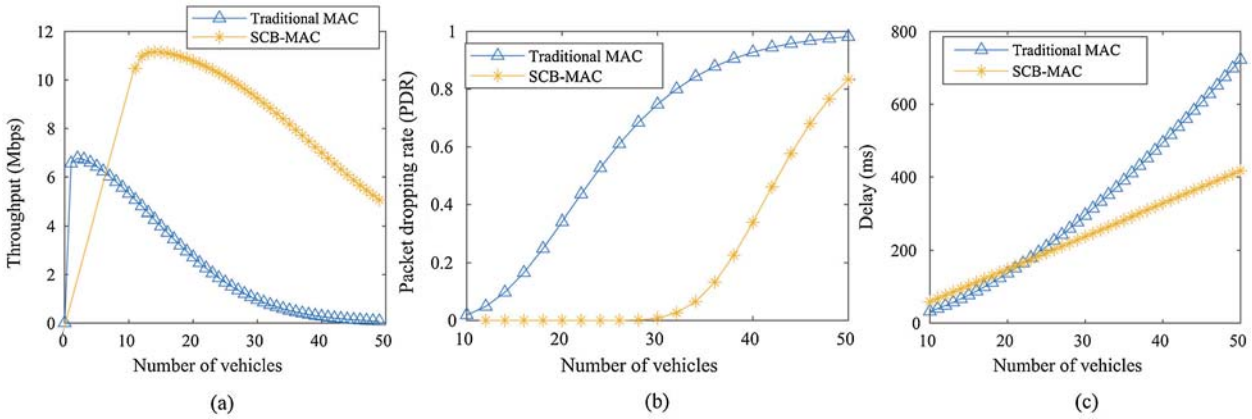


Figure 6: (a) Throughput, (b) PDR and (c) delay comparison between SCB-MAC and traditional MAC

Firstly, as the safety messages are not using the internal network, the load of messages are less. Secondly, rather than broadcasting immediately SCB-MAC uses RTS/CTS to check the existence of the CMs first and then transmits to remove the hidden node problem. Thus, the increment of throughput is significant but with the increment of vehicles, collisions are also increasing which decrease the throughputs gradually for all types of systems. For example, while the number of vehicles reaches to 40, traditional MAC protocol is overloaded and too much collision decreases the throughput to almost 0 while proposed SCB-MAC can maintain a throughput rate near to 6 Mbps. Moreover, the maximum throughput of the SCBMAC protocol is about

12 Mbps for NSMT, where previously proposed methods like [7,11,14] have achieved 1.1, 1.3 and 11 Mbps respectively.

5.2.2 Packet Dropping Rate of the NSMT Protocol

To calculate PDR of the network the following equations are derived in [2]:

$$PDR_{nsd} = (1 - P_s)^{M_{rnsd}} \tag{3}$$

where M_{rnsd} are the maximum retransmission limit for NSMT. To ensure the availability of safety messages there is no limit for retransmission, which increase overhead and increment of PDR. In the proposed method, safety messages are not using the internal network which decreases the PDR rate of the network. Thus, PDR is less than traditional MAC in the proposed method although the retransmit limit is the same. Fig. 6b shows that the PDR for the proposed SCB-MAC is near to 0 until the number of vehicles reaches to 30 and after that, it increases but always less than traditional MAC protocols.

5.2.3 Delay Analysis

In [2], Shah et al. presented the transmission delay of a cluster-based system could be calculated as:

$$E[D] = E[T_{interval}] - \frac{P_{fdrop}}{1 - P_{fdrop}}.\ E[T_{drop}] \tag{4}$$

So delay for non-safety messages will be:

$$E[D_{nsd}] = T_e \left(n - \frac{P_{drop}}{1 - P_{drop}} \times \frac{2}{1 + CW + M_{rnsd} CW/2} \right) \tag{5}$$

Fig. 6c shows the average packet transmission delay against the number of vehicles. As the proposed method uses RTS/CTS handshake before sending any non-safety messages, initially the transmission delay is a little higher than the traditional MAC protocol. But with the increment of the number of vehicles, the traditional MAC system faces rapid increment of transmission delay because of collisions, while SCB-MAC keeps it manageable.

5.2.4 Results and Discussions

The cluster-based protocol is based on IEEE802.11 Distributed Coordination Function [DCF]. Performance of the IEEE802.11 can be found in [43–49]. NSMT achieved maximum throughput of 12 Mbps, while some previously proposed method achieved [7,11,14] have achieved 1.1, 1.3 and 11 Mbps respectively. Increasing number of messages increases collisions which result to decrement of throughput and increment of PDR and delay. For SCB-MAC the internal network will be available only for non-safety messages because the safety messages will be transmitted by the internet. Therefore, the full network is available only for non-safety messages and that results in throughput increment. Maintain a throughput of 6 Mbps for NSMT, while the number of vehicles reaches to 40. In the same state throughput of traditional MAC protocol is close to zero. SCB-MAC is free from hidden node problem as only live nodes could receive non-safety messages which are achieved by RTS/CTS handshaking. By removing hidden node problem SCB-MAC can minimize PDR and transmission delays. When the total number of vehicles is 50, the transmission delay of the MAC protocol reaches to double (800 ms) than the proposed protocol.

6 Security Analysis

In this section, we will discuss the security features of the SCB-MAC protocol. Blockchain with CA and public key infrastructure provide strong security to the transferred safety messages. The security features are the followings:

6.1 Source Authentication and Non-Repudiation

We propose a PKI based digital signature method which is considered as secure until the attacker succeeds to get the private key. Each of the vehicles is physically verified by CA during registration. CA is responsible to ensure the safety and security of identities. To perform a transaction in the blockchain a vehicle has to encrypt the safety messages by using its private key to confirms its identity and nonrepudiation. The blockchain server will verify the vehicle's identity before creating a block.

6.2 Privacy Preservation

The real identity of the vehicles is securely stored by CA by mapping it with their public key. The vehicles use to communicate with others by using their public keys to disclose their original identity to the public. Therefore, even if an adversary could get the public-private key pairs it is not possible to guess the real identity of the vehicles. The proposed SCB-MAC ensures the privacy of the vehicles with the help of CA.

6.3 Security, Integrity and Confidentiality of Messages

All the SMTs are encrypted by RSA-1024 cryptographic algorithm which ensures security, integrity and confidentiality of the messages. RSA-1024 considered strong enough as the key attacker have to perform at least 280 according to [36] or 2112 according to [37] operations to break the keys. The blockchain server checks for the integrity of the message by matching the hash value by decrypting the message. Any modification affects the hash value and that message will be rejected.

6.4 Attack Prevention

PKI based digital signature algorithms are considered as secure until an attacker creaks the private key [50]. So, the communication channel used in SCB-MAC is theoretically secured. It also prevents the messages from being modified and fabricates by comparing the hashing value. Even if the adversary got the public-private key pair, it is not possible to get the hash of the former block in the blockchain. So, a fabricated message with wrong hash value will be rejected. So, reply attack from an unknown source similarly rejected. Moreover, the digital signature-based system prevents impersonate attack because it is not possible to generate a valid signature on behalf of a vehicle. However, CA confirms the physical identity of the vehicles and the blockchain server checks the authentication information before block generation. No unauthorized entity, as well as no vehicles with multiple fake identities, could perform any operation in the system. Thus, we can say that the system is free from Sybil attack or unknown source attack.

Additionally, SCB-MAC can prevent DDoS attack as the blockchain never accepts any unauthorized entity to perform any operation and they will be blocked by the server from sending further messages to the blockchain. DDoS, man-in-the-middle attack, Sybil attack, replay attack, etc. are the attacks that can harm a VANET system [51]. By using public-key cryptography based digital signature, SCB-MAC is safe from these attacks. Additionally, proposed signature method does not depend on verifier table, thus the system is safe from stolen verifier table attack.

6.5 Decentralization, Flexibility, Temper-Resistance, Immutability, Fairness, Transparency, Robustness

SCB-MAC utilizes the features of blockchain. It provides a decentralized and distributed environment to store data in a platform-independent and flexible way. Ethereum platform can be accessed by using *metamask* wallet [30], which could perform operations from any kind of computers and mobile devices using any operating system like Windows, MAC, Linux and any cell phone that uses iOS or Android. All the members have a copy of all the blocks in the blockchain, which prevent the system from single-point-of-failure and provides robustness. The storage structure of blockchain is chronological which is ensured by hashing. It does not allow anyone to change the content even the sequence of blocks which ensures immutability and tamper-free storing of the safety message. All the members in the cluster are equally treated while operating on the blockchain to ensure the fairness of the system. By using smart contracts, a vehicle could disable the safety message which is no more valid. In that case, the message is still stored in the blockchain and every member can see it as an invalid message. Even if any blockchain is reset, data blocks of it are archived in the cloud under the supervision of the CA. It could be used in future for accident investigation, traffic violation, etc. This storing method could help law enforcement authority during the investigation of accidents.

7 Conclusion

For VANETs, cluster based VANET systems are performing very well to reduce PDR, increase throughput and maintain hard time constraint for SMT. To keep the performance of the cluster-based system and introduce security features in it, SCB-MAC is proposed. Firstly, an Ethereum blockchain is used to store and distribute the safety messages in a decentralized environment with flexibility, temper-resistance, immutability, transparency and robustness features. Secondly, a PKI based digital signature algorithm (RSA-1024) is used to ensure the authentication, non-repudiation, integrity and confidentiality of the safety messages. Thirdly, a CA is responsible to generate asymmetric keys for the vehicles and to preserve the privacy of the vehicle's real identity. The blockchain is implemented and tested in a realistic platform. The results show that it is possible to complete 65 message transmission within SDR of 100 ms. Therefore, the introduction of blockchain with digital signature method does not harm the SDR for SMT. Moreover, by using secure vehicles registration process it is possible to register 30 vehicles in every second. SCB-MAC provides source authentication, privacy preservation of the vehicles, attack prevention with the typical facilities of blockchain, digital signature methods. Numerical analysis is presented to check the performance of non-safety message transmission protocol and found that it performs better than the traditional MAC protocol in terms of throughput, delay and PDR. When the transmission rate of the traditional MAC protocols fall down to zero, the proposed NSMT maintain a rate of 7 Mbps and when the number of vehicles reaches to 50, transmission delay increases to 800 ms for MAC protocols while proposed method faces a delay of 400 ms only. Moreover, the PDR of NSMT is zero while the traditional MAC protocols' PDR reached to almost 60%. In future, we will try to implement a light-weight consensus method for blockchain to ensure the trustworthiness of vehicles. Additionally, we are planning to find a suitable communication protocol to exchange messages between the blockchains from neighbour clusters and also the feasibility of a secured protocol for the non-safety message will be tested in future.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Kishida, M. Iwabuchi, T. Shintaku, T. Sakata, T. Hiraguri and K. Nishimori, "IEEE standard for local and metropolitan area networks part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications, 2012," *IEICE Transactions on Communications*, vol. 96, no. 2, pp. 419–429, 2013.

[2] A. S. Shah, H. Ilhan and U. Tureli, "CB-MAC: A novel cluster-based MAC protocol for VANETs," *IET Intelligent Transport Systems*, vol. 13, no. 4, pp. 587–595, 2018.

[3] I. Ali, M. Gervais, E. Ahene and F. Li, "A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in VANETs," *Journal of Systems Architecture*, vol. 99, no. 1, pp. 101636, 2019.

[4] M. Ahmed, "False image injection prevention using iChain," *Applied Sciences*, vol. 9, no. 20, pp. 4328, 2019.

[5] M. Ahmed and A. S. K. Pathan, "Blockchain: Can it be trusted?," *Computer*, vol. 53, no. 4, pp. 31–35, 2020.

[6] Rinkeby Testnet, "Rinkeby Test Network," 2020. [Online]. Available: https://www.rinkeby.io/.

[7] H. Wang, R. P. Liu, W. Ni, W. Chen and I. B. Collings, "VANET modelling and clustering design under practical traffic, channel and mobility conditions," *IEEE Transactions on Communications*, vol. 63, no. 3, pp. 870–881, 2015.

[8] F. Yang and Y. Tang, "Cooperative clustering-based medium access control for broadcasting in vehicular ad-hoc networks," *IET Communications*, vol. 8, no. 17, pp. 3136–3144, 2014.

[9] F. Yang, Y. Tang and L. Huang, "A multi-channel cooperative clustering-based MAC protocol for VANETs," in *2014 Wireless Telecommunications Symp.*, Washington, DC, USA, IEEE, pp. 1–5, 2014.

[10] H. Su and X. Zhang, "Clustering-based multichannel MAC protocols for QoS provisionings over vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3309–3323, 2007.

[11] N. Gao, L. Tang, S. Li and Q. Chen, "A hybrid clustering-based MAC protocol for vehicular ad hoc networks," in *2014 Int. Workshop on High Mobility Wireless Communications*, Beijing, China, IEEE, pp. 183–187, 2014.

[12] K. A. Hafeez, L. Zhao, J. W. Mark, X. Shen and Z. Niu, "Distributed multichannel and mobility-aware cluster-based MAC protocol for vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 8, pp. 3886–3902, 2013.

[13] S. Ucar, S. C. Ergen and O. Ozkasap, "Multihop-cluster-based IEEE 802.11p and LTE hybrid architecture for VANET safety message dissemination," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2621–2636, 2015.

[14] M. Zhang, C. Li, T. Guo and Y. Fu, "Cluster-based content download and forwarding scheme for highway VANETs," *China Communications*, vol. 15, no. 4, pp. 110–120, 2018.

[15] X. Zhang and X. Chen, "Data security sharing and storage based on a consortium blockchain in a vehicular ad-hoc network," *IEEE Access*, vol. 7, pp. 58241–58254, 2019.

[16] U. Javaid, M. N. Aman and B. Sikdar, "DrivMan: Driving trust management and data sharing in VANETS with blockchain and smart contracts," in *2019 IEEE 89th Vehicular Technology Conf.*, Kuala Lumpur, Malaysia, IEEE, pp. 1–5, 2019.

[17] L. Xie, Y. Ding, H. Yang and X. Wang, "Blockchain-based secure and trustworthy Internet of Things in SDN-enabled 5G-VANETs," *IEEE Access*, vol. 7, pp. 56656–56666, 2019.

[18] M. Wagner and B. McMillin, "Cyber-physical transactions: A method for securing VANETs with blockchains," in *2018 IEEE 23rd Pacific Rim Int. Symp. on Dependable Computing*, Taipei, Taiwan, IEEE, pp. 64–73, 2018.
[19] X. Zhang, R. Li and B. Cui, "A security architecture of VANET based on blockchain and mobile EDGE computing," in *2018 1st IEEE Int. Conf. on Hot Information-Centric Networking*, Shenzhen, China, IEEE, pp. 258–259, 2018.
[20] R. Shrestha, R. Bajracharya and S. Y. Nam, "Blockchain-based message dissemination in VANET," in *2018 IEEE 3rd Int. Conf. on Computing, Communication and Security*, Kathmandu, Nepal, IEEE, pp. 161–166, 2018.
[21] R. Shrestha, R. Bajracharya, A. P. Shrestha and S. Y. Nam, "A new type of blockchain for secure message exchange in VANET," *Digital Communications and Networks*, vol. 6, no. 2, pp. 177–186, 2019.
[22] Y. T. Yang, L. D. Chou, C. W. Tseng, F. H. Tseng and C. C. Liu, "Blockchain-based traffic event validation and trust verification for VANETs," *IEEE Access*, vol. 7, pp. 30868–30877, 2019.
[23] B. Leiding, P. Memarmoshrefi and D. Hogrefe, "Self-managed and blockchain-based vehicular ad-hoc networks," in *Proc. of the 2016 ACM Int. Joint Conf. on Pervasive and Ubiquitous Computing: Adjunct*, Heidelberg, Germany, pp. 137–140, 2016.
[24] M. Singh and S. Kim, "Branch based blockchain technology in intelligent vehicle," *Computer Networks*, vol. 145, pp. 219–231, 2018.
[25] C. Lai and Y. Ding, "A secure blockchain-based group mobility management scheme in VANETs," in *2019 IEEE/CIC Int. Conf. on Communications in China*, Changchun, China, IEEE, pp. 340–345, 2019.
[26] Z. Lu, Q. Wang, G. Qu and Z. Liu, "Bars: A blockchain-based anonymous reputation system for trust management in VANETs," in *2018 17th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications/12th IEEE Int. Conf. on Big Data Science and Engineering*, New York, NY, USA, IEEE, pp. 98–103, 2018.
[27] Z. Lu, W. Liu, Q. Wang, G. Qu and Z. Liu, "A privacy-preserving trust model based on blockchain for VANETs," *IEEE Access*, vol. 6, pp. 45655–45664, 2018.
[28] Truffle Blockchain Group, "Truffle Suite," 2020. [Online]. Available: https://www.trufflesuite.com/.
[29] Truffle Blockchain Group, "Ganache," 2020. [Online]. Available: https://www.trufflesuite.com/ganache.
[30] ConsenSys Formation, "Metamask," 2020. [Online]. Available: https://metamask.io/.
[31] I. Z. Schlueter, "NPM," 2020. [Online]. Available: http://www.npmjs.com/.
[32] J. Papa, "Lite-server," 2020. [Online]. Available: https://github.com/johnpapa/lite-server.
[33] Ethereum Foundation, "Ethereum," 2020. [Online]. Available: https://Ethereum.org/.
[34] Ethereum Foundation, "Remix ide," 2020. [Online]. Available: https://remix.Ethereum.org/.
[35] Etherscan, "The Ethereum Blockchain Explorer," 2020. [Online]. Available: https://etherscan.io/.
[36] E. Barker and Q. Dang, "NIST special publication 800-57 part 1, revision 4," *NIST, Tech. Rep.*, 2016. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-4/archive/2016-01-28.
[37] S. R. Singh, A. K. Khan and S. R. Singh, "Performance evaluation of RSA and elliptic curve cryptography," in *2016 2nd Int. Conf. on Contemporary Computing and Informatics*, Noida, India, IEEE, pp. 302–306, 2016.
[38] R. K. Nirala and M. D. Ansari, "Performance evaluation of loss packet percentage for asymmetric key cryptography in VANET," in *2018 Fifth Int. Conf. on Parallel, Distributed and Grid Computing*, Solan Himachal Pradesh, India, IEEE, pp. 70–74, 2018.
[39] S. Ucar, S. C. Ergen and O. Ozkasap, "Multihop-cluster-based IEEE 802.11 p and LTE hybrid architecture for VANET safety message dissemination," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2621–2636, 2015.
[40] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, pp. 1–32, 2014.
[41] M. A. Karabulut, A. Shah and H. Ilhan, "Performance modeling and analysis of the IEEE 802.11 MAC protocol for VANETs," *Journal of the Faculty of Engineering and Architecture of Gazi University*, vol. 35, no. 3, pp. 1575–1587, 2020.

[42] A. S. Shah, H. Ilhan and U. Tureli, "Performance and complexity analysis of MAC protocol for VANETs," in *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conf.*, Vancouver, BC, Canada, IEEE, pp. 1081–1086, 2019.

[43] M. A. Karabulut, A. S. Shah and H. Ilhan, "Performance modeling and analysis of the IEEE 802.11 DCF for VANETs," in *2017 9th Int. Congress on Ultra Modern Telecommunications and Control Systems and Workshops*, Munich, Germany, IEEE, pp. 346–351, 2017.

[44] M. A. Karabulut, A. S. Shah and H. Ilhan, "The performance of the IEEE 802.11 DCF for different contention window in VANETs," in *2018 41st Int. Conf. on Telecommunications and Signal Processing*, Athens, Greece, IEEE, pp. 1–4, 2018.

[45] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[46] D. Malone, K. Duffy and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 159–172, 2007.

[47] X. Ma, X. Chen and H. H. Refai, "Unsaturated performance of IEEE 802.11 broadcast service in vehicle-to-vehicle networks," in *2007 IEEE 66th Vehicular Technology Conf.*, Baltimore, MD, USA, IEEE, pp. 1957–1961, 2007.

[48] M. I. Hassan, H. L. Vu and T. Sakurai, "Performance analysis of the IEEE 802.11 MAC protocol for DSRC safety applications," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 8, pp. 3882–3896, 2011.

[49] Q. Wu and J. Zheng, "Performance modeling of IEEE 802.11 DCF based fair channel access for vehicular-to-roadside communication in a non-saturated state," in *2014 IEEE Int. Conf. on Communications*, Sydney, NSW, Australia, IEEE, pp. 2575–2580, 2014.

[50] M. Al-Bassam, "SCPKI: A smart contract-based PKI and identity system," in *Proc. of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, Abu Dhabi, United Arab Emirates, pp. 35–40, 2017.

[51] W. Stallings, *Network security essentials: Applications and standards (international edition)*, 4/e, Pearson Education India, 2011. [Online]. Available: http://thuvienso.bvu.edu.vn/handle/TVDHBRVT/15994.