Computer Science and Information Systems 20(1):137–156 https://doi.org/10.2298/CSIS220921056L

# Secure Cloud Internet of Vehicles Based on Blockchain and Data Transmission Scheme of Map/Reduce

Hua-Yi Lin

Department of Information Management, China University of Technology,
Taiwan, ROC
calvan.linmsa@gmail.com

**Abstract.** Over the past few years, because of the popularity of the Internet of vehicles and cloud computing, the exchange of group information between vehicles is no longer out of reach. Through WiFi/5G wireless communication protocol, vehicles can instantly deliver traffic conditions and accidents to the back end or group vehicles traveling together, which can reduce traffic congestion and accidents. In addition, vehicles transmit real-time road conditions to the cloud vehicle management center, which can also share real-time road conditions and improve the road efficiency for pedestrians and drivers. However, the transmission of information in an open environment raises the issue of personal information security. Most of the security mechanisms provided by the existing Internet of vehicles require centralized authentication servers, which increase the burden of certificate management and computing. Moreover, the road side unit as a decentralized authentication center may be open to hacking or modification, but due to personal privacy and security concerns, vehicle-to-vehicle is not willing to share information with each other. Therefore, this study is conducted through blockchain to ensure the security of vehicle-based information transmission. Moreover, the elliptic curve Diffie–Hellman (ECDH) key exchange protocol and a secure conference key mechanism with direct user confirmation combined with the back-end cloud platform Map/Reduce is proposed to ensure the identities of Mappers and Reducers that participate in the cloud operation, avoid malicious participants to modify the transmission information, so as to achieve secure Map/Reduce operations, and improves vehicle and passenger traffic safety.

**Keywords:** Internet of vehicles, blockchain, ECDH, Map, Reduce.

## 1. Introduction

Recently, with the gradual popularization of 5G and electric vehicles, the cloud Internet of vehicles (CIoVs) has become more feasible. CIoVs are able to connect the vehicle and surrounding devices to share information with each other, and transfer a considerable quantity of obtained data individually to the back-end platform of cloud computing for a huge amount of data calculation and analysis, and then obtain valuable information.

The framework and components of the cloud Internet of vehicles, as shown in Fig. 1, include V2V vehicle-to-vehicle communication, V2P vehicle-to-pedestrian, V2R vehicle-to-roadside equipment, V2G vehicle-to-group communication, V2N vehicle-to-network, V2I vehicle-to-infrastructure and V2X vehicle-to-everything [1][2]. Additionally, the Map/Reduce operation includes a master cloud operation server named master, and several mapper servers, which are cloud mapping operation servers, and reducer servers for cloud aggregation operation servers.

When the vehicle is moving, it can communicate and share information through V2X, and transmit the information to the Internet through the base station or the roadside equipment RSU, and then the base station or RSU forwards the message to the cloud service classifier (CSC) through routers. Subsequently, CSC dispatches the received message to the corresponding platform of the cloud service to accomplish the Map/Reduce computation depending on the service category requested.

In the open wireless network, the on board unit (OBU) of the vehicle has a variety of interfaces to accomplish communication with the cloud service, RSUs and inside devices of the vehicle. Along with the navigation function, the vehicle also provides many road information for the unmanned vehicle and receives instructions from the cloud control center. Therefore, the vehicle terminal OBU is also an important target for hackers to attack or tamper with transmission data. In addition, RSU is an important core node of cooperative vehicle-road operations. RSU can connect the basic traffic equipment such as signal lights and the cloud control platform via wired interfaces. However, through these interfaces, intruders can launch attacks on roadside devices, affecting traffic safety [3][4].

In addition, we cannot certify the reliability of the identity of the cloud service platform selected from the cloud to participate in the implementation of the Map/Reduce operation, which highlights the lack of an integrated and effective data security protection architecture for the cloud Internet of vehicles.

Looking at the current stage, most of the solutions focus on the security of the Internet of vehicles, but do not incorporate the back-end cloud computing services. Many proposed solutions only focus on how to achieve the secure information transmission on the Internet of vehicles. For example, Insaf et al. [5] proposed a certificateless signcryption scheme for IoV. Pandi et al. [6] proposed batch authentication and key exchange protocols for VANETs. Since the plaintext message is available to unauthorized use and even vicious manipulation. Therefore, Jianfeng et al [7]. developed a secure message sharing mechanism based on an attribute encryption technique using blockchain, which is structured by RSUs. Or, most studies only focus on the security research of cloud service platforms. For example, Tian et al. [8] proposed the cloud enabled robust authenticated key agreement scheme and Yuting et al. [9] proposed a cloud data security sharing scheme using blockchain.

To sum up, the current research topic of Internet of vehicles information security only focuses on the single Internet of vehicles and does not combine the architecture and mechanism of information security transmission from the vehicle terminal to the cloud platform. In addition, most of the information security transmission protocols of the Internet of vehicles currently rely on the centralized certificate server CA or the trust authentication server TA. Once TA is damaged or computing resources are insufficient, it may not be capable of providing effective information security services for the Internet of vehicles.

However, the blockchain technology can achieve trust decentralization through consensus algorithm, which can avoid relying on a single CA or TA server, and provide the identity authentication and the trust mechanism for the Internet of vehicles. Behind the digital identity management, blockchain can bring a trusted and unique identity identification to the Internet of vehicles system.

Similarly, the life-cycle information of the device is stored on a distributed ledger, just like it is calibrated for cells in the human body. The key information of the certificate application, certificate issuance, signature check, certificate revocation and other processes can be recorded on the chain to achieve controllable traceability of vehicle production, vehicle registration, property right management, owner identity authentication, IoV equipment authentication and other links.

Based on the trusted identity authentication and security mechanism, the Internet of vehicles system can merge accumulated information by multiple OBU devices to update the traffic environment in real time and provide more precise and real-time information for autonomous vehicles. And drivers don't have to worry about privacy, because blockchain can preserve the privacy of traffic participants by merging with privacy enhancement protocols that provide anonymity and untraceability when sending data.

Accordingly, this study primarily intends to propose a decentralized information security transfer protocol and incorporate the Internet of vehicles to the platform of the cloud service to achieve the secure information delivery on Map/Reduce operations, thus providing a research direction on the cloud Internet of vehicles of information security.

The remainder of this document is organized as follows. Section 2 introduces blockchain and its related research. In Section 3, this study details our proposed blockchain based secure Map/Reduce information transmission. Section 4 describes the secure data transmission analyses and evaluations. Then, in Section 5, we will discuss the status of this study, outline future research options, and conclude this study.

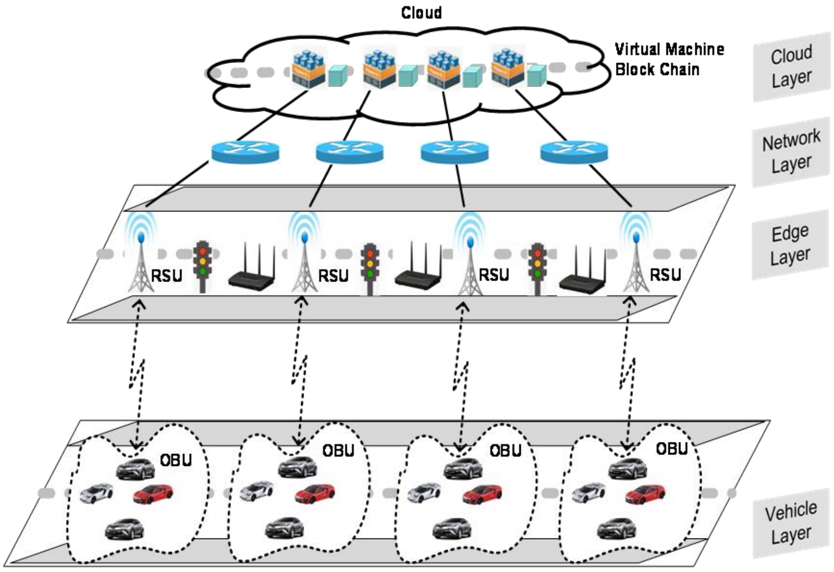

**Fig. 1.** Framework of cloud Internet of vehicles

## 2. Related work

Over the past few years, blockchain has become the mainstream of cryptocurrency. A blockchain is a point-to-point decentralized database. Compared with conventional databases, data is stored in one place, and blockchain distributes these data in numerous minor places, which are called nodes.

Blockchain has the following properties: 1. Decentralization 2. Anonymity 3. Tamper-resistance 4. Data consistency 5. Transparency of information [10][11]. In addition, a blockchain is composed of numerous blocks, which are then tied together to form a blockchain. The data of each block contains two types of data, and there are block header and block body. Fig. 2 depicts the detailed block structure, and Fig. 3 represents our proposed blockchain infrastructure of cloud Internet of vehicles.

Moreover, the block header possesses the following types of data [12][13].

(1). Previous block hash: This hash value is computed by the header of the previous block.

(2). Time stamp: The time stamp for generating the current block.

(3). Nonce: The number of the workload algorithm and the difficulty target of the workload algorithm.

(4). Merkle tree root hash: This is the hash value of the body of the current block. It is also the hash value of the root node of the Merkle tree, which is calculated through the Merkle tree algorithm.

As we know, the Merkle tree is a tree-like formation. Each nonleaf node is marked with a hash value. We use this tree-like structure to obtain the hash value of a string of data, and the time stamp represents what happened at that time on the blockchain. And make sure that each block is linked sequentially. In this case, the Merkle tree root is the root node mentioned above, and the lower node is all transaction records. The block body is equivalent to transactions [14], which includes the following part and trait [15].

(1). Transaction: It is the information of generating the block body, which includes the generation time, the number of accepted transactions, the hash value of the Merkle tree node of the transaction, the address recognized by the transaction, the transaction's digital signature, the index number of the transaction record (used to query the transaction address), data and the record size, etc. Each transaction record has a hash value of a Merkle tree node, which confirms that the transaction cannot be duplicated or forged.

(2). Genesis block. The genesis block is the first block, and its former block hash would be null. When the system generates a blockchain, it creates a genesis block first. Other blocks use the hash of the former block in the header to remember the hash value of the previous block and achieve a complete chain.

(3). Blockchain cannot be modified. Since a change in the transaction record means that the Merkle tree root value of the body in header will be changed, which causes the hash of the entire block header need to change, because the integrity of the chain is cracked. Consequently, the next block's previous block hash must likewise be adjusted, so if someone would like to modify a block's transaction record, they would have to change all subsequent blocks, which is practically impossible.

To sum up, as blockchain refining is rapid and irreversible, once the data is modified in the transmission process, it can be instantly detected, which is very suitable for the distributed cloud Internet of vehicles environment. Therefore, this study would like to employ blockchain to protect the CIoVs security and achieve secure Map/Reduce operation on the cloud platform.

In recent years, many research topics on the Internet of vehicles have been proposed. Azees et al. [16] submitted an anonymous authentication mechanism for a vehicular ad hoc network with security during handovers between RSUs, and consumed less computing power and cost.

Jing et al. [17] provided a lightweight authentication based on blockchain and a key agreement scheme for IoVs that improves the authentication efficiency through a multi-TA model. Authors used blockchain to reserve the authentication information and cross-domain authentication of vehicles, and consequently to defend the privacy information of users. Meanwhile, the recommended scheme employed a lightweight computing operation to lower the authentication time of the vehicle and hence accomplish the entire authentication process.

Bagga et al. [18] designed a batch authentication scheme for IoVs based on blockchain. There are two types of authentications: (1) The authentication of vehicle to vehicle: In a cluster, this mode enables the vehicle to authenticate neighboring vehicles. (2) Batch authentication makes it possible for a group of vehicles to be authenticated by their RSUs. Eventually, the vehicles and RSUs in the cluster can cooperatively create a group key. RSUs then collect secure vehicle data and forms multiple transactions, including the vehicle information and the personal vehicle information for group members.

Cui et al. [19] offered a consortium blockchain based on secure and effective data sharing among vehicles [20]. Within conventional vehicle systems, data sharing should take place with road side units. In this study, the authors leverage a consortium of decentralized technology to reach trackable data sharing between anonymous vehicles and actually accomplish used data sharing. Additionally, combining 5G and blockchain allows data sharing without the use of RSUs. Through delegation, the authors provided an improved challenge validation consensus algorithm in order to make it more appropriate for distributed IoVs. Eventually, an exhaustive analysis demonstrates that the proposed mechanism is effective and secure.

Li et al. [21] designed a blockchain based distribution scheme for mutual healing group keys in a dedicated network of unmanned aerial vehicles. Primarily, the ground control station (GCS) has built a private blockchain where group keys delivered by GCS are stored. Concurrently, the membership certification of a dynamic list of unmanned aerial vehicles Ad-Hoc network is likewise handled using blockchain. Under various attack patterns, a basic mutual healing scheme and an improved protocol were provided from the mechanism of the longest lost chain to retrieve the lost group keys from the node with the assistance of its neighbors.

Moreover, Lu et al. [22] offered a privacy maintaining authentication scheme based on blockchain for vehicle Ad-Hoc networks. Ma et al. [23] proposed a secure announcement sharing based on attributes among vehicles through blockchain. Authors developed a blockchain based on privacy maintaining authentication named BPPA mechanism for vehicular Ad-Hoc networks. In BPPA, this research used the blockchain to continuously and unalterably store all the certificates and transactions to achieve the transparent and verifiable activities of the semi-TAs. Furthermore, this research extended the traditional blockchain architecture to offer a distributed authentication mechanism without the revocation list. In order to reach circumstantial privacy, this research authorized a vehicle to deploy various certificates. A linkage between the certificate and the real identity is ciphered and reserved in a blockchain. In the event of disagreement, the linkage can only be disclosed.

In general, all of the above studies focus on vehicle NET. However, the information on the Internet of vehicles will eventually be transferred to the cloud for massive data processing to obtain value-added information. Therefore, the above studies lack a discussion of the combination of the Internet of vehicles and the cloud computing of the back end. Consequently, this study will propose an information security transmission architecture utilizing a blockchain mechanism combined the vehicle terminal with cloud.

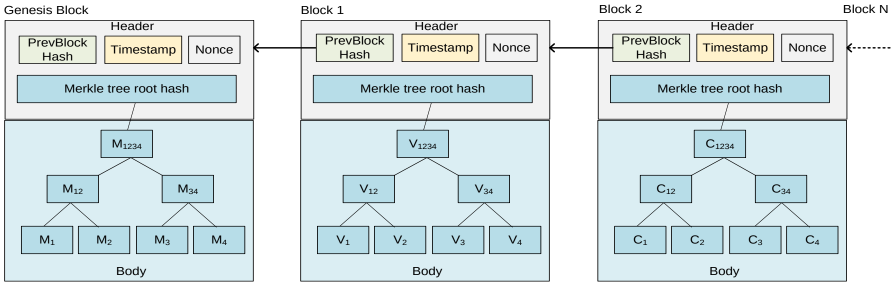

**Fig. 2.** The detailed block structure

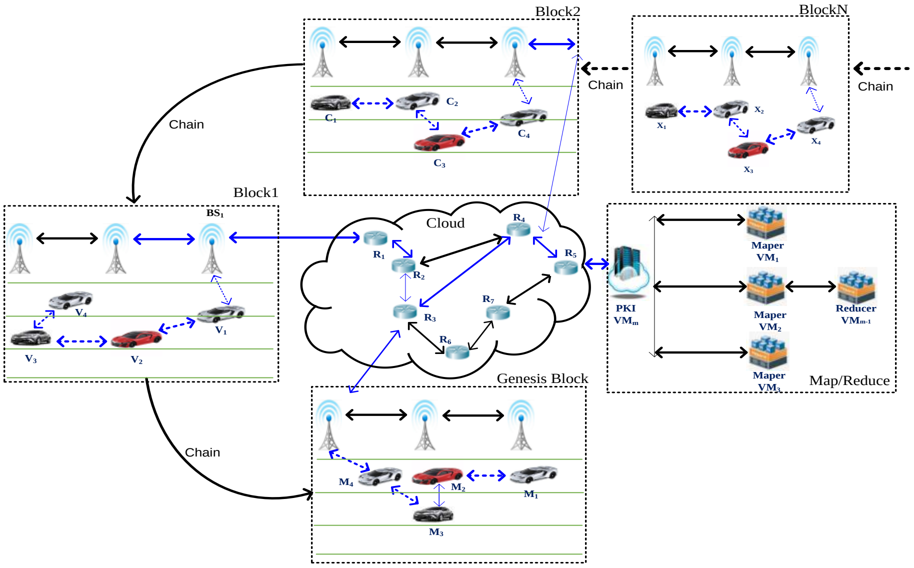

**Fig. 3.** The blockchain infrastructure of cloud Internet of vehicles

## 3. Blockchain based secure Map/Reduce data transmission

Based on the above argument, this study proposes a blockchain based secure Map/Reduce data transmission scheme for CVoTs, which can assure that data will not be tampered during the process of data transmission. Moreover, when moving vehicles would like to transmit gathered data by OBU to each other and the back-end cloud computing service platform through vehicles. It also ensures the information security of cloud computing.

For data security, we adopt multisignature to ensure the authenticity of decentralized transactions without trust centers for blockchain. This study assumes that vehicle $V_1$, $V_2$, $V_3$,..., $V_i$ are within the wireless transmission range of the same RSU. Firstly, the vehicles involved in the operation collectively determine a large prime number $P$ at least greater than 512 bits, a primitive element $\alpha$ in FG($P$) and a hash function $f$. Subsequently, each vehicle $V_i$ selects a private key $k_i \in [1, p\text{-}1]$, calculates and announces the corresponding public key $z_i = \alpha^{ki}$ mod $p$. In addition, the public key $z = \prod_{i=1}^{n} z_i$ of this group is the multiplication of the public key of all the vehicle members. The process of digital multisignature for the secure data transmission via the group vehicle is as follows:

**Step 1.** Individual digital signature: Each vehicle member $V_i$ selects an integer $r_i \in [1, p\text{-}1]$ and computes $w_i = \alpha^{ri}$ mod $p$ to be the value of the commitment, and subsequently broadcasts $w_i$ to all other vehicle members. When all members have broadcast $w_i$, $i$ =1, 2, 3..., $n$. Then each member of the group calculates the following equation for himself.

$W = \prod_{i=1}^{n} w_i \bmod p.$ (1)

Each member of the group utilizes the private keys $k_i$ and $r_i$ to generate an individual signature for the plaintext data $D$ to be transmitted, and obtains the following individual digital signature.

$S_i = k_iD' - r_iw \bmod p\text{-}1.$ (2)

$\{w_i, S_i\}$ is the result of the individual signature and satisfies $D' = F(D)$. Subsequently, $\{w_i, S_i\}$ is securely transmitted to the RSU, which verifies the signatures of each vehicle and combines the individual signature into a multisignature. The role of the RSU is to serve the vehicle and it does not possess any key itself. The RSU then adopts the following equation to verify the individual digital signature $\{w_i, S_i\}$ according to the public key $z_i$ of the vehicle member $V_i$.

$z_i^{D'} = w_i^{w}\alpha^{Si} \bmod p.$ (3)

**Step2.** Multisignature: After the RSU has obtained and verified all individual digital signatures, it can transfer the individual signatures $w_i$ and $S_i$ where $i$ =1, 2, 3..., $n$, and then merge them into a multisignature $\{w, S\}$, which also satisfies

$S=S_1+S_2+\ldots+S_n \bmod p\text{-}1.$ (4)

**Step 3.** The verification of multisignature: Any member can verify the signature of the plaintext $D$ according to the only announced public key $z$. The verification equation is as follows.

$z_i^{D'} = w_i^{w}\alpha^{Si} \bmod p.$ (5)

Lemma: If $z_i^{D'} = w_i^{w}\alpha^{S} \bmod p$, then the multisignature $\{w, S\}$ can be verified and accepted.

Proof: As we know, each vehicle member's signature $\{w_i, S_i\}$ satisfies the following equation.

$z_i^{D'} = w_i^{w}\alpha^{Si} \bmod p.$ (6)

If the above equation is multiplied for $n$ times, where $i$=1, 2, 3…, $n$. This study can infer that the multisignature $\{w, S\}$ is correct as described below.

$\prod_{i=1}^{n} z_i^{D'} = \prod_{i=1}^{n} w_i^{w}\alpha^{Si} \bmod p.$

$\prod_{i=1}^{n}(z_i)^{D'}=(\prod_{i=1}^{n} w_i)^{w}\alpha^{S1+S2+\ldots+Si} \bmod p.$ (7)

$z^{D'} = w^{w}\alpha^{S} \bmod p.$

Here, this study assumes when each vehicle on the move sends data $D_1$~$D_i$, through the routing path $V_i \leftrightarrow V_3 \leftrightarrow V_2 \leftrightarrow V_1$, as shown in Fig. 4. This study adopts the blockchain algorithm to secure the transfer of data. We assume that four vehicles $i$ = 4 and the number of transmitted data blocks are four. Table 1 describes the usage of notations in secure information transfer.

**Step 1.** This study employs the multisignature scheme to confirm the transaction record. First of all, each $V_1$~$V_4$ performs the multisignature on all transmitted data $D_1$~$D_4$. Subsequently, this study adopts SHA1 to compute the hash value of each vehicle's multisignature block to obtain $H(S_{Di})$, and concatenate the multisignature result as Node $0_i=[H(S_{Di})\|(S_{Di})]$, $i$=1~4. For example, Node $0_1=[H(S_{D1})\|(S_{D1})]$, Node $0_2=[H(S_{D2})\|(S_{D2})]$, Node $0_3=[H(S_{D3})\|(S_{D3})]$, Node$0_4=[H(S_{D4})\|S_{D4}]$

**Step 2.** Subsequently, this study combines the two adjacent nodes and performs the hash algorithm to get their parent $\text{Node}_{1[(i+1)/2]} = [\text{H}(\text{H}(\text{S}_{Di})|\text{H}(\text{S}_{Di+1}))||\text{S}_{Di}|\text{S}_{Di+1}]$, $i = 1, 3, 5, 7\ldots$

**Step 3.** Repeat the operation of Step 2 until obtain the root node named Markle root, as shown in Fig. 5.

When a block is corrupted or modified, the Merkle tree root value can be obtained by recalculating from the corrupted node through to the Merkle tree root node path. In addition, we can also determine where the corrupted node is, according to the following steps.

**Step1.** Take $\text{S}_{D1}$, $\text{S}_{D2}$, $\text{S}_{D3}$, $\text{S}_{D4}$ as input and compute the newer hash value $\text{H}^*$ of the root node $\text{Node2}_1$ and verify whether the original $\text{H}(\text{H}(\text{H}(\text{S}_{D1})|\text{H}(\text{S}_{D2}))|\text{H}(\text{H}(\text{S}_{D3})|\text{H}(\text{S}_{D4})))$ is equal to the $\text{H}^*$ result. If they are different, check their children node $\text{Node1}_1$ and $\text{Node1}_2$.

**Step2.** Perform the similar hash operations, if $\text{Node1}_1$ is the same and $\text{Node1}_2$ is different, then this study checks $\text{Node1}_2$'s child $\text{Node0}_3$ and $\text{Node0}_4$.

**Step3.** Perform the similar hash operations, if $\text{Node0}_3$ is the same and $\text{Node0}_4$ is different, then this study checks Node04 and eventually finds out the exact corrupted node.

**Table 1.** The usage of notations in secure information transfer

| Symbol | Description |
|---|---|
| $\text{V}_\text{i}$ | Identification of a vehicles |
| $\text{D}_\text{x}$ | Delivered data of $V_x$ |
| $\text{S}_{\text{Di}}$ | Multisignature of delivered data $D_i$ |
| $\text{R}_\text{x}$ | Identification of a router |
| $\text{H}^*(\text{S}_{Dx})$ | SHA1 hash operation of the multisignature data $\text{S}_{Dx}$, * represents the newer hash operation |
| $\text{H}(\text{S}_{Dx})$ | SHA1 hash operation of the multisignature data $\text{S}_{Dx}$ |
| $\text{VM}_\text{i}$ | Identification of a virtual machine |
| TS | Time stamp |
| $\text{ID}_{\text{VMx}}$ | Identity of a virtual machine $VM_x$ |
| $\text{EK}_K$ | Encipher data utilizing the key of $K$ |
| \|\| | Concatenation operator |

During operation, the proposed method only consumes the $\text{O}_{\log}N$ time complexity of comparison, where $N$ is the amount of data blocks. Additionally, the time complexity of generating this Merkle tree is O($n$) for the number of hash computations.

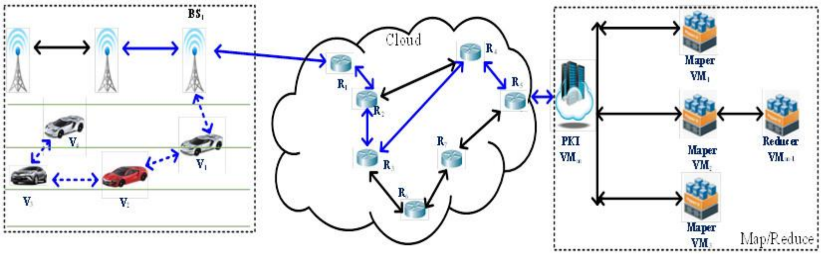

**Fig. 4.** The partial enlarged drawing of the secure data transmission of CIoVs

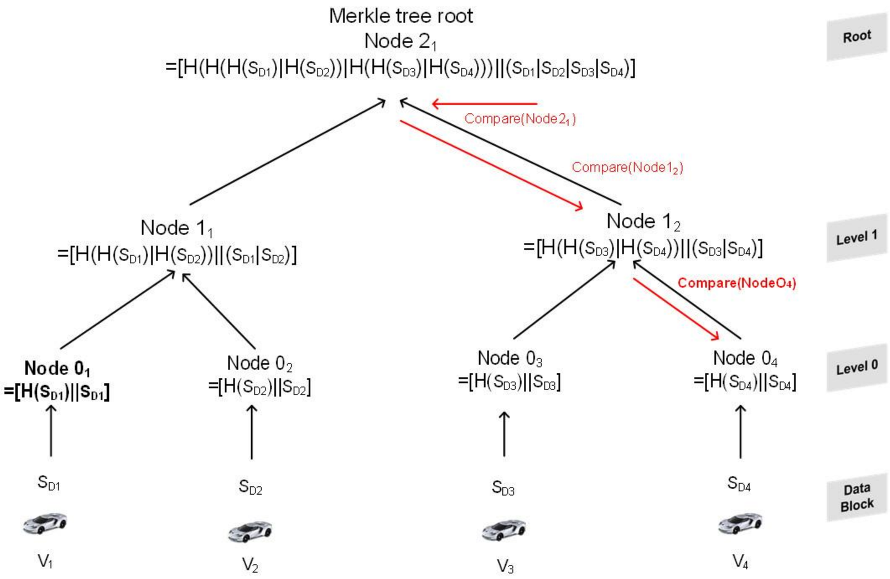

**Fig. 5.** The operation of the Merkle tree root

### 3.1. The Method of the Secure Data Transmission

When the message is transmitted from the vehicle $V_1$ to the cloud via base station $BS_1$, here we assume that $BS_1$ and all routers have passed the security authentication before being deployed. Additionally, the delivered message passes through the following routing path $BS_1 \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow$PKI VM$_m$, as shown in Fig. 4.

Initially, when the vehicle $V_1$ transmits data to the cloud platform through the base station $BS_1$, this study adopts the ECDH protocol to secure the transfer of data. ECDH is similar to the conventional Diffie-Hellman key protocol. Both parties can establish the session key on the insecure channel. Subsequently, the two parties encrypt and decrypt data through the session key. The key length must be 1024 bits to provide a higher level of security. Although ECDH uses the Diffie-Hellman key protocol to implement elliptic curve cryptosystems, ECDH only requires 160 bits of key length and consumes less computing resources to achieve the same security strength [24][25][26]. Therefore, ECDH is very suitable for the network of vehicles that lack computing resources.

In the ECDH key agreement, the base station $BS_1$ and the router $R_1$ need to establish a session key before performing secure communication. Initially, both parties choose the same elliptic curve $y^2 = x^3 + ax + b$, and assign primes belonging to GF($P$) as coefficients $a$ and $b$, where $P$ is equivalent to the Diffe-Hellman generator. $BS_1$ and $R_1$ each have a key pair containing ECC private key $K$, which is a random integer, and perform elliptic curve encryption and decryption with public key $C$, where $C$= $KP$. Additionally, ($K_V$, $C_V$) represents a key pair $V$, and ($K_R$, $C_R$) represents a key pair $R$.

Initially, the base station $BS_1$ selects a private key $K_{BS1}$, and then calculates $C_{BS1} = K_{BS1}P$. The router $R_1$ also selects a private key $K_{R1}$ and calculates that $C_{R1} = K_{R1}P$.

Subsequently, the base station $BS_1$ transmits $C_{BS1} = K_{BS1}P$ to the router $R_1$, and $R_1$ transmits $C_{R1}= K_{R1}P$ to the base station $BS_{1.}$ When each party receives the message sent by the other party, it then multiplies its private key by the received message. Eventually, both sides can figure out the same session key $S_{BS1R1} = K_{\mathrm{BS1}}C_{R1} = K_{\mathrm{BS1}}K_{R1}P = K_{R1}K_{BS1}\ P = K_{R1}C_{BS1}$. In the same way, the session keys between routers can be deduced in this study as $S_{R1R2}$, $S_{R2R3}$ and $S_{R3R4}$, as shown in the blue line of Fig. 4.

### 3.2. The method of Secure Data Transmission in Cloud

After $BS_1$ receives the message sent by $V_1$, in order to protect the security of cloud data transmission, this study employs ECDH key agreement to secure the data transmission in cloud environment. First of all, $BS_1$ exploits the common session key of $BS_1$ and $R_1$, represented as $S_{BS1R1}$, to encrypt and protect the received Merkle tree root HMAC of blockchain, timestamp, type of service and the routing path, then transmits the encrypted output to the $R_1$ router.

#$BS_1 \rightarrow R_{1.}$

$\mathrm{EK}_{S_{BS1R1}}[(BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$

After receiving, $R_1$ decrypts the encrypted data via the common session key $S_{BS1R1}$, and appends its *ID* to the routing path. Subsequently, according to the routing table, $R_1$ and the next router $R_2$ cooperatively figure out the common session key $S_{R1R2}$ utilizing the ECDH agreement, and then $R_1$ enciphers the entire data $[(R_1, BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$ and forwards the enciphered result to the $R_2$ router.

#$R_1 \rightarrow R_2$

$\mathrm{EK}_{R1R2}\ [(R_1, BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$

When $R_2$ obtains the transferred data, it then deciphers the enciphered data utilizing the $S_{R1R2}$ common session key, then and appends itself *ID* to the routing path. Subsequently, according to the routing table, $R_2$ and the next router $R_3$ cooperatively figure out the $S_{R2R3}$ common session key utilizing ECDH, then and $R_2$ enciphers the entire data $[(R_2, R_1, BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$ and forwards the enciphered result to $R_3$.

#$R_2 \rightarrow R_3$

$\mathrm{EK}_{R2R3}[(R_2, R_1, BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$

When $R_3$ obtains the transferred data, it then deciphers the enciphered data via the $S_{R2R3}$ common session key, and appends itself *ID* to the routing path. Subsequently, according to the routing table, $R_3$ and the next router $R_4$ cooperatively figure out the $S_{R3R4}$ common session key utilizing ECDH, and subsequently $R_3$ encrypts the entire data $[(R_3, R_2, R_1, BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$ and forwards the encrypted result to $R_4$.

#$R_3 \rightarrow R_4$

$\mathrm{EK}_{R3R4}[(R_3, R_2, R_1, BS_1)|ToS|TS|[\mathrm{H(H(H(S}_{D1})|\mathrm{H(S}_{D2}))|\mathrm{H(H(S}_{D3})|\mathrm{H(S}_{D4})))||(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$

Once $R_4$ obtains the transferred data, it then deciphers the enciphered data via the $S_{R3R4}$ common session key, then and appends itself *ID* to the routing path. Subsequently, according to the routing table, $R_4$ and the next router $R_5$ cooperatively figure out the $S_{R4R5}$ common session key utilizing ECDH, and later $R_4$ encrypts the entire data $[(R_4, R_3, R_2, R_1, BS_1)|ToS|TS\|[\mathrm{H}(\mathrm{H}(\mathrm{H}(\mathrm{S}_{D1})|\mathrm{H}(\mathrm{S}_{D2}))|\mathrm{H}(\mathrm{H}(\mathrm{S}_{D3})|\mathrm{H}(\mathrm{S}_{D4})))\|(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$ and forwards the encrypted result to $R_5$.

$\#R_4 \rightarrow R_5$

*EKR4R5[(R4, R3, R2, R1, BS1)|ToS|TS|[H(H(H(SD1)|H(SD2))|H(H(SD3)|H(SD4))) ||(SD1|SD2|SD3|SD4)]]*

After receiving, $R_5$ executes the aforementioned similar operations and transfers the enciphered result to the PKI VM$_m$.

$\#R_5 \rightarrow$ PKI VM$_m$

$\mathrm{EK}_{R5\mathrm{VMm}}[(R_5, R_4, R_3, R_2, R_1, BS_1)|ToS|TS\|[\mathrm{H}(\mathrm{H}(\mathrm{H}(\mathrm{S}_{D1})|\mathrm{H}(\mathrm{S}_{D2})) |\mathrm{H}(\mathrm{H}(\mathrm{S}_{D3})|\mathrm{H}(\mathrm{S}_{D4})))\|(\mathrm{S}_{D1}|\mathrm{S}_{D2}|\mathrm{S}_{D3}|\mathrm{S}_{D4})]]$

Similarly, the PKI VM$_m$ deciphers the enciphered data via $S_{R5\mathrm{VMm}}$, then and determines the type of service according to *ToS*. Since the PKI VM$_m$ is the master virtual machine, it is responsible for the Map/Reduce operations. Subsequently, the PKI VM$_m$ initials the secure Map/Reduce operations as follows.

### 3.3. The Secure Map/Reduce Operations by Direct User Authentication using the Conference Key Agreement

For confirmation of identity accuracy of back-end servers participating in cloud computing and the security of data transmission between each other, this study employs the conference key protocol of indirect user confirmation function to obtain a common conference key and then achieve a secure Map/Reduce data transfer protocol. The entire system divides the operations into two parts that are prepare phase and the phase of the distribution of the conference key.

**Prepare phase:**

**First step.** The PKI VM$_m$ selects $N$, $q$, $r$, s and $d$ as parameters, where $N$=$qr$, $sd$=1 mode $L$ and $L$=lcm($q$-1, $r$-1).

**Second step.** The PKI VM$_m$ generates an $n$ dimensional vector $B$=($b_1$,$b_2$,…,$b_\mathrm{n}$), where $1 \leqq b_i \leqq L\text{-}1$ and $1 \leqq i \leqq n$. When $B$ is determined, $P$ can be figured out. Here, this study lets $P$ = ($h^{b1}$ mod $N$, $h^{b2}$ mod $N$,…, $h^{bn}$ mod $N$) = ($h_1$,$h_2$,…,$h_n$), where $h$ is distributed at the root of GF($q$) and GF($r$). Since this system is verified by name, therefore each participant VM$_i$ has a public $ID_i$. Subsequently, VM$_i$ applies to the PKI VM$_m$ for registration of the key pair ($Z_i$, $K_i$) using binary $ID_i$. Where $Z_i$ is the secret key of VM$_{i.}$ and binary code $ID_i$ = ($D_{i1}$, $D_{i2}$,…$D_{ik}$), $D_{ij} \in \{0, 1\}$, $1 \leqq j \leqq k$. Here we can adopt the MAC address as $ID_i$. When the PKI VM$_m$ receives $ID_i$, then it uses a one-way hash function $H$ to compute $ID_i$ so that H($ID_i$)=($x_{i1}$, $x_{i2}$,…, $x_{in}$), where $x_{ij} \in \{0, 1\}$, $1 \leqq j \leqq n$. Subsequently, the PKI VM$_m$ calculates the VM$_i$'s secret key $Z_i$=$(ID_i)^{-d}$ mod $N$.

**Distribution of the conference key:**

This study adopts the PKI VM$_m$ as the master, and VM$_1$, VM$_2$,…VM$_{m\text{-}2}$, VM$_{m\text{-}1}$ are virtual machines. There are $M$ virtual machines joining this operation. Additionally, VM$_1$~VM$_{m\text{-}2}$ are also mappers that participate in Map/Reduce operations, and VM$_{m\text{-}1}$ is the Reducer. The following steps describe the distribution of the conference key, and Fig. 6 describes the detailed procedure.

**First step.** The PKI VM$_m$ selects a key $K$ from 1 to $N$-1 as the conference key. Then, for each participant VM$_i$, the PKI VM$_m$ must calculate $H(ID_i)=(x_{i1}, x_{i2},\ldots x_{in})$, $F_i=\prod_{l=1}^{n} (h_l \bmod N)^{xil} \bmod N = h^{ki} \bmod N$.

**Second step.** After the PKI VM$_m$ calculates $F_i$ for each participant VM$_i$, it picks an arbitrary number $w$ and computes $C_1 = h^{sw} \bmod N$, $C_2 = Z_M h^{H(t,c1)w} \bmod N$, where $H$ is the hash function announced by the system, and two parameters there are a time stamp $t$ and $K_{VMi}=(F_i)^{sw} \bmod N$.

**Third step.** When all participants $VM_i$, where $1 \leqq i \leqq M$-1, have $K_{VMi}$. Then the PKI VM$_m$ can construct the following Lagrange interpolation polynomial. Subsequently, the PKI VM$_m$ broadcasts ($C_1$, $C_2$, $a_0$, $a_1$,…, $a_{m\text{-}2}$, $t$) to $VM_i$.

$$A(x)=\sum_{s=1}^{M-1}(K+ID_s)\prod_{j=1,j\neq s}^{M-1}\frac{(X-K_{VMj})}{(K_{VMs}-K_{VMj})} \bmod N = a_{M\text{-}2}X^{M\text{-}2}+\ldots+a_1X+a_0 \bmod N. \quad (8)$$

**Forth step.** After receiving ($C_1$, $C_2$, $a_0$, $a_1$, …, $a_{M\text{-}2}$, $t$), VM$_i$ calculates h($t$, $C_1$) and verifies whether the following equation is correct.

$$\frac{(C_2)^s}{(C_1)^{H(t1,C1)}} \equiv ID_M \bmod N. \quad (9)$$

If the verification of the above equation is accurate, the participant VM$_i$ can identify the PKI VM$_m$ to avoid a request from a counterfeiter. VM$_i$ subsequently calculates the following equation.

$$K_{VMi} = (C_1)^{ki} \bmod N = h^{swki} \bmod N. \quad (10)$$

Eventually, VM$_i$ can figure out the conference key $K$ of this task using the $A(x)$ polynomial.

$$A(K_{VMi})=a_{M\text{-}2}\, K_{VMi}+\ldots+a_1K_{VMi}+a_0 \bmod N = K + ID_i \bmod N, \text{ and} \quad (11)$$

$$K \equiv K + ID_i - ID_i \pmod N).$$

After the identities of all participants in the Map/Reduce operation are confirmed, the system can employ the $K$ conference key to execute secure Map/Reduce encryption and decryption operations as shown in Fig. 7.

When the cloud computing center PKI VM$_m$ receives data from the router $R_5$, it then uses the hash function to confirm the received data integrity, then and employs the conference key $K$ to encrypt the transmitted data [H(S$_{Di}$)||S$_{Di}$] and deliver the result to the members of joining Map/Reduce operations, VM$_1$, VM$_2$,….VM$_{m\text{-}2}$,VM$_{m\text{-}1}$, where VM$_{m\text{-}1}$ is the reducer and VM$_m$ is the PKI and master.

When the mapper VM$_i$ receives the encrypted data[H(S$_{Di}$)||S$_{Di}$] as shown in Fig. 7-①, and then decrypts it utilizing the conference key $K$ and obtains [H(S$_{Di}$)||S$_{Di}$]. Subsequently, VM$_i$ uses the hash function to calculate the newer H*(S$_{Di}$). If H(S$_{Di}$)= H*(S$_{Di}$), then the data has not been modified during the data transmission, and thus this study can ensure the integrity of transmitted data as shown in Fig. 7-②. Subsequently, VM$_i$ encrypts [ID$_{\text{VM1}}$, TS, (H(S$_{D1}$)|S$_{D1}$)]$_{\text{EK}k}$ and delivers the encrypted result to the Reducer VM$_{m\text{-}1}$ as shown in Fig. 7-③.

When the reducer VM$_{m\text{-}1}$ receives data segments from VM$_1$, VM$_2$,…VM$_{m\text{-}2}$. Each data segment is decrypted utilizing the conference key $K$ to verify the (H(S$_{Di}$)| S$_{Di}$) integrity, where $1 \leqq i \leqq m$-2, using the hash function. If H(S$_{Di}$)=H*( S$_{Di}$), where * represents the newer HMAC result. Then the reducer merges each data segment S$_{D1}$~S$_{Dm\text{-}2}$ to obtain the complete original data $D$ as shown in Fig. 7-④, and subsequently the reducer encrypts the result utilizing the conference key $K$ and sends the encrypted result back to the PKI VM$_m$ to complete the secure Map/Reduce operations as shown in Fig. 7-⑤. The complete secure Map/Reduce operation is as shown in the Fig. 7.

| PKI VM$_m$ | VM$_i$, |
|---|---|
| 1.Selects a key $K$ from 1 to $N$-1 as the conference key<br>2.Calculates $H(ID_i) = (x_{i1}, x_{i2}, \ldots x_{in})$, $F_i = \prod_{l=1}^{n} (h_l \, mod \, N)^{xil}$ mod $N = h^{ki}$ mod $N$ for each participant VM$_i$<br>3.Picks a stamp $t$, $K_{VMi} = (F_i)^{sw}$ mod $N$ and an arbitrary number $w$ and then computes $C_1 = h^{sw}$ mod $N$, $C_2 = Z_M h^{H(t,c1)w}$ mod $N$ | |
| | 4.Then all participants $VM_i$ have $K_{VMi}$. |
| 5.VM$_m$ constructs the Lagrange interpolation polynomial as the equation (7)<br>6.Broadcasts ($C_1$, $C_2$, $a_0$, $a_1$,…, $a_{m\text{-}2}$, $t$) to $VM_i$. | |
| | 7.VM$_i$ calculates h($t$,$C_1$) and verifies whether $\frac{(C_2)^s}{(C_1)^{H(t1,C1)}} \equiv ID_M \, mod \, N$ is correct<br>8.VM$_i$ calculates $K_{VMi} = (C_1)^{ki}$ mod $N = h^{swki}$ mod $N$<br>9.VM$_i$ figures out the conference key $K$ using the equation (7) |

**Fig. 6.** The procedure of obtaining the common conference key

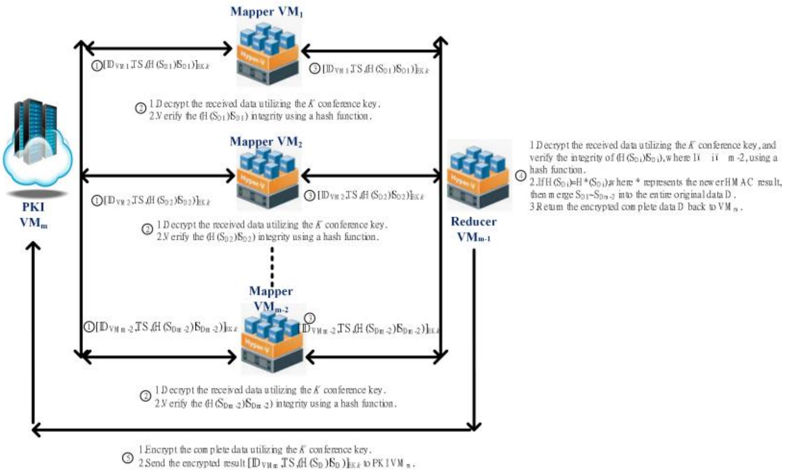
Mapper VM$_1$
①[ID$_{VM1}$,TS,(H(S$_{D1}$)|S$_{D1}$)]$_{EKk}$
③[ID$_{VM1}$,TS,(H(S$_{D1}$)|S$_{D1}$)]$_{EKk}$
② 1.Decrypt the received data utilizing the K conference key.
2.Verify the (H(S$_{D1}$)|S$_{D1}$) integrity using a hash function.
Mapper VM$_2$
①[ID$_{VM2}$,TS,(H(S$_{D2}$)|S$_{D2}$)]$_{EKk}$
③[ID$_{VM2}$,TS,(H(S$_{D2}$)|S$_{D2}$)]$_{EKk}$
② 1.Decrypt the received data utilizing the K conference key.
2.Verify the (H(S$_{D2}$)|S$_{D2}$) integrity using a hash function.
PKI
VM$_m$
Reducer
VM$_{m-1}$
④ 1.Decrypt the received data utilizing the K conference key, and verify the integrity of (H(S$_{Di}$)|S$_{Di}$), where 1≦ i≦ m-2, using a hash function.
2.If H(S$_{Di}$)=H*(S$_{Di}$), where * represents the newer HMAC result, then merge S$_{D1}$~S$_{Dm-2}$ into the entire original data D.
3.Return the encrypted complete data D back to VM$_m$.
Mapper
VM$_{m-2}$
①[ID$_{VMm-2}$,TS,(H(S$_{Dm-2}$)|S$_{Dm-2}$)]$_{EKk}$
③[ID$_{VMm-2}$,TS,(H(S$_{Dm-2}$)|S$_{Dm-2}$)]$_{EKk}$
② 1.Decrypt the received data utilizing the K conference key.
2.Verify the (H(S$_{Dm-2}$)|S$_{Dm-2}$) integrity using a hash function.
⑤ 1.Encrypt the complete data utilizing the K conference key.
2.Send the encrypted result [ID$_{VMm}$,TS,(H(S$_D$)|S$_D$)]$_{EKk}$ to PKI VM$_m$.

**Fig. 7.** The secure Map/Reduce operation

## 4. Secure Data Transmission Analyses and Computing Evaluations

This investigation focuses on the evaluation of the efficiency of the aforementioned models for secure data transmission and conducts several security analyses. In addition, this study also shows that the proposed methods improve the efficiency of secure data transfer, and reduce the recomputing time of blockchain when nodes depart or participate. As well, this study evaluates the convergence time of performing the interpolation polynomial and the verification of multisignature. In addition, our proposed decentralized key management without CA or TA is equipped with a flexible and upgradeable framework. Here is the comprehensive security analysis.

(1) Data integrity: Adding the prev_hash value to blockchain provides a framework for the verification of the integrity of the entire blockchain. If a hacker modifies some past transaction in the previous block $N$-1, the hash value in the later block $N$ will be invalid, even if the hacker modifies the Merkle tree and the root value in block $N$-1. Since, the distribution character of blockchain handles such mismatches in hash [27][28].

(2) Avoid single point of failure: Because the data of blockchain is typically stored among many vehicles in a decentralized network, the system and data resists technical breakdowns and malicious attacks very well. Every node within this network can replicate and store copies of the database. This means there is no single point of failure, although an offline single node does not affect network availability or security [29]. On the other hand, many traditional databases are based on one or more servers and are more susceptible to mechanical failures and network offences.

(3) Transparent: In this open system, the private information in the transaction is encrypted, and all vehicles with maintenance functions collectively maintain it. Anyone can query each blockchain data through the open interface, so the whole system information is highly transparent.

(4) Confirmation of identity in cloud computing: Once VM$_i$ receives ($C_1$, $C_2$, $a_0$, $a_1$,…, $a_{M\text{-}2}$, $t$) to calculate h($t$, $C_1$) and verify whether the following equation is correct.

$$\frac{(C_2)^s}{(C_1)^{H(t1,C1)}} \equiv ID_M \, mod \, N. \qquad (12)$$

If the above equation is accurate, the participant can verify the identity of the PKI VM$_m$ to avoid a request from a counterfeiter.

(5) Better execution efficiency: When a vehicle leaves or joins the network of the CIoVs environment, the Markle root value must be recalculated. Due to the b-tree architecture, a new Markle root value can be obtained by recalculating the branch derived from this node to the root. If there are $N$ vehicle communications and the height of the binary tree is Log$_2N$, so as long as after Log$_2N$ stages of the operation can get the new Markle root value.

(6) The number of stages for computing the entire blockchain: When the vehicle takes part in or departure from the CIoVs environment, this study evaluates that the number of stages required by blockchain must be recalculated. As shown in Table 2, in the best case, the number of stages for recomputing the entire blockchain is only log$_2M$ when a vehicle located in the block number $N$ departs from this system, where $M$ is the

amount of vehicles. However, in the worst case, the amount of stages for recomputing the entire blockchain will increase to $N$ * (Log$_2M$) when a vehicle located in the block number 2 leaves. Generally, the number of stages for recomputing the entire blockchain will increase to ($i$-1) * (Log$_2M$) when a vehicle located in the block number $i$ leaves.

**Table 2.** The number of stages for recomputing the entire blockchain

| Various case | Best case | General case | Worst case |
|---|---|---|---|
| $M$<br>vehicles in each block | A leaving/joining vehicle in block $N$ | A leaving/joining vehicle in block $i$ | A leaving/joining vehicle in block 2 |
| The number of stages | Log$_2M$ | ($i$-1) * (Log$_2M$) | $N$ * (Log$_2M$) |

(7) The convergence time of computing the entire blockchain: For the convenience of discussion, under the condition of fixed blockchain number $N$=4, the convergence time of computing the entire blockchain of this system is discussed under the best case, general case and worst case with the growth in the amount of vehicles. Fig. 8 depicts that as the number of vehicles increases, the convergence time of computing the entire data increases. Especially, in the worst case, the convergence time steeply bumps up, since it must to recalculate $N$-1 blocks. Additionally, in the best case, this system only needs to recalculate 1 block, and therefore the convergence time increases stably.

(8) The convergence time of calculating the conference key: For the sake of convenient discussion, under the condition of the fixed reducer number is 1. This study evaluates that the convergence time of calculating the conference key $K$ needs to perform the interpolation polynomial to figure out the conference key $K$. Figure 9 depicts that as the number of the mapper virtual machine $M$ raises, this system has to calculate the $M$-2 degree of polynomial. As the result, the convergence time of obtaining the conference key $K$ increases as polynomial and exponential growth.

(9) Blockchain mainly generates Merkle tree root through hash function operations, so SHA1 used in this study is compared with various common hash functions for efficient evaluation. We take the file size from 0M bytes to 60M bytes as the input sample, and after 1000 tests the performance is as shown in the Figure 10. The average execution time of the MD5 algorithm for 1000 times is near 230ms, the SHA1 algorithm for 1000 times is near 320ms, and the SHA256 algorithm for 1000 times is near 480ms. In terms of security, SHA256 is obviously the most secure, but it takes considerably longer than the other two [30]. MD5 is comparatively straightforward to generate collisions and be cracked, so SHA1 is the most effective encryption algorithm among the three. In consideration of the need for better computing performance due to the rapid change of vehicles and topologies, this survey employs SHA1 as the hash function.

(10) The verification of multisignature: Any member can verify the signature of the plaintext $D$ according to the only announced public key $z$. As we know, each vehicle member's signature $\{w_i, S_i\}$ satisfies the following equation.

$$z_i^{D\,'} = w_i^{w} \alpha^{Si} \bmod p. \tag{13}$$

When the above equation is multiplied for $n$ times, where $i$ = 1, 2, 3…, $n$. This study can infer that the multisignature $\{w, S\}$ is correct as described below.

$$\prod_{i=1}^{n} z_i^{D'} = \prod_{i=1}^{n} w_i^{w}\alpha^{Si} \bmod p.$$
$$\prod_{i=1}^{n}(z_i)^{D'}=(\prod_{i=1}^{n} w_i)^{w}\alpha^{S1+S2+\ldots+Si} \bmod p. \qquad (14)$$
$$z^{D'}= w^{w}\alpha^{S} \bmod p.$$

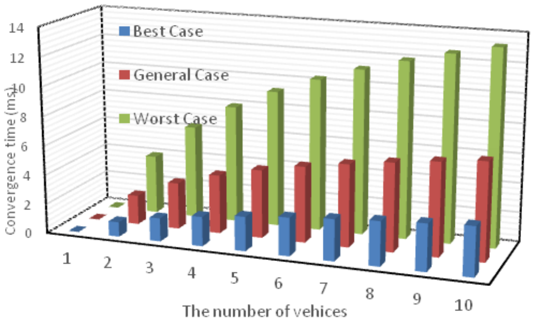

**Fig. 8.** Under the fixed number of blocks *N*=4

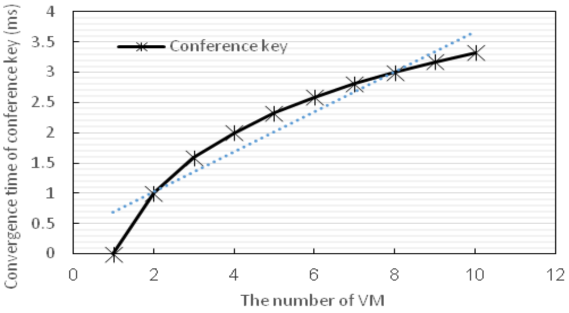

**Fig. 9.** The convergence time of calculating the conference key

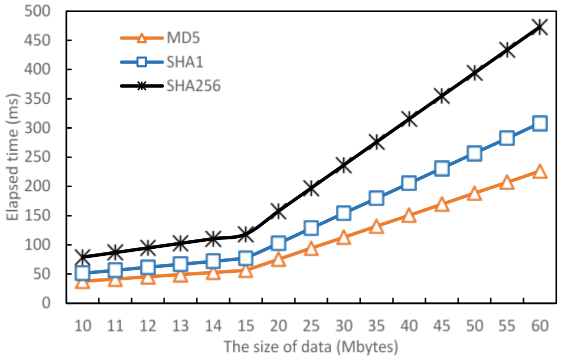

**Fig. 10.** The elapsed time of generating the Merkle tree root

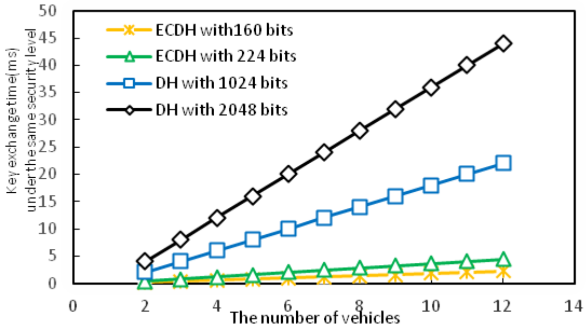

**Fig.11.** Under the same security level, the key exchange time along with the various number of vehicles

(11) The comparison of DH and ECDH key exchange protocols: This study compares the DH and ECDH key exchange protocols to achieve the same security strength under different key lengths, and the time required by both parties to encipher and decipher the transferred data utilizing the session key generated by DH and ECDH. When the transferred data traverses a different number of vehicles, we compare the time required by pairwise encryption and decryption. Figure 11 depicts that under the same security strength, DH needs 1024 (2048) bits. However, ECDH requires only 160(224) bits. Along the transmission path, the required time for the encryption and decryption of adjacent pairwise vehicles increases as the number of vehicles passing by increases. However, experimental results show that the required time for the encryption and decryption of ECDH is less than that of DH.

## 5. Conclusion

Since the development of the cloud Internet of vehicles must encounter the problems and requirements of identity authentication and security trust. This paper depicts the security issues among the vehicle, RSUs, network and cloud, and therefore proposes a blockchain strategy that is different from the conventional centralized CA authentication and privacy protection mechanism to ensure the secure data transfer of the cloud Internet of vehicles. The characteristics of this study are as follows. 1. The proposed scheme can reduce computing resources and costs without third-party verification; 2. Decentralization makes it difficult to tamper with transmitted data; 3. Transactions are secure, private and highly efficient, and 4. Data transparency. Finally, we propose a secure Map/Reduce operations by direct user authentication of the conference key agreement to deal with the security computing of the cloud system, so that the application of blockchain technology among the vehicle, RSUs, network and cloud in CIoVs is more complete.

As the cloud Internet of vehicles integrate 5G, artificial intelligence, huge amounts of data, blockchain and other forward-looking technology of emerging industries, there will be a huge security demand for intelligent construction in the near future.

**Acknowledgment.** This paper is supported by the Ministry of Science and Technology (MOST), Taiwan, under grants MOST 111-2221-E-163-002.

## References

1. Anusha, V., Basudeb, B. Sourav, S., Ashok, K. D., Neeraj, K., Youngho, P.: Blockchain-Enabled Certificate-Based Authentication for Vehicle Accident Detection and Notification in Intelligent Transportation Systems. IEEE Sensors Journal 21(14), 15824-15838. (2021)
2. Angtai, L., Guohua, T., Meixia, M., Jianpeng G.: Blockchain-based cross-user data shared auditing. Connection science 34(1), 83-103. (2021)
3. Qilei, R., Ka, L. M., Muqing, L., Bingjie, G., Jieming, M.: Intelligent design and implementation of blockchain and Internet of things-based traffic system. International Journal of Distributed Sensor Networks 15(8). (2019)
4. Uzair, J., Muhammad, N. A., Biplab, S.: A Scalable Protocol for Driving Trust Management in Internet of Vehicles With Blockchain. IEEE Internet of Things Journal 7(12), 11815-11829. (2020)
5. Insaf U., Noor, U. A., Muhammad, A. K., Hizbullah, K., Saru, K.: A Lightweight and Provable Secured Certificateless Signcryption Approach for Crowdsourced IIoT Applications. Symmetry 11(11), 1386. (2019)
6. Pandi, V., Maria, A., Sergei, A. K., & Joel, J. P. C. R.: An Anonymous Batch Authentication and Key Exchange Protocols for 6G Enabled VANETs. IEEE Transactions on Intelligent Transportation Systems 23(2), 1630-1638. (2022)
7. Jianfeng, M., Tao L., Jie C., Zuobin, Y., Jiujun, C.: Attribute-Based Secure Announcement Sharing Among Vehicles Using Blockchain. IEEE Internet of Things Journal 8(13), 10873-10883. (2021)
8. Tian, L., Xuchong, L., Ruhul, A., Wei L., Meng, Y. H.: RETRACTED ARTICLE: Cloud enabled robust authenticated key agreement scheme for telecare medical information system. Connection science 33(4), 1-XX. (2021)
9. Yuting, Z., Zhaozhe, K., Zhaozhe, K.: BCAS: A blockchain-based ciphertext-policy attribute-based encryption scheme for cloud data security sharing. International journal of distributed sensor networks 17(3). (2021)
10. Dong, W., Huanjuan, W., Yuchen, F.: Blockchain-based IoT device identification and management in 5G smart grid. EURASIP Journal on Wireless Communications and Networking 125. (2021)
11. Houshyar, H., Pajooh, Mohammed, A. R., Fakhrul, A., Serge, D.: IoT Big Data provenance scheme using blockchain on Hadoop ecosystem. Journal of Big Data 8(114). (2021)
12. Razi, I., Talal, A. B., Muhammad, A. Khaled, S.: Trust management in social Internet of vehicles: Factors, challenges, blockchain, and fog solutions. International Journal of Distributed Sensor Networks 15(1). (2019)
13. Suaib, A. A. F. M., Mohiuddin, A., Shahen, S. A. F. M., Adnan, A., Kayes, A. S. M., Ahmet, Z.: Blockchain-Based Authentication Protocol for Cooperative Vehicular Ad Hoc Network. Sensors 21(4), 1273. (2021)
14. Xu, W., Xuan, Z., Wei N., Ren, P. L., Guo, Y. J., Xinxin, N., Kangfeng, Z.: Survey on blockchain for Internet of Things. Computer Communications 136, 10-29. (2019)
15. Zeng, W., Hui, H., Yuping, Z., Chenhuang, W.: A secure and efficient data deduplication framework for the internet of things via edge computing and blockchain. Connection Science 34(1), 1999-2025. (2022)
16. Azees, M., Vijayakumar, P., Deborah, L. J., Karuppiah, M., Christo, M. S.: BBAAS: Blockchainbased anonymous authentication scheme for providing secure communication in VANETs. Security and Communication Networks 2021(6679882). (2021)

17. ing, Z., Xiaoliang, W., Qing Y., Wenhui, X., Yapeng, S., Wei, L.: A blockchain-based lightweight authentication and key agreement scheme for internet of vehicles. Connection Science 34(1), 1430-1453. (2022)
18. Bagga, P., Sutrala, A. K., Das, A. K., Vijayakumar, P.: Blockchain-based batch authentication protocol for internet of vehicles. Journal of Systems Architecture 113(8), 101877. (2021)
19. Cui, J., Ouyang, F., Ying, Z., Wei, L., Zhong, H.: Secure and efficient data sharing among vehicles based on consortium blockchain. IEEE Transactions on Intelligent Transportation Systems 23(7), 8857-8867. (2021)
20. Muhammad, F., Sandi, R., Kyung, H. R.: Decentralized Trusted Data Sharing Management on Internet of Vehicle Edge Computing (IoVEC) Networks Using Consortium Blockchain. Sensors 21(7), 2410. (2021)
21. Li, X., Wang, Y., Vijayakumar, P., He, D., Kumar, N., Ma, J.: Blockchain-based mutual-healing group key distribution scheme in unmanned aerial vehicles ad-hoc network. IEEE Transactions on Vehicular Technology 68(11), 11309–11322. (2019)
22. Lu, Z., Wang, Q., Qu, G., Zhang, H., Liu, Z.: A blockchain-based privacy-preserving authentication scheme for VANETs. IEEE Transactions on Very Large Scale Integration (VLSI) Systems 27(12), 2792–2801. (2019)
23. Ma, J., Li, T., Cui, J., Ying, Z., Cheng, J.: Attribute-based secure announcement sharing among vehicles using blockchain. IEEE Internet of Things Journal 8(13), 10873–10883. (2021)
24. Hua, Y. L.: Integrate the hierarchical cluster elliptic curve key agreement with multiple secure data transfer modes into wireless sensor networks. Connection Science 34(1), 274-300. (2022)
25. Hua, Y., L., Meng, Y. H.: A dynamic key management and secure data transfer based on m-tree structure with multi-level security framework for Internet of vehicles. Connection science 34(1), 1089-1118. (2022)
26. Lin, H. Y., Hsieh, M. Y., Li, K. C.: Flexible group key management and secure data transmission in mobile device communications using elliptic curve Diffie-Hellman cryptographic system. International Journal of Computational Science and Engineering 12(1), 47-52. (2016)
27. Caixiang, F., Sara, G., Hamzeh K., Petr, M.: Performance Evaluation of Blockchain Systems: A Systematic Survey. IEEE Access 8, 126927-126950. (2020)
28. Wei, L., Lijun, X., Ke, Z., Mingdong, T., Dacheng, He., Kuan, C. L.: Data fusion approach for collaborative anomaly intrusion detection in blockchain-based systems. IEEE Internet of Things Journal 9(16), 14741-14751. (2021)
29. Wei, L., Yongkai, F., Kuan, C. L., Dafang, Z., Jean, L. G.: Secure Data Storage and Recovery in Industrial Blockchain Network Environments. IEEE Transactions on Industrial Informatics 16(10), 6543–6552. (2020)
30. Li, Z., Jianbo, X.: Blockchain-based anonymous authentication for traffic reporting in VANETs. Connection Science 34(1), 1038-1065. (2022)

**Hua-Yi Lin** received the Ph.D. degree in Engineering Science from the National Cheng Kung University, Taiwan, in 2006. He is currently an associate professor in Dept. of Information Management at the China University of Technology, Taiwan.

*Received: September 21, 2022; Accepted: November 10, 2022.*