




A Consortium Blockchain-Based Model for Data Sharing in Internet of Vehicles

Qifan Wang¹, Lei Zhou¹, Zhe Tang¹, and Guojun Wang²(✉) 

¹ School of Computer Science and Engineering, Central South University, Changsha 410083, China

² School of Computer Science, Guangzhou University, Guangzhou 510006, China
csgjwang@gmail.com

Abstract. Internet of Vehicles (IoV) provides a broad range of services of data exchange of traffic information to improve the effectiveness of smart vehicles. However, the security issues in Internet of Vehicles are multifaceted: data theft, message tampering and forgery, etc., which results in possibilities of incorrect data sharing. To address above issues, we proposed an efficient blockchain-based data sharing model. In this paper, we leverage the consortium blockchain and enhanced Diffie-Hellman algorithm to build a trust decentralized verifying mechanism, which is designed to secure the data sharing process in IoV. To improve the performance, we optimize the consensus mechanism on our blockchain-based system by decreasing the consensus delay without affecting the correctness of consensus verifying. The security analysis and simulation experiments show that our model meets security requirements and the overhead from our system is acceptable for IoV.

Keywords: Internet of Vehicles · Consortium blockchain · Access policy · PBFT Algorithm · Diffie-Hellman key agreement algorithm

1 Introduction

In recent decades, the widely deployed Internet of Vehicles (IoV) [1] represents a trend of developing smart transportation, because the Intelligent Transportation System (ITS) is playing an increasingly significant role in improving the efficiency of transportation systems. ITS introduces information technology to the transportation infrastructures and aims to improve road safety and traffic efficiency. Since the IoV is a core component of ITS, it develops a new type of self-organizing network consisting of mobile vehicles with sensing, computing, storage and wireless communication capabilities and basic communication facilities on the road [2]. With IoV, it becomes possible to control the whole process of transportation in an intelligent way, to effectively improve traffic safety and facilitate user driving, and also to provide an effective platform for daily transportation or value-added services such as travel and entertainment etc.

There are two main communication modes, that is, vehicle-to-vehicle (V2V) and vehicle-to-roadside units (V2R), respectively, in the traditional IoV architecture. Various of communication mechanisms can be adopted in each model to achieve better performance in different IoV scenarios. However, the main issue in IoV is still about the security. In IoV service scenario, vehicles are constantly moving, which leads to a complex communication environment. There are many security risks involved in the communication process, such as: message tampering, forged identity, message stealing, etc. In recent years, many researchers address those IoV security issues by using identity authentication, information integrity monitoring and other checking approaches [3]. Those secure IoV architecture [4] is divided into four layers: the first layer is the central management organization, which is the certificate authority (CAs), and the second layer is composed of a series of security domains, responsible for managing the encryption-related information. The third layer consists of RSUs (Roadside Units), which are built on the road by following certain rules, and the fourth layer is the vehicle nodes. However, all the above mechanisms could be susceptible to some attacks, such as: the single point of failure [5], data lost, data tampering, etc., and all of them would lead to information leakage, traffic accidents, or other untrusted data sharing problems. In this paper, we proposed a consortium blockchain-based model for data sharing in IoV, and the traditional information transmission mode in IoV is optimized through the data encryption, access policy matching, time stamp and the improved consensus algorithm. Our contributions can be generalized as follows:

- We developed a secure data sharing system based on consortium blockchain, which becomes the core part of distributed vehicle networking communication architecture, and adopted symmetric encryption and Diffie-Hellman key agreement algorithm.
- We optimized the consensus algorithm for our model, which can efficiently improve the consensus efficiency. In addition, we improved the consensus system effectiveness by introducing the access control policy.
- We evaluated the security advantages in our model, and the result of simulation experiment is acceptable for data sharing in IoV.

The remainder of this paper is organized as follows. In Sect. 2, we briefly discuss the related work. Section 3 contains the architecture of data sharing model in IoV. Then we present the detailed design of our model in Sect. 4. In Sect. 5, we give the security analysis and conduct the performance evaluation for our model through the simulation experiment. Finally, we come into our conclusion in Sect. 6.

2 Related Work

IoV faces many opportunities and challenges, many researchers have tried to solve the security problems mentioned in former section in IoV. Zeadally [6] proposed to patch a time-stamp when the vehicle sends messages to others, and

the receiver can detect anti-replay attack by checking the consistency of timestamp. Varshney [7] proposed to add a digital signature in sharing data. When the vehicle sends messages to others, the message is signed by using the private key held by the vehicle, and the receivers use the sender's public key to verify the signature, which can ensure the non-repudiation of the information. But all above approach has a significant problem that key or signature used in a centralized sever, which may lost its power after attacking.

In recent years, blockchain, a decentralized distributed database which generates lots of blocks that store transaction record, has gradually been regarded as an significant technique that can be used in IoV. In order to solve the problem of single point failure, Li [8] proposed an anonymous authentication mechanism, called "CreditCoin", to check the entry of malicious nodes in the Ad-hoc network. In addition, it was created based on the blockchain technology to effectively protect the privacy of users and ensure the security of communication between vehicle nodes. Zhang [9] proposed a scheme about secure data sharing and storage based on a consortium blockchain, it guarantees that the data stored in the RSU is safety in tamper-proof device. Meanwhile, smart contracts are used to limit the triggering conditions for preselected nodes when transmitting and storing data. But these schemes can't solve many problems in IoV.

Blockchain is managed and maintained by all of the participating computational nodes, even part of nodes become untrusted, the entire system should still work. Although the blockchain brings many possibilities for the IoV, the issues of efficiency in consensus mechanism need to be resolved. Thus, Castro and Liskov proposed Practical Byzantine Fault Tolerance (PBFT) [10], which reduced the complexity of the Byzantine protocol from exponential to polynomial, made it possible to apply the Byzantine protocol in distributed systems. Gan et al. [11] proposed an improved practical Byzantine fault-tolerant consensus algorithm, they optimized PBFT's consensus process to improve consensus efficiency and improved the method of PBFT's master node selection. Consensus verified by a few important nodes will significantly reduce the number of messages broadcast in the network. In the digital currency-based applications, the weights can also correspond to the user's currency, thus achieving a consensus mechanism which is similar to the proof of stake (PoS) [12]. A problem that cannot be ignored in the consensus mechanism is sybil attack [13] caused by the free entry and exit of nodes. The consensus based on the proof mechanism is usually applied to the public chain which allows the free access of nodes, and the PoW mechanism is used by Bitcoin and Ethereum. The consensus based on voting mechanism is generally applied to the consortium blockchain authorized by the node.

3 System Model

In our work, the goal of secure model is to build a trust data sharing system in IoV. We predefine a reasonable scenario: A city is divided into several regions according to the partition of urban transportation system, i.e., business center, airport, railway station, as shown in Fig. 1. Each region can be regarded as a

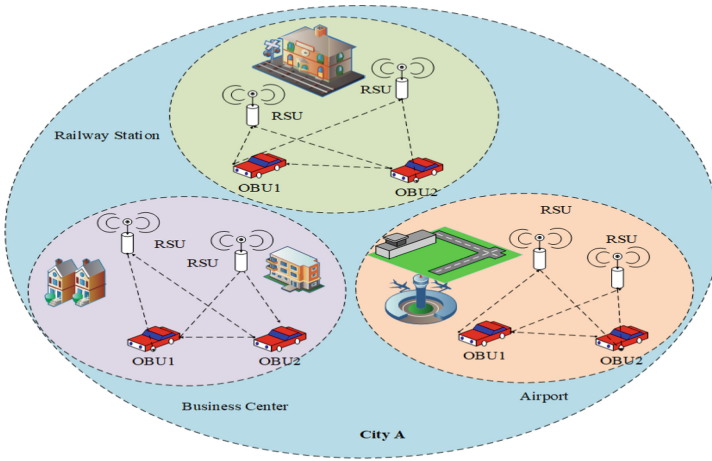


Fig. 1. System architecture diagram.

vehicle convergence zone and some main facilities are selected as RSU. In each specific alliance member, a retail store, a gas station, a toll station, or a transportation station can be selected as the RSU. The RSU periodically broadcasts road condition information at regular intervals based on actual conditions. The general content includes: time, location, and traffic conditions in the coverage area.

In our model, each zone comprises multiple RSUs and OBUs (Onboard Units), as a alliance member of the whole IoV system. The size of each part depends on the local scenario, for example, business center is a region where has a larger daily traffic flow, and the size should be greater in order to access more RSU and OBU nodes. Meanwhile, to maintain a stable IoV network, we assumed a reasonable communication range for each node in consortium. Depending on the capability of existing IoV wireless technology [14], the wireless communication range is about 1KM in normal conditions. Therefore, in our model, with a radius of 1KM, one RSU can be deployed every 2KM to achieve full coverage.

OBU, represents the vehicle in the consortium blockchain network. OBU sends and receives the message from other nodes. To ensure a trust communication, we leverage the consensus mechanism and optimize Diffie-Hellman algorithm to protect the data. In addition, we adopt access control policy to reduce the overhead. The OBU who requests information can leverage their attributes to match others' access policy.

RSU, refers to the fixed building in network, which possesses a greater storage and computational capacity. RSU collects private information including the position and speed of vehicles from sensors, monitors the running status of the vehicle who passes through the range covered by RSU. Meanwhile, RSU is a roadside unit node which is capable of storing and forwarding information, and taking part in the consensus process as an accounting node in the consortium

blockchain. To improve the communication's efficiency and reduce the overhead, RSU is also designed to store the encrypted messages.

After the system initialization, the workflow in this model is designed as follows:

- First, we assume OBU1 is the data provider, and it has the information requested by OBU2.
- Second, OBU2 broadcasts its attribute set and request to other nodes. Other nodes will check the attribute set using their self-defined access control policy when they receive message sent by OBU2. In this model, we make the assumption that OBU1 receives the attribute set sent by OBU2.
- Third, as long as OBU1 matches the request from OBU2, OBU1 will sent cryptography parameters to OBU2 and encrypt the message using the symmetric key. Meanwhile, a transaction between OBU1 and OBU2 is initiated and the related block will be sent to blockchain. Once the transaction succeed, OBU1 sends the ciphertext to the nearby RSU.
- Last, after verifying the transaction block, the RSU then send the corresponding ciphertext to OBU2. OBU2 decrypts the ciphertext with the symmetric key to get the data.

In whole system, the main challenges include the security of information transfer, message tampering and efficiency issues. We adopt consortium blockchain to solve the problem of tampering and over centralization by leveraging this decentralized architecture. We leverage symmetric encryption and DH key agreement algorithm to ensure the security of the messages and keys. We also adopt access control policy and improve PBF-T algorithm to enhance the system efficiency. Moreover, we will make detailed description in the following sections about the implementation of the system.

4 System Design

Since it has different situations in real transportation environment, we select a typical scenario as the Fig. 2 shown. The vehicle node OBU2, traveling in the urban area of the city A, try to collect real-time traffic information of the current vehicle node, including: speed, direction, location, road congestion index and other information, such as, information about the car park. Then we will implement each phase based on the above application.

4.1 Data Sharing Phase in IoV

We assume OBU2 traveling on the road request to get real-time traffic information about the car park. Thus, OBU2 broadcasts in the network to find a OBU which hold its needed data. We leverage the consortium blockchain to process the data sharing transaction which ensure the contents without tampering. Once the transaction about information requested by OBU2 is verified, it will be stored in the block. In addition, the content of the data will be encrypted before

transmission. Using the Diffie-Hellman key agreement algorithm [15], OBU2 and the message sender can obtain a key through the session. We demonstrate the model with simple DH algorithm, but it may be attacked by the man-in-the-middle attack (MITM), however, this can be solved by using some enhanced DH algorithm [16,17] which is out of our goal. The detailed steps are as follows:

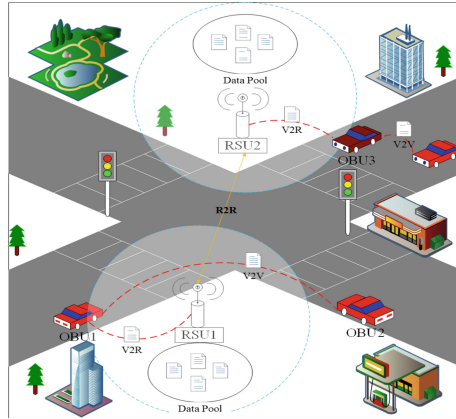


Fig. 2. Business center for data sharing.

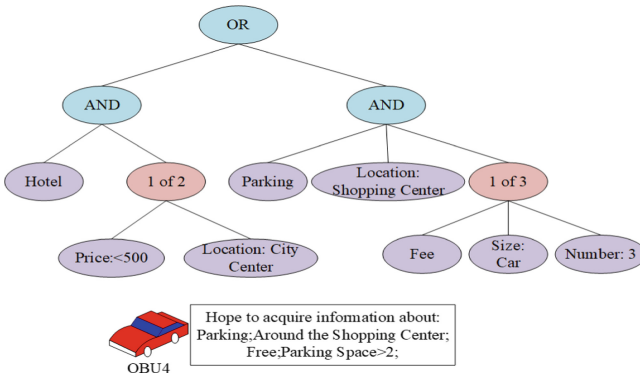


Fig. 3. Access policy tree

Initialization: We define public parameters p and g , and g should satisfy: $2 \leq g \leq p - 1$, those two parameters p, g are open to whole network nodes. The broadcast request $br = \langle p, g, S \rangle$ including: the prime number p and integer g , where g is the generator of p , OBU2's attribute set $S = \{Parking; Shoppingcenter; S'\}$, $S' = \{Free; ParkingSpace > 2\}$.

Attribute Matching: The nodes nearby the OBU2 will receive the request from OBU2, here we refer the existing routing mechanism. We assume that OBU1 has the corresponding information requested by OBU2 and the attribute set S in the OBU2's request matches the access policy tree defined by OBU1. Once successfully matching, the match is successful, as shown in the Fig. 3, the values of the first two leaf nodes in the right subtree in the access policy tree $\{Parking; Location : ShoppingCenter\}$ satisfy the first two attributes in S , $\{ParkingSpace = 3\}$, satisfying the last one in S . For '1 of 2' or '1 of 3', it represents a successful match as long as an attribute in S matches one of the two or three leaf nodes. Therefore, the attribute matches successfully.

Key Agreement: Then, OBU2 produces a private random number A , A satisfies $1 \leq A \leq p - 1$. Meanwhile, it calculates $Y_a = g^A \pmod p$ and sends this value to OBU1. In same workflow, OBU1 chooses a private random number B , which is required $1 \leq B \leq p - 1$ and calculates $Y_b = g^B \pmod p$ and sends it to OBU2. At the same time, the data known by OBU1 are p, g, B, Y_a , and OBU1 obtains the negotiation key K_b by calculating $K_b = (Y_a)^B \pmod p$. Meanwhile, OBU2 receives the corresponding information sent by OBU1 and obtains the key K_a by calculating $K_a = (Y_b)^A \pmod p$. According to the algorithm, $K_a = K_b$. This result can be verified as follows: For OBU2: $K_a = (Y_b)^A \pmod p = (g^B \pmod p)^A \pmod p = g^{(B \times A)} \pmod p$. For OBU1: $K_b = (Y_a)^B \pmod p = (g^A \pmod p)^B \pmod p = g^{(A \times B)} \pmod p$.

Data Sharing: The message m sent by OBU1 is encrypted by the symmetric key K to get the ciphertext c , and the time-stamp t is attached to get $\langle c, t \rangle$ and then sent to the nearby RSU where stored and maintained by RSU. OBU1 uses the negotiation key Y_b to encrypt the symmetric key K , and get the encrypted file $c.key$. OBU1 then sends $c.key$ and index information about the RSU where stored the ciphertext c to OBU2.

OBU2 can get the symmetric key K by decrypting $c.key$ with K_a . Then OBU2 initiates a transaction, the transaction structure includes a version number v , a transaction input tx_{in} , an output tx_{out} , and a lock time $locktime$. After the transaction parties OBU1 and OBU2 confirm this transaction, the related transaction information is sent to the accounting node for a consensus. Only if the consensus is successful, the RSU storing ciphertext c will allow OBU2 to obtain this ciphertext after the transaction is completed. Among them, the RSU confirms whether the transaction is successful by querying whether a block in the blockchain contains this transaction. RSU can verify the existence of the transaction in the block in a short time by utilizing the Merkle root in the block header and Bloom Filter [18].

However, the above mechanism works on the vehicle nodes which are in same region's group (nodes in same group can communicate each other without message forwarding). Considering the data sharing between different group's nodes, the message should be forward by one of RSUs. We assume that there is an OBU3 traveling in the surrounding area of airport, as shown in Fig. 1, it first launches the information requirement related to business center. OBU3 then queries the block record on the consortium blockchain to find the transaction record which

match the requirement by querying the information abstract. To remove the invalid or outdated data, it filters the useless block by calculating the timestamp. After that OBU3 finds a matching transaction record successfully (We assumed the related information comes from OBU1), then OBU3 and OBU1 exchange a secret key, and OBU3 initiates the transaction. Since OBU3 and OBU1 are in different groups without directly communication channel. This can be solved by leveraging the powerful node, RSU, to forward the session message, and the routing algorithm can choose the existing ad-hoc routing approaches [19].

4.2 Consensus Phase

In the data sharing phase, the consortium blockchain verify the correctness of the data transaction between OBU1 and OBU2. The verifying process works in whole accounting nodes, which selected from all of the OBUs and RSUs through PBFT voting mechanism. When the verification is successful, the corresponding block will be generated to record the transaction information about OBU2 and OBU1. However, the traditional PBFT algorithm cannot satisfy the actual IoV scenario due to the long delay of consensus computing. In this section, we propose an efficient practical byzantine fault-tolerant algorithm (EPBFT). Among those accounting nodes, the consensus process of the primary node leader is in a view v , and v is a consecutive numbered integer in each view. Among them, there are the following roles: requester, primary node, and replica node. The three role functions are as follows:

Requester: After the transaction originator OBU2 initiates the transaction, the transaction and the signatures are sent to the network. If the node receiving the information is not the accounting node, the message is broadcast to other nodes; otherwise, the signature should be verified and message will be written to the cache when the signature is correct. The request format is $\langle REQUEST, t, trans, Sig \rangle$, where t is the time stamp to unique the request which promise the valid information applying. The $trans$ represents the transaction, Sig represents the OBU2's signature on this transaction.

Primary Node: In this paper, the primary node is mainly responsible for receiving transactions, and after a period of time, the received transaction generates a block. The primary node is a node selected from the accounting nodes participating in the consensus, and also serves as the private key generation center, and the other replica nodes act as the signer group. A block is generated when a certain amount of transaction $trans$ is stored in the primary node's cache or after a certain time interval Δt .

Replica Node: Replica node is mainly responsible for picking up messages sent from primary node and other replica nodes, executing some corresponding verifications, finally sending the consensus result back to the requester.

A collection of all nodes is represented by R , and each copy in the collection can be represented by an integer from 0 to $|R| - 1$. The ideal number of nodes in the collection is: $|R| = 3f + 1$. f is the maximum number of failed nodes, $|R|$ is the number of nodes. Although more nodes can be deployed, this will only reduce system performance and will not help the consensus process.

View Change: Considering the height h of the current block and the view number v , the selection of the primary node number p is determined by the following formula: $p = (h - v) \bmod N$. When the primary node fails, the consensus request is not initiated within Δt time after the consensus starts, or the primary node is suspected from the node, the view change will be initiated and the view will be switched, then the primary node is replaced. The specific process is as follows:

- When the replica node finds that the primary node is invalid or suspects that the primary node is a malicious node, it broadcast a message about view change to other nodes in the cluster. The format of the message is $\langle view - change, V_{old}, V_{new}, h, p, \Delta t, S_i \rangle$, V_{new} represents the new view number, S_i represents the node that initiated the view change.
- The other replica nodes will verify that V_{old} is the same as the current view number when they receive the message about view change. Then they compute the formula: $V_{new} = V_{old} + 1$. Replica nodes will check whether the primary node is invalid. If the primary node is valid and the proposal sent by the primary node is not considered to be problematic, the message about view change will be ignored; if the verification is passed, they will broadcast view change confirmation message, $\langle view - change - confirmation, V_{old}, V_{new}, h, p, \Delta t, S_i \rangle$.

When the replica node receives the confirmation values from $2f + 1$ replica nodes, then it will start to enter the primary node re-voting phase, and obtains a new primary node according to the steps above.

In the application scenario of IoV, the transaction originator OBU2 is located in a region with many nodes, including RSUs and OBUs. For OBUs, it is a mobile node that appears in various areas according to the user's own driving intentions. Information owned by nodes may lag behind other nodes due to personal reasons or failure of some nodes themselves. Based on the dynamic check mechanism, the Δt_{check} is used as the time interval, and the replica node backs up the data including $v, h, pre.hash$. After the backup data is verified, it can be saved to the same state.

The transaction originator OBU2 initiates a transaction, signs the transaction with its own private key, and broadcasts it to other nodes in the network. If the accounting node receives the transaction, it verifies whether the transaction is legal. If it is legal, the transaction will be recorded in the cache; if other non-accounting nodes receive the information about the transaction, they only need to broadcast to other nodes. Thus, the specific steps of this algorithm are as follows:

- **Consensus Request Phase:** After Δt time or the primary node p stored a number of transactions, the message is broadcasted to other accounting nodes participating in the sharing process. The format of the message is $\langle \langle consensus - request, v, h, p, d, \sigma_p \rangle, block \rangle$, v represents the view number, h is the height of the current block, p is the current primary node number,

block is the block information propagated by the primary node, d is the summary of the block, and σ_p is the signature generated by the primary node p using the ECDSA signature algorithm to verify the integrity of the information.

- **Consensus Confirmation Phase:** After receiving the consensus request sent by the primary node, the replica node $Node_i, i \in 0, 1, \dots, N - 1$, participating in the verification, sends a consensus confirmation message to the other node. A confirmation message is broadcast to other nodes than itself, and messages generated during the consensus request phase and the consensus confirmation phase are written to the message log. The format of the message is $\langle consensus - confirm, v, h, d, Node_i, \sigma_i, Result \rangle$, *Result* represents the result of the verification about the signature. If the result is 1, it means the signature verification is successful; otherwise, it means the signature is invalid. The flag for completion of the consensus confirmation phase is to receive $2f + 1$ acknowledgment messages from different replica nodes, and then issue the *block* from the primary node.

After the rest of the nodes confirm that the current round of consensus computing is completed, the nodes delete the transactions recorded in their own cache and start a new round of consensus.

5 Security Analysis and Evaluation

In this section, we evaluate our system about the security and performance. Through simulate the system, the experimental result compares to previous proposed approaches to show the improvement of our new model.

5.1 Security Analysis

Security is a very serious issue that need to be addressed in IoV application. Incorrect vehicle information can cause some extremely terrible accidents and threaten driver's lives. Due to the importance of users' privacy, users is not willing to share all of their traveling information, including locations, directions, destination etc., with others [20]. Since there are some traditional attacks in IoV [21, 22], we summarize some of them into follow aspects as follows:

Confidentiality Attacks: Eavesdropping [23] is an simple attack targeting confidentiality by sniffing transmitted communication messages and eventually intercepting passwords, etc.

Integrity Attacks: This type of attack contains some typical features, which including message spoofing and tampering, timing attack, etc. The worse situation may cause an accident and threaten users' lives. Messages shared in V2V and V2I communications can be tampered with and influence user's judge in the real-time traffic. Attackers may add delay between packets which causes a reception behind the time and finally, traffic congestion or even accidents.

Privacy Attacks: For example, the vehicle can be tracked and their privacy about location will be leaked. Sybil attack, mentioned in the section of related

work, represents that a malicious vehicle sends wrong numerous messages to other vehicles with different fabricated identities. And the messages transmitted in IoV may be theft by other attackers, what's worse, they can get some important information and even influence the whole system.

Our model enable to against these attacks to promise a security data sharing in IoV through following aspects:

Decentralization: By leveraging the consortium blockchain, nodes in network no need to trust other nodes in generally. Attacks on those traditional data sharing servers will affect whole data sharing process. Fortunately, server in blockchain does not depend on the trusted third-party entity but verified by whole consortium nodes. Meanwhile, times-stamp in the message help us defend timing attack. This decentralized storage system has good scalability and reliability.

Data Security: We adopt Diffie-Hellman key agreement algorithm. It generate a couple secure and private key at both sides, the DH key is privacy, but data can be secure encrypted and decrypted by each other without leaking the cryptography information. It can be proved by the following example: We assume that Eve, who is a attacker, hacked the relevant information, including p, g, Y_a, Y_b . Even in this case, Eve is hard to crack the key K_a or K_b . In fact, if p is large prime number, it's extremely hard for Eve to get A according to the following formula: $g^A \bmod p = Y_a$. The time complexity of the most efficient algorithm for calculating this problem is $O(\sqrt{p})$, so the difficulty in solving the problem in computing ensures the security of the Diffie-Hellman algorithm. Meanwhile, we adopt some improved schemes to address the MITM attack which is mentioned in Sect. 4.1.

Fault Tolerance: Our model proposed the EPBFT consensus mechanism to ensure the system working normally, even 33% of the RSUs or OBUs in the entire system are compromised. According to the EPBFT algorithm, if there are f invalid RSU nodes in the whole network and the total number of nodes satisfies $n \geq 3f + 1$, our proposed system can defend against the security attacks initiated by invalid RSU nodes. It ensure that the final consensus result are not changed based on this proof.

5.2 Performance Evaluation

After we analyzed the requirements of IoV data sharing system, for example, Scalable, Lightweight, Key security, etc., shown in Table 1. Our system meets most of the requirements and reach an acceptable performance. Also, comparing to other existing approaches, our mechanism shows more advantages as the follows.

In this paper, we leverage the consortium blockchain as the key mechanism to construct a decentralize system for IoV data sharing with trust trading partners. Meanwhile, the suitable access policy matching mechanism help the whole system reduce the generation of unnecessary blocks. The improved EPBFT algorithm in this paper makes some improvements based on the traditional PBFT algorithm, and simplifies the algorithm steps to make it suitable for the vehicle networking scenario. Improving the primary node selection and view change

Table 1. Evaluation between our scheme and other schemes

	Azees et al.'s scheme [24]	Liu et al.'s scheme [25]	Dorri et al.'s scheme [26]	Our scheme
Decentralization	NO	YES	YES	YES
Traceable	NO	YES	YES	YES
Scalable	NO	NO	YES	YES
Lightweight	NO	NO	YES	YES
Key security	NO	NO	NO	YES
Low overhead and high efficiency	YES	NO	NO	YES

methods in the algorithm to prevent malicious nodes from leading the consensus process. Through the data synchronization and verification methods, the reliability of each node is guaranteed. It is better than other algorithms in their schemes.

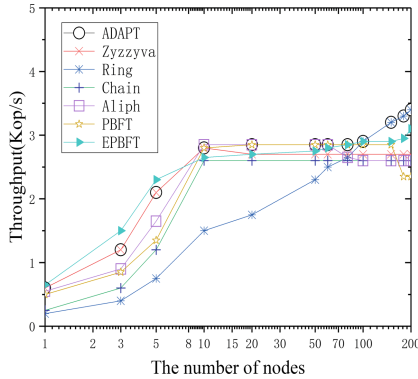


Fig. 4. The comparison of throughput for consensus algorithms

We evaluated the system throughput and delay. Throughput generally refers to the number of transactions processed by the system per unit time. The throughput indicates the ability of the system to withstand the data loading, transactions applying, processing and answering. Due to the difficulties of deploying experimental nodes in real-world IoV scenarios, we conducted simulation experiments to simulate the performance of improved consensus algorithm. The results are reported in Figs. 4 and 5. We can see that our scheme is better than other schemes in the simulated experimental environment. We refer to Bahsoun’s scheme [27], to test the performance of our model by assessing the throughput and the delay when changing the number of nodes. The number of nodes in the Fig. 4 refers to the number of nodes that send transaction requests. In the traditional PBFT algorithm, the client sends a request to the master node. In the improved PBFT algorithm, we set the vehicle node to broadcast transaction

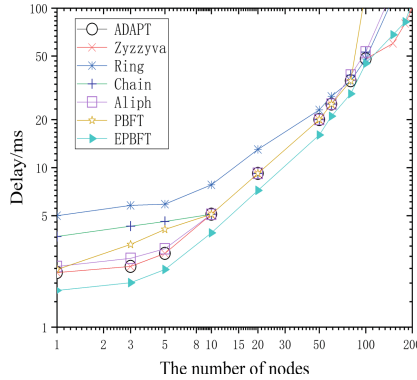


Fig. 5. The comparison of delay for consensus algorithms

records to the whole network. This mechanism is more suitable for P2P networks. When the number of nodes requested is small, the throughput of several schemes in the figure increases with the number of nodes, and the throughput of several schemes is improved. Our scheme has a certain improvement compared with the schemes such as Ring; As the number of nodes continues to increase, the throughput may even decrease. In Fig. 5, as the number of nodes sending transaction records increases, the consensus delay also increases rapidly. Therefore, sending too many nodes will affect the performance of the system. In summary, our scheme has higher throughput than other schemes when the number of nodes is constant.

6 Conclusion

In this paper, we present a secure data sharing system for IoV by leveraging the consortium blockchain. Since the blockchain technology is continuously developed and improved, it provides an effectively mechanism against the data tampering and is applicable to the vehicle data transferring. In our system, we built a secure traffic data sharing model in IoV system. Consortium blockchain in IoV is well matched the different zone in city, and proposed a local trusted data verifying mechanism. In addition, to solve the privacy of data when it is exchanged in whole network, we proposed the corresponding key agreement algorithm for IoV data encrypted/decrypted. Since performance is the key issue of blockchain, we developed the modified EPBFT algorithm, it decreased the consensus delay, and only useful blocks are created in chain, which significantly reduce the overhead during the process of the data transaction in IoV. Simulation result shows that our model performs a secure application and the performance has been greatly improved and compared with the exiting consortium blockchain scenarios. Future researches should focus on the cross-chain technology that are common in the research of blockchain and attempts should be made to apply it to the IoV system

to solve the cross-chain transactions between different blockchains in different cities.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Grant 61632009, in part by the Guangdong Provincial Natural Science Foundation under Grant 2017A030308006, and in part by the High-Level Talents Program of Higher Education in Guangdong Province under Grant 2016ZJ01.

References

1. Contreras-Castillo, J., Zeadally, S., Guerrero-Ibañez, J.A.: Internet of vehicles: architecture, protocols, and security. *IEEE Internet Things J.* **5**(5), 3701–3709 (2017)
2. Wang, Q., Duan, G., Luo, E., Wang, G.: Research on internet of vehicles' privacy protection based on tamper-proof with ciphertext. In: Wang, G., Atiquzzaman, M., Yan, Z., Choo, K.-K.R. (eds.) *SpaCCS 2017*. LNCS, vol. 10656, pp. 42–55. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-72389-1_4
3. Huang, X., Xu, C., Wang, P., Liu, H.: LNSC: a security model for electric vehicle and charging pile management based on blockchain ecosystem. *IEEE Access* **6**, 13565–13574 (2018)
4. Kang, J., Yu, R., Huang, X., Zhang, Y.: Privacy-preserved pseudonym scheme for fog computing supported internet of vehicles. *IEEE Trans. Intell. Transp. Syst.* **19**(8), 2627–2637 (2017)
5. Li, Y., Qi, F., Tang, Z.: Traceable and complete fine-grained revocable multi-authority attribute-based encryption scheme in social network. In: Wang, G., Atiquzzaman, M., Yan, Z., Choo, K.-K.R. (eds.) *SpaCCS 2017*. LNCS, vol. 10656, pp. 87–92. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-72389-1_8
6. Zeadally, S., Hunt, R., Chen, Y.S., Irwin, A., Hassan, A.: Vehicular ad hoc networks (VANETS): status, results, and challenges. *Telecommun. Syst.* **50**(4), 217–241 (2012)
7. Varshney, N., Roy, T., Chaudhary, N.: Security protocol for VANET by using digital certification to provide security with low bandwidth. In: *2014 International Conference on Communication and Signal Processing*, pp. 768–772. IEEE (2014)
8. Li, L., et al.: CreditCoin: a privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles. *IEEE Trans. Intell. Transp. Syst.* **19**(7), 2204–2220 (2018)
9. Zhang, X., Chen, X.: Data security sharing and storage based on a consortium blockchain in a vehicular adhoc network. *IEEE Access* **7**, 58241–58254 (2019)
10. Castro, M., Liskov, B., et al.: Practical byzantine fault tolerance. In: *OSDI*, vol. 99, pp. 173–186 (1999)
11. Gan, J., Li, Q., Chen, Z., Zhang, C.: Improvement of blockchain practical Byzantine fault tolerance consensus algorithm. *J. Comput. Appl.* **39**(7), 2148–2155 (2019)
12. Kiayias, A., Russell, A., David, B., Oliynykov, R.: Ouroboros: a provably secure proof-of-stake blockchain protocol. In: Katz, J., Shacham, H. (eds.) *CRYPTO 2017*. LNCS, vol. 10401, pp. 357–388. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63688-7_12
13. Lin, J., Li, M., Yang, D., Xue, G., Tang, J.: Sybil-proof incentive mechanisms for crowdsensing. In: *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, pp. 1–9. IEEE (2017)

14. Huang, W., Li, P., Zhang, T.: RSUs placement based on vehicular social mobility in VANETs. In: 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), pp. 1255–1260. IEEE (2018)
15. Bresson, E., Chevassut, O., Pointcheval, D., Quisquater, J.J.: Provably authenticated group Diffie-Hellman key exchange. In: Proceedings of the 8th ACM Conference on Computer and Communications Security, pp. 255–264. ACM (2001)
16. Johnston, A.M., Gemmell, P.S.: Authenticated key exchange provably secure against the man-in-the-middle attack. *J. Cryptol.* **15**(2), 139–148 (2002)
17. Gennaro, R.: Multi-trapdoor commitments and their applications to proofs of knowledge secure under concurrent man-in-the-middle attacks. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 220–236. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28628-8_14
18. Guan, Z., et al.: Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities. *IEEE Commun. Mag.* **56**(7), 82–88 (2018)
19. Royer, E.M., Toh, C.K.: A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Pers. Commun.* **6**(2), 46–55 (1999)
20. Samad, A., Alam, S., Mohammed, S., Bhukhari, M.: Internet of vehicles (IoV) requirements, attacks and countermeasures. In: Proceedings of 12th INDIACom; INDIACom-2018; 5th International Conference on “Computing for Sustainable Global Development” IEEE Conference, New Delhi (2018)
21. Sun, Y., et al.: Security and privacy in the internet of vehicles. In: 2015 International Conference on Identification, Information, and Knowledge in the Internet of Things (IIKI), pp. 116–121. IEEE (2015)
22. Abassi, R.: Vanet security and forensics: challenges and opportunities. *Wiley Interdiscip. Rev.: Forensic Sci.* **1**(2), e1324 (2019)
23. Zeng, Y., Zhang, R.: Wireless information surveillance via proactive eavesdropping with spoofing relay. *IEEE J. Sel. Top. Sig. Process.* **10**(8), 1449–1461 (2016)
24. Azees, M., Vijayakumar, P., Deboarh, L.J.: EAAP: efficient anonymous authentication with conditional privacy-preserving scheme for vehicular ad hoc networks. *IEEE Trans. Intell. Transp. Syst.* **18**(9), 2467–2476 (2017)
25. Liu, H., Zhang, Y., Yang, T.: Blockchain-enabled security in electric vehicles cloud and edge computing. *IEEE Netw.* **32**(3), 78–83 (2018)
26. Dorri, A., Steger, M., Kanhere, S.S., Jurdak, R.: Blockchain: a distributed solution to automotive security and privacy. *IEEE Commun. Mag.* **55**(12), 119–125 (2017)
27. Bahsoun, J.P., Guerraoui, R., Shoker, A.: Making BFT protocols really adaptive. In: 2015 IEEE International Parallel and Distributed Processing Symposium, pp. 904–913. IEEE (2015)