*Proceedings of 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN 2018)*

# Research on Task Scheduling Strategy: Based on Smart Contract in Vehicular Cloud Computing Environment

1st Jun Fan, 2nd,* Ru Li, 3rd Shuo Li
*dept. of computer science*
*Inner Mongolia University*
Hohhot, China
1st csfj@imu.edu.cn, 2nd,* csliru@imu.edu.cn, 3rd cslishuo@126.com

***Abstract*—With the development of computer and network technology, on-board computers have more stronger capability of computing and network communication. Multiple vehicles can form an Autonomous Vehicular Cloud (AVC) for the certain need to provide cloud computing services for customers. However, the infrastructure of AVC is open completely, how to guarantee the non-repudiation of task execution information is a very important issue during the process of task execution. Blockchain technology is a proven technology with information security and non-repudiation in distributed environments. Smart contract of Ethereum used to design a kind of task scheduling strategy which is suitable for the AVC environment in this paper. The strategy can guarantee the non-repudiation of task execution information. This paper builds the private chain of Ethereum on simulation vehicles node and creates and deploys the smart contract. Experiments show that the task scheduling strategy is very effective.**

***Keywords—Vehicular cloud, Security, Task scheduling, Blockchain, Smart contract, Non-repudiation***

## I. Introduction

In recent years, with the development of computer and network communication technology, the performance of on-board equipment has greatly improved. In order to utilize the computing, storage and network resources of multiple vehicles and share the data collected by the sensor devices conveniently, the Autonomous Vehicular Cloud (AVC) can be formed among multiple vehicles [1]. It can be used as part of an intelligent transportation system or as an emergency service in a war or disaster environment.

There is a challenge when trying to utilize the resources that owned by vehicles in the company's parking lot into the vehicular cloud. Since the vehicle node that actually pays resources needs to get rewards, when a certain task is executed in the vehicular cloud, how to ensure that the information that the vehicle node actually pays for performing the task will not be tampered with, that is, the labor result will not be taken out by other vehicles. In other words, the execution information of the task must have non-repudiation.

Non-repudiation is an important security issue in cloud computing. Compared with the traditional cloud computing, information transfer between vehicles in the AVC is carried out wirelessly. The infrastructure of AVC is open completely, so the security issues become more prominent and important. The Ref. [2] puts forward that there are five core security goals for trusted cloud services, including confidentiality, integrity, availability, controllability, and non-repudiation. In AVC, most of the relevant researches are on the first four security goals. Only a few of research is on non-repudiation[3,4,5]. Among these researches results, the implementation of non-repudiation requires relying on Trusted Third Party (TTP) which are usually implemented by cloud server. Communication between TTP and vehicles needs to be relayed by Road Side Unit (RSU) in Vehicle to Infrastructure (V2I) structure. In the absence of RSU or RSU damaged, it is not applicable in Vehicle to Vehicle (V2V) structure.

In 2009, Satoshi Nakamoto invented bitcoin, and the blockchain technology was born [6]. For many years of stable operation, bitcoin has demonstrated to us that blockchain technology has excellent security. Ethereum is a new generation of digital currency systems based on blockchain technology and it is a public blockchain platform [7]. Smart contracts are written using the Solidity programming language and can be deployed on Ethereum and triggered by messages. The data generated by the smart contract and its operation can be stored in the blockchain with non-repudiation. This paper proposes a task scheduling strategy (TS-SC) based on blockchain technology and smart contracts, which realizes the non-repudiation of task execution information under the V2V structure in the AVC.

## II. Related Work

In 2010, Olariu first proposed the concept of autonomous vehicle cloud, and preliminary discussed its architecture, application scenarios and problems.

The problem of security is important in the AVC. Due to related researches are still in initial stage, so there are very few research on the issue of non-repudiation.

Ref. [3] proposes a security policy for cloud storage services. The strategy uses an Attribute Based Encryption (ABE) access control strategy and combines Elliptic Curve Cryptography (ECC) to realize the non-repudiation of data storage. The security mechanism proposes in [4] is divided into three stages. The first stage is to initialize and establish the mechanism. The second stage is to perform identity authentication. The third stage is to establish private communication. This mechanism implements multiple security features while reducing the time required to complete communications. Ref. [5] proposes a security mechanism under the vehicular cloud environment. The mechanism mainly includes the authorization process, key generation and distribution process, which can meet basic security requirements, including signing rights, confidentiality, and message integrity, privacy and non-repudiation. However, the above researches on non-repudiation are all based on the V2I structure and must involve the participation of TTP, it does not apply to the V2V structure. This paper proposes a strategy of task scheduling in AVC based on blockchain technology and

This paper is supported by the National Natural Science Foundation of China (Project No. 61363079).

**978-1-5386-4870-4/18/$31.00 ©2018 IEEE**

smart contracts, which can achieve the non-repudiation of task scheduling information under the condition of V2V structure.

The contribution of this article is as follows: We propose the design and framework of TS-SC strategy. We have implemented TS-SC's smart contracts prototype and deploy it on the private chain of Ethereum.

## III. TASK SCHEDULING STRATEGY BASED ON SMART CONTRACT

### A. Scene Description

The design of TS-SC is based on the V2V structure. The scene is a parking lot, the vehicles are in a stationary state and the distance between the vehicles is close. The communication between the vehicles can be transmitted in multi-hop, and all vehicles in the parking lot can realize P2P communication. Regardless of the situation in which the vehicle leaves the AVC during the execution of the task.

### B. Framework of TS-SC

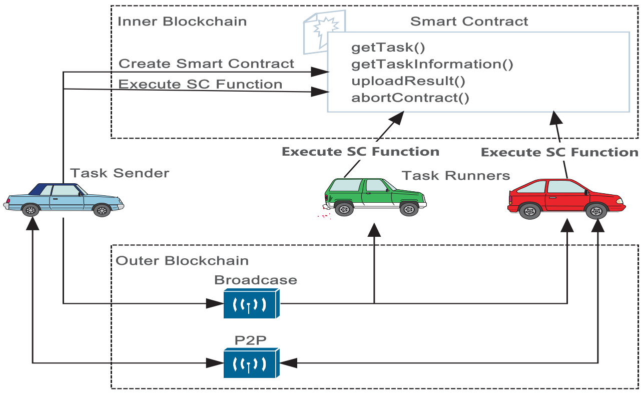


Fig.1 Framework of TS-SC.

Fig. 1 shows the framework of the TS-SC. The smart contract can only be created and initialized by the task sender node. Some operations of smart contracts are completed in the blockchain, and some operations need to be completed outside the blockchain. For example, the task sender node sends the smart contract information to the task runner node in a broadcast manner, and the task runner node returns the task execution result to the sender node through the P2P method to realize communication outside the blockchain.

### C. Task Scheduling Process

Step 1. First, a smart contract is created and initialized by the task sender node. The task basic information and task content need to be set when the contract is initialized.

Step 2. The task sender node sends the smart contract address in broadcast form to the surrounding nodes. The node that receives the smart contract address execute the function of getTaskInformation() to obtain the basic information of task, including the address of task sender node, the amount of compensation, and the node restriction requirements.

Step 3. If the node decides to retrieve the task, execute the function of getTask() to obtain the task content. If successful, the smart contract records the node as a legitimate task runner node.

Step 4. When a node has successfully picked up a task, it starts executing the task.

Step 5. When the execution is completed, the task runner node needs to send the result directly to the task sender node, and at the same time execute the function of uploadResult() to submit the result to smart contract.

Step 6. When the task sender node receives the returned result, the task sender node can destroy the smart contract.

## IV. ANALYSIS OF EXPERIMENTAL RESULTS

We chose six PCs to simulate six vehicle nodes, nodes are under the same coverage of WiFi and can communicate with each other with Point-to-Point (P2P). The client of Ethereum is installed on six PCs, and the private chain of Ethereum is created using custom Genesis block files. Through the creation of block files to adjust the difficulty of mining, so that each PC not only can quickly obtain mining coins and rewards, but also shorten the interval of block packaging. And deploy the smart contract written above. The content of the task is to perform a factorial operation of 10. The calculation result can be returned correctly and relevant information are recorded on the blockchain which has non-repudiation.

## V. CONCLUSION

This article introduces the blockchain technology for the first time to solve the non-repudiation problem of task scheduling information under the AVC environment based on the V2V structure. The TS-SC design and framework are proposed and the TS-SC smart contracts prototype is written. The experimental results show that the proposed strategy is feasible.

## ACKNOWLEDGMENT

This paper is supported by the National Natural Science Foundation of China (Project No. 61363079).

## REFERENCES

[1] Basagni S, Conti M, Giordano S, et al. The Next Paradigm Shift: From Vehicular Networks to Vehicular Clouds[C]// Wiley-IEEE Press, 2013:645-700.

[2] DING Yan, WANG Huai-Min, SHI Pei-Chang, et al. “Trusted Cloud Service[J]”,. CHINESE JOURNAL OF COMPUTERs, 2015, 38(1):133-149.

[3] Hegde N, Manvi S S. “Thesis Proposal Summary: Key Management Authentication and Non Repudiation for Information Transaction in Vehicular Cloud Environments[C]”, IEEE International Conference on Cloud Computing in Emerging Markets, 2017:157-160.

[4] Kim M, Jang I, Choo S, et al. “On security in Software-defined vehicular cloud[C]”, International Conference on Information and Communication Technology Convergence, 2016:1259-1260.

[5] Kim M, Jang I, Choo S, et al. “On security in Software-defined vehicular cloud[C]”, International Conference on Information and Communication Technology Convergence, 2016:1259-1260.

[6] S. Nakamoto. “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2008,Available: https://bitcoin.org/bitcoin.pdf. Accessed on: November 28,2017.

[7] Ethereum Blockchain App Platform. [Online]. Available: https://ethereum.org/. Accessed on: November 28, 2017.