Trusted Computing System Based on Blockchain

Yizhen Wei[a, 1,2*], Jintao Zhu[1], Xiaoxiao Shang[1]

wyz_gs@163.com[a]

School of Computer Science and Artificial Intelligence, Wuhan Textile University, HUBEI WUHAN 430000[1]
School of Computer, Wuhan Vocational College of Software and Engineering, HUBEI WUHAN 430000[2]

Abstract: Cloud computing is an important part of the current digital society. With the popularization of cloud computing, the defects of its opaque calculation process and unverifiable calculation process and results cannot be ignored, and cloud administrators may also be concerned about the calculation results due to their own interests. to modify. Although the emergence of Ethereum smart contracts has solved the problem of insufficient decentralization of calculations and unverifiable calculation results and calculation processes, due to the repeated calculation of all nodes, it leads to low efficiency and intelligent credible calculations with limited computing power. In response to the above problems, proposes a trusted computing system based on blockchain. By adopting off-chain computing + on-chain verification, it not only retains the high-performance advantages of off-chain computing, but also realizes the decentralization of computing. Make the calculation process and results more reliable.

Key words: blockchain; trusted computing; smart contract; cloud computing;

1 Introduction

With the development of science and technology, big data and cloud computing have begun to provide people with better and higher-quality services. While enjoying the services, it should also be noted that big data has brought massive computing needs, and scientists and engineers have responded accordingly. Solutions such as parallel computing, cloud computing, and edge computing are proposed. Most of these solutions rely on large-scale clusters to achieve high performance and fast speed, but most of these solutions fail to make the calculation results credible. In these solutions, users must trust the equipment hardware, software and cloud management personnel All function as expected. However, many things can go wrong, hardware can be damaged, software can be infected with viruses, and even administrators can modify a calculation result for their own benefit. In the final analysis, these services are all run by centralized institutions. These centralized institutions fail to provide a decentralized and transparent service, which makes users unable to verify the calculation results and calculation process well. Since Bitcoin [1] in 2008. Since its appearance, its underlying blockchain [2] Technology has developed rapidly, especially Ethereum [3] The launch of the (Ethereuem) platform enables smart contracts [4] it has become an important research direction of trusted computing. Smart contracts inherit the decentralization, transparency, and verifiability of the underlying blockchain, and achieve a certain degree of trusted computing. However, because the smart contract is based on the mechanism of calculation and verification by all nodes on the

ICBBEM 2023, May 19-21, Hangzhou, People's Republic of China

DOI 10.4108/eai.19-5-2023.2334331

chain, the computing performance of the smart contract cannot be very high, and only some simple calculations can be done. Therefore, the current smart contracts cannot well meet the massive computing needs brought by big data. But the blockchain does provide a good starting point for trusted computing, as it already achieves the following desirable properties:

1) The Ethereum consensus computer provides users with limited trusted computing capabilities in a zero-trust environment.

2) The decentralized ledger of the blockchain provides perfect transparency and stability, and smart contracts also inherit these characteristics.

3) The incentive of virtual currency [5] Inextricably linked to blockchain development, incentives can be used to reward and recruit more participants.

Therefore, the text combines the advantages of trusted computing based on smart contracts and the performance of cloud computing, and proposes a trusted computing system based on blockchain, which is realized by off-chain computing + on-chain verification, which not only retains the decentralization of smart contracts It also retains the high-performance advantages of traditional cloud computing.

2 Related research

The emergence of smart contracts provides new ideas for cloud computing, edge computing and trusted computing. Many scholars have carried out more research on it. Wang Yuxin [6] proposed a trusted distributed computing platform based on blockchain. By publishing computing tasks on smart contracts, calculators participate in bidding, thus realizing a decentralized and distributed cloud computing framework. However, its architecture requires bidders to have the ability to complete the calculation of the task, so its system has high requirements on the hardware of the participants, and its verification of the correctness of the calculation task adopts the method of test case verification. The calculator not only needs to calculate the correct use case but also Test cases need to be calculated, which will lead to repeated and inefficient calculations, and the task issuer also needs to calculate the correct test cases for verification, which contradicts the condition that the issuer does not have computing power. Xu Liang [7] proposed a trusted distributed computing system based on blockchain technology, which divides the blockchain nodes into storage nodes and computing nodes, and then elects the master node and distribution node from the computing nodes to verify the computing tasks respectively. The distribution of the distribution, although the calculation task only requires the calculation node to perform one calculation, but the verification requires the repeated verification of the master node and each storage node. And its scheme is based on the consortium chain, which does not take into account the fact that the master node and the distribution node do evil, which may lead to security problems in the calculation process. Scholars such as Xiaohong [8] also proposed a method to solve the problem of insufficient cpu-intensive computing capacity for smart contracts on the chain. The solution uses off-chain computing, and uses trusted hardware off-chain to improve system security. sex and credibility. But the hardware may also go wrong, and the hardware is managed by a centralized organization/person. Although the system has improved some levels of security and decentralization, it still cannot get rid of the problems of centralization and credibility. Although

the above papers have solved some trusted computing problems, their respective defects and disadvantages are still mainly focused on security and performance. Therefore, this paper combines the advantages of the above papers, makes some improvements, and proposes a trusted computing system based on blockchain by using off-chain computing + on-chain verification. Off-chain computing can guarantee the performance and speed of computing tasks, and on-chain verification can ensure the credibility and security of computing tasks. In order to ensure the efficiency of the verification on the chain, the idea of task splitting is proposed, so that the verification on the chain only needs to verify a small section of the program, and does not require each node to complete the verification of the entire program. In addition, task splitting can also reduce the threshold for the blockchain to participate in verification and calculation, making the system more decentralized.

3 Design and Construction of Trusted Computing Model

In order to achieve better security, this study decided to adopt the public chain model, which does not require nodes, nor does it assume that nodes are trustworthy. The consensus algorithm of the blockchain adopts the pos consensus [9] Compared with the traditional pow consensus, pos does not have problems such as waste of computing power and unstable cycle of generating new blocks [10] Although both pow and pos have consensus and are attacked [11], but it is unlikely for an attacker to spend huge costs to attack the pos consensus at the risk of the blockchain being forked [12], so the pos consensus is relatively safer. The blockchain system mainly provides interfaces for task release, submission of task results, and task verification, as well as query interfaces for task query and calculation results. Users with computing needs can publish tasks to the blockchain through the task publishing interface. After the task is released, the calculator can query the new computing task through the task query interface, and then download the computing task to the local. After calculating the result, pass The interface for submitting task results is submitted to the chain. After being verified by the verifier, the calculator can receive rewards, and the demander can finally query the calculation results through the query interface. The complete task calculation process is shown in Figure 1 below:

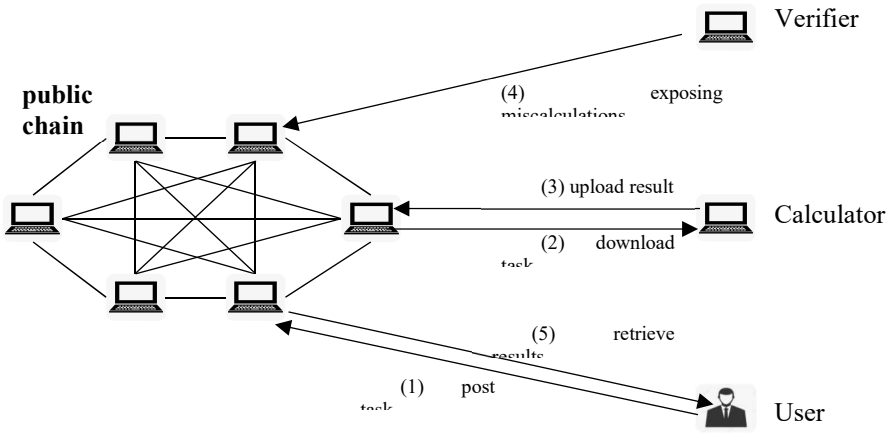

Figure 1: Task calculation process

3.1 Publish calculation tasks

Users with computing needs need to code the smart contract according to the business. After the business coding, they also need to specify the number of splits during the running of the program. With the number of splits, the subsequent calculator will perform the split during the task calculation process. Point-by-point variables are snapshotted, and then packaged and uploaded to the blockchain. For example, in the experiment below, we test the continuous Keccak256Hash calculation of a certain value. If the number of splits is 10 and the total number of calculations is 100 billion, then the calculator needs to pack the Keccak256Hash value for every 10 billion calculations. The more split points, the smaller the resources required for verification and calculation of a single split point, which makes the participation threshold of calculation and verification lower, so that the system can better utilize the idle computing power under the chain. A schematic diagram of task splitting with a split number of 3 is shown in Figure 2 below:

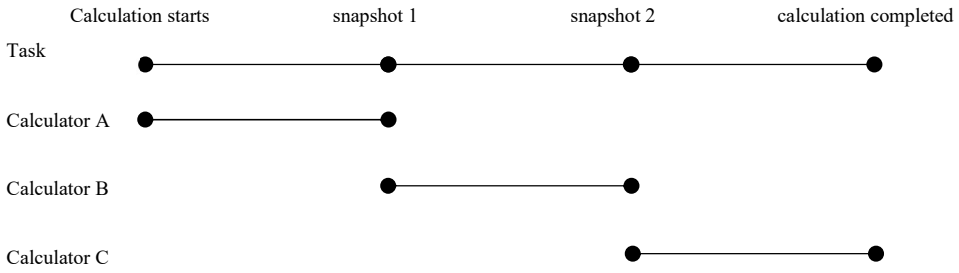

Figure 2: Schematic diagram of computing task splitting

In the figure, it can be seen that a task is divided into three sections, which are calculated by calculator A, calculator B and calculator C respectively, then these three calculators can get one-third of the task reward respectively. After the split point and the smart contract are all prepared, the user can publish the task. When releasing the task, it needs to attach task remuneration. If the task remuneration is too small, no calculator is willing to calculate it. If the remuneration is large, the task may It will have a higher priority, and the time to complete the calculation will be faster.

3.2 Submit calculation results

Users with computing resources can choose to pay a certain amount of deposit in the smart contract to become a calculator. Only after becoming a calculator can they be eligible to submit calculation results. Paying the deposit is mainly based on game theory [13] This can prevent evil people from attacking a large number of registered calculators/verifiers through sybils, resulting in calculation/verification blockage. When the calculator detects a new calculation task on the chain, it will download the smart contract, then perform the calculation in the local environment, and record the value of the program variable at the corresponding stage to take a snapshot. After the calculation is completed, the snapshot array will be compared with the result. And package it and upload it to the chain. After uploading, the calculation result needs to wait

for a period of time to pass the verification. If there is no objection from the verifier during this period, the result is considered correct and the calculator can receive the reward.

3.3 Verify calculation results

Users with verification resources can pay a deposit in the smart contract to become a verifier. After the verifier detects that a new computing task snapshot and results are submitted on the chain, it will verify the task, that is, restore the program variables according to the snapshot data in the snapshot array Arr[n], and then perform operations according to the smart contract until The next snapshot point is n+1, and then compare the calculation result with the snapshot of Arr[n+1]. If they are the same, it means that the calculator is strictly following the smart contract for calculation and there is no cheating. If not, then the verifier needs to submit this error to the chain. After receiving the verification request on the chain, all nodes will jointly verify the snapshot from n to n+1, and there is no need to verify the complete program. When the verification on the chain also finds an error, indicating that the calculator has cheated, then the deposit paid by the calculator will be fined and rewarded to the verifier on the chain. If it fails, it means that the verifier submitted wrong verification information, causing the whole node to conduct a verification, so the verifier's deposit will be confiscated to the node as a block reward.

4 Model Security Analysis

In security analysis, the security of this system depends on two basic assumptions:

1) The POS consensus at the bottom of the block is safe.

2) The potential perpetrators of the system are rational and have limited resources.

Assuming that in the first point, there have been too many related researches on pos consensus security, so this article will not repeat them. Assume that in the second point, potential perpetrators are rational and have limited resources because perpetrators with unlimited resources can publish wrong calculation results through a large number of registered calculators, and relatively few verifiers cannot keep up with the verification speed. It will cause some erroneous calculation results to pass the verification time, so that the system gives wrong calculation results. But because of this assumption, the use of the system avoids this possibility.

Because in addition to the underlying consensus, the system only has external interfaces, so we will analyze the three interfaces one by one.

4.1 Post task

For users who have not paid the deposit, they can only publish tasks. If the perpetrators use Sybil attacks, they will release a large number of tasks with extremely low rewards to cause chain congestion. Then, we can set a minimum remuneration threshold for the calculation task, so that the perpetrators need to pay no less than the threshold remuneration to succeed when publishing the task, thus increasing the cost of this evil method.

If the perpetrator intentionally releases an impossible calculation task (for example, an operation involving division by 0), then the task does not have a correct result, so the task will time out, and the perpetrator can withdraw the principal through the interface. And a large number of

calculators have done a lot of useless calculations. In this case, a certain percentage of handling fee needs to be added. No matter whether the task is completed or not, the handling fee will be deducted. By adding the handling fee, it also increases the cost of doing evil, so that this problem can be solved.

4.2 Computing tasks

Calculators can do evil by submitting wrong calculation results. If the wrong results are passed, the perpetrators will not only get rewards, but users who spend commissions will not get the correct results, which will have a serious impact on the system. But in the second chapter, we know that the answer submitted by the calculator will be verified for a period of time. As long as there is an honest verifier in the whole system, the verifier will submit the error to the chain after detecting an error off the chain, resulting in The perpetrator's deposit was confiscated. In this way, because of the existence of the verifier, the calculator will not risk being confiscated to do evil

4.3 Validation results

Malicious verifiers can submit wrong verification requests to the chain to do evil. Since the verification on the chain is carried out by all nodes, this kind of malicious behavior will lead to repeated verification between nodes, thus making the chain inefficient. But because of the deposit, if the verifier submits wrong verification information, the deposit of the verifier will be fined and rewarded to the node as a block reward. So that doesn't happen either.

5 Experiment analysis

The ubuntu system used in this experiment uses the application chain developed by the go language based on the cosmos sdk. The host is configured with 8 cores and 16G memory, and then simulates different nodes in the real blockchain through connections between different ports. The trusted calculation in the experiment is to perform 100 billion Keccak256Hash calculations on a certain value, and the task splitting is set to 10 times

5.1 Performance comparison experiment

The comparison object of the experiment is full-node computing based on the chain. The experimental method is: count the total number of calls to the Keccak256Hash function for all nodes in the system, on-chain and off-chain. The experimental data obtained from the experiment are shown in the figure below:

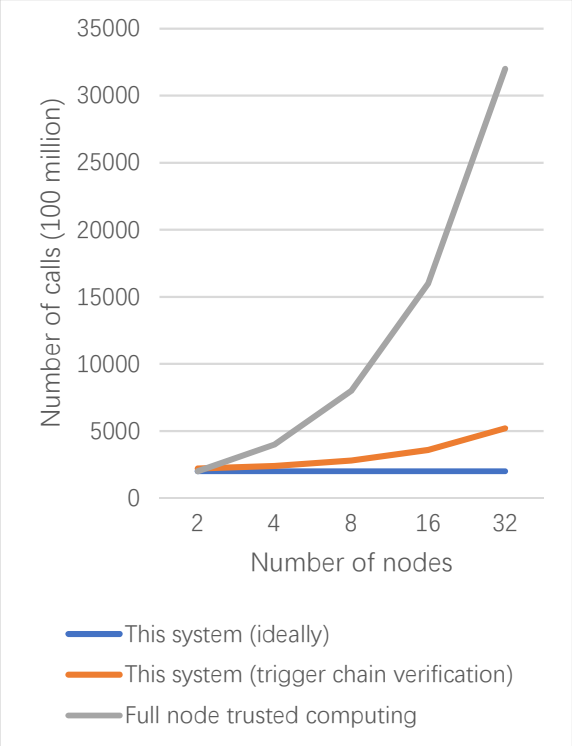

Figure 3: Comparison of experimental data between full nodes and this system

From the above chart, it can be seen that with the increase of the number of nodes, the number of function calls increases sharply based on the trusted computing of the whole node. However, in the ideal case (calculators and verifiers are both running honestly), the number of function calls is the same as that of the number of nodes is irrelevant, because both calculation and verification are processed off-chain, and the total number of calculations only needs to be calculated once off-chain + verified once off-chain, which is 200 billion times. When on-chain verification is required, the page only needs all nodes to verify a snapshot point together, and the number of task splits is 10, and each node only needs to call 10 billion calculation functions.

5.2 Safety comparison experiment

The comparison object of the experiment is the traditional cloud computing system. The experimental method is: by placing the attack script inside the software, and then evaluating the security of the system. The experimental data are shown in the table below:

Table 1: Experimental data table of traditional cloud computing and this system

Comparison Method/Test Object	Cloud Computing System	This System
Can errors be detected	No	Yes
Is the calculation wrong?	Yes	No

safety	low	high

From Table 1, we can see that when the system is under attack, the common cloud computing system is very dangerous, it will enter an unstable state, and the calculation result will be wrong. However, in this system, due to the existence of external verifiers and calculators, tasks can still be calculated and verified as usual, and the calculated results are also in line with expectations.

6 Subsequent optimization

This system basically achieves credible decentralized computing. Compared with traditional on-chain computing, it adopts more efficient off-chain computing. However, the current calculation verification method is only suitable for CPU-intensive calculations. If the calculation requires a large number of io files, the cost of uploading these files to the chain is very high, so subsequent io-intensive calculations also need to be optimized. Moreover, limited by the author's programming level, task splitting can only be split for a specific application at present, and it is impossible to split a general program.

7 Epilogue

Aiming at the disadvantages of centralization, opacity, and unverifiable calculation process and results that are easy to appear in traditional cloud computing systems, and the trusted distributed computing platform based on blockchain in the latest research [5] And a trusted distributed computing system based on blockchain technology [6] The performance and security issues that arise in the paper, this paper synthesizes the advantages of the above papers, and proposes a trusted computing system based on blockchain, which ensures the security and decentralization of the system through blockchain technology, and in the consensus model A more secure pos model is adopted, and the method of on-chain verification + off-chain calculation is adopted. On-chain verification can solve the problems of opaque calculation information and insufficient decentralization of the calculation process. The off-chain calculation method can solve the problems of performance and resource waste. question. A task splitting method has also been added to split a large computing task, which effectively reduces the threshold for participation in computing and verification, making the system more decentralized. Finally, through the design of the economic model, the honest participants in the system can be rewarded, and the perpetrators in the system can be punished, so that the system can maintain a healthy, safe and efficient state for a long time.

Contact information

Name: Wei Yinzhen (1976), female, doctor, associate professor, main research field: knowledge organization and service, E-mail: wyz_gs@163.com

Name: Zhu Jintao (1997), male, master student, main research field: blockchain, E-mail: 1378256170@qq.com

Name: Shang Xiaoxiao (1997), female, master student, main research field: blockchain, E-mail: 70646269@163.com

Fund projects: 1" Research on Trusted Sharing and Service of Scientific Data Based on Blockchain under the Background of Marketization of Data Elements ", funded by National Social Science Foundation (No. 21BTQ074)

2 "Research on Data Service and Security for the Fourth Paradigm of Social Science", funded by Social Science Foundation of Ministry of Education, (No.20YJA870017)

3 "Research on the Integration and Service of Digital Archive Resources of Traditional Culture in Hubei from the Perspective of Cultural and Tourism Integration" (No. : 20ZD096) funded by Hubei Social Science Foundation

4”Research on Vocational Education Evaluation by Big Data Analysis ”（No.:2022C151）funded by Educational Science Planning of Wuhan

References

[1].Linnhoff-Popien C , Schneider R , Zaddach M . Digital Markplaces Unleashed. Berlin, Germany: Springer, 2018

[2].Qiu Xun. Research on the Challenges and Countermeasures of Blockchain Technology[J]. Computer Age, 2021(1):25-28. DOI:10.16644/j.cnki.cn33-1094/tp.2021.01 .006.

[3].Ethereum White Paper. A next-generation smart contract and decentralized application platform［EB/OL］.［2020-02-10］.https://github.com/ethereum/wiki/wiki/White-Paper.

[4].Zhou Jin, Ye Liping, Ni Yiyang, et al. A Review of Smart Contract Research [J]. China New Communications, 2021, 23(3): 37-39. DOI: 10.3969/j.issn.1673-4866.2021.03.017.

[5].Zhang Lu. Research on blockchain-driven supply chain financial innovation from the perspective of game [J]. Economic Issues, 2019(4):48-54.

[6].Wang Yuxin. Design and Implementation of a Trusted Distributed Computing Platform Based on Blockchain [D]. Heilongjiang: Harbin Institute of Technology, 2019.

[7].Xu Liang. Research on Trusted Distributed Computing Based on Blockchain Technology [J]. Modern Computer, 2021(11): 43-47. DOI: 10.3969/j.issn.1007-1423.2021.11.008.

[8].D. Xiaohong, J. Linru, J. Yuan, C. Lin, L. Taoyong and L. Bin, "Intelligent computing scheme of blockchain based on trusted execution environment," 2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 2022, pp. 399-407, doi: 10.1109/ICAICA54878.2022.9844520.

[9].Luo Caihua. Application Research of PoS Consensus Algorithm on Multi-Party Distributed Ledgers [J]. Modern Computer, 2020(17):12-15. DOI: 10.3969/j.issn.1007-1423.2020.17.002.

[10].Chen Dingle. Comparative Research on Blockchain Consensus Algorithms [J]. Software, 2019, 40(4): 219-221. DOI: 10.3969/j.issn.1003-6970.2019.04.048.

[11].Han Jian, Zou Jing, Jiang Han, et al. Research on Bitcoin Mining Attacks [J]. Journal of Cryptography, 2018, 5(5): 470-483. DOI: 10.13868/j.cnki.jcr.000257 .

[12].Wang Jian, Chen Gongliang. Research on Bifurcation of Bitcoin Blockchain [J]. Communication Technology, 2018, 51(1): 149-155. DOI: 10.3969/j.issn.1002-0802.2018.01.027.

[13].Yang Shaojie, Zheng Kun, Zhang Hui, et al. A k-anonymous location privacy protection scheme based on the fusion of game theory and blockchain [J]. Computer Application Research, 2021,38(5):1320-1326. DOI :10.19734/j.issn.1001-3695.2019.10.0654..