利用web3技术来发现训练节点，共包括5种角色，虽然系统看起来很全面，但是没有技术细节和代码。数据以加密的方式从storage 节点传输到comput节点，不是FL。没看懂加密解密的过程

Blockchain-Assisted Privacy-Preserving Data Computing Architecture for Web3

Shaoyong Guo, Fan Zhang, Song Guo, Siya Xu, and Feng Qi

no code

The authors design a blockchain-assisted privacy-preserving distributed data computing architecture to breakthrough the limitations of existing researches. The proposed architecture ensures the secure and trustworthy computing with state channel and computing sandbox.

Abstract

Web3 has received a lot of attention since its emergence. It aims to provide users with more diverse and vivid web services as well as the complete control over their own data. To support the development of Web3, privacy-preserving decentralized data computing schemes need to be studied. In earlier works, blockchain was used for trusted data sharing. However, due to the lack of computing attribute, blockchain is not capable enough to ensure the trustworthiness of the distributed computing process. Besides, the distributed computing method requires a large amount of data transmission and the current privacy protection researches seldom consider the problem of user privacy. In this article, we design a blockchain-assisted privacy-preserving distributed data computing architecture to breakthrough the limitations of existing researches. The proposed architecture ensures the secure and trustworthy computing with state channel and computing sandbox. We also design a sandbox location obfuscation method based on onion routing technology, making it difficult for attackers to identify the sandbox location or infer user privacy. Our solution fully considers the characteristics of Web3 and can well support the diverse Web3 applications.

Introduction

Web3 has introduced a broad and bright future for each of us, where web users are clothed with the authority of managing their network data and no third-party can make use of it without the data owner's permission [1]. The potential of Web3 is fascinating, however it is still in its infancy. Part of the reason is that the core of Web3 is to build applications on blockchain. As infrastructure, blockchain has established a trusted decentralized storage structure for network data. But how to process the data in order to support the on-chain Web3 applications remains a question. To support and speed up the development of Web3, a decentralized data computing architecture is in need.

Web3 users mostly use web services through devices like smartphone or laptop whose local storage, computing, and communication resources are limited. To support their application needs, third-party computing services are required. However, semi-trust computing servers will bring new risks to the authenticity of computing results. And blockchain is unable to ensure the trust of computing process due to the lack of computing attribute. Besides, the cooperation between storage and computing servers requires a large amount of data transmission. By monitoring and analyzing the network traffic, an attacker can infer user's information and steal user privacy [2].

In order to address the aforementioned problems, we establish a privacy-preserving data computing architecture for Web3 based on blockchain and onion routing technology. Our contributions are as follows.

We propose a blockchain-assisted privacy-preserving distributed data computing architecture for Web3. The complex interaction during data computing process is simplified as multiple data transmissions between data senders and data receivers. Through the sharing of computing, communication and storage resources, various Web3 applications are supported.

We propose a state channel-based trusted computing sandbox for distributed Web3 applications. Each task participant establishes a secured local computing environment and performs the computing task collaboratively through state channel. State channel expands the computing attribute of the blockchain, and realizes the trusted data transmission during computing process.

We propose a sandbox location obfuscation method based on onion routing technology. By deploying the proposed distributed computing architecture on an onion routing network, it is more difficult for attackers to analyze network traffic, thereby stealing user information and destroying computing tasks. Through theoretical analysis and experimental verification, the safety of the method is illustrated.

The rest of the article is organized as follows. We next present the related works of this article. Following that we introduce the blockchain-assisted privacy-preserving data computing architecture. Then we introduce the trusted computing sandbox and location obfuscation method. The following section completes the security analysis and experimental verification. The final section gives the conclusion.

Related Works

Web3

The idea of Web3 has penetrated into our lives, but its related researches are still in the early

Digital Object Identifier:
10.1109/MCOM.001.2200408

Shaoyong Guo, Fan Zhang (corresponding author), Siya Xu, and Feng Qi are with Beijing University of Posts and Telecommunications, China; Song Guo is with the Hong Kong Polytechnic University, Hong Kong.

stage. Liu *et al.* provided the first academic definition for Web3 and designed the first interoperable platform that can be used in the Web3 era, laying a good foundation for other Web3-related researches. But it mainly aimed at Web3 application providers, without considering the privacy of Web3 users [3]. Sarathchandra *et al.* designed an Ethereum based decentralized social network architecture. The architecture provides users with more privacy and data ownership, but causes low efficiency due to the throughput limit of blockchain [4]. Even though there are not many researches focused on Web3, many solutions of the decentralized computing architecture have been presented, which are very useful for the Web3-related researches.

Blockchain-Assisted Distributed Data Computing

Jia *et al.* designed an industrial IoT architecture based on blockchain and federated learning. They developed a privacy-preserving data aggregation scheme to achieve multi-protections of data in distributed data sharing and model sharing [5]. Yin *et al.* improved the federated learning-based data sharing scheme by introducing algorithms such as function encryption, and proposed a new hybrid privacy-preserving approach that provides a more secure and efficient data sharing service [6]. Peng *et al.* focused on the security issues and proposed VFChain, a verifiable and auditable federated learning framework based on the blockchain system, ensuring the correctness of training procedure [7]. Cao *et al.* proposed a federated learning framework using directed acyclic graph (DAG) based blockchain, which solved the problems of device asynchrony and anomaly detection in federated learning [8]. However, the above schemes only studied the federated learning based distributed computing architecture. Limiting the distributed computing method to federated learning lowers the scalability of the architecture. Thus, it is not capable enough to support the various computing needs of Web3 applications.

Privacy-Preserving Distributed Data Computing

In the aspect of data privacy, Zhang *et al.* proposed an industrial Internet privacy-preserving data sharing framework interfering with the original data by differential privacy [9]. K. Tjell *et al.* proposed a fully distributed private aggregation protocol that not only drastically reduces the communication and computation overhead, but also is resilient to node dropouts [10]. Zhou *et al.* proposed a novel privacy-preserving federated learning framework for edge computing, where each client and the application server adds noise before sending the data in order to protect data privacy [11].

In the Web3 scenario, the privacy may not only be leaked by user itself, but also by the communication process with others due to their social attributes [12]. So, to realize the privacy-preserving distributed data computing for Web3, we not only need to ensure user's control over their own data, but also need to ensure the privacy of the entire process of distributed data computing. Although the above schemes realized the privacy-preserving data sharing, they only focused on the privacy of original data, overlooking the privacy during data computing.

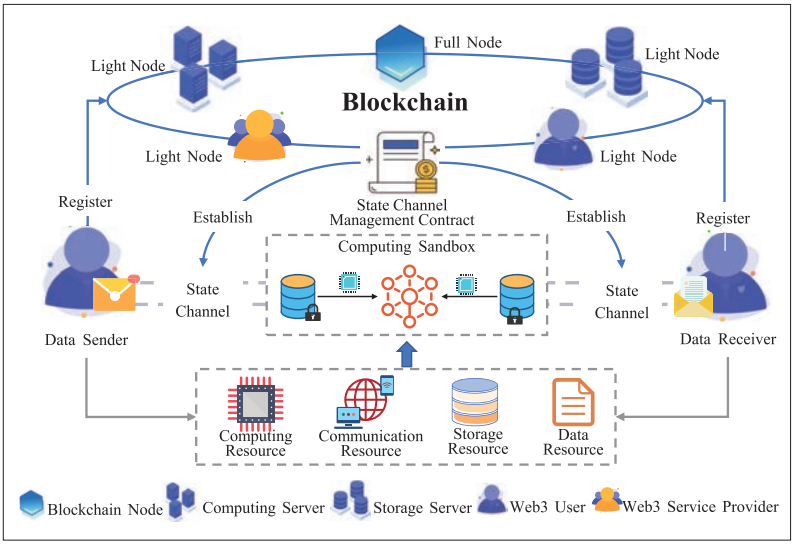

FIGURE 1. Blockchain-assisted privacy-preserving data computing architecture for Web3.

Blockchain-Assisted Privacy-Preserving Data Computing Architecture for Web3

5个角色

This section describes the working principle of the proposed architecture, including the general architecture, the design of smart contracts and workflow.

General Architecture

The proposed architecture involves five types of participants, as shown in Fig. 1.

Web3 Users: Individual users who have limited resources and intend to obtain high-quality Web3 services. As light nodes on the blockchain, Web3 users only store a part of the blockchain data. They don't participate in the consensus, but can use some upper-layer functions on the blockchain through smart contracts.

Storage Servers: Devices that have long-term idle storage resources and agree to provide storage services. They are also light nodes on the blockchain. And IPFS (Inter Planetary File System) technology is often used to ensure data security and storage credibility.

Computing Servers: Devices with long-term idle computing resources. They are light nodes on the blockchain and agree to undertake the assigned computing tasks.

Blockchain Nodes: Devices that belong to the blockchain system. They are full nodes which store all the on-chain data and support the blockchain operations including consensus.

Web3 Service Providers: Individuals or enterprises that develop Web3 applications and provide Web3 services. They are in charge of designing the distributed Web3 computing tasks and providing the required information for task execution.

In the proposed architecture, blockchain provides a medium by which Web3 users can subscribe the Web3 application services and assign the computing tasks to computing servers.

The execution of computing tasks relies on the cooperation between multiple servers. The complex interaction between two parties can be simplified as the data transmission operations between data senders and data receivers.

The trusted data transmission is realized through the state channel. As a blockchain capacity exten-

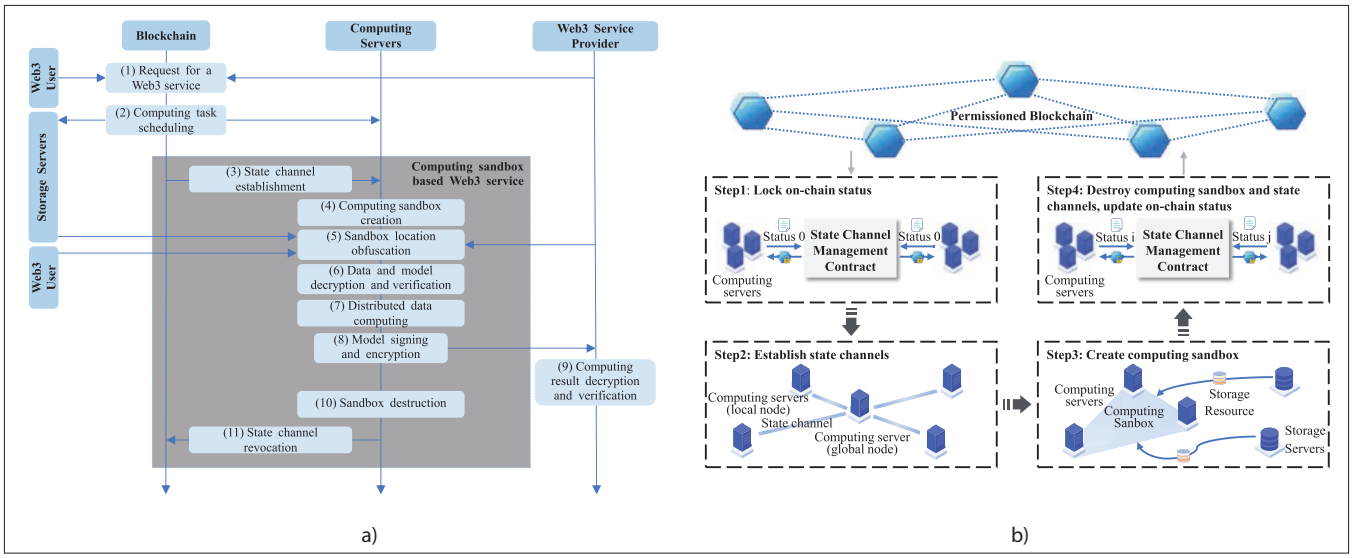

FIGURE 2. Distributed data computing process based on state channel and computing sandbox: a) overall workflow; b) computing sandbox workflow.

sion technology, it not only preserves the trust attribute of blockchain but also avoids the large amount of computing overhead caused by consensus.

In the aspect of data privacy, the asymmetric encryption algorithm is introduced. The data needs to be transmitted is encrypted with the public key of the data receiver, which prevents the leakage of user's privacy.

The Design of Smart Contracts

The main functions of the proposed architecture are performed through smart contracts. In this section, we describe the details of register contract and state channel management contract.

Register Contract: The register contract is responsible for on-chain account registration of the light nodes on blockchain, aiming to record the information of nodes and resources.

The registration information is composed of basic user information, communication resource information, storage resource information, computing resource information and data resource information. The basic user information is forced, and the rest is optional. Basic user information includes the public key of the node, network address of the device, and so on. Communication resource information reflects the communication situation of the device, including bandwidth, and so on. Storage resource information includes the main parameters of the storage device, such as the resource type and usable capacity. Computing resource information includes the main parameters of the computing devices, such as the resource type and the number of CPU (Central Processing Unit) cores. Data resource information includes the data abstract, ownership, storage address, and so on, which are associated with data storage and usage.

State Channel Management Contract: The state channel management contract is responsible for establishing and revoking state channels. Besides, it also acts as a regulator when there are arguments between the two parties of state channel.

To establish a state channel, both parties need to send a request to the state channel management contract respectively to lock their on-chain status. In the meantime, their off-chain status mapping is created, which updates along with the interaction between the two parties. When they want to revoke the channel, the newest status needs to be sent to the state channel management contract. Unlike consensus, state channel designs the challenge period mechanism to guarantee the credibility of off-chain operations. During the set time period, the authenticity of the current status can be challenged. If successful, the on-chain status will be updated according to the newly uploaded status.

Workflow

The workflow of the proposed architecture is shown in Fig. 2a.

Request for a Web3 Service: The Web3 user sends a request to the blockchain to establish a computing task according to the resource requirements provided by the Web3 service provider, including data type, the type of computing resource, and so on.

Computing Task Scheduling: According to the requirements, the task scheduling contract sends requests to the selected computing and storage servers to let them execute the computing task cooperatively.

State Channel Establishment: The participated computing servers send a request to the state channel management contract separately to establish state channels.

Computing Sandbox Creation: The computing servers generate the computing sandbox and its encryption key pair collaboratively. The public key is uploaded to the blockchain.

Sandbox Location Obfuscation: The related data and initial model is stored by the storage servers and the Web3 service provider. To initiate the task, the data is signed with the private key of the data sender and encrypted with the public key of sandbox, before sent to the computing sandbox.

Data and Model Decryption and Verification: Computing servers decrypt the injected data and model with the private key of sandbox and verify their integrity with the corresponding public key.

Distributed Data Computing: The selected computing servers perform the computing task collaboratively, transmitting the data through state channel following the computing sandbox location obfuscation method.

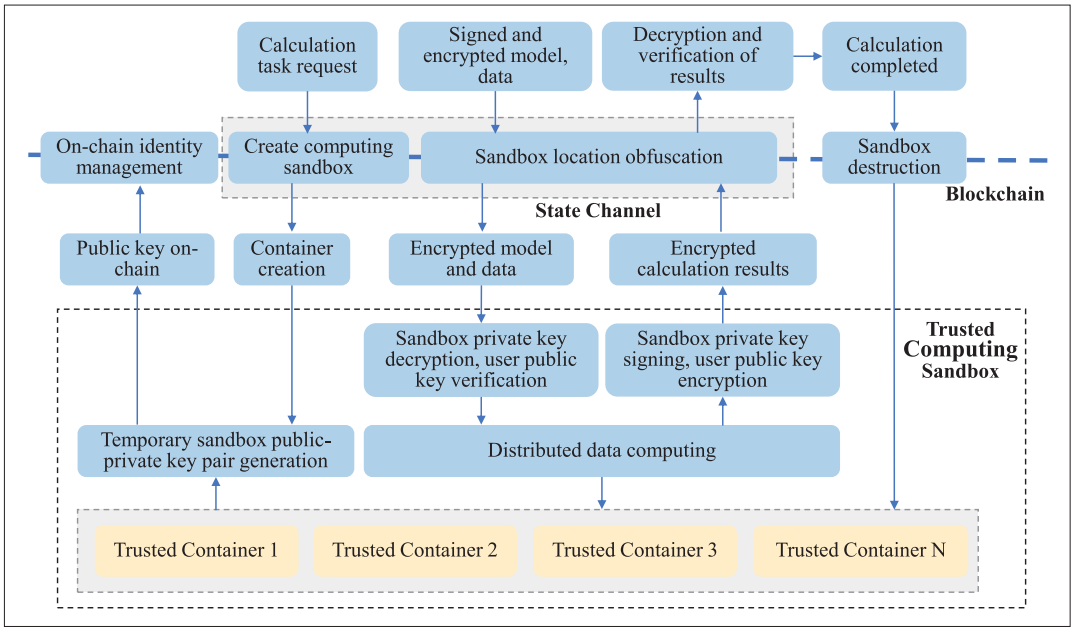

FIGURE 3. State channel-based trusted computing sandbox.

Model Signing and Encryption: Computing servers sign the computing result with the private key of sandbox and encrypt it with the public key of the Web3 service provider. After that, the result is sent to the Web3 service provider.

Computing Result Decryption and Verification: Web3 service provider decrypts the computing result with its own private key and verifies the integrity of the result with the public key of the computing sandbox.

Sandbox Destruction: After the computing is completed, the computing servers destroy the sandbox and the data in it.

State Channel Revocation: The computing servers submit a status update to the blockchain to revoke the state channel.

The specific life cycle of the trusted computing sandbox based on state channel mainly includes four steps, as shown in Fig. 2b.

Step1 — Lock On-Chain Status: According to the task scheduling result, some computing servers and storage servers are chosen to participate in the computing task. After receiving the task request, computing servers will send a request to the on-chain state channel management contract to lock their on-chain status.

Step2 — Establish State Channels: The state channels are established according to the data interaction requirements of the computing task. Taking federated learning as an example, there are usually a large number of communication needs between local nodes and the global node, while there are few communication needs between local nodes. Therefore, it is necessary to establish a state channel between the global node and each local node separately.

Step3 — Create Computing Sandbox: The computing sandbox is composed of the computing resources on each computing server participating in the task. Each server establishes a local independent computing space for the task. Based on container technology, the computing program of the task is deployed. The computing server has no read or write access to the data in this computing space.

Step4 — Destroy Computing Sandbox and State Channels, Update On-Chain Status: When the task is finished, computing servers will destroy the computing sandbox. After that, they will send a request to the blockchain to update their on-chain status, that is, the result of the computing task, and revoke the state channels.

Trusted Computing Sandbox and Sandbox Location Obfuscation Method

This section describes the detailed design of the state channel-based trusted computing sandbox structure and onion routing-based sandbox location obfuscation method.

State Channel-Based Trusted Computing Sandbox

To ensure the efficient and trusted off-chain computing, a trusted computing sandbox is established on the selected computing servers. The specific process of creating and scheduling the computing sandbox is shown in Fig. 3.

Since the trusted computing sandbox is carried out through containers, the scheduling of the sandbox can be regarded as the creation and deployment of the trusted container. After the state channels are established, the selected computing servers create a computing sandbox collaboratively by deploying a local container. A pair of asymmetric encryption keys is also generated, and the public key is uploaded to the blockchain. Before performing the computing task, computing servers use the private key of the computing sandbox to decrypt the data and initial model, and then use the public key of the Web3 user and Web3 service provider to verify the integrity and authenticity of them. When the task is completed, the container will be destroyed, together with the data in it. The corresponding encryption key pair stored on-chain also becomes invalid. The life cycle management of the computing sandbox prevents the leakage of data privacy and improves the utilization rate of resources.

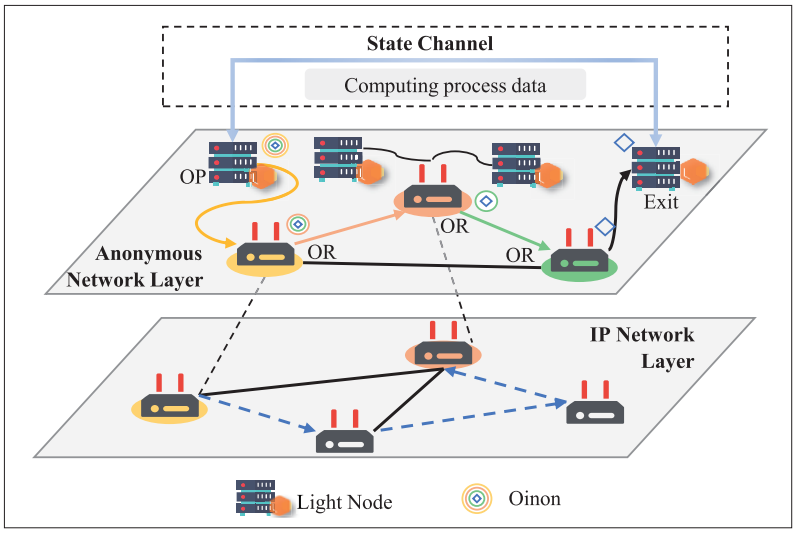

FIGURE 4. Onion routing-based sandbox location obfuscation method.

Sandbox Location Obfuscation Method Based on Onion Routing

The proposed architecture involves a lot of data transmission. To prevent data privacy leakage, we introduce the asymmetric encryption algorithm. However, an attacker can monitor the network traffic and analyze the routing pattern to obtain information of users and tasks. To solve the above problem, we introduce anonymous network technology to achieve sandbox location obfuscation.

Considering the resource limitation of computing servers, the sandbox location obfuscation method is based on onion routing. Onion routing is an an anonymous network technology which can anonymize the user identity and sandbox location with a relatively low communication delay and bandwidth overhead, as shown in Fig. 4.

In each data transmission process, there are three characters, which are onion proxy (OP), onion router (OR) and Exit. OP and Exit are the starting point and ending point of the data transmission circuit. OR is the router of the onion routing network, and is responsible for messages forwarding. Blockchain is used to act as the directory server in the onion routing network to store the information of onion routers, public key of devices, and so on. The actual route between two ORs is determined by the IP network.

To initiate the anonymous communication to Exit, OP needs to send a request to the blockchain in order to obtain the information of onion routers in the network, which includes the available OR list, the status and public key of each OR.

By choosing several ORs from the available OR list, OP can create an anonymous communication circuit to the Exit. The data needed to be transmitted is encrypted with the public key of each OR in the circuit by the inverse order. For example, we assume that there are three ORs in the circuit, naming OR1, OR2 and OR3. The data from OP passes through OR1, OR2 and OR3 in sequence, and is finally transferred to the EXIT. So, the data should be encrypted using the public keys of OR3, OR2, and OR1 in order. The multiple-layered encrypted data structure is called the onion.

In the process of data transmission, each OR peels off one layer of the onion, that is, decrypts the onion with its own private key. After decrypted by the last OR, the data is obtained and sent to the Exit.

With the proposed method, the network traffic caused by the data transmission between computing servers is obfuscated which makes it difficult for attackers to obtain the information of sandbox location and user identity by analyzing the network traffic.

Security Analysis and Experimental Verification

Security Analysis

According to the previous analysis, the blockchain based architecture can hardly support the Web3 applications. First, blockchain lacks computing attribute, making it difficult to guarantee the trustworthiness of the distributed computing process. Second, the frequent data transmission between computing servers can easily lead to data and user privacy leakage. To address the above issues, we design a series of mechanisms to improve the architecture.

For the trustworthiness of the distributed computing process, we design the trusted computing sandbox structure. Although the task is performed on the computing servers, they don't have the permission to access the computing space. The computing program is packed in the container, which can perform automatically based on the injected data and model. By establishing a state channel between the computing servers participating in the same task, the frequent data transmission is recorded off the chain and verified through mechanisms like the "challenge period".

In the aspect of data privacy, we introduce the asymmetric encryption algorithm. Data is encrypted with the public key of sandbox before it is transmitted. The private key of sandbox is kept by the selected computing servers, so that only the servers participating in the task can obtain the original data.

In the aspect of user privacy, we design the sandbox location obfuscation method. To avoid leaving traceable information during message forwarding, the traditional routing protocol is replaced by the anonymous network protocol. In this way, attackers can hardly obtain the user information or identify the location of the computing sandbox through the analysis of network traffic and routing pattern.

Experimental Verification

In order to test the performance of the proposed sandbox location obfuscation method, we design the experiment to compare the security, robustness and service execution latency of data transmission based on onion routing, Herbivore and traditional routing. We use a shadow simulator to generate an onion routing network. Beside the network, we design two different types of virtual nodes to simulate the possible attacks, which are normal nodes and malicious nodes. Malicious nodes can cause damage to the communication circuit. To reduce the experimental error, we take the average of ten experimental results.

As shown in Fig. 5a, the probability of communication circuit being damaged raises with the increase of malicious nodes in the network. The probability of data transmission based on onion routing being attacked is lower than that based

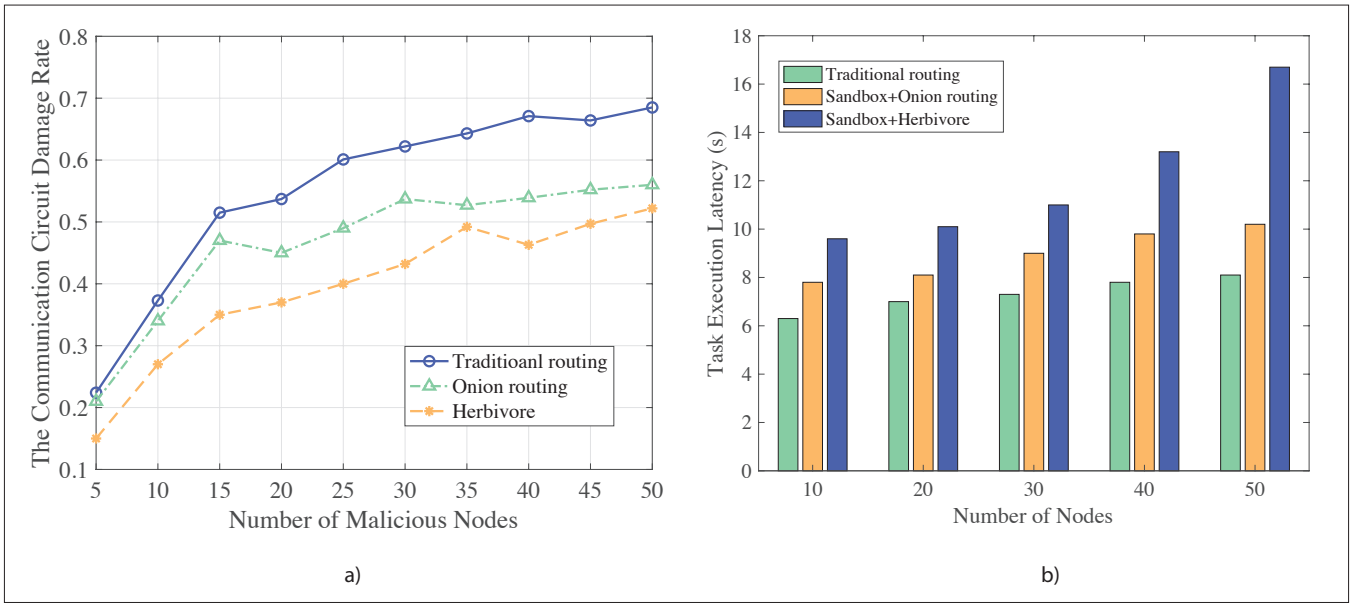

FIGURE 5. Performance verification of privacy protection method based on sandbox and onion routing technology: a) Damage rate; b) Task execution delay.

on traditional routing, but higher than that based on Herbivore. As shown in Fig. 5b, the task execution delay increases with the increase of computing nodes. In the proposed privacy protection method, the sandbox creation, sandbox destruction, and message decryption operations by ORs require additional time, so the task execution delay is larger than that in the traditional routing network. But the delay is smaller than the Herbivore-based privacy protection method. Since Herbivore is an anonymous network technology based on logical broadcasting, it introduces more communication delay in exchange for higher security, which is not suitable for data transmission in large-scale networks. In general, the delay and security are both considered in the privacy protection method based on the proposed sandbox and onion routing.

Conclusion

To ensure the computing trust and communication privacy of the Web3-oriented distributed computing, this article proposes a blockchain-assisted privacy-preserving data computing architecture. Under the architecture, a state channel-based trusted computing sandbox structure is designed to establish a trusted distributed computing environment in order to support the computing requirements of the Web3 application. To guarantee the user privacy and computing security, a sandbox location obfuscation method based on onion routing is designed making it difficult for attackers to locate computing sandboxes and infer user information. According to the analysis and experiment, the security of the computing architecture is illustrated. However, there are some remaining problems including the reduction in communication efficiency and the increase in communication overhead. To further improve the efficiency of this architecture, we intend to improve the computing sandbox location obfuscation method by introducing the machine learning algorithm to enable intelligent selection of ORs and furthermore balance the security with communication overhead.

Acknowledgments

This work was supported by the National Key R&D Program of China (2021YFF0901703).

References

[1] S. Filipcic, "Web3 & DAOS: An Overview of the Development and Possibilities for the Implementation in Research and Education," *Proc. 2022 45th Jubilee Int'l. Conf. Information, Communication and Electronic Technology*, 2022, pp. 1278–83.
[2] R. Gupta *et al.*, "Blockchain and Onion-Routing-Based Secure Message Exchange System for Edge-Enabled IIoT," *IEEE Trans. Industrial Informatics*, vol. 19, no. 2, 2023, pp. 1965–76.
[3] Z. Liu *et al.*, "Make web3.0 Connected," *IEEE Trans. Dependable and Secure Computing*, vol. 19, no. 5, 2022, pp. 2965–81.
[4] T. Sarathchandra and D. Jayawikrama, "A Decentralized Social Network Architecture," *Proc. 2021 Int'l. Research Conf. Smart Computing and Systems Engineering*, vol. 4, 2021, pp. 251–57.
[5] B. Jia *et al.*, "Blockchain-Enabled Federated Learning Data Protection Aggregation Scheme With Differential Privacy and Homomorphic Encryption in IIoT," *IEEE Trans. Industrial Informatics*, vol. 18, no. 6, 2022, pp. 4049–58.
[6] L. Yin *et al.*, "A Privacy-Preserving Federated Learning for Multiparty Data Sharing in Social IoTS," *IEEE Trans. Network Science and Engineering*, vol. 8, no. 3, 2021, pp. 2706–18.
[7] Z. Peng *et al.*, "Vfchain: Enabling Verifiable and Auditable Federated Learning via Blockchain Systems," *IEEE Trans. Network Science and Engineering*, vol. 9, no. 1, 2022, pp. 173–86.
[8] M. Cao, L. Zhang, and B. Cao, "Toward On-Device Federated Learning: A Direct Acyclic Graph-Based Blockchain Approach," *IEEE Trans. Neural Networks and Learning Systems*, 2021, pp. 1–15.
[9] X. Zheng and Z. Cai, "Privacy-Preserved Data Sharing Towards Multiple Parties in Industrial IoTS," *IEEE JSAC*, vol. 38, no. 5, 2020, pp. 968–79.
[10] K. Tjell and R. Wisniewski, "Private Aggregation With Application to Distributed Optimization," *IEEE Control Systems Letters*, vol. 5, no. 5, 2021, pp. 1591–96.
[11] H. Zhou *et al.*, "PFLF: Privacy-Preserving Federated Learning Framework for Edge Computing," *IEEE Trans. Information Forensics and Security*, vol. 17, 2022, pp. 1905–18.
[12] G. Yang *et al.*, "Socially Privacy-Preserving Data Collection for Crowdsensing," *IEEE Trans. Vehicular Technology*, vol. 69, no. 1, 2020, pp. 851–61.

Biographies

Shaoyong Guo is with the department of State Key Laboratory of Networking and Switching Technology, and received Ph.D degree at Beijing University of Posts and Telecommunication. His research interests include blockchain application technology, distributed intelligence, edge computing, and so on. He is

undertaking many key researches, development projects and fund projects, and contributed to a number of pioneering standards proposals in ITU-T. Email: syguo@bupt.edu.cn.

FAN ZHANG received her B.E. degree from Beijing University of Posts and Telecommunications. She is currently pursuing the Ph.D degree with Beijing University of Posts and Telecommunications, Beijing, China. Her research interests include blockchain, federated learning, distributed data computing and data sharing. Email: zf851258786@bupt.edu.cn.

SONG GUO received his Ph.D. in computer science from University of Ottawa. He is currently a full professor at Department of Computing, the Hong Kong Polytechnic University. Prior to joining PolyU, he was a full professor with the University of Aizu, Japan. His research interests are mainly in the areas of cloud and green computing, big data, wireless networks, and cyber-physical systems.

SIYA XU received Ph.D. degree in communication and information system from Beijing University of Posts and Telecommunications. She is currently a Lecturer with State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. Her research interests Communication Network Management, SDN/NFV, Smart Grid and Internet of Things. Email: xusiyaxsy@bupt.edu.cn.

FENG QI received the M.E. degree in computer application from the Northeastern University, Shenyang, China, in 1996. He is a Professor with Beijing University of Posts and Telecommunications, Beijing, China, and is also a Researcher with Peng Cheng Laboratory, Shenzhen, China. He has also written more than ten ITU-T International Standards and Industry Standards. He served as the Vice Chairman of the ITU-T Study Group 4 and Study Group 12. His research interests include communications software, network management, and business intelligence. Prof. Qi has won two National Science and Technology Progress Awards. Email: qifeng@bupt.edu.cn.