International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 4690-4701

https://doi.org/10.70517/ijhsa464312

Blockchain mechanisms and applications for ensuring data integrity and privacy in distributed networks

Yihui Deng[1] and Sanxiang Xiao[2,*]
[1] Experimental Training Center, Guangzhou College of Applied Science and Technology, Guangzhou, Guangdong, 511300, China
[2] School of Computing, Guangzhou College of Applied Science and Technology, Guangzhou, Guangdong, 511300, China
Corresponding authors: (e-mail: sanxiang_003@126.com).

Abstract This paper focuses on the dual application scenarios of blockchain in distributed networks and proposes a data integrity and security scheme based on blockchain technology. Decentralized authentication and data integrity verification based on smart contract and consensus mechanism solves the single point of failure and trust risk problem of traditional centralized architecture. Secure handling and dynamic auditing of private data is realized through verifiable computing protocol and homomorphic encryption. Two data integrity verification architectures are proposed to optimize the efficiency of data integrity verification and data security by combining the tamper-proof feature of blockchain and zero-knowledge proof. When the number of challenge data blocks is 1000, the communication overhead of this paper's scheme is only 13.08KB, and the computation overhead is 91.84%, 87.92%, and 53.81% lower than that of RDIC, SCLPV, and IBPA, respectively. The scheme in this paper has high processing efficiency in security analysis, and the maximum error rate of performing operations with different bits will not exceed 0.019%. Out of 2000 attacks, the number of successfully attacked is only 12, which is much better than the control method. The research results provide theoretical support and practical path for the landing of blockchain in high privacy demand scenarios such as Internet of Things and medical treatment.

Index Terms blockchain, smart contract, data integrity, verifiable computing protocol, privacy protection

I. Introduction

Due to the strong dependence of users on the communication network in special scenarios, it is particularly important to quickly reconfigure the network when it is damaged in order to restore the communication to ensure the normal execution of services and the security of data transmission [1], [2]. With the development of communication network from traditional wired data transmission to wireless data transmission, the current communication network restructuring problem in the case of network damage due to force majeure has been a huge problem for research [3], [4]. In distributed network scenarios, data usually need to be transmitted and stored between different nodes, the network suffers from targeted and random attacks, network damage leads to changes in network connectivity, the reconfiguration process is faced with the impact of non-trustworthy factors of reorganization of the network, the network stability is poor, and the normal application of the business is seriously jeopardized [5]-[7]. In a distributed network environment, the collection and use of data may involve a large amount of personal privacy information, however, these data may be misused by untrustworthy entities, leading to personal privacy leakage [8], [9]. And in some cases, individual nodes in a distributed network environment need to share data, but sharing sensitive data may lead to unpredictable privacy leakage [10]. Therefore, ensuring data integrity and privacy protection becomes a challenge.

Blockchain technology is a new distributed transmission network application technology with decentralization, P2P distributed network structure, consensus mechanism, and cryptographic applications, which enables self-management, verification, and delivery of nodes [11]-[13]. The P2P distributed network of blockchain provides a decentralized and reliable transmission network structure, which ensures the data synchronization and trusted security of blockchain network to a certain extent through the consensus protocol [14]. Based on the above characteristics, blockchain research has also been applied step by step to complex network fields, such as drone networks, nautical communication networks, and private network communication networks, and other scenarios characterized by point-to-point transmission, rapid structural changes, and the need for communication to be securely marked [15]-[17]. These studies provide a foundation for data integrity and privacy protection in distributed networks.

In this paper, we first deeply analyze the practice of blockchain in distributed network security, covering the technical path of authentication and data integrity verification. A data integrity verification scheme is designed based on blockchain technology, and a security enhancement scheme based on verifiable computing protocol is proposed. The performance advantages of this paper's scheme in data integrity are verified through experimental quantitative analysis. The analysis is carried out from three perspectives of efficiency, error rate, and attack resistance to evaluate the application effect of this paper's scheme in privacy protection.

II. Application of blockchain technology to distributed network security

The openness and dynamics of distributed networks pose serious challenges to data security, and traditional centralized architectures can hardly meet the urgent needs of data integrity and privacy protection due to the vulnerability of single point of failure and trust intermediaries. Blockchain technology provides a new paradigm for building a trustworthy distributed environment by virtue of decentralized ledger, consensus mechanism and cryptography tools.

II. A. Application Scenarios of Blockchain Technology in Distributed Network Security

II. A. 1) Distributed authentication

Authentication is a fundamental part of network security, and traditional centralized authentication servers face the problems of single-point failure and trust risk. The distributed ledger and consensus mechanism of blockchain provides a solution idea for decentralized authentication system. Specifically, a pair of public and private keys can be generated for each user or device on the blockchain, and the hash value of their identity information can be recorded. When a user or device initiates an access request, any node in the blockchain network can verify whether its public key signature matches the record on the chain. Taking IoT as an example, a large number of low-power devices need to verify each other's identities in a centerless server scenario, while blockchain can store the registration information of these devices in each node in a decentralized manner and execute the authentication logic automatically through smart contracts. In this way, even if a node is attacked or taken offline, other nodes can still provide complete and reliable authentication data.

II. A. 2) Data integrity verification

Information is transmitted and stored in distributed environments and is often at risk of tampering, loss or duplicate writes. Traditional practices often rely on centralized servers to periodically verify or back up data, however, when the server itself is attacked or internal data is maliciously tampered with, subsequent verification loses its meaning. Blockchain technology provides a more trustworthy solution for data integrity verification through a tamper-proof ledger structure. For example, in a cross-hospital data sharing scenario in the healthcare industry, if a patient's medical records need to be circulated among multiple hospitals, healthcare insurance providers and testing centers, each update will generate new examination reports or diagnostic information. After writing these key information in the form of encrypted hash into the blockchain, if there is a data modification in any link, the hash comparison in the ledger will immediately show the mismatch. As another example, in supply chain management, the entry of product information by different nodes, if tampered with, will seriously affect the subsequent traceability process and quality control. Utilizing blockchain to record key data ensures that the data submitted by each participant is “written once and cannot be changed”, thus enhancing the credibility of the information. If a node attempts to falsify or delete data, the non-tampering nature of the ledger will make this operation easily detectable.

II. B. Data integrity validation

II. B. 1) Blockchain-based single TPA, CSP validation methods

Whether before or after the introduction of blockchain technology, data integrity verification methods based on single TPA and CSP architectures are the most common, and one of the more typical architectures is shown in Figure 1. There are three roles in this architecture: data owner (DC), TPA, and CSP, and the architecture is divided into the following steps during verification:

(1) Initialization phase

In this stage some system parameters are initialized. First, two random large prime numbers p , q are chosen and $p-1$ is divisible by q . Second, n values from Z_p^* are randomly chosen to form the vector $G=[g_1,g_2,\cdots,g_n]$.

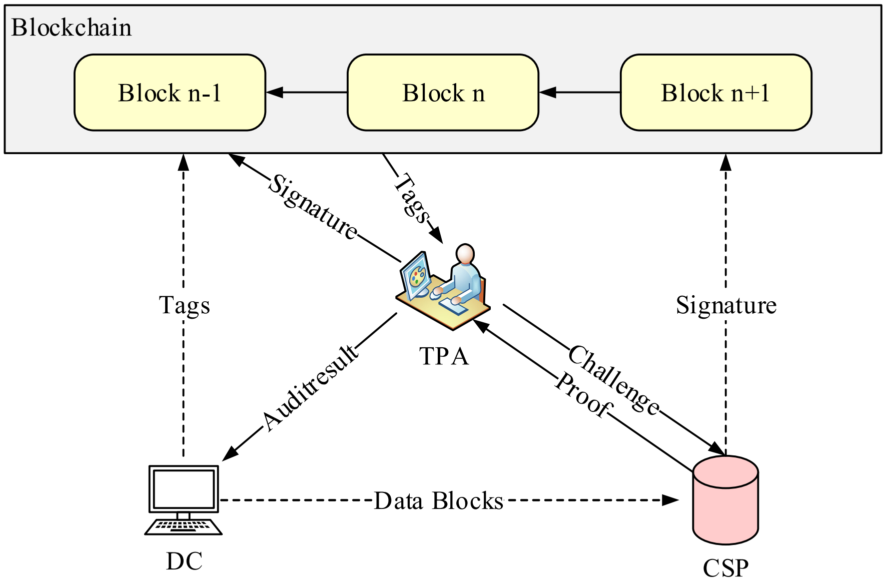

Figure 1: Single TPA,CSP architecture

(2) Storage Phase

In this phase, DC will locally divide the file M into α blocks, i.e., $M=(m_1,m_2,\cdots,m_i,\cdots,m_\alpha)$. Also divide each block into n zones, i.e., $m_i=(m_{i1},m_{i2},\cdots,m_{in})$. And each data block m_i is encrypted using equation (1). Each encrypted data block is called Tag. Where m_{il} is the data in the l th zone of the i th block of the file, g_l is the l th value in G, and Tag_i is the Tag value of the i th block.

$$Tag_{g_i}=\prod_{l=1}^{n} g_l^{m_{il}} \bmod p\left(1\le i\le \alpha\right) \tag{1}$$

After the calculation, DC uploads the Tag value into the blockchain, and at the same time, DC sends the original data to CSP for storage.

(3) Validation phase

In this phase, a validation request is sent by the TPA, called Challenge, denoted as $chal$. The $chal$ contains the data block number to be validated, i.e., $chal=\left(s_1,s_2,\ldots,s_x\right)$. The CSP, upon receiving the $chal$, will take out the corresponding data block $\left(m_{s_1},m_{s_2},\ldots,m_{s_x}\right)$, and compute the data integrity $proof$ using equation (2) and send it to TPA. where $g_t\in G$, m_{it} is the data in the t th zone of the i th block of the file.

$$proof=\prod_{t=1}^{n} g_t^{\sum_{i=s_1}^{s_x} m_{it}} \bmod p \tag{2}$$

When TPA receives $proof$, it takes out the Tag corresponding to the corresponding data block from the blockchain and calculates it with formula (3), if the calculated value is the same as $proof$, then the data is complete, then it will upload the verification result to the blockchain for storage, and it will also inform DC that the data is complete. Where x is the block number.

$$H=\prod_{i=1}^{x} Tag_i \bmod p \tag{3}$$

According to the architecture shown in Fig. 1, joining the blockchain takes on the work of data storage, which ensures the openness, authority and non-tampering of the data. Each node in the blockchain can verify the validation results of TPA, but since it is a centralized TPA and CSP, the two will have a single point of failure.

II. B. 2) Blockchain-Based Multi-TPA Single CSP Verification Approach

In a data integrity verification approach based on a multi-TPA single CSP architecture, the more typical architecture is shown in Figure 2.

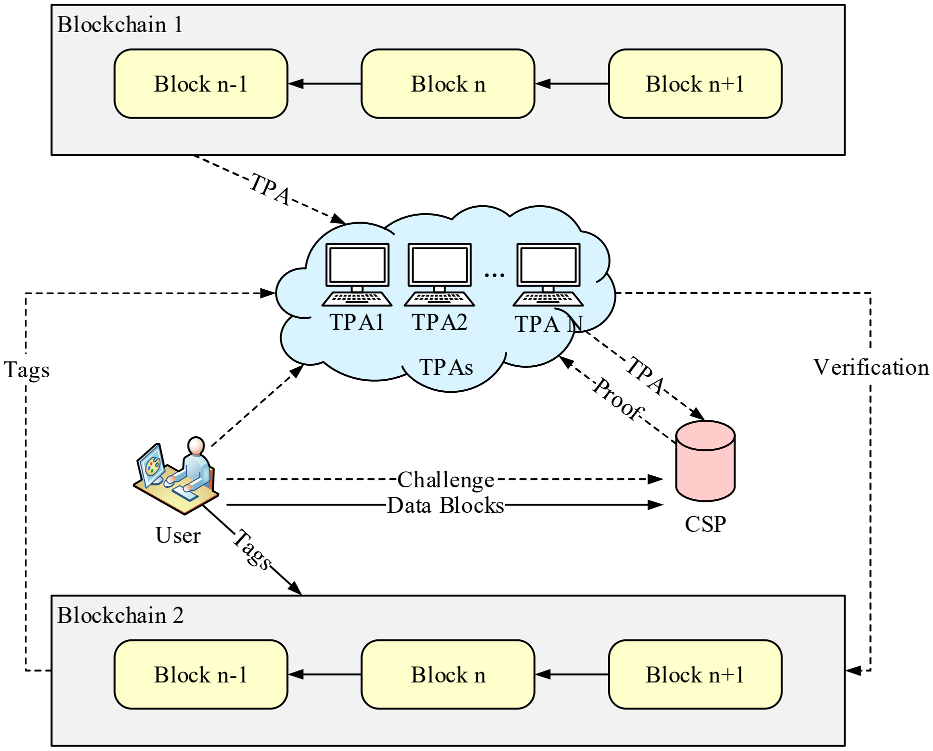

Figure 2: Multi-TPA single CSP architecture

In this architecture there are two chains, one chain stores the service metrics information of all TPAs and the other chain stores the data encryption and validation information. The specific verification steps are as follows:

(1) Initialization phase

This phase denotes G_1, G_2 and G_T as three multiplicative cyclic groups, where the orders of G_1 and G_2 are prime p. Let g be the generating element of G_2. The bilinear map $e: G_1 \times G_2 \to G_T$. Let $H: \{0,1\}^* \to G_1$ be the hash function that maps string data to points in G_1; and let $h: G_1 \to Z_q^*$ be another hash function that maps a point in G_1 to a point in Z_q^*. The user generates a random key $x(x \in Z_q^*)$ and computes the public key $y = g^x$.

(2) Storage phase

The user divides his file F into n blocks, i.e., $F = \left(F_1, F_2, \cdots, F_i, \cdots, F_n\right)$, and generates a random value $u(u \in G_1)$, and then uses equation (4) to calculate the label $tag\phi_i$ for each block. where $H(F_i)$ is the value of the hash of the i block.

$$tag\phi_i = \left(H(F_i)\cdot u^{(F_i)}\right)^x \tag{4}$$

After the computation, the user will upload the set $\Phi = \{\phi_i \mid i \in n\}$ consisting of $tag\phi_i$ and the randomized values into Blockchain2, and at the same time, the user will send the original data to CSP for storage.

(3) Verification phase

In this phase, firstly, it is necessary to obtain the service metrics information of all TPAs from Blockchain1 and select the appropriate TPAs for the user according to the user's needs.Secondly, the user selects a group of TPAs with id-set $LTid$ to perform the label-based subtask LT. Meanwhile, the user selects another set of TPAs with id set $DTid$ to perform the data-based subtask DT. Subsequently, the user .generates two seeds sd_{bt} and sd_{ra}, where the seed sd_{bt} is used to generate the block number of the data to be verified and the seed sd_{ra} is used to generate its corresponding random number v_i. Finally, $\{LTid, DTid, sd_{bt}, sd_{ra}\}$ is sent to the CSP.

After receiving the data, the CSP first uses sd_{bt} to generate the validated data block number and form the set I; it uses sd_{ra} to generate a series of random numbers and form the set $chal = \{v_i \mid i \in I\}$. After that, the proof is computed using Eqs. (5) and (6) and uploaded into Blockchain2 for storage. where $\phi_i \in \Phi$ and $F_i \in F$.

$$LT = \prod_{i \in I} \phi_i^{v_i} \tag{5}$$

$$DT = \sum\nolimits_{i \in I} F_i v_i \tag{6}$$

Perform the label-based subtask of TPA calculation using equation (7) and perform the data-based subtask of TPA calculation using equation (8) and upload the results of both calculations into Blockchain2. If these two values are equal then the data is complete. Where, $v_i \in chal$.

$$VoLT = e(LT, g) \tag{7}$$

$$VoDT = e\left(\prod\nolimits_{i \in I} H(F_i)^{v_i} \cdot u^{DT}, y\right) \tag{8}$$

In the architecture shown in Fig. 2, although the problem of single-point failure of TPA can be avoided to a certain extent, and the TPA information and data encryption and authentication information are stored in two chains separately, which reduces the storage pressure on the chain, but the CSP will have a single-point failure potential and slow authentication speed.

II. C.Security analysis

Since the concept of smart contracts was introduced by blockchain technology, blockchain-based smart contract systems have gained practical applications in many fields due to their feature of ensuring correct code execution without the need for a trusted third party. However, blockchain public contract invocation parameters and codes can lead to privacy leakage and limit its deployment in sensitive data applications. In order to land smart contracts in privacy-demanding scenarios and promote the integration of blockchain technology with the real economy, there is an urgent need to introduce data privacy protection mechanisms for smart contracts. Existing protection schemes are increasing trust assumptions, zero-knowledge proof, and full homomorphic encryption, but each has its own limitations. Therefore this paper proposes a publicly verifiable verifiable computing protocol. The security of verifiable computation should include correctness, unforgeability, efficiency, and for publicly verifiable verifiable computation, it also needs to satisfy the publicizability.

(1) Correctness analysis:

First, the correctness of the computation process of the contract executor is inherited from the correctness of the MHE, and the correctness of the decryption process is inherited from the correctness of the FHE decryption process. If all participants run the protocol in a semi-honest way, then the verification process satisfies the correctness definition for checks against A, B, C, and D without loss of generality, as exemplified by the check against A, as shown in expression (9). And for L_{span} and P, as shown in expressions (10) and (11)

$$\hat{V}_{mid} = r_v \alpha_v (v_{mid}(s) + \delta_v t(s)) = \alpha_v r_v (v_{mid}(s) + \delta_v t(s)) = \alpha_v V_{mid} \tag{9}$$

$$\begin{aligned} L &= \beta(r_v v_{mid}(s) + r_w w_{mid}(s) + r_y y_{mid}(s) \\ &\quad + (r_v \delta_v + r_w \delta_w + r_y \delta_y) \cdot t(s)) \\ &= \beta(r_v v_{mid}(s) + r_v \delta_v t(s) + r_w w_{mid}(s) \\ &\quad + r_w \delta_w t(s) + r_y y_{mid}(s) + r_y \delta_y t(s)) \\ &= \beta(r_v V_{mid} + r_w W_{mid} + r_y Y_{mid}) = \beta L_{span} \end{aligned} \tag{10}$$

$$\begin{aligned}
P &= (v_{io}(s)+\gamma_v t(s)+v_{mid}(s)+\delta_v t(s))\\
&\quad\cdot(w_{io}(s)+\gamma_w t(s)+w_{mid}(s)+\delta_w t(s))\\
&\quad-(y_{io}(s)+\gamma_y t(s)+y_{mid}(s)+\delta_y t(s))\\
&= (v(s)+\gamma_v t(s)+\delta_v t(s))\cdot(w(s)+\gamma_w t(s)\\
&\quad+\delta_w t(s))-(y(s)+\gamma_y t(s)+\delta_y t(s))\\
&= t(s)\left(\frac{v(s)\cdot w(s)}{t(s)}+(\delta_v+\gamma_v)(\delta_w+\gamma_w)t(s)\right.\\
&\quad\left.+(\delta_v+\gamma_v)w(s)+(\delta_w+\gamma_w)v(s)\right)\\
&\quad-t(s)\left(\frac{y(s)}{t(s)}-(\delta_y+\gamma_y)\right)\\
&= t(s)\left(\frac{v(s)w(s)-y(s)}{t(s)}+\delta_v w(s)\right.\\
&\quad\left.+\gamma_v w(s)+\delta_w v(s)+\gamma_w v(s)-\delta_y-\gamma_y\right)\\
&\quad+t(s)(\gamma_v\gamma_w t(s)+\gamma_v\delta_w t(s)+\gamma_w\delta_v t(s)+\delta_v\delta_w t(s))\\
&= h'(s)\cdot t(s)=H\cdot t(s)
\end{aligned} \tag{11}$$

Therefore, it can be obtained that the scheme of this paper satisfies the definition of correctness.

(2) Unfalsifiability analysis:

For a polynomial-time adversary A, the scheme of this paper is satisfying the definition of unforgeability under the PKE and PDH assumptions. First the coefficients of the terms in $V_{mid}, W_{mid}, Y_{mid}, H$, which are the intermediate variables of the circuit C, are fixed by $\hat{V}_{mid}, \hat{W}_{mid}, \hat{Y}_{mid}, \hat{H}$, respectively, so if A can find the coefficients about the terms of $v_{mid}(x), w_{mid}(x), y_{mid}(x)$ in a way that satisfies the correspondence between $V_{mid}, W_{mid}, Y_{mid}, H$ and $\hat{V}_{mid}, \hat{W}_{mid}, \hat{Y}_{mid}, \hat{H}$ without executing the circuit to obtain the intermediate values, then it represents that the adversary A is capable of solving the PKE hard problem.

For A to find a polynomial relation such that $V(x)\cdot W(x)-Y(x)\neq H(x)\cdot t(x)$ exists, but $V(s)\cdot W(s)-Y(s)=H(s)\cdot t(s)$, if this is true, it means that A is able to construct a non-zero polynomial $r(x)=V(x)\cdot W(x)-Y(x)-H(x)\cdot t(x)$, and s happens to be a root of $r(x)$, if A can find this polynomial, it means that the adversary A can solve the PDH problem.

In summary, the proposed scheme meets the definition of unforgeability under the assumptions of PKE and PDH.

(3) Exposability analysis:

For a fixed X and τ, the explicit values of the intermediate variables of the computational circuit C are encoded onto $v_{mid}, w_{mid}, y_{mid}$, which are generated in the verifier's perspective as $V_{mid}, W_{mid}, Y_{mid}$, and for the purposes of the bind $V_{mid}, W_{mid}, Y_{mid}$ the protocol requires the contract executor to generate $\hat{V}_{mid}, \hat{W}_{mid}, \hat{Y}_{mid}$. The contract executor encodes the intermediate variables into the ciphertext space corresponding to pk_1 as part of the credential π_1. The credential π_1 will be switched to the ciphertext space corresponding to pk_2 to generate the credential π_2 by the public key federation switching by the data owner, while the data owner and the data demander will provide V_{io}, W_{io}, Y_{io} to get the final credential $\pi=(\pi_2, V_{io}, W_{io}, Y_{io})$. To ensure that the contract executor has not tampered with the inputs and outputs of the computational circuitry, the verifier computes $P=(V_{io}+V_{mid})\cdot(W_{io}+W_{mid})-(Y_{io}+Y_{mid})$, which is bounded by $H=P/t(s)$ and $\hat{H}$ generated by the contract executor. To ensure the disclosability of the scheme in this paper, $V_{mid}, W_{mid}, Y_{mid}$ hide the real $v_{mid}, w_{mid}, y_{mid}$ by adding uniformly randomly sampled values $\delta_v t(s), \delta_w t(s), \delta_y t(s)$, and ditto for V_{io}, W_{io}, Y_{io} rely on adding random values $\gamma_v t(s), \gamma_w t(s), \gamma_y t(s)$ to hide the real v_{io}, w_{io}, y_{io}.

For the sake of the openness of the scheme protocol of this paper, i.e., to ensure that by $(\tau, \pi, \mathrm{vk}, \mathrm{X}, \mathrm{Y})$, the adversary cannot infer the explicit inputs and outputs of the circuit C, the computational algorithms according to the scheme protocol of this paper define the simulators Sim_1 and Sim_2. Assuming that the simulator Sim_1 is capable of generating simulation-assisted authentication messages τ' and generating trapdoors $\sigma = \left(sk_2, s, \alpha, \alpha_v, \alpha_w, \alpha_y, r_t, r_w, r_y\right)$, with τ', σ as output.

The simulator Sim_2 takes as input the simulation auxiliary information τ', the trapdoor σ, and $v_{io}(x), w_{io}(x), y_{io}(x)$, and generates the simulated credentials π'. The Sim_2 randomly selects $v(x), w(x), y(x)$ and computes $h(x)$ by $t(x)$ with the computation rule shown in expression (12).

$$h(x) = \frac{v(x) \cdot w(x) - y(x)}{t(x)} \tag{12}$$

Sim_2 computes $v_{mid}, w_{mid}, y_{mid}$ by $v(x), w(x), y(x)$ with the computation rule shown in expression (13).

$$\begin{cases} v_{mid}(x) = v(x) - v_{io}(x) \\ w_{mid}(x) = w(x) - w_{io}(x) \\ y_{mid}(x) = y(x) - y_{io}(x) \end{cases} \tag{13}$$

Then Sim_2 can be encoded by the trapdoor σ, which utilizes s in σ to compute $v_{mid}, w_{mid}, y_{mid}, h(s)$, and then utilizes α in σ to generate $\alpha v_{mid}(s), \alpha w_{mid}(s), \alpha y_{mid}(s)$, and generate simulation credentials π' according to the computational algorithm.

Since π' generated through the simulator satisfies the validation equation and the statistical distribution of the credentials π' as well as the decrypted plaintexts are identical to the statistical distribution of π and its plaintexts generated by the real data owner, the contract executor, the credentials π' generated through the simulator are indistinguishable from the credentials π generated through the real data owner, the contract executor, to the verifier are indistinguishable. As a result, the contract verifier cannot learn anything about the plaintext inputs and outputs of the computational circuit C and the intermediate states by interacting with the protocol of the scheme herein, and thus the disclosability of this protocol is established.

(4) Efficiency Theory Analysis:

In the ProbGen phase, one of the steps with the highest demand for computational resources is the interaction between data owners via secure multi-party computation to generate random polynomials for each line of the circuit C, which is only related to the number of lines in the circuit C, and thus can be generated first prior to the execution of the contract, or by specifying a A third party generates it first, and then the owner extracts it during recomputation, so it can be excluded from the computation time consumption. Moreover, the computation required for the generation process of τ is of order $O(N)$, where N is the complexity of the circuit C. In the Verify phase, the computation required for the verification by the contract verifier is of order $O(1)$. Therefore, analyzed theoretically, the scheme in this paper is an efficient verifiable computing protocol.

III. Blockchain-based data integrity and security analysis

In this experiment, the linear pairing algorithm provided in the pairing-based cryptography library as well as the GNU multi-precision arithmetic library is used for code implementation in C+. The experimental environment is set up on a virtual machine. To optimize the cryptographic efficiency and security, a base field size of 512 bits is configured and Barreto-Naehrig (BN) curve is used.

III. A. Data integrity

III. A. 1) Performance

A comparative analysis between this scheme and other data integrity auditing schemes is conducted, and the results of the communication overhead comparison are shown in Fig. 3. When the number of challenge data blocks is 1000, the communication overhead of this scheme is only 13.08KB, which is significantly better than the RDIC (16.06KB) and SCLPV (14.97KB) schemes. Although this scheme is comparable to the IBPA scheme in terms of communication cost, this scheme achieves conditional identity privacy, supports the key generation center to effectively trace the real identity in the malicious, and successfully defends against malicious third-party auditors by exploiting the decentralized nature of the blockchain. Therefore, this scheme achieves a reasonable balance of communication overhead in the auditing process.

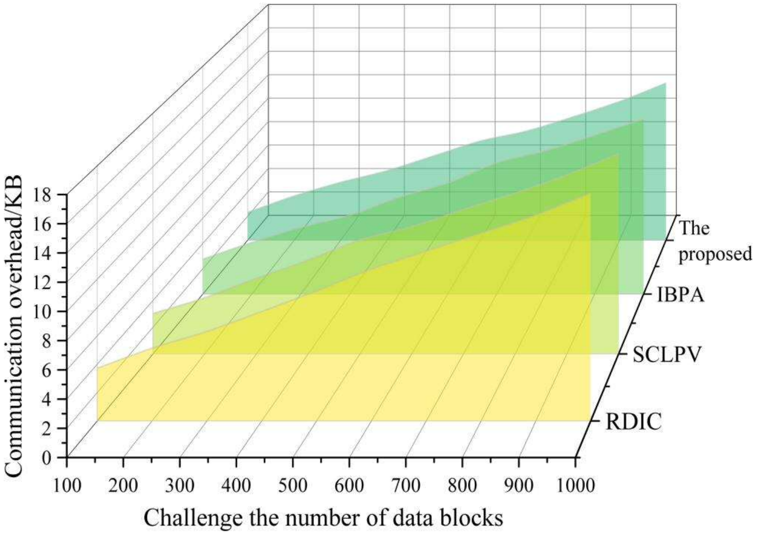

Figure 3: Comparison results of communication overhead

In addition, this scheme significantly reduces the computational complexity by simplifying the λ's to the constant level by adopting the homomorphic hash function to design the signature, making the computational overhead of this part almost negligible. A comparison of the experimental results of the computational overhead in the integrity audit phase is shown in Fig. 4. With the growth of the number of challenge data blocks, the computational overheads of this scheme are all lower than those of the comparison schemes, and when the number of challenge data blocks is 1000, the computational overheads of this scheme are 91.84%, 87.92%, and 53.81% lower than those of RDIC, SCLPV, and IBPA, respectively. Therefore, this scheme achieves higher efficiency in the integrity audit phase in terms of computational cost.

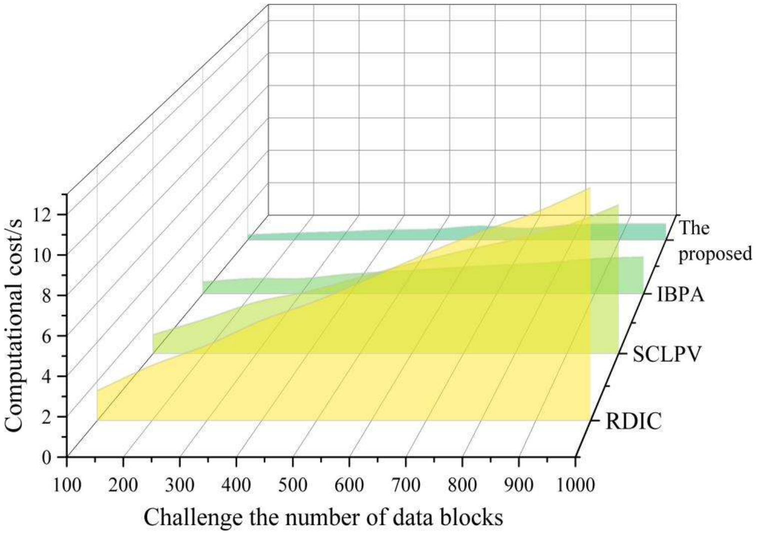

Figure 4: Computational overhead comparison results

The test was analyzed by collecting and calculating the "gas" costs required for different types of transactions on the Ethernet platform. A comparison of the costs of different types of transactions is shown in Figure 5, where the first deployment of a smart contract consumes the highest amount of "gas", which is significantly more than that consumed in the subsequent invocation phases. Although the initial deployment cost is higher, the "gas" cost of subsequent operations, such as challenge contract invocation, is relatively low, consuming 9,728.51gas, and showing a stable trend. This indicates that despite the significant initial deployment cost, the overall transaction cost is predictable and can be maintained in a reasonable range. This facilitates frequent interactions and maintenance at a later stage at a relatively low cost.

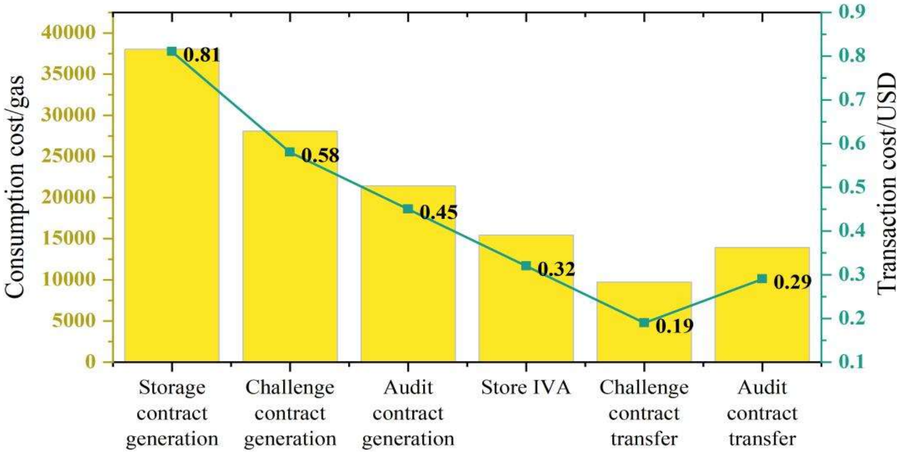

Figure 5: Comparison of the results of different types of transaction costs

III. A. 2) Processing efficiency

The experiments in this paper were conducted for 50 cycles of consensus, and the comparison of the TPS of transaction processing per second of each scheme is shown in Fig. 6.The average value of the TPS of IBPA is 68.07, while the TPS in the scheme of this paper fluctuates around 80, with an average of 84.04, which is the best performance among the four schemes.

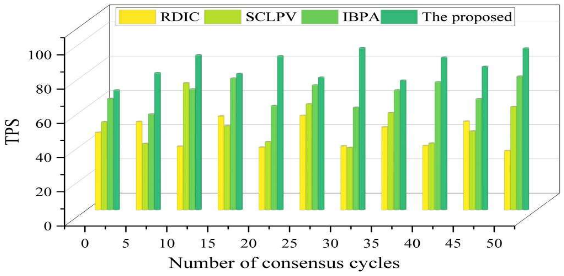

Figure 6: TPS comparison of transaction processing per second

The experiment also conducted 50 rounds of consensus, and the comparison of the system block-out time of each scheme is shown in Fig. 7. The block-out time of RDIC fluctuates around 23s, a block is generated in the system of SCLPV in an average of around 15s, and the average system block-out time of IBPA is 6.71s. Whereas in this paper's scheme, the system generates a block in 2.06s on average. Therefore, the block out speed of the blockchain in this paper's scheme is higher than that of other schemes, and combined with the transaction processing per second of our scheme, we can find that the system of the integrity audit scheme is greatly improved in the blockchain throughput.

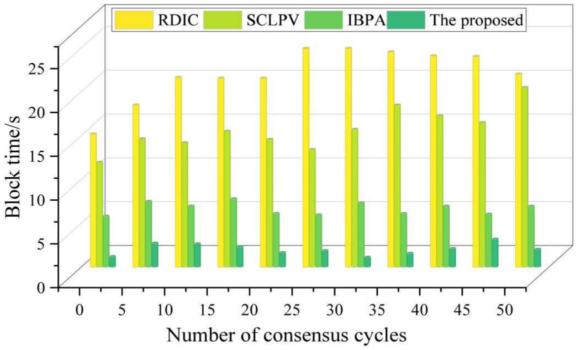

Figure 7: System block time comparison of each scheme

III. B. Security analysis

III. B. 1) Efficiency

The group sizes selected for testing are 128bit, 512bit, 2048bit and 8192bit, and the time overheads of encryption, decryption and homomorphic operations involved in the scheme are tested. To avoid chance results, 500 test experiments are conducted to calculate the average value, and the average time overhead results obtained are shown in Table 1. where Enc(m) denotes an encryption operation, Dnc(Enc(m)) denotes a decryption operation, and Enc(m_1)Enc($m_{2)}$ denotes a homomorphic addition operation. The average time overhead under 128bit, 512bit, 2048bit, and 8192bit is 0.828ms, 1.308ms, 6.514ms, and 44.409ms, respectively, which is more efficient in processing.

Table 1: Average time cost results(ms)

	128bit	512bit	2048bit	8192bit
Enc(m)	0.735	1.018	4.973	30.286
Dnc(Enc(m))	0.092	0.287	1.536	14.117
Enc(m_1)Enc(m_2)	0.001	0.003	0.005	0.006
Overall	0.828	1.308	6.514	44.409

III. B. 2) Error rate

In order to evaluate the plaintext and ciphertext computation accuracy of the encryption algorithm used in the scheme of this paper, the scheme of this paper is applied to perform the ciphertext and plaintext computation of the data under different bit numbers, and the encryption performance is judged based on the error rate of the data before and after the computation. The variable error rates of encryption operations under different bit numbers are shown in Fig. 8. Whether the cloud storage data is 1-bit, 2-bit, or 3-bit, the error rate is basically flat when performing data encryption and decryption operations. Meanwhile, the higher the number of cloud storage data bits, the smaller the error rate of encryption and decryption operations, and the smaller the fluctuation state, which is mainly due to the fact that the value of 1-bit is too small, so there is a change in the error of cloud storage data encryption and decryption calculations. However, when applying the method of this paper to carry out the encryption and decryption operation of cloud storage data with different digits, the error between the encryption and decryption calculation results and the actual data is extremely small, and the error rate has been kept within a low level, and the maximum will not exceed 0.019%. This can show that the experimental group method has good encryption performance.

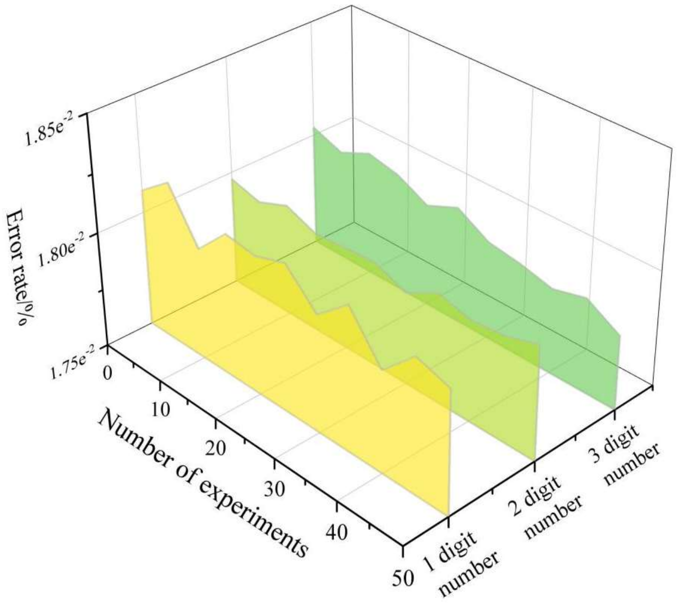

Figure 8: Variable error rate of encryption operation under different bits

III. B. 3) Resistance to attack

In order to further verify the attack resistance of cloud storage data under the methods of this paper, after applying the methods of this paper and the two data security methods, MHE and FHE, respectively, for multi-party privacy protection of cloud storage data, different numbers of impersonation attacks, i.e., the attacker tries to disguise himself as a legitimate user in order to gain access to the cloud storage data, are performed on the cloud storage data. Under each group of attacks, the number of successful attacks under the three methods is recorded, that is, the attacker succeeds in gaining access to the cloud storage data or performs unauthorized operations. The results of attack resistance under different methods are shown in Fig. 9. Under multiple impersonation attacks, after this paper's method carries out multi-party privacy protection of cloud storage data, the number of successful attacks is only 12 out of 2000 attacks, which is much better than the control method. It shows that the cloud storage data under this paper's method has the best attack resistance, which further verifies that this paper's method is significantly better than the control two methods, and can better protect the security of cloud storage data.

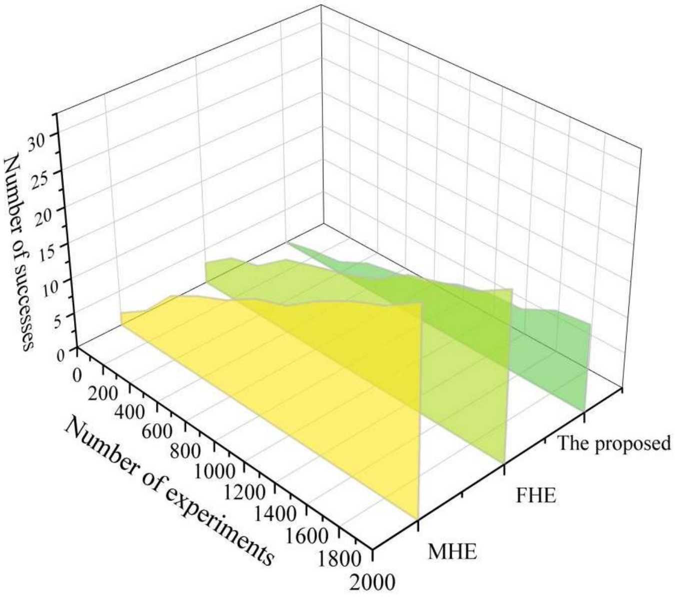

Figure 9: Anti-aggression results under different methods

IV. Conclusion

In this paper, we systematically explore the innovative application of blockchain technology in the field of distributed network data security and propose a blockchain-based data integrity and security analysis scheme.

When the number of challenge data blocks is 1000, the communication overhead of this paper's scheme is only 13.08KB, which is significantly better than that of RDIC (16.06KB) and SCLPV (14.97KB) schemes, and the computation overhead is 91.84%, 87.92%, and 53.81% lower than that of RDIC, SCLPV, and IBPA, respectively.The TPS of IBPA in the 50-cycle consensus The average value is 68.07, while the TPS in this paper's scheme fluctuates around 80 with an average value of 84.04, which is the best performance among the four schemes. At the same time, the system generates a block in 2.06s on average, which is greatly improved in blockchain throughput.

In the security analysis, the average time overhead of this paper's scheme is 0.828ms, 1.308ms, 6.514ms, and 44.409ms for 128bit, 512bit, 2048bit, and 8192bit, respectively, which is more efficient in processing. The error rate of performing operations with different bit numbers has been kept within a low level, and the maximum does not exceed 0.019%. Out of 2000 attacks, the number of times being successfully attacked is only 12, which is much better than the control method.

References

[1] Chen, J., Zhao, F., & Xing, H. (2020). Research on Security of Mobile Communication Information Transmission Based on Heterogeneous Network. Int. J. Netw. Secur., 22(1), 145-149.

[2] Sova, O. (2022). Development of methodological principles of routing in networks of special communication in the conditions of fire damage and radio electronic flow. Technology audit and production reserves, 3(2/65), 24-28.

[3] Gandotra, P., & Jha, R. K. (2017). A survey on green communication and security challenges in 5G wireless communication networks. Journal of Network and Computer Applications, 96, 39-61.

[4] Nazir, R., Laghari, A. A., Kumar, K., David, S., & Ali, M. (2021). Survey on wireless network security. Archives of Computational Methods in Engineering, 1-20.

[5] Liang, W., Huang, Y., Xu, J., & Xie, S. (2017). A distributed data secure transmission scheme in wireless sensor network. International Journal of Distributed Sensor Networks, 13(4), 1550147717705552.

[6] Khooi, X. Z., Csikor, L., Divakaran, D. M., & Kang, M. S. (2020, June). DIDA: Distributed in-network defense architecture against amplified reflection DDoS attacks. In 2020 6th IEEE Conference on Network Softwarization (NetSoft) (pp. 277-281). IEEE.

[7] Ge, X., Yang, F., & Han, Q. L. (2017). Distributed networked control systems: A brief overview. Information Sciences, 380, 117-131.

[8] He, J., Cai, L., Cheng, P., Pan, J., & Shi, L. (2018). Distributed privacy-preserving data aggregation against dishonest nodes in network systems. IEEE Internet of Things Journal, 6(2), 1462-1470.

[9] Du, J., Jiang, C., Gelenbe, E., Xu, L., Li, J., & Ren, Y. (2018). Distributed data privacy preservation in IoT applications. IEEE Wireless Communications, 25(6), 68-76.

[10] Wang, L., Zhong, H., Cui, J., Zhang, J., Wei, L., Bolodurina, I., & He, D. (2024). Privacy-Preserving and Secure Distributed Data Sharing Scheme for VANETs. IEEE Transactions on Mobile Computing.

[11] Zhao, X., Lei, Z., Zhang, G., Zhang, Y., & Xing, C. (2020). Blockchain and distributed system. In Web Information Systems and Applications: 17th International Conference, WISA 2020, Guangzhou, China, September 23–25, 2020, Proceedings 17 (pp. 629-641). Springer International Publishing.

[12] Saidi, H., Labraoui, N., Ari, A. A. A., Maglaras, L. A., & Emati, J. H. M. (2022). DSMAC: Privacy-aware Decentralized Self-Management of data Access Control based on blockchain for health data. IEEE Access, 10, 101011-101028.

[13] Christodoulou, K., Chatzichristofis, S. A., Sirakoulis, G. C., & Christodoulou, P. (2019). RandomBlocks: A Transparent, Verifiable Blockchain-based System for Random Numbers. J. Cell. Autom., 14(5-6), 335-349.

[14] Javed, M. U., Rehman, M., Javaid, N., Aldegheishem, A., Alrajeh, N., & Tahir, M. (2020). Blockchain-based secure data storage for distributed vehicular networks. Applied Sciences, 10(6), 2011.

[15] Alkadi, R., Alnuaimi, N., Yeun, C. Y., & Shoufan, A. (2022). Blockchain interoperability in unmanned aerial vehicles networks: State-of-the-art and open issues. Ieee Access, 10, 14463-14479.

[16] Yang, T., Cui, Z., Alshehri, A. H., Wang, M., Gao, K., & Yu, K. (2022). Distributed maritime transport communication system with reliability and safety based on blockchain and edge computing. IEEE Transactions on Intelligent Transportation Systems, 24(2), 2296-2306.

[17] Nguyen, D. C., Pathirana, P. N., Ding, M., & Seneviratne, A. (2020). Blockchain for 5G and beyond networks: A state of the art survey. Journal of Network and Computer Applications, 166, 102693.