EncryTrust

A new type of distributed
encryptioncomputing protocol

EncryTrust Protocol White Paper

VERSION 1.0.0

2019.6

Table of Contents

Abstract

Aiming at the Internet data interaction, as well as data security problems such as data fraud, data interception and data leakage in the data transaction process, EncryTrust provides a decentralized trusted encryption computing scheme TAAS (trust as a service) based on blockchain and cryptography. It enables the peer-to-peer blockchain network with high security, high privacy and simple multi-party cooperation of cryptography technology, and thus realizes decentralized data processing and transaction management, releasing the value potential of mass Internet data.

First, the transaction authentication mechanism based on zero-knowledge proof can effectively verify the identity of both parties and improves the efficiency of identity authentication. It can safely filter illegal users without the third-party notary. Second, the data sharing mechanism based on partial homomorphic encryption allows information exchange between the data provider and the recipient to be completed without destroying the sensitive data source. Thirdly, the data processing mechanism based on the secure multi-party computation enables enterprises in an individual/coordinated manner to do various data computing tasks, secures the privacy of inputted data from all parties in the calculation

process, and breaks down the data silos. Finally, the trusted data transaction smart contract platform architecture, with the smart contract coding the trade rules, prevents compliance risks and improves the credibility of trade data processing. This paper aims to provide new ideas and technical support for the privacy protection of data sharing and data transaction on blockchain.

A new type of distributed encryption computing protocol

1. Research Background

Technologies such as mobile Internet, cloud computing and big data have spawned a large amount of user data (identity, e-commerce, social, financial, medical, etc.) while providing more new service models. However, the value of these massive amounts of data is untouched.

On the one hand, the current data operations such as data collection, sharing, distribution, analysis and utilization will lead to direct or indirectly user privacy disclosure, raising people's concerns over data security; on the other hand, there is no suitable infrastructure framework that can technically permits data providers and users with proper, legal, and secure completion of data transactions, processing, and value realization, either individually or collaboratively.

However, as data flow is the new norm now, data sharing, circulation and fusion computing have become the rigid demands. The traditional static isolation security protection methods are no longer apply to protect the data flow security. Causes of risks are anfractuous and interlaced in every

aspect of the big data industry chain: external attacks and internal leaks, technical weaknesses and management defects, as well as new risks triggered by emerging technologies and models and traditional security issues. Many problems in the dynamic use of data have not been properly addressed. The primary issues are presented in the following aspects.

1.1 Increasingly serious data theft and leakage

Due to the great value and centralized storage management model, big data has become a key target of cyber attacks. Ransomware attacks and data leakage problems against big data are becoming more and more serious, and major data security incidents are frequently occurred. The number of global data leakages from 2013 to 2018 is on the rise. Data theft and leakage will pose serious security threats to individuals, enterprises and even countries. Data leakage has become a key factor restricting the development of the digital economy.

Frequent cross-border flows of data can engender new security risks in addition to the possible data leakages. As the attack is more focused on intervening and manipulating analysis results rather than stealing data and paralyzing the system, the damage it causes is more subtle and imperceptible data deviation instead of direct and easy-to-detect system

downtime and information leakage, which may escalate the cyber security incidents to industrial production safety accidents.

1.2 Gradually emerged confidentiality protection issues in dataprocessing

Data processing is an unique application scenario of big data. In the data processing stage, new and more valuable data products are formed through analyzing and mining the collected multi-source heterogeneous data.

More and more enterprises and organizations need to collaborate with industrial chains to carry out production activities based on data flow. In the data exchange phase, through sharing or trading collected data and analysis results with internal organization or external organization, static data information can be transformed into dynamic circulation production information, realizing the data value in the circulating process.

In data cooperation and sharing between enterprises and organizations, data will flow through the boundaries of organizations and systems, realizing cross-system access or multi-party data aggregation for joint operations. The security of personal information, trade secrets or unique data resources in the cooperation is a prerequisite for enterprises or

organizations to cooperate over data sharing, and also a must-be-solved problem to ensure the orderly data flow.

Massive multi-source data is converged on big data platforms. A data resource pool for multiple data providers and data consumers at the same time, strengthened data isolation and access control, as well as accessible but unchangeable data, are the new requirements for data security in the big data environment.

1.3 Complicated data flow path and imperfect tracking technology

The complex big data application as well as frequent data sharing and exchange complicates the data flow path. The whole process, from producing data to destroying data, is no longer a one-way, single-path simple flow mode and not limited to intra-organizational flow anymore. Instead, the data will flow from one data controller to another.

In this process, it is more difficult to trace the cross-bordered data controller in a heterogeneous network or the source of full-path data in safety domains, especially the credibility of data in data provenance, as well as the binding between data markers and data content. Since the data is no longer under the owner's control after circulation and transaction

and the data tracking technology is not mature enough to track the final destination and usage of the data, the data is in danger of losing control.

In the data sharing process, traditional data access control technology cannot solve the problem of data authorization management and data flow tracking across organizations. It is difficult to realize real-time monitoring and auditing of data processing activities of data receivers with mere contract or protocol, which easily leads to data abuse.

1.4 Existing data trading methods are difficult to meet market demand

Trading is the basic way to reflect the value of data, and also the key to data revitalization and efficient data usage. The typical data transaction model is the agent model with three roles: data provider, data agent (mediator), and data consumer. The data provider owns the data and sells it to the data agent, the data agent can sell both the original data and the analysis results with higher value based on the original data after mining to the data consumer. In the process of data transaction and circulation, as the owner no longer controls the data, the data agent holds the ownership of the data. Therefore, the data security protection capability of the agent determines the security of the data in the process of analysis and mining, data operation and maintenance.

At the current stage, data providers are often reluctant to participate in data sharing for compliance and security reasons. The trading platform regards itself as an authoritative and credible third party. It is difficult to achieve true trust in the absence of a well-developed data protection mechanism. Only by establishing an information platform that conforms to data characteristics and ensuring the security and rights of data through technical mechanisms rather than just commitments, can the smooth flow of data be guaranteed and accelerated.

As an emerging technology that combines decentralization, transparency, and non-tamperability, the blockchain is now viewed as the most promising technical paradigm for realizing the value potential of Internet data. As developers design applications with different functions based on the blockchain, blockchain technology has extended its application to financial credit, product traceability, Internet of Things, social, e-commerce and other business areas, and the privacy protection is growing from the protecting the anonymity in the transaction to the ownership and use rights of the original data. Completion of the compliance data processing, application and transaction becomes a big challenge under the premise of protecting user identity privacy and transaction privacy. This paper will focus on the privacy protection of blockchain data processing and data transaction.

2. Building TAAS encryption computing

In view of the current security problems such as data theft and data leakage as well as the rigid demand for secure and reliable data transactions, this paper proposes to build a new type of distributed encryption computing protocol-EncryTrust, providing a complete set of TAAS (Trust-as-a-Service) solution for platforms and users who need to process and transact a large number of data, to maximize the data value potential in the Internet era.

TAAS distributed encryption computing scheme, through zkSNARK (Zero-Knowledge Succinct Non-Interactive Argument of Knowledge), partial homomorphic encryption and secure multi-party computation and other technologies, realizes multi-party addition and multiplication calculation in cryptograph, thereby to realize massive data processing, transaction and circulation without distorting and leaking the data.

As a Layer 2 solution, EncryTrust can access various public chains to provide encryption computing as well as privacy protection services for different public chains to meet the privacy and security computing needs of different nodes in each public chain. At the level of technical design, EncryTrust is not a specific encryption algorithm, but a combination of multiple encryption algorithms and privacy computing technologies.

Overall, the protocol involves several functional modules: transaction authentication mechanism based on zero-knowledge proof, data processing mechanism based on partial homomorphic encryption, trade execution mechanism based on secure multi-party computing, and transaction system based on EncryTrust protocol which includes distributed file exchange system with encrypted communication, data verification engine, c-side application support, high-quality fundamental dataset, and MPI-based multi-party secure communication mechanism.

Below is the technical architecture diagram

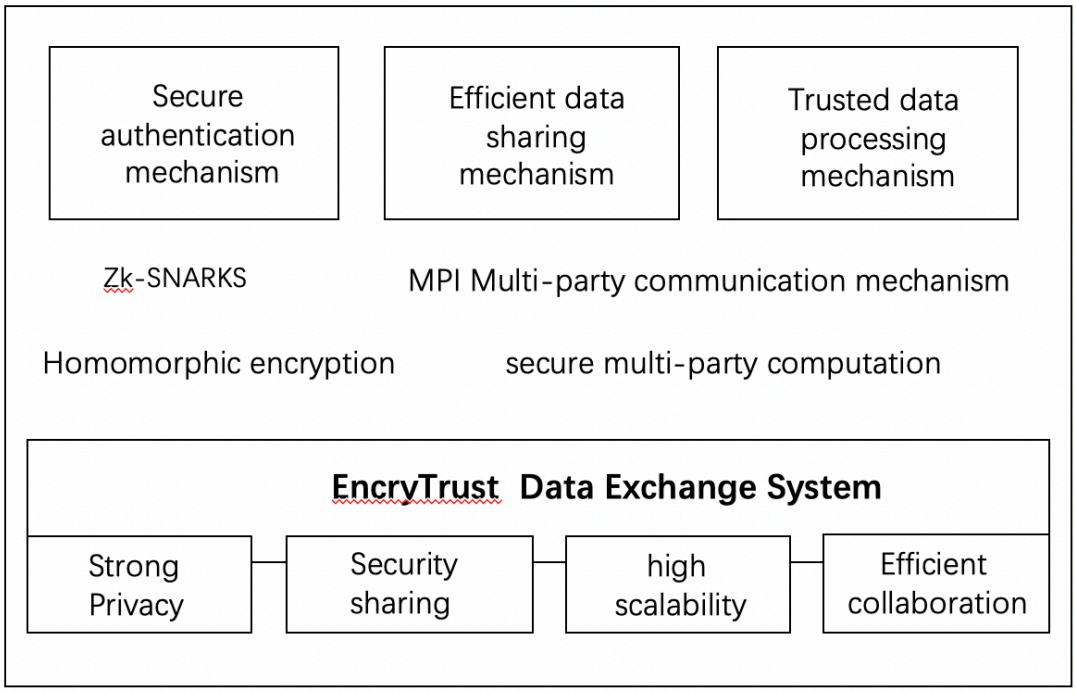

Figure 1. TAAS encryption computing service solution

Characteristics of the complete encryption calculation protocol:

Allow both parties in the data transaction in the blockchain network to quickly and accurately verify the identity of each other.

Allow users in the blockchain network to conduct secure, trusted, peer-to-peer data transactions.

Allow bulk transfer, processing and calculation of massive data on blockchain without leaking specific data information.

Allow multiple data providers (enterprise/enterprise) to achieve reliable results at a lower cost without compromising their own data and without trusting a third party, in order to break the data silos between enterprises.

Enable to build a high performance and a highly scalable data transaction system based on this protocol.

Based on the above characteristics, we named the protocol EncryTrust (short for Encryption Trust): it is encryption trust or trust empowering. The technical details of the protocol will be discussed in the following article.

3. The transaction verification mechanism based on zero knowledge proof

Channeled with smart contracts, blockchain technology has been used in various fields such as finance, insurance, lottery, voting, supply chain, notary, real estate, Internet of Things and cloud computing in recent years. When smart contract on blockchain interacts with the real world, it is often necessary to input a large amount of information for correlation calculation, and execute the results in accordance with established rules.

However, many real information and data entered into smart contracts are often not widely available to the public. Providing authenticated real data to smart contracts (ie, guarantee the authenticity and privacy of the data at the same time) is critical, without revealing any data privacy.

Zero-knowledge proof is that the prover knows or owns information, and the prover can make the verifier believe that their assertion is correct without providing any private information to the verifier. Zero-knowledge proof is essentially an authentication protocol, and if it can be applied to data authentication, it will provide effective solutions to many problems. Among them, zk-SNARKs (zero-knowledge Succinct Non-interactive Argument of Knowledge) is the latest widely used

zero-knowledge proof technology, and has been successfully applied to the anonymous digital currency ZeroCash.

An ideal data transaction process is where two sides of the data exchange can remain anonymous and can exchange data without knowing each other’s identity and personal information when still ensuring the minimized data transaction risk and low cost.

User privacy is one of the major concerns when it comes to ETS system. By adapting information hiding technology based on bilinear pairing theory and zkSNARK zero-knowledge proof technology, users can choose to submit encrypted, hidden personal private information as part of identity authentication. As a result, even if the authentication node is under attack, the user's private information will not be disclosed.

The ETS privacy protection subsystem uses zero-knowledge proof technology to enable authentication of user private information, thereby encrypting user information.

The basic functions of the ETS privacy protection subsystem includes key issuance, setting identity information and verifying identity information to revoke the key certificate.

3.1 Relevant theories

In order to realize the concealment of user information, the ETS privacy protection subsystem uses zk-SNARK zero-knowledge proof technology. The implementation process is as follows:

First, convert the program that needs to be proved (for example: hash function) to the constraint system; secondly, transform the constraint into QAP (Quadratic Arithmetic Program) [2],[3], which is to convert the numerical comparison in each constraint into a comparison between polynomials.

Due to the large number of constraints, the polynomial becomes very complex. However, according to Schwartz-Zippel theorem, when there is no correct polynomial, the required value cannot be output. Since QAP consists of a series of polynomials, when the prover wants to prove that he has knowledge, he has to use the program as a linear combination of polynomials and put his own knowledge as a polynomial parameter into a polynomial for calculation. However, in an open system, the certifier exposing his own parameters leads to knowledge leakage, and the disclosure of polynomial leak causes that anyone can generate proof, which raises a question: how to protect both the polynomial and the

parameters, and whether the results of the substitution of the parameters into the polynomial are correct.

For example:

Here is a encryption function$E()$: Alice wants to give Bob a proof and she has x=3, y=4, x+y=7, and then she sends $E(x), E(y), E(x+y)$ to Bob, Bob calculates whether $E(x)+E(y)$ is equal to $E(x+y)$. If it is, Alice does have the value of x and y. Extending the problem in this example, we can still achieve similar calculations after multiple linear combinations of polynomials, and we actually put $P()$ into the $E()$ for calculation. Alice wants to give Bob a proof, but what is actually needed is that Alice's parameters are substituted into Bob's polynomial for calculation. Both parties do not want to disclose their information, so they use the technique that mentioned above to verify whether the polynomial is equal to the given result. We can achieve this crucial E() function by using the elliptic curve pariring operation. The value after encryption by the E() function can still satisfy the polynomial operations, and the Schwartz-Zippel theorem can also be used to estimate the equivalence. The specific process is as follows:

Alice produces 4 polynomials L, R, O, H

Bob produces a random number $s \in \mathbb{F}p$,and then calculates $E(T(s))$

Alice sends the result of the encrypted calculation of s to Bob.

For example:

$$E(L(s)), E(R(s)), E(O(s)), E(H(s))$$

Bob checks if the equation is true

$$E(L(s) \cdot R(s) - O(s)) = E(T(s) \cdot H(s))$$

However, from

$\mathrm{E(L(s))}, \mathrm{E(R(s))}, \mathrm{E(O(s))}, \mathrm{E(H(s))}$and $\mathrm{L(s)}, \mathrm{R(s)}, \mathrm{O(s)}, \mathrm{H(s)}$,

we can infer that the result is not a specific generation and it can not meet the nature of zero knowledge, it is necessary to introduce some random disturbances that do not affect the verification into the result. This is called random T-shift:

$$\mathrm{L_z} := \mathrm{L} + \delta_1 \cdot \mathrm{T}, \mathrm{R_z} := \mathrm{R} + \delta_2 \cdot \mathrm{T}, \mathrm{O_z} := \mathrm{O} + \delta_3 \cdot \mathrm{T}$$

So far, Alice still needs to send "s" to Bob, and there is still no non-interactive feature. Next we will discuss how to add this feature. The CRS (COMMON REFERENCE STRING) model is used to introduce the random number "s" at the beginning of the system establishment:

$$(E_1(1), E_1(s), \dots, E_1(s^d), E_2(\alpha), E_2(\alpha s), \dots, E_2(\alpha s^d))$$

Disclose the string.

Refer the encrypted value constructed by the random number “s” when generating the proof and verification certificate.

Generate proof

Select the desired value from the public parameters and calculate

$$a = E_1\big(P(s)\big), b = E_1(aP(S))$$

Calculate and verify

Select x and y and then calculate

$$a = E_1(x)\ b = E_2(y)$$

$$E(ax) = e(E_1(x), E_2(a), E(y) = e(E_1(1), E_2(y))$$

If two functions evaluate to the same result, it means ax=y, a sign that the process is validated.

3.2 Key issuance and setting the identity information

The ETS system adapts the zero-knowledge proof technique[33], which enables the user's private identity information to be authenticated while protecting the user's private identity information from being leaked to the network, thus realizing the user information privacy [34].

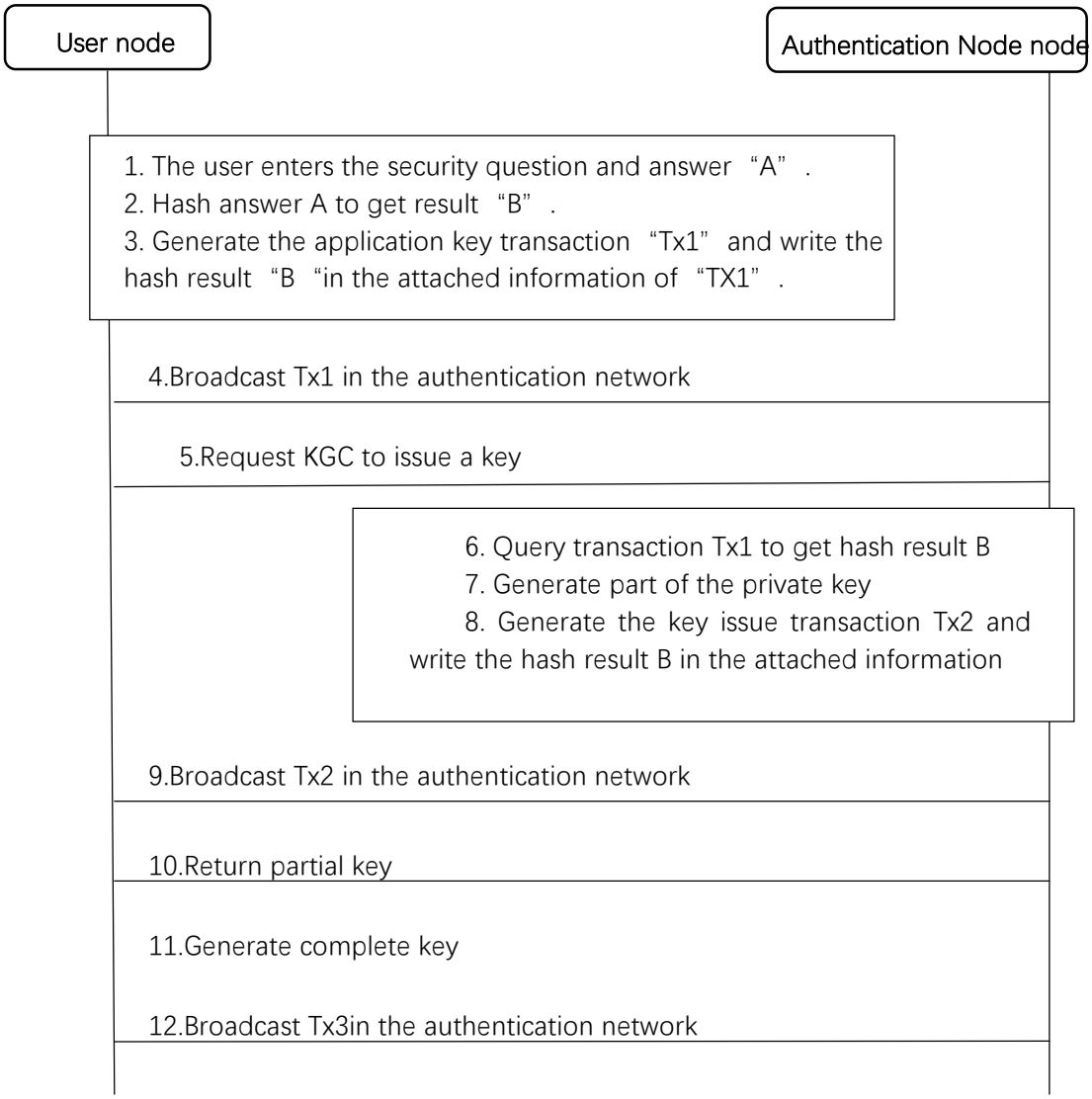

Figure2.Flow chart ofKey issue and identity information setting

The implementation process of the basic functions of the system, including key issuance, setting identity information, verifying identity information, and revoking key certificate, will be discussed in detail.

Set the identity information implementation process:

- The user node enters the security question and its answer A;
- The user node hashes the answer A to the security question to obtain the hash result H(A);

- When requesting to issue a key to the authentication node through the network, the user synchronously transmit the H(A) to the authentication node;

- When the authentication node generates and issues the key, it includes H(A) in the incidental information of the transaction.

Through the above four steps, the user's identity information is set in the identity authentication system.

Verify identity information and revoke key:

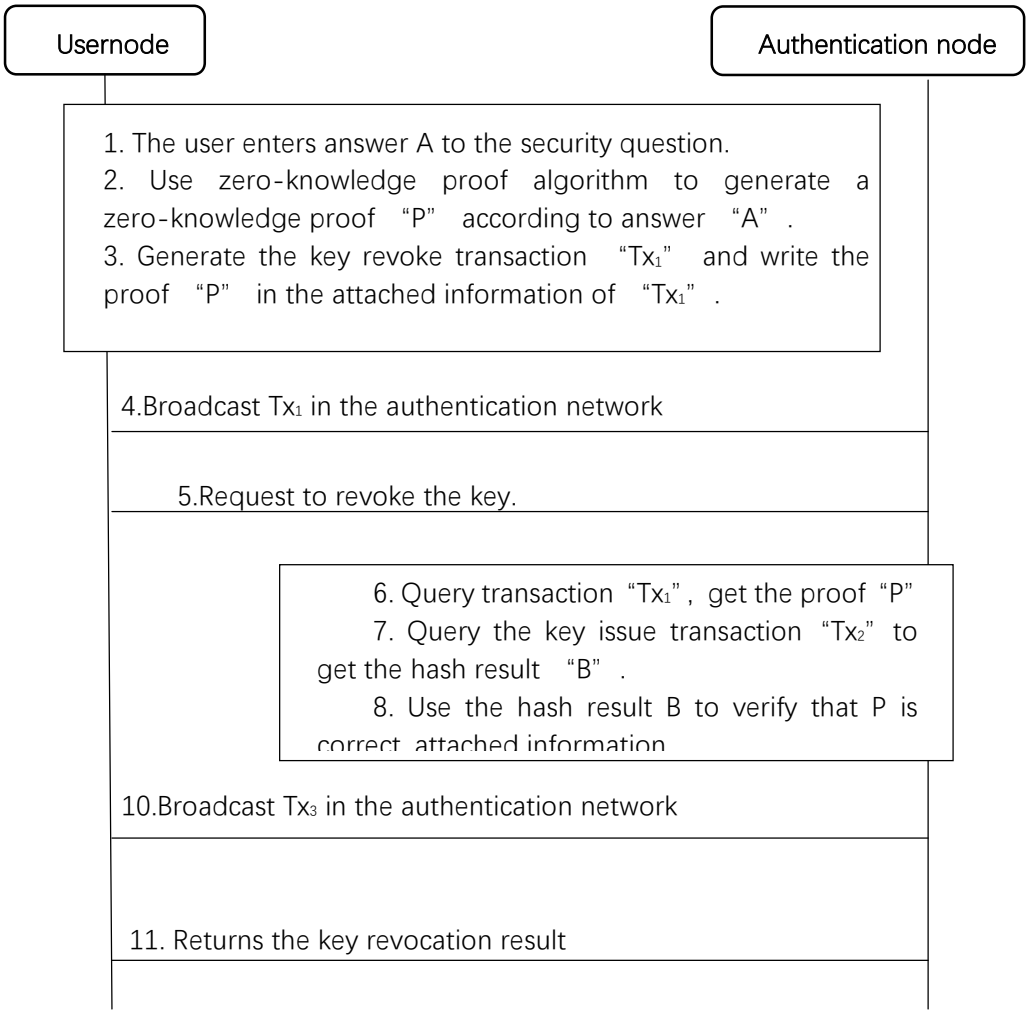

Figure3. Flow chart of authentication information and revocation key

The process by which a user revokes a key using identity information is as follows:

- The user node enters the answer A to the security question;
- The user node sends the proof C in the zero-knowledge proof algorithm by using the answer A to the security question as a parameter.

Any node in the authentication network can use proof C and hash result B of previous security question answer as parameters, plus with the

validation function in the zero-knowledge proof algorithm to check whether the person who generated the proof has answer A to the security question. All nodes are not aware of the answer A to the security question when verifying the proof C and the hash result B, but can verify whether the user who generated the proof C has the answer A to the security question.

- When the user node generates the application to revoke the key, they will include proof C in the incidental information of the transaction.
- The authentication node finds the hash result B written in the key insurance and the proof C in the key revocation application, and uses validation function in the zero-knowledge proof algorithm by inputting B and C as parameters to verify the proof C.
- If the proof C is verified, the authentication node will generate a key revocation transaction to revokes the key of the user.

4. Data sharing mechanism based on blockchain and homomorphic encryption

In many real-life situations, users inevitably need to interact with other individuals or companies in order to gain access qualifications or access

to various services. However, it is precisely during these data interaction that the user's various data is leaked, stolen or even illegally collected and sold.

Therefore, we are in urgent need of a solution that can protect user data and sensitive information (ie, not releasing the decryption key of sensitive data), and at the same time can guarantee the legitimacy and verifiability of the transaction (ie, allowing users to perform friendly, reliably data interaction and sharing).

The homomorphic encryption algorithm in cryptography has this potential. Homomorphic encryption, also known as privacy homomorphism, enables operations on encrypted data without decrypting the information. The advantage of this technology is that it maintains the data and transaction privacy while still calculating the data. Also, only the decryption key can access the details of the data and transactions, which makes the data on the public chain private.

For example, it allows an insurance company to determine whether to settle a claim without knowing the medical health record history and the ID of claim adjuster; it allows the e-commerce platform to assess the user’s credit without obtaining the user's shopping records and shopping IDs.

Homomorphic encryption is a symmetric encryption algorithm that allows the user to perform a specific operation on the cryptograph directly. The result of its decryption is the same as that for the plaintext.

Homomorphic encryption was originally proposed by Rivest and other few people in 1978 as an encryption transform technique that allows direct operation over cryptograph.

If two cryptographs $E(x)$ and $E(y)$ equals $E(x+y)$ after operation, that is, $F(E(x)\ ,\ E(y)) = E(x+y)$, F representing an arbitrary operation, E is an addition homomorphic algorithm.

If $E(x)$ and $E(y)$ equals $E(x\times y)$ after operation, that is, $F(E(x)\ ,\ E(y) = E(x\times y))$, then E is a multiplicative homomorphism. If $E(x)$ and y equals $E(x\times y)$ after operation, that is, $F(E(x), E(y) = E(x\times y))$, then E is a mixed multiplicative homomorphism. If an encryption scheme satisfies both the addition homomorphism and the multiplication homomorphism, it is called fully homomorphic encryption.

The purpose of fully homomorphic encryption is to find an encryption algorithm that can perform any number of addition and multiplication operations on the encrypted data, so that the result of a certain kind of operation on the encrypted data is exactly equal to the encrypted cryptograph after pre-operation.

The principle of the fully homomorphic encryption algorithm is shown in the figure:

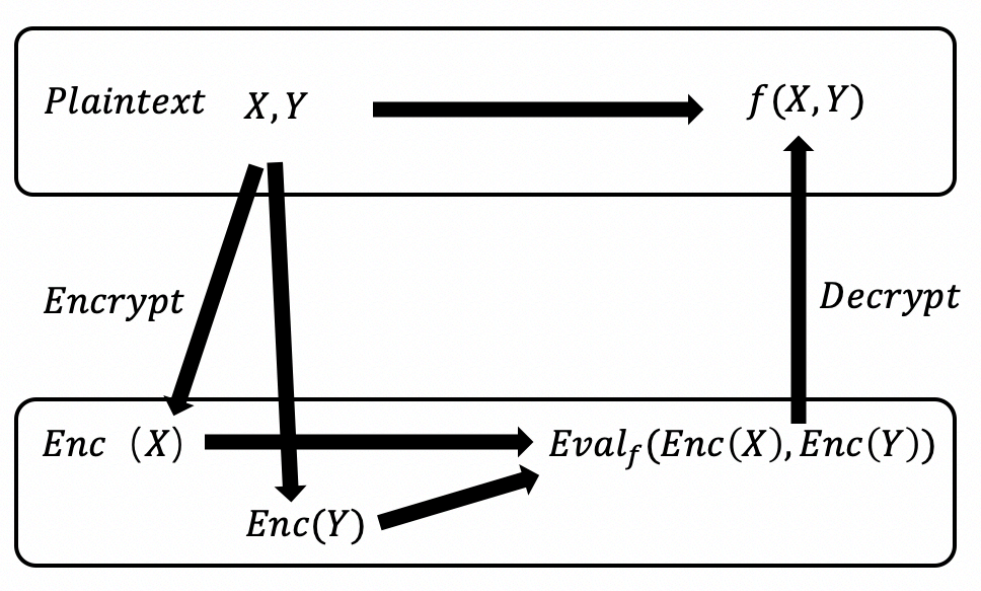

Figure4. Homomorphic Encryption Schema

The fully homomorphic encryption can realize the operation on cryptograph by the keyless party, which can reduce the communication cost and transfer the computing task, thereby balancing the computational cost of each party; in addition, fully homomorphic encryption enables the decrypting party become the only recipient of the final result without being able to obtain message and homomorphic calculation method of each cryptograph, which can improve the security of the information. Due to the advantages of fully homomorphic encryption technology in computational complexity, communication complexity and security, more

and more research forces are investing in the exploration of its theory and application.

In 2009, Craig Gentry made the first fully homomorphic encryption scheme [4], which immediately set off a wave on fully homomorphism research. A large number of follow-up work also emerged. The representative fully homomorphic encryption schemes include the DGHV scheme [5] based on the integer, the BGV scheme based on RLWE[6] and the GSW scheme based on the approximate eigenvector [7].

This paper proposes a data sharing scheme based on the GSW scheme.

In 2013, Gentry and other people used the “approximate eigenvector” technique to design a hierarchical fully homomorphic encryption scheme without computing keys, which can meet the homomorphic capability requirements in many application scenarios in a simplest way, and it has no need to involve the computing key in the homomorphic calculation.

The following is a brief introduction to the basic theoretical principles of the scheme GSW. In order to clearly describe, we added the parameter setting algorithm Setup and decomposed the key generation algorithm into a private key generation algorithm and a public key generation algorithm. As we noted that any circuit can be decomposed into a series

of addition and multiplication operations, we also decompose the homomorphic calculation algorithm into homomorphic addition and homomorphic multiplication.

For a detailed description, we need two tools: one is LWE proposed by Regev[8] to ensure security, and the other is a matrix G proposed by Micciancio and Peikert [9] to support homomorphic operations.

Considering the finite field Zq, the decision type LWE is to Distinguish $(B, sB + e)$ from(B, u). Here, $B \leftarrow Z^{(n-1)m}, s \leftarrow Z_q^{n-i}, e \leftarrow D^m, u \leftarrow Z_q^m$ .Regev proved that LWE is difficult. It is now widely accepted that LWE is even resistant to quantum attacks.

• We assume that G is a public matrix with the following properties: For arbitrary u, it is easy to extract the short vectors $G^{-1}(u)$ that satisfies $G\ G^{-1}(u) = u$.

Now let's describe the scheme GSW in detail:

· Setup$(1^n, 1^l)$: Given the safety parameters n , the maximum homomorphic depth L , select the common parameters $prms = (n, m, q, D)$, set $K = [logq] + 1$。

· SKGen(prms) : Pick $s \leftarrow Z_q^{n-1}$ at random , set the secret key$sk = t =$

$(-s||1) \in Z_q^n$。

- PKGen(sk)：Pick $B \leftarrow Z_q^{(n-1)\times m}$at random, Extract the gaussian error $e \leftarrow Dm$,Calculate $b = sB + e$, set the public key $pk = A = (B, b) \in Z\_q\^(n\times m)$。

- Enc(μ,pk)：Given the plaintext message $\mu \in \{0,1\}$， pick the matrix $R \leftarrow Z_2^{m\times nk}$at random , and then calculate the ciphertext$C = AR + \mu G \in Z_q^{n\times nk}$

- Dec(sk,C)：set $w = (0,0,\dots,q/2)^T$, firstly calculate $tC - G^{-1}(w) = \mu tw + e = \frac{\mu q}{2} + e$,Then deciding whether the message 0 or 1 should be sent based on the size of the value.

  Note: as long as the ciphertext satisfies the form (called decryption form) and is a small matrix, it can be successfully decrypted.

  Add :$\mathrm{C}_1 \oplus C_2 = C_1 + C_2 = \mathrm{B}(R_1 + R_2) + (\mu_1 + \mu_2)\,\mathrm{G}$, It can be seen that homomorphic addition satisfies the decryption form.

  Multi：$C_1 \oplus C_2 = C_1 - G^{-1}(C_2)$

  Note that $C_1 G^{-1}(C_2) = (\mathrm{B}R_1 + \mu_1 \mathrm{G}) G_{-1}(\mathrm{C2})$

$$= \mathrm{B}R_1 G_{-1}(C_2) + \mu_1 \mathrm{G}\, G^{-1}(C_2)$$

$$= BR_1 G^{-1}(C_2) + \mu_1 C_2$$

$$= \mathrm{B}R_1 G_{-1}(C_2) + \mu_1(\mathrm{B}R_2 + \mu_2 \mathrm{G})$$

$$= \mathrm{B}(R_1 G_{-1}(C_2) + \mu_1 R_2) + \mu_1 \mu_2 \mathrm{G}$$

So homomorphic multiplication also satisfies the decryption form.

The data sharing scheme based on homomorphic encryption is expected to be widely used in all kinds of privacy data sharing scenarios, where the fully homomorphic encryption can be used to support basic data sharing operations.

5. Data processing mechanism based on secure multi-party computing (SMPC)

Existing blockchain techniques only support simple scripting languages on a specific instruction set and are not sufficient to support the execution of complex contracts. In order to maximize the rights of the data contractors and avoid the disclosure of private information due to contract execution, participants in smart contracts involving data transactions are often required to have privacy and the ability to resist adversary attacks during contract execution. However, the existing

trading instruction set is limited to the integrity and ownership authentication of the contract data. There is no necessary security measure for the privacy of the participant data, and the security risk in the script operations also exists.

After researching the security problem of smart contract execution in the blockchain, we propose a smart contract based on secure multi-party computation (SMPC), a secure multi-party computing algorithm for linear secret sharing, and adopt the non-blocking MPI in the MPI communication mechanism to support the mutual communication between the SMPC parties. It enables the participants can communicate correctly, and systematically gives the SMPC-based intelligent contract execution flow, and is a fair SMPC method with privacy of input data and correctness of the calculation.

5.1 Secure multiparty computing

In cryptography, SMPC means that multiple parties can perform a computational task in a coordinated manner, and maintain the privacy of the input during the operation process, and the consistency of the final calculation results.

Specifically, SMPC is that, n participants P_1 , P_2 , $\cdots$ P_n need to perform a certain computing task together.

$$f(x_1, x_2, \cdots, x_n) = (y_1, y_2, \cdots, y_n)$$

Each party "P_i" can only get its own input "x_i"and can only get its own output"y_i". SMPC has the following security requirements:

- Loyalty: most of the participants are loyal, and loyalty means that the participants perform the tasks in full accordance with the regulations.
- Termination: in a limited time, the loyal party can terminate the execution of computing tasks.
- Privacy: no participant can get input of other participants.
- Consistency: all loyal parties end up with the same output.

If there are fewer than t participants are dishonest in a domain of size$n$ (mostly participants are loyal, such as $t < n/3$), SMPC can provide completely secure and trusted calculation results.

During the execution of the SMPC, the computing tasks of all parties are consistent, and there is no centralized special node. This feature can guarantee the fairness of the smart contract that introduces SMPC. The privacy of SMPC are reflected in the smart contract, and the privacy of all parties' input data to the contract are protected.

Figure 5 is a flow chart of the SMPC addition and subtraction method. According to the data flow diagram given by the SMPC addition and

subtraction flow chart, assuming that we have n participants, taking calculating the sum of a and b as an example, the flow graph is shown in figure 6.

As the execution process of the SMPC algorithm is established on the MPI multi-party communication mechanism and the MPI supports non-blocking communication, it can meet the threshold as "t" in the SMPC. However, when sharing privacy, since the key fragments that need to be sent to each party are different, the distribution function in the MPI is adapted in order to send the key fragments participants involved in the operation. When the parties receive the key fragments through the MPI non-blocking communication, they calculate it, and then send the calculation result to the reconstructing party through MPI non-blocking communication. The reconstructing party collects the calculation results of other parties through the collection function in the MPI, and secretly reconstructs the received results to obtain the final result.

The addition and subtraction of SMPC is mainly divided into the following three stages:

1) Secret sharing

Based on the Shamir's secret sharing scheme [10], and use the Lagrange interpolation formula to complete the basic secret sharing of threshold(t, n). The process is as follows:

In F_p, for a given secret a, pick $t-1$ random numbers $(r_1, r_2, \cdots, r_{t-1})$, Let $r_0 = a$ and form a polynomial equation $f_a(x) \sum_{i=0}^{t-1} r_i x^i$. For any participant with identity in distributed computing P_i (where, i $\in [1, n]$), the shared value of secret a obtained is $a_i = f_a(x_i)$;

In the same way, for a given secret b, pick $t-1$ random numbers $(l_1, l_2, \ldots, l_{t-i})$, Let $l_0 = b$ and form a polynomial equation $f_b(x) \sum_{i=0}^{t-1} l_i x^i$. For any participant with identity x_i in distributed computing P_i (where, i $\in [1, n]$), the shared value of secret a obtained is $b_i = f_b(x_i)$.

2) Calculation stage

Each participant $P_i$receives the value that needs to be calculated through the receiving function through the MPI non-blocking communication, performs respective calculations, obtains the calculation result "c_i", and transmits the calculation result through the transmission function in the MPI non-blocking communication;

3) Secret reconstruction

Participants use the Collection Function in MPI communication to collect the results sent by other nodes, and then secretly refactor them. If the original secret value c is recovered from the result of $\{c_1, c_2, \ldots c_m\}$ calculated by $m(m \geqslant t)$ participants, then $c = \sum_{i=1}^{m} \alpha_i c_i$ can be solved, where $\alpha_i = \sum_{j=1, j\neq i}^{n} j/(j-i)$, $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ is called a recombination vector.

In the pseudo-code, the Collection Function in MPI communication function is used to collect $c_i$sent from the P_i of participants, and then the secret reconstruction is carried out to obtain the final result.

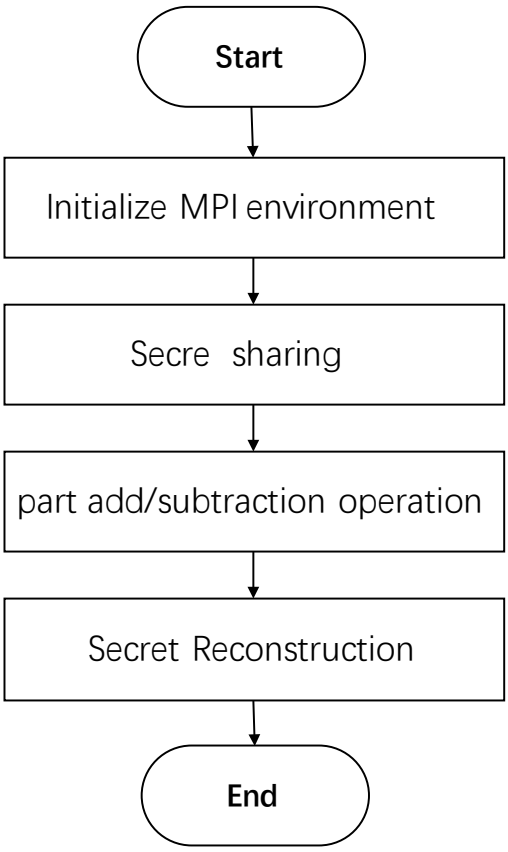

Figure 5. Flow Chart of SMPC addition and subtraction

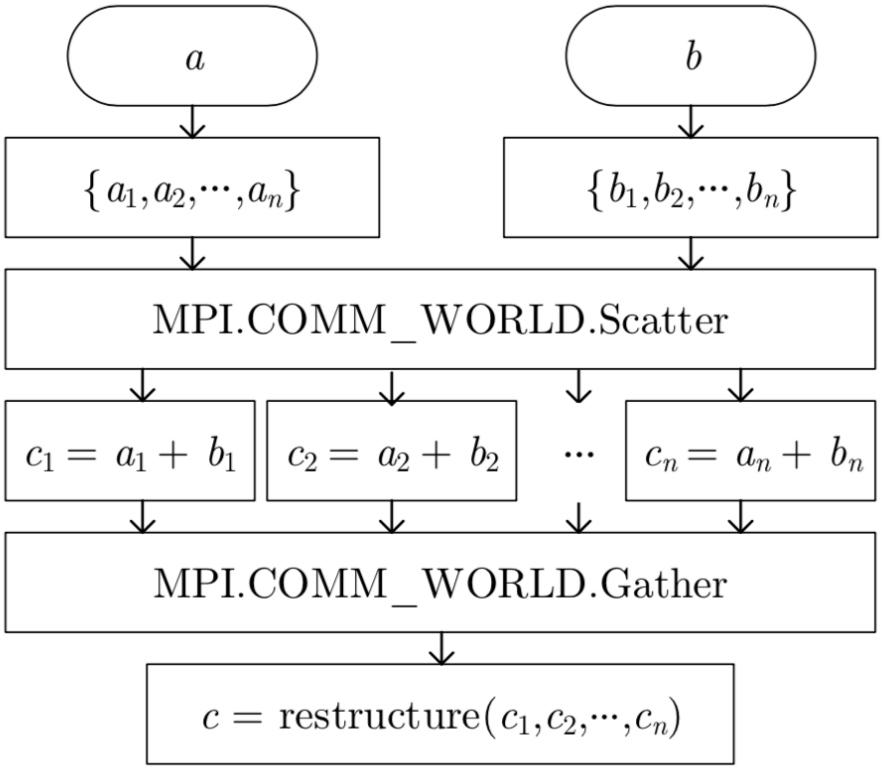

Figure 6. Data flow chart of SMPC addition and subtraction

In order to ensure the anti-attack of the above SMPC process, the verifiable secret sharing mechanism is further introduced as follows: take "p" of the multiplication group Z_q^* ,the generator is g, as to obtain a cyclic subgroup (g). Use the verifiable secret sharing to share a, broadcast $\{g_i = g^{r_i} \, mod \, q\}_{i=0}^{t-1}$ (among which, $p|(q-1)$), Then, by verifying that the equation $g^{f_b(x)} = \prod_{i=0}^{t-1} g^{{l_i}^{x^i}} = \prod_{i=0}^{t-1} {h_i}^{x^i}$ is true, we can verify whether the sharing $f_b(x)$ is correct. By verifying that $g^{f_a(x)+f_b(x)} = \prod_{i=0}^{t-1}(g_i.\ h_i)^{x^i}$ is true, the calculation result of $f_a(x) + f_b(x)$ can be verified to be correct. If at least one party has correct

calculation result, the final correct calculation result can be reconstructed, that is, the threshold value is t。

5.2 Multi-party communication mechanism based on MPI

It is extremely important to realize efficient communication between multiple participants in secure multi-party computing. This paper uses MPI [11, 12] communication mechanism to meet the communication requirements in the secure multi-party computing process.

The functions in table 1 belong to MPI system functions. **Init()**initializes the MPI execution environment, establishes the connection between multiple MPI processes, and prepares for subsequent communication. **Finalize**() function to terminate the execution environment of MPI. The **Size**() function returns the number of processes contained in a given communication domain, that is, the number of participants; The **Rank**() function returns the process number in the given communication domain, i.e., the given participant id.

Table1. System functions in MPI

Method	**Function**
MPI.Init()	Initialize the MPI environment

MPI.COMM_WORLD.Size()	Returns the number of processes in the communication group
MPI.COMM_WORLD.Rank()	Returns the process number of the calling process in the communication group
MPI.Finalize()	Finalize the MPI execution environment

Described in Table 2 is the communication function in the MPI. When MPI is applied to multi-party secure computing communication, the main applied functions are broadcast, collection, dissemination, and fully exchange functions. The secret distribution of the task is the **Scatter**() function which is used to distribute the generated shared fragments to the participants; the parties use the **Gather()** function to collect the key fragments; the **Alltoall**() function is used to complete the message exchange between different parties.

Table2. Communication functions in MPI

Method	**Function**
MPI.COMM_WORLD.Bcast()	Broadcast function in one-to-many communication

MPI.COMM_WORLD.Scatter()	Scatter function in one-to-many communication
MPI.COMM_WORLD.Gather()	Gather function in many-to-one communication
MPI.COMM_WORLD.Alltoall()	Alltoall function in multi-to-Many communication

---

In the SMPC-SC execution system, the above functions are mainly used to implement the underlying communication. There are two types of attacks that possibly exist in the system: truncation attacks and communication privacy attacks. The non-blocking paradigm is used to prevent truncation attack in the MPI, and the communication privacy problem is mainly solved by the secure multi-party computing in the computing layer.

This article prevents truncation attacks by time constraints and threshold limits:

1) Time constraint: complete communication or end communication within a certain period of time.

2) Threshold limit: increase the threshold limit when contracting the time. The number of messages correctly transmitted before the communication is turned off must be no less than the threshold.

In MPI-1.8 version, peer-to-peer communication has satisfied the functions of blocking and non-blocking communication. Blocking calls mean that the current thread is suspended until the result is returned.

Non-blocking communication means that the call does not block the current process until the result is returned.For an effective non-blocking communication, it is required that the communication must be shut down for a limited time while the number of loyal parties exceeds the threshold.

The underlying asynchronous communication of an SMPC is too complex to describe，the simple send and receive functions are described below.

Figure 7 shows the flow chart of the non-blocking communication sending function with time constraints and threshold restrictions. First, a time constraint is established, assuming that n messages are sent to n processes in t time using the Isend function. The final time used is denoted as t_c', and the number of messages successfully sent is denoted as n'. Within time t_c, n messages are sent and communication is shut down. When $t_c' \geqslant t_c$, if more than$\frac{2}{3}n$messages are sent successfully,

then the communication is closed successfully; In other cases, the communication fails.

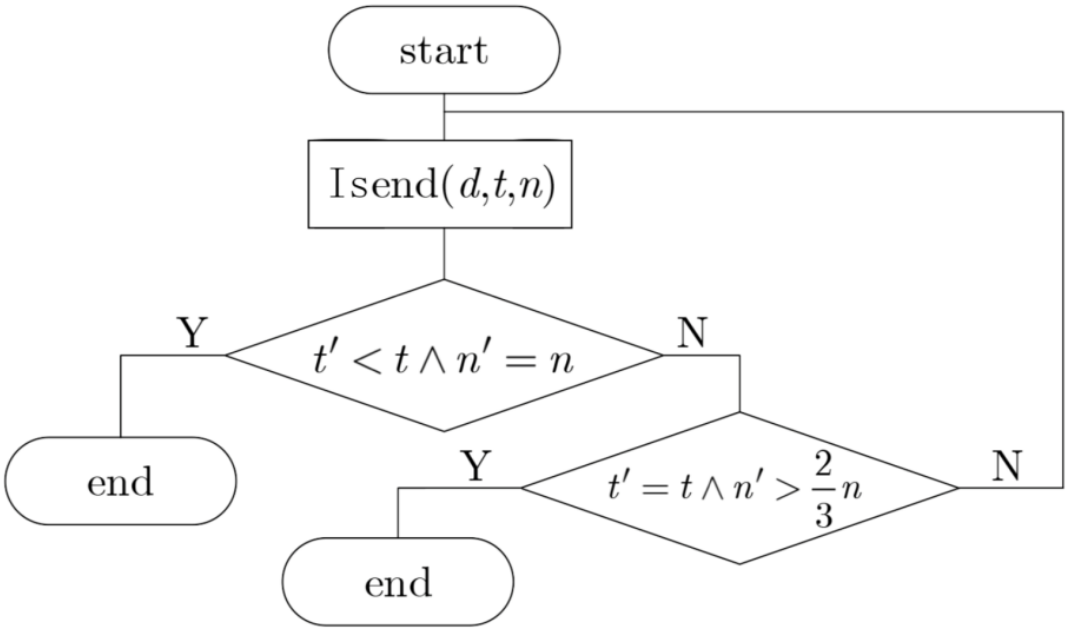

Figure 7. Flowchart of non-blocking send function flowchart

6. Data transaction system based on ETS protocol

Although data flow and transactions are the major trends today, the current data transactions (that is, the storage, transfer and transaction of data) often face the risks of data being stolen and tampered. The ability of data transaction platform to retain transaction data becomes a potential threat to both parties and the biggest obstacle to data trading.

An ideal data transaction process is the process of storing, transferring and trading data while minimizing the risk of data transactions and minimizing costs.

Based on the research of cryptography, blockchain technology and open source mechanism, this paper proposes to design a decentralized new data transaction system (we call it ETS data transaction system), which technically guarantees that the platform has no access to view, copy and keep transaction data, and the transaction data is only safely transferred between the two parties.

The ETS data trading system makes the transaction process more convenient on the basis of guaranteeing the privacy of the data owner. The transaction process and information are transparent and open to both parties, and the communication between the data contributors, owners, users and other parties is secure. Specifically, the system has the following characteristics:

First, the transaction data is highly confidential.

Data is stored and managed by the provider, and the owner controls the data; and zero-knowledge proof technology ensures the security of data transmission information. The operation of transaction is open and

transparent to both parties, but it is confidential to the other parties and protects both parties in the data asset trade

Second, the transaction process is highly confidential.

In the process of data asset trading, data assets can only be acquired by the purchaser. Any other third party, including the platform, cannot obtain data, which reduces the concerns of both parties about the data retained by the platform during the traditional transaction process.

Finally, the transaction is safe.

The transaction is accurate, and once the transaction is initiated, it definitely will be successfully completed. The trading behavior is recorded, inquired, and appealable. Each transaction guarantees privacy and, at the same time, is non-repudiation, and can be tracked, and third-party platforms cannot obtain data content.

6.1 Overall architecture design

The entire ETS Data Trading Platform consists of three parts.

Trading platform

As for the platform, it needs to process a large number of data transactions and information transfers, deal with user requests such as query and authentication, manage flows according to different request

types, cooperate with different nodes in the blockchain network, and transmit information such as account books through the P2P network.

The trading platform is constructed and operated by ETS to build a basic communication structure in order to realize the connection and communication hiding between participants, authenticate participant and control access, maintain public records for index records and transaction records of data, formulate data specifications and transactions rules, maintain transaction order, and assist the transaction party to complete the traceability of data.

Participants.

The provider of the data is responsible for maintaining the backup of the blockchain of the public area and supervising the correctness of the blockchain records in the public area;

Maintain private data that you can share, provide query services, initiate queries, and obtain external data.

Node

Connected with the ETS trading platform, participants realize the backup of the record blockchain on the public trading platform, complete the blockchain generation and submission of its own data, receive messages from the public trading platform, and realize secure communication with

the peer (query and being queried) . It can be developed by the participants themselves according to the open protocol, or the trading platform will provide the source code.

Based on considerations for performance, it is not necessary for all participating nodes to involve in the process of transaction consensus, accounting, etc. Some nodes with low quality will reduce the consensus efficiency of the whole system. Therefore, ordinary nodes are only responsible for some regular operations, some specific nodes will be responsible for transaction consensus, and accounting nodes are responsible for the transaction process mechanism of the entire system, and as a result the consensus and accounting for the results of the whole network transaction are guaranteed. After the transaction reaches a consensus, the accounting node records the result of the transaction, forms a block, and issues the block to other nodes, as well as different nodes respectively verify and submit to the local ledger to complete the accounting process.

This kind of collaborative management mechanism separates the core from the non-core processes. Some simple routine query and authentication operations can be addressed by the ordinary nodes. As for the transaction consensus, confirmation, and accounting with high performance and efficiency requirements, the trading nodes with

high-performance can deal with that, which can avoid waste of resources and improve the efficiency of the network.

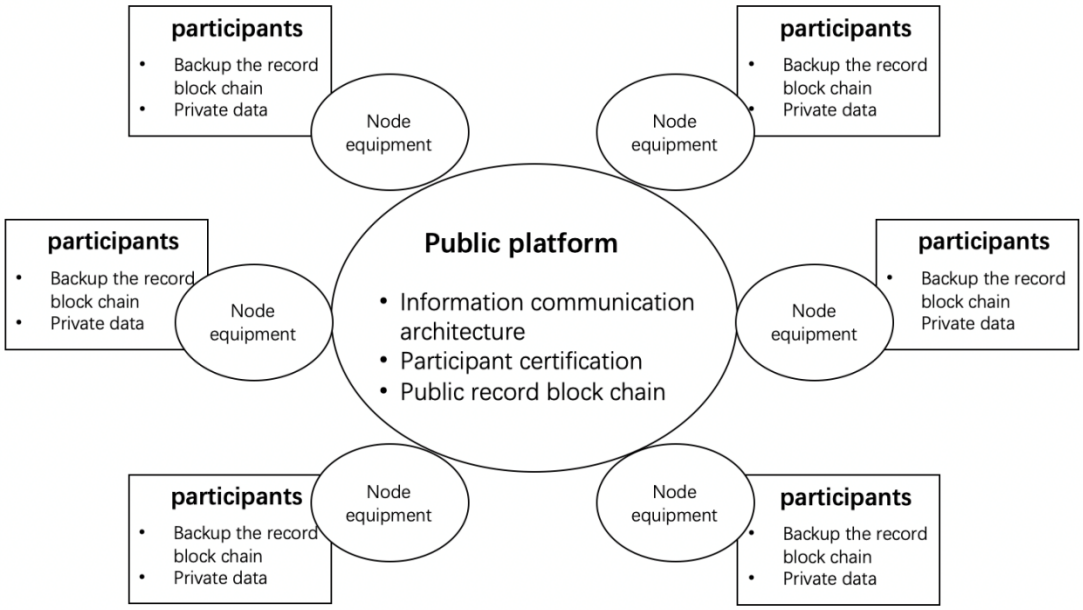

Figure8. Data Trading Platform Architecture

6.2 Joining and authentication of participants

- Participants apply to join, the trading platform confirms the identity of the participants and allows the participants to join, and then participants send digital certificates issued by third-party certification companies (such as CA) or self-generated digital certificates to the trading platform to prove identity, and the platform authenticate the identity with digital certificate.

- The participant generates any number of public-private key-pair pools, signs the public key pool data with the private key of the identity

certificate and sends it to the trading platform. After the trading platform uses the participant's identity certificate for authentication, the public key in the public pool is associated with the party and also serves as the identity of the participant. This step can be initiated at any time after the completion of step 1), or can be initiated multiple times.

The purpose of using multiple public keys is that participants can use multiple identities to publish data indexes, so that other participants cannot know the source of the data. Of course, if real-name transactions are used, multiple public keys are not needed.

6.3 Data index submission

Participants can choose data that they wan to share to the public or trade.

Calculate the HASH value of the subject identification ID (such as ID card + name), extract the subject information description, calculate the HSAH value of the subject data content, and price the data to form the basic data index information.

Select the key pair used for the shared data from the public key pool (the same principal record should keep the same key), add the public key information in the information, and use the corresponding private key to sign the data to form a set of data record application package and send them to the platform.

The data record application package generally contains several basic types of information:

- · The hash value of the subject ID
- · Subject data description
- · The hash value of the data content
- · The data provider's public key
- · Data price
- · Data provider's private key signature

When the above data is updated, the participants should resubmit the data index and description.

6.4 Block generation of data index

The ETS transaction platform receives the record application package, extracts the corresponding public key, verifies whether the public key is in the submitted public key pool, confirms that the public key is owned by the effective participant, uses the public key to verify the record signature and confirms that this information is sent to the corresponding party. The platform will extract the cryptograph index information in the application as a record.

The ETS trading platform aggregates the records sent by all participants, adds them to the current block, and links them with the blocks generated in the previous period.

The generation of the block introduces a random competition allocation mechanism, so that each participant node can obtain the equal opportunity to generate the block, and also the information of each node can be recorded in the chain. It ensures all participating nodes in the chain are jointly maintaining the information. .

The ETS trading platform periodically aggregates the new index and updated index submitted by each participant to generate the newest block onto the chain. The contents cannot be changed and deleted, and only the records can be added. Each participant can download and retrieve the index chain to confirm whether the index provided by themselves is in the index chain and thus supervises the trading platform.

6.5 Data retrieval and acquisition

When the participant needs to query whether there is a certain subject data information in the ETS platform, it is necessary to first retrieve the corresponding index in the platform. The inquiring party obtains the identity information of the subject (such as ID card + name), performs HASH processing on the identity information, obtains the the

cryptograph of identity information, and searches by the cryptograph. This search can be done in two ways.

A. The cryptograph index value is sent to the ETS trading platform, and the query result is returned by the platform.

B. Download the blockchain file of the index from the ETS trading platform, and parse the cryptograph index on the ETS trading platform to retrieve whether there is a subject index that needs to be queried.

The above two retrieval methods are only applied to query whether the enterprise provides the data information of the subject on the ETS platform, and does not return the real data. If it exists, the full index record corresponding to this index can be obtained.

After searching, data can be obtained after knowing that there is a corresponding subject record on the ETS platform. Proceed as follows:

- Querying party accesses the service address in the record;
- Data querying party selects the public-private key pair used in the request, makes the request and sends it;
- Data provider receives the request, extracts the public key information and verifies the signature, ensuring that the requesting party is a legitimate participant;

- The data provider extracts the relevant information of the subject from its own database, encrypts the data with the public key of the requesting party, then signs with its own private key, generates a response packet, and sends it back to the querying party.

- After receiving the response packet, the querying party first verifies the response packet signature by using the provider's public key, confirms that the response packet is sent by the provider, decrypts the recorded data with their own private key, obtains the original data, hashes the original data, and compares it with the previous recorded hash in the block to verify whether it is the record information declared in the block.

  ETZS trading platform aggregates the daily transaction records generated by the platform to form a trading block and forms a chain structure with the previous transaction records. Each participant can download and retrieve the transaction chain to confirm whether their transaction is in the transaction chain, which forms the supervision of the trading platform, and also provides evidence for subsequent transaction traceability and appeal.

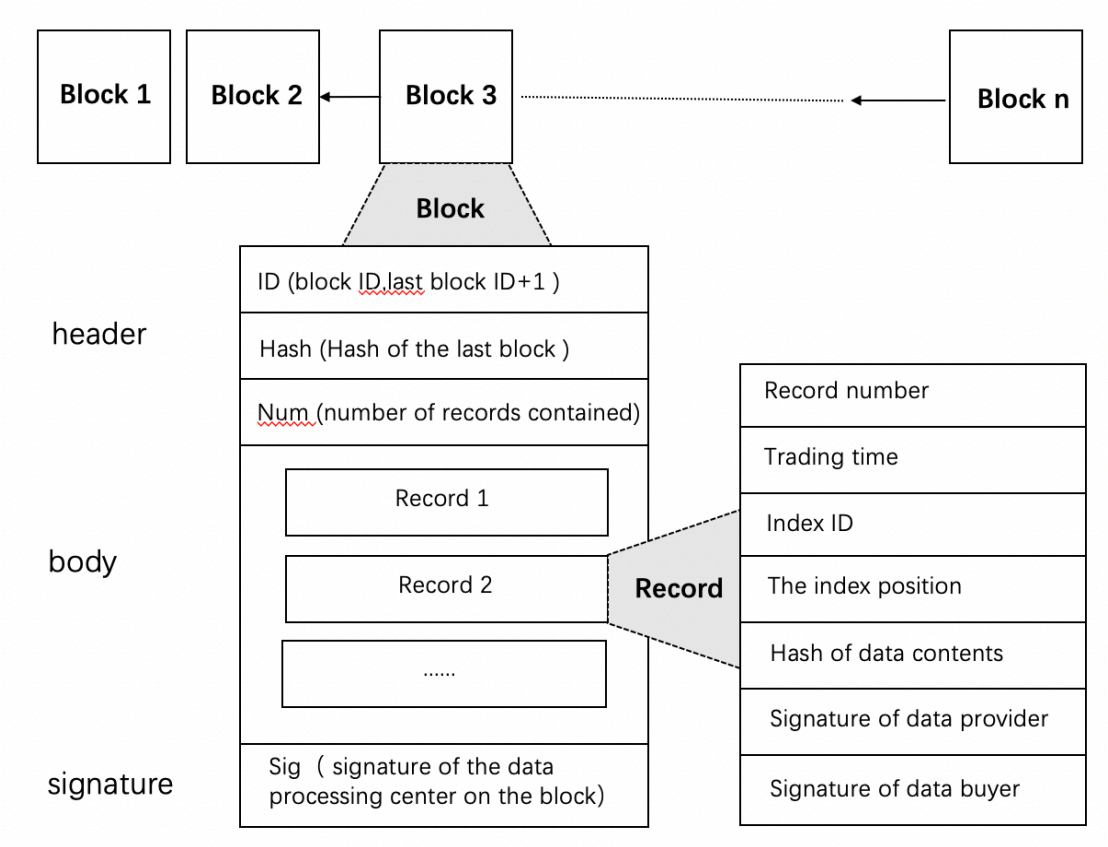

Figure9. Schematic diagram of data trading chain

6.6 Data sharing and application

In the actual application scenario, each enterprise/institution can complete the sharing and collaborative application of various data based on the ETS data transaction platform. As shown in the figure, the enterprise publishes an index of shared data in the form of cryptograph on the ETS trading platform. The cryptograph is a one-way hash algorithm. The original text cannot be derived from the cryptograph alone. No one knows what the companies/institutions have released, and the information has no risk of being disclosed.

The user ID to be queried by the enterprise 1 is encrypted by the same encryption algorithm and then queried in the encrypted index. Anyone can download the query, and the query action and query information will not be disclosed to other companies.

Enterprise 1 can send a data acquisition request to Enterprise 4 and Enterprise 6 that can provide data. Requests can only be sent to businesses that can provide data.

Enterprise 4 and Enterprise 6 return data after receiving the request, and the data is transmitted to Enterprise 1 after encryption with the public key of Enterprise 1.

The company that receives the request must respond because it has been declared in the index and cannot be denied.

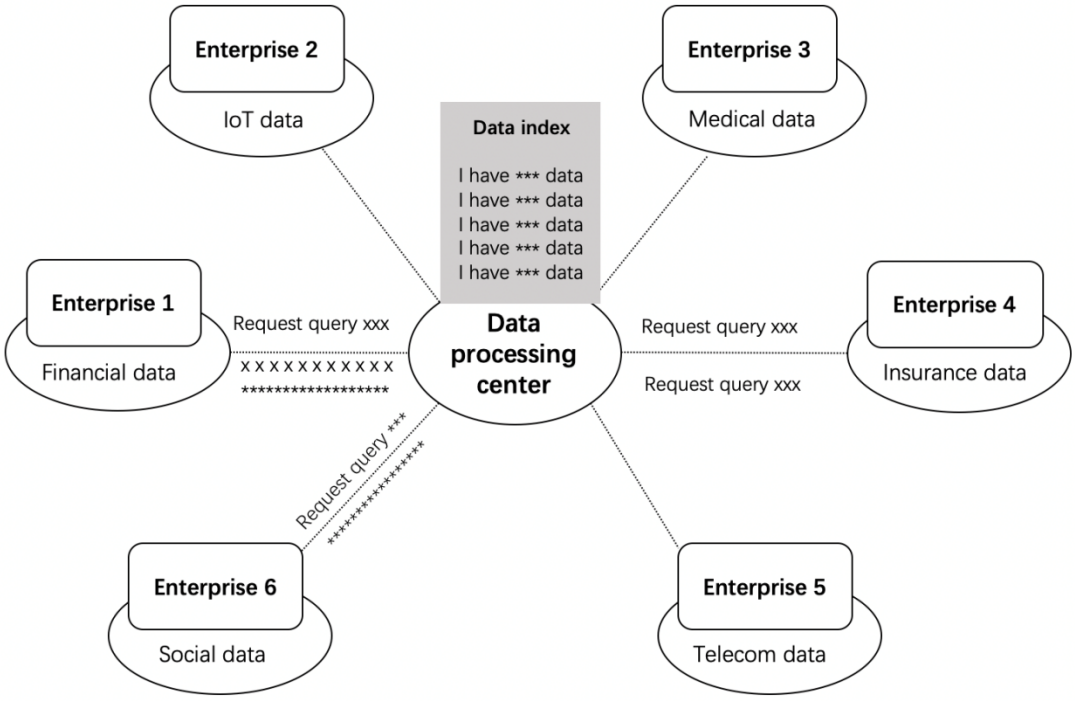

Figure10. Data sharing flow chart

Enterprise 1 receives the encrypted data sent by Enterprise 4 and Enterprise 6 respectively, and decrypts the data using their own private key. The public key encryption of the querying enterprise can only be unlocked by the querying enterprise, and any other third party is unable to decrypt it.

7. Application scenarios

The TaaS (trust-as-a-service) data service solution based on the EncryTrust protocol will provide a high-performance one-stop data privacy solution for the blockchain economy, effectively addressing the data privacy protection and transaction issues in various scenarios, including the medical area, financial field, and attention economy. With more reliable, secure, and efficient data transaction collaboration platform services, participants inthe data service area can easily build various business scenarios, including but not limited to the following core scenarios.

7.1 Financial big data

In the era of big data, all financial companies need to hold comprehensive and real-time customer data, collect all kinds of fragmented data to form

user portraits, accurately assess the credit status of each customer, and thus make corresponding credit granting and other services. This requires different financial companies to exchange and share data with each other to form a complete user portrait. However, at this stage, many financial companies suffer from data silos and data fragmentation because of lack of trust and cooperation channels. There is an urgent need for an emerging technology to enable the circulation as well as reasonable, secure sharing of financial data between different financial companies, and to improve the liquidity and utilization of financial data.

7.2 Health care

In the era of health care and big data, a large amount of medical data is being collected. Simply removing the name and ID card information from a group of medical data and putting them into public use poses great security threats. When this group of data is linked with another group of data, the personal data has the chance of being completely exposed. If the data includes more sensitive genetic data, the threat on privacy and security will become severer. When the genetic data is matched with some pathological data, it is easy to locate the specific individual.

Obviously, the privacy security “black hole” of medical data sharing has become a pressing problem, but the future vision of precision medical

carefascinates countless medical companies and scientific research groups, and also promotes the needs for new technical solutions to release the huge potential of health medical big data.

7.3 Smart Internet of Things

Blockchain technology is applied in the field of Internet of Things, which can effectively improve the centralized data storage mode [13] of the traditional IoT. All nodes in the blockchain network record complete data information, jointly protect the security of the device data of IoT , and reduce the cost of maintaining the centralized database in the traditional IoT application [14]. Therefore, IoT applications based on blockchain are springing up.

However, the open and transparent nature of the blockchain also exposes the transaction information between devices to the entire network, posing a serious threat to transaction data privacy. By analyzing the transaction information and obtaining the device identity information, the malicious user attacks the device. Therefore, it is urgent to apply the blockchain privacy protection technology and related encryption protocols to protect the security of the Internet of Things and to ensure that users cannot obtain detailed transaction information in the network, and the

verification node can only verify the validity of the transaction and cannot obtain specific transaction information.

7.4 Attention Economy

As the digital economyboosted by the Internet is becoming more and more active, big data has become a "treasure" for all players to chase after, and the advertising industry is a prominent field. In the era of information explosion, attention is a scarce resource. When advertisers are trying to attract users' attention, they need to maximize the benefits of each piece of information. This requires sufficient big data analysis and precise positioning in the early stage to build a network to recommend and deliver ads to users.

Advertising strategy holds desperate need for multidimensional communication of enterprise data. For example, whena travel enthusiast visited two websites, the enterprise needs to combine the behavior data of the user on both platforms and conduct in-depth and accurate analysis on the data to adjust relevant promotional strategies. Of course, all major platforms are not willing to take the risk of "contributing" their own data to others. This process requires a secure data transaction platform that restricts the data usage access of all parties, andwiththe encryption technology and privacy masking technology, companies can only obtain

other data after they get the authorization from the company. This method can realize the proper protection and processing of data.

In addition to the above scenarios, data privacy computing service are also widely used in various industries such as social, insurance, telecommunications, education, etc. The process of adding data onto the blockchain in almost all practical application scenarios will face the problem of data privacy protection. Also, as the blockchainservesa open, transparent, non-tamperable decentralized network, it cannot be put into any real application if the data ownership and usage rights cannot be reasonably regulated.

8. Token economy model

ETS is the core assets in the EncryTrust system with a total circulation of 2 billion. ETS has a wide application value in the data economy, providing an important value medium for the blockchain in the data economy and large-scale commercial applications. In general, the application value of ETS mainly has the following aspects:

EncryTrust development resource consumption, consumption of various system resources on EncryTrust, data transaction usage scenarios, investment value circulation, etc.

Table 3. ETS application scenario description table

Applications	ETS Consumption Description
Consumption of ETS development resources	· Registration to be a developer； · TAASAPI call；
Consumption of various system resources on ETS	· Creat an account； · Upgrade the account； · Call the data transaction service； · Call a smart contract； · Use all types of infrastructure within the EncryTrust ecosystem.
Data transaction usage scenarios	· When dApps need to use enterprise/personal data, the enterprise/individual grants dApps to use their personal data to obtain ETS rewards. · When the company/individual needs to view other people's data, it needs to be authorized by the party and need to pay the corresponding ETS.
Investment Value Circulation	· Institutions/individuals providing false/deceptive data services will be fined for pledge of ETS. · The scarcity and strong application requirements of ETS itself support its huge circulation value. · Users can trade ETS in major exchanges around the world, sharing the value of EncryTrust's ecological growth.

1) Become an ETS node and get a trade verifier reward

2) Obtain ETS by completing an activity or task

3) Get ETS rewards for community contributions

4) Earn service fees by developing applications ETS

5) Data transaction income ETS

The initial distribution mechanism of ETS:

Table 4. ETS Allocative Desicion

Percentage	Program	Description
30%	Ecological construction	Used for ETS ecological construction, developer community cultivation, for paying cooperation costs, holding technical competitions, pushing the application and ecological construction of ETS technology, and demonstrating and enhancing business value.
10%	Early investors	Used to distribute to early angel investors.
15%	Private placement	Used to distribute to private investors.
21%	Team	Issued ETS to reward the founding team's significant manpower, resources, and material resources during the preparatory period and development, as well as the significant contribution to ETS in terms of technology and operations, while motivating them to continue to play an important role in the development of ETS.
14%	Foundation	used for day-to-day operations in team development, rewarding block nodes and marketing campaigns.
10%	University	As a support fund for academics and

Laboratory and Developer Support Fund	industry, it is used to support technical problems and participate in the development of ETS systems.

reference

[1] Ben-Sasson, Eli, et al. "Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture." USENIX Security Symposium. 2014.

[2] Parno, Bryan, et al. "Pinocchio: Nearly practical verifiable computation." Security and Privacy (SP), 2013 IEEE Symposium on. IEEE, 2013.

[3] Gennaro, Rosario, et al. "Quadratic span programs and succinct NIZKs without PCPs." Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2013.

[4] Craig Gentry. A Fully Homomorphic Encryption Scheme (Ph.D. thesis). Available at

http://crypto.stanford.edu/craig/, 2009.

[5] Van Dijk M, Gentry C, Halevi S, et al. Fully homomorphic encryption over the integers. Gilbert H. Advances in Cryptology–EUROCRYPT 2010. Berlin, Heidelberg: Springer, 2010. 24–43.

[6] Brakerski Z, Gentry C, Vaikuntanathan V. (Leveled) fully homomorphic encryption without bootstrapping. Proceedings of the 3rd

Innovations in Theoretical Computer Science Conference. Cambridge, MA, USA. 2012. 309–325.

[7] Gentry C, Sahai A, Waters B. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically- faster, attribute-based. Canetti R, Garay JA. Advances in Cryptology (CRYPTO 2013). Berlin, Heidelberg: Springer, 2013. 75–92.

[8] Regev O. On lattices, learning with errors, random linear codes, and cryptography. Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing. Baltimore, MD, USA. 2005. 84–93.

[9] MICCIANCIO D, PEIKERT C. Trapdoors for lattices: Simpler, tighter, faster, smaller[C]. In: Advances in Cryptology—EUROCRYPT 2012. Springer Berlin Heidelberg, 2012: 700–718.

[10] PURSLEY M, ROEFS H. Numerical evaluation of correlation parameters for optimal phases of bi- nary shift-register sequences[J]. IEEE Transactions on Communications, 1979, 27(10): 1597–1604. [DOI: 10.1109/TCOM.1979.1094293]

[11] WANG C H, ZHAO C, XU X G, et al. Distributed paralel computing environment: MPI[J]. Computer Science, 2003, 30(1): 25–26. [DOI: 10.3969/j.issn.1002-137X.2003.01.007].

王萃寒, 赵晨, 许小刚, 等. 分布式并行计算环境: MPI[J]. 计算机科学, 2003, 30(1): 25–26. [DOI: 10.3969/j.issn.1002- 137X.2003.01.007]

[12] FAGG G E, DONGARRA J J. FT-MPI: Fault tolerant MPI, supporting dynamic applications in a dynamic world[C]. In: Recent Advances in Parallel Virtual Machine and Message Passing Interface—EuroPVM/MPI 2000. Springer Berlin Heidelberg, 2000: 346–353. [DOI: 10.1007/3-540-45255-9_47]

[13] ZHANG Y, WEN J. The IoT electric business model: using blockchain technology for the Internet of things[J]. Peer-to-Peer Networking and Applications, 2017, 10(4): 983-994.

[14] CHAKRAVORTY A, WLODARCZYK T, RONG C. Privacy preserv- ing data analytics for smart homes[C]//Security and Privacy Work- shops (SPW), 2013 IEEE. IEEE, 2013: 23-27.

A new type of distributed
encryptioncomputing protocol