electronics

MDPI

Article

Trust Hardware Based Secured Privacy Preserving Computation System for Three-Dimensional Data

Munan Yuan [1,2,*], Xiaofeng Li [1,*], Xiru Li [1], Haibo Tan [1] and Jinlin Xu [1]

[1] Hefei Institute of Physical Science, Chinese Academy of Sciences, Hefei 230031, China; xrli@hfcas.ac.cn (X.L.); hbtan@hfcas.ac.cn (H.T.); jlxu@hfcas.ac.cn (J.X.)
[2] Science Island Branch, Graduate School of University of Science and Technology of China, Hefei 230026, China
* Correspondence: mnyuan@hfcas.ac.cn (M.Y.); xfli@hfcas.ac.cn (X.L.); Tel.: +86-551-6559-5684 (M.Y.); +86-551-6559-1219 (X.L.)

Abstract: Three-dimensional (3D) data are easily collected in an unconscious way and are sensitive to lead biological characteristics exposure. Privacy and ownership have become important disputed issues for the 3D data application field. In this paper, we design a privacy-preserving computation system (SPPCS) for sensitive data protection, based on distributed storage, trusted execution environment (TEE) and blockchain technology. The SPPCS separates a storage and analysis calculation from consensus to build a hierarchical computation architecture. Based on a similarity computation of graph structures, the SPPCS finds data requirement matching lists to avoid invalid transactions. With TEE technology, the SPPCS implements a dual hybrid isolation model to restrict access to raw data and obscure the connections among transaction parties. To validate confidential performance, we implement a prototype of SPPCS with Ethereum and Intel Software Guard Extensions (SGX). The evaluation results derived from test datasets show that (1) the enhanced security and increased time consumption (490 ms in this paper) of multiple SGX nodes need to be balanced; (2) for a single SGX node to enhance data security and preserve privacy, an increased time consumption of about 260 ms is acceptable; (3) the transaction relationship cannot be inferred from records on-chain. The proposed SPPCS implements data privacy and security protection with high performance.

Keywords: blockchain; dual hybrid isolation transaction; Ethereum; Intel SGX; privacy; 3D data; trusted execution environment

check for updates

Citation: Yuan, M.; Li, X.; Li, X.; Tan, H.; Xu, J. Trust Hardware Based Secured Privacy Preserving Computation System for Three-Dimensional Data. *Electronics* **2021**, *10*, 1546. https://doi.org/10.3390/electronics10131546

Academic Editor: Juan M. Corchado

Received: 16 May 2021
Accepted: 22 June 2021
Published: 25 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the rapid development of internet technology, three-dimensional (3D) geometry models of real-world objects have been employed to collect anthropometric data for many application scenarios. 3D model reconstruction has become an important business method for merchants to attract customers [1–5]. Massive 3D data emphasizes challenges around storage, management and computation. In traditional, the related services of 3D data are deployed in high-performance centralized servers [6,7]. However, the trusted third party with a large capacity and powerful computation inevitably inherits the single-point attack problems, which cannot preserve the security and ownership of stored data. Without the data owner's approval, such attacked data can be applied to proliferate huge commercial profits by unauthorized organizations [8]. Moreover, data owners have no awareness about how important their personal data is. For example, the refined 3D model reconstruction reveals personal information, including sex and appearance [9–15]. 3D sensitive data confirmation, security and privacy problems have attracted extensive attention. The cryptographic technique is considered as an effective scheme to relieve pressure from the demand of data privacy and security protection in traditional cloud scenes [16–24]. However, these methods have poor performance for centralized single point of failures and complex computation.

From the statement above, a distributed, security and privacy-preserving scheme should be designed for 3D sensitive data. A peer-to-peer distributed ledger technology blockchain has caused explosive development to solve the distrust problem in multi-parties [25–28]. Through a consensus mechanism, blockchain can ensure consistency, immutability and security of on-chain data. However, such technology does not set access barriers for any participating nodes, which inevitably brings privacy leakage problems. Combining the advantages of distributed blockchain and traditional privacy protection mechanisms, many privacy-preserving approaches based on blockchain have been proposed [29–31]. These proposals bring much performance loss in massive sensitive data computation, due to slow computing efficiency and expensive blockchain storage. Some blockchain-based methods with access control have been provided [32,33]. Assuming that data owners and users are honest, the blockchain framework with access control can realize data rights, data security and data confidentiality. However, it is impossible to avoid a data proprietary shift from a permitted user to a non-permitted user without the data owner's consent for authorized users to access the raw data.

As a fully isolated hardware-level approach, trusted execution environment (TEE) technology has been integrated into large-scale usage to ensure the confidentiality of inner applications. The TEE technology makes data computation details invisible to non-secure areas and provides reliable correct execution proof through an inner crypto-mechanism. Compared with homomorphic encryption and zero-knowledge proof, TEE technology achieves similar privacy protection effects but an efficient execution with hardware support. Therefore, the hybrid architecture of TEE and blockchain technology is an effective way to solve the data privacy protection problems, which has been covered by many studies [34–39]. Although these methods guarantee that the raw data cannot be directly accessed, personal privacy leakage arises from analyzing the on-chain transaction data. Some research revealed that the real identity of accounts in the blockchain can be inferred by analyzing transaction relationships among different accounts and off-chain auxiliary information [40–42]. Once an account in historical transaction records is linked to the real world, all transaction information related to the account will be revealed.

Inspired by the scheme of hybrid architecture of TEE and blockchain, we design a secured privacy-preserving computation system (SPPCS) based on trust hardware. To the best of our knowledge, the SPPCS system is capable of sensitive data and transaction relationship protection. The key to this achievement is the combination of reputation management, decentralized storage system, blockchain technology and TEE mechanism. There are many mature solutions for reputation feedback management, which are not in the main focus of SPPCS system research [43,44]. Reputation feedback management provided by [43] is used in this paper. Along with proof of computation correctness from TEE execution, the fundamental rights of data can be guaranteed.

Thus, the main contributions of this paper are as follows:

- Hierarchical separation computation: The SPPC system adopts a hierarchical architecture: computation off-chain, storage off-chain and consensus on-chain. TEE performs critical data computation and attests to correct and complicated execution off-chain. Distributed data storage, independent from consensus and computation, provides secure and large-capacity storage. On-chain nodes are responsible for maintaining a consistent global state and performing consensus updates efficiently.
- Data leverage scheme: Data expectation matching is realized by similarity computation of registered data text attributes with a graph structure. According to reputation value, the graph structure could be altered. Moreover, the registered costs should affect data provider ranking in a similarity computation list. In this way, the SPPCS voids network resource consumption caused by invalid transactions effectively.
- Dual hybrid isolation model: To obscure the connection between the sender and receiver of a transaction, we design a transaction isolation model based on the TEE leader scheme. The TEE leader completes the transactions with the sender and the receiver asynchronously to cover up the direct relationship, which can avoid privacy

exposure issues caused by individual identity disclosure. To prevent access the all the original data, we also construct another hybrid model based on non-accessible attributes of TEE. Data requesters obtain computation results instead of accessing original data to preserve data confidentiality.

The rest of this paper is organized as follows: we present the related work of privacy-preserving management in Section 2. Section 3 introduces the preliminaries throughout this paper. Section 4 details the architecture of SPPCS and specific implementation process. The experimental performance evaluation is conducted in Section 5. Section 6 analyzes the security and privacy of SPPCS proposed in this paper. Conclusions and discussions are presented in Section 7.

2. Related Work

2.1. Privacy with Principles

In the early stage when it was realized that privacy protection is necessary for 3D sensitive data, various rules and constraints imposed by humans have been proposed [45,46]. To alleviate the embarrassment from body data collected with 3D devices, Golden et al. [45] stipulated that 3D data acquisition and analysis should not be performed in the same space. Moreover, 3D data machines cannot store sensitive data to avoid privacy leakage problems. Mironenko et al. [46] described some countries that defined 3D data protection legislation in a formal way, including transparency, individual participation, purpose specification and limitations, minimization, data quality and integrity, security and accountability and auditing. However, these principles were performed through restricting people and should not be considered as a panacea in privacy protection.

2.2. Network Implementation in Privacy Protection

Methods were proposed to mitigate the invasion pressure of 3D sensitive data storage and transformation over a network. Zuo et al. [47] proposed a two-factor secret key protection mechanism combined with attribute-based fine-grained access control to protect the confidentiality of shared sensitive data. Madankar et al. [21] utilized a cryptography method to preserve the privacy of biometric data. Jung et al. [48] designed an accountability model, AccountTrade, to prevent dishonest behavior, such as reselling others' datasets. Bindahman et al. [9] discussed the privacy problems produced by 3D body scanning technology and described the key points of a privacy-preserving framework. Wang et al. [10] proposed a time machine-like method to relive experiences with virtual reality and protect individual privacy in virtual worlds. Sekhavat et al. [13] utilized augmented reality devices to watch physical wearing with the human alternative mode. Compared with standing in front real cameras, this method can protect human body privacy.

Some filters and deep learning methods have been proposed to protect sensitive data privacy. Without image restoration, Wang et al. [49] proposed a coded aperture camera system with lens-free and a training mechanism to recognize human action in a privacy-preserving way. Sarwar et al. [50] designed an adaptive space exploration to configure the privacy filter automatically, which reduces face resolution to preserve privacy. Chriskos et al. [51] utilized the singular decomposition method and proposed image projection on hypersphere methods to identify the individual rather than accessing the face data for identification. Yu et al. [52] developed a feature-based approach with AlexNet and designed an object-based approach with a deep learning classifier. Through configuring fine-grained settings, the approach could protect data privacy. Li et al. [53] proposed a deep convolutional neural network with k-anonymity, l-diverse and t-closeness to protect face privacy. However, training data become available to be exposed for malicious users who can access these data. Li et al. [12] designed a deep convolution neural network to remove sensitive 3D information and utilized a GAN-based generative adversarial network framework to synthesis models for parameter extraction. Xue et al. [20] presented a privacy-preserving method through which sensitive data can be stored and analyzed on any untrusted device through bloom filters, which could defend the privacy leakage from

original data at the storage level. Based on encrypted face features, encrypted machine learning parameters and additive secret sharing secret, Ma et al. [11] proposed a lightweight adaptive boosting face recognition method to protect facial data privacy.

Nevertheless, all the methods mentioned above are based on centralization, which neither solve the problem of privacy leakage caused by a single point of failure nor the data resale and repeated use without the knowledge of the data owner.

2.3. Peer-to-Peer Technology

Kim et al. [54] designed a decentralized data platform for 3D point clouds to share efficiently and securely through a Dat protocol and a discrete global grid system. However, this platform cannot ensure the confidentiality of data use. With peer-to-peer (P2P) technology, 3D sensitive data privacy protection research still needs to be explored in depth. However, there have been many research activities to solve the privacy of sensitive data problem with blockchain in other fields.

Wang et al. [55] proposed a fine-grained access control decentralized storage framework based on blockchain. The security and privacy of stored data are enhanced by a public key encryption mechanism. This general method is not fully suitable in 3D data privacy protection, due to the inevitableness of reselling and accessing original data.

In internet of things (IoT) data security field, Lu et al. [56] proposed a blockchain-based federated learning framework to maintain data privacy. However, the training sets could reveal some original data privacy, which is fatal to 3D sensitive data.

Howard et al. [29] proposed a decentralized computation platform, Enigma, to prevent accessing raw data, based on blockchain technology. Al-Zahrani et al. [8] used blockchain and Data as a Service (DaaS) to construct a privacy-preserving data sharing model, which preserved the data controllability for data owner. Thus, this model cannot guarantee the data traceability and quality estimation for the data user.

For electronic medical data privacy problems, Zheng et al. [57] introduced a wearable healthy data sharing system based on blockchain to enable users to easily control and securely share their sensitive data. Yaji et al. [58] presented MeDShare, a blockchain-based system to provide medical data provenance, auditing and access control on untrusted parties. MeDShare revoked offending entities rights to access data in case of violation activities detected, with minimal risk to privacy exposure. Al Omar et al. [59] proposed a blockchain-based privacy-preserving platform for healthcare data, which leveraged pseudonymity of cryptographic functions to protect sensitive data privacy. These solutions cannot avoid various problems caused by accessing raw data. In addition, when anonymity is no longer secure in the real world, it will lead to identity exposure and pose challenges for privacy protection.

Among digital currencies, Li et al. [30] provided a privacy data storage protocol to preserve identity privacy in blockchain applications, based on the complete anonymity of the ring signature. BlockMaze, proposed by Guan et al. [31], an account-model blockchain that hided the linkage between transaction amounts and the sender–recipient. Along with dual balance model implemented by zk-SNARKs, BlockMaze has strong privacy guarantees. Although these cryptography functions provide users with greater privacy, they are not suitable for massive 3D data with overhead computation.

The idea of combining blockchain and TEE has been focused on the privacy preservation in data computation. Cheng et al. [38] proposed Ekiden, a system combined blockchain with TEE technology to separate the consensus from heavy computation and preserve data confidentiality. Similarly, Zhang et al. [35] leveraged blockchain as information interaction medium on-chain and utilized TEE off-chain computation to design Cerberus, which can guarantee privacy and efficiency of data procession. Dai et al. [34] designed SDTE, a blockchain-based data trading ecosystem to prevent dishonest activities in trading platforms. In SDTE, only analysis results can be obtained and the raw data cannot be accessed. Based on the blockchain and TEE, Wang et al. [36] proposed Hybridchain to improve execution efficiency through decoupling smart contract computation from consensus.

A secure key-value memory system has been designed in Hybridchain to mitigate TEE memory limitations. Maddali et al. [37] presented VerBlock, a blockchain-based framework combined with TEE technology, to reduce endorsements in enterprise blockchains. Zhang et al. [39] presented an electronic voting system to protect the privacy of voters and votes based on Ethereum, smart contracts and the TEE environment. These proposals did not notice the identity leakage of the IP address associated with the owner/requester of data, whereas the pitfall should be considered to design 3D privacy protection schemes.

Although blockchain has excellent performance in data security, data auditing and sensitive data privacy protection, some immature defects, such as low scalability, double-spending attack and quantum algorithms in breaking cryptographic cracking, limit the wide application of this technology.

3. Preliminaries

In this section, we briefly discuss the relevant background knowledge throughout this paper.

3.1. Blockchain

As a tamper-resistant distributed ledger technology, the blockchain is linked together through cryptographic hashes as blocks and chains. The block on-chain record batches of valid transactions and processes with an iterative hash value confirmation of the previous block to original genesis block, which defends data modification effectively. In the blockchain system, all members can trust each other in a non-secure environment. The blockchain technology has been applied in decentralized cryptocurrency successfully, such as Bitcoin, Ethereum, Zcash, etc. Moreover, the development of smart contracts over the blockchain, especially Ethereum, contributes to performing trust governance in medical, identity management, asset management, and supply chain fields, etc.

The smart contact is written by Turing-complete programming languages and executed transparently in accordance with the code written on all block nodes. Ethereum supports a stack-based Ethereum Virtual Machine (EVM) to execute smart contacts. A smart contract should be considered as a contact account, which can be executed similarly with users' send transactions to the normal account. The GAS is introduced to limit the transaction overhead and avoid the infinite loop of smart contract execution.

Asymmetric cryptography, as the cornerstone of blockchain, contains a key pair public/private key. The data are encrypted with a public key and decrypted using a private key. The data signature is signed with a private key and validated by the corresponding public key. The public key is usually shared publicly, whereas the private key is only known to the owner. It is impossible for anyone to infer the private key from the public key and vice versa, which is guaranteed by the security of the encryption algorithm.

3.2. IPFS

An Inter Planetary File System (IPFS), as a decentralized storage structure, is a content-addressed distributed network that stores large amount of data easily without any central coordinators [60]. The file stored in IPFS is recognized with a unique cryptographic hash, which can be identified by version-control history to remove duplication. IPFS facilitates concurrent access to stored data with a high-throughput performance.

3.3. Trust Hardware with SGX

The trust hardware relies on chip isolation to create a trusted execution environment (TEE) to prevent unauthorized access from tampering with or learning the inside state. Software Guard Extensions (SGX) is integrated into CPU architecture designed by Intel and has to be the most prominent TEE technology [61]. SGX preserves the confidentiality, security and integrity of sensitive data operation by creating a secure container enclave. The interfaces of enclaves could be defined at the initialization phase, after which the internal content will not be modified. The access control of SGX is shown in Figure 1.

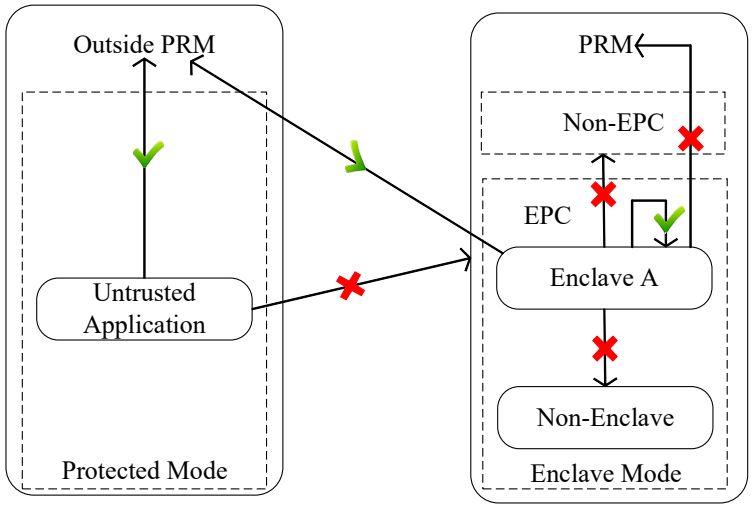


Figure 1. The access control of SGX.

SGX provides local attestation and remote attestation to help remote entities verify whether the particular code is executed inside the enclave securely and correctly. Local attestation is used by enclave A to prove its identity to enclave B directly, with the same platform key. The enclave uses remote attestation to provide proof to different platform enclaves. At first, the quota enclave encrypts its contents with a hardware-protected enhanced private ID (EPID) key to construct the QUOTE structure, which is attested by remote enclaves on other platforms. Then, the remote enclave verifies the validity of the quote through IAS, which only permits the legitimate entity to issue remote attestation requests. The content received from the remote enclave can be decrypted with a group public key. After checking the validation hash value which summarizes the code and initial data, the validity results will be sent back to the original enclave.

The enclave in SGX uses sealing instructions to store sensitive data outside or read sealed data into secure memory, which greatly reduces the overhead of secure memory. Two sealing strategies have been supported in SGX: sealing data with a platform key permit other enclaves to access data on the same platform; sealing data use a special key that prevents sharing data with other enclaves on the same platform.

3.4. Graph Sturcture

The semantic text clustering methods based on graph structure has been researched and developed in depth for decades [56,62–64]. In the graph structure, the weight-graph is considered as the most common way of representing a correlation in text content. A weight context graph $G = \{N, E, V\}$ consists of N, the set of nodes ($\{n_1, n_2, \ldots, n_m\}$), E, the set of edges ($E \subseteq N * N$, $\{e_{12}, e_{13}, \ldots e_{ij}, \ldots, e_{mn}\}$), and V, the weight adjacency matrix $\left\{v_{e_{12}}, v_{e_{13}}, \ldots v_{e_{ij}}, \ldots, v_{e_{mn}}\right\}$. The weight $V_{e_{ij}}$ of the edge denotes the correlation degree between the nodes n_i and n_j. If $G' = \{N', E', V'\}$ is the sub-graph of G, denoted as $G' \subseteq G$, G' meets $N' \subseteq N$ and $E' \subseteq E \cap (N \times N')$. For graph structure G_1, G_2 and G_*, if $G_* \subseteq G_1$, $G_* \subseteq G_2$, and no any other graph structure $|G_{**}| > |G_*|$, the G_* should be considered as the maximum common sub-graph between G_1 and G_2. The maximum common sub-graph contains information of nodes and edges, which can reflect the similarity in different graphs to cluster context categories into different groups.

4. The Proposed SPPCS Overview

In this section, we detail the scheme design of SPPCS. As described in Figure 2, the SPPCS contains five modules: Data Supplier (DP), Data Requester (DR), Ethereum Blockchain (EB), Distributed Storage (DS), and Trusted Computation Pool (TCP). The interaction operations of these modules include registering, requirement matching computation, hybrid

transaction execution, data analysis and evaluation feedback in three phases. Tables 1 and 2 summarize the abbreviations and role notations used in the paper.

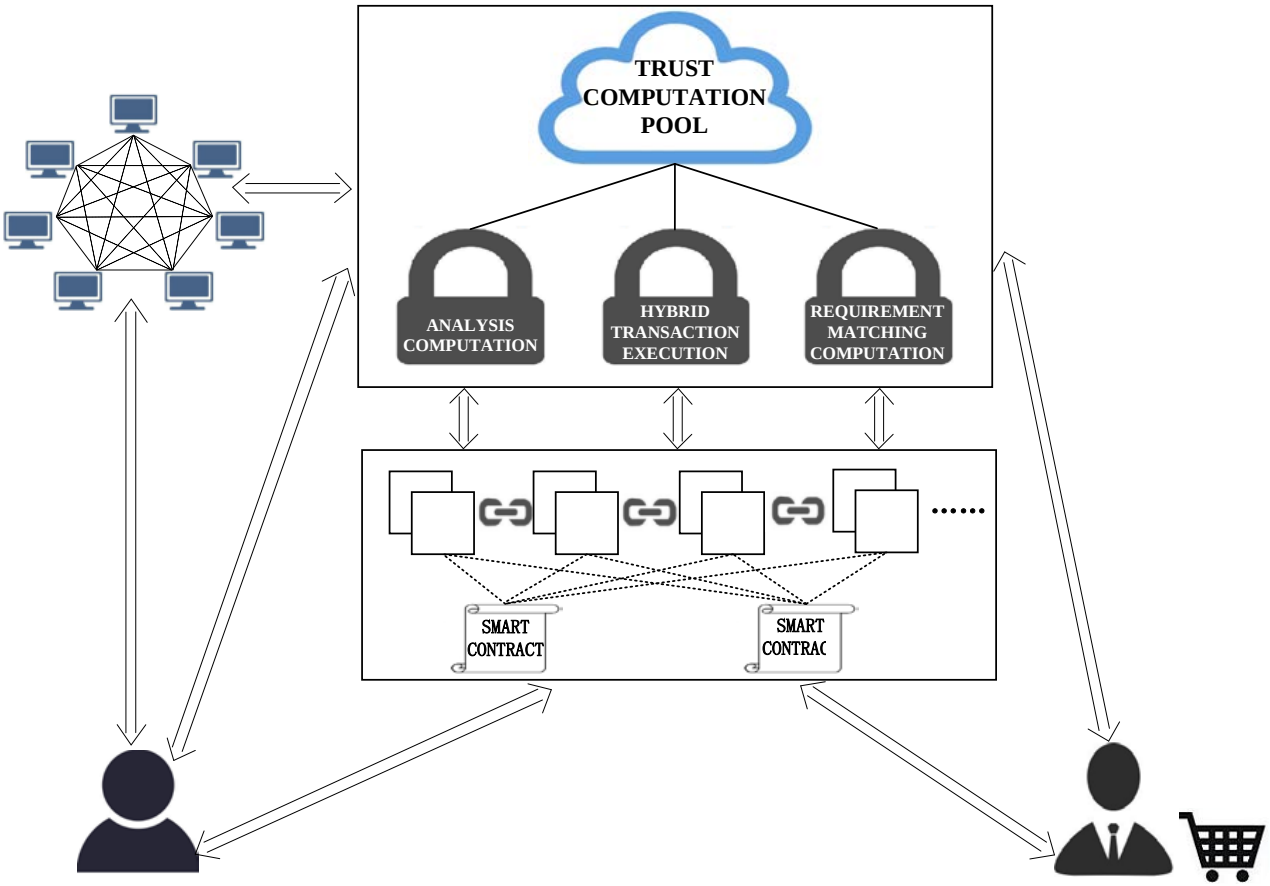


Figure 2. The architecture of SPPCS.

Table 1. Abbreviations.

Notation	Description
3D	the three-dimension
TEE	the trusted execution environment
SPPCS	the secured privacy-preserving computation system
SGX	the Software Guard Extensions
P2P	the peer to peer
IPFS	Inter Planetary File System, a decentralized storage structure
IAS	the Intel Attestation Server
PRM	the preserved random memory
EPC	the enclave page cache
EPID	the enhanced private ID
DP	the identity provides some attributes data
DR	the entity requires some attributes data
EB	Ethereum blockchain
DS	distributed storage
TCP	trusted computation pool consists of trusted nodes
TN	trusted node with TEE
CN	consensus node
ECC	the Ellipse Curve Cryptography

Table 1. *Cont.*

Notation	Description
CTNP	complex trusted node pool
USN	unique signature notion
SNN	the serial number notation
dp_word	the keyword designated by DP
Num_TN_{req}	the required number of TN
ori_data	the DP original data
loc_index	the DP original data storage location index
$Code_est$	the quality evaluation code of required data
$Code_comp$	the DR computation code
$Data_{pre_est}$	the data matching value pre-estimated from DP
prc	the DP register contract
$rdrc$	the DR delegation register contract
nrc	the TN register contract
$pimc$	the DP information management contract
$timc$	the TN information management contract
T_{attri}	the DP data attribute, such as data type, data amount, unit price . . .
T_{req}	the DR requested data, such as data description, expected size, bid price . . .
$tran_num$	the TN leader successful transaction number
$Ptype_{TN}$	the TN type and TN leader must be complex value
List_TN	the list produced by most selected TNs
Serialized_List_TN	the serialized List_TN
$item_num$	the number of items in selected transaction list

Table 2. Role notation summary.

Notation	Description
Pk_{id_role}	the public key for *id_role*, which can be one of DP, DR, TN, TN enclave and CN
Sk_{id_role}	the private key for *id_role*, which can be one of DP, DR, TN, TN enclave and CN
Rep_{id_role}	the reputation, *id_role* can be one of DP, DR and TN
$EPrice_{id_role}$	the expected price for *id_role*, which can be DP, DR and TN
K_{id_role}	the symmetric key used for *id_role*, which can be original DP data and DR computation code
$RMResult_{id_role}$	the requirement matching analysis result for *id_role*, which can be normal result list and serialized result list
$Store_{id_role}$	the storage methods of *id_role*, which can be seal and unseal

4.1. System Framework

The corresponding description of each entity in Figure 2 is shown as follows:

- DP: The unity owns sensitive data. In our system, DP can register data attributes, individual reputation value and expected unit price with the smart contracts.
- DR: The users who require data from DP for their purposes perform three tasks in TCP to preserve data privacy. (1) After DRs broadcast demands on-chain, data matching analysis requirement should be delegated to TCP off-chain. (2) DRs obtain requirement matching list results from TCP and the results verification will be published on-chain. (3) DRs conduct hybrid transaction with DPs through TCP on-chain.
- EB: Consensus nodes of EB are responsible for maintaining the consistency of data transaction records on-chain and validating state updates with the TEE attestation. EB in SPPCS also acts as an interaction medium among DP, DR and TCP.
- DS: IPFS, as a typical representative of DS, provides a safe and large-capacity storage. SPPCS utilizes IPFS to separate storage from consensus and computation. The IPFS receives encrypted data and returns location indexes.

- TCP: Trusted computation nodes, with SGX-enabled CPU and bios, should be considered as the core of TCP. These trusted nodes conduct delegated requirement matching computations, hybrid transactions and data analysis computations with REE and TEE. REE supports interaction interfaces between on-chain and off-chain. TEE guarantees confidentiality in the data process with SGX remote attestation, SGX seal and unseal and cryptographic methods. The delegation request is sent to TCP and the trusted computation node queries related information and executes the corresponding analysis. The trusted computation node securely stores the results outside the enclave with SGX seal, otherwise the reads seal data into the enclave with SGX unseal. Simultaneously, the trusted computation node sends execution results to EB for verification by SGX remote attestation and cryptographic methods. A quorum of trusted computation nodes run a hybrid transaction protocol to complete transactions and obscure the relationship between DR and DP.

4.2. Data Leverage Scheme

DR can verify whether the potential data of DP meets expectations, through delegating a data requirement computation to trusted nodes of TCP. Inspired by [63], we convert data matching analysis to compute the similarity graph structures. We denote the attribute item of required data as text $T = \{t_1, t_2, \ldots, t_n\}$ and the data attribute of DP nodes registered on-chain as a set of text items $N = \{n_1, n_2, \ldots, n_m\}$, where t_n represents the n index keyword to describe required data and n_m represents the m index data attribute expression of a DP node. The value of n_m is defined as the product of occurrence frequency in T and DP node reputation, calculated as Equation (1):

$$Value(n_m) = \frac{Num(n_m)}{T_{num}} * NR(n_m), \tag{1}$$

where $Num(n_m)$ is the number of n_m occurrences times, T_{num} is the number terms of T, $NR(n_m)$ is the normalized reputation of n_m.

The semantic relationship between two nodes of N is represented as $E = \{e_{12}, e_{13}, \ldots e_{ij}, \ldots, e_{mn}\}$, where e_{ij} is the edge of n_i and n_j. With the weight value of edges being represented as $V = \left\{v_{e_{12}}, v_{e_{13}}, \ldots, v_{e_{ij}}\right\}$, the calculation formula is as Equation (2):

$$\begin{cases} v_{e_{ij}} = \frac{Value\left(n_{i+j}\right)}{Value(n_i) + Value\left(n_j\right) - Value\left(n_{i+j}\right)}, & if\ i \neq j \\ v_{e_{ij}} = 1, & if\ i = j \end{cases}, \tag{2}$$

where $Value\left(n_{i+j}\right)$ is the co-occurrence of n_i and n_j in T.

All DP node attribute descriptions into are represented in the weighted graph $G = \{G_1, G_2, \ldots, G_n\}$, of which the maximum common sub-graph sets can be constructed. For G_1 and G_2, the maximum common sub-graph SG_{12} consists of all co-occurrence nodes and their similarity nodes, where $v_{e_{ij}}$ exceeds the threshold weight value. The similarity of G_1 and G_2 can be calculated as Equation (3):

$$Similarity(G_1, G_2) = \frac{Num(SG_{12})}{Num_{max}(G_1,\ G_2)}, \tag{3}$$

where $Num(SG_{12})$ is the number of co-occurrence nodes between G_1 and G_2, $Num_{max}(G_1,\ G_2)$ is the max number of nodes in G_1 and G_2.

We extend Equation (3) to obtain all nodes' similarities in G, denoted as $SG = \left\{SG_{12}, SG_{13}, \ldots, SG_{ij}, \ldots, SG_{mn}\right\}$, and leverage the k-means algorithm to cluster DP nodes into different attribute sets. The DR data matching list is composed of DPs above the similarity threshold value.

4.3. Workflow of SPPCS

The workflow of SPPCS is divided into three phases: register phase, requirement matching computation phase, and hybrid transaction execution phase. In this paper, we do not take pitfalls arising from TEE into consideration, since many mitigations have been proposed to protect the security and availability of TEE [65,66]. For simplicity, we implement the credible reputation scheme in [43] and assign tasks to selected trusted nodes.

1. Register Phase

In SPPCS, the DP, DR and trusted node (TN) should register entity information on EB with a smart contract, respectively. DR data requirement, DP data attribute, DP reputation, DP data quality estimation, TN reputation, TN expected prices and TN type should have been recorded on-chain. The main activities of this phase are initialize attestation, leader selection, attribute and requirement description, data storage and pre-estimation and registration. The registration steps are detailed in Figure 3.

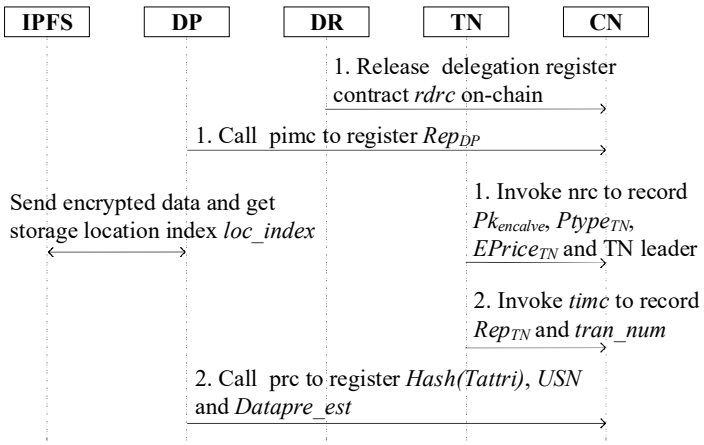


Figure 3. The register phase of SPPCS.

(a) Initialize attestation: The Ellipse Curve Cryptography (ECC) key pair of the DP, DR, TN, TN enclave and consensus node (CN) are denoted as $< Pk_{id_role}, Sk_{id_role} >$, where Pk is the public key, Sk is the private key and id_role is one of DP, DR, TN, TN enclave and CN. The TN initializes reputation Rep_{TN}, node type $Ptype_{TN}$, enclave public key $PK_{Enclave}$ and expected price $EPrice_{TN}$. Similarly, the DP initializes reputation Rep_{DP} and estimates the quality of own data $Data_{pre_est}$.

(b) Leader selection: $Ptype_{TN}$ can be set that is normal or complex, with a normal default value. Different $Ptype_{TN}$ TNs have different $EPrice_{TN}$ ranges and only complex $Ptype_{TN}$ TNs can perform hybrid transactions and run a leader election protocol. Complex $Ptype_{TN}$ TNs, with much higher upper limit $EPrice_{TN}$ values, consist of a complex trusted node pool (CTNP), where the winner in each round of elections updates the leader management smart contract. The term of the leader node is determined by reputation and the successful transactions number. The leader reputation is influenced by DR comments and the successful transaction rate. The number of successful transactions increased by one with a smart contract triggered by the DR validation. The smart contract records the reason for leader re-election, which takes place in these scenarios: ① the number of successful transactions reaches a set value; ② the leader node reputation value is lower than the set value. When DPs notice the leader node is inconsistent with the DR transaction period, DPs can continue transaction for reason ① and reject it for reason ②.

(c) Attribute and requirement description: DPs extract data characteristics into text-item sets T_{attri}, such as data type, data amount, unit price, etc. Moreover, DP creates a unique signature notion (USN), defined as DP signature of the keyword hash value $Sig_{Sk_{DP}}(Hash(dp_word))$, where the keyword is designated by the DP. DR converts

the demand into keyword sets T_{req} including required data description, expected size, bid price, required number of TN Num_TN_{req} and unit delegation fee.

(d) Data storage and pre-estimation: DP generates a one-time symmetric key K_{store} to encrypt original data $Enc_{K_{store}}(ori_data)$ in each round of transactions. $Enc_{K_{store}}(ori_data)$ should be uploaded to IPFS and DP obtains the corresponding location index loc_index. DR publishes the quality evaluation code $Code_est$ of required data, based on which all DPs pre-estimate the data matching value $Data_{pre_est}$.

(e) Registration: DR releases the delegation register contract *rdrc* designed in next Section 4.4 to broadcast T_{req}, $Hash(Code_est)$ and $Code_est_{loc_index}$. DP and TN complete registration by using the corresponding smart contracts. DP registers the contract *prc* records T_{attri} hash value $Hash(T_{attri})$, USN and $Data_{pre_est}$. DP information management contract *pimc* mainly contains Rep_{DP} on-chain. TN register contract *nrc* includes $Pk_{enclave}$, $Ptype_{TN}$, $EPrice_{TN}$ and TN leader. Rep_{TN} and the transaction number *tran_num* are registered in TN information management contract *timc*.

2. Requirement Matching Phase

As the driver of SPPCS, data requirement contracts of DR are broadcast on-chain. Similar to SDTE [34], appropriate TNs need to be selected to conduct the requirement matching analysis. The main contents of this stage, shown in Figure 4, are as follows:

(a) Delegation protocol: Corresponding to step 1 to step 4 in Figure 4, the DR broadcast requirement via blockchain starts the process of requirement matching analysis. The TNs first notice the DR demand contract *rdrc* and establish secure channels to reply to DR with a TN signature $Sig_{Sk_{TN}}(Hash(Rep_{TN}, EPrice_{TN}), Enc_{Pk_{DR}}(Rep_{TN}, EPrice_{TN}))$ pre-emptively. When DR receives a response node, it conducts an SGX remote attestation to validate TEE platform. Then, the DR verifies the signature of TN $Sig_{Sk_{TN}}(Hash(Rep_{TN}, EPrice_{TN}), Enc_{Pk_{DR}}(Rep_{TN}, EPrice_{TN}))$ with Pk_{TN} and decrypts $Enc_{Pk_{DR}}(Rep_{TN}, EPrice_{TN})$ with Sk_{DP}. Only if $Hash(Rep_{TN}, EPrice_{TN})$ equals $Hash(Dec_{Pk_{DR}}(Enc_{Pk_{DR}}(Rep_{TN}, EPrice_{TN}))$ can the node be a candidate. According to a combination strategy of higher Rep_{TN} and lower $EPrice_{TN}$ with the shortest response time, the DR publishes Num_TN_{req} candidate nodes on-chain and deposits double either of the Num_TN_{req} TNs expected.

(b) Requirement analysis off-chain: According to the Num_TN_{req} TNs selected by DR, these TNs first run an information synchronization protocol, such as Gossip, to maintain consistency of computation results. Step 5 in Figure 4 presents the potential DP and establishes a secure channel with any one of the selected TNs recorded on-chain to send T_{attri} encrypted with $Pk_{enclave}$, denoted as $Sig_{Sk_{DP}}(Enc_{Pk_{enclave}}(T_{attri}))$. The selected TNs verify the identity of DP with Pk_{DP} and then $Dec_{Sk_{enclave}}(Enc_{Pk_{enclave}}(T_{attri}))$ to compare the consistency between $Hash(Dec_{Sk_{TN}}(Enc_{Pk_{encalve}}(T_{attri})))$ and $Hash(T_{attri})$ in *prc*. In particular, $Dec_{Sk_{enclave}}$ ensure T_{attri} is validated and executed in the TEE environment. If the previous two hashes have same value, TNs share validated T_{attri} in ciphertext and then execute further computations in enclaves with the graph structure $Graph\left(N_{attri_{text}}, E_{attri}, V_{weight_attri_{text}}\right)$. It is assumed that the requirement matching analysis result list and corresponding hash value are denoted as $RMResult_{List_TN}$ and $Hash(RMResult_{List_TN})$. The $RMResult_{List_TN}$ contains Pk_{DP}, T_{attri}, $EPrice_{TN}$ and Rep_{TN}. $Hash(RMResult_{List_TN})$ published on-chain can not only protect the privacy of the requirement matching analysis result, but also makes it tamper-proof and provide verification. The serial number notation (SNN) is utilized to anonymize Pk_{DP} and is regarded as the representative item of the serialized result list $RMResult_{serialized_List_TN}$ corresponding line. The $RMResult_{serialized_List_TN}$ contains SNN, T_{attri}, $EPrice_{TN}$ and Rep_{TN} and the hash value, which can be represented as $Hash(RMResult_{serialized_List_TN})$. $RMResult_{List_TN}$, $Hash(RMResult_{List_TN})$, $RMResult_{serialized_List_TN}$ and $Hash(RMResult_{serialized_List_TN})$ are encrypted with $Pk_{enclave}$ and stored outside with the seal method. As described in step 6 in Figure 4, TNs send $Hash(RMResult_{List_TN})$ and $Hash(RMResult_{serialized_List_TN})$ on-chain.

Then, TNs sign encrypted final results with Sk_{TN} and send $Sig_{Sk_{TN}}(Enc_{Pk_{DR}}(< Hash(RMResult_{List_TN}), RMResult_{serialized_List_TN} >))$ to DR through the OCALL method in step 7 of Figure 4. DRs are unable to obtain the information of DP from $RMResult_{serialized_List_TN}$.

(c) Execution result validation: Corresponding to step 6 in Figure 4, the smart contract recognizes the final computation result with Num_max of $< Hash(RMResult_{List_TN}), Hash(RMResult_{serialized_List_TN}) >$, where Num_max is the maximum occurrence number and is more than $Num_TN_{req}/2$. When DR receives a response node, it conducts SGX remote attestation to validate the TEE platform, as shown in step 8 of Figure 4. Then, the DR utilizes Pk_{TN} to check whether $Sig_{Sk_{TN}}(Enc_{Pk_{DR}}(< Hash(RMResult_{List_TN}), RMResult_{serialized_List_TN} >))$ is from published TNs and validates the consistency between received $Hash(RMResult_{List_TN})$ and $Hash(RMResult_{List_TN})$ on-chain to ensure no tampering. In step 9 of Figure 4, the DR also considers the maximum occurrence number of $< Hash(RMResult_{List_TN}), RMResult_{serialized_List_TN} >$ as the validation result. When the final computation result is recognized by the smart contract as the same with the DR validation result, the smart contract is triggered to provide TNs with the correct computation and return extra change for DR with the reputation feedback received.

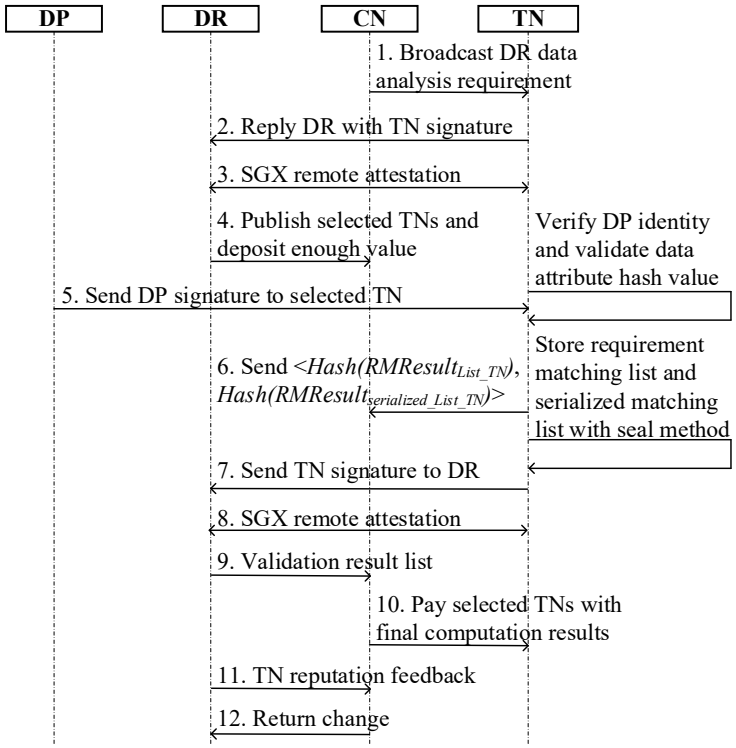


Figure 4. The requirement matching phase of SPPCS.

3. Hybrid Transaction Phase

In this phase, the essence characteristic can be presented as a proxy execution and prepayment computation, as described in Figure 5. In step 1, DR sends a selected transaction SNN, USN and deposits double the ether of the (DPs + TN leader) expected value via smart contract, according to $RMResult_{serialized_List_TN}$ and CTNP leader state registered on-chain. Moreover, DR places hash of computation code $Hash(code_comp)$, encrypted code location $Enc_{Pk_{enclave}}(code_comp_{loc_index})$, and encrypted symmetric key $Enc_{Pk_{enclave}}\left(K_{code_comp}\right)$ on-chain with a computation contract. Leader REE retrieves information on-chain and utilizes ECALL to send each item, SNN, USN, $Hash(code_comp)$, $Enc_{Pk_{enclave}}(code_comp_{loc_index})$

and $Enc_{Pk_{enclave}}\left(K_{code_{comp}}\right)$ into TEE, where the items are packed into an unvalidated transaction list. Step 2 and step 3 present corresponding DPs, which verifies the TN leader from TN register contract and creates a session key to transmit transaction identity proof $Enc_{Pk_{enclave}}(Sig_{Sk_{DP}}(dp_word))$ to the leader node. Then, in step 4, the TN leader TEE verifies hash of dp_word with USN and the validated DP is noticed by the leader REE via the OCALL method. Step 5, the TN leader REE sends $Enc_{Pk_{DP}}(SNN, USN)$ and deposits a double DP expected price on-chain with a smart contract. In step 6, DP sends $Enc_{Pk_{enclave}}(loc_index)$, $Enc_{Pk_{enclave}}(K_{store})$, $Hash(loc_index)$ and $Hash(ori_data)$ on-chain for TN leader verification. Simultaneously, the TN computation contract pays a remuneration for DP. Then, the TN leader selects appropriate DP data for the DR code following these rules: ① One-time execution: the TN leader loads all DP data to conduct a computation and obtain all rewards at once, with all selected DPs responding. ② Multiple execution: According to the number of items in the selected transaction list, *item_num* is even or odd, the TN leader could load responding DP data in batches to process with the DR code. If *item_num* is even, the TN leader should select an appropriate even number of responding DP data each time, where the appropriate even number is less than *item_num*. If *item_num* is odd, the TN leader first selects a responding DP data number greater than 1 (3, 5, 7 ...). These principles break the one-to-one transaction relationship, which protects the correlation privacy of transaction parties. From step 7 to step 9, the TN leader pays after the encrypted computation results and the corresponding hash value is published on-chain. The change scheme makes the TN leader give feedback on DP and DR to evaluate the TN leader to obtain an extra deposit.

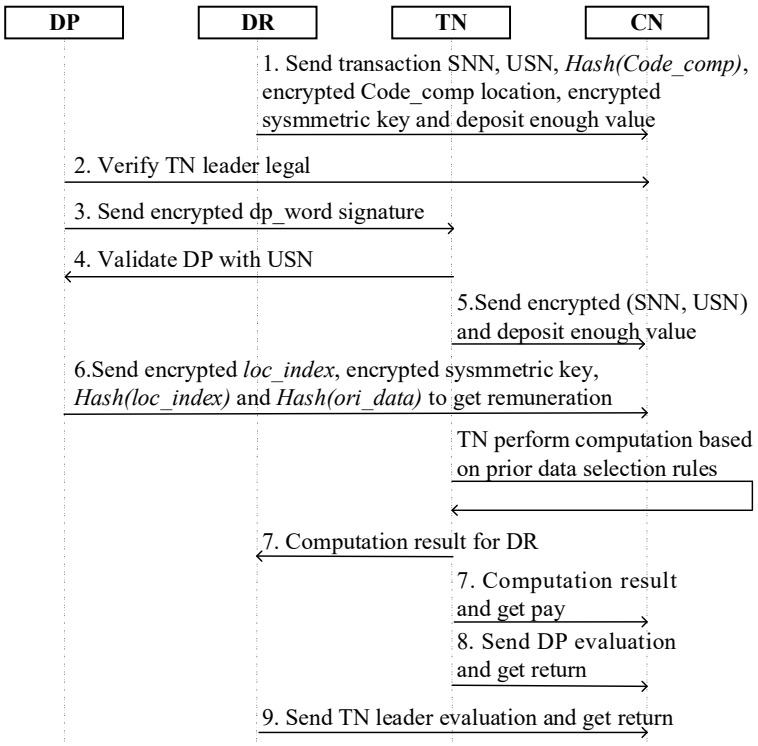


Figure 5. The hybrid transaction phase of SPPCS.

4.4. Smart Contract and Algorithm Design

In this section, we introduce the smart contract related interfaces and algorithm logic implemented in this paper. The SPPCS contains seven smart contracts—namely, the DR delegation register contract, DR computation contract, TN information management contract, TN register contract, TN computation contract, DP information management contract and DP register contract. Then, as described in Figure 6, we detail the function and composition of these smart contracts.

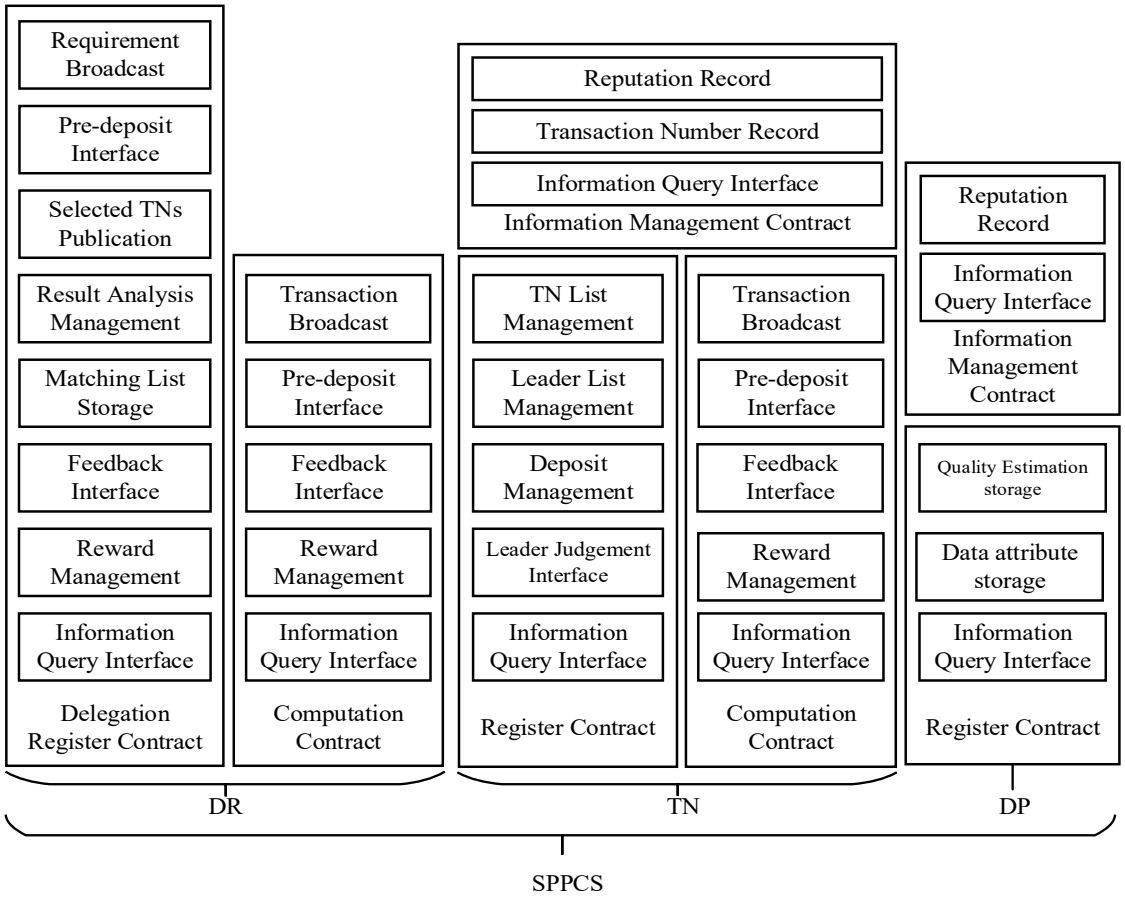


Figure 6. The smart contract of SPPCS.

The DR delegation register contract broadcasts data demands EB, which makes DP and TNs notice data requirement. The pre-deposit interface enables DR deposits to be double the expected ether to pay for selected TNs after data matching analysis. Result analysis management counts the analysis result and sets the highest occurrence result list as the final result. DR registers the contract and publishes selected TN information, stores the data matching list hash value and provides a feedback interface to make reputation comments on the selected TNs. Reward management is used to send appropriate ether to selected TNs' addresses and return charge to encourage DR to give reputation feedback. The DR delegation register contract also provides an information query interface for DR and TN to retrieve related transaction information.

The DR computation contract broadcasts the selected transaction list on-chain and deposits enough value through the pre-deposit interface. The feedback interface is provided to make reputation comments on the TN leader. The DR computation contract utilizes reward management to pay for the leader and returns change to encourage DR to give reputation feedback. The information query interface is used to check related transaction information.

The TN information management contract stores TNs' reputation values and leader successful transaction numbers, which are called a DR computation contract and new TN or new leader initialize phase in the TN register contract. The information query interface in this smart contract helps the prospective DP and DR select appropriate TNs according to the contract information.

The TN register contract records TN and leader TN specifications by TN list management and leader list management. The leader TN utilizes the deposit interface to deposit some ether, which is a penalty for a dishonest leader. Leader judgment management counts

the negative vote number from TN list and call TN information management contract to clear the reputation value when the count number is greater than two-thirds of TNs.

The TN computation contract will not be detailed for the similar composition and function with the DR computation contract.

The DP information management contract maintains DPs reputation, which is called a TN computation contract and new DP initialize phase in the DP register contract. The information query interface makes TN recommend and select an appropriate DP to execute the transaction.

The DP register contract mainly records data quality pre-estimation value and stores data attribute descriptions. The information query interface is used to check DP data information for verification.

Off-chain computation involves a DR data matching process and results analysis provided by DR. We list the pseudo-code of data matching analysis off-chain computation in Algorithm 1.

Algorithm 1 Data Matching Analysis Algorithm

input: encrypted data attribute item, encrypted keyword, DP register contract id and DR delegation register contract id
output: encrypted data matching list with a correctness proof by TEE
begin
  get selected TN list from blockchain and load into TEE
  get data requirement demands from blockchain and load into TEE
  if *timestamp* < *set-time* **then**
    get *data attribute hash* from blockchain and load into TEE
    parse T_{req} as $G1\left(N_{attri_{text}}, E_{attri}, V_{weight_attri_{text}}\right)$
    validate $Hash(Dec_{Sk_{enclave}}(Enc_{Pk_{enclave}}(T_{attri})))$ with *data attribute hash*
    if *data attribute hash is invalidated* **then**
      abort
    else
      if *validated* T_{attri} *is not new* **then**
        wait another T_{attri}
      else
        synchronization encrypted DP data with selected TNs
        parse T_{attri} as $G2\left(N_{attri_{text}}, E_{attri}, V_{weight_attri_{text}}\right)$
        perform $Similarity(G_1, G_2)$ calculation and obtain
        $SG = \left\{SG_{12}, SG_{13}, \ldots, SG_{ij}, \ldots, SG_{mn}\right\}$
  else
    cluster different attribute sets with *k*-means
    send encrypted $< Hash(RMResult_{List_TN}),\ RMResult_{serialized_List_TN} >$ to DR
    send $< Hash(RMResult_{List_TN}),\ Hash(RMResult_{serialized_List_TN}) >$ on-chain
  if *contract verify result success* **then**
    receive payment and reputation comments from blockchain
  else
    abort
end

5. Implementation and Performance Evaluation

We implemented and tested the SPPCS prototype based on the Ethereum private chain for sensitive data management. In the SPPCS scheme, the main activities are conducted by DRs, DPs and TNs. The DR and DP nodes were run in ThinkPad X1 carbon notebook with I7-10710 CPU and 16 G memory. The TNs and TN leader candidates were executed in 32 G memory Dell 7080-MT that provides SGX support with I7-10700 CPU. The smart contracts were deployed on a test EB private chain and encoded with solidity language. The enclave

computation code was written in C/C++. We used ECC asymmetric encryption, AES symmetric encryption and a SHA256 hash algorithm to protect interactive data among DRs, DPs and TNs outside of TEE.

In order to validate and evaluate the performance of SPPCS, we simulated the entire operating mechanism, which should include: (1) data encryption; (2) security attestation; (3) execution privacy; (4) transaction efficiency. Transaction efficiency depends on the consensus performance of Ethereum, which is open to all. The performance of data encryption and security attestation are easily tested by encryption technology and TEE hardware. However, in SPPCS performance section, we mainly focus on execution privacy evaluation. We created twelve enclaves to represent TNs on two machines, with three enclaves from different SGX are TN leader candidates. As in the data description listed in Tables 3 and 4, two DR (labeled as DR1 and DR2) nodes were set to publish demands and ten DP (labeled as DP1, DP2, . . . , DP10) nodes were assigned to supply data.

Table 3. The DR data demand information.

Node Account	Data Demand Description		Delegation Contract Address
DR1:0xDF353A4B5B0FB8F29B33185FF63CE4986F1B6C0F	input:	two-foot cloud data	0x7CE6AF1006F0474FBA430E4B6E9ACE82ACA584AB
	output:	foot length	
	expected unit size:	less than 6 MB	
	expected resolution:	1 mm depth accuracy	
	expected data acquisition:	Intel RealSense	
	bid price:	1.87 Milliether	
	required TN number:	3	
	Delegation fee:	0.37 Milliether	
DR2:0x24FFA197D6D00203 2343C4CE4FA3980DB167C398	input:	face cloud data	0x7CE6AF1006F0474FBA4 30E4B6E9ACE82ACA584AB
	output:	the distance between eyes	
	expected unit size:	less than 10 MB	
	expected resolution:	1 mm depth accuracy	
	expected data acquisition:	Intel RealSense	
	bid price:	0.75 Milliehter	
	required TN number:	5	
	Delegation fee:	0.1 Milliether	

Table 4. The DP data attribute information.

Node Account	Data Attribute Description		Attribute Hash
DP1:0X7ECF438AED8B220900F9381FC3D317F26B176684	data:	two-foot cloud data	0XFEA958EF07F41B615C1B36760C4DF9D89CDEA34F27F43CECB8A0F60B4FE3E8DB
	data amount:	10 groups	
	unit size:	2 to 5 MB	
	unit price:	more than 1.5 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F0474FBA430E4B6E9ACE82ACA584AB	
	hash of USN:	0X5B779E7EF4C7792E484B146CE224415F46727C9EEC21D2132DBFC5F753C44C4B	

Table 4. *Cont.*

Node Account	Data Attribute Description		Attribute Hash
DP2:0x994CDD94D999F282 93412FF9AE1F7BCEE8270BF6	data:	two-foot cloud data	0X32D26333AD8D7A72 5391DA6B8CCB131B21C85 6671AE809363D638C 89DB73DED4
	data amount:	500 groups	
	unit size:	2 to 6 MB	
	unit price:	more than 0.2 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F04 74FBA430E4B6E9A CE82ACA584AB	
	hash of USN:	0X33FACA8AA322A FD9B47A49AE0BA 8C0040AF5C989FA1 CB0E4D089A42A376946FF	
DP3:0x1B83F6F1EEC42A23CA BD86BDC161D980BB425A55	data:	two-foot cloud data	0X0479180CA0C14869E132 C0636BA6C874A5EABC7F533 F822037FFD19D64F49F1D
	data amount:	1 group	
	unit size:	10 MB	
	unit price:	1 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F0474F BA430E4B6E9ACE 82ACA584AB	
	hash of USN:	0XC54786BCFEAF46FC D9D877DC04D726B C6D16B74BF741D487F 9CB41D894DE6949	
DP4:0x9E977724A12FAFD04F 6190FCD8F9E50F97597128	data:	right foot cloud data	0X936C40C1802E74C E98A9E2CDA2A95F7A0 B2358D26389B58B3BC C08CCC5F1070A
	data amount:	10 groups	
	unit size:	3 to 5 MB	
	unit price:	more than 1.2 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F0 474FBA430E4B6E9A CE82ACA584AB	
	hash of USN:	0XC4A47A5829 69241CBEA4124C3 F1B41B987E84F69330F3 25961B19F1B3930C5B2	
DP5:0xADB9A686AC0ED116F4 139A71F163F1B67624ALE4	data:	right foot cloud data	0XDC62CB2395A000DE8F29 9DBF3C498FB792182F5D2B8 4DD08A8DB1B407EA0A1B6
	data amount:	5 groups	
	unit size:	2 to 4 MB	
	unit price:	more than 1 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F0474 FBA430E4B6E9A CE82ACA584AB	
	hash of USN:	0X66D0FD0872FAD6 52360A2CC950B3B68D 20ED55BA563CF9871 882C165F82E7DE8	

Table 4. *Cont.*

Node Account	Data Attribute Description		Attribute Hash
DP6:0x238E262CE05A87E03 CAB1E8CD2A5BFF472ED07CC	data:	face cloud data	0X2CF73EC50C44FE C18230D174CC6FD34B D05FA49A6523F96C E8E98AD7F21170F7
	data amount:	3 groups	
	unit size:	2 to 7 MB	
	unit price:	more than 0.5 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0xBB04015B84FED E3F98270054BC2168 9DCDEF765B	
	hash of USN:	0XFAFBE27F3718 FA3B86D2390929E2 D43973A3074AE9CF46 0B7BD5615B4A4F9C45	
DP7:0x60E7E355A38839718 36231436AF9230309FDC3AE	data:	two-foot cloud data	0X348BEFC97E10 E337253922446C7 CC729965B146E76F 54FDE2398AC6E7D95D031
	data amount:	1 group	
	unit size:	3 MB	
	unit price:	1 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F04 74FBA430E4B6E9 ACE82ACA584AB	
	hash of USN:	0X5A5C1BB6E 46E8F4955DAA 41CA4107F49D2 4111FD3B587C6CF8997 6E1E0906F96	
DP7:0x60E7E355A3883971836 231436AF9230309FDC3AE	data:	face cloud data	0X5F814EB245DC90C 89C87CF7FA316B9E058 79396BF417E3224F4 A0E4B92F24BBC
	data amount:	2 groups	
	unit size:	5 to 6 MB	
	unit price:	more than 0.5 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0xBB04015B84FEDE3F982 70054BC21689DCDEF765B	
	hash of USN:	0XC43ED4200761EFA1201 76862DCD9DF4DBD623120C22 73DFD758176349215AD2C	
DP8:0xDE6372D25 72404DB317F43AD49 C5997EF46C3AAF	data:	face cloud data	0X7AD3F0B3C9 AF4F836571D798C A6EF6A9F403C9B5508 C7311369F101E91C4FDBA
	data amount:	10 groups	
	unit size:	2 to 5 MB	
	unit price:	more than 1 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0xBB04015B84FEDE3F982 70054BC21689DCDEF765B	
	hash of USN:	0XCCB75618F5EF3CB E5E1B8C3E7504FE4F98F70D BB9CF2050D5034D C8539A2A8EB	

Table 4. *Cont.*

Node Account	Data Attribute Description		Attribute Hash
DP9:0xEC6E3D1BEB36FFDF 6C0180D42EB17BE66EFA0885	data:	two-foot cloud data	0X35EBDD4367911A 1DE45B31EF6011D1F 3191F1F93B345B299 E22EB4DD644FC67C
	data amount:	50 groups	
	unit size:	1 to 4 MB	
	unit price:	1.5 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0X7CE6AF1006F047 4FBA430E4B6E 9ACE82ACA584AB	
	hash of USN:	0XA6C995A0FB7E F493337042F00725 5A22C14FC710C631 43C81B02C30452406CC9	
DP9:0xEC6E3D1BEB36FFDF 6C0180D42EB17BE66EFA0885	data:	face cloud data	0X971F95007041C E9D7AF161BD5F9 17B4BF6EB161A27620C 71153D7CA4DA325720
	data amount:	5 groups	
	unit size:	3 to 6 MB	
	unit price:	0.75 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0xBB04015B84 FEDE3F98270054B C21689DCDEF765B	
	hash of USN:	0X6A2808BEA0D390 E0A755ED9644FD9E 998E8CFACE696A02D4 7B0334FAB94DF84B	
DP10:0xF67C41CD59F097C8 22A1CCFCDFBB61CF6FE28169	data:	face cloud data	0X03E88CF0BFD85868BE4D 372CA7930C78DEE5EBAB3C 8015FD0E44DE0DB1143D1B
	data amount:	8 groups	
	unit size:	2 to 6 MB	
	unit price:	more than 0.8 Milliether	
	data resolution:	1 mm depth accuracy	
	data acquisition:	Intel RealSense	
	responding:	0xBB04015B84FE DE3F98270054BC21 689DCDEF765B	
	hash of USN:	0XA5F7F7DB06D75B3D E2FB61B3247AE74 8BAB88636A11DCED38A2D 4877A3F019BA	

With the command line tool, we can view DP register contract. As shown in Figure 7, the data attributes were recorded as a <DP account: data attribute hash>. The cryptographic functions ensure the confidentiality of DP data attributes and provide a validated method.



Figure 7. The data attribute value of DP register contract.

5.1. SGX Execution Performance Evaluation

In the data requirement matching phase, the execution time performance is mainly affected by the number of SGX nodes. More SGX nodes need longer synchronization time to obtain a data matching list. For the same DR1 request, seven potential DPs were interested in providing data. Without consideration of time consumption in DP identity validation, we performed a data matching computation 500 times with different numbers of TNs and recorded the result acquisition time in Figure 8.

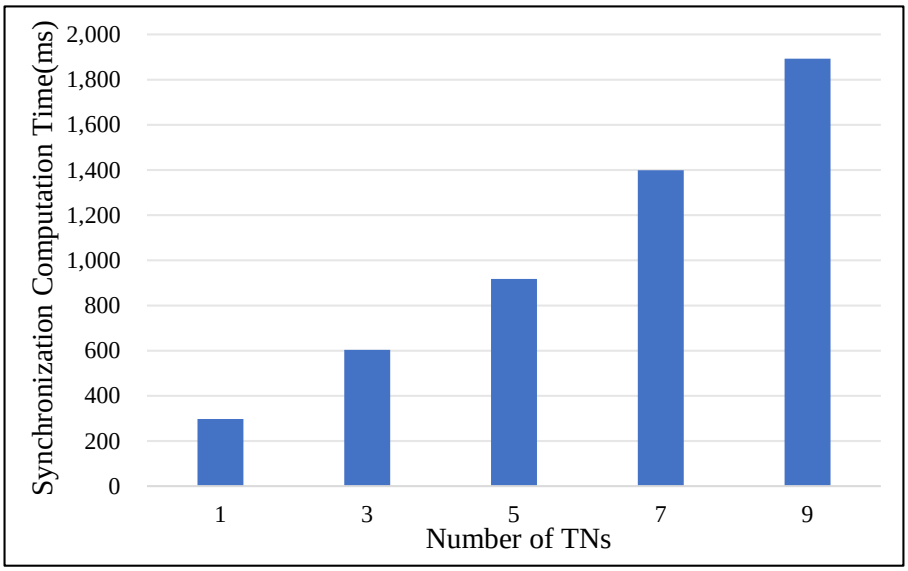


Figure 8. The data demand matching computation time of different number TNs.

From Figure 8, we can conclude that when the number of TNs increases from 1 to 3 and then to 5, the time difference is basically maintained at about 310 ms. However, when the number of TN increases to 7 and 9, the time difference increases by about 490 ms. The reason for this is that we simulated five enclaves as TNs on the same machine and four extra enclaves as TNs on another machine during this experiment. Whether the TN number is 1, 3 and 5, the time difference mainly exists in SGX operations with the same machine. The increased time difference of 7 and 9 TNs is probability consumed in a network link transmission. We believe that when the SPPCS is running, the time consumption for data matching computation of every additional TN should be increased by more than 490 ms, and the reliability of computation has been improved. Therefore, DR needs to balance the system computation performance with the result list safety when setting the required TN number.

The time-consuming hybrid computation is decided by data transfer capacity, calculation size and SGX operations. In the same network environment and computing platform, fewer data should definitely allow for data transmission and calculation to be carried out more quickly. Then, in the same network environment and TN Leader, we mainly tested the time overhead of SGX operations. For the foot point cloud data of A about 4 MB (selected from DP1), B about 2 MB (selected from DP2), C about 4 MB (selected from DP7) and D about 5 MB (selected from DP9), we separately executed the same foot analysis algorithm 1000 times in the same hardware machine with SGX enabled/disenabled and recorded the corresponding average consumption times in Figure 9.

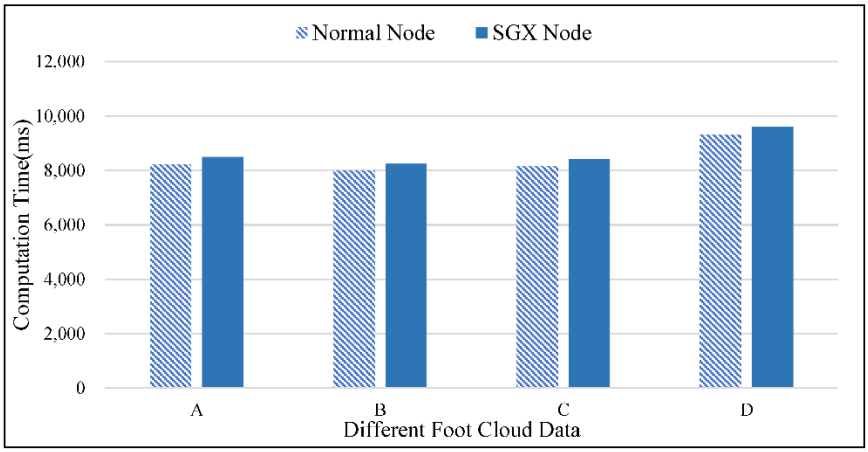


Figure 9. The computation performance comparison between ordinary environment and TEE.

Whether in TEE or an ordinary execution environment, the calculation time becomes longer as the data capacity increases. However, the time-consuming difference with different execution environments subtly increased by about 260 ms, which is mainly produced by SGX operations: load encrypted data from untrusted space and output encrypted results from a trusted region. The time consumption cost of SGX security guarantee is acceptable.

5.2. Transaction Execution Privacy Evaluation

In SPPCS, DRs and DPs complete data transactions with the TN leader. The multiple DPs and DRs could, respectively, trade with the leader node at the same time, and the corresponding transaction records are shown in Figure 10. DR1 and DR2 deploy computation contracts correspondingly to selected data matching lists on-chain and deposit double the expected value of the TN leader and DPs, as recorded in blocks 157 and 160 of Figure 10a. In this test, the selected data matching list of DR1 contains DP1, DP2, DP7 and DP9. DR2 selects DP6, DP7 and DP9 as appropriate traders. The TN leader validates identity authentication information from multiple responding DPs (DP1, DP7 and DP6) and obtains original DP data whilst paying a remuneration. Although DP7 provides both DR1 and DR2 demand data at the same time, DP1 and DP7 follow the even-numbered list transaction rules, whereas DP6 and DP7 do not meet the odd-numbered list transaction rules. As presented in Figure 10b, according to the data selection rules described in in hybrid transaction phase, TN leader prioritizes DP1 and DP7 to perform calculation analysis for DR1. Then, the TN leader continues to validate DPs until the data transaction with DP6 and DP9 is complete through the smart contract. As DP6, DP7 and DP9 are in line with the odd-numbered list transaction rules, the TN leader selects DP6 and DP9 for DR2 computation at one time. After the TN leader validates the last DP2 and acquires corresponding data, DP2 is selected to complete the DR1 computation. Block 166, block 175, block 176, block 194 and block 204 in Figure 10c record DPs sending data to the TN leader in detail. Figure 10d records transactions between the TN leader and DRs (DR1 and DR2). The TN leader returns computation results to DRs (DR1 and DR2) and is paid by calling the corresponding contract.

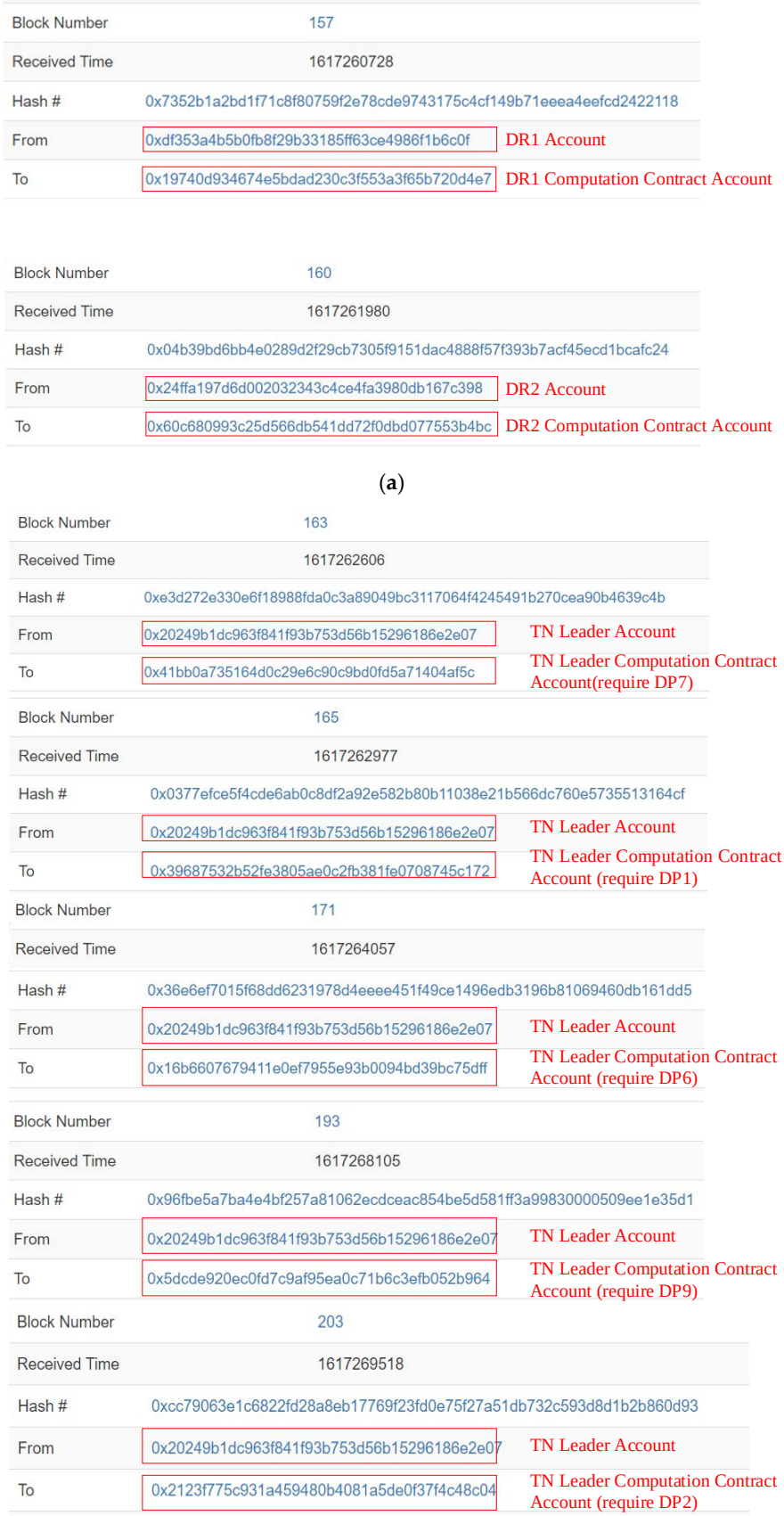
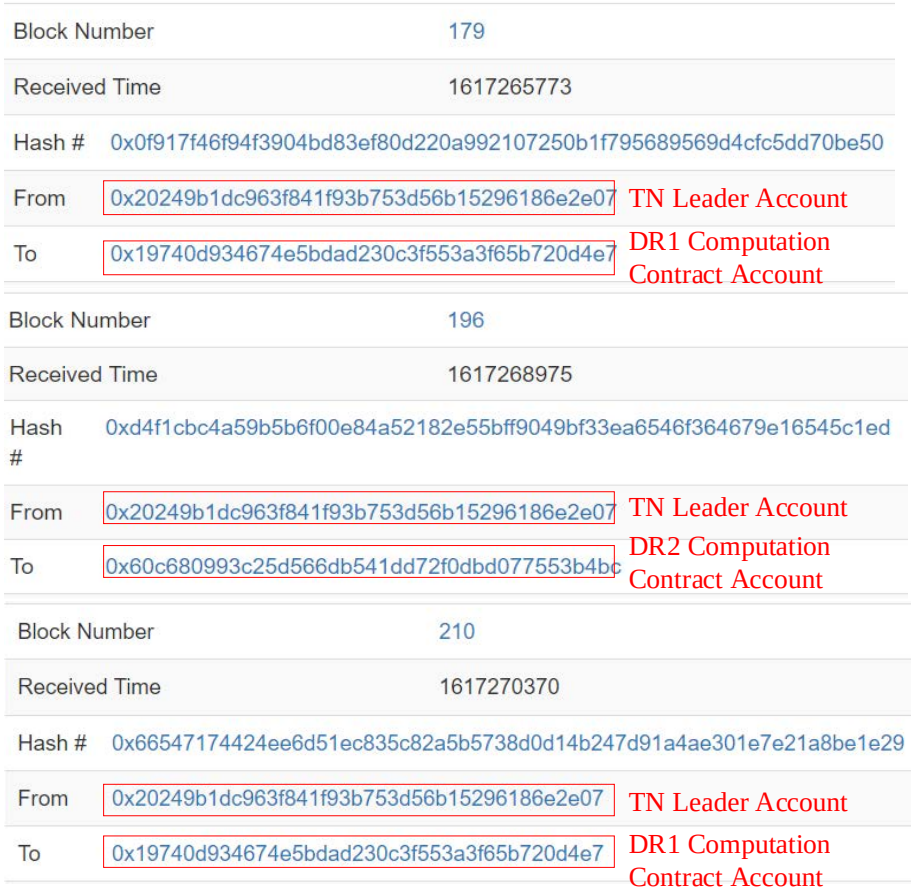


Figure 10. *Cont.*

Block Number	166
Received Time	1617263621
Hash #	0x3dd883491580cbeb39d212b7e6e617f9770eb6bca84256b2c236a9cc3ca052d1
From	0x7ecf438aed8b220900f9381fc3d317f26b176684
To	0x39687532b52fe3805ae0c2fb381fe0708745c172
Block Number	175
Received Time	1617264654
Hash #	0x9e9b832dcb8cf55fedc228061bb5631d9d79581375c9829d5d087095905e4802
From	0x238e262ce05a87e03cab1e8cd2a5bff472ed07cc
To	0x16b6607679411e0ef7955e93b0094bd39bc75dff
Block Number	176
Received Time	1617264970
Hash #	0xd678d51f10fc1bfb538af261cf98827a5306d85806e061444d50b754144f834b
From	0x60e7e355a3883971836231436af9230309fdc3ae
To	0x41bb0a735164d0c29e6c90c9bd0fd5a71404af5c
Block Number	194
Received Time	1617268342
Hash #	0xb18189d021005d5e6604da6a31b6e08e6b0f846e1d1eefac1deb2b63451c6e4a
From	0xec6e3d1beb36ffdf6c0180d42eb17be66efa0885
To	0x5dcde920ec0fd7c9af95ea0c71b6c3efb052b964
Block Number	204
Received Time	1617269952
Hash #	0x922dd910b7a3c8360b074b270cbfbbfab41ef7b12f054ba3f7ee649e69e49a18
From	0x994cdd94d999f28293412ff9ae1f7bcee8270bf6
To	0x2123f775c931a459480b4081a5de0f37f4c48c04

(**c**)

Figure 10. *Cont.*



(**d**)

Figure 10. The transaction process of DRs, DPs and TNs. (**a**) DR1 and DR2 publish computation contract. (**b**) TN leader prioritizes appropriate DP with selection rule. (**c**) DPs send data to TN leader at different times. (**d**) TN leader perform computation for DR1 and DR2.

From transaction process recorded in Figure 10, there is no correspondence between DP data and the TN leader computation data. The transaction relationship between DRs and DPs cannot be inferred directly. Even if a DP leaks private information in the real world, it is impossible to infer more information with records on-chain. Compared with general blockchain-based transaction systems, and although the extra cost is mainly focused to invoke smart contracts for data demand matching computation and TN leader status maintenance, the transaction privacy is highly guaranteed. Moreover, our SPPCS reduces the consensus burden by transferring complex computation off-chain to maintain a high performance.

6. Security and Privacy Analysis

The SPPCS, proposed in this paper, combines EB, smart contract technology, IPFS and TEE mechanism to guarantee data confidentiality. In this section, we analyze the security and privacy of this scheme from denial of service (DoS) attack, malicious fraud attack and data exposure.

6.1. DoS Resistance

TEE nodes are vulnerable to DoS attack by DRs and may repeatedly broadcast computation tasks without any following operations or DPs could continuously deliver data to waste network resources. Moreover, DR and DP nodes also deal with the problem of network bandwidth waste caused by deliberately frequently invoking smart contracts.

The SPPCS sets some rules against these problems: additional certain fee to be paid for invoking a smart contract, maximum number limitation of matching authentication and restrictions on remote attestation range.

6.2. Malicious Fraud Resistance

In the register phase, attackers could intercept requests to prevent DRs, DPs and TNs from completing registration. As a random event, participants complete registration by releasing or invoking smart contracts according to their wishes. Unless an interception attack is launched against a specific node, there is a very small probability that attackers block the legal request to prevent registration. Network attacks on specific targets can be prevented by adopting some network security strategies. In the computation and transaction phases, dishonest attacks may occur in DRs, DPs and TNs. The DR intentionally refuses to recognize TN calculation results and avoids paying TNs. The DP sends redundant invalid raw data to TN for extra remuneration. The TN fraud has two main purposes: provide dishonest analysis results to defraud rewards and maliciously steal or leak the original DP data. Moreover, collusion attacks are carried out between any two roles of the DR, DP and TN to obtain more rewards or utilize computing resources for free. The SPPCS mitigates dishonest attacks through combing the reputation feedback mechanism with a smart contract. In the data attribute matching phase, the DR needs a pre-deposit value and the final calculation result is decided by a delegation register contract, which avoids DR refuse to pay and obtains the wrong results. The TN with inconsistent calculation results should receive low reputation comments from smart contract, which greatly reduces the probability of the TN being selected again. In the computation phase, the TN validates whether original data hash is in line with the value registered on-chain, which refuses DP supply inappropriate data. Additionally, the encrypted original data are pulled into the TEE enclave, which prohibits data processions to be accessed and guarantees that the computation results cannot be tampered with. The change mechanism encourages DR to objectively evaluate TN, and TN to honestly evaluate DP. Furthermore, the pre-deposit broke DRs conduct the collusion attack and the reputation scheme effectively resists the collusion attack between DR and TN.

6.3. Confidentiality and Consistency

In SPPCS, various encryption and signature technologies are utilized to resist crack ciphertext during transmission. Blockchain consensus technology ensures the consistency of data status. In particular, TEE guarantees that the original data, process data and intermediate operations are inaccessible, which fundamentally maintains data confidentiality and privacy. Moreover, combined with TEE and blockchain, the dual hybrid transaction model is designed to isolate potential inferred connections between transaction parties. The transactions between DPs and DRs are completed through the TN leader. After obtaining the data matching list, the DR publishes selected transaction items, DPs provide corresponding encrypted original data to the TN leader and the blockchain maintains DR/DP transaction records with the TN leader simultaneously. As the TN leader receives multiple DR transaction entries during the term, the association between DRs and DPs is impossible to infer, which enhances the privacy and security of the SPPCS system.

7. Conclusions and Future Work

At present, unconscious sensitive data (such as 3D data) usage in a confidentiality-preserving and benefit-reserved manner is still a challenge. In this paper, the SPPCS is proposed, a trust hardware-based blockchain system to enhance privacy protection during sensitive data flow processes. The SPPCS decouples analysis computation and storage from consensus to achieve hierarchical separation computation. With a graph structure, the data leverage scheme of SPPCS performs a similarity computation of data expectation description and data provider description. The registered reputation value and costs on-chain are adopted by SPPCS as an important indicator to adjust similarity

computation results and influence the final data requirement matching list. Specifically, the difference between the selected matching data quality evaluation and registered quality pre-estimation should affect the data provider reputation. According to the reputation value, the SPPCS prevents dishonest participants and reduces invalid data transactions effectively. Moreover, with the TEE security guarantee, the SPPCS designs a dual hybrid isolation model to set restrictions on accessing raw data and transaction parties' relationships. The analysis and experiments conducted on the Ethereum private chain and SGX demonstrates that (1) the SPPCS needs to consider the balance of enhanced security and more than 490 ms time consumption produced by two more TNs, whilst multiple TNs perform privacy computations; (2) the SPPCS can support execution security and privacy preservation with about extra 260 ms time consumption for SGX operations; (3) the SPPCS confuses the transaction direction between participants and prevents privacy disclosure of transaction relationship from on-chain records.

In the current SPPCS, we do not solve the problem of computation overhead increased by large-capacity data exceeding the enclave memory limit. Moreover, the threshold for TN leader candidates is difficulty, which needs to deposit a large value as a commitment. Future research should focus on two aspects: (1) a security memory expansion scheme should be designed to improve the applicability of SPPCS; (2) the TN leader operating mechanism should be optimized to attract more participants.

Author Contributions: Conceptualization and methodology, M.Y.; investigation, M.Y. and X.L. (Xiru Li); writing—original draft preparation, M.Y.; writing—review and editing, M.Y.; visualization, H.T.; supervision and project administration, X.L. (Xiaofeng Li); resources, J.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant 61602435, in part by the National Natural Science Foundation of Anhui under Grant 1708085QF153.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yuan, M.; Li, X.; Xu, J.; Jia, C.; Li, X. 3D Foot Scanning Using Multiple Realsense Cameras. *Multimed. Tools Appl.* **2020**. [CrossRef]
2. Zhao, T.; Li, S.; Ngan, K.N.; Wu, F. 3-D Reconstruction of Human Body Shape from a Single Commodity Depth Camera. *IEEE Trans. Multimed.* **2019**, *21*, 114–123. [CrossRef]
3. Roth, J.; Tong, Y.; Liu, X. Adaptive 3D Face Reconstruction from Unconstrained Photo Collections. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2127–2141. [CrossRef]
4. Cha, Y.W.; Price, T.; Wei, Z.; Lu, X.; Rewkowski, N.; Chabra, R.; Qin, Z.; Kim, H.; Su, Z.; Liu, Y.; et al. Towards Fully Mobile 3D Face, Body, and Environment Capture Using Only Head-Worn Cameras. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 2993–3004. [CrossRef] [PubMed]
5. Liu, F.; Tran, L.; Liu, X. 3D Face Modeling from Diverse Raw Scan Data. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 9407–9417.
6. Chermprayong, P.; Kraichan, C. 3D Scanning with Ai-Powered Embedded System Streaming Digital Signage Via Redundant Network Attached Storage and Hybrid Cloud Storage. In Proceedings of the 2020 15th International Workshop on Semantic and Social Media Adaptation and Personalization SMA, Zakynthos, Greece, 29–30 October 2020; pp. 1–6.
7. Van Oosterom, P.; Martinez-Rubi, O.; Ivanova, M.; Horhammer, M.; Geringer, D.; Ravada, S.; Tijssen, T.; Kodde, M.; Gonçalves, R. Massive Point Cloud Data Management: Design, Implementation and Execution of a Point Cloud Benchmark. *Comput. Graph.* **2015**, *49*, 92–125. [CrossRef]
8. Al-Zahrani, F.A. Subscription-Based Data-Sharing Model Using Blockchain and Data as a Service. *IEEE Access* **2020**, *8*, 115966–115981. [CrossRef]
9. Bindahman, S.; Zakaria, N.; Zakaria, N. 3D Body Scanning Technology: Privacy and Ethical Issues. In Proceedings of the 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), Kuala Lumpur, Malaysia, 26–28 June 2012; pp. 150–154.
10. Wang, C.Y. [Dc] Privacy-Preserving Relived Experiences in Virtual Reality. In Proceedings of the 2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), Atlanta, GA, USA, 22–26 March 2020; pp. 531–532.
11. Ma, Z.; Liu, Y.; Liu, X.; Ma, J.; Ren, K. Lightweight Privacy-Preserving Ensemble Classification for Face Recognition. *IEEE Internet Things J.* **2019**, *6*, 5778–5790. [CrossRef]

12. Li, Q.; Zheng, Z.; Wu, F.; Chen, G. Generative Adversarial Networks-Based Privacy-Preserving 3D Reconstruction. In Proceedings of the 2020 IEEE/ACM 28th International Symposium on Quality of Service (IWQoS), Hangzhou, China, 15–17 June 2020; pp. 1–10.
13. Sekhavat, Y.A. Privacy Preserving Cloth Try-on Using Mobile Augmented Reality. *IEEE Trans. Multimed.* **2017**, *19*, 1041–1049. [CrossRef]
14. Sarwar, O.; Rinner, B.; Cavallaro, A. A Privacy-Preserving Filter for Oblique Face Images Based on Adaptive Hopping Gaussian Mixtures. *IEEE Access* **2019**, *7*, 142623–142639. [CrossRef]
15. Shirai, S.; Whitehill, J. Privacy-Preserving Annotation of Face Images through Attribute-Preserving Face Synthesis. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 21–29.
16. Karle, T.; Vora, D. Privacy Preservation in Big Data Using Anonymization Techniques. In Proceedings of the 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), Pune, India, 24–26 February 2017; pp. 340–343.
17. Xiong, L. Adaptive Differentially Private Data Release for Data Sharing and Data Mining. In Proceedings of the 2013 IEEE 13th International Conference on Data Mining Workshops, Dallas, TX, USA, 7–10 December 2013; p. 891.
18. Chhabra, A.; Arora, S. An Elliptic Curve Cryptography Based Encryption Scheme for Securing the Cloud against Eavesdropping Attacks. In Proceedings of the 2017 IEEE 3rd International Conference on Collaboration and Internet Computing (CIC), San Jose, CA, USA, 15–17 October 2017; pp. 243–246.
19. Zhang, X.; Liu, C.; Nepal, S.; Pandey, S.; Chen, J. A Privacy Leakage Upper Bound Constraint-Based Approach for Cost-Effective Privacy Preserving of Intermediate Data Sets in Cloud. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *24*, 1192–1202. [CrossRef]
20. Xue, W.; Hu, W.; Gauranvaram, P.; Seneviratne, A.; Jha, S. An Efficient Privacy-Preserving IoT System for Face Recognition. In Proceedings of the 2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT), Sydney, Australia, 21 April 2020; pp. 7–11.
21. Madankar, M.; Sawarkar, S.D.; Pete, D.J. Biometric Privacy Using Various Cryptographic Scheme. In Proceedings of the 2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN), Lonavala, India, 23–24 November 2018; pp. 159–162.
22. Yang, P.; Xiong, N.; Ren, J. Data Security and Privacy Protection for Cloud Storage: A Survey. *IEEE Access* **2020**, *8*, 131723–131740. [CrossRef]
23. Al-Janabi, S.; Al-Shourbaji, I.; Shojafar, M.; Shamshirband, S. Survey of Main Challenges (Security and Privacy) in Wireless Body Area Networks for Healthcare Applications. *Egypt. Inform. J.* **2017**, *18*, 113–122. [CrossRef]
24. Chouhan, A.; Kumari, A.; Saiyad, M. Secure Multiparty Computation and Privacy Preserving Scheme Using Homomorphic Elliptic Curve Cryptography. In Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 15–17 May 2019; pp. 776–780.
25. Subramanian, G.; Thampy, A.S.; Ugwuoke, N.V.; Ramnani, B. Crypto Pharmacy—Digital Medicine: A Mobile Application Integrated with Hybrid Blockchain to Tackle the Issues in Pharma Supply Chain. *IEEE Open J. Comput. Soc.* **2021**, *2*, 26–37. [CrossRef]
26. Malik, S.; Dedeoglu, V.; Kanhere, S.S.; Jurdak, R. Trustchain: Trustchain: Trust Management in Blockchain and IoT Supported Supply Chains. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 184–193.
27. Gourisetti, S.N.G.; Mylrea, M.; Patangia, H. Evaluation and Demonstration of Blockchain Applicability Framework. *IEEE Trans. Eng. Manag.* **2020**, *67*, 1142–1156. [CrossRef]
28. Ekawati, R.; Arkeman, Y.; Suprihatin, S.; Sunarti, T.C. Design of Intelligent Decision Support System for Sugar Cane Supply Chains Based on Blockchain Technology. In Proceedings of the 2020 2nd International Conference on Industrial Electrical and Electronics (ICIEE), Lombok, Indonesia, 20–21 October 2020; pp. 153–157.
29. Howard, S.; David, L.S.; Alex, P. Chapter 15 Enigma: Decentralized Computation Platform with Guaranteed Privacy. In *New Solutions for Cybersecurity*; MIT Press: Cambridge, MA, USA, 2018; pp. 425–454.
30. Li, X.; Mei, Y.; Gong, J.; Xiang, F.; Sun, Z. A Blockchain Privacy Protection Scheme Based on Ring Signature. *IEEE Access* **2020**, *8*, 76765–76772. [CrossRef]
31. Guan, Z.; Wan, Z.; Yang, Y.; Zhou, Y.; Huang, B. Blockmaze: An Efficient Privacy-Preserving Account-Model Blockchain Based on Zk-Snarks. *IEEE Trans. Dependable Secur. Comput.* **2020**, 1. [CrossRef]
32. Guo, H.; Li, W.; Nejad, M.; Shen, C. Access Control for Electronic Health Records with Hybrid Blockchain-Edge Architecture. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 44–51.
33. Di Francesco Maesa, D.; Mori, P.; Ricci, L. Blockchain Based Access Control. In Proceedings of the IFIP International Conference on Distributed Applications and Interoperable Systems, Neuchâtel, Switzerland, 19–22 June 2017; Springer: Cham, Switzerland, 2017; pp. 206–220.
34. Dai, W.; Dai, C.; Choo, K.K.R.; Cui, C.; Zou, D.; Jin, H. Sdte: A Secure Blockchain-Based Data Trading Ecosystem. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 725–737. [CrossRef]
35. Zhang, D.; Fan, L. Cerberus: Privacy-Preserving Computation in Edge Computing. In Proceedings of the IEEE INFOCOM 2020—IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 6–9 July 2020; pp. 43–49.
36. Wang, Y.; Li, J.; Zhao, S.; Yu, F. Hybridchain: A Novel Architecture for Confidentiality-Preserving and Performant Permissioned Blockchain Using Trusted Execution Environment. *IEEE Access* **2020**, *8*, 190652–190662. [CrossRef]

37. Maddali, L.P.; Thakur, M.S.D.; Vigneswaran, R.; Rajan, M.A.; Kanchanapalli, S.; Das, B. Veriblock: A Novel Blockchain Framework Based on Verifiable Computing and Trusted Execution Environment. In Proceedings of the 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS), Bengaluru, India, 7–11 January 2020; pp. 1–6.
38. Cheng, R.; Zhang, F.; Kos, J.; He, W.; Hynes, N.; Johnson, N.; Juels, A.; Miller, A.; Song, D. Ekiden: A Platform for Confidentiality-Preserving, Trustworthy, and Performant Smart Contracts. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy (EuroS&P), Stockholm, Sweden, 17–19 June 2019; pp. 185–200.
39. Zhang, Y.; Li, Y.; Fang, L.; Chen, P.; Dong, X. Privacy-Protected Electronic Voting System Based on Blockchin and Trusted Execution Environment. In Proceedings of the 2019 IEEE 5th International Conference on Computer and Communications (ICCC), Chengdu, China, 6–9 December 2019; pp. 1252–1257.
40. DuPont, J.; Squicciarini, A.C. Toward De-Anonymizing Bitcoin by Mapping Users Location. In Proceedings of the 5th ACM Conference on Data and Application Security and Privacy, San Antonio, TX, USA, 2–4 March 2015; pp. 139–141.
41. Natoli, C.; Gramoli, V. The Balance Attack against Proof-of-Work Blockchains: The R3 Testbed as an Example. *arXiv* **2016**, arXiv:1612.09426.
42. Dhillon, V.; Metcalf, D.; Hooper, M. The Dao Hacked. In *Blockchain Enabled Applications: Understand the Blockchain Ecosystem and How to Make It Work for You*; Apress: Berkeley, CA, USA, 2017; pp. 67–78.
43. Ott, M.; Choi, Y.; Cardie, C.; Hancock, J.T. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies—Volume 1, Portland, OR, USA, 19–24 June 2011; pp. 309–319.
44. Zhao, K.; Tang, S.; Zhao, B.; Wu, Y. Dynamic and Privacy-Preserving Reputation Management for Blockchain-Based Mobile Crowdsensing. *IEEE Access* **2019**, *7*, 74694–74710. [CrossRef]
45. Golden, M.; Pietra, P.; Teufel, H. Privacy Impact Assessment for Tsa Whole Body Imaging. 2008. Available online: https://www.hsdl.org/?abstract&did=234244 (accessed on 17 October 2008).
46. Mironenko, O. Body Scanners versus Privacy and Data Protection. *Comput. Law Secur. Rev.* **2011**, *27*, 232–244. [CrossRef]
47. Zuo, C.; Shao, J.; Liu, J.K.; Wei, G.; Ling, Y. Fine-Grained Two-Factor Protection Mechanism for Data Sharing in Cloud Storage. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 186–196. [CrossRef]
48. Jung, T.; Li, X.; Huang, W.; Qiao, Z.; Qian, J.; Chen, L.; Han, J.; Hou, J. Accounttrade: Accountability against Dishonest Big Data Buyers and Sellers. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 223–234. [CrossRef]
49. Wang, Z.W.; Vineet, V.; Pittaluga, F.; Sinha, S.N.; Cossairt, O.; Kang, S.B. Privacy-Preserving Action Recognition Using Coded Aperture Videos. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 1–10.
50. Sarwar, O.; Rinner, B.; Cavallaro, A. Design Space Exploration for Adaptive Privacy Protection in Airborne Images. In Proceedings of the 2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Colorado Springs, CO, USA, 23–26 August 2016; pp. 159–165.
51. Chriskos, P.; Zoidi, O.; Tefas, A.; Pitas, I. De-Identifying Facial Images Using Singular Value Decomposition and Projections. In Proceedings of the 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 26–30 May 2014; pp. 1240–1245.
52. Yu, J.; Kuang, Z.; Zhang, B.; Zhang, W.; Lin, D.; Fan, J. Leveraging Content Sensitiveness and User Trustworthiness to Recommend Fine-Grained Privacy Settings for Social Image Sharing. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 1317–1332. [CrossRef]
53. Li, T.; Lin, L. Anonymousnet: Natural Face De-Identification with Measurable Privacy. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 56–65.
54. Kim, T.; Kim, K.; Lee, J.; Matono, A. Dotloom: Toward a Decentralized Data Platform for Massive Three-Dimensional Point Clouds. In Proceedings of the 2019 IEEE International Symposium on Multimedia (ISM), San Diego, CA, USA, 9–11 December 2019; pp. 255–2553.
55. Wang, S.; Zhang, Y.; Zhang, Y. A Blockchain-Based Framework for Data Sharing with Fine-Grained Access Control in Decentralized Storage Systems. *IEEE Access* **2018**, *6*, 38437–38450. [CrossRef]
56. Lu, Y.; Huang, X.; Dai, Y.; Maharjan, S.; Zhang, Y. Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial Iot. *IEEE Trans. Ind. Inform.* **2020**, *16*, 4177–4186. [CrossRef]
57. Zheng, X.; Mukkamala, R.R.; Vatrapu, R.; Ordieres-Mere, J. Blockchain-Based Personal Health Data Sharing System Using Cloud Storage. In Proceedings of the 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom), Ostrava, Czech Republic, 17–20 September 2018; pp. 1–6.
58. Yaji, S.; Bangera, K.; Neelima, B. Privacy Preserving in Blockchain Based on Partial Homomorphic Encryption System for AI Applications. In Proceedings of the 2018 IEEE 25th International Conference on High Performance Computing Workshops (HiPCW), Bengaluru, India, 17–20 December 2018; pp. 81–85.
59. Al Omar, A.; Rahman, M.S.; Basu, A.; Kiyomoto, S. *Medibchain: A Blockchain Based Privacy Preserving Platform for Healthcare Data*; Springer: Cham, Swtzerland, 2017; pp. 534–543.
60. Benet, J. IPFS—Content Addressed, Versioned, P2P File System. *arXiv* **2014**, arXiv:1407.3561.
61. Bao, Z.; Wang, Q.; Shi, W.; Wang, L.; Lei, H.; Chen, B. When Blockchain Meets Sgx: An Overview, Challenges, and Open Issues. *IEEE Access* **2020**, *8*, 170404–170420. [CrossRef]

62. Niwattanakul, S.; Singthongchai, J.; Naenudorn, E.; Wanapu, S. Using of Jaccard Coefficient for Keywords Similarity. In Proceedings of the Lecture Notes in Engineering and Computer Science, Hong Kong, China, 13–15 March 2013; pp. 380–384.
63. Jin, C.; Bai, Q. Text Clustering Algorithm Based on the Graph Structures of Semantic Word Co-Occurrence. In Proceedings of the 2016 International Conference on Information System and Artificial Intelligence (ISAI), Hong Kong, China, 24–26 June 2016; pp. 497–502.
64. Das, S.; Muhuri, S.; Chakraborty, S.; Biswas, S. Graph Based Keyword Extraction for Similarity Identification among Born-Digital News Contents. In Proceedings of the 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; pp. 1–7.
65. Wu, M.; Guo, S.; Schaumont, P.; Wang, C. Eliminating Timing Side-Channel Leaks Using Program Repair. In Proceedings of the 27th ACM SIGSOFT International Symposium on Software Testing and Analysis, Amsterdam, The Netherlands, 16–21 July 2018; pp. 15–26.
66. Rachid, M.H.; Riley, R.; Malluhi, Q. Enclave-Based Oblivious Ram Using Intel's Sgx. *Comput. Secur.* **2020**, *91*, 101711. [CrossRef]