IEEE Access,Volume:9,Issue Date:21.June.2021

Data Trust Framework Using Blockchain Technology and Adaptive Transaction Validation

SARA ROUHANI AND RALPH DETERS, (Member, IEEE)
Department of Computer Science, University of Saskatchewan, Saskatoon, SK S7N 5C9, Canada
Corresponding author: Sara Rouhani (sara.rouhani@usask.ca)

ABSTRACT Trust is the main barrier preventing widespread data sharing. The lack of transparent infrastructures for implementing data trust prevents many data owners from sharing their data and concerns data users regarding the quality of the shared data. Data trust is a paradigm that facilitates data sharing by forcing data users to be transparent about the process of sharing and reusing data. Blockchain technology proposes a distributed and transparent administration by employing multiple parties to maintain consensus on an immutable ledger. This paper presents an end-to-end framework for data trust to enhance trustworthy data sharing utilizing blockchain technology. The framework promotes data quality by assessing input data sets, effectively manages access control, and presents data provenance and activity monitoring. We introduce an assessment model that includes reputation, endorsement, and confidence factors to evaluate data quality. We also suggest an adaptive solution to determine the number of transaction validators based on the computed trust value. The proposed data trust framework addresses both data owners' and data users' concerns by ensuring the trustworthiness and quality of the data at origin and ethical and secure usage of the data at the end. A comprehensive experimental study indicates the presented system effectively handles a large number of transactions with low latency.

INDEX TERMS Blockchain, data trust, data sharing, distributed, access control.

I. INTRODUCTION

Data sharing has become a big concern regarding privacy and confidential issues, abusing data, and legal and ethical violations. The lack of a transparent and trustworthy framework for data trust hinders many data owners from sharing their data, which could be vital for many research purposes. Data sharing is not merely a big concern for data owners, but also data users are concerned about the trustworthiness and reliability of the provided data at the origin. Hence, trust is a two-way problem for both data owners and data users.

Data trust is a fairly new concept that aims to facilitate data sharing by forcing data users to be transparent about the process of sharing and reusing data. Data trust entails legal, ethical, governance and organizational structure as well as technical requirements for enabling data sharing. Previous studies have suggested the potential of web observatory [1] and institutional repositories [2] for implementing data trust.

Blockchain technology has salient potential to effectively present the essential properties for creating a practical data trust framework by transforming current auditing practices and automatic enforcement of smart contracts logic, without relying on intermediaries to establish trust.

Many other studies have investigated blockchain potential for data sharing, establishing trust and access control. However, those are mostly scattered studies that have focused on a particular step or specific aspect in data sharing or have taken one side of the parties in data sharing by addressing only data owners' concerns.

Blockchain can be used as a data trust interface between data controllers and data users. The distributed, secure and reliable nature of the blockchain can reinforce the trustworthiness of the data trust framework.

O'Hara [1] introduces eight properties that should be considered for data trust architecture, including (1) discovery, (2) provenance, (3) access controls, (4) access, (5) identity management, (6) auditing of use, (7) accountability, (8) impact. Some of these properties, such as provenance, auditing of use, and accountability, already exist in the blockchain. Because blockchain provides a secure, immutable record of transactions, and all blocks are linked together through their hash values. Some other properties,

The associate editor coordinating the review of this manuscript and approving it for publication was Laxmisha Rai.

such as discovery, access control, access, and impact, could be implemented through smart contracts and be executed on permissioned blockchain. Identity management can be addressed by membership service in permissioned blockchains. Ultimately, accountability can reach because multiple peers validate transactions through consensus mechanisms, and the immutable ledger is maintained precisely through cryptographic methods. Besides, every peer has a copy of the ledger, and the network can easily recognize any inconsistency. Figure 1 illustrates how each element in a permissioned blockchain can be mapped to the required properties for data trust architecture stated by [1].

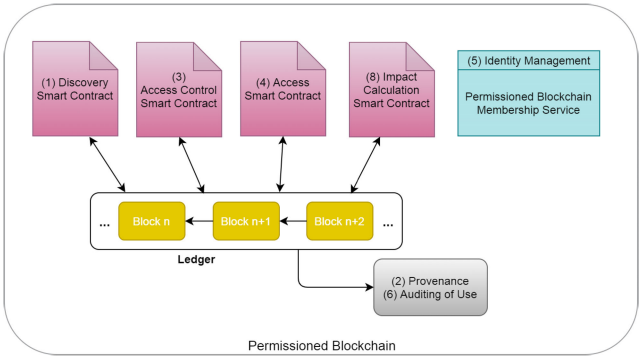

FIGURE 1. Blockchain as an infrastructure for data trust.

In this study, we propose an end-to-end framework for data trust based on blockchain, which ensures the trustworthiness and quality of the data at origin for data users and ethical and secure usage of data for data owners. First, we introduce a trust model to assess input data sets' trustworthiness using three parameters: data owner endorsement and reputation, data asset endorsement and data owner confidence level in the provided data set. All these parameters are recorded on the ledger, and they will be updated with every new transaction. We also apply adaptive transaction validation using Hyperledger Fabric state-based endorsement based on datasets trust value. Finally, we conduct a comprehensive performance analysis to demonstrate our system's efficacy in handling large sets of transactions and scaling across multiple organizations. We state that our system presents all the properties required for data trust. At the same time, it benefits from transparency, immutability, security offered by blockchain technology, and smart contracts' automation capabilities [3].

The rest of the paper is structured as follows. Section II introduces the concept of data trust, followed by a blockchain-based data trust framework. Section III discusses related studies. The architecture of the proposed framework is outlined in section IV. In section V, we present a trust model to formulate the trust value for the input data sets. Sharing data and access management are presented in section VI. VII presents the experimental results and evaluation of the system. In section VIII, we evaluate our system based on O'Hara's [1] data trust properties. And, Section X concludes the paper.

II. DATA TRUST CONCEPT

Trust is a multidisciplinary and multifaceted concept that has been defined in various disciplines, such as sociology, economics, psychology, computation, information and computer science, to model different types of relationships. Trust's definition can be challenging since it embraces the facets of ethics, emotions, values, and various disciplines. Fundamentally, trust is a relationship between trustor and trustee in which the trustor relies on the trustee based on a given criterion. Cho *et al.* [4] summarize the trust definition after studying trust across multiple disciplines. They define trust as the inclination of the trustor to accept a subjective belief that a trustee will exhibit responsible behaviour to maximize the trustor's interest under uncertainty of a particular situation based on the cognitive assessment of previous experience with the trustee.

Typically, digital trust is considered a computational value established from a relationship between trustor and trustee, and measured by trust parameters and evaluated by a defined mechanism [5].

Since the development of big data and data science, many technologies have advanced rapidly. However, the processes of collaborative data sharing, reusing data, and data privacy protection are still facing serious problems, such as privacy and confidential issues, abusing data, illegal reusing of data, and legal and ethical violations. Data trust is a mechanism to address these problems by providing a structured and solid framework for data stewardship and facilitating the process of access to data.

The main idea is that we have a well-structured, transparent and trustable framework for data stewardship. Sharing data might compromise data subjects' interests at risk by exposing their personal information. Still, it could lead to loss and fine for the organization and consequently reputational damage. Therefore, it is expected to mitigate some of the perceived risks of data sharing with data trust. Moreover, the framework must be designed to respond to both data owners' and data subjects' concerns by providing ethical principles underlined in the data trust framework and ensuring the data users regarding the data's trustworthiness and quality. Data owner refers to an individual or an organization that owns and controls the data. Data subjects are individuals that the data is related directly or indirectly to their personal information. Data users or data consumers are individuals or organizations that use the data for data scientists and analyzing purposes.

Although data trust could be a law arrangement or contractual agreement, it is possible to be programmed into the architecture that satisfies specified requirements. It is essential to understand the interaction between different actors and components and establish mutual trust. O'Hara proposes eight essential properties that underlie data trust architecture [1], (1) discovery, (2) provenance, (3) access controls, (4) access, (5) identity management, (6) auditing of use, (7) accountability, (8) impact. He proposed the Web Observatory as a candidate technology to carry out the required operation of a data trust.

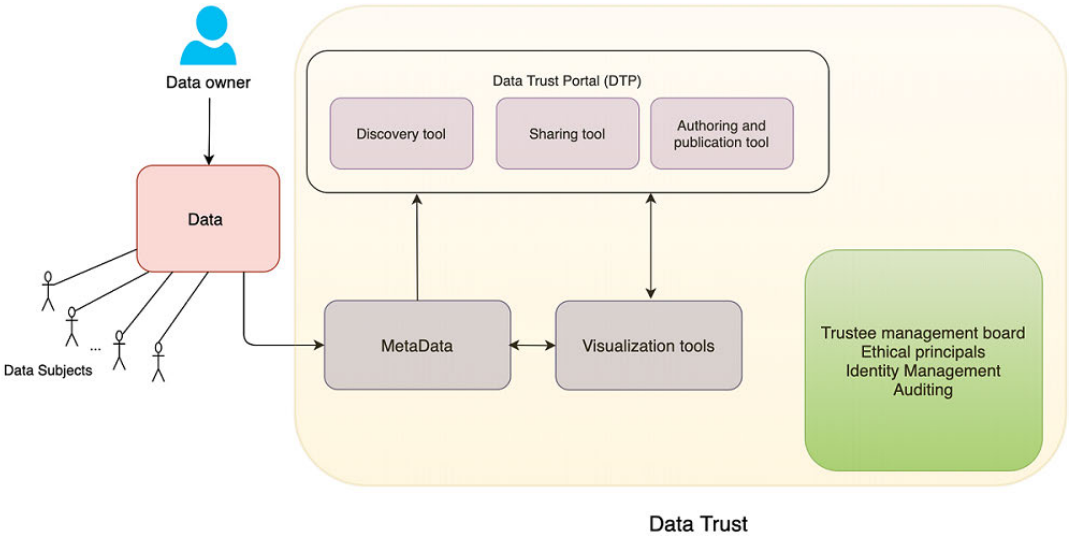

FIGURE 2. Data Trust Portal (DTP) Architecture by O'Hara [1].

- Discovery refers to the process of discovering the quality and properties of data by data users in the first place.
- Provenance refers to the ability of data users to access the historical record and metadata about the data.
- Access control refers to the ability of data owners to control and manage access permissions toward their data.
- Access refers to the mechanism that provides access for data users.
- Identity management refers to the ability of data owners to identify and authenticate data users.
- Auditing of use refers to providing a transparent history of data usage.
- Accountability refers to achieving accountability by access control and auditing of use.
- Impact refers to assessing the value, usage and misuse of data through recorded information in the data trust.

Figure 2 illustrates O'Hara's architecture for data trust called Data Trust Portal(DTP) for sketching out the above features inspired by web observatory infrastructure. The data does not store in DTP, data owners hold the data, and they are responsible for their data protection and the implemented interface method to provide access. DTP is a platform for implementing secure discovery and sharing protocol using metadata about data properties and provenances.

Alsaad *et al.* [2] also introduced institutional repositories, which is both a mature platform and open framework, as an alternative technology for data trust infrastructure.

Stalla-Bourdillon *et al.* [6] presents a workflow that addresses data protection by design approach to achieve effective data usage, sharing, and reusing. The authors emphasize that such a design requires well-defined data governance roles and processes. They represent data trust through three core layers as represented in figure 3: (1) the data layer (2) the access layer (3) the process layer.

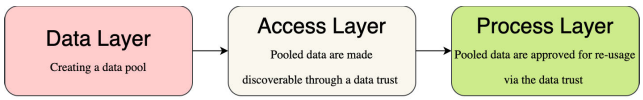

FIGURE 3. The workflow of data protection by design approach [6].

In the data layer, interested parties sketch out the step of creating data pools by making sure that everyone is aware of the legal requirements for data protection by design, specifying the authorized individuals to decide and act on the pooled data, and preparing data for sharing by applying technical procedures to remove personal identifiers, such as de-identification and anonymization [7].

In the access layer, the data becomes discoverable for eligible parties by specifying standardized access through centralized or peer-to-peer technical solutions, which are complemented by monitoring and auditing processes.

In the process layer, the pooled data are approved for re-usage via the data trust. This layer controls data usage through standardized risk assessments and ensures that data are tailored to queries.

III. RELATED WORKS

Various studies have investigated blockchain's potential for trusted data sharing. Some studies have considered incentive mechanisms to encourage data owners to share their data without losing control and ownership. Data's quality and trustworthiness have been assessed through multiple trust models such as reputation-based models, smart contract verification, and algorithmic solutions. Table 1 outlines the summary of incentive-based and quality control for data stewardship using blockchain technology.

Shala *et al.* [22] introduced an incentive mechanism to motivate low trusted peers in the IoT network to increase their trust score. The incentivization system uses control loops

TABLE 1. Blockchain based data sharing and quality control studies.

Study	Incentive-based	Quality control	Blockchain network	Implementation	Application domain
Xuan et al. [8]	Yes	-	-	Partially	General data sharing
Shrestha et al. [9]	Yes	-	Ethereum (public)	Yes	General data sharing
Shen et al. [10]	Yes	-	-	Partially	Cloud data
Chen et al. [11]	Yes	Yes	Ethereum (public)	Yes	Internet of Vehicles (IoV)
Zhu et al. [12]	Yes	-	-	Partially	Medical data
Su et al. [13]	Yes	Yes	-	Yes	Disaster rescue
Casado et al. [14]	-	Yes	-	No	IoT
Zheng et al. [15]	-	Yes	Ethereum (public)	Yes	Medical data
Cappiello et al. [16]	-	Yes	Ethereum (public)	Yes	General data sharing
Huang et al.[17]	-	Yes	Ethereum (public)	Yes	Crowd sensing
An et al. [18]	-	Yes	-	Yes	Crowd sensing
Wang et al. [19]	Yes	Yes	Gervais's Bitcoin-Simulator	Yes	Crowd sensing
Zovolokina et al. [20]	Yes	Yes	Hyperledger Fabric (permissioned)	Yes	Digital car dossier
Dedeoglu et al. [21]	-	Yes	Custom private blockchain	Yes	IoT
Shala et al. [22]	Yes	-	-	Yes	IoT
Yue et al. [23]	-	-	-	-	Big data sharing
Brandao et al. [24]	-	-	-	-	Smart places
Kang et al. [25]	-	Yes	Consortium blockchain	Yes	Vehicular edge computing
Yang et al. [26]	-	Yes	-	Yes	Vehicular networks
Kim et al. [27]	-	-	-	Yes	Wireless sensor network
Kochovski et al. [28]	-	-	Ethereum (public)	Yes	Fog computing
Wei et al. [29]	-	-	-	Yes	Cloud data
Choudhury et al. [30]	-	Yes	Hyperledger Fabric (permissioned)	Yes	Medical data
Hang et al. [31]	-	Yes	Hyperledger Fabric (permissioned)	Yes	Fish farm
Presented system	-	Yes	Hyperledger Fabric (permissioned)	Yes	General data sharing

that contain a target trust score. For the service providers with low trust scores, a package of incentives, such as discounts for other services, will be sent to encourage them to offer a better service in exchange for the promised benefits. In [12], authors presented an incentive-based model to encourage medical data owners to share their high-quality data (real and practical) and earn revenues, as well as miners who get benefit by participation and validating transactions. Wang *et al.* [19] introduced a privacy-preserving incentive mechanism to achieve high-quality contribution in crowdsensing. The trust model motivates participants to share their high-quality sensing data and profit in the form of Bitcoin or Monero cryptocurrencies. Miners verify the quality of data and earn revenue as well. Zavolokina *et al.* [20] provided a financial incentive for participating in the network and provides high-quality data for car dossiers. The system expects to fix errors by punishing harmful behaviour. They employ smart contracts for automatically calculating and enforcing incentives. Shrestha and Vassileva [9] utilized blockchain and smart contracts to encourage data owners to share their research data without losing control and ownership of it. The system incentivizes the participants by providing access to aggregate, anonymized data to involve them as miners in the network.

In [25], a subjective logic model has been used to assess nodes' reputation to ensure high-quality data sharing in the vehicular network. Dedeoglu *et al.* [21] presented a trust model to assess the quality of data observed by sensor nodes in the IoT network. The model consists of three elements: evidence from other neighbour sensor nodes' observations, the data source's confidence, and its reputation. They also employ blockchain to control the quality of shared data by detecting inaccurate or suspicious data captured by IoT devices or mobile crowdsensing. Choudhury *et al.* [30] ensured data quality while maintaining data privacy. Regulatory agencies assess the quality of data as network participants. Data privacy is ensured by creating activity-specific private channels. An *et al.* [18] presented a lightweight consensus mechanism called delegated proof of reputation (DPoR) to solve the heavy computation problem appropriate for data quality control in crowdsensing nodes. Huang *et al.* [17] ensured the quality of collected data from sensor nodes in the crowdsensing network through verification rules embedded in smart contracts. Su *et al.* [13] designed a two-tier reinforcement learning (RL)-based incentive algorithm to improve high-quality data sharing. Casado-Vara *et al.* [14] also presented a cooperative algorithm based on game theory in the edge computing layer to promote data quality and false data detection.

IV. SYSTEM ARCHITECTURE

As we discussed earlier, our proposed system aims to establish a data trust framework beneficial for both data owners and data users. To tackle this goal, we present two main components in our system architecture. 1) a trust model to examine the quality of input data sets and 2) a secure and traceable access control management. Figure 4 presents our data trust framework architecture.

We model trust for the input data sets. For any initial data set, our system calculates its trust value through a blockchain-based application. This value is used to ensure only trusted data sets are confirmed, and the system only records trusted data assets on the ledger. Section V explains

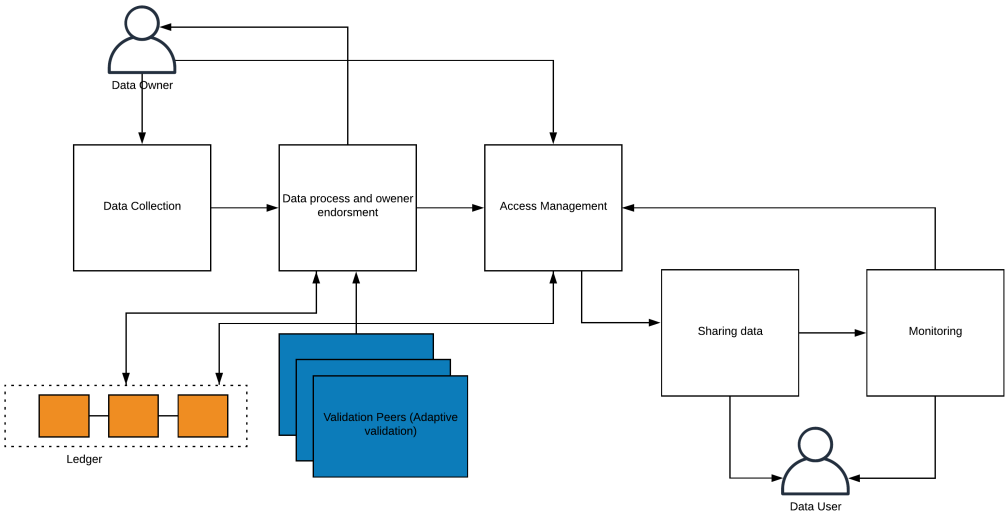

FIGURE 4. End to end data trust architecture with adaptive validation.

which parameters are involved and how the trust value is calculated.

The data sets with lower trust values are considered suspicious, and they are required to be validated by more verifiers. This adaptive selection of the number of verifiers provides an acceptable trade-off between the system's data assets quality and its performance and resource consumption. In order to prevent a data breach by data investigators, they would have access to a small chunk of data. The accuracy and quality of the data are examined through that small chunk.

Once the data sets information is recorded as data assets on the ledger, the data users interested in accessing a data set can prepare a request to access the data set. The data owner will receive the requests directly, and they will decide to give access to their data sets under which terms and conditions. Using blockchain and exploiting smart contracts, all transactions are automatically enforced, and there are no third parties involved. The access control policies could be served through a smart contract, created and stored on the blockchain directly by the resource owner. Moreover, the data owners can transparently query the immutable and permanent storage of blockchain and trace those who had access to their data set previously and currently have access to the history of access requests despite the data owners' response type. Section VI describes the details of implemented smart contracts for access control and consent management.

V. TRUST MODEL

In this section, we discuss how we calculate the trust value for the input data sets. Later, this value will be available for the data users interested in a particular data set. Moreover, the system adaptively includes this value to define the number of verifiers for confirming the data set. The higher trust value requires fewer verifiers; therefore, the data set will be verified faster.

A. TERMINOLOGY

We call data sets *data assets* as they are assets that we use in our distributed data trust framework to manage them and control access to them. Every data asset has an owner (data owner) that has full control over the data. Every data asset has a unique identifier (key).

Data owners are the one who provides the data asset. The data owner has full control to decide who will have access to the data, for what purpose and the access permission level and conditions. The system provides a transparent history of all granting and revoking access permissions executed by the data owner.

The data sets provided by data owners may or may not include personal data. In any case, they require a precise procedure to share data sets. Sharing data sets that include personal data requires additional mechanisms to protect data subjects by preparing the data set in a way that does not contain individually identifiable information anymore. The pre-requisite steps must be provided to ensure sufficient privacy safeguards to protect data subjects' personal information. These mechanisms could include de-identification or anonymization. De-identification refers to erasing or replacing personal identifiers to make it difficult to re-establish a link between the individual and their personal information. Anonymization refers to the permanent removal of the link between the individual and their personal data to the degree that it would be practically impossible to re-establish the link [32]. This part is out of our study's scope, and we assume that the provided data sets have passed de-identification or anonymization steps.

Data users are potential users interested in data sets, such as data analysts, data miners, or data scientists, to extract knowledge, insights, and meaningful or profitable patterns from the data set. They use our data trust framework to discover and request access to their intended data sets.

They also require trust in the quality of the provided data set.

B. INPUT DATA SETS TRUST ASSESSMENT MODEL

In the represented data trust framework, we use a trust factor to examine the level of trust in the input data sets. This trust factor is considered to determine the number of verifiers to examine the data set's quality before confirming the data set and record it as a valid data asset in the ledger.

In our adaptive trust model, we model trust for the data asset with the key of i as:

$$Trust_i = f(reputation_{uj}, endorsment_{uj}, \lambda(confidence_i)) \quad \lambda \in [0, 1] \qquad (1)$$

where f is a function with three input parameters, including data owner's reputation, $reputation_{uj}$, data owner's endorsement, $endorsment_{uj}$, and data owner's confidence for the provided dataset, $confidence_i$. The λ factor determines the impact of the selected confidence in the total trust score. Later, we discuss how the λ coefficient is calculated based on the received assessments from data users. Figure 5 represents the trust assessment model for input data sets.

We have implemented three transactions in our smart contracts for calculating reputation, endorsement, and confidence parameters. The smart contracts read the required values from the ledger and calculate the specified parameter value. Finally, another transaction is implemented to invoke the three mentioned transactions to obtain reputation, endorsement, and confidence values and calculate the current data set's trust value. The details of the implementation of the smart contract and each transaction are presented in section VII

1) REPUTATION

The data owner's reputation is calculated considering the user's previous successful transactions as well as the minimum and the maximum number of successful transactions for all users, using the min-max normalization:

$$reputation_{uj} = \frac{\sum_{i=1}^{n} T_s - \min\limits_{\forall u \in U} \sum_{i=1}^{l} T_s}{\max\limits_{\forall u \in U} \sum_{i=1}^{h} T_s - \min\limits_{\forall u \in U} \sum_{i=1}^{l} T_s} \qquad (2)$$

where $\sum_{i=1}^{n} T_s$ is the number of successful transactions that the user uj have had so far. Also $\min\limits_{\forall u \in U} \sum_{i=1}^{l} T_s$ and $\max\limits_{\forall u \in U} \sum_{i=1}^{h} T_s$ are respectively the minimum and maximum number of successful transactions for all users in the framework.

More successful transactions, which means more submitted valid data assets by the data owner, increases the data owner's reputation. Multiple validators examine every input data asset; the number of these validators will be defined based on the trust factor. Because the new users do not have any history of valid data assets, their reputation score is relatively low. As they interact more honestly, they gain more reputation. Subsequently, their following transactions will require fewer validators, and their transactions will get validated faster.

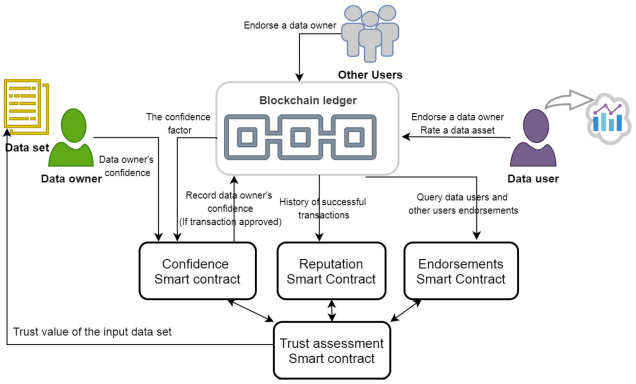

FIGURE 5. Trust assessment of input data sets.

2) ENDORSEMENT

Data owners can receive two types of endorsements. The first type can be received from any user in the system who knows the data owner; for example, they could have worked together previously. The second type of endorsement can be received from the data users who have previously studied a data set provided by the current data owner. The data owner could receive an endorsement based on the data user's experience if the data user approved the provided data set's high quality. The second type of endorsement has a more substantial influence on the data owner's total endorsement score, defined by the α factor. Endorsement value for the data owner j is calculated as:

$$endorsment_{uj} = 1 - e^{-(\sum endors_{uj} + \alpha \sum endors_{d_{uj}}) / \beta} \qquad (3)$$

where $\sum endors_{uj}$ is the total endorsements that the user uj has received from other ordinary users, and $\sum endors_{d_{uj}}$ is the number of endorsements, the submitted data asset has received in the framework. Also, β is a factor that defines the speed of reaching the highest possible endorsement value, which is one here. α is a positive weight determining the importance of data asset endorsement versus user endorsement.

Figure 6 illustrates the effects of alpha and beta on the user total endorsement value assuming $\sum endors_{uj}$ is 50 and $\sum endors_{d_{uj}}$ is 30.

3) CONFIDENCE

For any initial data set, the data owner enters a confidence value between 0 to 1 ($considence_i \in (0, 1]$) to express its confidence in the provided data set. This value will only be considered in the data asset's total trustworthiness if there is a history of previously entered data sets by the current data owner that was studied by a data user.

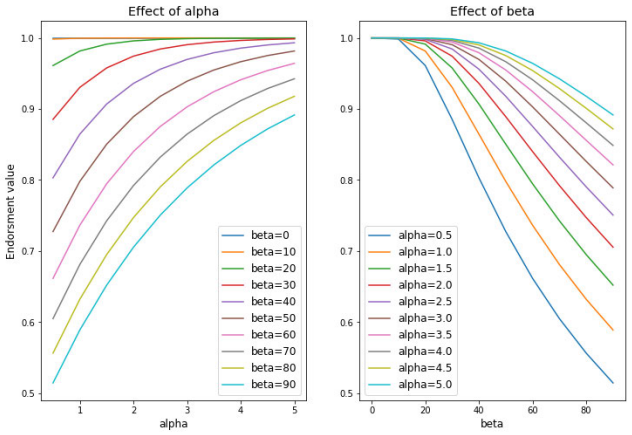

FIGURE 6. The effect of alpha and beta on the data owner's endorsement.

For the new data owners that the system does not have any assessment of their previous confidence value, the initial λ is zero. It means their confidence will not be considered to calculate the trustworthiness of the provided data set. However, this confidence value will be recorded in the system. Later, once a data user rates the data set's quality, the difference between the data owner's provided confidence and the data user's rate will be calculated as Δ. The proposed calculation of the λ value is given as:

$$\Delta = DataSetRating - DataOwnerConfidence$$
$$\lambda_{new} = \begin{cases} \Delta \geq 0 & min(1, \lambda_{old} + \phi(1-\Delta)) \\ \Delta < 0 & max(0, \lambda_{old} + \phi\Delta) \end{cases}$$
$$\lambda \in [0, 1]$$
$$\phi > 0 \qquad (4)$$

where ϕ is a factor that determines the speed of reaching the maximum of λ, which is one. Adaptively, as the data owners provide more accurate predictions about the quality of their data set through their entered confidence value, their input confidence value will later have more effect on the total trust value of the new data set.

VI. ACCESS MANAGEMENT AND SHARING DATA ASSETS

As discussed previously, our end-to-end data trust framework addresses both data owners' and data users' concerns. In the previous section, we explained how our data trust model calculates the trustworthiness of input data sets and how it adjusts data owners' confidence in their provided data sets. This section describes how we can implement a secure and trustable access management system using distributed ledger technology. We demonstrate how to design smart contracts to meet the requirements of the data trust framework.

Blockchain's features, such as transparency, auditability and trust distribution, along with leveraging smart contracts, make it possible to achieve secure and fine-grained access control [33], thereby promoting the data-trust framework. This section introduces our access control and consent management components for the presented data trust framework.

Once the data is approved, the data set could be recorded on a secure cloud storage service provided by the data trust implementing party or stay at the owners' side. In the second case, the owner party is responsible for providing access to the data set, but still, they adopt the blockchain-based access control and consent management system. Despite the location of storing the data sets, for any new data asset added to the data trust framework, a new record will be recorded on the ledger, including data asset id, the owner of the data asset and the hash value of the data asset. Any access permission requires the data asset owner's digital signature for approval.

Access permission can be granted to a particular user or a sub-group of users belonging to one or multiple organizations.

Three primary smart contracts are responsible for handling access requests, consent management and access provenance. Figure 7 illustrates the smart contracts design and interactions.

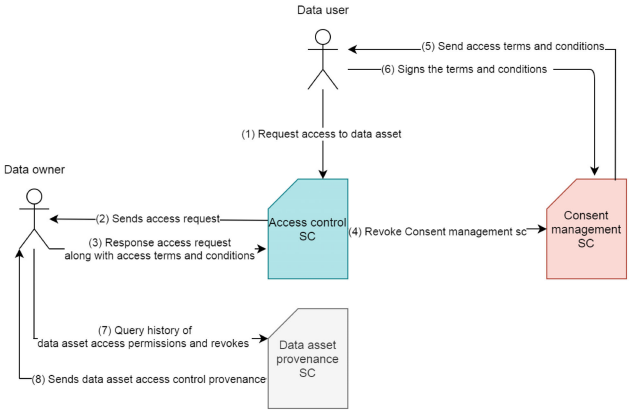

FIGURE 7. Design of smart contracts for access control, consent management and data provenance.

Access control smart contract receives access requests, checks default access permissions. Suppose the user has access to the data set based on the system's previous rules and prior consent between the data owner and data user. In that case, it submits a transaction for recording the access request and the result of access permission.

If there are no default access rules that match the current access request, it sends the data owner's access request. The data owner investigates the access request and decides to accept or reject it based on metadata provided by the data owner. Suppose the response received from the data owner agrees with the access request. In that case, the transaction invokes the consent management smart contract to handle the agreement between the data owner and the data user.

The consent management smart contract is responsible for sending the owner's terms and conditions to the data user and collecting the required signatures. It records the agreement to be permanently stored on the ledger.

When new access permission is granted, the list of users who have access to the data asset will be updated, and the id of the access requester will be added to the list. When access permission is revoked, the id of the user will be deleted from the list of current users who have access to the data. When we query the ledger and get the list of users who have access to the data, only the list of users who currently have access will be returned as the latest state of the ledger. However, when we use the *getHistoryForKey()* method, all the history and traces of access permissions and access revocations will be queried from the ledger.

Data asset provenance smart contract is implemented to query all the access requests, permissions, and revokes toward data assets. All transactions are appended to the blockchain history, whether valid or invalid. Therefore, all access requests are recorded on the ledger despite their acceptance and rejection outcome. They are valuable resources to analyze in the future and detect possible threats. If somebody had tried to compromise the security of the systems, all the access attempts were recorded on the transactions' history. Data owners are able to query the data users who currently have access to their data assets as well as query the previous history of access changes. Utilizing smart contracts for querying data asset provenance makes it possible to generate customized and flexible queries.

VII. SYSTEM IMPLEMENTATION

In this section, system implementation with regard to smart contracts design and implementation and adaptive endorsement.

We have used Hyperledger Fabric version 2.2.0 for implementing the system. Hyperledger Fabric [34] is a well-known permissioned blockchain with a modular structure, enabling pluggability and customization. It also supports private communication and private data collections. Organizations can establish private communication by creating separate channels. Each organization can participate in separate channels so that it can develop multiple private communications. Each channel has a separate ledger, and the communications are restricted to the organizations within the channel. Moreover, the ledger can be privatized granularly to include only a particular set of participants.

A. SMART CONTRACT DESIGN

Figure 8 presents the transactions implemented in smart contract (chaincode) to assess data trust. As discussed in section V, reputation, endorsement, and confidence are three main factors to asset the given data set's trustworthiness. Every data asset that is added to the system has multiple properties as following:

- dataID: a unique identifier for data assets.
- dataOwner: the userID that creates the data asset.
- hash: the hash value of the data asset stored in off-chain storage.
- owenrConfidence: the owners confidence in the provided data set. A float value between 0 to 1.

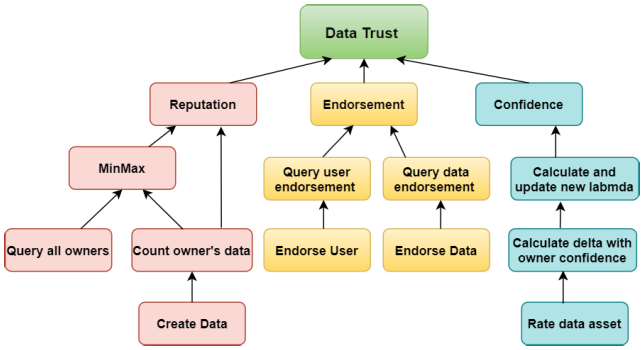

FIGURE 8. Smart contract transactions.

- endorsement: the number of endorsements that the data set is received. It is initialized to 0.
- rating: the rating value that the data set is received from the users who studied the data set. It is a float number between 0 to 1.
- lambda: this is a factor discussed in formula 4, that indicates the effect of data owner's confidence which initialized to 0.
- trust: the trust value that later will be calculated based on reputation, endorsement and confidence. It is initialized to null.

Reputation is based on the number of previous successful transactions that recorded a new data set to the system. This value will be calculated based on the ratio of the other users' reputations.

MinMax is a heavy computation transaction that queries all data assets stored on the ledger by dataOwners properties, counts the number of data for each owner's assets, and stores the minimum and the maximum number of owner's data stored on the ledger. Since MinMax is a heavy transaction, we do not invoke the MinMax transaction to calculate the users' reputation. The system can regularly (for example, every hour) execute the MinMax transaction and update the MinMax value on the ledger. Reputation transaction instead just queries the value of MinMax data stored on the ledger. This leads to significantly improving the performance of reputation transactions and, eventually, data trust transaction performance.

The endorsement is calculated based on two items, data asset endorsements and the owner of the data asset endorsements. Endorse User and Endorse data are two respective transactions that update data owners' endorsements and data assets' endorsements. Endorsement transaction calculates the data asset's endorsement based on querying the data related to the data asset endorsement and the owner of the data asset endorsement.

Every user who has studied the data asset can rate the quality of data by submitting a RateData transaction with a ratio between 0 to 1. Delta will be calculated based on the difference between the average ratings that the data asset has received and the owner's initial confidence. The new value

TABLE 2. Workload transactions.

Transaction name	Description	Actions on ledger
CreatData	Create a data asset belong to a random owner with random confidence	Write
MinMax	Queries all data assets stored on the ledger by dataOwners properties, counts the number of data for each owner's assets, and stores the minimum and the maximum number of owner's data stored on the ledger	Read-Write
Reputation	Calculates the reputation value for the input user by counting the ownser data and reading the MinMaX value form the ledger	Read
EndorUser	Read the stated of the data asset and updates the Endorsement value for the input user (increment by 1)	Read-Write
EndorsData	Read the stated of the data asset and updates the Endorsement value for the input data (increment by 1)	Read-Write
Endorsement	Calculate the endorsement value for the input data by querying the endorsement values belong to the data owner and data asset	Read
RateData	Read the stated of the data asset and add new ratring to the data asset	Read-Write
Lambda	Calculate the value of lambda based on delta (owner confidence and dataasset average rating), Update the new value for the lambda	Read- Write
DataTrust	Updates the value of Reputation, Endorsement, and Lambda (the effect on confidence) for the input data asset	Read-Write

for the lambda will be calculated and updated on the ledger accordingly. Confidence transaction queries the latest state for the lambda value for the data owner of the respective data asset.

Data trust transaction invokes the three transactions, Reputation, Endorsement and Confidence, and returns each item's respective values.

B. ADAPTIVE VALIDATION

Endorsement policies determine the smallest set of organizations with respect to their roles required to endorse and sign a transaction for it to be valid. The running peers refer to the endorsement policy to decide whether a transaction is valid. For example, OR('Organization1.member', AND('Organization2.peer', 'Organization3.peer')) is an endorsement policy that indicates either a member from organization1 must sign the transaction or one signature from a peer of the Organization2 MSP and one signature from a peer of the Organization3 MSP are required.

Hyperledger Fabric allows flexible endorsement at three different levels, chaincode level endorsement, collection-level endorsement and key-level endorsement.[1]

Chaincode-level endorsements are agreed to by channel members. It means all transactions implemented in the chaincode follow the same endorsement policy.

Collection-level endorsement policy sets the endorsement for a collection of data that are kept private in the channel.

Key-level or stated-based endorsement provides a granular level, which we can exploit to achieve adaptive validation based on data owners' trustworthiness and data assets. In the key-level endorsement, we can implement a different endorsement policy for any single data stored on the ledger. This way, we can set a looser policy that requires fewer signatures for data assets that are higher trustworthy. We can set a tighter endorsement policy that requires more verifiers for less reliable data assets.

VIII. EVALUATION

In this section, we present the result of the performance evaluation of the system. For implementing the system, we used

[1]https://hyperledger-fabric.readthedocs.io/en/release-2.2

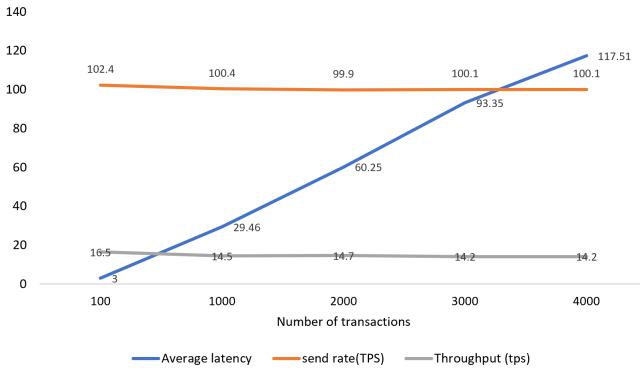

FIGURE 9. Send rate, throughput and average latency for CreateData transaction.

Hyperledger Fabric v2.2.0, and for measuring the system's performance, we used Hyperledger Caliper v0.4.2. The system setup is based on 32GB memory and 4vCPU. The default network setup includes two organizations and two peers. For the last experiment, a new organization is added to the network.

We have initialized the ledger with 1000 data assets, randomly belonging to 10 participants, and their confidence value is a random number between 0.1 to 1. Alpha, beta, phi values have been assumed to be the same constant number for all participants. Table 2 describes the transactions that are invoked during test runs and their actions on the ledger.

Data owners create a new data asset on the ledger by invoking the CreateData transaction. 9 illustrates average latency, send rate and throughput for CreateData transaction in five rounds experiments, in which a different number of transactions were generated in each round.

Figure 10 shows MinMax transaction average latency, send rate and throughput. MinMax is a heavy computational transaction that queries and counts all users' data assets and returns the minimum and maximum values. It stores these values on the ledger, which will be queried later for calculating data owners' reputations. This transaction can be regularly invoked to update the value of MinMaX. Therefore, Reputation transactions and, consequently, DataTrust

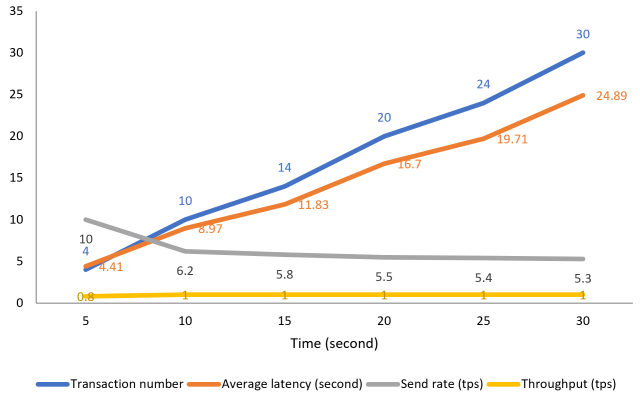

FIGURE 10. Calculate and record the minimum and maximum number of data assets belong to a single user (MinMax).

transactions do not overload with computing instantaneous MinMax values.

EndorsData and EndorsUser are two transactions that update the endorsement values for data assets and data owners. Ratedata also updates a data asset's rating by adding a new rating to the data asset rating array. Figures 11 and 12 shows the send rate, throughput and average latency for these transactions in five rounds experiments with a different number of transactions.

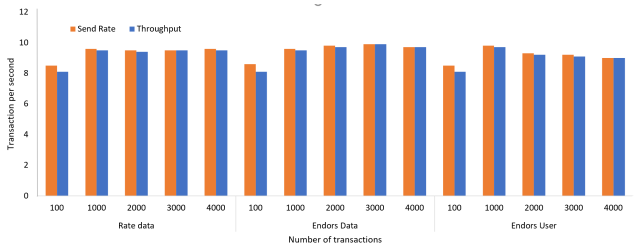

FIGURE 11. Sendrate and throughput for RateData, EndorsData, and EndorsUser transactions.

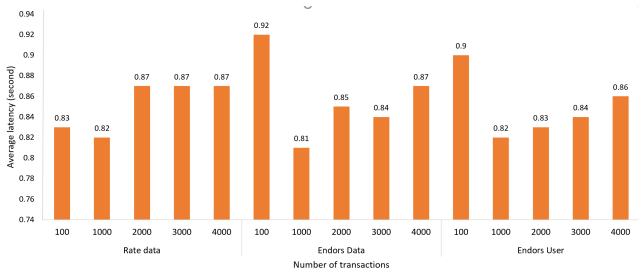

FIGURE 12. Average latency for RateData, EndorsData, and EndorsUser transactions.

A comparison between send rate and throughput in all transactions involved in computing data trust are presented in figures 13 and 14 respectively. This evaluation is also based on five rounds with a different number of transactions. Confidence and Endorsement transactions present the highest

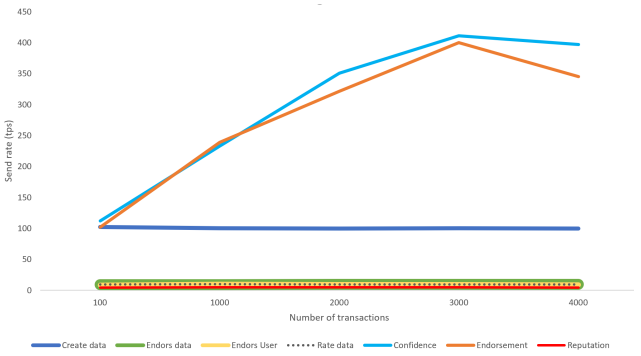

FIGURE 13. Send rate for all transaction.

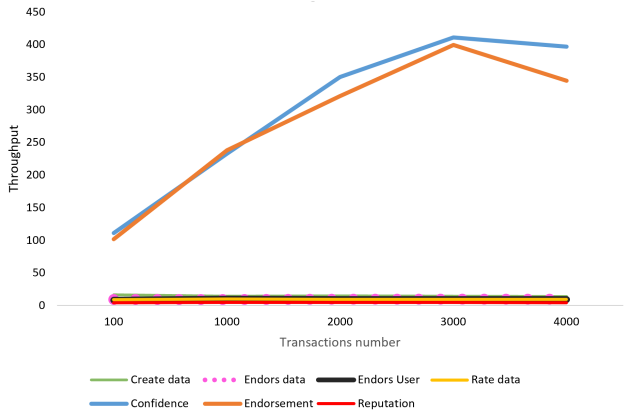

FIGURE 14. Throughput for all transaction.

send rate and throughput values as they query a single data asset and they calculate the Endorsement and Confidecne based on data asset properties explained in section VII (ownerConfidence, endorsement (both data and owner), rating, and lambda). CreateData transaction comparatively handles a high send rate (around 100 tps), but its throughput is between 16.4 to 14.2. EndorsData, EndorsUser, and RateData perform and update action on the ledger, and their send rate and throughput are around 10 tps. Reputation transaction has the lowest send rate and throughput, which are about 4.8 tps. Relatively, Reputation is a heavy transaction because it queries all data assets by filtering the data owner matches the current data owner.

Figure 16 presents the average latency for DataTrust transaction based on time. The left vertical axis indicates the number of successfully executed transactions during the given time, presented in the right vertical axis. The average latency for computing the data trust for 2502 transactions is 0.37 seconds completed in 500 seconds.

We added a new organization and one peer to the network. We run the experiments on all transactions again. The comparison result between the average latency of all transactions in a network with two organizations and two peers and three organizations and three peers is presented in figure 17. As it is illustrated in the graph, adding a new organization does not affect performance.

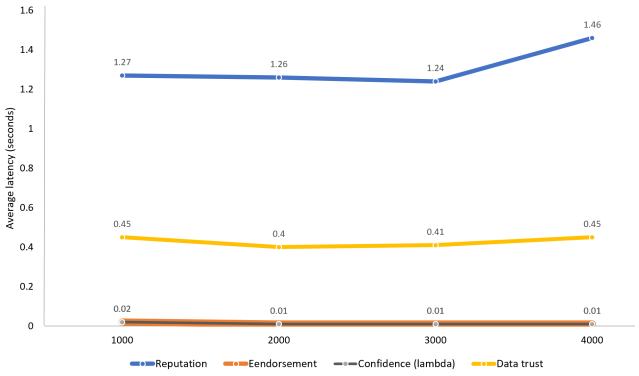

FIGURE 15. Average latency for DataTrust, Reputation, Endorsement and Confidence transactions.

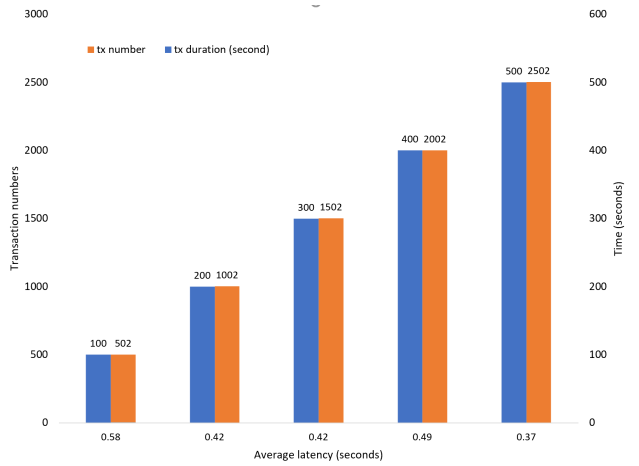

FIGURE 16. Average latency for DataTrust based on time.

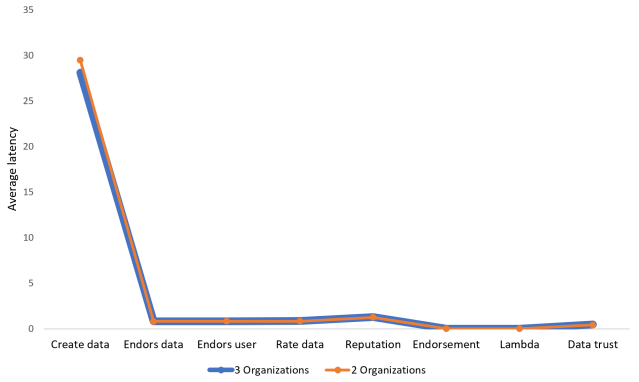

FIGURE 17. Increase the number of organizations.

IX. DISCUSSION

This section outlines how our proposed blockchain-based data trust framework enforces all the eight major requirements for the data trust represented by O'Hara [1]. Our scheme is shown to be sufficient, practical, and secure for trustworthy data sharing.

A. DISCOVERY

We have used a permissioned blockchain for implementing the system. Unlike public blockchains, in permissioned blockchains, only authenticated stakeholders can interact with the blockchain and access the data recorded on the ledger. Desirability permissioned blockchains such as Hyperledger Fabric [34] provide a more granular level of access control for participants, so users joining blockchain can have various access restrictions to the different components of blockchain by associating policies.

Authenticated data users are able to discover the available data assets, the properties of data sets represented as meta-data through the system interface.

The information related to data assets' quality is also available for the data users through the trust value calculated by our proposed method. The details of each parameter involved in trust calculation for the represented data owner and data set are also available for potential data users. Besides, once the dataset confirms through a transaction validator, they will add a review to the data set. Potential data users are allowed to access this review and discover the quality of the data set.

Most importantly, if a data user has previously studied the data set, the data user review regarding the quality of the data set is recorded on the ledger and available for future data users. It is an essential resource as the data users who studied and analyzed the data could bring the most accurate perspective on the data's quality.

B. PROVENANCE

Once the data owners add their data sets as data assets to the system, they must attach metadata related to the data provenance, such as the data origin, collection time, and collection method. This information can help both transaction verifiers and data users to assess the quality of the data.

Moreover, every time a data set is modified, an associated transaction is generated to update the data asset properties on the ledger. It helps to query data provenance and trace data evolution by identifying actual operations that have been performed on the data sets.

C. ACCESS CONTROL

Data owners have full control over their data assets. They are the ones who decide on who gets access to their data, and by exploiting smart contracts, their access management is enforced automatically. Smart contracts also enable fine-grained access checks to verify the authenticity of submitted transactions.

As we discussed in detail in section VI, data owners can set default users who can have access to their data sets or receive access requests from any data users. They inspect the request based on the purpose of the data users, and they decide to deny or accept the request and under which circumstances.

The consent management smart contract records the consent between the data owner and data user on the ledger based on data owner specified conditions and possible penalties in case of data user violation.

D. ACCESS

The data sets that include personal data must be de-identified or anonymized before sharing to ensure that individuals'

interests are not compromised by providing access to their information. Besides, Hyperledger Fabric supports private data and private communication, which could be desirable for the data owners who do not want to expose the metadata associated with their data to all system users. They can exploit this feature and share their data assets info with their interested parties. Access to the data provenance can also be limited through customized policies in the smart contracts (Chaincode) [35]. For example, data users can send requests to data owners to access reading the data set provenance.

E. IDENTITY MANAGEMENT

In Hyperledger Fabric as a permissioned blockchain, a digital identity encapsulated in an X.509 digital certificate must be issued for every actor and user before interacting with the blockchain. This identity is essential to determine the correct permissions over resources and access to information users have in a blockchain network. A digital identity can include additional attributes to specify the person or organization holding that identity. These attributes help data owners to identify those attempting to get access to their data assets.

F. AUDITING

Auditing is one of the primary purposes of introducing blockchain to implement a data trust framework. Blockchain enables us to audit every process and interaction in the system. In the context of data sharing, blockchain provides an immutable audit trail of data modifications, access requests, access grants and revocations.

Data owners are able to query the history of transactions regarding access requests and modifications on access permissions on their data assets. Data users are also able to audit data assets origin and history of updates on the data sets.

Furthermore, monitoring the immutable log of data transactions to automatically generate audit trails and record any data breach attempts facilitates detecting possible threats [30].

G. ACCOUNTABILITY

Our proposed data trust framework with exploiting blockchain features increases transparency with respect to quality, access and usage of data.

Once the data owners accept access requests to their data sets, data users become accountable for using the data under their control. The access enforcement point is an off-chain process that provides access to data sets based on the specified access limits represented by the data owner. The monitoring unit is responsible for tracking data users' actions. Monitoring the immutable log of activities generated by data users and detecting any violation or misuse can lead to penalty and recording permanently on the history of the data usage on the ledger.

H. IMPACT

Identifying value, use, and misuse of data requires invoking data experts, who can participate as verifier nodes in the blockchain. The data owners can specify their data value and access conditions on their data assets, and they can be programmed into smart contracts. The penalty calculation must also be included in the consent management smart contract. The methods of measuring the impact of the data are out of the scope of this paper.

X. CONCLUSION

Current systems are limited in providing a practical and transparent approach for data sharing due to the lack of trust in both parties. In this paper, we introduced an end-to-end data trust framework using permissioned blockchain. Our presented framework assesses the quality of input data using a novel trust model, including the data owner's reputation, endorsements and confidence in provided data. Therefore, the data users ensure that the quality of available data sets has been adaptively examined and updated. In our represented framework, data owners also benefit from secure, transparent, and automatic access management handled by smart contracts. Data owners have complete control over their data assets, and they are the only actors in the system who can regulate access permissions without relying on third parties. Data owners can also monitor and trace access regulations and modifications on their data assets by exploiting blockchain's provenance and audibility features. Furthermore, valuable logs can be extracted from the ledger to present a transparent view of the system, identify suspicious requests, and detect protocol breaches leading to discovering possible threats. Evaluation results indicate the system's effectiveness in handling a large number of transactions for writing, updating, and querying trust parameters value.

As a future direction, we are looking toward improving the credibility of our framework by adding incentives to encourage honest participation of the users by adding endorsements and ratings. Moreover, identifying invalid assessments because of inputs from disruptive users is another important step to enhance the solution.

REFERENCES

[1] K. O'hara, "Data trusts: Ethics, architecture and governance for trustworthy data stewardship," Univ. Southampton, Southampton, U.K., Tech. Rep., 2019.

[2] A. Alsaad, K. O'Hara, and L. Carr, "Institutional repositories as a data trust infrastructure," in *Proc. Companion Publication 10th ACM Conf. Web Sci.*, Jun. 2019, pp. 1–4.

[3] S. Rouhani and R. Deters, "Security, performance, and applications of smart contracts: A systematic survey," *IEEE Access*, vol. 7, pp. 50759–50779, 2019.

[4] J.-H. Cho, K. Chan, and S. Adali, "A survey on trust modeling," *ACM Comput. Surv.*, vol. 48, no. 2, pp. 1–40, Nov. 2015.

[5] Z. Yan and S. Holtmanns, "Trust modeling and management: From social trust to digital trust," in *Computer Security, Privacy, and Politics: Current Issues, Challenges, and Solutions*. Hershey, PA, USA: IGI Global, 2008, pp. 290–323.

[6] S. Stalla-Bourdillon, G. Thuermer, J. Walker, L. Carmichael, and E. Simperl, "Data protection by design: Building the foundations of trustworthy data sharing," *Data Policy*, vol. 2, pp. 1–10, Jan. 2020.

[7] G. S. Nelson, "Practical implications of sharing data: A primer on data privacy, anonymization, and de-identification," in *Proc. SAS Global Forum*, 2015, pp. 1–23.

[8] S. Xuan, L. Zheng, I. Chung, W. Wang, D. Man, X. Du, W. Yang, and M. Guizani, "An incentive mechanism for data sharing based on blockchain with smart contracts," *Comput. Electr. Eng.*, vol. 83, May 2020, Art. no. 106587.

[9] A. K. Shrestha and J. Vassileva, "User data sharing frameworks: A blockchain-based incentive solution," in *Proc. IEEE 10th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, Oct. 2019, pp. 0360–0366.

[10] M. Shen, J. Duan, L. Zhu, J. Zhang, X. Du, and M. Guizani, "Blockchain-based incentives for secure and collaborative data sharing in multiple clouds," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 1229–1241, Jun. 2020.

[11] W. Chen, Y. Chen, X. Chen, and Z. Zheng, "Toward secure data sharing for the IoV: A quality-driven incentive mechanism with on-chain and off-chain guarantees," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 1625–1640, Mar. 2020.

[12] L. Zhu, H. Dong, M. Shen, and K. Gai, "An incentive mechanism using shapley value for blockchain-based medical data sharing," in *Proc. IEEE 5th Int. Conf. Big Data Secur. Cloud (BigDataSecurity) Int. Conf. High Perform. Smart Comput., (HPSC) IEEE Int. Conf. Intell. Data Secur. (IDS)*, May 2019, pp. 113–118.

[13] Z. Su, Y. Wang, Q. Xu, and N. Zhang, "LVBS: Lightweight vehicular blockchain for secure data sharing in disaster rescue," *IEEE Trans. Dependable Secure Comput.*, early access, Mar. 13, 2020, doi: 10.1109/TDSC.2020.2980255.

[14] R. Casado-Vara, F. de la Prieta, J. Prieto, and J. M. Corchado, "Blockchain framework for IoT data quality via edge computing," in *Proc. 1st Workshop Blockchain-Enabled Netw. Sensor Syst.*, 2018, pp. 19–24.

[15] X. Zheng, R. R. Mukkamala, R. Vatrapu, and J. Ordieres-Mere, "Blockchain-based personal health data sharing system using cloud storage," in *Proc. IEEE 20th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Sep. 2018, pp. 1–6.

[16] C. Cappiello, M. Comuzzi, F. Daniel, and G. Meroni, "Data quality control in blockchain applications," in *Proc. Int. Conf. Bus. Process Manage.* Vienna, Austria: Springer, 2019, pp. 166–181.

[17] J. Huang, L. Kong, H.-N. Dai, W. Ding, L. Cheng, G. Chen, X. Jin, and P. Zeng, "Blockchain-based mobile crowd sensing in industrial systems," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6553–6563, Oct. 2020.

[18] J. An, J. Cheng, X. Gui, W. Zhang, D. Liang, R. Gui, L. Jiang, and D. Liao, "A lightweight blockchain-based model for data quality assessment in crowdsensing," *IEEE Trans. Comput. Social Syst.*, vol. 7, no. 1, pp. 84–97, Feb. 2020.

[19] J. Wang, M. Li, Y. He, H. Li, K. Xiao, and C. Wang, "A blockchain based privacy-preserving incentive mechanism in crowdsensing applications," *IEEE Access*, vol. 6, pp. 17545–17556, 2018.

[20] L. Zavolokina, F. Spychiger, C. Tessone, and G. Schwabe, "Incentivizing data quality in blockchains for inter-organizational networks–learning from the digital car dossier," Univ. Zurich, Zürich, Switzerland, Tech. Rep., 2018.

[21] V. Dedeoglu, R. Jurdak, G. D. Putra, A. Dorri, and S. S. Kanhere, "A trust architecture for blockchain in IoT," in *Proc. 16th EAI Int. Conf. Mobile Ubiquitous Syst., Comput., Netw. Services*, 2019, pp. 190–199.

[22] B. Shala, U. Trick, A. Lehmann, B. Ghita, and S. Shiaeles, "Blockchain and trust for secure, end-user-based and decentralized IoT service provision," *IEEE Access*, vol. 8, pp. 119961–119979, 2020.

[23] L. Yue, H. Junqin, Q. Shengzhi, and W. Ruijin, "Big data model of security sharing based on blockchain," in *Proc. 3rd Int. Conf. Big Data Comput. Commun. (BIGCOM)*, Aug. 2017, pp. 117–121.

[24] A. Brandão, H. S. Mamede, and R. Gonçalves, "Systematic review of the literature, research on blockchain technology as support to the trust model proposed applied to smart places," in *Proc. World Conf. Inf. Syst. Technol.* Naples, Italy: Springer, 2018, pp. 1163–1174.

[25] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.

[26] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1495–1505, Apr. 2019.

[27] T.-H. Kim, R. Goyat, M. K. Rai, G. Kumar, W. J. Buchanan, R. Saha, and R. Thomas, "A novel trust evaluation process for secure localization using a decentralized blockchain in wireless sensor networks," *IEEE Access*, vol. 7, pp. 184133–184144, 2019.

[28] P. Kochovski, S. Gec, V. Stankovski, M. Bajec, and P. D. Drobintsev, "Trust management in a blockchain based fog computing platform with trustless smart oracles," *Future Gener. Comput. Syst.*, vol. 101, pp. 747–759, Dec. 2019.

[29] P. Wei, D. Wang, Y. Zhao, S. K. S. Tyagi, and N. Kumar, "Blockchain data-based cloud data integrity protection mechanism," *Future Gener. Comput. Syst.*, vol. 102, pp. 902–911, Jan. 2020.

[30] O. Choudhury, I. Sylla, N. Fairoza, and A. Das, "A blockchain framework for ensuring data quality in multi-organizational clinical trials," in *Proc. IEEE Int. Conf. Healthcare Informat. (ICHI)*, Jun. 2019, pp. 1–9.

[31] L. Hang, I. Ullah, and D.-H. Kim, "A secure fish farm platform based on blockchain for agriculture data integrity," *Comput. Electron. Agricult.*, vol. 170, Mar. 2020, Art. no. 105251.

[32] C. A. Kushida, D. A. Nichols, R. Jadrnicek, R. Miller, J. K. Walsh, and K. Griffin, "Strategies for de-identification and anonymization of electronic health record data for use in multicenter research studies," *Med. Care*, vol. 50, pp. S82–S101, Jul. 2012.

[33] S. Rouhani and R. Deters, "Blockchain based access control systems: State of the art and challenges," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell.*, Oct. 2019, pp. 423–428.

[34] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, and S. Muralidharan, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proc. 13th EuroSys Conf.*, Apr. 2018, pp. 1–15.

[35] S. Rouhani, R. Belchior, R. S. Cruz, and R. Deters, "Distributed attribute-based access control system using a permissioned blockchain," 2020, *arXiv:2006.04384*. [Online]. Available: https://arxiv.org/abs/2006.04384

SARA ROUHANI is currently pursuing the Ph.D. degree in computer science with the University of Saskatchewan, Canada, under the supervision of Prof. Ralph Deters. She is a Research Assistant with the Multi-User Adaptive Distributed Mobile and Ubiquitous Computing (MADMUC) Laboratory led by Prof. R. Deters and Prof. J. Vassileva. Her research interests include distributed systems, blockchain, smart contracts, and data trust.

RALPH DETERS (Member, IEEE) received the Ph.D. degree from the Federal Armed Forces University in Munich, Germany, in 1998. He joined the University of Saskatchewan, as a Research Associate, in 1998. He is currently a Full Professor with the Department of Computer Science, University of Saskatchewan. His research areas focus on distributed ledger technology, the IoT, and cloud/edge computing.