Privacy-Preserving Data Sharing and Data Subject Control in a Data-Driven Economy: A Blockchain Approach Using Hyperledger Fabric

S.S. Ogar, O.Y. Ogunlola, O.O. Abereowo, O.D. Alowolodu, B.K. Alese
Department of Cyber Security
Federal University of Technology
Akure, Nigeria

Abstract

Today, we live in an era where data is invaluable, and an incomprehensible amount of data is created daily. The companies that have gained a competitive edge are the ones that are already embracing business intelligence and data utilization, no matter the industry. In the case of Google, users' data has helped ensure customized services. The collaboration and sharing of these data among enterprises have emerged as a significant privacy and economic concern. The concerns include the risk of data breaches, legal consequences, and lack of incentive mechanisms for Data Subjects. Using Google Inc. as a case study, this paper will address privacy issues in data-sharing using blockchain technology- Hyperledger Fabric. Hyperledger Fabric is an open-source enterprise-grade permissioned distributed ledger technology (DLT) platform designed for use in enterprise contexts that delivers some key differentiating capabilities over other popular distributed ledger or blockchain platforms [1]. This solution will ensure that user data collected by Google Inc. based on the agreed privacy and data-sharing policy between the user and the company is adhered to in a permissioned and trusted environment. Users will have the advantage of data privacy and compensation through digital tokens as they share their data with interested data consumers, thus becoming active actors that stand to profit from the data-driven economy. The Data Subject is an end user whose personal data can be collected. In this paper, User and Data Subject are used interchangeably.

Keywords: Data Subject, Data Consumer, Digital Token, Blockchain, Hyperledger Fabric.

1. Introduction

Data is essential to big, medium, and small companies in understanding, delivering, and constantly improving services depending on customer preferences. We generate data that describes us from technologies we interact with daily, whether active or passive. Mobile phones and other products we buy, online or offline, serve as touchpoints. Companies can leverage these insights for product improvements, business strategy, and marketing campaigns to cater to target customers [2].

Data collection has tremendously aided technology improvement and targeted customer services and features. However, this has raised security concerns about users' privacy. Companies share users' data with third parties outside the agreed privacy and data-sharing policies between customers and the company, thereby violating the confidentiality of users' data, which needs to be maintained by companies. Individuals who utilize a service have the right to have their data stored securely, and it cannot be accessed by those who lack authorization or those outside the accepted privacy and data-sharing policies issued by the company rendering the service.

This paper adopts blockchain technology to address confidentiality concerns in data sharing and allow data subjects to have control over their data consumers. Hyperledger Fabric is a blockchain with a modular architecture that offers a unique approach to consensus that enables performance at scale while preserving privacy, unlike a public blockchain such as Ethereum. Rather than an open, permissionless system, Fabric offers a scalable and secure platform that supports private transactions and confidential contracts [1]. Sharing and preserving data can pose many risks, such as data breaches and potential shortcomings in security measures. Hyperledger Fabric ensures the privacy of individuals' personal information. It also ensures that only authorized identities can access the data using smart contracts, and these contracts can enable specific capabilities for different identities within the system. Therefore, there is a strong requirement for a decentralized approach to sharing and storing data, where users have confidence in the security and privacy of their data, are included in the data economy monetization, and all parties involved can see a holistic view of all transactions and interactions.

2. Related Works

Existing methods for data sharing often lack one or more critical aspects: they either compromise

 | DOI: 10.20533/jitst.2046.3723.2024.0105

privacy, are computationally expensive, and fail to compensate data subjects in the data-driven economy, or they neglect the need for a data transaction market system to enable data tokenization.

A framework [3] was developed for cross-domain image sharing on blockchain to eliminate intermediary entities. However, the framework's drawback is its inefficiency in handling the complexity of privacy in an unclear regulatory environment. In this use case, Hyperledger fabric is the most suitable. Hyperledger Fabric supports a network of known participants operating under a regulated environment with a modular architecture that handles privacy.

A blockchain-based application was proposed by [4]. This stores the unique identifier of medical data using a proof of work consensus algorithm. However, the proof of work consensus mechanism is computationally expensive and not economy-driven.

[5] proposed a solution that analyzed different blockchain frameworks and identified the Hyperledger Fabric framework as the most suitable for preserving the privacy and security of patients' records in a healthcare record system. However, the solution did not consider data tokenization.

[6] discussed and identified data pricing as one of the key issues of the digital economy for establishing a data-transaction market system. This paper emphasized data tokenization as a key issue in data transactions. With the popularity of mobile terminal devices, an ever-increasing amount of end-to-end personal information or personal data is being produced, endowed with specific property attributes.

PrivySeC [7] is a privacy-preserving data-sharing framework for the IoT ecosystem. PrivySeC enables the data owners to select and share Personally Identifiable Information (PII) and non-Personally Identifiable Information (non-PII) with authorized parties as per an unforgettable mutual contract. However, this framework did not inherently consider data tokenization.

3. System Design

The feasibility of eliminating competition in a data-driven economy, even though it has been proposed that data sharing is inevitable, is quite infeasible. Google Inc., a major big data company, generated $279.8 billion in annual revenue in 2022 while ensuring personalized services and improving the quality of services to their users [8]. Therefore, there is a need to regulate a data-driven economy to include the data subject, the provider of services (i.e. Google), and other interested data consumers- any third party interested in sharing the wealth of data.

Using Google Inc. as a case study, Google's blockchain operator created the network on Hyperledger Fabric by defining the network configuration, which is contained in a "configuration block." The user identity is configured within the network so that individuals who have accepted the privacy policy can access Google Inc.'s services. This network comprises two major components: the organizations and the assets. The user data is treated as an asset on the network. Google has default access to the confidential data of the user based on the agreed privacy policy, which allows it to use the data to improve the services it renders to the user. Users have complete control over which third-party organization they want to authorize to access their confidential data. A utility token is implemented on the network to ensure data monetization by any user willing to share his data with any third-party organization.

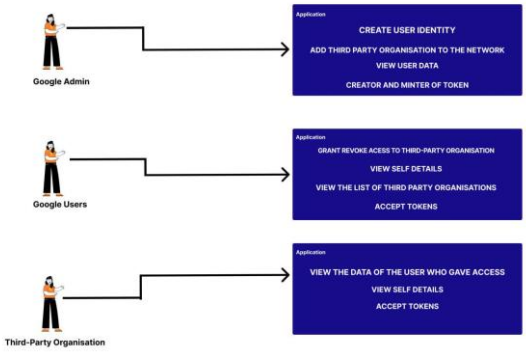

Figure 1. Use cases of the 3 Basic Actors on the Network

A system architecture is depicted in Figure 2. The architecture explains the sample network in detail. The architecture illustrates a network on Hyperledger Fabric that is comprised of two organizations. The network denotes each organization by a peer, with an illustration of a peer with authority to access private data and one not on the same network. The blockchain operator sets up the initial configuration of the network and the necessary credentials for the network participants.

For the given scenario, a single channel named 'my Channel,' two organizations (Google Inc. and a third-party organization), and an asset (user data) is sufficient. Google Inc. creates a user identity within the organization on the network to manage the user's assets, which accepts Google's privacy policy. The users' assets are accessible to Google but inaccessible to other-party organizations except if the users give the access. RAFT consensus protocol is used as the consensus protocol on the network since every application component is pluggable. Within each organization, we can have different roles and identities defined or configured, such as admin and user identities. The POSTMAN collection exposes the REST APIs to simulate the frontend interface. The ability to organize and share API requests makes

 | DOI: 10.20533/jitst.2046.3723.2024.0105

Postman Collections a valuable tool for API development and testing. However, more organizations can be successfully added when network runs and joins with the same Channel. Fabric provides support for two databases, LevelDB and CouchDB. LevelDB is the default state database embedded in the peer node.

LevelDB stores chain code data as simple key-value pairs and only supports key, key range, and composite key queries. CouchDB is an optional, alternate state database that allows you to model data on the ledger as JSON and issue rich queries against data values rather than the keys. The CouchDB support also allows you to deploy indexes with your chain code to make queries more efficient and enable you to query large datasets [9]. In this Network, CouchDB is supported. The ledger consists of two states: the World State and the Blockchain. The world state is a database that holds the current values of a set of ledgers. The blockchain is a transaction log that records all the changes that have resulted in the current world state [8]. Figure 3 shows that we have some personal data described herein in the integer file. These are the initial data when the Network is up.

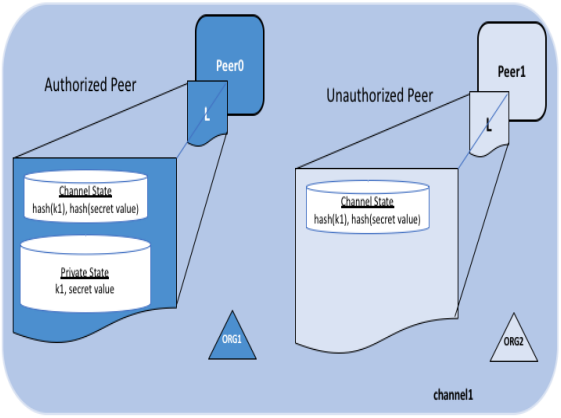

Figure 2. System Architecture

```
func (s *SmartContract) InitLedger(ctx contractapi.TransactionContextInterface) error {
    cars := []Car{
        Car{Make: "Toyota", Model: "Prius", Colour: "blue", Owner: "Tomoko"},
        Car{Make: "Ford", Model: "Mustang", Colour: "red", Owner: "Brad"},
        Car{Make: "Hyundai", Model: "Tucson", Colour: "green", Owner: "Jin Soo"},
        Car{Make: "Volkswagen", Model: "Passat", Colour: "yellow", Owner: "Max"},
        Car{Make: "Tesla", Model: "S", Colour: "black", Owner: "Adriana"},
        Car{Make: "Peugeot", Model: "205", Colour: "purple", Owner: "Michel"},
        Car{Make: "Chery", Model: "S22L", Colour: "white", Owner: "Aarav"},
        Car{Make: "Fiat", Model: "Punto", Colour: "violet", Owner: "Pari"},
        Car{Make: "Tata", Model: "Nano", Colour: "indigo", Owner: "Valeria"},
        Car{Make: "Holden", Model: "Barina", Colour: "brown", Owner: "Shotaro"},
    }
```

Figure 3. These are the initial sample data when the network is up

3.1. A Private Collection

Data is shared through a peer-to-peer gossip protocol and is only visible to authorized organizations. The private data is stored in a state database of authorized peers and can be accessed using a chain code. The ordering service is not part of this process and does not have access to the private information. To facilitate cross-organization communication, anchor peers must be established on the Channel, and the CORE_ PEER _GOSSIP _EXTERNALENDPOINT must be configured for each peer through the gossip protocol.

In this network sample, *peer0.org1.example.com* and *peer0.org2.example.com* are the anchor peers in org1 and org2, respectively. Both are configured in the GitHub repository: */BasicNetwork-2.0/artifacts/docker-compose.yaml*. Every peer on the channel endorses, orders, and records the transaction in their ledgers. The transaction hash serves as proof and is used to check the validity of the state. It can also be used for auditing purposes. In this application, there can be, at most, *nC1* private data collections where n is the number of organizations on the Network.

3.2. Implementation Example

Let us consider the scenario where there are three organizations named Org. 1 (Google Inc.), Org 2, and Org 3 (third-party organizations). The Number of organizations, n=3.

Therefore, the total number of private collections:

$$PC=nC1 \quad (1)$$

Organization 1 with default access to data on the network, r = 1

$$PC= n!/(n-r)!\ r \quad (2)$$

$$PC= 3!/(3-1)!\ 1! \quad (3)$$

$$PC=3 \quad (4)$$

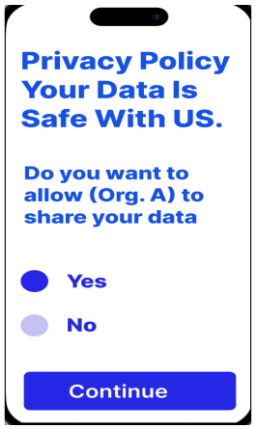

Figure 4. The Figma design page

 | DOI: 10.20533/jitst.2046.3723.2024.0105

Therefore, the Network can accommodate three private collections (see Figure 4):

- *Org (1) PrivateCollection* will contain user data, which is authorized only for Organization 1.
- *Org (1, 2) PrivateCollection* will also contain user data authorized by the data subject by accepting the policy to share with Org 2.
- Lastly, *Org (1, 3) PrivateCollection*, which is private to both Organization 1 and Organization 3, will also contain user data authorized by the user to be shared with Org 3.

3.3. UTXO Model

Data must be treated as an asset and Data Subject as the owner. The Unspent Transaction Output (UTXO) model is one of the approaches to defining assets in Hyperledger Fabric, where account balances are encoded into past transaction records. With this model, the UTXO state corresponds to a unique KVS entry that is created once (the coin state is “unspent”) and destroyed once (the coin state is “spent”). Every state may be seen as a KVS entry with logical version 0 after creation; when it is destroyed again, it receives version 1. There should not be any concurrent updates to such entries (e.g., attempting to update a coin state in different ways amounts to double-spending the coin) [10].

In this case, Google is assumed to be the organization with a central banker role. A Google admin can mint new “PrivCy” tokens for third parties, which are data consumers. We give value to this token by backing it up with a tangible asset such as USD or gold. A Google admin uses the Mint function to create a UTXO representing “PrivCy” tokens. The Mint function reads the certificate information of the interested data consumer and assigns the UTXO ownership to the data consumer ID. PrivCy token will be a means of exchange of value amongst the network participants.

4. Implementation

The experiment demonstrates the steps in creating a user and a third-party organization. To create a user, they must first accept Google's privacy policy. Then, the network admin uses the AdminContract to create a user identity under the Google organization. To do this, the admin connects to the network using their certificate and creates a transaction that adds the user to the ledger and the user's identity to the blockchain network. After the user has been created, the server generates a temporary password for the user to log into the network.

Regarding creating a third-party organization, the third party provides details to the admin, who then connects to the Network and adds the organization as an identity to the blockchain network. The organization's credentials are stored in CouchDB. Users and third-party organizations now added as identities in the blockchain network can connect to the network using their respective certificates for future interactions.

4.1. System interface

The interface below shows a client/user with the name OGAR registered in Org1, and a token is issued for authentication and verification by JSON Web Token. Username Ogar stored his data on the network using the following details. The network ensures the data is only accessible to Org1 while the hash of the data is available across all organization peers on the Network.

In Figure 5, a client/user with the name OGAR is registered in Org1, and a token is issued for authentication and verification by JSON Web Token

Figure 6: Username Ogar stored its data on the network with the following details. The network ensures the data is only accessible to Org1 while the hash of the data is available across all organization peers on the Network.

Figure 7 represents the username Eve registered on Org2. A token is generated for authentication and verification.

4.2. Comparing the hash in the database of both Organizations

Comparing the hash of Figures 8 and 9, the hash computed is the same across all databases on the network.

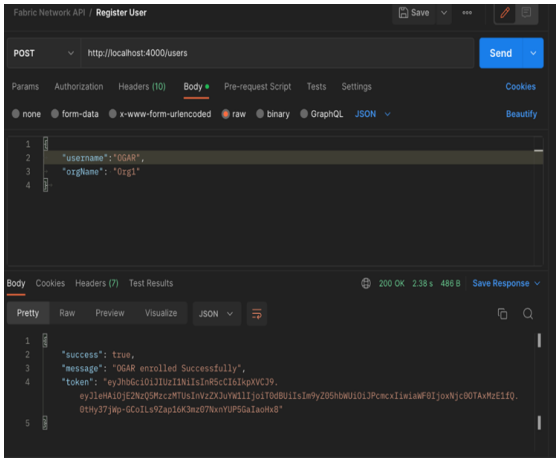

Figure 5. A user with the name OGAR is registered in Org1 and a token is issued for authentication and verification by Json Web Token

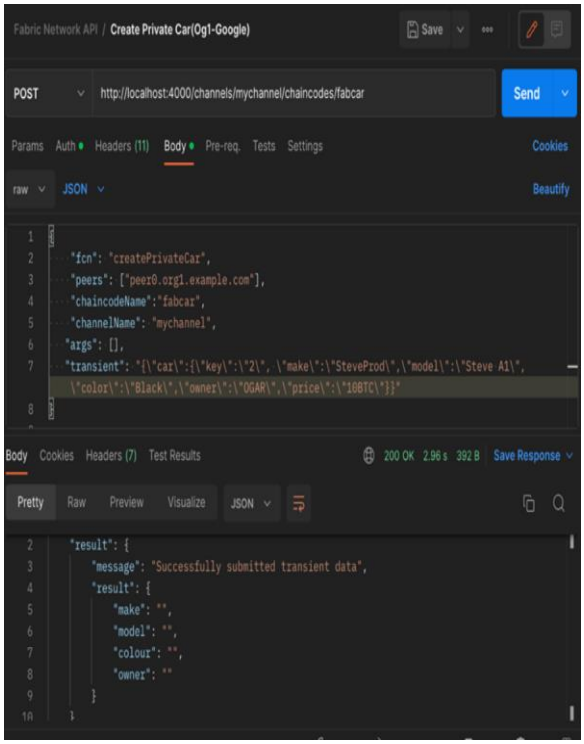

Figure 6. Username Ogar stored its data on the network with the following details

The network ensures the data is only accessible to Org1 while the hash of the data is available across all organization peers on the network.

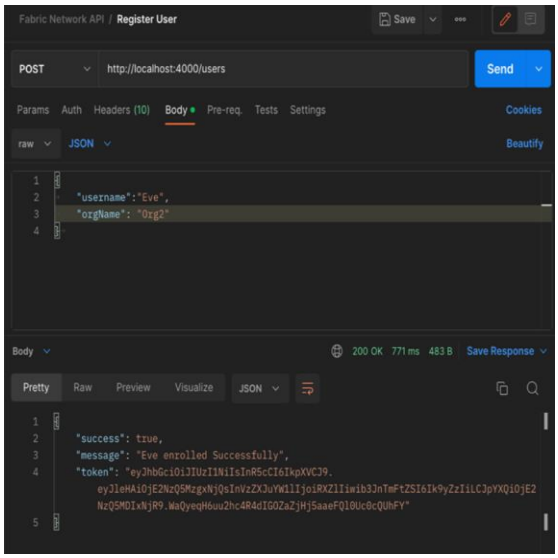

Figure 7. Represents username, Eve registered on Org2. A token is generated for authentication and verification

5. Conclusion

Hyperledger Fabric is a blockchain framework that is designed for enterprise use cases and has features

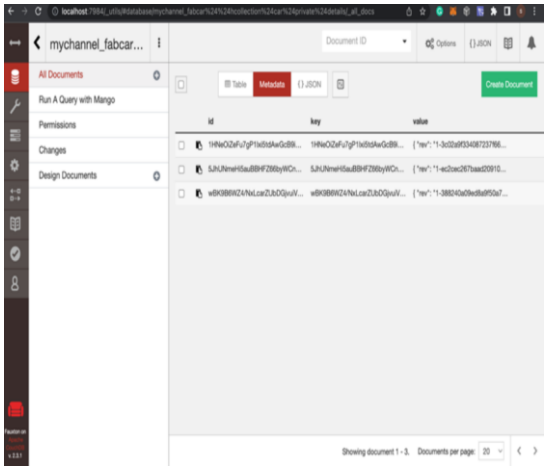

Figure 8. Database of Org 1

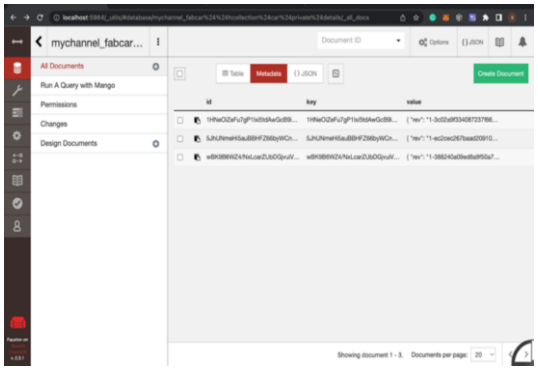

Figure 9. Database of Org 2

such as policies, smart contracts, and secure identities [11]. It allows for interoperability among multiple organizations. To improve the solution for production-grade applications, the network calls should be done over HTTPS to provide transport-level security, and better UI/UX strategies can be applied to enhance the user experience. This paper addresses privacy on an organizational level. To ensure privacy and scalability on the user level, a different approach from the implemented privacy data collection feature provided by the Hyperledger Fabric framework is required where privacy is considered for $n * r$. Where n is the number of Google users, and r is the number of data consumers. Overall, Hyperledger Fabric provides a secure and confidential platform for storing and managing data within a private data collection.

6. References

[1] Hyperledger Fabric. (2020). Open, proven, enterprise-grade DLT. Hyperledger. https://www. lfdecentralizedtrust.org/blog/2020/01/30/welcome-hyperledger-fabric-2-0-enterprise-dlt-for-production (Access Date: 9 June 2023).

[2] Exploding Topics. (2023). Amount of data created

 | DOI: 10.20533/jitst.2046.3723.2024.0105

daily. Exploding Topics; https://explodingtopics.com/blog/data-generated-per-day (Access Date: 13 June 2023).

[3] Patel, V. (2019). A framework for secure and decentralized sharing of medical imaging data via blockchain consensus. Health Informatics Journal, 25(4), 1398–141110.1177/1460458219833091.

[4] Sharma, P., Namasudra, S., Crespo, R. G., Parra-Fuente, J., and Trivedi, M. C. (2023). EHDHE: Enhancing the security of healthcare documents in IoT-enabled digital healthcare ecosystems using blockchain. Information Sciences, 629, 703–718.

[5] Abutaleb, R. A., Alqahtany, S. S., and Syed, T. A. (2023). Integrity and privacy-aware, patient-centric health record access control framework using a blockchain. Applied Sciences, 13(2), 1028.

[6] Xu, J., Hong, N., Xu, Z., Zhao, Z., Wu, C., Kuang, K., Wang, J., Zhu, M., Zhou, J., Ren, K., Yang, X., Pei, J., and Shum, H. (2023). Data-driven learning for data rights, data pricing, and privacy computing. Engineering, 25, 66–76.

[7] Makhdoom, I., Abolhasan, M., Lipman, J., Piccardi, M., and Franklin, D. (2024). PrivySeC: A secure and privacy-compliant distributed framework for personal data sharing in IoT ecosystems. Blockchain: Research and Applications, 5(4), 100220.

[8] Statista. (2023). Revenue of Google from 1st quarter 2008 to 1st quarter 2023. Statista; https://www.statista.com/statistics/267606/quarterly-revenue-of-google/ (Access Date: 30 April 2023).

[9] Hyperledger Fabric. (2023). Using CouchDB. Hyperledger Fabric; https://hyperledger-fabric.readthedocs.io/en/latest/couchdb_tutorial.html?highlight=CouchDB (Access Date: 13 June 2023).

[10] Hyperledger Fabric. (2023). The ledger', Hyperledger Fabric; https://hyperledger-fabric.readthedocs.io/en/latest/ledger/ledger.html?highlight=world%20state#world-state (Access Date: 12 June 2023).

[11] Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., Singh, G., Smith, K., Sorniotti, A., Stathakopoulou, C., Vukolić, M., Weed Cocco, S., and Yellick, J. (2018). Hyperledger fabric: A distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference (EuroSys '18) (Article 30). Association for Computing Machinery.

 | DOI: 10.20533/jitst.2046.3723.2024.0105