SAID: ECC-Based Secure Authentication and Incentive Distribution Mechanism for Blockchain-Enabled Data Sharing System

Muhammad Rizwan[1], Muhammad Noman Sohail[2], Alia Asheralieva[1], Adeel Anjum[2], Pelin Angin[3]
[1]*Department of Computer Science and Engineering*, *Southern University of Science and Technology*, Shenzhen, China,
e-mail:{rizwanramay@gmail.com, aasheralieva@gmail.com}
[2]*Department of Computer Science*, *COMSATS University*, Islamabad, Pakistan,
e-mail:{nomansohail13@gmail.com, adeel.anjum@comsats.edu.pk}
[3]*Department of Computer Engineering*, *Middle East Technical University*, Ankara, Turkey, e-mail:{pangin@ceng.metu.edu.tr}

2021 IEEE International Conference on Blockchain (Blockchain) | 978-1-6654-1760-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/BLOCKCHAIN53845.2021.00080

***Abstract*—The fast digital transformation the world experienced in the past few decades has created a multitude of information resources distributed over the Internet and accessed by many different entities. The sharing of these resources needs to be secure and trustworthy to protect the privacy of data and data providers. Privacy and reliability are major concerns especially in blockchain-based data storage systems, as data are held by different owners. Each data provider tries to increase its share of profit by collaborating with other data providers. To ensure secure data sharing in such a decentralized environment, authentication and verification of shared data is needed. This paper proposes a mutual authentication mechanism based on Elliptic Curve Cryptography (ECC) using hash-based message authentication code (HMAC) to provide anonymity and integrity during the communication process in blockchain-based data sharing. To evaluate the fairness of profit sharing based on data providers' contributions and data utility, we propose the use of the Shapley value techniques. The performed security analysis demonstrates that our proposed approach is resilient against replay, secret disclosure, and traceability attacks.**

***Index Terms*—Elliptic Curve Cryptography, Consortium Blockchain, Authentication, Shapley value, Incentive Distribution**

I. Introduction

The big data era has created high *volumes* of data generated by a *variety* of digital data resources at very high *velocities*. Perhaps the most important property of big data is the *value* it provides to users, which can only be achieved if we can ensure high *accuracy*. The aggregation of data from multiple data providers in a centralized platform raises security and privacy concerns, as well as reliability problems due to the single point of failure. The integration of data items collected from various sources may be vulnerable to integrity breaches as well as authenticity issues. Due to these concerns, there is a need for frameworks and models to evaluate and increase the trustworthiness of data generated and disseminated by various systems. The evaluation of data trustworthiness is a challenging task, because the data is provided by untrusted sources holding microdata about entities having overlapping attributes. Certain data providers may breach the integrity of data using machine learning techniques to insert missing values in the datasets. The integrity of data may also be breached during the transmission process, therefore verification mechanisms for denying false data need to be integrated into the data sharing frameworks. Such verification of data prevents untrustful data providers from participating in the data mashup process. Data providers in a decentralized data sharing system need monetary incentives to keep supporting the storage and sharing of large volumes of valuable data. For such systems, the monetary value for sharing data can be determined through dynamic game theory auction mechanisms [1] by evaluating data providers' participation and the utility of their provided data.

Blockchain is an emerging technology that offers trustworthy and secure interactions between data providers and data consumers when sharing data. It also facilitates auditing of data usage with provenance traceability and transparency. By adopting blockchain in data sharing systems, only authorized data consumers can access data from decentralized data sources [2].

We propose a secure data sharing scheme using blockchain among different participants, including data providers and data consumers, hence achieving the confidentiality and integrity of data during transmission between different sources in a decentralized data sharing system. The proposed model assesses the data providers' trustworthiness and the trustfulness of their provided data. To achieve these objectives, both data providers and consumers' authentication is essential to provide authenticity and integrity. Authentication is performed using Elliptic Curve Cryptography (ECC) to achieve a high level of security and performance in the system. The Intuition of using ECC for designing protocol because it provide more security then other schemes with less computational time. For example, a 256 bit ECC key provides equal security that a 3072 bit RSA key provides. The smaller key size used in ECC appealing for devices with limited resources. Hashed Message Authentication Code (HMAC) is used to provide integrity during data transmission. The main objective to use

978-1-6654-1760-0/21/$31.00 ©2021 IEEE
DOI 10.1109/Blockchain53845.2021.00080

HMAC is its collisions resistance nature as compared to other hashing schemes. Moreover, it simplify the keys management during data sharing process.

Once the data provider is authenticated, it becomes a part of the blockchain system and can participate in the bidding process for sharing its data. Consumers need to be registered to request data and access the requested data from data providers. The profit share of participants including data providers and collaboration platforms is distributed fairly using a Shapley value approach.

The main contributions of this paper can be summarized as follows:

- We develop a security protocol to authenticate data providers and data consumers in blockchain-based data-sharing. For this purpose, we use ECC to register and authenticate the participants. Our proposed protocol reduces the time delay during the authentication process.
- We propose an integrity and anonymity preservation mechanism for data in transit, using hashed message authentication code (HMAC).
- We design a profit-sharing mechanism using Shapley value [3] that distributes incentives fairly among the data providers based on their contributions in data sharing.

II. Related Work

Many research efforts have been put into the development of authentication and authorization protocols in sensitive data sharing systems in recent years, especially with increasing security breaches taking place in a variety of platforms. Rostampour et al. [4] and Kumari et al. [5] provided an authentication protocol based on ECC. Though these proposed schemes are computationally efficient, they do not offer a solution against secret disclosure attacks and do not ensure integrity during data transmission. Kumari et al. [6] proposed a protocol that provides security against various attacks and proved efficient during the authentication process. However, it does not provide security against offline password guessing attacks, desynchronization attacks, impersonating attacks, traceability attacks and insider attacks. The problems of [6] are addressed by Safkhani et al. [7] using physically uncloneable functions (PUFs). A smart card is used to verify the identity of the respective device as the authentication parameters are setup and invoked when the request is generated. A PUF sends different responses to every incoming challenge. The behavior of PUFs make it hard for the attacker to trace back the parameters as they change with each response. It improves the security features of the protocol and enhances the privacy of data. A user-centric access policy model was proposed by Truong et al. [8] for achieving a fine-granular sharing system to manage private data with the help of smart contracts complying with GDPR legislations.

Researchers have also proposed various data-sharing models. Khokhar et al. [9] proposed a data-sharing mechanism assuming a semi-trusted Cloud Service Provider (CSP). However, in the presence of a semi-trusted third party, there arise serious privacy concerns when data from various resources are mashed up at a CSP environment. [9] does not provide fair distribution of profit among the participants, if one or more players provide the data from a coalition of data providers. So, there is a need of a robust and efficient approach that provides security and integrity of the data during the sharing process. The incentive to give the data held by multiple data providers must be fairly distributed based on their contribution in a decentralized environment. Shi et al. [10] proposed a multimedia data sharing scheme based on blockchain in which different roles are used in the system to identify the possible attacks on an entity of every role. Cryptographic protocols are used to provide the immutability property of blockchain to achieve integrity and confidentiality while sharing the data. Chi et al. [11] developed a secure framework for data sharing based on community detection in a decentralized environment. It includes three layers (blockchain, data, detection) at which the authentication and interaction of data are performed. The clients are categorized based on required data, which enhances the efficiency of the data-sharing system.

Fair profit sharing was also considered in some recent data sharing system proposals. Cong et al. [12] developed an architecture for sharing data securely between the data providers and the users in a cloud environment. This sharing mechanism is based on blockchain to achieve security and privacy of the data owners. A method was designed to calculate the fair share of profit between the multiple data providers. The method is based on a game theory approach that quantifies the share based on participation. However, the integrity of data during transit is still a major problem that needs to be solved. Li et al. [13] proposed an auction mechanism based on price incentives for a cloud environment that balances the interest between the data provider and data consumers. They explored a pricing strategy based on consumer interest and service requests simultaneously. An algorithm was proposed, which dynamically allocates the resources according to the pricing threshold using the approximate optimization technique [12]. In their model, the service providers make a pricing decision competitively with other service providers on resources requested by data consumers. The resources are only quantified when the requirements of data consumers are met. The balance of interest between the data providers and consumers can be resolved more precisely by the method proposed by Song et al. [14]. They proposed a combinational auction mechanism in which the problem of allocation of resources is solved during different time periods of resource requests. Solving this problem proposes two pricing curves including the staircase and continuous curves that relieve the resource request congestion in the cloud environment. Khan et al. [15] used a game theoretic approach to model the incentive distribution among resource sharing IoT devices. A non-cooperative game theory approach was used, as the participants maximize their share of profit independently [16]. Federated learning was used to optimize the resource sharing and allocation according to the domain of the system. Zhao et al. [17] developed an incentive distribution mechanism that ensures the quality of data provided by participants that

ultimately defines their profit on the basis of quality and contribution.

III. System Model and Proposed Methodology

Let N = $\{DP_1, DP_2, DP_3, ..., DP_n\}$ be the data providers, which are part of the blockchain network. A Blockchain Authenticator (BA) (gateway to blockchain network) is responsible for access control in the blockchain network. Each data provider is required to be trusted and authenticated with a BA to participate in a data auction as shown in Figure 1. The trustworthiness of the data providers is defined after successful verification and labeling as a trusted entity. In the consortium blockchain network, the set of registered data consumers DC = $\{DC_n \mid n = 1.....I\}$ send requests to the DPs. Data consumers also need to communicate with a BA to become a participant of the data sharing system.

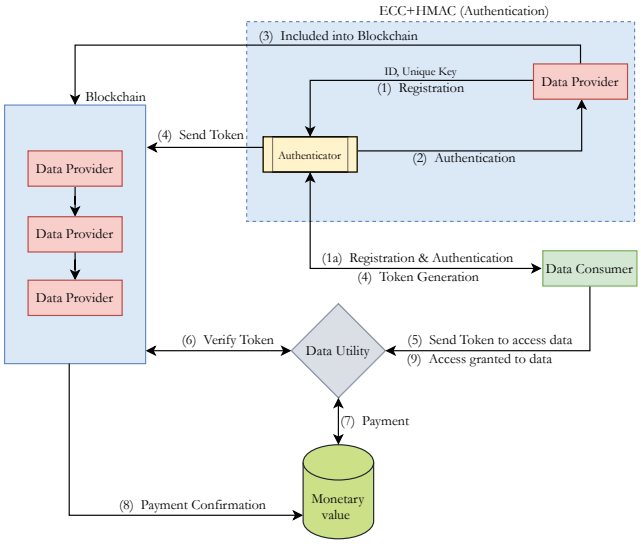

Fig. 1. Architecture of proposed framework.

The distribution of profit is managed with the help of a cooperative game theory technique, Shapley value, discussed later in the paper. Fair distribution of profit among the data providers' coalition is performed based on their participation and utility of their provided data.

During the data sharing process the security and privacy of data must be preserved. For this purpose we have consider following security requirements during protocol development:

1) **Replay Attack:** In this attack, the adversary attempts to capture the message as well as try resend the previous message sent by the authorized participant.
2) **Secret Disclosure attack:** The adversary eavesdrops the communication between legitimate participants to disclose the share secret key between them in order to access data.
3) **Traceability and Unlinkability attacks:** In these attacks, the attacker may able to trace the communication link between the participants in order to link them with transmitted message.

TABLE I
Notations used in the proposed framework

Notations	Description
EC_p	Elliptic Curve of prime order
G	Generator point of the elliptic curve
P	Large prime number
SK_{BA}	Secret key associated with Authenticator
PK_{DP},PK_{BA}	Public keys of Provider and Authenticator
T, Na	Timestamp and Nonce
DP_i	Data Provider i
DC_i	Data Consumer i
SK_{DP}	Secret key of Data Provider
K_{PB},K_{CB}	Shared Secret of Data Provider and Consumer with Authenticator, respectively
$H(.)$,E_k	Hash and Encryption Functions, respectively
R_{BA}, R_{DC}, R_{DP}	Random numbers of Authenticator, Data Consumer and Provider, respectively
TID	Transaction ID
$\|$	Concatenation
AT	Access Token generated for Consumer
DAT_{DP}, DAT_{DC}	Data Access Token for Data Provider and Consumer, respectively
A_n	Set of data attributes
C	Data Characteristics Function for each DP
S	Coalition of DPs.
$w(S)$	Worth Function generated by coalition S
φ_i	Incentive of corresponding DP from $w(S)$

A. Registration and Authentication Phase

Data providers need to be registered and authenticated to become participants in the blockchain-enabled data sharing system. In this phase, an Elliptic Curve EC_p is considered, which generates the pair of points on the curve with the point G on the curve as the generator point. With both points a and b lying on the coordinates of the curve, the condition $4a^3 + 27b^2 \neq 0$ is satisfied for the given non-singular Elliptic Curve:

$$EC_p(a, b) : y^2 = x^3 + ax + b \mod p \tag{1}$$

It is important to mention here that the Discrete Logarithm Problem (DLP) always considered in cyclic group Z_p^* that can be defined an element $\beta \in G$ and primitive element α. The DLP is to a integer x, as $1 \leq x \leq n$ such that:

$$\alpha^x \equiv \beta \, mod \, p \tag{2}$$

From equation 2 it is clearly shows that it hard to calculate the DLP when using EC.

Now let us suppose data provider DP sends a registration request to the BA. The BA chooses a secret key SK_{BA} randomly from Z_p^* and calculates its public key $PK_{BA} = SK_{BA}.G$ by point multiplication. Now the BA generates timestamp T for the current session and hashes the concatenation of PK_{BA}, DP, and T. Here, we assume that the hash function H(.) is known to data providers.

$$M_B = H(PK_{BA} \parallel DP_1 \parallel T) \tag{3}$$

The BA sends M_B, EC_p, PK_{BA}, and G to DP to calculate the shared secret (K_{PB}) with the BA. After sharing secrets, both use it for future communication. Upon receiving the parameters, DP first checks the freshness of the timestamp, and if valid, then calculates the hash of the received M_B.

$$M'_B = H(PK_{BA} \parallel DP_1 \parallel T) \tag{4}$$

Now DP chooses the secret number or key $SK_{DP} \in Z_p^*$, calculates $PK_{DP} = SK_{DP}.G$ and shares it with the BA. Both DP and BA are engaged in Diffie-Hellman Key Exchange (DHKE) to calculate shared secret key K_{PB}. After sharing the secret K_{PB}, DP is included in the blockchain network.

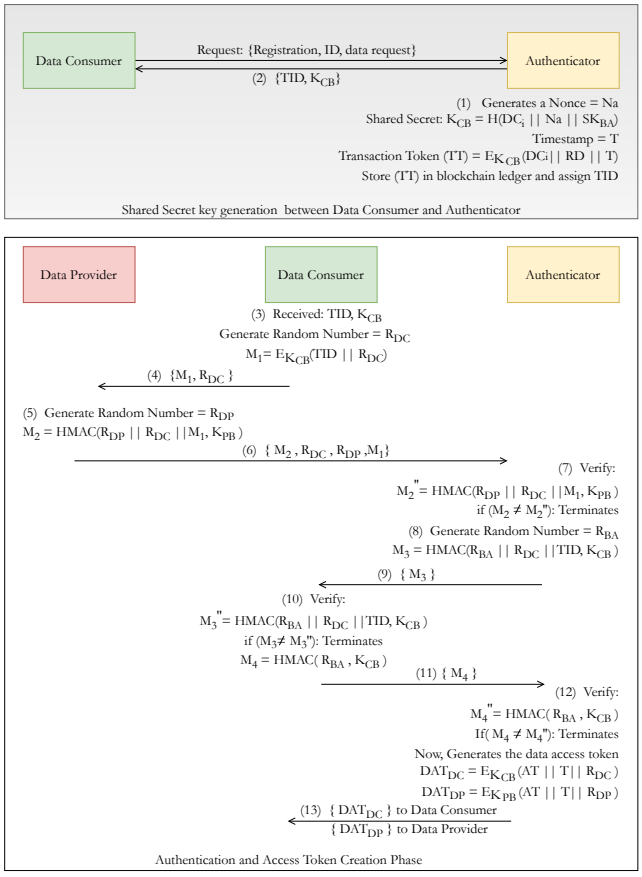

Fig. 2. Proposed Security Protocol

For a data consumer DC_i to access the data, it needs to get access token (DAT) after authentication and accepting the data provider's access policy. After receiving the request from DC_i, the BA first verifies DC_i by checking if it exists in the database or not. If it does, then it proceeds, otherwise it terminates the process. The BA generates a Nonce Na and calculates K_{CB}.

$$K_{CB} = H(DC_i \parallel Na \parallel SK_{BA}) \tag{5}$$

Here, SK_{BA} is a secret key associated with the BA and K_{CB} is the shared key between BA and DC_i, which is shared with DC_i using a secure channel. BA encrypts the ID of the consumer, requested data (RD), and T using shared key K_{CB}. The encrypted message is stored on the blockchain ledger and marked with a transaction identifier (TID). These generated parameters TID and K_{CB} are shared with DC_i to get access token (DAT). In the second step, DC_i generates a random number R_{DC} and encrypts it along with the received TID using K_{CB}. DC_i sends the encrypted message M_1 to DP. Upon receiving the message from DC_i, DP generates a random number (R_{DP}) and signs the message.

$$M_2 = HMAC(R_{DP} \parallel R_{DC} \parallel M_1, K_{PB}) \tag{6}$$

DP sends the signed message M_2 to BA where K_{PB} is the shared key between DP and BA. In the third step, BA first verifies M_2. If verification fails, then the process will be terminated, otherwise it will proceed with the protocol. Again, the BA generates a random number R_{BA} and calculates the HMAC of the message (M_3) and sends it to DC. After receiving the message M_3, DC computes the signature of the message and compares it with the received message (M_3). If verified, then DC signs the received R_{BA} using K_{CB} and sends the message (M_4) to BA. After receiving the message from the DC, the BA verifies the message (M_4). If verified, then BA generates DAT_{DC} and DAT_{DP} for DC and DP, respectively.

$$DAT_{DC} = E_{K_{CB}}(AT \parallel T \parallel R_{DC}) \tag{7}$$

$$DAT_{DP} = E_{K_{PB}}(AT \parallel T \parallel R_{DP}) \tag{8}$$

Upon receiving DAT_{DC} and DAT_{DP} both DC and DP decrypt the token (AT) using their respective shared secret keys K_{CB} and K_{PB}.

IV. Revenue Generation and Profit Calculation Process

In this section, we will briefly describe the revenue generation and distribution model among the parties. Both DC_S and DP_S are part of the consortium blockchain, which are participants of the data sharing system. It is assumed that DP_S holds trustful and reliable data of dynamic attributes. The data providers collaborate with each other to respond to the received data requests from consumers.

Each data provider holds data of multiple attributes, which quantify the contribution of each data provider in the data block shared with the consumer. Data providers try to incentivize access to their data to achieve profit on sharing the data.

A. Revenue Generation Process

The value of data depends on the characteristics of data. The impact of characteristics on the value of data is directly proportional, which largely affects the contributions of data providers. The characteristics of data contain nature, quality, complexity, and usability of data.

The request from a data consumer includes the data type which the consumer needs and the lifetime of the data. The

request is broadcast to all the data providers that are part of the blockchain network. [fontupper=]

Algorithm 1: Incentive Generation Algorithm
Input: **Data Request (R)**
Output:Incentive **W(R)**
1: Start
2: Received **Data request (R)** from **Consumer (DC)** $\rightarrow$ **R (Data Type,time lag)**
3: Coalition(S) of data providers from **N=**$\{DP_1, DP_2, DP_3, ..., DP_n,\}$ on the basis of type of data requested, $\forall |N| \leftarrow \exists\ DP_i^s = R \because S \subseteq N$
Response by coalition of data providers:
i. **Data providers N** Publish available data attributes **A =** $\{A_1, A_2, A_3, ..., A_n\}$ such that, $A_n = \sum_{i=1}^{n} a_i \quad \forall i = 1, 2, 3, ..., n$
ii. Based on characteristics **C** of data aggregated monetary value is determined **C** $\propto \sum_{i=1}^{n} DP_i^A \propto$ **Monetary value,**
$\forall$ **A =** $\{A_1, A_2, A_3, ..., A_n\}$
4: DC receives **Response** from **Coalition S** $\leftarrow$ (data attributes, monetary value)
5: If (Response== Agree**) then**
i. $\forall |S| \leftarrow \sum$ **Data(D)**$_{i|S|i=1} \rightarrow$ stored:**DB(block)**
ii. **DBR**$\leftrightarrow$**DC**: Access Granted
iii. Incentive **W(S)** is Generated
6: Return [(S)]
7: else
return to **Step 1**
8: End

When the request is received, providers form a coalition S having the data of same type as requested. Coalition S is a subset $S \subseteq N$ of all players in a game such that $|S| < |N|$. This coalition generates the revenue collectively by sharing data. Attributes of the data define the dimensions of data held by providers. Let data provider DP_1 have the data with A = $\{a_1, a_2, a_3, ..., a_n\}$, where there exist some universal attributes that have equal value for all the players. However, if any of the data providers have additional attributes, they have more weight of contribution to the coalition.

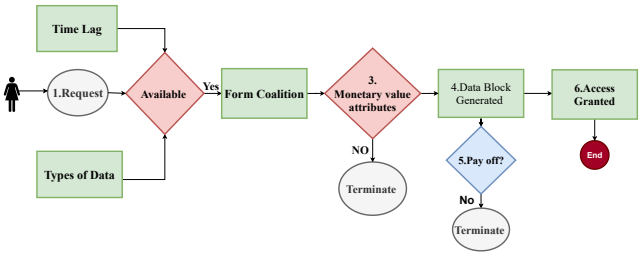

Fig. 3. Process Flow Diagram of Incentive Generation.

$A_n = \sum_{i=1}^{n} a_i \quad \forall \quad a_i, i = 1, 2, 3, 4...n$ A = $\{A_1, A_2, A_3, ..., A_n\}$ is the set of attributes of data held by each data provider of N = $\{DP_1, DP_2, DP_3, ..., DP_n\}$ such that $DP_n = A_n$. Each data provider's attribute function may have a low or high value depending on the attributes, because the data provider has data with multiple attributes that differ from other providers' data. $A = \sum_{i=1}^{n} A_i \ \forall\ A_i, i = 1, 2, 3, 4...n$ Data coming from data providers are mashed up in a single data block with available characteristics. Characteristics are the possible set of data attributes of all the providers in coalition S. The monetary value of the provided data depends on the characteristics of data. The value of data varies with the characteristics of the data.

C $\propto \sum_{i=1}^{n} DP_i^A \propto$ **Monetary value,** Where **C** is the characteristics function that is directly proportional to the total attributes of data provided by all the data providers. The response to the data consumers consists of the baseline monetary value of data, including available data attributes. The total revenue of the coalition depends on contributed data with offered attributes.

Data consumers must agree with the response from the coalition of data providers to access the data block. The incentive is paid to the coalition by the data consumer with the agreement. If the consumer does not consent to the response, the request for data will be discarded.

The contribution of data providers is evaluated with respect to their provided data. Contribution is determined using the characteristics function for all providers of the coalition. $w(s)$ is the worth or revenue function of the coalition **S** after sharing the data with the **DC**. Incentive on data is distributed among the DPs after calculating the contribution of each data provider of the coalition **S** using Shapley value.

B. Profit Calculation Process

Here we use Shapley value to calculate the contribution of each provider to determine their profit share from the revenue **W(S)** generated by coalition S. The purpose of using Shapley value is that it specifies the incentive of players according to their participation in a cooperative game. As in our data sharing framework, data providers share data with data consumers cooperatively to achieve maximum incentive. By sharing data, the coalition of data providers generate incentives, which are distributed among them based on their contribution to the coalition, which is calculated through Shapley value. Shapley value has unique properties to ensure fairness, such as efficiency, symmetry, null players etc.

Let N denote the set of participants, i.e. the data providers. Let $S \subseteq N$ denote the data provider coalition. The data provider's coalition generates revenue by sharing the data with the data consumers. Let V(N) denote the worth function of coalition S. Let $SP_i(S)$ denote the data provider i's income in the coalition S.

$$w(N) = \sum_{i \in s}^{n} SP_i^{(S)} \tag{9}$$

Each data provider's contribution is calculated through the function $w(S)$. The contribution of the data provider i in the $S \subseteq N\{i\}$ is defined by $\triangle_i(w, S)$=$w(SU\{i\}) - w(S)$. We use the Shapley value, which is an appropriate scheme for distributing the revenue among the data providers. The revenue, which comes from data consumers to data providers changes dynamically among the set of data providers. The following definition of the Shapley value is used to distribute the revenue.

$$\varphi_i(N, w) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)}{N!} [w(s \cup \{i\} - w(s)] \tag{10}$$

Where $\frac{|S|!(|N|-|S|-1)!}{|N|!}$ is the factor of weighting and S_i denotes the coalition of data providers except for the data provider i. The Shapley value scheme measures the data provider's contribution and distributes the revenue among the participants according to their contribution.

V. Security Analysis

In this section, we perform a security analysis of our proposed protocol.

A. Formal Protocol Verification

We used the Scyther Tool [18] to verify our security protocol formally. The Scyther tool is integrated with Python language and is utilized for security protocol verification. Claims in Scyther are used to develop security properties for the protocol. There are different claims defined in Scyther, such as Secret, Alive, Nisynch, Niagree, and Weakagree. We consider Secret, Niagree, and Nisynch claims to analyze our proposed protocol. To check the protocol's desynchronization possibility, we consider the Nisynch claim. The Niagree claim describes that if the values between participants are changed, the agreement will not commit as the value is referred to as secret among the participants. We have implemented our protocol using the Scyther language called SPDL (Security Protocol Description Language) by creating three different roles in the protocol including Consumer, Provider and Authenticator. Events in SPDL need to be well-formed for smooth execution of the protocol. Consumer, Provider and Authenticator roles were defined in SPDL to model a framework to check the proposed protocol's security features. We describe the security claims that are considered to share the access token with the data consumers. As the secret token is being shared between data providers and consumers if it is not encrypted and hashed during the transmission, the attack probability becomes higher in the context of unauthorized access. The attacker capabilities to attack the communication to capture the information have been evaluated and the resilience of our proposed protocol has been verified.

B. Theoretical Security Analysis

This section evaluates our security protocol theoretically against various adversarial attacks.

1) Resilience Against Replay Attack: In this attack, the attacker pretends to be a legitimate party by capturing and replaying messages received from a legitimate sender. Our proposed protocol prevents the pretending device from replaying the message. During the secret key and access token generation phase, each protocol participant is involved in the process. With the involvement of private keys (K_{PB}, K_{CB}) and random numbers (R_{DC}, R_{DP}, R_{BA}), messages between the involved parties are different from session to session, preventing the attacker from capturing and manipulating the session between the parties.

2) Prevention of Secret Disclosure Attack: In this attack, the attacker captures the message and discloses the secret which is communicated between the parties. Our proposed protocol provides security during communication, which prevents the attacker from revealing the secret. As the secret keys are shared between data consumer and blockchain authenticator, such as Data Access Token (DAT_{DC}, DAT_{DP}), this information is encrypted using secret keys K_{CB}, K_{PB}. The critical information remains confidential by encrypting the messages during the communication session. Attackers cannot guess the secret keys generated with a finite elliptic curve's help. To some extent, we assume the attacker can figure out the sending and receiving parties, but it cannot disclose the secret because it does not know the shared secret between them. Besides, each communicating party generates HMAC of the messages (M_1, M_2, M_3) using shared secrets so that they can verify the authenticity of sending and receiving entities.

3) Prevention of Traceability and Unlinkability Attack: In these attacks, the attacker wants to find the constant value from the transmitted message in order to trace back the communicating parties using protocol messages. Our proposed protocol prevents the adversaries from tracing the identities of the communicating parties and their linkage. As discussed earlier, random numbers are involved during the message sharing session. These random numbers are freshly generated for every session, so the attacker cannot find a fixed random number that prevents the attacker from tracing the protocol's participants. Moreover, the attacker cannot reveal any information during the current communication session if it has previous session information because of the lack of change in messages' dynamics for every session.

VI. Performance Analysis

A. SAID Performance Analysis

The runtime performance of our proposed scheme has been analyzed by setting up a blockchain environment. We have used Ganache-Cli and Node.js to implement the protocol and determine its feasibility and functionality. Client and authentication servers were setup to register the participants. Ganache's purpose is to produce smart contracts for each transaction whenever the participants are registered and authenticated with the authenticator server. We have recorded readings to check the time complexity for registration and authentication. When the client wants to register for the first time, it takes relatively more time for authentication than after the registration. During the client's registration, an access token is generated, which is then used by the client for authentication. We have run this process multiple times to record readings to calculate the average time for registration and authentication. When the request is sent to the server, it assigns a transaction id to the received request. After the authentication of the transaction id, it issues an access token to the requester. The computation cost is measured in GAS value. For each transaction, it takes 90737 GAS out of the total 113421 GAS limit. GAS is defined as the full fee or price value needed to execute contracts or conduct a

transaction successfully on a blockchain platform. The time

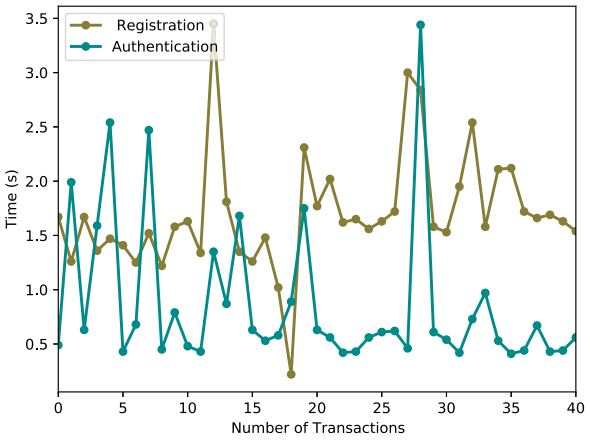

Fig. 4. Time comparison of registration and authentication.

complexity of registration and authentication is measured by recording various transactions for each request received by the blockchain authenticator. Figure 4 shows that registration time is relatively high as compared to authentication. We have recorded multiple transactions to calculate their average time complexity. The time complexity for registration is 1.71 s and for authentication it is 0.81 s, which is quite low as compared to registration time. We have used a different platform (a consortium blockchain) for our system as compared to other related security protocols. Our proposed protocol has low latency as compared to the framework proposed in [19]. The computational latency in [19] is 3.64 s for the best case during recording transactions that is relatively high as compared to our protocol, whose computation times for registration and authentication are 1.26 s and 0.58 s respectively for the best case.

VII. Profit Distribution Analysis

In this section, we present the simulation results of the proposed profit distribution model using coalition game theory. We used NumPy v1.19, Matplotlib v3.3.2, and Pandas v1.1.3 in Python for the simulations. Fig. 5 shows the revenue distribution of the different data providers' coalitions for different coalition sizes. Fig. 6 shows the probability of awards for different data providers. It can be observed from the graphs that as the data provider provides more data, its award probability will increase. This means that when the data provider provides more data, more rewards will be given to the data provider.

Variations on the profit are significant because the value or worth of attributes are dynamic in the corresponding datasets. Data providers of the coalition provide data on demand according to requested attributes. All the providers having common attributes share the data that varies in quantity and quality. The revenue generated after sharing the data is fairly distributed

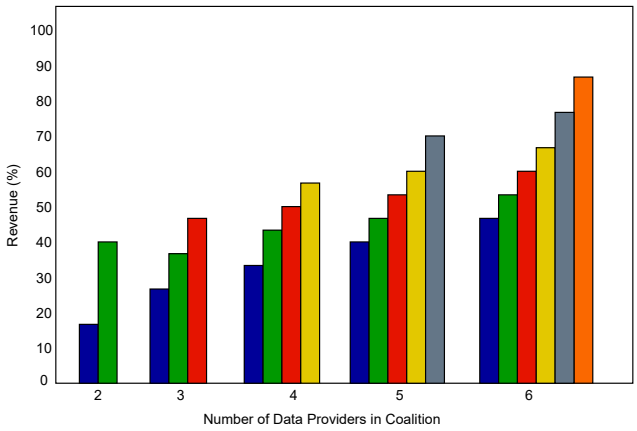

Fig. 5. Revenue percentage of data providers.

among the players. The share of each data provider somehow depends on the coalition's size. However, in [9] if one or two data providers provide the data, they will not receive incentives unless all the other data providers take part in the bidding process. The simulation results show that if the coalition has an increasing number of data providers, the average share of profit for each decreases simultaneously.

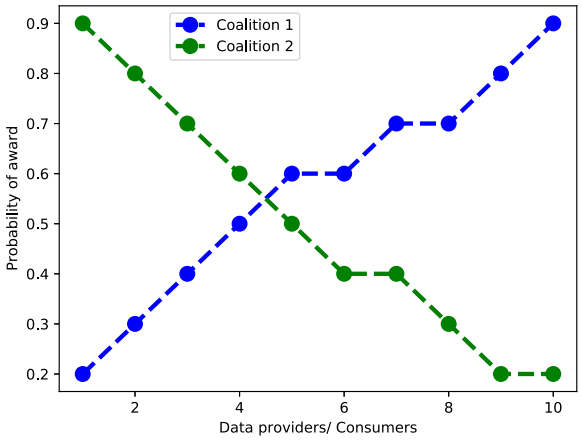

Fig. 6. Probability of awards for coalitions.

In Fig. 6 Coalition 1 shows that, if there is an increasing number of requests from consumers then the probability of award for the providers increases relative to the request. Coalition formation depends on the requested data type, so if there is an increasing number of requests for the corresponding data type, then the award probability increases for those providers in the coalition that contribute to the coalition. However, Coalition 2 shows that if there is an increasing number of data providers in a coalition, but data requests from consumers remain constant, then the probability of award for providers relatively decreases.

CONCLUSION

This paper proposes a secure data sharing and incentive distribution scheme based on blockchain. Specifically, we have proposed SAID, a security verification protocol using ECC and HMAC that verifies the participants of the sharing system while preserving privacy. We have demonstrated that SAID is resilient against replay, secret disclosure, and traceability attacks, as well as other active and passive attacks. Formal security analysis showed that SAID is secure and efficient against unauthorized access to data. After authentication, legitimate data providers participate in the bidding process and generate incentives. This paper also proposes an incentive distribution mechanism using Shapley value to provide fairness. We have shown that our incentive generation and distribution mechanism is efficient and distributes profit among the data providers based on their contributions.

In future work, we will evaluate the trustfulness and valuation of data provided by the data providers. Also, we will evaluate detection of data imputation by the data providers to maximize their share of profit.

ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Project No. 61950410603; and in part by the Characteristic Innovation Project No. 2021KTSCX110 of Guangdong Provincial Department of Education. The Corresponding Author is Alia Asheralieva.

REFERENCES

[1] S. Jing, R. Li, Z. Niu, and J. Yan, "The application of dynamic game theory to participant's interaction mechanisms in lean management," *Computers Industrial Engineering*, vol. 139, p. 106196, 2020.

[2] Y. Wang, Z. Su, N. Zhang, J. Chen, X. Sun, Z. Ye, and Z. Zhou, "Spds: A secure and auditable private data sharing scheme for smart grid based on blockchain and smart contract," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2020.

[3] A. E. Roth, *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press, 1988.

[4] S. Rostampour, M. Safkhani, Y. Bendavid, and N. Bagheri, "Eccbap: A secure ecc-based authentication protocol for iot edge devices," *Pervasive and Mobile Computing*, vol. 67, p. 101194, 2020.

[5] S. Kumari, M. Karuppiah, A. K. Das, X. Li, F. Wu, and N. Kumar, "A secure authentication scheme based on elliptic curve cryptography for iot and cloud servers," *J. Supercomput.*, vol. 74, no. 12, pp. 6428–6453, 2018.

[6] A. Kumari, S. Jangirala, M. Y. Abbasi, V. Kumar, and M. Alam, "Eseap: Ecc based secure and efficient mutual authentication protocol using smart card," *Journal of Information Security and Applications*, vol. 51, p. 102443, 2020.

[7] M. Safkhani, N. Bagheri, S. Kumari, H. Tavakoli, S. Kumar, and J. Chen, "Reseap: An ecc-based authentication and key agreement scheme for iot applications," *IEEE Access*, vol. 8, pp. 200851–200862, 2020.

[8] N. B. Truong, K. Sun, G. M. Lee, and Y. Guo, "Gdpr-compliant personal data management: A blockchain-based solution," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1746–1761, 2020.

[9] R. H. Khokhar, F. Iqbal, B. C. M. Fung, and J. Bentahar, "Enabling secure trustworthiness assessment and privacy protection in integrating data for trading person-specific information," *IEEE Transactions on Engineering Management*, vol. 68, no. 1, pp. 149–169, 2021.

[10] K. Shi, L. Zhu, C. Zhang, L. Xu, and F. Gao, "Blockchain-based multimedia sharing in vehicular social networks with privacy protection," *Multimedia Tools and Applications*, vol. 79, no. 11, pp. 8085–8105, 2020.

[11] J. Chi, Y. Li, J. Huang, J. Liu, Y. Jin, C. Chen, and T. Qiu, "A secure and efficient data sharing scheme based on blockchain in industrial internet of things," *Journal of Network and Computer Applications*, vol. 167, p. 102710, 2020.

[12] P. Cong, G. Xu, T. Wei, and K. Li, "A survey of profit optimization techniques for cloud providers," *ACM Computing Surveys (CSUR)*, vol. 53, no. 2, pp. 1–35, 2020.

[13] S. Li, J. Huang, and B. Cheng, "A price-incentive resource auction mechanism balancing the interests between users and cloud service provider," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 2030–2045, 2020.

[14] H. Song, J. Zhu, and Y. Jiang, "On truthful auction mechanism for cloud resources allocation and consumption shifting with different time slots," *Concurrency and Computation: Practice and Experience*, p. e6122, 2020.

[15] L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. H. Nguyen, and C. S. Hong, "Federated learning for edge networks: Resource optimization and incentive mechanism," *IEEE Communications Magazine*, vol. 58, no. 10, pp. 88–93, 2020.

[16] M. Cong, H. Yu, X. Weng, and S. M. Yiu, *A Game-Theoretic Framework for Incentive Mechanism Design in Federated Learning*, pp. 205–222. Cham: Springer International Publishing, 2020.

[17] B. Zhao, S. Tang, X. Liu, and X. Zhang, "Pace: Privacy-preserving and quality-aware incentive mechanism for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 20, no. 5, pp. 1924–1939, 2021.

[18] C. J. F. Cremers, "The scyther tool: Verification, falsification, and analysis of security protocols," in *Computer Aided Verification* (A. Gupta and S. Malik, eds.), (Berlin, Heidelberg), pp. 414–418, Springer Berlin Heidelberg, 2008.

[19] M. M. Akhtar, D. R. Rizvi, M. A. Ahad, S. S. Kanhere, M. Amjad, and G. Coviello, "Efficient data communication using distributed ledger technology and iota-enabled internet of things for a future machine-to-machine economy," *Sensors*, vol. 21, no. 13, p. 4354, 2021.