Accountable Cross-Border Data Sharing Using Blockchain Under Relaxed Trust Assumption

Mohammad Shahriar Rahman, Abdullah Al Omar, Md Zakirul Alam Bhuiyan, Anirban Basu, Shinsaku Kiyomoto, and Guojon Wang, *Member, IEEE*

***Abstract*—Cross-border data sharing for knowledge generation is a challenging research direction since an application may access personal data stored in countries different from the one where the application is accessed from. In this article, we propose a cross-border data sharing platform where a global cloud is built atop multiple security gateways that are set up in different countries. Once an application requests access to data from a particular country or region, the global cloud collects the data stored in local data hubs through that region's security gateway. While transferring the data to the global cloud, the security gateway records this transfer information on a blockchain maintained by the global cloud. When an application reports any misbehavior (e.g., providing wrong data type or incorrect data) against a security gateway, the global cloud verifies the claim by auditing the blockchain and punishes the misbehaving security gateway if the claim is true. In the case of false misbehavior report, the application itself will be punished by the global cloud. Thus, our platform provides an accountable data sharing function using blockchain that relies on a relaxed trust assumption on the data providers. We include five algorithms to handle data access request, data sharing, blockchain transaction, detecting, and punishing misbehaving entities. In the algorithms, we also introduce how the transaction takes place in the platform. Thus, the proposed platform is able to handle misbehaving data sender, data receiver, or any entity participating in the platform. We analyze our platform empirically by showing different graphs, which have been generated by a number of experiments on blockchain environment. We also delineate how the multilayer of signature (Elliptic Curve Digital Signature Algorithm) acts in our platform.**

***Index Terms*—Blockchain, cross-border data sharing, data hub, global cloud, security gateway, trust assumption.**

I. Introduction

THE Internet of Things (IoT) [1], [2] paradigm is rapidly gaining momentum in modern wireless telecommunications. IoT devices, such as smart sensors designed to monitor temperature, pressure, and other environmental conditions, and wearable devices to measure an individuals state of health, generate vast amounts of time sequence data [3]. These data are stored in the cloud and analyzed for useful information like personal preferences and to predict the environmental conditions surrounding people and the next action that people may take. The impact will increase if the heterogeneous data stored in multiple countries/regions can be organically integrated. As the data obtained from IoT devices are mostly sensitive information related to an individual, namely, personal data, concerns over security, and privacy is an obstacle for the participation of users. Individual users often have limited control or no control over how their personal data are stored, transferred across domain boundaries, or used. To address the concerns and data control issue researchers proposed several blockchain-based solutions [4]–[6]. Blockchain is mainly a distributed ledger, an elaborated description of which has been given in Section III.

A use case for cross-border data sharing without blockchain is depicted in Fig. 1. We show the blockhain-less use case of cross border data sharing to show the real life necessity of blockchain. Data from different IoT devices are stored in the cloud. Those cloud stored data are accessed from different regions/countries. But the data transaction information are not being stored in the blockchain. In this case, malicious intended users may cause harm to the sensitive data. We consider a scenario where a malicious user may affect the cross-border data sharing. Suppose, *Entity A* is wearing a smart watch and his personal data are being stored in a cloud, which could be accessed by any entity from different parts of the world. If any malicious *Entity B* gets access to this personal data then data privacy and security of *Entity A* could be compromised. If in these data his/her sensitive private information is included then it will directly cause data loss to *Entity A*. Also, *Entity B*'s access to that data are recorded as transactions on the blockchain. This type of uncontrolled access to personal data may result in data loss, theft, and alteration. Also, no one can be held accountable for such untoward incident.

Manuscript received May 5, 2019; revised August 14, 2019 and October 11, 2019; accepted November 29, 2019. Date of publication May 5, 2019; date of current version October 9, 2020. Review of this manuscript was arranged by Department Editor K.-K. R. Choo. *(Corresponding author: Mohammad Shahriar Rahman.)*

M. Shahriar Rahman is with the Department of Computer Science and Engineering, University of Liberal Arts Bangladesh, Dhaka 1209, Bangladesh (e-mail: shahriar.rahman@ulab.edu.bd).

A. Al Omar is with the Department of Computer Science and Engineering, University of Asia Pacific, Dhaka 1205, Bangladesh (e-mail: omar.cs@uap-bd.edu).

M. Z. A. Bhuiyan is with the Department of Computer and Information Sciences, Fordham University, New York, NY 10458 USA (e-mail: mbhuiyan3@fordham.edu).

A. Basu is with the University of Sussex, BN1 9RH Brighton, U.K. (e-mail: a.basu@sussex.ac.uk).

S. Kiyomoto is with the Information Security Laboratory, KDDI Research, Saitama 356-0003, Japan (e-mail: kiyomoto@kddi-research.jp).

G. Wang is with the School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, China (e-mail: csgjwang@gzhu.edu.cn).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEM.2019.2960829

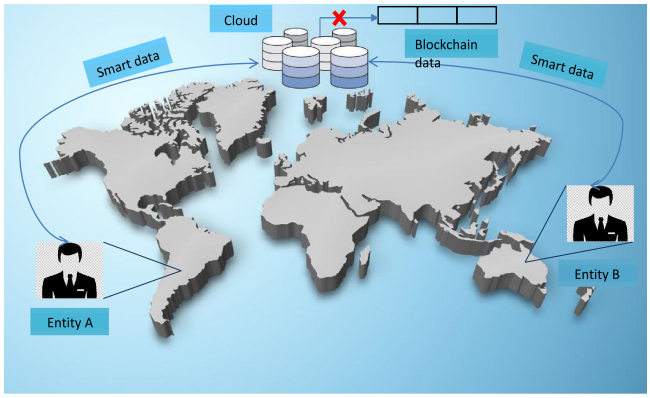


Fig. 1. Data sharing/accessing from any part of the world is important. Using blockchain is also important in the case of cross border data sharing to audit the overall behavior of the entities. This figure presents an use case of cross-border data sharing without blockchain.

So, accountability of data access should be introduced in these scenarios where data could be accessed with verification.

A. Motivation

A recent regulation adopted by the European Union, European General Data Protection Regulation (2016/679) [7], is a legal step, which, among other things, seeks accountability in access to the personal data of EU citizens, and control over the movement of this data both within and outside the EU. It is expected that with increasing trust, decentralized multicloud environments are about to unlock great potential for future data analysis [8]–[12]. However, as the data are shared across borders, the type and quality of data that has been asked for by the data receivers can be compromised due to negligence or malicious intent of the data provider, namely due to misbehavior of the data provider. Previous data sharing platforms implicitly assume that the data provider is trustworthy. In these platforms, if a data receiver ends up with incorrect data, it cannot claim that it has been provided with unwanted data. Also, those platforms cannot provide the accountability of data sharing, namely who provided which data to whom at which time. The functionalities to detect and prove such misbehavior of a data provider are not addressed in those platforms. Even the misbehaving or malicious data receivers are not detected/addressed in those platforms as well.

Our proposed system addresses the aforementioned challenges of cross-border data sharing platform, which increases accountability and trust upon the data sharing cloud platforms. Trust refers to the fact that the users can rely on the platforms for receiving the expected data. When wrong data are provided by the data providers registered with the platforms, we impose certain penalty for the misbehavior to resolve such issues. Thus, the whole platform becomes increasingly trustworthy. Our system deals with the data requests separately and controls each transaction. A data provider is not assumed to be a trusted entity and its act is controlled through penalty mechanism in our proposed system. Data receivers are also controlled so that they are refrained from doing any malicious act to harm data provider or the system. Our platform, thus, works under relaxed trust assumption on the data provider/data receiver such that they need not be completely trusted for their behavior. Both the entities (data provider and data receiver) will have to comply with the policy and regulations of their respective regions for data transaction. Data receiver gets an option to report misbehavior in the case of irrelevant or inequivalent data transaction. At that point, security gateways will check the authenticity of that report and will take action accordingly. Each transaction of data are recorded in the blockchain for auditing [13], which solves the challenge of being anonymous in case of malicious behavior of entities. Each transaction that would be recorded in the blockchain consists of corresponding data transaction's active entity (data provider and data receiver) along with the timestamp. Auditing will be done with that stored information. Global cloud of our system will act as a medium between the data receiver and security gateway. Security gateways will represent a particular region and will be connected to all the data providers of that region. As mentioned earlier, we have considered relaxed trust assumptions on the data providers/receivers. Relaxed trust assumption means that the platform only trusts that the data provider will send the data. Whether the data are right or wrong is not guaranteed by the platform. Hence, we introduce the misbehavior report and punishment process in our platform.

B. Our Contribution:

Our contributions in this article are as follows.

1) We propose a framework for accountable cross-border data sharing platform integrating blockchain. We propose several algorithms to handle data sharing, misbehavior reporting, and blockchain transaction.
2) Data providers and applications involved in cross-border data sharing platform are accountable.
3) Accountability has been achieved by using permissioned blockchain.
4) The accountability protocol allows any application to report misbehavior through a global cloud. The global cloud knows which data provider was reported against using a blockchain verification, and can punish the misbehaving data provider. Similar penalty mechanism is also applicable for misbehaving applications.
5) Global cloud only knows which device is transacting which data. Only this knowledge will not compromise the privacy of the data providers because we have assumed that the data owner herself will be kept anonymous to the global cloud utilizing standard anonymization techniques.
6) Signature will allow our system to attain data integrity. Multilayer signature has been utilized in various protocols. Global cloud and security gateway can verify whether the data has been sent from the corresponding data provider or not through the signature.
7) Misbehaving data sender, data receiver, or any entity participating in the platform will be handled by the platform.
8) The platform allows to share data under relaxed trust assumed on the data providers/data receivers.

C. Paper Organization:

The rest of the article is organized as follows. Section II discusses the previous works, Section III introduces the platform architecture, Section IV describes the data sharing protocol, Section V discusses Algorithms and penalty mechanism for misbehaving parties, Section VI describes the security and performance analysis. Finally, Section VII concludes this article.

II. Related Work

In this section, we have discussed the previous work related to our platform.

The intelligent Knowledge-as-a-Service (iKaaS) [14] platforms have been proposed as a way to share data across borders whereby a global cloud collects data from multiple local cloud systems that are set up in different countries. Although there have been few technical studies on security and privacy for decentralized multicloud environments, these studies have not focused on privacy issues in relation to the cross-border transfer of personal data [15], [16].

Hidano *et al.* [17] designed a security gateway to interpret the regulations in both countries that is capable of flexibly controlling the access permissions of the application while taking privacy into consideration. In this model, the privacy certificate authority (CA) is built for each country as an executive agency responsible for the national regulations governing the handling of personal data. The security gateway refers the privacy certificate issued by the privacy CA in the country where the application exists and the security policy configured by the privacy CA in the country where the local cloud is set up, in order to interpret the regulations in both countries. However, they did not provide a way to configure access permissions for the applications of each union, city, or area. To address this problem, Hidano *et al.* [18] proposed a hierarchical model of multiple privacy CAs, which allows the iKaaS platform to be applied to cases involving multiple regulations related to personal data in the same region. However, it cannot handle the accountability of data sharing. Also, this platform does not handle trust issues on participating entities.

Trans-border data sharing requires some strict policy to be maintained due to privacy and security of the data. Seddon and Currie *et al.* [19] discussed a comparative review of the regulatory and compliance issues surrounding cloud computing. Their main interest includes the healthcare data and IT. They discussed how two nations U.S.-EU (HIPAA or EU data privacy laws), converge in the same policy line to share personal private healthcare data between them. They also discussed a conceptual framework for addressing the issues in healthcare and IT industry in the case of data sharing over cloud. Hörandner *et al.* proposed a privacy preserving framework (CREDENTIAL) [20] to share data over cloud. The use of proxy re-encryption and redactable signatures allows their framework to mitigate obstacles on security and privacy for data sharing. They introduced hardware-based multifactor authentication mechanisms to overcome the present insecure situation of cloud computing platforms. Nalin *et al.* have described the current state of the eHealth infrastructure in [21] for cross-border health data exchange in Europe. They have described about CONFIDO (EU-funded research project for secure cross-border health data exchange) and shown how the security of the EU eHealth data infrastructure is being reinforced by it. They enlisted all the key points relating security of the project including: Trusted Execution Environment, Physical Unclonable Function, and Homomorphic Encryption mechanisms.

A blockchain is a distributed database that maintains a continuously-growing list of records called blocks secured from tampering and revision [22]–[27] through a series of transactions and verifications by the miners. Smart contract [28], [29] based technology leverages the accountability of platforms. Third party based platforms are accepting blockchain for its smart contract based service, which requires no trusted third party. In a permissioned blockchain [30], every transaction contains a list of miners authorized to verify the transaction. This can be predecided by the source or the owner of the data associated with the transaction.

Chang *et al.* [31] have described the opportunities and the challenges of adopting blockchain in the cross-border trades. The potential of blockchain in cross-border supply chain market is the main focus point of their research. Qui *et al.* [32] have conducted a SWOT analysis on SWIFT and ripple (based on blockchain technology) systems, which are being used in the cross-order remittance market. The result of their analysis is pointing out that blockchain technology based systems will eventually revolutionize the cross-border remittance industry.

III. Proposed Architecture

In this section, we have described our platform, each entity, and their roles.

On this platform, as shown in Fig. 2, a global cloud is hierarchically built atop multiple security gateway systems that are set up in different countries. The global cloud organically integrates the data stored in the data hubs, and the integrated data are provided for various types of applications as knowledge. Security and privacy are controlled by a security gateway at the entrance to each region. When using the platform, the application can access the data for different countries, conduct multiple-scale analyses, and compare different countries. On the other hand, if the application accesses personal data in different countries, the platform must handle the data in accordance with the regulations governing personal data in both the country where the application exists and the country where the security gateway is set up.

Scope of the entities in our platform has been described here.

Global cloud: Global cloud accepts data requests from the applications and works accordingly. Global cloud is connected with all the security gateways and collects data through this connection. It verifies all the misbehavior report of the application and punishes the entity with misbehavior.

Security gateway: Security gateway takes the request from the global cloud and sends it to the data hubs. Other than processing data security gateways help the platform to find the misbehaving entity.

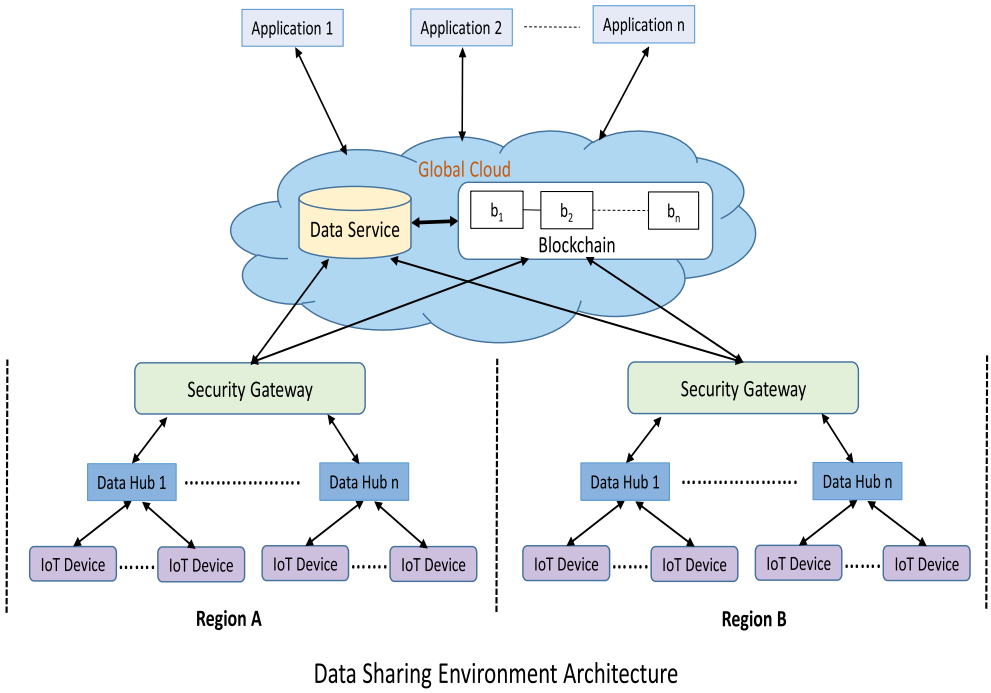


Fig. 2. Cross-border data sharing platform architecture.

Data hub: Data hubs collect data from the IoT devices. These also help the platform to identify the misbehaving IoT devices.

Application: It can request data from the cloud and also can report misbehavior.

A. Security Gateway

Each region has a security gateway at the touch point with the global cloud. The queries from the application and the data on the data hub are all exchanged through the security gateway. The security gateway handles the access control of the application under the rules governing the handling of personal data both for the country of the application and the country of the data hub. It may also deal with privacy control on behalf of the data owners as in [18]. The security gateway uses a token to control the access of the application. When an application requests access to the data hub's storage, the security gateway generates a token and returns it to the application. The application with the token can request the data any number of times until the token has expired. The application is then required to specify the data hub's ID or the types of data that it wants to have access to and its membership certificate. When issuing the token, the security gateway refers to the membership certificate and the security policy in order to comply with the rules of both countries. Thus, the gateway determines whether or not the application is permitted to access the data.

B. Blockchain

The blockchain is maintained by the global cloud. Every time a security gateway forwards data to the global cloud's data service module, this action is recorded on the blockchain as a transaction. When an application reports a misbehavior to the global clouds data service module, the data service module verifies the claim by checking the blockchain. We assume a permissioned blockchain, whereby every transaction contains a list of miners authorized to verify the transaction. This can be predecided by the membership of the platform.

Transactions are recorded on the blockchain with the help of smart contracts of our system. Smart contracts are the medium (code) to write something in the blockchain. Whenever global cloud tries to record the transaction in a block, global cloud needs to verify itself to the blockchain environment. Global cloud uses the verified account (node of any blockchain network) of the blockchain network to write the transaction. After verification the global cloud becomes authorized to write the transaction in the blocks. To audit a particular transaction in the blockchain data service module will also take the advantage of smart contracts. Data service module will check the transaction through the hash-id of that transaction. After the audit our platform will take decision on misbehavior reports.

IV. Protocol for Data Sharing

In this section, we have sketched the protocol for data sharing. We have discussed the steps on how data are being accessed and also how we are providing the accountability to the users.

The data sharing protocol is divided into the following two stages: 1) data access and 2) accountability

Table I describes the notations that are used in the following sections.

TABLE I
TERMINOLOGY TABLE

Notation	Description
$\mathcal{A}_{\text{id}}$	Registered Applications of our protocol
c_i	Certificate of registration
τ	Time variant token
$\mathcal{DH}_i$	Data Hubs
$\mathcal{SG}_i$	Security Gateways
$\mathcal{R}_{\mathcal{A}\text{id}}$	Region of $\mathcal{A}_{\text{id}}$
$\mathcal{R}_{\mathcal{DH}_i}$	Region of $\mathcal{DH}_i$
δ_{req}	Requested data
δ_{res}	Data $\mathcal{A}_{\text{id}}$'s responded with
RAC	Security policy checker (for country of $\mathcal{A}_{\text{id}}$ & $\mathcal{DH}_i$)
$\mathcal{GC}$	Global Cloud
$\varphi(p)$	Misbehavior report, where p is the penalty
$\mathcal{M}$	Metadata of data transaction

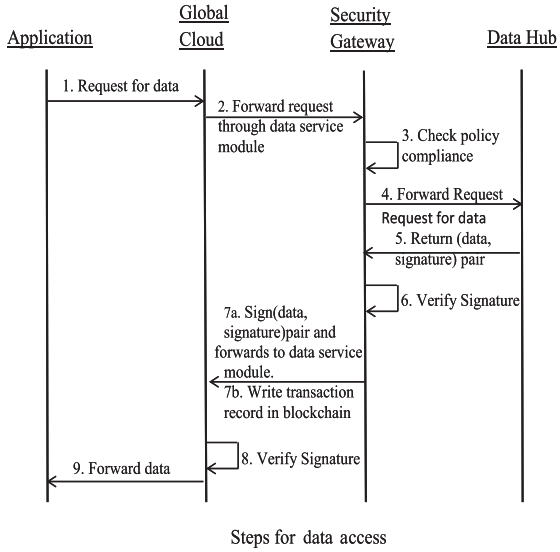


Fig. 3. Steps for data access management.

A. Data Access

Steps involved in data access are shown in Fig. 3.

1) An application sends request to global cloud to access a particular type of data from a particular region.
2) The global cloud processes this request through its data service module. Upon checking the type and location of the data requested for, the service module forwards this request to the security gateway of that region.
3) The security gateway checks whether the request is compliant with the policy and regulation of that region.[1] Security gateway rejects the request if the check fails, otherwise the request is forwarded to the data hub that stores the requested type of data.
4) The data hub generates a (data, signature) pair and forwards it to the security gateway.

[1]This will not compromise the security of any region/country as our platform will work only with the privacy policy of any region/country which is public like, General Data Protection Regulation of EU.

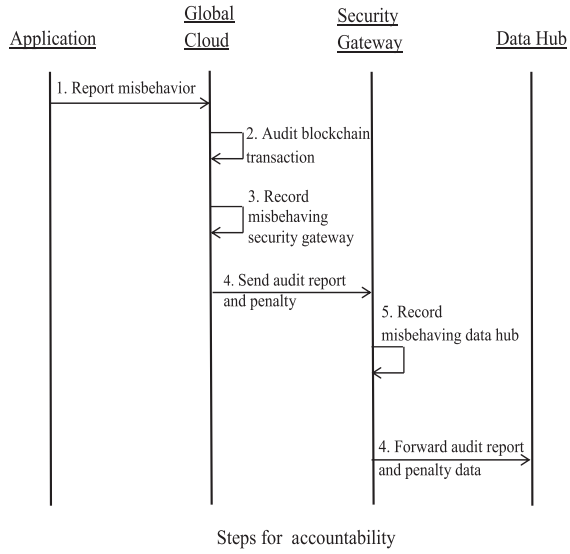


Fig. 4. Steps for performing accountability.

5) The security gateway verifies the signature. If the verification fails, the corresponding data is discarded.
6) If the signature verification succeeds, the security gateway does the following tasks.
   a) Signs the (data, signature) pair that it received from the data hub, sends the [signature, (data, signature)] pair to the data service module of the global cloud, and stores the data transfer information as a transaction in its local storage as a supplementary data to the data provider or application in the platform.
   b) Writes the transaction record in the blockchain to verify the misbehavior of data provider or application later.
7) The global cloud verifies the signature of the security gateway. The (data, signature) pair is discarded if the verification fails.
8) If the verification succeeds, the global cloud forwards the data to the requesting application.

B. Accountability

Steps involved to achieve accountability are shown in Fig. 4.

1) An application reports a misbehavior to global cloud. This report includes the type of data, its source region and the time when the data were received.
2) The data service module in the global cloud accesses the blockchain and verifies the claim made by the application by auditing the blockchain.
3) If the reported misbehavior is found true, the global cloud computes the amount/type of penalty or deterrence for the misbehavior, and records the details of the misbehaving security gateway and its punishment in its database.
4) The global cloud sends the audit report and the penalty or deterrence information to the security gateway.

Algorithm 1: Data Access and Misbehavior Report by Applications.

Input: $\mathcal{A}_{\mathrm{id}}, PWD, t_{\mathrm{i}}, \tau, \mathcal{R}_{\mathcal{A}_{\mathrm{id}}}, \mathcal{R}_{\mathcal{DH}_i}, \delta_{req}$
Output: $\delta_{res}, \varphi(p)$
$f_{\mathcal{A}_{\mathrm{id}}}$ = financial payment of $\mathcal{A}_{\mathrm{id}}$
{payment will be done during registration of $\mathcal{A}_{\mathrm{id}}$ in this system}
$r_A^{f^{\mathcal{A}_{\mathrm{id}}}, \mathcal{R}_{\mathcal{A}_{\mathrm{id}}}} \longleftarrow \{ \mathcal{A}_{\mathrm{id}}, PWD \}$
{Registered Application Verification}
if ($\mathcal{A}_{\mathrm{id}} \in \mathcal{A}_{\mathrm{id}}^{Registered}$) **then**
  $c_i \longleftarrow certify(\mathcal{A}_{\mathrm{id}}, t_i)^{\tau}$
  {Certificate will be generated once until τ expires }
  if ($c_i = valid^{\tau}$) **then**
  $\delta_{\mathrm{res}} \longleftarrow DAR\{\delta_{\mathrm{req}}\}^{\mathcal{R}_{\mathcal{A}_{\mathrm{id}}}}, \mathcal{R}_{\mathcal{DH}_i}, c_i$
  {DAR-Data Access Request}
    if ($\delta_{\mathrm{req}} \equiv \delta_{\mathrm{res}}$) **then**
      return δ_{req}
    else
      Revoke $\varphi(p)$
      {penalty p will be decided upon the type of φ}
      return $\varphi(p)$
    end if
  else
    Revoke a request to update τ by paying $f^{\mathcal{A}_{\mathrm{id}}}$
  end if
else
  return $\emptyset$
end if

5) The security gateway checks the audit report to see if the reported misbehavior is against itself. If not, the misbehavior was from a data hub. The security gateway stores the audit report and the corresponding penalty or deterrence of the data hub.
6) The security gateway forwards the audit report and the penalty or deterrence information to the corresponding data hub.
7) If an application reports a false misbehavior then it will be penalized.

V. Protocol Construction

In this section, we have introduced five algorithms and described them. We have also introduced the penalty mechanism of our platform.

A. Algorithms for the Platform

1) Algorithm 1: Upon registering with the system, $\mathcal{A}_{\mathrm{id}}$ will get a membership certificate $c_i \longleftarrow \mathrm{certify}(\mathcal{A}_{\mathrm{id}}, t_{\mathrm{i}})^{\tau}$ (time-variant function) where t_{i} is the time stamp of the $\mathcal{A}_{\mathrm{id}}$ denoting their registration stamp in this system and τ is the token, which is controlling the time-variant attribute. Accessing data from a system-certified member will request with τ, $\mathcal{R}_{\mathcal{A}\mathrm{id}}$, $\mathcal{R}_{\mathcal{DH}_i}$, and δ_{req}. $\mathcal{R}_{\mathcal{A}_{\mathrm{id}}}$ is the region of $\mathcal{A}_{\mathrm{id}}$, $\mathcal{R}_{\mathcal{DH}_i}$ is the region of $\mathcal{DH}_i$ and δ_{req} to be accessed from the data service module. Another functionality of data service module of our system is reporting of misbehavior φ by $\mathcal{DH}_i$ or by $\mathcal{SG}_i$ after verifying δ_{res} against δ_{req}.

2) Algorithm 2: Data request from $\mathcal{A}_{\mathrm{id}}$ will be handled by $\mathcal{GC}$. DAR will transfer the request to $\mathcal{GC}$ along with $\mathcal{R}_{\mathcal{A}_{\mathrm{id}}}$, $\mathcal{R}_{\mathcal{DH}_i}$, c_i for processing the requests. These requests will be transferred to the particular $\mathcal{R}_{\mathcal{DH}_i}$'s $\mathcal{SG}_i$ if c_i of corresponding $\mathcal{A}_{\mathrm{id}}$ is passed as a valid certificate from Algorithm 1. Here, $\mathcal{R}_{\mathcal{DH}_i}$ will be equivalent to its $\mathcal{R}_{\mathcal{SG}_i}$. Response of $\mathcal{SG}_i$ will be a signed version of (data, signature) pair (pair contains the signature of $\mathcal{DH}_i$). This multilayer signed data are denoted Sign(data,signature). Verification of this multilayer signed data will be done by $\mathcal{GC}$ itself before passing the verified data (only data) to requester.

Upon receiving δ_{res}, $\mathcal{A}_{\mathrm{id}}$ checks the equivalence between δ_{res} and δ_{req}. If any inconsistency (e.g., corrupted data, data from different region) found into δ_{res} then $\mathcal{A}_{\mathrm{id}}$ will report φ to $\mathcal{GC}$. Equivalence of δ_{res} and δ_{req} will be checked through keyword matching. $\mathcal{A}_{\mathrm{id}}$ will define some keywords in δ_{req} and after the response δ_{res} will be checked for those similar keywords in it (e.g., if δ_{req} is about healthcare data then in δ_{res} healthcare related keyword will be searched). Through this approach the equivalence will be checked and verified. To verify φ, $\mathcal{GC}$ will audit blockchain and if the φ is found true then the amount or type of penalty will be returned in the form of $\varphi(p)$ (here, p is the amount or type of penalty). If the reported φ is found wrong then the misbehavior reporting $\mathcal{A}_{\mathrm{id}}$ will be penalized. At the same stage $\mathcal{GC}$ will store φ and p separately in a publicly accessible database.

3) Algorithm 3: Registered $\mathcal{SG}_i$ (by financial payment, $f^{\mathcal{SG}_i}$) represents particular regions of our system. Each $\mathcal{SG}_i$ will be connected to all the $\mathcal{DH}_i$s of that region and will process the requests and responses of $\mathcal{A}_{\mathrm{id}}$ and $\mathcal{DH}_i$, respectively. Along with handling the request and response, $\mathcal{SG}_i$ will issue the updated token, τ' to each $\mathcal{A}_{\mathrm{id}}$ who will receive δ_{res} after successful checking of the valid data request with the regional access control (RAC). RAC checks the security policies of both the countries (country of $\mathcal{A}_{\mathrm{id}}$ & $\mathcal{DH}_i$). τ' will be updated until the maximum allowable number of services reached or the allocated time is expired. $\mathcal{SG}_i$ will verify the (δ_{res}, signature) pair sent by $\mathcal{DH}_i$ and, then, will generate Sign[(δ_{res}, signature) pair].

Each time the response by the $\mathcal{SG}_i$ will be a multilayer signed version of the data. After responding with it $\mathcal{SG}_i$ will write a transaction on blockchain to be verified later by the entities. Reported φ will be checked whether it is about $\mathcal{SG}_i$. If the φ is not about $\mathcal{SG}_i$ then $\mathcal{SG}_i$ will send the φ to the corresponding $\mathcal{DH}_i$ and will update database.

4) Algorithm 4: $\mathcal{DH}_i$ are the only source of data in our system. Each request of $\mathcal{A}_{\mathrm{id}}$ will be validated by the entities operating between $\mathcal{A}_{\mathrm{id}}$s and $\mathcal{DH}_i$s but the requested data will be responded by $\mathcal{DH}_i$ solely. A single $\mathcal{DH}_i$ could have a lot of data types in it to response with. So, after getting the data request δ_{req}, $\mathcal{DH}_i$ will search for the equivalent data in it. In case of a successful search, $\mathcal{DH}_i$ will keep the data for further processing otherwise it will discard the δ_{req}. After each successful search $\mathcal{DH}_i$ will sign the data so that it could be validated later by the

Algorithm 2: Response to $\mathcal{A}_{\text{id}}$s and φ Report Passing by Global Hub.

Input: $DAR\{\delta_{\text{req}}\}^{\mathcal{R}_{\mathcal{A}_{\text{id}}}}, \mathcal{R}_{\mathcal{DH}_i}, c_i, \delta_{\text{res}}$ [from $\mathcal{SG}_i$], Corresponding Block-id $\mathcal{B}_{\text{id}}$ where the transaction of that particular request has been written, $\varphi\{p\}$, $f^{\mathcal{A}_{\text{id}}}$
Output: p, δ_{res}

$f^{\mathcal{SG}_i}$ = financial payment of $\mathcal{SG}_i$
{payment of registration $\mathcal{SG}_i$}
DB$^{\texttt{public}}$ {Public Database}
$r_{\text{SG}}{}^{\{f^{\mathcal{SG}_{\text{i}}}, \mathcal{R}_{\mathcal{SG}_{\text{i}}}\}} \longleftarrow$ { $\mathcal{SG}_{\text{i}}, PWD$ } {Registered Security Gateway Verification}
$k() \longleftarrow \{\delta_{\text{req}}\}^{\mathcal{R}_{\mathcal{A}_{\text{id}}}, \mathcal{R}_{\mathcal{DH}_i}, c_i}$ {$k()$ will return a}
where, $a \in \delta_{\text{req}}, \mathcal{R}_{\mathcal{DH}_i}, c_i$
for $i = 1$ to n **do**
  if ($\mathcal{R}_{\mathcal{DH}_{\text{i}}} = \mathcal{R}_{\mathcal{SG}_{\text{i}}}$) **then**
    $\mathcal{SG}_{\text{i}} \longleftarrow a$ {send the request to the corresponding region's $\mathcal{SG}_{\text{i}}$}
  end if
end for
{Global Cloud will receive response from $\mathcal{SG}_{\text{i}}$}
$\mathcal{GC} \longleftarrow a', \tau'$
{here, a' $\longleftarrow Sign(\delta_{\text{res}}, signature), a$}
  if ($verify_{\text{sign}}(Sign(\delta_{\text{res}}, signature)) = TRUE$) **then**
    $\tau = \tau'$
    update c_i
    return δ_{res} {Only δ_{res} will be returned}
  else
    delete δ_{res}
    return $\emptyset$
  end if
  { $\mathcal{GC}$ may get report from $\mathcal{A}_{\text{id}}$}
  $\mathcal{GC} \longleftarrow \varphi^{a', c_{\text{i}}}$
  audit($\mathcal{B}_{\text{id}}$, c_i, a', $\mathcal{R}_{\mathcal{SG}_{\text{i}}}$)
  calculate p
  if ($audit_{\text{response}} = TRUE$) **then**
    $f^{\mathcal{SG}_i} = f^{\mathcal{SG}_i} - p$ {$\mathcal{SG}_i$'s financial statement will be updated}
    DB$^{\texttt{public}} \longleftarrow \mathcal{SG}_i, p, c_i, \varphi$
else
    $f^{\mathcal{A}_{\text{id}}} \longleftarrow \mathcal{SG}_i$-$p$ {$\mathcal{A}_{\text{id}}$'s financial statement will be updated if it reports wrong φ}
    DB$^{\texttt{public}} \longleftarrow \mathcal{A}_{\text{id}}, p, c_i, \varphi$
  end if

Algorithm 3: Data Forwarding and φ Report Processing by Security Gateway.

Input: a, $\varphi(p)$
Output: τ', $\varphi(p)$ [if φ, is about $\mathcal{DH}_i$], $\mathcal{B}_{\text{id}}$
j=0 {To choose the compliant $\mathcal{DH}_i$}
Send $f^{\mathcal{SG}_{\text{i}}}$ to GC
{On registration and if $f^{\mathcal{SG}_{\text{i}}}=\emptyset$ by $\varphi(p)$}
$\mathcal{SG}_{\text{i}} \longleftarrow a$
for $i = 1$ to n **do**
  {checks whether $\text{DAC}_{\delta_{\text{req}}}$ is compliant with $\text{DAC}_{\mathcal{DH}_i}$}
  if ($DAC_{\delta_{\text{req}}} \equiv DAC_{\mathcal{DH}_i}$) **then**
    j=i
  else
    rejects δ_{req}
  end if
end for
if (j$\neq$0) **do**
  $\mathcal{DH}_j \longleftarrow \delta_{\text{req}}$
  $\mathcal{SG}_{\text{i}} \longleftarrow (\delta_{\text{res}}, signature) - pair$
  if ($verify_{\text{sign}}(\delta_{\text{res}}, signature) = TRUE$) **then**
    Signature($\delta_{\text{res}}, signature$) $= Sign((\delta_{\text{res}}, signature) - pair)$ {multi-layer signed data}
    generates τ'
    writes transaction on blockchain
    stores $\mathcal{B}_{\text{id}}$
    return τ', $\mathcal{B}_{\text{id}}$
  else
    delete$((\delta_{\text{res}}, signature) - pair)$
  end if
else
  return $\emptyset$
end if
{ $\mathcal{SG}_i$ may get report from $\mathcal{GC}$}
if ($\varphi(p) \in \mathcal{SG}_i$) **then**
  DB$^{\texttt{public}} \longleftarrow p$ {updates p for $\mathcal{SG}_i$}
else
  $\mathcal{DH}_j \longleftarrow \varphi(p)$
  DB$^{\texttt{public}} \longleftarrow p$ {updates p for $\mathcal{DH}_j$}
end if

entities operating between $\mathcal{DH}_i$ and $\mathcal{A}_{\text{id}}$. After signing the data $\mathcal{DH}_i$ will send the $(\delta_{\text{res}}, \text{signature}) - \text{pair}$ to its $\mathcal{SG}_{\text{i}}$. All the valid data requests will go through the above discussed process and will be sent to the $\mathcal{A}_{\text{id}}$ through the entities between them.

5) Algorithm 5: The smart contracts have been used in our platform, which include the process of transaction. Only $\mathcal{SG}_i$ and $\mathcal{GC}$ are able to interact with the blockchain. Whenever they try to interact, authenticity of their corresponding nodes/accounts of blockchain network will be checked. Upon successful authentication they will be able to request for the transaction. In the case of getting $\mathcal{B}_{\text{id}}$ first, the transaction will be done by $\mathcal{SG}_i$. Then, blockchain will return the $\mathcal{B}_{\text{id}}$ through smart contract. Whenever any φ will be reported, $\mathcal{GC}$ will audit the blockchain for verification. For this reason $\mathcal{GC}$ will authenticate it self through the smart contract. Then, it will request with the particular $\mathcal{B}_{\text{id}}$. After getting $\mathcal{B}_{\text{id}}$ blockchain will return the $\mathcal{M}$ or the metadata.

B. Penalty Mechanism in the Platform

We assume that the type of penalty is in financial form. At the beginning, any security gateway who intends to join the platform (similarly, if a data hub joins a security gateway),

Algorithm 4: Response to Data Requests by $\mathcal{DH}_i$.

Input: δ_{req}
Output: $(\delta_{\text{res}}, signature) - pair$
$f^{\mathcal{DH}_i}$ = financial payment of $\mathcal{DH}_i$
$m = 0$, d_n {n types of data $\mathcal{DH}_i$ holds}
Send $f^{\mathcal{DH}_i}$ to $\mathcal{SG}_i$
{On registration and if $f^{\mathcal{DH}_i}=\emptyset$ by $\varphi(p)$}
$\mathcal{DH}_i \longleftarrow \delta_{\text{req}}$
for $i = 1$ to n **do**
  if $(\delta_{\text{req}} \equiv d_n)$ **then**
    $m = n$
  else
    delete δ_{req}
  end if
end for
if $(m \neq 0)$ **then**
  $\delta_{\text{res}} = d_m$
  $signature(\delta_{\text{res}}) \longrightarrow (\delta_{\text{res}}, signature)$
  return $(\delta_{\text{res}}, signature)$
else
  delete δ_{req}
  return $\emptyset$
end if
if $(\varphi(p) \in \mathcal{DH}_i)$
  update p
else
  delete δ_{req}
end if

Algorithm 5: Blockchain Transaction.

Input: $\mathcal{M}$
Output: $\mathcal{B}_{\text{id}}$
$flag = FALSE$
if $(\mathcal{SG}_i = authentic\_account)$ **then**
  $recieve \longleftarrow \mathcal{M}$
else
  decline
  {In the case of unauthorized blockchain node it will not recieve the transaction}
end if
if $(flag == TRUE)$ **then**
  return $\mathcal{B}_{\text{id}}$
else
  return $\emptyset$
  {Smart contract will stop running.}
end if
{Next following steps will run for $\mathcal{GC}$}
if $(\mathcal{GC} = authentic\_account)$ **then**
  $\mathcal{GC} \longrightarrow \mathcal{B}_{\text{id}}$
{ B_{id} will be sent to the blockchain.}
if $\mathcal{B}_{\text{id}}$ is TRUE **then**
  $\mathcal{GC} \longleftarrow \mathcal{M}$
end if
else
  return $\emptyset$
end if

commits through financial payment.[2] If audit fails, then, the security gateway has to pay a penalty. The penalty (if any) is also logged in the global cloud's database. The security gateway pays a penalty of p to the global cloud. In case a data hub has to pay the penalty, p is paid by the data hub to the global cloud through the security gateway. global cloud verifies that misbehaving party's account has enough balance to pay the penalty p. The penalty transaction handles the transfer of penalty between the misbehaving party and the global cloud. If it does not have sufficient funds for paying the penalty, then its membership is temporarily revoked until the payment is made. This essentially means that the misbehaving party cannot send the data to anyone else. Once the penalty information is put on the global cloud's database, the information is public and available to applications.

VI. Security and Performance Analysis

In this section, we have discussed the security and performance analysis elaborately. Empirical overview of our platform are shown through graphs with description.

A. Accountability

External entities of our system are accountable for their behaviors. If any misbehavior is reported by the $\mathcal{A}_{\text{id}}$, then the misbehavior is analyzed by the entities and necessary penalty is imposed on the particular entity.

1) $\mathcal{A}_{\text{id}}$ can report φ against the data if it is not corresponding to its δ_{req}.
2) After φ is reported, the system will check for the authenticity of φ. $\mathcal{GC}$ will audit the blockchain to check whether this φ is right or wrong. In this way any $\mathcal{A}_{\text{id}}$ is accountable for its misbehavior report.
3) If the φ is not correct then the penalty will be generated for $\mathcal{A}_{\text{id}}$ by the $\mathcal{GC}$.

B. Verifiability

1) $\mathcal{A}_{\text{id}}$ will hold an account in our system. Each time $\mathcal{A}_{\text{id}}$ will have to authenticate itself to the system.
2) Only verified $\mathcal{A}_{\text{id}}$ will be able to interact with the system.
3) $\mathcal{R}_{\mathcal{DH}_i}$ of each important entity $\mathcal{R}_{\mathcal{A}_{\text{id}}}$ can be verified.
4) Each $\mathcal{DH}_i$ is verifiable with its signature in the data sharing process. The signature will be verified with the public key of $\mathcal{DH}_i$.
5) $\mathcal{SG}_i$ is also verifiable with its signature.
6) $\mathcal{GC}$ and $\mathcal{SG}_i$ will verify $\mathcal{SG}_i$ and $\mathcal{DH}_i$, respectively.

C. Integrity

1) Security policies of each region is checked in the system.
2) The security policies are preserved at the time of data share/access. RAC preserves it for the entities.

[2]We do not explicitly define how the payment will be made by the parties. It could be in the form of annual membership fee or on credit.

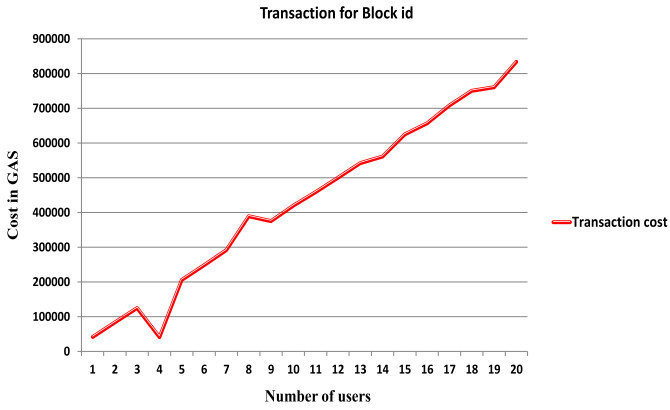


Fig. 5. Transaction for block-id generation.

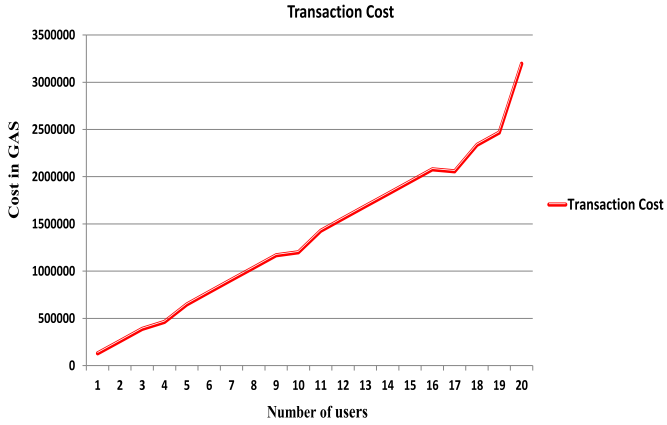


Fig. 6. Transaction cost without block-id generation.

3) $\mathcal{GC}$ and $\mathcal{SG}_{\mathrm{i}}$ can write on blockchain, which gives the transaction integrity.
4) Through (δ_{res},signature) pair, the integrity of $\mathcal{DH}_{\mathrm{i}}$ is checked.
5) With (δ_{res},signature) pair, the $\mathcal{SG}_{\mathrm{i}}$ are being verified.

D. Performance Analysis

We setup an environment to evaluate our protocol by writing programs using Solidity 0.5.0 and JAVA 1.8 with a computer Intel(R) Core(TM) i5, CPU-3.30 GHz, 8 GB of RAM, Windows 10, 64-bit OS. We utilized Elliptic Curve Digital Signature Algorithm (ECDSA) to implement the signatures.

1) Block-id Generation in Blockchain: We take the block-id generation cost solely to see how it behaves during the transaction period. Number of users for transaction is increasing by one in the Fig. 5. The graph is increasing with the increasing number of users. It is almost linear. First users transaction takes nearly 041 million amount of gas.[3] Last user of our test takes 083 million gas. But in the experiment user 4 takes same amount of gas as the first user. In the case of user 8, the amount of gas increases significantly. So we are assuming that the network stability also has an impact on the transaction. The faster the network will be, the lesser the gas amount will be taken to make the transaction.

2) Transaction Cost Without Block-id Generation Cost: We calculate the amount of gas each transaction takes to finish the data transaction in our setting. The first user takes nearly 12 million amount of gas to transact. Fig. 6 is nearly linear till user 16, then it is showing a different form. Again, we assume that stability of network pays a role in gas fee. After user 16, the stability of network was reformed so we get a better performance in this period. But in the period of user 10 to user 16 the graph is totally stable. The reason of this stability is also related to the stability of network. However, the user 20 finishes approximately at 31 million gas cost, which is at the linear stage with the previous users (1–15).

[3]Gas is the unit to measure the cost in blockchain network, which measures how much it takes to deploy a smart contract in the network and also gas has an actual value in currency.

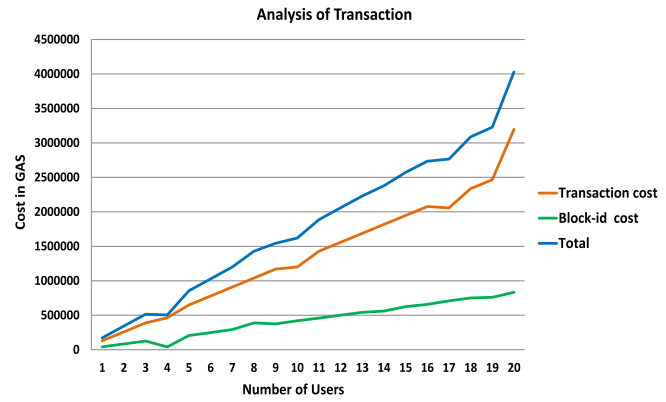
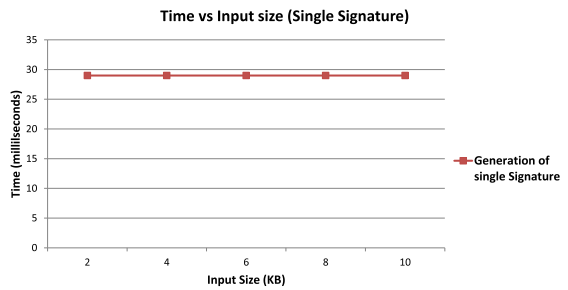


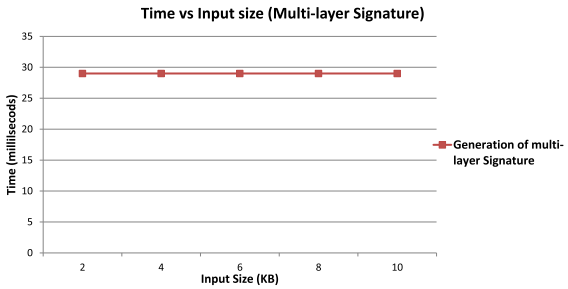
Fig. 7. Analysis of transaction cost in blockchain.

3) Total Amount of Cost for the Transaction: In Fig. 7, we can see there are three lines in the graph. The blue line denotes the total gas cost of the transaction in blockchain. In this analysis, we can see that in the beginning for user 1, the total and the transaction cost without block-id generation are similar. But the block-id generation cost is low with respect to them. Total amount for a transaction is approximately 17 million where the block-id generation costs 12 million and only transaction costs nearly 12 million of gas. In the case of last user (user 20) total amount of gas needed is 4 million, which is too high in comparison with the block-id generation cost. So we can conclude by saying that the transaction cost increases with the number of user though the block-id generation cost is very low in the beginning (for user 1). For a set of 20 users whenever $\mathcal{SG}_{\mathrm{i}}$ and $\mathcal{GC}$ will transact highest amount of gas could be 4 million of gas in our setting.

4) Signature Analysis: ECDSA [33] has been used to generate the signatures in our system. Each transaction of data needs two signatures to be signed and verified. Upon the justification of data request data hubs $\mathcal{DH}_i$ will share their signed data with a signature. Security gateways $\mathcal{SG}_i$ will verify the signature and ensure that the data has been sent from a trusted or authenticated data provider. Later, $\mathcal{SG}_i$ will again sign on the (data, signature) pair and share the pair with global cloud $\mathcal{GC}$. Then, $\mathcal{GC}$ will



(a)



(b)

Fig. 8. Analysis of signatures. (a) Time for single signature. (b) Time for multilayer signature.

verify the signature and the verified data will be sent to data requester.

We have analyzed the signature function with time and input size. We take 2 to 10 kB of data to be signed by the program. Fig. 8(a) delineates the behavior of our signature with increasing input size. We can see that the graph is linear and parallel to *x*-axis, which means that the signature will need similar amount of time with the increasing input size. Size of input has no effect on the time to sign the program with ECDSA. Fig. 8(b) depicts the behavior of multilayer signature. This graph is similar as 8(a), which means to sign multiple time or twice the signature will take same time as single sign. Input size has no effect on this multilayer signature. It will take 29 ms to generate a signature in both cases. We can conclude by saying that input size has no effect on signature generation though it will be signed twice.

E. Comparison With Related Works

This section shows a comparison between our platform and other related platforms. We consider certain properties/metrics of the platforms for fair comparison. If a particular platform has the property in it then we marked it with "Y" otherwise marked with "N". Here, Table II compares our platform with other existing systems and literature presented in this article. With the various metrics derived from the careful analogy of the systems, a conclusion can be drawn that our platform has greater advantages as compared to the systems presented.

VII. Conclusion

In this article, we proposed a data sharing platform for cross-border data utilization. The platform leverages a global cloud that collects the requested data from a data provider operating in a different region. While providing data, the data provider needs to record the transaction on a global blockchain. The global cloud can audit the transaction if there is any report of misbehavior from the applications and can penalize misbehaving data provider or application. In the platform data sender, data receiver, or any participating entity in the platform can be punished for the misbehavior. Our proposal, thus, provided a platform for accountable cross-border data sharing where an application does not need to fully trust a data provider while requesting data access. We are yet to investigate searchability on encrypted data through this platform. Prototyping the proposed platform for large scale data, data providers, and data receivers will be our future work.

TABLE II
Comparison Between Proposed Platform and Other Related Systems

Metric	[17]	[33]	[34]	Our Platform
Trusted Platform	Y	Y	Y	N
Trusted Data Provider/Receiver	N	Y	N	Y
Accountability of Data Provider/Receiver	N	Y/N[4]	N	Y
Penalty Mechanism	N	N	N	Y
Cross-border Policy Adherence	Y	N	N	Y
Privacy of Data Owner	Y	Y	Y	Y
Multi-cloud Environment	Y	N	N	Y
Blockchain-Based	N	Y	Y	Y

Acknowledgment

The views, opinions, and/or findings contained in this article are those of the author. These are not related to his work at Hitachi and should not be interpreted as an official Hitachi position, policy, or decision, unless so designated by other documentation.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.
[2] P. P. Ray, "A survey on internet of things architectures," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 30, no. 3, pp. 291–319, 2018.
[3] Q. Chi, H. Yan, C. Zhang, Z. Pang, and L. Da Xu, "A reconfigurable smart sensor interface for industrial WSN in IoT environment," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1417–1425, May 2014.
[4] C. Liu, Y. Xiao, V. Javangula, Q. Hu, S. Wang, and X. Cheng, "Normachain: A blockchain-based normalized autonomous transaction settlement system for IoT-based e-commerce," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4680–4693, Jun. 2019.
[5] J. Huang, L. Kong, G. Chen, M. Wu, X. Liu, and P. Zeng, "Towards secure industrial IoT: Blockchain system with credit-based consensus mechanism," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3680–3689, Jun. 2019.
[6] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka, "Security services using blockchains: A state of the art survey," *IEEE Commun. Surv. Tut.*, vol. 21, no. 1, pp. 858–880, Jan./Mar. 2019.
[7] "EU General Data Protection Regulation," Apr. 2016. [Online]. Available: https://www.eugdpr.org/eugdpr.org.html
[8] H. De Meer, H. C. Pöhls, J. Posegga, and K. Samelin, "On the relation between redactable and sanitizable signature schemes," in *Proc. Int. Symp. Eng. Secure Softw. Syst.*, 2014, pp. 113–130.
[9] J. Poncela *et al.*, "Smart cities via data aggregation," *Wireless Pers. Commun.*, vol. 76, no. 2, pp. 149–168, 2014.
[10] A. Urbieta, A. González-Beltrán, S. B. Mokhtar, M. A. Hossain, and L. Capra, "Adaptive and context-aware service composition for IoT-based smart cities," *Future Gener. Comput. Syst.*, vol. 76, pp. 262–274, 2017.

[11] P. Vlacheas *et al.*, "Enabling smart cities through a cognitive management framework for the internet of things," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 102–111, Jun. 2013.

[12] T. Wang, G. Zhang, M. Z. A. Bhuiyan, A. Liu, W. Jia, and M. Xie, "A novel trust mechanism based on fog computing in sensor–cloud system," *Future Gener. Comput. Syst.*, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17323658

[13] G. D. Hunt and L. Koved, "Auditing certified blockchain checkpoints," US Patent App. 15/422,980, May 31 2018.

[14] P. Vlacheas *et al.*, "An overview and main benefits of an intelligent knowledge-as-a-service platform," in *Proc. 34th Wireless World Res. Forum Meeting*, 2015.

[15] D. Kelaidonis *et al.*, "Cloud internet of things framework for enabling services in smart cities," in *Designing, Developing, and Facilitating Smart Cities*. New York, NY, USA: Springer, 2017, pp. 163–191.

[16] B. Faber, G. C. Michelet, N. Weidmann, R. R. Mukkamala, and R. Vatrapu, "BPDIMS: A blockchain-based personal data and identity management system," in *Proc. 52nd Hawaii Int. Conf. Syst. Sci.*, 2019, pp. 6855–6864.

[17] S. Hidano, S. Kiyomoto, Y. Murakami, P. Vlacheas, and K. Moessner, "Design of a security gateway for ikaas platform," in *Proc. Int. Conf. Cloud Comput.*, 2015, pp. 323–333.

[18] S. Hidano, A. R. Biswas, and S. Kiyomoto, "Hierarchical privacy cas for cross-border transfer of personal data," in *Proc. Int. Sump. Mobile Internet Secur.*, 2016, pp. 1–12.

[19] J. J. Seddon and W. L. Currie, "Cloud computing and trans-border health data: Unpacking U.S. and EU healthcare regulation and compliance," *Health Policy Technol.*, vol. 2, no. 4, pp. 229–241, 2013.

[20] F. Hörandner, S. Krenn, A. Migliavacca, F. Thiemer, and B. Zwattendorfer, "Credential: A framework for privacy-preserving cloud-based data sharing," in *Proc. 11th Int. Conf. Availability, Rel. Secur.*, 2016, pp. 742–749.

[21] M. Nalin *et al.*, "The european cross-border health data exchange roadmap: Case study in the italian setting," *J. Biomed. Informat.*, vol. 94, 2019, Art. no. 103183.

[22] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf

[23] J. Lindman, V. K. Tuunainen, and M. Rossi, "Opportunities and risks of blockchain technologies–a research agenda," in *Proc. 50th Hawaii Int. Conf. Syst. Sci.*, 2017, pp. 1533–1542.

[24] F. Glaser, "Pervasive decentralisation of digital infrastructures: A framework for blockchain enabled system and use case analysis," in *Proc. 50th Hawaii Int. Conf. Syst. Sci.*, 2017, pp. 1543–1552.

[25] M. Yang, T. Zhu, K. Liang, W. Zhou, and R. H. Deng, "A blockchain-based location privacy-preserving crowdsensing system," *Future Gener. Comput. Syst.*, vol. 94, pp. 408–418, 2019.

[26] A. Al Omar, M. Z. A. Bhuiyan, A. Basu, S. Kiyomoto, and M. S. Rahman, "Privacy-friendly platform for healthcare data in cloud based on blockchain environment," *Future Gener. Comput. Syst.*, vol. 95, pp. 511–521, 2019.

[27] A. Al Omar, M. S. Rahman, A. Basu, and S. Kiyomoto, "Medibchain: A blockchain based privacy preserving platform for healthcare data," in *Proc. Int. Conf. Secur., Privacy Anonymity Comput., Commun. Storage*, 2017, pp. 534–543.

[28] R. Frank, "ISO/TC 307-Blockchain and distributed ledger technologies," 2017. [Online]. Available: https://www.iso.org/committee/6266604.html

[29] S. Dhupkar, N. Mehta, H. Singh, and T. S. McGuire, "Collateral management with blockchain and smart contracts apparatuses, methods and systems," US Patent App. 16/125,608, Jan. 3, 2019.

[30] T. Hardjono, N. Smith, and A. S. Pentland, "Anonymous identities for permissioned blockchains," Jan. 21, 2019, 2014. [Online]. Available: https://petertodd. org/assets/2016-04-21/MIT-ChainAnchor-DRAFT.pdf

[31] Y. Chang, E. Iakovou, and W. Shi, "Blockchain in global supply chains and cross border trade: A critical synthesis of the state-of-the-art, challenges and opportunities," 2019, *arXiv:1901.02715*.

[32] T. Qiu, R. Zhang, and Y. Gao, "Ripple vs. swift: Transforming cross border remittance using blockchain technology," *Procedia Comput. Sci.*, vol. 147, pp. 428–434, 2019.

[33] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ECDSA)," *Int. J. Inf. Secur.*, vol. 1, no. 1, pp. 36–63, 2001.

[34] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using blockchain for medical data access and permission management," in *Proc. 2nd Int. Conf. Open Big Data*, 2016, pp. 25–30.

[35] H. Shafagh, L. Burkhalter, A. Hithnawi, and S. Duquennoy, "Towards blockchain-based auditable storage and sharing of iot data," in *Proc. Cloud Comput. Secur. Workshop*, 2017, pp. 45–50.

Mohammad Shahriar Rahman received the B.Sc. degree in computer science and engineering from the University of Dhaka, Dhaka, Bangladesh, in 2006, and the M.S. and Ph.D. degrees in information science from the Japan Advanced Institute of Science and Technology, Nomi, Japan, in 2009 and 2012, respectively,

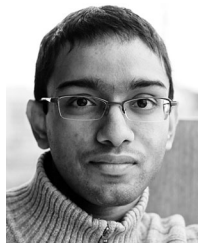
He is currently an Associate Professor of Computer Science and Engineering with the University of Liberal Arts Bangladesh.

Abdullah Al Omar received the B.Sc. degree in computer science and engineering from Department of Computer Science and Engineering (CSE), University of Asia Pacific (UAP), Dhaka, Bangladesh, in 2016.

He is currently working as a Lecturer of Computer Science and Engineering with his alma mater, CSE, UAP.

Md Zakirul Alam Bhuiyan received the Ph.D. degree in computer science and technology from Central South University, China, in 2013.

He is currently an Assistant Professor with the Department of Computer and Information Sciences, the Fordham University, New York, NY, USA, the Founding Director of Fordham Dependable and Secure System Lab (DependSys).



Anirban Basu received the Bachelor of Engineering (Hons.) degree in computer systems engineering and the Ph.D. degree in computer science from the University of Sussex, Brighton, U.K., in 2004 and 2010, respectively.

He is a Researcher with Hitachi R&D in Japan, and a Visiting Research Fellow with the University of Sussex.

Shinsaku Kiyomoto received the B.E. degree in engineering sciences and the M.E. degree in materials science from Tsukuba University, Tsukuba, Japan, in 1998 and 2000, respectively.

He is currently a Senior Researcher with the Information Security Laboratory of KDDI R&D Laboratories (now KDDI Research, Inc.).

Guojon Wang (M'08) received the B.Sc. degree in geophysics and the M.Sc. and Ph.D. degrees in computer science from Central South University, Changsha, China, in 1992, 1996, and 2002, respectively.

He is currently a Professor of Computer Science with the School of Computer Science and Educational Software, Guangzhou University.