2018 IEEE Confs on Internet of Things, Green Computing and Communications, Cyber, Physical and Social Computing, Smart Data, Blockchain, Computer and Information Technology, Congress on Cybermatics

Toward Privacy-Assured Health Insurance Claims

Xinchi He
Tandy School of Computer Science
University of Tulsa
Tulsa, OK
xinchi-he@utulsa.edu

Sarra Alqahtani
Tandy School of Computer Science
University of Tulsa
Tulsa, OK
sarra-alqahtani@utulsa.edu

Rose Gamble
Tandy School of Computer Science
University of Tulsa
Tulsa, OK
gamble@utulsa.edu

***Abstract*— According to HIPAA (Health Insurance Portability and Accountability Act), the medical insurance claim process is carried out by healthcare providers, insurance companies, and clearinghouses. The clearinghouse coordinates the medical insurance claims between providers and insurance companies. As centralized communication hubs, clearinghouses may maliciously or unintentionally leak patient information. In this work, we propose a distributed solution to replace the role of clearinghouses during health insurance claim process and mitigate the risk of data leakage among parties in Healthcare sector. Our solution enhances the patients' privacy protection through developing a HIPAA compliance system for the medical insurance claim process in a decentralized manner using blockchain technology. Blockchain ensures transaction integrity and anonymity of cryptocurrencies by using distributed immutable ledgers. We first design data structures for patient information, medical service record, insurance payment, and insurance agreements within the ledger. We then focus on defining smart contracts for privacy assurance, as well as automating the insurance claim process. We implement and evaluate the proposed framework with Hyperledger Fabric, showing promising performance and response time.**

Keywords—Blockchain, HIPAA, insurance claims, privacy

I. INTRODUCTION

The health insurance claim process is one of the most vexing problems of healthcare sector. Claims are prone to fraud and can consume the time and energy of the patient and healthcare provider. The claim process starts when a patient needs a service from a healthcare provider (e.g. physician and hospital). The provider uses the patient's insurance plan to determine initial service fees. To determine final service fees, the health insurer first validates services received from the provider against their shared payment agreement, accounting for various historical data points (e.g. deductible, copayment, and coinsurance). The insurer communicates the results to the patient and provider. Aggregating a patient's historical records across different providers to determine the shared cost can require significant time.

For example, when the patient has a back problem, her primary physician may refer her to a neurologist for an MRI. Her insurance plan requires a specific set of procedures to be undertaken before it will pay for the MRI, such as X-rays or physical therapy. In this case, the primary physician must examine the patient history and insurance plans before determining the next procedure. Providers might contact the insurer directly, which increases the patient's wait time for the addressing the problem. If the provider mistakenly assumes the insurance plan covers a procedure or the insurer gives imperfect requirements, then the patient may be required to pay out of pocket. The available solutions to automate this process help to alleviate the problem to some extent but they also introduce new issues related to patient and provider privacy.

Some large insurers allow providers to submit their claim information directly to their proprietary systems. The advantages are that the provider can submit a claim without a middleman. The direct claim also does not impose additional fees on the providers or insurers. However, submitting claims directly to each insurer's system vastly increases the opportunity for claim errors. This 'free' direct claim submission can become expensive in terms of lost claims, wasted time, billing errors, and claim denials.

Another alternative to direct claim submission is transmitting claims through a third-party system or clearinghouse. A clearinghouse is a centralized system for healthcare providers to transmit electronic claims to insurers securely, protecting patient health information by meeting the HIPAA privacy standards. The clearinghouse is responsible for scrubbing claims by checking them for errors and verifying they are compatible with the insurer's software. The clearinghouse validates that the designated procedure matches the diagnosis code submitted with the claim. One drawback of using the clearinghouse is that the providers and insurers share their sensitive information with a third-party system, which may result in data leakage between competitors. Another issue is that clearinghouses require an initial enrollment period prior to transmitting claims for the first time, which can take up to four weeks. If the insurer is not enrolled in the same clearinghouse as the provider, the claim is sent to a clearinghouse that the insurer is enrolled in. Transferring the claim between clearinghouses increases claim processing time, as well as the chances of it lost or have data leakage.

Permissioned blockchain is a good candidate to address the issues of traceability and transparency in a distributed health insurance claim system with multiple participants exchanging information and engaging in a collaborative manner. Compared to public blockchain, in which any party can join anonymously, permissioned blockchain acts as a gateway for participation and grants permission to join the network or initiate transactions [1]. Blockchain, in general, relies on a distributed and shared database called a *ledger* that consists of a linked sequence of blocks, holding secured time-stamped transactions [2]. By keeping all transactions in the ledger, aggregating information

978-1-5386-7975-3/18/$31.00 ©2018 IEEE
DOI 10.1109/Cybermatics_2018.2018.00273

from different providers can be performed, allowing the insurer to make decisions spontaneously. Another important blockchain component is the *smart contract* (SC), which is a self-executing script to allow business logic or legal contracts to be executed on the distributed ledger. SCs can be used to automate the insurance policies and the access control rules.

Traditional distributed databases differ from blockchain in mutability and read/write management [3]. While the records in a distributed database are usually mutable, they are immutable in the distributed blockchain ledger. The only way to update the ledger records is to append new blocks to it. Immutability improves the integrity of blockchain solutions. Furthermore, read/write management is administrated by a centralized system in traditional distributed database. Whereas in blockchain, read/write authorization is decentralized without trusting any third parties. This feature is beneficial to the health insurance claim system because there are multiple healthcare providers and insurers involved in transactions and they do not necessarily trust each other for transaction security.

By using blockchain technology, we design a decentralized insurance claim system to replace the clearinghouse. The distributed ledger provides the platform for healthcare stakeholders to interact, while serving as a transparent, comprehensible repository of insurance claims' transactions. The transparency of blockchain allows healthcare participants to efficiently monitor the claim workflow, detect frauds, and resolve claim disputes. However, the full transparency of an open ledger would violate the privacy of patients and providers. According to HIPAA, the patient's information must not be shared or seen by any party without the patient's permission. Most healthcare providers have specific permissions required by patient that allow them to share the data with other providers for comprehensive healthcare, forcing the allowance of some data exchange with other providers. Also, the payment information can be leaked between providers who share an open blockchain ledger. In the proposed solution, we design and prototype a SC to address data leakage given a shared ledger used to automate the insurance claim process. Privacy is managed using the access control policies to indicate what each stakeholder can create and retrieve from the ledger based on its role and authority. The second task of the SC is to automate the insurance claim process by implementing the logic necessary to construct the provider and insurer agreements as well as the patient and insurer agreements based on ledger information. Once the SC is deployed to blockchain, it becomes executable and the agreements are confined to only the appropriate stakeholders.

The next section discusses the related work of blockchain and its healthcare systems. Section III discusses the proposed system, Section IV presents the implementation details while Section V shows the evaluation results.

II. Related Work

Privacy and security issues are listed as one of the challenges in current healthcare industry. The issues include confidentiality of patient's information, data integrity and availability, trust, and access control. Blockchain technology has been shown to provide enhanced security by encryption and cryptology, assured integrity through the distributed ledger, enforced permission for participants, and authenticated data exchange [4].

To improve the health insurance claim process, Culver [5] believes blockchain and the SC can be used to fully automate non-complex claims because the business logic is relatively simple, and the automation can help save significant costs. Advanced Blockchain [6] is an architecture for e-Health to provide reliable, secure and efficient electronic health record (EHR) exchange using blockchain. A new data structure is introduced for EHR storage and a two-dimensional processing methodology is used to address the activities for patients and service providers. The defined data structure contains only limited medical service record and billing attributes. Similarly, a process flow to use blockchain in health care sector has been proposed to enable interoperability among patients and health care related organizations [7]. Providers could store the service record in the ledger. Then, related organizations could query the records to perform the next actions. Stagnaro introduces the "on chain" and "off chain" concepts for ensuring minimum ledger usage for scalability and performance. No detailed solutions or associated evaluations are provided.

In [8], Peterson argues that it is essential and challenging to share healthcare data between institutions. Blockchain technology is adopted in this work to support the interoperability for different institution. However, no additional security or privacy protection methodology is considered, such as access control. MedRec is introduced by Azaria, et al. [9] in a decentralized record management mechanism with blockchain technology for electronic medical records (EMRs). Authentication, confidentiality, accountability and data sharing is handled within such framework. It is believed HIPAA regulation and other compliances can be beneficial by using MedRec. However, patient's EHRs and related EMRs that being logged in the ledger still raise privacy concerns without enforcing protection mechanism.

Guo, et al. [10] demonstrate an attribute-based signature scheme with multiple authorities to secure the privacy of patient's EHR, thus preventing the collusion attacks by sharing pseudorandom function seeds among authorities. EHR is one of essential components that needed during the health insurance claim process, thus this protocol can be easily adopted to our architecture to mitigate EHR security and privacy concerns. In order to integrate blockchain for data sharing and collaboration in mobile healthcare application, authors in [11] use permissioned blockchain and segregated channels to ensure privacy protection for interactions between user and healthcare institutions when used through mobile applications. Hyperledger Fabric is used for the implementation.

III. Using Blockchain for Insurance Claims

The system design presented on uses a shared ledger to retain financial transactions for health insurance claims along with the medical services received by the patients generating those claims and, thus, involving the health provider, insurer(s), and patient. Each entity has access to the blockchain according to their roles. The patient sees the service history and financial transactions for each claim. Providers and insurers see the historical medical service records and payment information only when they have the corresponding access privileges.

A. System Architecture

Figure 1 presents the general architecture of the proposed system designed using the two-dimensional blockchain structure found in [6]. Healthcare stakeholders access the distributed ledger to query or post financial transactions and medical history through the blockchain client. Concurrently, an external web service enables the message passing between different parties for notification purposes. For example, a patient visits her primary physician, and then the physician refers the patient to a specialist for further diagnosis and treatment. After seeing the patient, the primary physician creates a new medical service record in the ledger and uses the external messaging system to notify the specialist about the referral and the patient's insurer about the bill.

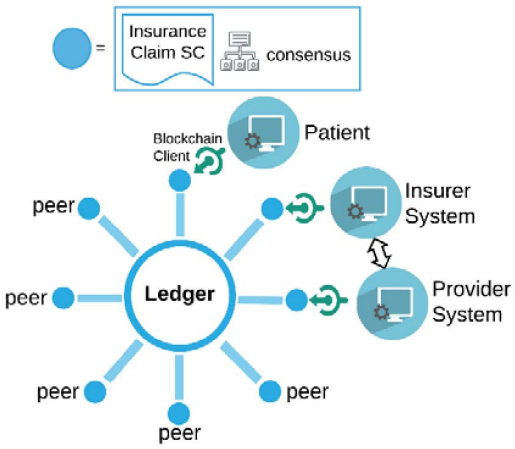

Fig. 1. Health insurance claim system using blockchain

Separating the messaging system from the blockchain architecture increases the scalability of the solution. The minimum usage of blockchain should be considered at design time as the periodical queries coming from different institutions. These queries check whether there are new transactions that would unnecessarily burden blockchain network, impacting overall performance. Another reason for separation is to maintain compatibility with the legacy communication systems [12] that exist between health care institutions.

Within the architecture design displayed in Figure 1 is the representation of the healthcare institutions, their peers, clients, and the ledger. Each peer node is owned by a single institution and holds the same copy of the immutable ledger and the insurance claim SC. The blockchain client is a gateway to receive API calls from different institutions and invokes the SC to interact with the ledger. Its consensus algorithms ensure transaction integrity. For our system, each peer must have the insurance claim SC installed during the registration phase.

B. Ledger Entries

As mentioned previously, one insurance clearinghouse function is scrubbing the claims sent by providers to enhance their compatibility with the insurer's system. It is possible for the clearinghouse to maliciously or mistakenly leak patient medical information and insurance information to adversaries or competitors. Within our design, the decentralized claim system overcomes these issues by (a) standardizing the data structure for the patient information, claim information, insurance policies, and agreements and (b) maintaining transparent claim details to all participants under least privilege to preserve the individual's privacy. The ledger strictly adheres to the claim strict data structure that is maintained and checked by the SC. There are three different types of ledger entries: (i) patient healthcare records, (ii) insurance claim records, and (iii) logs for agreements between providers and insurers.

The data structure for the patient's healthcare information is shown in Figure 2. We initially assume that the patient information is already present in the ledger based on their insurance plans. Each patient record consists of a header, access control list (ACL), insurance plan, and protected health information (PHI). A patient ID indexes the patient record. The ACL secures the patient's privacy by listing only authorized providers and their access rights (read/write/both). Chart ID identifies the health problem to be treated. Expiration is when the access control becomes invalid. The Insurance plan contains a list of health insurance plans that the patient holds, as represented by insurer ID and policy information. The PHI section contains patient's historical health information.

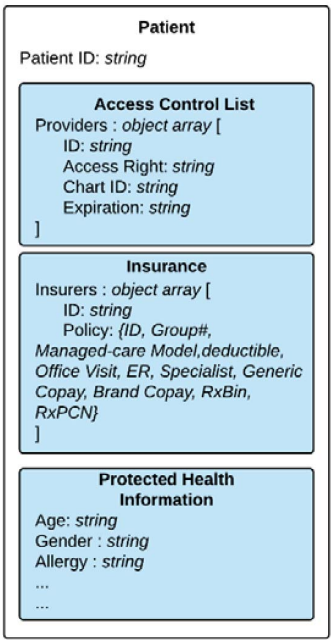

Fig. 2. Data structure for patient's information

Figure 3 (a) shows the data structure to log medical and financial information in the ledger during the insurance claim process. Each claim entry consists of header, provider, and insurance sections. The header section contains the Timestamp of the entry, Chart ID, Transaction ID for the payment transaction, Service ID indicating the medical service performed, Provider ID, Medical code to identify the diagnosis and treatment procedures, Bill, Insurer ID, Paid Amount by each contracted insurer, Status of insurer response (initiated, accepted, pending, or rejected) Notes, and Cost Sharing indicating the amount of the bill that the patient pays.

The data structure for the claim entry creates the medical service records and logs the insurance payment transactions.

Entries in the ledger cannot be updated, forcing a new entry to be created to log each update. Thus, a new claim entry leaves unrelated fields empty. For instance, a medical service entry leaves the insurance fields of the entry empty. A separate design layout to divide the claim entry into two data structures increases the complexity of the SC to correlate the entries.

To automate the insurance claim process, we define another data structure representing agreements between providers and insurers in blockchain. Inspired by our usage policy in [13], we express an initial structure of the agreement's rules in executable XML. Further investigation will generalize the policy rules to cover all cases in the insurance agreement plans. The agreement data structure appears in Figure 3 (b) with a simple example of the policy rule that indicates "MRI procedure (name) is not covered until procedures of X-Ray (prev) and physical therapy (prev) have been already applied with an exception of patients over (ge) 60 years old".

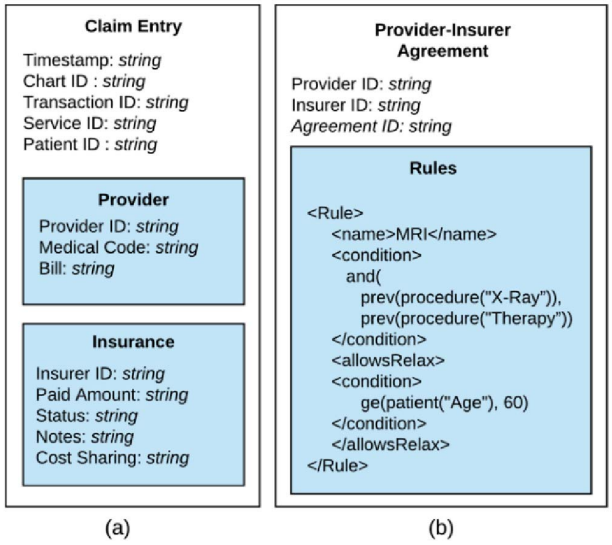

Fig. 3. Data structure for the insurance claim (a) and agreement (b) in the ledger

Table 1 shows possible queries that each entity in the system can invoke. The provider can create new medical service records in the ledger for the patient. The provider can retrieve all medical history of the patient's chart identified by the Chart ID. Either during the insurance claim process or when bills are finalized, the provider can also query the related insurance transactions using its medical Service ID but cannot see the Bill or payment to other providers for the same chart.

Insurers can manually create payment transactions or automate the task with the SC according to the agreements between patients and providers. When an insurer is notified of an incoming insurance claim or retrieves a patient's pending claims and related medical history through their API calls to the ledger, the insurer can retrieve the relevant medical bills and post back to the ledger the amount paid according to the policy. A patient can retrieve all of the on-going entries related to the medical chart and payments. Moreover, the patient is responsible for granting and revoking access rights for and from providers. By allowing the patient to do so, our solution becomes a patient-centric system to fully support HIPPA rules.

TABLE 1. PERMITTED QUERIES FOR HEALTHCARE INSTITUTIONS

Entity	Query
Provider	createService(Chart ID, patient ID, provider ID, Medical Code)
	getMedicalHistory(Chart ID, provider ID)
	queryPayment(Service ID, provider ID)
Insurer	createTransaction(Service ID, Insurer ID, Paid Amount, Status, Note)
	getHistory(Patient ID)
Patient	getChartInfo(Chart ID)
	grantAccess(Provider ID, Access Right, Chart ID, Expiration)
	revokeAccess(provider ID)

Assume a patient goes to St. John Urgent Care and is referred to St. John Hospital for further diagnosis and treatment. The patient has Blue Cross Blue Shield (BCBS) and Aetna health insurance plans, with BCBS providing the primary coverage. The patient is billed $600 at St. John Urgent Care clinic and $6,800 at St. John Hospital.

Figure 4 shows the first medical service record posted by St. John Urgent Care. Line 2 is the local time when the service record posted to the ledger, Line 3 is the medical chart ID. Transaction ID is bland in Line 4 because it relates to when the insurer posts financial information to the ledger. Line 5 is the service ID for this medical service record and Line 6 is the hashed patient ID. Lines 8-10 are the provider's ID, medical code and total amount of the bill of the medical service. Line 14-17 and 20-23 shows BCBS and Aetna are listed as patient's health insurance plans and are ready to receive claims.

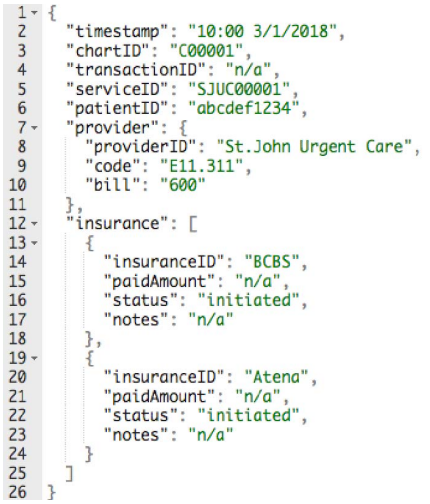

Fig. 4. Provider view for entry creation

The provider at St. John hospital retrieves the medical service record with patient's access token provided by the patient as shown in Figure 4, in which the SC automatically masks another provider's bill to inhibit provider to provider data leakage. When the patient completes treatment at St. John hospital, Entry 2 in Table 2 will be posted to the ledger for a new service record. This new service record contains new medical code and new medical bill amount.

TABLE 2. CASE STUDY ENTRIES IN THE LEDGER

	Entry 1	Entry 2	Entry 3	Entry 4	Entry 5	Entry 6
Timestamp	10:00 3/1/2018	13:00 3/2/2018	9:31 3/10/2018	16:21 3/11/2018	8:17 3/12/2018	12:52 3/12/2018
Chart ID	C00001	C00001	C00001	C00001	C00001	C00001
Transaction ID	n/a	n/a	BCBS_00001	BCBS_00002	AT_00001	AT_00002
Service ID	SJUC_00001	SJ_00001	SJUC_00001	SJ_00001	SJUC_00001	SJ_00001
Patient ID	abcdef1234	abcdef1234	abcdef1234	abcdef1234	abcdef1234	abcdef1234
Provider ID	St. John Urgent Care	St. John Hospital	St. John Urgent Care	St. John Hospital	St. John Urgent Care	St. John Hospital
Medical Code	E11.311	E11.311.1A	E11.311	E11.311.1A	E11.311	E11.311.1A
Bill	$600	$6,800	$600	$6,800	$600	$6,800
Insurance ID	BCBS, Atena	BCBS, Atena	BCBS	BCBS	Atena	Atena
Paid Amount	n/a, n/a	n/a, n/a	$400	$5,500	$100	$500
Status	initiated, initiated	initiated, initiated	accepted	accepted	accepted	accepted
Notes	n/a, n/a	n/a, n/a	n/a	n/a	n/a	n/a

When BCBS and Aetna are notified by the external message system for insurance claims by chart ID C0001 in corresponding medical service records, BCBS and Aetna will query the billing information and Entry 1 and 2 will be shown as the results. When BCBS post the paid amount, Entry 3 and 4 will be posted in the ledger. Similarly, Entry 5 and 6 will be posted to the ledger when Aetna pays the medical bills.

Chart ID C00001 and service ID SJUC00001 will be used when querying insurance payments from St. John Urgent Care, Entry 1, 3 and 5 will be shown as the results to display the original bill and insurance payments by two insurance companies. Similarly, Entry 2, 4, 6 will be shown as the results when St. John hospital query for the insurance payments by using chart ID C00001 and service ID SJ00001.

When all the bills are paid, and the patient would like to gather all the information related to both provider and insurer, chart ID and patient ID will be used to retrieve all available entries related to the medical case, and Entry 1 to 6 will be all shown as the results.

C. Access Control and Privacy

A distributed access control policy is developed for the blockchain using the SC. Given the natural latency issue in blockchain, identifying and enforcing the access rights of each healthcare stakeholder is done at the access request time. The developed access control policies regulate the access to critical or valuable resources, which are represented as medical charts. We define our access control policies using attribute-based access control (ABAC) model [14].

An ABAC policy combines a set of rules expressing conditions over a set of attributes paired to the stakeholders, to the charts or to the insurance policies and agreements. The rules must be satisfied accordingly for the access right to be granted. The previously described data structures implement and evaluate the access control policies in real-time. In other words, the access rights are embedded in the ledger data. The patient ACL is used to authorize providers while the insurance policies in the patient's data are used to authorize the insurers. The simplified form of the ABAC rule as follow:

[
Subject: *Stakeholder ID*
Resource: *Chart | Service | Patient | Agreement*
Action: *read | write*
Condition Set: *Boolean constraints*
]

In Figure 5, the access control policies are developed to authorize the healthcare stakeholders for queries listed in Table 1. In (1), the provider is authorized to create a service for a specific patient only when the patient has granted this provider "write" access on either the specified Chart ID or on "all" of the patient's charts and this access right has not expired. Recall that blockchain does not allow entries to be updated, hence the ABAC rule in (1) uses the latest entry of the patient's access control list to authorize the providers. The ABAC rule in (2) is similar to (1) but with "read" access. In (3), the provider can see the payment transactions that are only related to its medical services. For the insurer, ABAC in (4) and (5) authorizes the insurer only when the patient has an insurance plan with it. Since all data in blockchain is patient-centric, there is no authorization rule for the patients to see their claims or medical history, grant, or revoke access. Instead, verifying the patient's identity during the authentication process is a mandatory.

Our approach adheres to the concept of sharing the same ledger between all healthcare stakeholders. However, an obvious drawback of this approach is improper leakage of financial information across providers that are part of the same medical chart, starting with the one that initiates the chart and including those that the patient is referred to. Specifically, billing information and cost rates must be private for each provider. We embed into the SC a privacy control rule to hide financial information from other providers in the shared chart. Upon receiving the query *getMedicalHistory(Chart ID, provider ID)*, the SC filters out the bill amount. This rule assumes that all blockchain peers must have installed the SC that maintains this rule in addition to the access control rules. Besides encrypting the transactions, providers and insurer cannot directly access to the blockchain ledger on each peer node but through the access control smart contract enforcement, thus both medical and patient's information will be ensured for no leakage even though it is broadcasted.

D. Insurance Claim Process

The defined data structures in Figures 2 and 3, along with the queries in Table 1, the sharing cost estimation, and the insurance claim payment can be automated using the SC. The sharing cost estimation is an important application of the proposed approach since this process is still manually conducted in some cases. Using our solution, the provider can automatically get the sharing cost estimation once the patient grants it the access right to do so. The flow of the sharing cost estimation process is shown in Figure 6. The process starts

when the provider sends an estimation request to the blockchain. Then, the insurance claim SC, which has been installed in the provider's peer, receives this request and evaluates the authorization of the provider using the ABAC (1) in Figure 6. If the provider is not authorized by the patient to access the medical chart with Chart ID, then it rejects the request. Auditing this request will be part of the future work. For an authorized provider, the SC retrieves the patient's insurance policy from the ledger by searching Patient entries. The SC uses the agreement between the provider and the patient's insurer to find the cost rate of the recommended procedure defined by the Medical Code (step 5). It is important to mention here that this workflow does not cover the cases when the patient is insured by more than one insurance plan. This direction will be explored in future work. In step 7, The SC computes the sharing cost that the patient would pay using her historical information in the ledger. To do so, the SC uses the query of *getHistory(Patient ID)* to retrieve all payment transactions for the patient in order to find her deductible, copay and other costs. The estimated shared cost is finally returned to the provider in step A.1.

Automating the insurance claim payment becomes possible using the shared ledger and the SC. The distributed ledger keeps track of the historical payment information, the agreements between providers and insurers, and the insurance policies for the patients. The SC enforces the latest update of the agreements and insurance policies to issue an automatic payment to the providers or reject payment for the service on behalf of the insurer. In some critical cases, the decision requires human intervention by the insurer or through benefit coordination between insurers, which is usually performed to decide the cost rate when the patient is insured by more than one insurance plan. In these cases, the SC returns *"pending"* payment status.

The automatic payment workflow in Figure 6 repeats steps 1-6 of the cost sharing estimation process. In step B-1, the medical service is created. The SC calculates cost sharing using the patient's financial history and insurance policies in step 7. The payment is automatically issued in step B-2 followed by creating a transaction with the paid amount, related status and notes about the payment in step B-3. Finally, the information about the payment is returned to the provider including the payment details in step B-4.

IV. Implementation

We deployed Hyperledger Fabric version 1.0.5 as the blockchain framework to implement the proposed health insurance claim platform as it provides permissioned blockchain with certificate authorization. In addition, Hyperledger Fabric is a business blockchain framework, which can be deployed by defining organizations and peer nodes in templates.

Our case study starts with four organizations. Two peer nodes for each organization are defined. Two of the organizations represent the healthcare providers (Provider 1 and Provider 2), and the other two organizations are the insurance companies (Insurer 1 and Insurer 2) as shown in Figure 7. Moreover, each organization has one Certificate Authority (CA) service

1) Query: createService
ABAC Authorization Rule:
[
Subject: provider ID
Resource: patient ID
Action: write
Condition Set:
p= Latest(Patient(patient ID))
providers= p.AccessControlList.Providers
provider ID ∈ providers.ID
∧ provider.AccessRight == write
∧ (provider.ChartID == ChartID
∨ provider.ChartID == "all")
∧ SystemTime < provider.Expiration
]

2) Query: getMedicalHistory
ABAC Authorization Rule:
[
Subject: provider ID
Resource: chart ID
Action: read
Condition Set:
p= Latest(Patient(Entry(Chart ID).patient ID))
provider = p.AccessControlList.Providers(provider ID)
provider ≠ ∅
∧ provider.AccessRight == read
∧ (provider.ChartID = ChartID
∨ provider.ChartID == "all")
∧ SystemTime < provider.Expiration
]

3) Query: createTransaction
ABAC Authorization Rule:
[
Subject: Insurer ID
Resource: Service ID
Action: write
Condition Set:
Insurer ID ∈ Patient(Entry(Service ID).patient ID).Insurers
]

4) Query: getHistory
ABAC Authorization Rule:
[
Subject: Insurer ID
Resource: Patient ID
Action: read
Condition Set:
Insurer ID ∈ Patient(patient ID).Insurers
]

5) Query: queryPayment
ABAC Authorization Rule:
[
Subject: provider ID
Resource: Service ID
Action: read
Condition Set:
Entry(Service ID).provider ID == provider ID
]

Fig. 5. ABAC rules for blockchain queries

as the security gateway to register users and issue security tokens that interface with external APIs. A replicated Kafka ordering service is deployed with Apache Kafka cluster and its respective Apache ZooKeeper ensemble to provide production environment ready robustness and resilience with Crash Fault Tolerance consensus [15]. Apache Kafka is a distributed streaming platform. Apache ZooKeeper is a blockchain setup, the developed SC (chaincode) is installed on each peer node.

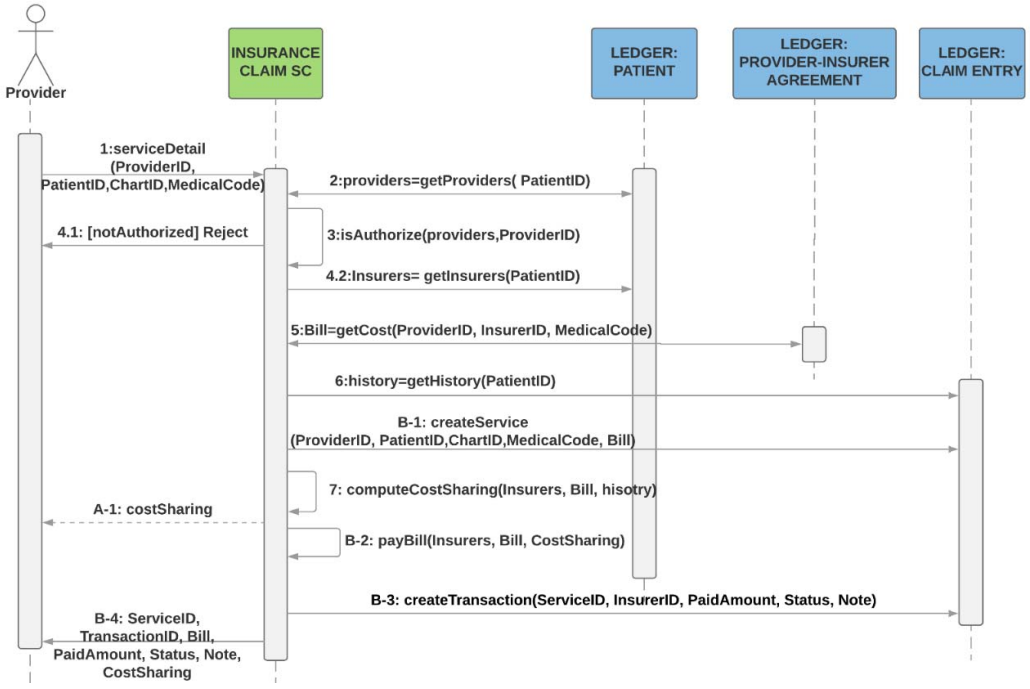

Fig. 6. The workflow of the sharing cost estimation

The blockchain network architecture in Figure 7 for the developed case study has the Peer 0 node in each organization as the anchor peer to bridge the communication across organizations. Four CAs register users and issue security tokens for client requests. A Hyperledger Fabric client is deployed in the cloud to expose the blockchain network as RESTful APIs for doctors, insurance agents, and patients to invoke queries on the ledger. The procedure is started by defining the network architecture using Hyperledger templates, and then creating certificates and channel artifacts to prepare for the network launch. Docker containers (software to provide operating system level virtualization) are launched, including an ordering service, peer nodes, and a CA. Once the blockchain network is deployed, we create a public channel and asked all peers to join. After peers join the channel, chaincode is instantiated to be ready for external parties to interact with the business logic in the blockchain network through the Hyperledger Fabric client.

V. Evaluation

We programmed simulations to evaluate the performance and scalability of our solution over a timer period in which the ledger has from 1 to 10,000 entries with both solo ordering and Kafka ordering. The medical charts were randomly generated to contain 10 to 20 medical service records and insurance payments to simulate real-world health insurance claim entries. Three probes were set to record response time of the API calls at entry creation, entry retrieval, and retrieval of the whole medical chart, respectively. We expected to observe a fairly consistent response time from all three probes as Hyperledger Fabric maintains a cached world state in the embedded database to avoid scanning the entire ledger to retrieve a certain state and exhausting the system resources.

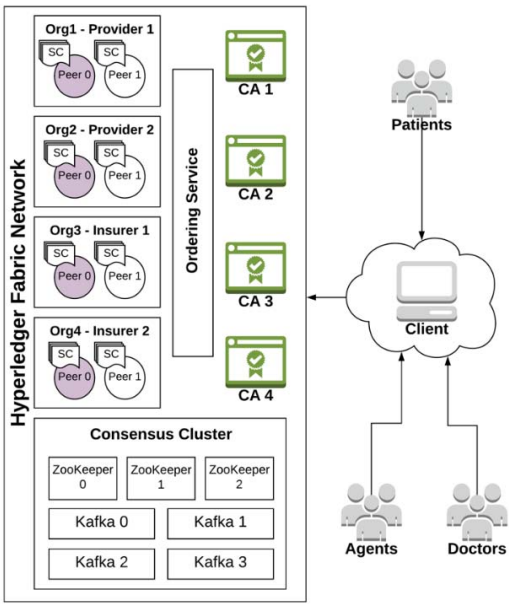

Fig. 7. Implementation architecture with Hyperledger Fabric

Figures 8, 9 and 10 show the average response time for entry creation, entry retrieval, and medical chart retrieval, respectively, when the ledger holds 100, 200, 500, 1000, 2000, 5000 and 10000 entries with both solo ordering and Kafka ordering as the consensus mechanism. All three figures show that the average response time tends to stabilize as the entry increases in the ledger and achieving a consistent response time.

Kafka ordering takes slightly longer for response time than that of solo ordering because the consensus requires more overhead, and solo ordering has no consensus for development and testing purposes. By applying the Kafka ordering with the Crash Fault Tolerance consensus algorithm to enhance robustness and resilience of the system, the response time is not increased significantly. Thus, the evaluation suggests that our solution is feasible with acceptable performance for the proposed insurance claim process even when the ledger entry amount increases. When compared to the average response time in 100 and 10000 entries, all three evaluations show the difference less than 100ms. Hence, the average response time through all the ledger is relatively stable.

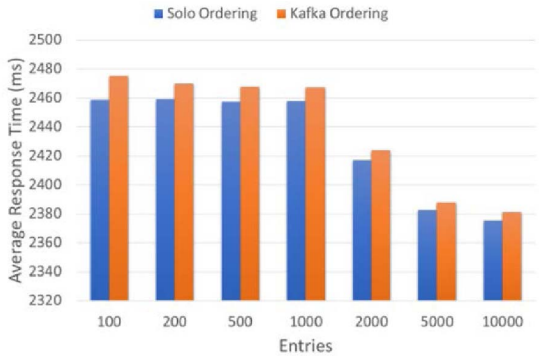

Fig. 8. Average response time for entry creation

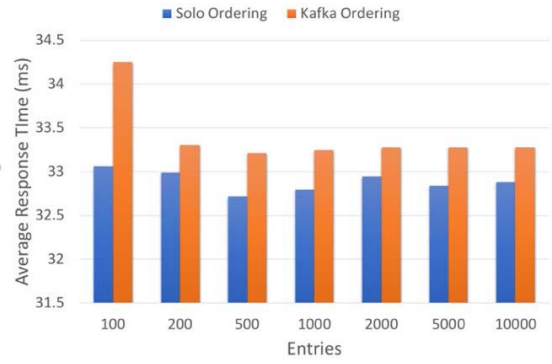

Fig. 9. Average response time for entry retrieval

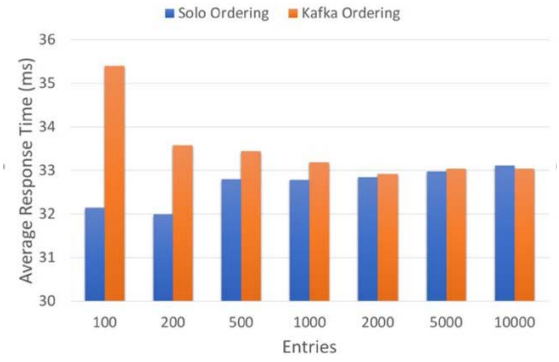

Fig. 10. Average response time for medical chart retrieval

VI. Conclusion & Future Work

In this paper, we design a prototype health insurance claim system using blockchain technology. We describe data structures to store patient information, medical service record, and insurance payments, as well as provider-insurer agreements in the distributed ledger. Attribute-based access control mechanism identifies the access rights for each stakeholder for privacy assurance. We use the Hyperledger Fabric blockchain to implement and evaluate the system design. The results indicate that our approach is feasible and efficient in terms of response time. For future research, we will improve our current work with multi-channel formation for better performance and scalability, as well as focus on the automation on health insurance claiming process to provide a robust and efficient workflow to increase claim efficiency and reduce operational costs

References

[1] X. Xu *et al.*, "A Taxonomy of Blockchain-Based Systems for Architecture Design," presented at the 2017 IEEE International Conference on Software Architecture (ICSA), Gothenburg, 2017, pp. 243-252.

[2] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2009, unpublished.

[3] G. Peters and E. Panayi, "Understanding Modern Banking Ledgers Through Blockchain Technologies: Future of Transaction Processing and Smart Contracts on the Internet of Money," in *Banking Beyond Banks and Money*, 2016, pp. 239-278.

[4] IBM, "Blockchain: The Chain of Trust and its Potential to Transform Healthcare – Our Point of View," 2016, unpublished.

[5] K. Culver, "Blockchain Technologies: A whitepaper discussing how the claims process can be improved," 2016, unpublished.

[6] W. Liu, S. S. Zhu, T. Mundie, and U. Krieger, "Advanced block-chain architecture for e-health systems," presented at the 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), Dalian, 2017, pp. 1-6.

[7] C. Stagnaro, "White Paper: Innovative Blockchain Uses in Health Care," 2016, unpublished.

[8] K. Peterson, R. Deeduvanu, P. Kanjamala, and K. Boles, "A Blockchain-Based Approach to Health Information Exchange Networks," 2016, unpublished.

[9] A. Azaria, A. Ekblaw, T. Vieira, and T. Lippman, "MedRec: Using Blockchain for Medical Data Access and Permission Management," presented at the 2016 2nd International Conference on Open and Big Data (OBD), Vienna, 2016, pp. 25-30.

[10] R. Guo, H. Shi, Q. Zhao, and D. Zheng, "Secure Attribute-Based Signature Scheme With Multiple Authorities for Blockchain in Electronic Health Records Systems," *IEEE Access,* vol. 6, 2018, pp. 11676-11686.

[11] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," presented at the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), 2017, pp. 1-5.

[12] F. Holotiuk, F. Pisani, and J. Moormann, "Unveiling the Key Challenges to Achieve the Breakthrough of Blockchain: Insights from the Payments Industry," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018, pp. 3537-3546.

[13] A. Marshall *et al.*, "Combining coordination with usage policies to improve mission scheduling resilience," presented at the 2015 Resilience Week (RWS), Philadelphia, PA, 2015, pp. 1-5.

[14] (2013). *Guide to attribute based access control (ABAC) definition and considerations (draft)*. Available: https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-162.pdf

[15] J. Sousa, A. Bessani, and M. Vukolić, "A Byzantine Fault-Tolerant Ordering Service for the Hyperledger Fabric Blockchain Platform," 2017, unpublished.