IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 29, 2018, accepted August 2, 2018, date of publication August 14, 2018, date of current version September 21, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2865535

Medshare: A Novel Hybrid Cloud for Medical Resource Sharing Among Autonomous Healthcare Providers

YILONG YANG[1], XIAOSHAN LI[1], NAFEES QAMAR[2], PENG LIU[1], WEI KE[3], BINGQING SHEN[1], AND ZHIMING LIU[4]

[1]Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau 999087, China
[2]Department of Health Administration, Governors State University, University Park, IL 60484 USA
[3]Macau Polytechnic Institute, Macau 999087, China
[4]School of Computer and Information Science, Southwest University, Chongqing 400715, China

Corresponding author: Zhiming Liu (zhimingliu88@swu.edu.cn)

This work was supported in part by the Macau Science and Technology Development Fund (FDCT) under Grant 103/2015/A3 and Grant 018/2011/A1, in part by the University of Macau under Grant MYRG 2017-00141-FST, in part by the 1000-Expert Program under Grant SWU116007, and in part by the National Natural Science Foundation of China under Grant 61562011, Grant 61672435, and Grant 61732019.

ABSTRACT Legacy electronic health record systems were not developed with the level of connectivity expected from them nowadays. Therefore, interoperability weakness inherent in the legacy systems can result in poor patient care and waste of financial resources. Simultaneously, healthcare providers are not yet ready to dispose of them. Large hospitals are also less likely to share their data with external care providers due to economic and political reasons. To overcome the barriers in the effective medical data exchange process, we present a novel hybrid cloud called MedShare, dealing with interoperability issues among disconnected but autonomously functioning healthcare providers. The proposed system architecture and its implementation is based upon: 1) custom data extractors to extract legacy medical data from the three hemodialysis centers under consideration; 2) negotiated and converted to a common data model in each of the private cloud of a provider; 3) indexed patient information using the HashMap technique into the public cloud that operates on private clouds, called a hybrid cloud; and 4) a set of services and tools installed as a coherent environment to exchange information smoothly. This paper enables healthcare professionals to appropriately access and securely share a patient's medical information. MedShare allows the healthcare providers and administrators to maintain the control of their patient data, which is always the primary concern in building a trustworthy environment for exchanging patient information. Medshare effectively addresses primary security and privacy concerns surrounding the deployment of data exchange process by including patient consent and a two-way authorization process.

INDEX TERMS Electronic health records (EHRs), health information exchange (HIE), healthcare providers, cloud computing, hybrid cloud, patient privacy, health care resource sharing.

I. INTRODUCTION

Legacy EHR systems have been mostly designed and implemented to meet the internal clinical needs of healthcare providers that have become obsolete and no longer meet the external needs of patients and local governments. Consequently, it stands in the way to an improved patient care and in offering a broad range of medical services, resulting in increased cost and clinical negligence. The future health information systems aim at the integration, interoperability, innovation, and intelligence [12], [24] for sharing the resource. Health Information Exchange (HIE) has seamlessly paved the way for introducing medical standards [2], [4], [10] that provide with a unified approach to medical vocabulary and exchange of information, but none of them has come of age to be used smoothly under local constraints. For example, a study [21] finds weak evidence of the 'meaningful use program (MU)' initiated by the 2009 Health Information Technology for Economic and Clinical Health (HITECH) Act on EHRs uptake due to data interoperability challenges. The study [36] presents the top ten technical issues in healthcare, which include privacy, quantity, security, and the implementation of electronic medical records. Moreover, the political and economic issues and healthcare providers of contingent factors [25] should take into an account in the development of medical information sharing.

Large medical-care providers seem reluctant to share their patient data with other healthcare providers [22]. They exchange patient information internally and are less likely to cooperate outside their network [22]. In this scenario, designing and developing an interoperable HIE system becomes a non-trivial task. It is not only because of complex workflows involving data acquisition, storing, communication, and manipulation, but also lacking in a coordinated effort to connect autonomous healthcare providers.

Albeit, healthcare networks are expected to: (a) support direct data exchange, (b) query-based exchange of patient-related information in an emergency situation, medication history, radiology reports and records of a diseased person hospitalized for emergency care, and (c) personalized patient data management by patients themselves like online banking. Architecting and implementing such an interoperable system, meeting the aforementioned requirements, needs a comprehensive and multifaceted approach to solve both technical and non-technical problems.

Our work is focused on connecting three individually operated healthcare providers in Macau SAR that are Hospital Conde S. Januário (HC), Kiang Wu Hospital (KW) and Macau University of Science and Technology Hospital (UH). However, the contribution of our work has wider implications and scope to build HIE systems confronting the similar challenges. Neither the autonomous EHR systems under consideration were developed using special instructions, nor were standards at the time of their birth. The concerned authorities are also not to ready to update their legacy systems, since the three hemodialysis centers have their fully functional and independent electronic health records in place. It is noteworthy to mention that two of the three hospitals participating in this work are private healthcare providers.

In the described medical settings, distributed information sharing is mandatory for effective patient care and monitoring where patients may want to switch a healthcare provider due to personal and financial reasons. MedShare is a simple yet robust EHR system developed to exchange medical resources for an improved patient care between isolated hemodialysis centers. The types of data shared in MedShare includes lab reports, radiology images, transcription reports and medication histories. MedShare works with legacy EHR systems in three steps: 1) it uses a data extractor to regularly extract legacy data of a patient located at a hemodialysis center, 2) it converts the data to a unified data format agreed upon by all the stakeholders and medical providers belonging to three hospitals, 3) it indexes the patient information in the trusted cloud by using the HashMap technique. Our approach integrates a set of services and tools that can be installed as a coherent environment on top of standalone EHRs. As discussed in [16], Operational Data Model (ODM) lacks explicit support for modern exchange mechanisms, our authentication mechanism is based on Representational State Transfer (RESTful) web services as our previous work that employs the same techniques to exchange medical information [27]. MedShare implementation, a hybrid cloud based EHR sharing system as shown in Fig. 1, promptly responds to dynamic data exchange requests such as the one detailed below:

Example: A doctor can request the hemodialysis records from all participating sources of a patient. The EHR sharing system returns a date-wise list of all the hemodialysis records of a queried patient. Furthermore, the doctor can access a detailed EHR on a specific date. MedShare allows an administrator to track the potential leaks in the system. For instance, when the system auditor wants to know the information accessed on a patient with ID 0221, the system will display the results based on the selected criteria. Hence, MedShare facilitates a distributed patient care, but it can also assist with distribution of hemodialysis tasks among the Macau hospitals transparently and securely. In the process, one of the challenging tasks was to identify the data exchange scenarios, capturing the intent behind and to identify the collaborating entities in a given scenario. These and other system goals are achieved by developing the system components such as authentication, EHR query, synchronization, and audit process.

Innovations: Comparing with state-of-the-art research on EHR systems, MedShare has several substantive innovations.

1) *Integration of a legacy EHR system into a private cloud environment:* MedShare integrates a legacy EHR system into the private cloud to mitigate the heterogeneity and complexity of interoperable systems.
2) *Interoperable private-cloud based autonomous legacy EHR systems:* Medshare helps extract, store and access EHRs in a private cloud without jeopardizing normal functioning of a legacy system. MedShare ensures a safe and reliable communication between the participating healthcare providers and offers a meaningful mechanism to develop trust between end-users.
3) *Segregation between owners and users of data using a hybrid cloud:* Principal control over shared EHRs is retained by their owners in their private clouds, however, these EHRs become locatable and accessible by other healthcare providers using the public cloud that integrates with the private clouds to index the physical location of each record in the system. Note that the public cloud stores de-identified patient HashMap.
4) *Two-way authorization and auditing in the hybrid cloud:* An EHR sharing request can only be generated by an internal doctor (or a nurse) who is first authorized by the patient. The authorization process relies on the both private and public clouds. Access to each patient record is separately maintained to be able to conduct a post-event analysis.

The remainder of the paper is organized as follows: Section II reviews the related literature. Section III presents data exchange scenarios from the hemodialysis centers in

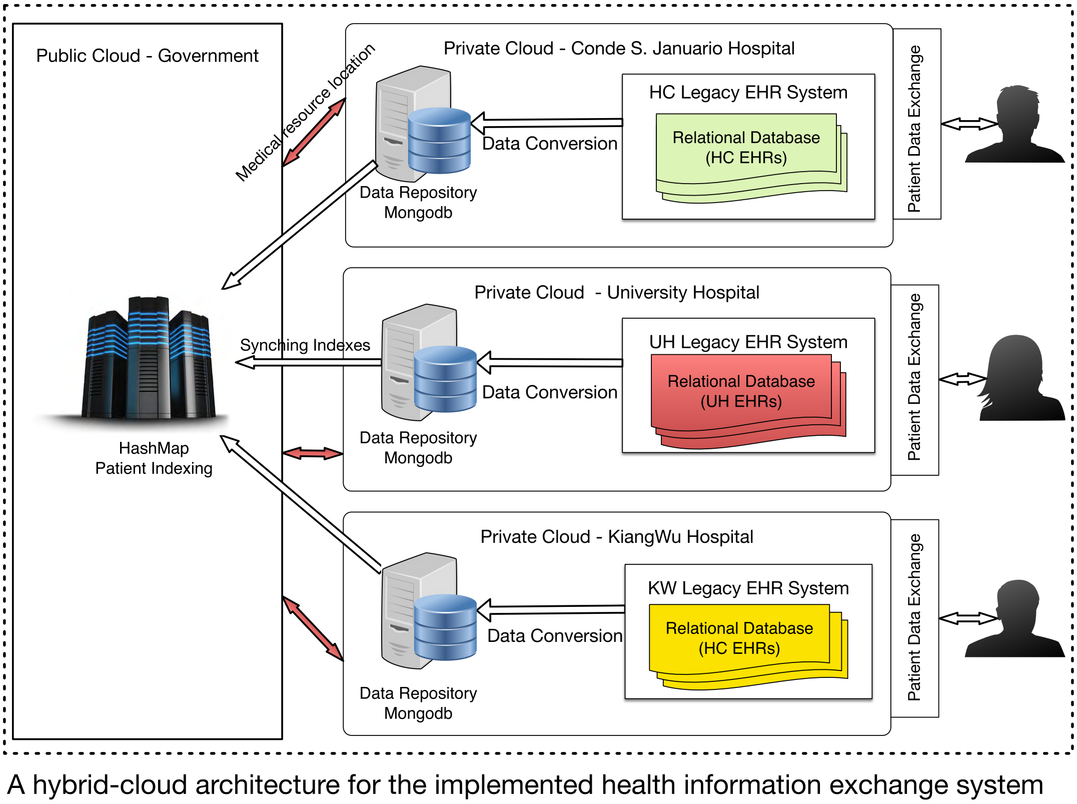

Public Cloud - Government
Medical resource location
Private Cloud - Conde S. Januario Hospital
HC Legacy EHR System
Relational Database (HC EHRs)
Data Conversion
Data Repository Mongodb
Patient Data Exchange
Private Cloud - University Hospital
UH Legacy EHR System
Relational Database (UH EHRs)
Synching Indexes
HashMap Patient Indexing
Private Cloud - KiangWu Hospital
KW Legacy EHR System
Relational Database (HC EHRs)
A hybrid-cloud architecture for the implemented health information exchange system

FIGURE 1. High level view of MedShare architecture.

Macau. Section IV introduces MedShare, a cloud-based system for medical data sharing. Section V presents a system prototype and its evaluation. Section VI concludes this paper and outlines the future work.

II. RELATED WORK

We review research on HIE systems associated with the cloud-based solutions that exchange patient information and use any of the followings in their system: a) cloud-based EHRs, b) legacy EHR systems, and c) technologies for privacy preservation in EHRs.

A. CLOUD-BASED EHR SHARING

Cloud computing provides users with flexible access and large storage capability and scalability that motivates hospitals to migrate EHR data from their own storage to the cloud [33]. The work [5] proposed the hybrid cloud-based framework for healthcare system with attribute-based encryption for data access. Moreover, the study [14] propose a framework for EHR data sharing which combines identity-based and attribute-based encryption to enforce access control. Another work [39] proposes a practical cloud solution for privacy preserving medical record sharing, which applied different level privacy concerns to different classifications of medical data. The study [6] proposed an EHR sharing and integration system in healthcare clouds and analyze the arising security and privacy issues in access and management of EHRs. In short, all the cloud related studies push the encrypted EHR data into the public cloud while preserving privacy by encryption techniques. Public cloud cannot 100% guarantee data security even if the data are encrypted, e.g., the leaks of celebrity photos from iCloud. Moreover, huge data synchronization cost cannot be ignore when pushing large medical data (e.g., CT image) to the cloud. Our proposed MedShare only pushes the de-identified indexing data to the public cloud, the privacy-sensitive data are still kept in the private cloud of hospitals. Furthermore, MedShare isolates the shared data from legacy EHR system to achieve high security and dependable demands in private clouds, and it contains a two-way authorization through private and public cloud with auditing.

B. LEGACY EHR SYSTEMS

Legacy EHR systems were not developed with a certain level of interoperability in mind, resulting in their inability to exchange medical resources. Yet, evidences [8] and [30] show that numerous benefits can be achieved by connecting legacy EHR systems, supporting an improved and integrated healthcare. But, large-scale adoption of such systems is impractical without addressing the privacy and security concerns [29]. On the other hand, larger hospital systems generally exchange electronic patient information internally, rather than with other external hospitals [22] to avoid losing patients. Under such circumstances, the adaptability of open standards for interoperable hospital systems is still far from practice. This situation calls health informatics researchers and users for a better interconnection among different hospitals. Another study shows that inter-organizational data exchange is one of the most important information system challenges [16], based on the user experiences with different regional health information exchange systems in Finland. Many efforts have been made in EHR sharing. One research problem is information interoperability. A recent work [32] combines metadata registries and semantic web technologies to uniquely reference, query and process a Common Data Element (CDE) to enable the syntactic and semantic interoperability. However, this research is limited to the interoperability of medical vocabulary.

C. PRIVACY PRESERVATION IN EHR

Privacy is another critical concern in medical resource sharing. The cross-domain authentication and fine-grained access control is studied in [34]. This study discusses an on-demand revocation if any of the two cooperating organizations are unwilling to share data anymore. Another approach [28] uses direct messaging, a secure e-mail-like protocol, to exchange encrypted health information online. The possibility of attacks on healthcare systems is discussed in [18]. Our work on security control will be deferred to the discussion of privacy and access control in detail.

One closely related work is eMOLST project [37] that handles data interoperability through: a) authenticating access to a shared medical resource by applying Single Sign-On (SSO) technique and b) a patient identity source system. It requires extra work to maintain a set of attributes associated with the patient. In contrast, our system computes the hash code of the patient identity number, uniquely representing each patient in the EHR sharing system. eMOLST requires a new system portal to access the EHRs, while our system is designed to work with EHR legacy system. Our patient indexing component that lets hospitals keep the data by themselves.

To address privacy and access control requirements in healthcare information systems, many studies have addressed different perspectives to secure medical data, e.g., [3], [15], [26]. A survey [31] across North America, Asia, and Europe shows that data sharing and data breaches are the biggest concerns for the users. NEHR [9] is a semi-distributed architecture that requires patient authentication to access a record. Other works [11] and [19] allow healthcare providers to access medical imaging data while ensuring protection of patient privacy using patient controlled access-key. The work [38] proposed a blockchain-based EHR sharing system, in which data access transactions are recorded in a tamper-proof blockchain for tracking and permission control. In contrast, our MedShare architecture provides a two-way authorization to provide better security to the healthcare network. Moreover, our locator service uses the de-identified HashMap to locate the resource, which reduces the risk of privacy breach. Moreover, the study [20] shows that de-identifying data provides no guarantee of anonymity. Thus, as shown in [38], our work not only audits the transactions but it provides a mechanism to perform a post-event analysis.

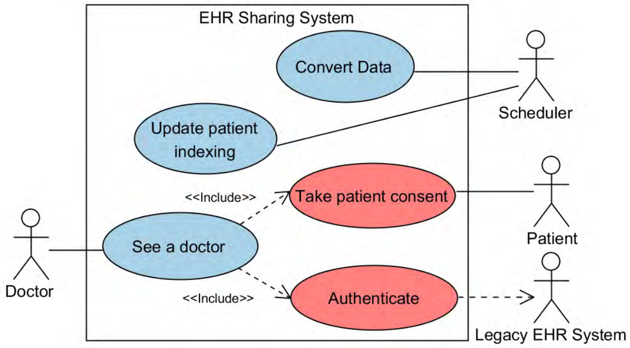


FIGURE 2. Use case diagram of the networked EHR system.

III. OVERVIEW OF HEMODIALYSIS CENTERS IN MACAU

The hemodialysis centers in Macau provide healthcare services to a large number of population. However, they are disconnected to share medical records of their patients. The patients visit a doctor in a hospital of their choice who is prescribed a hemodialysis treatment plan at specified date. If the patient suddenly decides to change their hemodialysis center, the exchange of patient information between hemodialysis centers becomes a bottleneck for the smooth delivery of medical services.

Macau citizens have confidence in public health systems that results in HC being the most visited hospital. Consequently, the initial diagnosis records and treatment plans are produced and stored in HC. Nonetheless, a patient may opt to go to another hospital, say UH, to take treatments due to unavailability of resources and their geographical location. The hemodialysis centers have no sharing platform in place. Therefore, it results in carrying paper-based by the patient medical data along with any other electronic data copies on CDs. It is noteworthy that patient privacy is well preserved with respect to the security of the EHR system in HC for a non-disclosure data agreement exists between KW and UH.

A. MEDSHARE REQUIREMENTS

The data-sharing problem leads to developing a hemodialysis network that should address the following functional and non-functional requirements. We use case diagram of Unified Modeling Language[1] (UML) to give a flavor of requirements of MedShare in Fig. 2.

Note that the legacy EHR systems (E.g, disconnected hemodialysis centers) are as actor in the use case diagram. The main functional requirements are listed below:

- The use case of seeing a doctor describes the procedure that a patient visits the doctor in a hospital, and the

[1] http://www.omg.org/spec/UML/

TABLE 1. An example of unified data format.

Attribute Name	HC Format	KW Format	UH Format	Unified Format
Patient identity	card_id	identitiy_id	id	patient_id
EHR identity	record_id	id_ehr	eid	ehr_id
Patient Name	p_name	patient_n	pname	patient_name
Name of the doctor	d_name	dotctor_n	dname	doctor_name

doctor requests for the related shared EHRs of the patient from other hospitals, if any.

- A doctor is authenticated and authorized to access a local medical record.
- A doctor may access medical records placed at another hospital through the same authentication service in their working hospital.
- The patient provides their consent, and authorizes the doctor to access their medical records. This guarantees that in a EHR sharing session, a patient authorization is recorded.
- The scheduler updates the local patient data in a unified format and updates it at the indexing server. These shared records should be regularly synchronized but not required to be updated in real-time. Note that a hemodialysis patient usually takes their next treatment after a specified time.

B. DATA FORMAT INCONSISTENCIES

Since the studied legacy EHR systems were autonomously designed and implemented, a number of database inconsistencies appeared at the time of implementation. The terminologies used to represent the EHR data were not based on any standard or common data format, which needed to be resolved first. TABLE 1 provides an example of the database entries from the three hospitals, though representing the same meanings, but with different names. The right most column presents the unified format agreed upon by the concerned authorities.

The unified EHR data format can significantly reduce the number of data inconsistencies between different EHR formats. Otherwise, each hospital requires a targeted data conversion for each corresponding hospital. In our unified EHR sharing scenario, each hospital only requires to conform to a single negotiated data format. However, EHR sharing (independent of unified format) requires bidirectional data conversion between two autonomous health care providers and the number of conversions can be calculated by the formula $n(n-1)$ if there are n hospitals. However, only n number of conversions are needed in a unified EHR sharing. Although, we currently have only three hospitals in the Macau EHR sharing case study, the network may grow well in the near future and other health providers and research institutes may take part in the data sharing process. In the future, the unified data format will ease the merger of a new healthcare provider into the MedShare.

Note that the unified data sharing format, data transformation and the negotiation process was directly held by the administration of the hospitals. For example, they can use a simple variable mapping like in Table 1, or adopt ontology-based model with their semantics mapping such as HL7 format [4], OpenEHR [10] standards, and other semantic models of EHRs [7], [23]. Our work was only confined to fill the technological gaps.

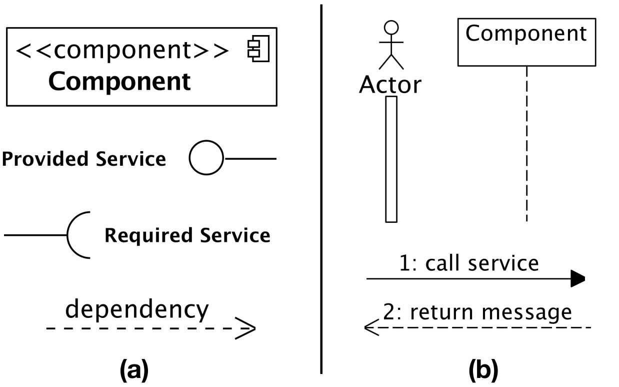


FIGURE 3. The part (a) of the Fig. 3 shows the self-explanatory UML component diagram and names its elements. Part (b) shows the elements of the component sequence diagram presenting the notion of actor and calling functions.

IV. MEDSHARE: INTEROPERABLE ARCHITECTURE

This section introduces the architectural aspects of the health information exchange system and elaborates on the technical details encountered in the development of the system. Our experience with developing a large system reveals that interoperability is not only the issue to enable two autonomous systems to exchange systems, but other non-technical factors also play a vital role. In this regard, one of the challenges lies in mediating the situation when autonomous health care providers are not interested to share the data of their patients cum customers and show a complete lack of interest in transferring the data to their competitors. After presenting the architectural details first, we will present a simple yet robust solution to this problem.

A. MEDSHARE ARCHITECTURE

We employ the component diagram based on UML notations of Fig. 3 (a) to present MedShare architecture. The architecture has two views: 1) External view: this represents the foundational block of resource sharing approach that allows for linking legacy EHR systems into a collaborative sharing of their data. 2) Internal view: this describes the design for the core components of the MedShare.

1) EXTERNAL VIEW

The Fig. 4 illustrates the external view of our system. Legacy EHR systems provide the services of *data conversion* to convert shared EHRs from a legacy system to a distributed EHR system. However, using the *authentication* service the doctor and the patient are authorized. By using both services from the local legacy systems, the unified EHR sharing system

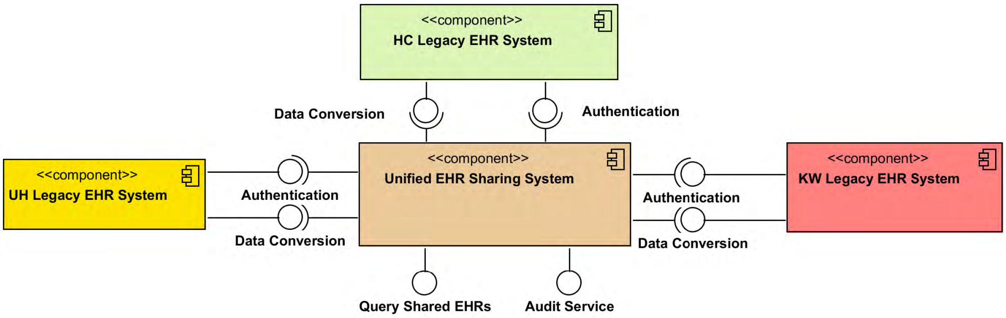


FIGURE 4. External view of distributed EHR sharing architecture.

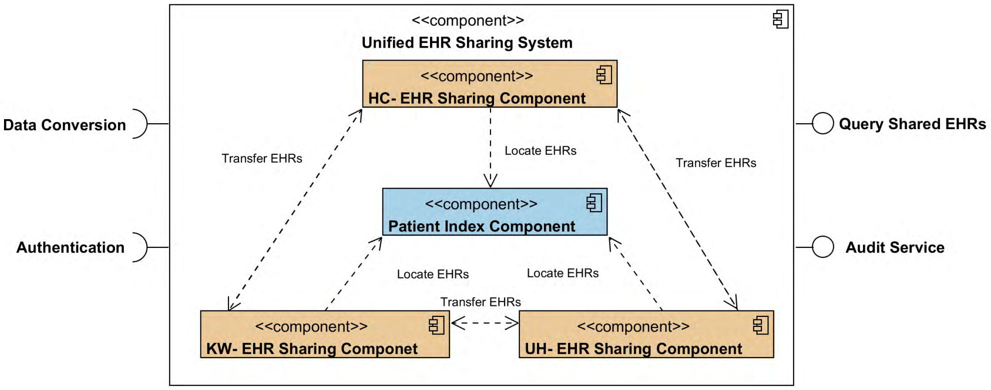


FIGURE 5. Internal view of distributed EHR sharing architecture.

provides two services: 1) It allows to run a query on the MedShare. 2) The audit service handles the privacy requirements of the system and post-breach data analysis, which is not detailed in this paper.

2) INTERNAL VIEW

The internal view of the unified EHR sharing system in Fig. 5 shows how the sub-systems collaborate to provide the required medical data querying mechanism from the different hospitals. The subsystems use the services provided by the index system in the data center to locate EHRs, then using the service of *transfer EHRs* in each subsystem to transfer all requested EHRs.

Note that only patient index component is deployed in public cloud, other components are deployed in private clouds of hospitals. Usually, the vendor lock-in problem may arise when the system is migrated in the future to a different cloud vendor that provides different technical stacks. To avoid vendor lock-in problem, a) we introduce MedShare in the component-based design, which is technically independent and the communications of the components are through standard RESTful web services. That can be easily implemented by the technical stacks such as NodeJS,.NET platform and Java EE. b) We adopt the container-based techniques such as Docker[2] to wrap the implemented MedShare components into Docker images, that can be seamless migrated and supported by most cloud providers without too much cost.

B. SCALABILITY AND RELIABILITY

MedShare architecture is scalable, that means it can involve more hospitals with their legacy EHR systems, which includes three steps: 1) A data extractor regularly extracts shared data from legacy systems and converts them into a unified data format. 2) HashMap are used to index the patient information in the public cloud, which is regularly synchronized with the EHR sharing component. 3) The authentications of the legacy EHR system are wrapped as RESTFul web service, which is connected to the unified EHR sharing system. MedShare is highly reliable, the deployed patient index component can be dynamically scaled to satisfy the performance requirements based on the nature of scalability and reliability of cloud techniques. The details of evaluation and demonstration are in sections V.

C. RESOURCE SHARING WORKFLOW

The high-level EHR sharing workflow involves many cooperating entities. The patient sees a doctor in an arbitrary hospital H_1 among HC, KW and UH. An EHR is generated and stored in the respective legacy EHR system of H_1. A scheduler regularly triggers to checks for the update on a particular EHR. This also synchronizes the shared EHRs located in the legacy EHR systems by updating the corresponding indexes

[2] http://www.docker.com/

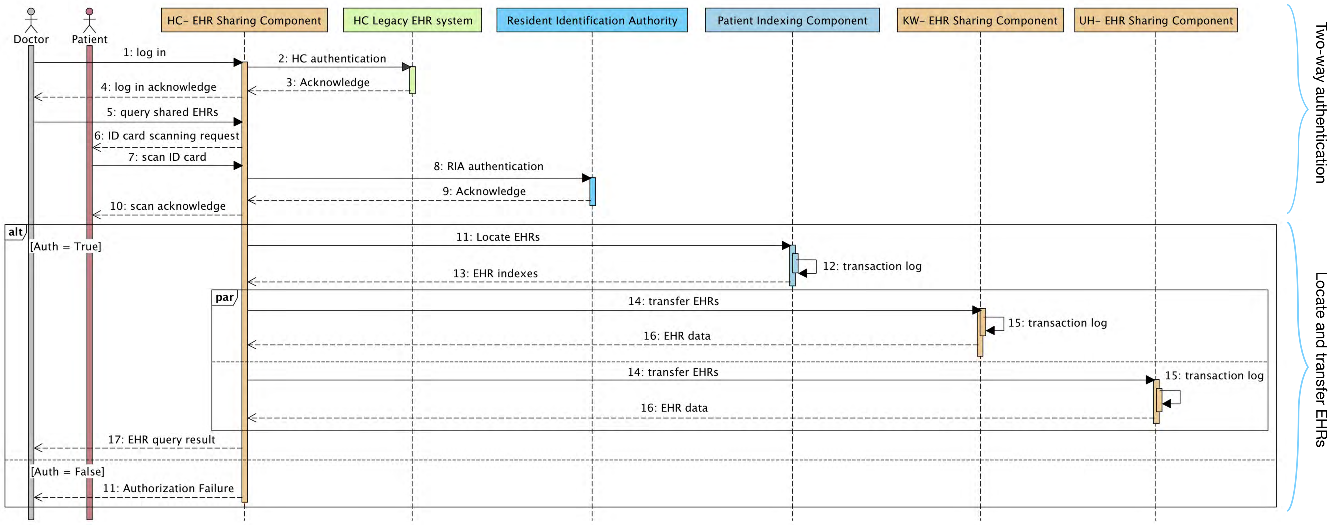


FIGURE 6. **Seeing a doctor by requesting all the hemodialysis records of a patient.**

on the patient indexing server. This allows a patient to see a doctor in another hospital H_2 who would now have access to the shared EHRs. At the time of requesting old EHRs of the patient, the doctor must be authorized by both the current hospital H_2 and the patient.

To understand the graphical notations used in the paper, non-familiar readers are referred to UML specification. However, for brevity, we provide the names and functions of the used notations in Fig. 3. Fig. 6 presents the detailed system usage scenario of the communication taking place between the actors and the EHR system. We use the component sequence diagram that allows to graphically show how the system components can interact with each other. Considering the proposed architecture, Fig. 6 contains two dependent processes: a) two-way authorization, and b) to locate EHR records and their transmission to another care provider.

1) TWO-WAY AUTHORIZATION

A doctor of HC is authorized through the authentication service in their working hospital (Please refer to *Steps 1, 2, 3, 4* in the diagram), and then the doctor runs the EHR queries on a patient data (*Step 5*). The patient then authorizes this request by scanning their ID card, the validity of ID card will be authenticated through the Resident Identification Authority (RIA) (*Step 6, 7, 8, 9, 10*).

2) LOCATE AND TRANSFER EHRs

The locate request can then be sent to the patient indexing center if the two-way authorization process successfully completes (*Step 11*). If the patient data is distributed over multiple locations (e.g., KW and UH), the query retrieves all the relevant index entries (*Step 13*). Finally, the data transmission request may be sent to one or more hospitals in the list (*Step 14*). Once the data transfer (*Step 16*) is completed in EHR sharing client of HC, the requested EHRs are displayed to the doctor (*Step 17*). The transactions are recorded in the log database for post-event analysis (*Step 12, 15*). This is important in case of a privacy and security breach. If the patient has EHRs in more than one hospital, the operations (*Step 14, 15, 16*) will be run in parallel for each of remote hospitals.

The systematic and precise use, as well as the understanding of UML notations for the specification of the workflow and resource sharing is based on our long term fundamental research on the rCOS[3] formal model-driven method of object and component-based systems. The semantics of the UML notations are formally defined in rCOS, including use case diagrams, class diagrams, sequence diagrams, interfaces and component diagrams. The formalization is essential for validation and verification of the system.

D. DATA MODEL AND PATIENT INDEXING

After discussed the use cases and workflow of distributed EHR sharing, the relevant entities and relations are described in Fig. 7. There are five entities: patient, EHR, hospital, patient index, and EHR index. Patient and EHR indexes are stored in the public cloud. Other entities are in the hospitals. One patient owns many EHRs, and EHRs are located in private cloud of the hospital. *Patient Index* contains the relation between patient and EHR, *EHR Index* contains the relations between EHRs and hospital. This domain model supports the functionalities of the distributed EHR sharing, which makes doctor find the desired EHRs of the patient through those indexes.

The patient indexing component stores all the references of the shared EHRs to facilitate data queries from the participating hospitals, i.e. requesters and providers. A requester poses a data location query to the patient index component

[3] https://en.wikipedia.org/wiki/RCOS_(computer_sciences)

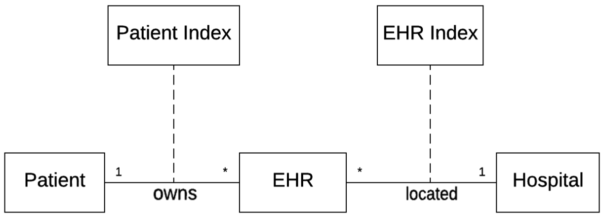


FIGURE 7. **MedShare data model.**

without a direct connection with a peer hospital. This is represented as ⇢ with a *locate EHRs* label and shows the dependency between components in Fig. 5. The label *transfer EHRs* provides access to the real data. The indexing component stores only the unique reference for each shared EHR, without any physical data relocation taking place from the original source. This approach offers two main advantages: 1) huge data synchronization burden is alleviated and 2) cybersecurity attacks and other threats from the internal users are minimized.

The *HashMap* technique is employed for patient indexing that includes a relationship between a patient and the EHR with the location. However, we leave it to the healthcare providers to decide about the segments of data to be indexed. Obviously, only the references are not enough. We need to store in the indexing server some attributes of a shared EHR that are not privacy-sensitive, as tags, along with the reference to the EHRs. The indexing server is then able to respond to queries based on these tags. Typically, the tags should include the source location, the encoded patient number, the date and time and the type of the EHRs. On the principle of facilitating queries while complying privacy policies, it also analyzes which set of tags is to be opened to the indexing server may be pre-negotiated between stakeholders.

There are two main reasons not to use central storage for patient data: 1) a hospital must push all the shared data into the data center before EHR sharing if the data center stores all data and 2) the local data should frequently be synchronized with the indexing server. That will lead to a huge synchronization burden to the data center because of enormous size of data. For example, imagine the CT scan examination report that may contain more than 1GB of data.

E. DATA QUERY STRUCTURE

As mentioned above, a data query includes two steps: 1) locating an EHR, and 2) the data transfer procedure. An EHR is located by a query, followed by the output. Hereunder, we illustrate this by using an example, which further will be detailed in the implementation section. Below, we provide the query attributes that includes patient identity, choosing the range of dates, EHR type and which hospitals to query.

Input Parameters for Locating:

Hashed Patient ID	Date Range	EHR Type	Hospitals

Output:

EHR ID	EHR Type	Date	Location

Note that the retrieved *ID* in above is used to access a particular EHR resource through *Transfer EHR service* as shown in the Fig. 5.

Input Parameters for Transferring:

EHR ID	EHR Type

The desired output is shown in a simplified way as follows. The output shown below is integrated into the graphical user interface of our toolset.

Output:

EHR Data

F. PROTECTING AGAINST CYBER ATTACKS

Patient data is highly sensitive which requires a carefully crafted security policy. Moreover, relocating data in the data exchange process involves a wide-range of data security threats, including the misuse of sensitive information. MedShare approach to data security begins with storing indices of all patient data into the trusted public cloud of a public healthcare provider. The actual data is stored in the private clouds of the hospitals. Our proposed approach includes a two-way authorization process to protect data from cyber-security attacks. EHR sharing request is only permitted and initiated by a doctor internally, and the request must be authorized by the patient and the data provider. The authentication process for doctors is implemented using the Role-based Access Control (RBAC) in the private cloud. The authorization mechanism is achieved by scanning patient's ID card, which are then authenticated by the public cloud of the Resident Identification Authority (RIA). Therefore, the authorization process combines both the private and public clouds. This two-way authentication is enforced to take patient consent and protect critical medical resources from outsiders. In a worst scenario, if the patient indexing server is compromised, the hashed patient identities are highly likely to remain protected and unidentifiable. Furthermore, secondary use of data by a doctor is not allowed as per the rules. Finally, all the operations in the resource sharing process are logged and stored to be able to investigate data breaches and perform audit services.

V. SYSTEM PROTOTYPING AND EVALUATION

We have presented a technique that allows to share and access patient data in a controlled environment. MedShare implements the following four layers to develop a working system.

A. MEDSHARE IMPLEMENTATION STACK

1) DATA INFRASTRUCTURE LAYER

The data infrastructure, as in Fig. 8, uses MongoDB [1] for data storage which is a NoSQL database. It is also a non-relational database. To deal complexity of medical records, it requires to have an adaptable data format to facilitate easy

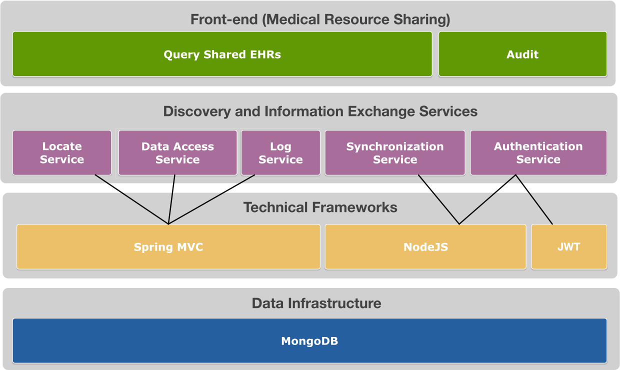


FIGURE 8. MedShare implementation stack.

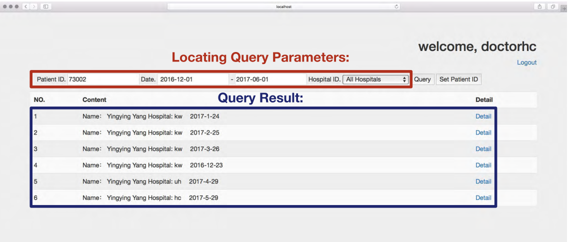
data transformations across multiple sources. This approach overcomes the bottlenecks of traditional databases. MongoDB also helps achieving mutability and scalability features for an EHR system.

2) TECHNICAL FRAMEWORK LAYER

All the components described in our architectural models are implemented by the lightweight Java EE framework Spring [13]. The required two-way authorization service in the legacy EHR system is implemented as a RESTFul web service by NodeJS [35] and JSON Web Token (JWT) [17]. A RESTful service can be defined as a means to hold query parameters. Contrary to JavaEE, NodeJS has the advantage of utilizing low resources to support high concurrency as well as scalable to industrial problems. JWT is a compact, URL-safe approach for representing claims between two communicating nodes. JWT provides the foundation of authentication service to RESTful web services. Thus, these two techniques guarantee the reliability and safety of the authentication process.

3) DISCOVERY AND INFORMATION EXCHANGE SERVICES LAYER

This layer has three Spring MVC services and two web services for authentication and synchronization. The *LocateService*, which is implemented using Spring MVC framework, identifies the required EHR location from the patients indexed in the MongoDB data infrastructure. The *LocateService* locates the EHRs based on the search conditions and transmits it to the doctor. The *DataAccess* is technically similar to *LocateService* but functions differently. It retrieves patient data from an identified source. The *AuthenticationService* provides the authorization service to the patient when a doctor requests for a specific EHR. The authentication also requires a service that integrates legacy EHR system into the authentication process. The *SynchronizationService* timely triggers the replication of the shared EHRs and updates the indexes in the patient indexing server. The *LogService* provides the log and tracking services to avoid data breach and trace irregularities. The authentication component is deployed in all the networked hospitals. The EHR query component is also deployed in the connected hospitals providing data transmission service, and also in the patient indexing server to support the location identification service. The *SynchronizationService* is deployed in the hospitals and data center to replicate shared EHRs and update indexes. The *LogService* is deployed on all servers because logs are generated and stored in the patient indexing server and all the other hospitals.



Patient ID. 73002 Date. 2016-12-01 - 2017-06-01 Hospital ID. All Hospitals Query Set Patient ID

NO.	Content	Detail
1	Name: Yingying Yang Hospital: kw 2017-1-24	Detail
2	Name: Yingying Yang Hospital: kw 2017-2-25	Detail
3	Name: Yingying Yang Hospital: kw 2017-3-26	Detail
4	Name: Yingying Yang Hospital: kw 2016-12-23	Detail
5	Name: Yingying Yang Hospital: uh 2017-4-29	Detail
6	Name: Yingying Yang Hospital: hc 2017-5-29	Detail

FIGURE 9. Locating a resource.

4) FRONT-END MEDICAL RESOURCE SHARING LAYER

This layer combines all the described layers. It directly uses the services available in the discovery and information exchange services layer. Using the front-end, an end-user can pose a query to the shared EHRs resources, and to retrieve a list of resources against a specific patient. The *Audit* service holds the system users accountable for their action in the system. By this, a doctor can distributively retrieve all the relevant records of a patient among all the participating hospitals while preserving patient privacy.

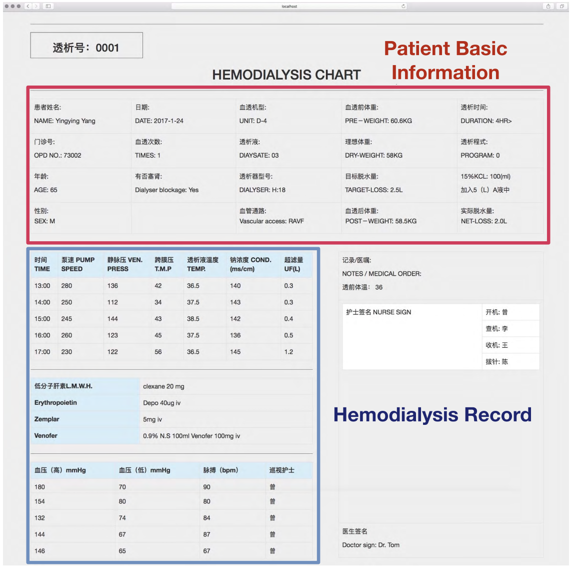
MedShare is freely available to download and reuse. The source code of the system is uploaded on GitHub.[4]

B. PROTOTYPE DEMONSTRATION

Let us assume that a doctor in HC hospital requests all the hemodialysis records of a patient named Yang Yingying. This scenario is depicted in Fig. 9 that provides a list of the hemodialysis records of Yang Yingying. The each record can individually be viewed by the doctor by clicking the *Details* link. For instance, the EHR corresponding to Sep 30, 2017, is shown in Fig. 10. The output includes two types of information: 1) the patient information, and 2) their hemodialysis records.

In order to monitor and track accessed data, MedShare allows the administrator to track the logs and investigate the specific operations performed by the users on a patient record. For example, when the auditor needs to trace the accessed EHR of a patient with ID 0221, the system can show the data accessed between two dates, as demonstrated in Figure 11. Our system demonstrates that it can support medical data interoperability between the hemodialysis centers in Macau without compromising the patient privacy. All the improperly accessed data will also be revealed to the administrator.

[4] https://github.com/yylonly/medshare



透析号：0001

HEMODIALYSIS CHART

患者姓名: NAME: Yingying Yang	日期: DATE: 2017-1-24	血透机型: UNIT: D-4	血透前体重: PRE－WEIGHT: 60.6KG	透析时间: DURATION: 4HR>
门诊号: OPD NO.: 73002	血透次数: TIMES: 1	透析液: DIAYSATE: 03	理想体重: DRY-WEIGHT: 58KG	透析程式: PROGRAM: 0
年龄: AGE: 65	有否塞肾: Dialyser blockage: Yes	透析器型号: DIALYSER: H:18	目标脱水量: TARGET-LOSS: 2.5L	15%KCL: 100(ml) 加入5（L）A液中
性别: SEX: M		血管通路: Vascular access: RAVF	血透后体重: POST－WEIGHT: 58.5KG	实际脱水量: NET-LOSS: 2.0L

时间 TIME	泵速 PUMP SPEED	静脉压 VEN. PRESS	跨膜压 T.M.P	透析液温度 TEMP.	钠浓度 COND. (ms/cm)	超滤量 UF(L)
13:00	280	136	42	36.5	140	0.3
14:00	250	112	34	37.5	143	0.3
15:00	245	144	43	38.5	142	0.4
16:00	260	123	45	37.5	136	0.5
17:00	230	122	56	36.5	145	1.2

低分子肝素L.M.W.H.	clexane 20 mg
Erythropoietin	Depo 40ug iv
Zemplar	5mg iv
Venofer	0.9% N.S 100ml Venofer 100mg iv

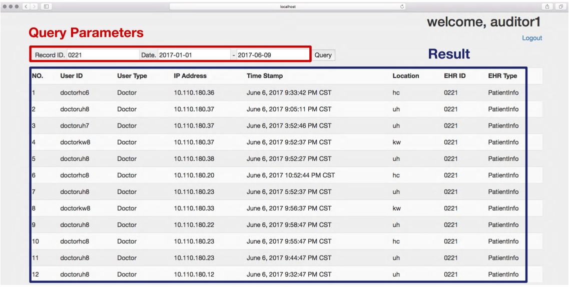
血压（高）mmHg	血压（低）mmHg	脉搏（bpm）	巡视护士
180	70	90	曾
154	80	80	曾
132	74	84	曾
144	67	87	曾
146	65	67	曾

记录/医嘱:
NOTES / MEDICAL ORDER:
透前体温：36

护士签名 NURSE SIGN	开机: 曾 查机: 李 收机: 王 拔针: 陈

医生签名
Doctor sign: Dr. Tom

FIGURE 10. A detailed hemodialysis report.



welcome, auditor1

Record ID. 0221 Date. 2017-01-01 - 2017-06-09

NO.	User ID	User Type	IP Address	Time Stamp	Location	EHR ID	EHR Type
1	doctorhc6	Doctor	10.110.180.36	June 6, 2017 9:33:42 PM CST	hc	0221	PatientInfo
2	doctoruh8	Doctor	10.110.180.37	June 6, 2017 9:05:11 PM CST	uh	0221	PatientInfo
3	doctoruh7	Doctor	10.110.180.37	June 6, 2017 3:52:46 PM CST	uh	0221	PatientInfo
4	doctorkw8	Doctor	10.110.180.37	June 6, 2017 9:52:37 PM CST	kw	0221	PatientInfo
5	doctoruh8	Doctor	10.110.180.38	June 6, 2017 9:52:27 PM CST	uh	0221	PatientInfo
6	doctorhc8	Doctor	10.110.180.20	June 6, 2017 10:52:44 PM CST	hc	0221	PatientInfo
7	doctoruh8	Doctor	10.110.180.23	June 6, 2017 5:52:37 PM CST	uh	0221	PatientInfo
8	doctorkw8	Doctor	10.110.180.33	June 6, 2017 9:56:37 PM CST	kw	0221	PatientInfo
9	doctoruh8	Doctor	10.110.180.22	June 6, 2017 9:58:47 PM CST	uh	0221	PatientInfo
10	doctorhc8	Doctor	10.110.180.23	June 6, 2017 9:55:47 PM CST	hc	0221	PatientInfo
11	doctoruh8	Doctor	10.110.180.23	June 6, 2017 9:44:47 PM CST	uh	0221	PatientInfo
12	doctoruh8	Doctor	10.110.180.12	June 6, 2017 9:32:47 PM CST	uh	0221	PatientInfo

FIGURE 11. Auditing access to a medical resource.

C. EVALUATION RESULTS

Our prototype implementation of MedShare is deployed in three private clouds named HC, KW, UH and one public cloud called PI. Private clouds are hosted on three *Dell T630* servers with Docker installed. The Dell server configurations are *Intel(R) Xeon(R) CPU E5-2603*, *16GB DDR4 memory*, *1TB 7200 RPM Hard Disk*, and *1000MB Ethernet connection*. Public cloud is hosted on Google Cloud with standard machine type *n1-standard-1*. Apache JMeter[5] is used for load-testing of its functional behavior and measuring the performance, which is installed on *iMac* with *Intel Core i7*, *16GB DDR3 memory*, *512 SSD*, *5G WiFi* connections to the router. Router has *100MB bandwidth* to ISP, and *1000MB bandwidth* to *Dell* server. The EHR data are retrieved from hemodialysis center of Kiang Wu hospital, and generated testing data for other two hospitals.

To test scalability and reliability of MedShare platform, it underwent three types of load-testing. In our testing scenario, we had 100,000 hemodialysis records with 4000 concurrency clients by default.

TABLE 2. The result on different concurrency clients.

Concurrency Clients	Number of Queries	Response Time (ms)	Timeout (%)	Throughput (#/s)	Bandwidth (KB/s)
1,000	18,984	630	0.00	1,580.01	4213.9
2,000	21,325	1,129	0.01	1,750.55	4671.8
3,000	27,778	1,739	0.05	1,700.46	4528.1
4,000	35,468	2,362	0.14	1,660.95	4401.1
5,000	45,263	3,051	0.42	2,120.44	5651.8
Average	29.764	1.782	0.12	1,782.20	4693.34

TABLE 3. Performance analysis against access modes.

Access Mode	Number of Queries	Response Time (ms)	Timeout (%)	Throughput(#/s)	Bandwidth(KB/s)
Open Homepage	4207	2162	0.01	347.32	698.81
Authorization	4111	55	0.12	343.05	225.22
Login	4111	1149	0.03	339.97	651.23
Locate Records	4075	3894	0.15	332.50	796.81
Query Records	3964	4648	0.67	328.02	2081.92

1) CONCURRENCY TESTING

We conducted load testing with different number of concurrency clients within 120 seconds. The performance is shown in Table 2. *Concurrency Clients* represent the concurrent number of clients in MedShare. *Total Queries* is the total number of data requests posed to MedShare. *Response Time* is the average response time during the load testing. *Timeout* represents the percentage of query requested that MedShare refuses to response beyond 10 ms. *Throughput* is the number of requests that are processed per second. *Bandwidth* is the maximum rate of data transfer measured in kilobytes per second. For clients from 1000 to 3000, MedShare has less than 0.1% query timeout. When the number of clients was increased from 4000 to 5000, MedShare has 0.14% to 0.42% request timeout. In average, *JMeter* sends 29,764 queries messages to MedShare, the average response time is 1,782 ms, the timeout is 0.12%, throughput is 1,782.20 queries per seconds, and bandwidth is 4693.34 kilobytes per second. The results indicate that MedShare can keep low timeout rate and response time even with a large number of concurrent clients. It thus demonstrates that MedShare can reliably be deployed and used to assist with sharing EHRs among autonomous but connected healthcare provider.

We also conducted load tests on each of the access modes to understand bottlenecks of the system. The result is shown in Table 3 that lists the access modes such as accessing the home-page of the system, authorization, login, locating records, and fetching particular a medical record. We observe that the *query records* step has the highest response time and timeout rate because it requires exchange of medical records across the integrated private clouds of the hospitals. The response time and timeout rate keep low in the other steps. Therefore, the efficiency and reliability highly depends on the privates clouds of the hospitals.

2) SCALABILITY OF COMPUTING RESOURCES

MedShare components are implemented as Docker containers that is highly scalable by the auto-scaling mechanism (e.g., to scale CPU resources) of Docker in the cloud. It reduces the response time and increases throughput. To demonstrate auto-scaling mechanism, we scale each component of MedShare up to three replicas with three CPU units

[5] https://jmeter.apache.org

TABLE 4. Analyzing the performance relative to CPUs and Replicas.

CPU	Replica	Response Time (ms)	Timeout (%)	Throughput (#/s)
1	1	3,051	0.42	2,120.44
2	2	1,529	0.13	2,750.55
3	3	831	0.01	3,120.44

TABLE 5. Size of datasets.

Number of Records	Response Time (ms)	Timeout (%)	Throughput (#/s)
100	1,929	0.06	1,960.32
1,000	2,129	0.10	1,871.42
10,000	2,207	0.12	1,861.43
100,000	2,362	0.14	1,660.95
Average	2,157	0.105	1,838.53

in the hybrid cloud. The result of 5000 concurrent clients is shown in Table 4. When scaling CPU and replica from one to three, the average response time is largely reduced from 3,051 ms to 831 ms, and the timeout rate is decreased from 0.42% to 0.01%. It demonstrates that MedShare can be scaled up to a significant number to support scalability.

3) DATASET SCALABILITY

Varying amounts of datasets were also tested, as shown in Table 5. Records ranged from 100 to 100,000 per dataset. The average noted response time was 2,362 ms, 0.105% timeout, and 1,838.53 throughput. The results show that the response time and timeout were stable for varying sizes of datasets. That also demonstrates that MedShare is highly robust with low variations in the performances. Thus, load testing shows that MedShare is highly scalable and reliable for EHR sharing in a hybrid cloud environment.

D. DISCUSSION AND LIMITATIONS

The MedShare platform is a scalable and reliable solution as a hybrid cloud for medical resource sharing among autonomous healthcare providers without compromising patient privacy. The main components include a two-way authorization process, integration of the legacy systems with independent EHRs storage, and locating a requested EHR through privacy-preserving mechanism such as hashed indexes. However, it has some limitations: 1) its reliability highly depends on the public cloud as EHRs can only be located through the public cloud. However, it can be alleviated by the nature of scalability and reliability provided by a cloud. 2) The extra costs are needed to implement data transformers from a specific EHR format of a hospital to the unified data format. The authentication service needs to be wrapped as a RESTful web service to communicate with MedShare. For this, we envision to extend our work providing automated tools to help with developing transformers and wrappers in the near future.

VI. CONCLUSION AND FUTURE PERSPECTIVES

We presented a generic solution for improving communication between autonomous healthcare providers by implementing a novel hybrid cloud architecture appropriately access and securely share a patient's medical information electronically. The legal and functional constraints in our implemented system play a supervisory role. We successfully negotiated a common data model to facilitate the automatization process while mitigating technical challenges in the form of conventional tools and technologies. Applying a standardized data format, such as HL7, also becomes a daunting task because of bilingual patient data storage, for example, both English and Chinese languages were being used by the healthcare professionals. MedShare ensured that participating healthcare providers have confidence in the developed system by owning and controlling their own data.

Our experience suggests that a gradual integration of a legacy EHR system into a cloud environment is essential. Healthcare provides are greatly concerned with patient privacy, which is now considered a foundation for modern healthcare systems. MedShare preserves patient privacy by a two-way authorization process that collects patient consent before making the data available through the public and private clouds. To integrate patient consent into a data-sharing scenario, our system takes the advantage of national identification cards to be swiped by patients to record their authorization consent. All patients in a hospital are uniquely identified by their identities that are hashed in the data indexing process. The patient indexing technique enables a secure data exchange environment without moving data to the public cloud. Thus, it significantly helps develop a sense of cooperation and collaboration among the connected healthcare operators.

Our future work includes developing an intense auditing process over shared personal medical data. To this end, we also aim to study potential attacks on the deployed system. In data sharing scenarios where multiple languages are used to store, process and communicate data, choosing a single language becomes a bottleneck making it complex to apply a unified data format. We intend to investigate these issues further, also how to extend our system to support multiple languages in it. This would require some additional work in natural language processing, requiring to resolve syntactic and semantic conflicts. We also aim to increase the number of hospitals in our interoperable resource sharing network. We also plan to report our findings on the scalability and openness of the system. Robust evaluation studies are needed to evaluate non-functional aspects of the system, including heterogeneity, resource management, transparency, and openness and performance analysis.

REFERENCES

[1] V. Abramova and J. Bernardino, ''NoSQL databases: MongoDB vs cassandra,'' in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, 2013, pp. 14–22.

[2] K. T. Adams *et al.*, ''An analysis of patient safety incident reports associated with electronic health record interoperability,'' *Appl. Clin. Inform.*, vol. 8, no. 2, pp. 593–602, 2017.

[3] O. Angiuli, J. Blitzstein, and J. Waldo, ''How to de-identify your data,'' *Commun. ACM*, vol. 58, no. 12, pp. 48–55, 2015.

[4] S. Bakken, K. E. Campbell, J. J. Cimino, S. M. Huff, and W. E. Hammond, ''Toward vocabulary domain specifications for health level 7–coded data elements,'' *J. Amer. Med. Inform. Assoc.*, vol. 7, no. 4, pp. 333–342, 2000.

[5] B. Balamurugan, P. V. Krishna, N. S. Kumar, and G. V. Rajyalakshmi, ''An efficient framework for health system based on hybrid cloud with ABE-outsourced decryption,'' in *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems*, L. P. Suresh, S. S. Dash, and B. K. Panigrahi, Eds. New Delhi, India: Springer, 2015, pp. 41–49.

[6] Y. Y. Chen, J. C. Lu, and J. K. Jan, ''A secure EHR system based on hybrid clouds,'' *J. Med. Syst.*, vol. 36, no. 5, pp. 3375–3384, 2012.

[7] M. del C. Legaz-García, C. Martínez-Costa, M. Menárguez-Tortosa, and J. T. Fernández-Breis, ''A semantic web based framework for the interoperability and exploitation of clinical models and EHR data,'' *Knowl.-Based Syst.*, vol. 105, pp. 175–189, Aug. 2016.

[8] I. Featherstone and J. Keen, ''Do integrated record systems lead to integrated services? An observational study of a multi-professional system in a diabetes service,'' *Int. J. Med. Inform.*, vol. 81, no. 1, pp. 45–52, Jan. 2012.

[9] L. L. Fragidis, P. D. Chatzoglou, and V. P. Aggelidis, ''Integrated nationwide electronic health records system: Semi-distributed architecture approach,'' *Technol. Health Care*, vol. 24, no. 6, pp. 827–842, 2016.

[10] S. Garde, P. Knaup, E. J. S. Hovenga, and S. Heard, ''Towards semantic interoperability for electronic health records: Domain knowledge governance for *openEHR* archetypes,'' *Methods Inf. Med.*, vol. 46, no. 3, pp. 332–343, 2007.

[11] Y. Ge, D. K. Ahn, B. Unde, H. D. Gage, and J. J. Carr, ''Patient-controlled sharing of medical imaging data across unaffiliated healthcare organizations,'' *J. Amer. Med. Inform. Assoc.*, vol. 20, no. 1, pp. 157–163, 2013.

[12] R. A. Greenes, *Health Information Systems 2025*. Cham, Switzerland: Springer, 2016, pp. 579–600.

[13] P. Gupta and M. C. Govil, ''MVC design pattern for the multi framework distributed applications using XML, Spring and Struts framework,'' *Int. J. Comput. Sci. Eng.*, vol. 2, no. 4, pp. 1047–1051, 2010.

[14] J. Huang, M. Sharaf, and C.-T. Huang, ''A hierarchical framework for secure and scalable EHR sharing and access control in multi-cloud,'' in *Proc. 41st Int. Conf. Parallel Process. Workshops*, Sep. 2012, pp. 279–287.

[15] M. Z. Hydari, R. Telang, and W. M. Marella, ''Electronic health records and patient safety,'' *Commun. ACM*, vol. 58, no. 11, pp. 30–32, 2015.

[16] H. Hyppönen, J. Reponen, T. Lääveri, and J. Kaipio, ''User experiences with different regional health information exchange systems in Finland,'' *Int. J. Med. Inform.*, vol. 83, no. 1, pp. 1–18, Jan. 2014.

[17] M. Jones and B. Campbell, and C. Mortimore, *JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants*, document RFC 7523, Internet-Draft draft-ietf-oauth-jwt-bearer-10, 2014.

[18] J. Kwon and M. E. Johnson, ''Protecting patient data—The economic perspective of healthcare security,'' *IEEE Security Privacy*, vol. 13, no. 5, pp. 90–95, Sep./Oct. 2015.

[19] S. G. Langer *et al.*, ''The RSNA image sharing network,'' *J. Digit. Imag.*, vol. 28, no. 1, pp. 53–61, Feb. 2015.

[20] Z. Liu, N. Qamar, and J. Qian, ''A quantitative analysis of the performance and scalability of de-identification tools for medical data,'' in *Foundations of Health Information Engineering and Systems*. Springer, 2013, pp. 274–289.

[21] S. T. Mennemeyer, N. Menachemi, S. Rahurkar, and E. W. Ford, ''Impact of the HITECH Act on physicians' adoption of electronic health records,'' *J. Amer. Med. Inform. Assoc.*, vol. 23, no. 2, pp. 375–379, 2016.

[22] A. R. Miller and C. Tucker, ''Health information exchange, system size and information silos,'' *J. Health Econ.*, vol. 33, pp. 28–42, Jan. 2014.

[23] A. Moreno-Conde *et al.*, ''Clinical information modeling processes for semantic interoperability of electronic health records: Systematic review and inductive analysis,'' *J. Amer. Med. Inform. Assoc.*, vol. 22, no. 4, pp. 925–934, 2015.

[24] A. Murua, E. Carrasco, A. Agirre, J. M. Susperregi, and J. Gómez, *Upgrading Legacy EHR Systems to Smart EHR Systems*. Cham, Switzerland: Springer, 2018, pp. 227–233.

[25] L. Nguyen, E. Bellucci, and L. T. Nguyen, ''Electronic health records implementation: An evaluation of information system impact and contingency factors,'' *Int. J. Med. Inform.*, vol. 83, no. 11, pp. 779–796, 2014.

[26] N. Qamar, J. Faber, Y. Ledru, and Z. Liu, ''Automated reviewing of healthcare security policies,'' in *Foundations of Health Information Engineering and Systems*. Berlin, Germany: Springer, 2013, pp. 176–193.

[27] N. Qamar, Y. Yang, A. Nadas, and Z. Liu, ''Querying medical datasets while preserving privacy,'' *Procedia Comput. Sci.*, vol. 98, pp. 324–331, Sep. 2016.

[28] J. J. Reicher and M. A. Reicher, ''Implementation of certified EHR, patient portal, and 'direct' messaging technology in a radiology environment enhances communication of radiology results to both referring physicians and patients,'' *J. Digit. Imag.*, vol. 29, no. 3, pp. 337–340, Jun. 2016.

[29] F. Rezaeibagha and Y. Mi, ''Distributed clinical data sharing via dynamic access-control policy transformation,'' *Int. J. Med. Inform.*, vol. 89, pp. 25–31, May 2016.

[30] C. Rinner *et al.*, ''Improving the informational continuity of care in diabetes mellitus treatment with a nationwide shared EHR system: Estimates from Austrian claims data,'' *Int. J. Med. Inform.*, vol. 92, pp. 44–53, Aug. 2016.

[31] S. Sheth, G. Kaiser, and W. Maalej, ''Us and them: A study of privacy requirements across North America, Asia, and Europe,'' in *Proc. 36th Int. Conf. Softw. Eng.*, 2014, pp. 859–870.

[32] A. A. Sinaci and G. B. L. Erturkmen, ''A federated semantic metadata registry framework for enabling interoperability across clinical research and care domains,'' *J. Biomed. Inform.*, vol. 46, no. 5, pp. 784–794, Oct. 2013.

[33] D. Sobhy, Y. El-Sonbaty, and M. A. Elnasr, ''MedCloud: Healthcare cloud computing system,'' in *Proc. Int. Conf. Internet Technol. Secured Trans.*, Dec. 2012, pp. 161–166.

[34] J. Sun and Y. Fang, ''Cross-domain data sharing in distributed electronic health record systems,'' *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 6, pp. 754–764, Jun. 2010.

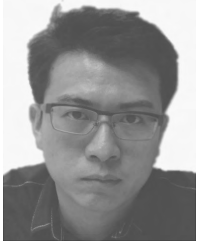
[35] S. Tilkov and S. Vinoski, ''Node.js: Using JavaScript to build high-performance network programs,'' *IEEE Internet Comput.*, vol. 14, no. 6, pp. 80–83, Nov./Dec. 2010.

[36] A. H. Turan and P. C. Palvia, ''Critical information technology issues in Turkish healthcare,'' *Inf. Manage.*, vol. 51, no. 1, pp. 57–68, Jan. 2014.

[37] G. von Laszewski, J. Dayal, and L. Wang, ''eMOLST: A documentation flow for distributed health informatics,'' *Concurrency Comput., Pract. Exp.*, vol. 23, no. 16, pp. 1857–1867, 2011.

[38] Q. I. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani, ''MeDShare: Trust-less medical data sharing among cloud service providers via blockchain,'' *IEEE Access*, vol. 5, pp. 14757–14767, 2017.

[39] J.-J. Yang, J.-Q. Li, and Y. Niu, ''A hybrid solution for privacy preserving medical data sharing in the cloud environment,'' *Future Gener. Comput. Syst.*, vols. 43–44, pp. 74–86, Feb. 2015.



YILONG YANG received the B.S. degree in computer science from the China University of Mining and Technology, China, in 2010, and the M.S. degree from Guizhou University, China, in 2013. He is currently pursuing the Ph.D. degree in software engineering with the University of Macau. He has been a fellow with United Nations University–International Institute for Software Technology, Macau. His research interests include automated software engineering and machine learning.

XIAOSHAN LI received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, in 1994. He is currently an Associate Professor with the Department of Computer and Information Science, University of Macau. His research interests are health information exchange, formal methods, object-oriented software engineering, real-time specification and verification, and semantics of programming language.

NAFEES QAMAR received the Ph.D. degree in computer science from Joseph Fourier University, France. He is currently the Health Informatics Program Director and an Assistant Professor with the Department of Health Administration, Governors State University, IL, USA. His research interests include health information exchange, medical data interoperability and security, healthcare software design and implementation, patient privacy preservation techniques, medical data de-identification, and formal verification and validation techniques.

PENG LIU received the B.S. degree in computer science and technology from the Shandong University of Science and Technology, China, in 2011, and the M.S. degree in computer architecture from North China Electric Power University in 2014. He is currently pursuing the Ph.D. degree with the University of Macau. His research interests include machine learning, image processing, and data mining.

WEI KE received the Ph.D. degree in computer applied technology from Beihang University, with a focus on programming languages, formal methods, and software engineering tools. He had successfully completed a couple of research projects funded by Macau FDCT in formal methods and software engineering.

BINGQING SHEN received the B.Eng. degree in communication engineering from Shanghai University, China, in 2006, and the M.Sc. degree in computer control and automation from Nanyang Technological University, Singapore, in 2008. He is currently pursuing the Ph.D. degree with the University of Macau. His research interests include virtual world, virtual world platform, and virtual wealth.

ZHIMING LIU received the Ph.D. degree in computer science from the University of Warwick, U.K. He was a Research Fellow and then a Senior Researcher with United National University–International Institute for Software Technology from 2002 to 2013 and the Chair Professor of software engineering at Birmingham City University from 2013 to 2015. He currently leads RISE, the Centre for Research and Innovation in Software Engineering, Southwest University, under the Innovative Talents Recruitment Program (1000-Expect Program). His main research interest is in the areas of formal methods, including real-time systems, fault-tolerant systems, health information systems, and object-oriented and component-based systems.