applied sciences

MDPI

Article

Privacy-Aware Cloud Architecture for Collaborative Use of Patients' Health Information

Fadi Alhaddadin [1,*] and Jairo Gutierrez [2,*]

[1] CUC Research and Innovation Center, CUC-Ulster University, 12th Floor, Excellence Tower, West Bay, Doha P.O. Box 200197, Qatar
[2] Department of Computer Science and Software Engineering, Auckland University of Technology WZ Building, 6 St. Paul Street, Auckland 1010, New Zealand
* Correspondence: fadi.alhaddadin@cuc-ulster.edu.qa (F.A.); jairo.gutierrez@aut.ac.nz (J.G.); Tel.: +974-40-198-189 (F.A.)

Abstract: Cloud computing appears to be the dreamed-of vision of the healthcare industry; it refers to means of storing and accessing data and programs over the Internet instead of the computer's hard drive. However, the adoption of cloud computing requires solving several issues, and information privacy is a major one. This work proposes a cloud architecture design for the healthcare information system. The proposed architecture enables storing and sharing information in a privacy-preserving manner. Patients' information in the proposed architecture is divided into four categories identified in the case study data analysis. User identity management protocol (U-IDM) is employed for controlling access to patients' information, and patients have means of control over who can access their information. A scenario-based instantiation validated the proposed architecture's privacy-preserving patient data exchange. The instantiation proved that the proposed architecture allows sharing healthcare information without violating the privacy of patients.

Keywords: cloud architecture; data privacy; data confidentiality; information sharing; health information; patient records

Citation: Alhaddadin, F.; Gutierrez, J. Privacy-Aware Cloud Architecture for Collaborative Use of Patients' Health Information. *Appl. Sci.* **2023**, *13*, 7401. https://doi.org/10.3390/app13137401

Academic Editor: Gianluca Lax

Received: 28 April 2023
Revised: 25 May 2023
Accepted: 26 May 2023
Published: 22 June 2023

1. Introduction

Recently, the healthcare sector has shown a growing interest in information technologies to facilitate new methods of collecting, managing, and analyzing health-related information. In fact, the healthcare sector is under pressure to embrace many new technologies that are available on the market, such as the Internet of Things (IoT) [1], and as consequence to such embracement, the amount of healthcare information is rapidly growing in detail and diversity, driven by record keeping, compliance, regulatory requirements, and of course, patient care [2]. Such information generates special value when it is exchanged and collaboratively used among different parties involved in the healthcare area [3]. Several researchers consider immediate access to previously generated medical records during healthcare service delivery highly important, it leads to effective ways of preventing and managing illnesses, as well as the discovery of new drugs and therapies [4].

Patients choose medical providers such as hospitals and pharmacies based on proximity, quality, cultural fit, and bedside manner. This fragments patient data in heterogeneous systems. Most of this data are housed in private, heterogeneous, dispersed health information systems [5]. Thus, health-related data in these systems cannot be easily accessible to provide a complete patient profile. For instance, when a patient visits a general practitioner, they often need additional medical services over time, such as magnetic resonance imaging scans or cholesterol and blood sugar tests.

Sharing healthcare information improves patient care by better understanding health requirements [6]. For that, the seamless exchange of multimedia clinical information is considered a fundamental requirement. Different technological approaches can be adopted

for enabling the communication and sharing of health record segments [7]. However, interoperability and data privacy are important obstacles to maximizing healthcare information sharing. Interoperability means two or more systems can exchange and use information [8]. It is the ability to share and use information across multiple system technologies seamlessly. It is a fundamental requirement for the health care system to derive the societal benefits promised by the adoption of electronic healthcare records [7].

1.1. Information Privacy

Information privacy is the desire of individuals to control or have some influence over data about themselves [9]. In other words, it is the right of individuals to determine how and to what extent the information they communicate to others is used. Healthcare data include sensitive records that should not be made available to unauthorized people to protect the privacy of patients. Information privacy protection is very essential to building users' trust in order to reach the full potential of information sharing in the healthcare domain [10,11]. Therefore, it is a mandatory step to adhere to legal frameworks such as the Health Insurance Portability and Accountability Act (HIPPA) [12] and the Data Protection Act [13]. Such frameworks clearly specify the responsibilities of organizations with regards to the privacy protection of personal health information. However, complying with these frameworks is both challenging and costly for healthcare organizations [14]. The main privacy challenge remains in the management of this collected data, which is still largely unaddressed. There are many policy-related issues, such as privacy policies, that must be addressed to realize the full potential of sharing healthcare information [15,16]. Sharing healthcare information using healthcare information systems based on privacy preservation rarely handles healthcare information sharing among healthcare-related entities at different places [17]; therefore, there is a need to address such collaboration based on privacy preservation.

1.2. Research Problem

Due to the diversity and complexity of the existing healthcare structure, in which patients' health information is distributed to multiple entities such as hospitals, healthcare centers, and cloud servers, an appropriate architecture is one of the most important design issues for sharing healthcare information in a privacy-preserving manner. A centralized architecture design would not be convenient due to the lack of interoperability of most healthcare information systems. Currently, there are no policies for healthcare data standardization and normalization for proper data governance; it is also determined that there is no existing single data standardization structure that can effectively share and interpret patient data within heterogeneous systems [18,19].

Despite the use of information technology solutions in the healthcare industry, there are various challenges encountered, such as high infrastructure management costs, dynamic needs for computational resources, scalability, multi-tenancy, and increased demand for collaboration [20]. The advancement of the healthcare industry requires modernizing healthcare information systems to facilitate collaboration and coordination among parties involved in the healthcare domain at lower costs. In healthcare, the availability of information, regardless of the location of the patient and the clinician, is a key driver toward patient satisfaction and healthcare service improvement. For that, there is a pressing need for a decentralized design of the architecture for healthcare information systems that allows for asynchronous interactions among parties involved in the healthcare domain with respect to privacy regulation [21,22].

1.3. Cloud Computing in Healthcare

Cloud computing appears to be the dreamed-of vision of the healthcare industry; it refers to means of storing and accessing data and programs over the Internet instead of the computer's hard drive [23]. Cloud computing meets the need for healthcare information sharing directly with various healthcare-related parties over the Internet, regardless of

their location or the amount of data being shared. The use of cloud computing in the healthcare sector has increasingly been highlighted as having great potential for facilitating data-driven innovations [24]. It offers functionality for managing information in a distributed, ubiquitous, and on-demand network with access to a shared pool of configurable computing resources [25]. Resources in cloud computing can be rapidly provisioned and released with minimal management effort, supporting several platforms, systems, and applications [26]. Due to resource elasticity and lower operational costs, cloud computing is appealing for healthcare. This allows novel healthcare development, delivery, and use [27,28].

Due to the rapidly growing applications of e-health systems, which have been hampered by the conventional healthcare information system's lack of interoperability, cloud computing is found to be the best option for the global e-healthcare systems that are in place [29]. However, the adoption of cloud computing in the healthcare domain faces privacy-related challenges [30,31]. Such challenges are caused by the fact that medical data and information that are classified as confidential are stored on cloud servers, a virtual world where information can be easily hacked [32].

Storing data on the cloud is a major concern for consumers, preventing healthcare cloud computing adoption [31]. The authors in [33] conducted a review of security and privacy-preserving challenges in e-health solutions. The review included various privacy-preserving approaches to ensure the privacy and security of electronic health records (EHRs) in the cloud. The review revealed a number of crucial privacy challenges that must be addressed before obtaining the full potential of cloud computing in the healthcare domain. The authors wrote "Studies must focus on efficient, comprehensive security mechanisms for EHR and also explore techniques to maintain the integrity and confidentiality of patients' information".

Cloud computing's biggest drawback in healthcare is third-party data storage [31]. Although data are usually encrypted, the owners require control over their data to perform operations such as updating records. In the normal process, data transferred to the cloud go through traditional encryption methods for security reasons; however, the data holder needs to decrypt the data whenever an operation is required on it. The data user provides the private key to the cloud provider to decrypt the data in order to execute any required calculations. The decryption of the data on the cloud provider's side causes privacy and confidentiality issues. Moreover, a patient's record may include information that might not always be needed for all different instances of medical treatments; for example, a patient who has a certain sexual disease might not want a practitioner at an emergency practice to access and read information related to such a disease when it is not needed in that particular treatment instance, therefore, accessing such unneeded information may also cause a breach of the patient's privacy.

Thus, to integrate cloud computing into healthcare while protecting patient privacy, a design must overcome these privacy issues. This paper proposes a cloud architecture design for storing and sharing patients' health data with respect to privacy and confidentiality. The proposed architecture adopts three main approaches to protecting the privacy and confidentiality of information: (1) Grouping patients' health information according to the need for it in different instances; (2) adopting the searchable symmetric encryption (SSE) mechanism; and (3) exploiting the user identity management protocol (U-IDM). The rest of this paper is organized as follows: Section 2 presents the related work. Section 3 presents the proposed cloud architectural design. This section includes the instantiation, implementation, and testing of the system. Section 4 includes discussion, and finally, Section 5 presents the conclusion and future research directions.

2. Related Work

Patients, families, and a diverse team of frequently highly specialized healthcare workers all contribute to the delivery of healthcare. To deliver exceptional treatment, all of these team members must be engaged in a collaborative and coordinated manner. Moreover,

information about patients' health generates special value when it is exchanged and collaboratively used among different parties involved in the healthcare area [6]. The definition of the term "collaboration" in the field of healthcare includes the concept of sensibly sharing a collective perspective that includes information, norms, social expectations, activity goals, and meaning. It is the communication that occurs among healthcare practitioners when sharing information and skills regarding patient care [34].

Researchers and individual users have paid a lot of attention to healthcare information among all the shared information [35]. In the digital healthcare era, electronic health records (EHR) have captured the processes of disease occurrence, development, and treatment, which makes them of great value and an essential tool to use for medical services [36]. It is of the utmost importance to harness medical information scattered across healthcare institutions to support in-depth data analysis and achieve personalized healthcare [37]. Therefore, a comprehensive and integrated healthcare infrastructure is required to facilitate the sharing of information among various healthcare institutions and domains.

The seamless exchange of vital information among healthcare practitioners played a significant role in reducing medical errors and facilitating better integration of health-related records [38]. However, to realize the full potential of collected medical data, health-related information technology systems and products are required to seamlessly share information among each other, but unfortunately, the vast majority of medical devices, electronic health records, and other systems lack interoperability [39]. Patients' health records are often stored in a non-coded, non-structured, and non-standard form, hindering the exchange of information among health information systems [40]. Heterogeneity is currently a major challenge in the healthcare industry to achieve interoperability, especially among proprietary applications provided by different vendors [39]. For instance, a hospital may use one or more applications to share clinical and administrative information, and each application may support multiple communication interfaces and protocols that must be modified and maintained.

To achieve interoperability among different systems in the healthcare domain, several efforts have been put forth by various desperate parties. The work in [41] provided a review of some proposed cloud architectures for healthcare, along with issues in both technologies and the crucial reasons for moving forward with a cloud-based e-healthcare system. The issues of security and privacy were highlighted as barriers to the adoption of cloud computing in the healthcare domain.

An approach toward achieving interoperability between information technology systems is the Unified Modelling Language (UML) [42]. UML, or Unified Modelling Language, is a standardized modelling language used in software engineering to visually represent software systems. UML has been extensively studied for its potential to facilitate communication and interoperability between information systems. Many UML-based approaches have been proposed to facilitate the communication of information systems, such as [43,44]. UML can be used to represent the structure, behavior, and interactions of different systems and components, providing a common language and framework for communication among stakeholders. Similarly, the work in [45] proposed a new approach to specifying data integration toward interoperability based on data models such as entity-relationship (ER) and UML. The authors draw attention to a critical problem that results from the incompatibility of data models, such as the use of proprietary terminology, data structures, data formats, and semantics by different software systems. Data must be shared between software systems, and frequently, challenging data conversions or transformations are necessary. Process modelling is also difficult due to the complexity of healthcare information systems and requirements, which accounts for the slow adoption of process modelling standards [46].

Enterprise Service Bus (ESB) [47] is another approach that serves as a platform for integrating different applications and services within an enterprise. EBS is an architectural pattern whereby a centralized software component performs integrations between applications. In a service-oriented architecture (SOA) [48], EBS implements a communication

mechanism between software programs that interact with one another by acting as a central hub or mediator, enabling communication and information exchange between different systems and services.

In [49], the authors conducted a study to understand and provide ongoing research topics, challenges, and future directions concerning ESB applications. While ESB is seen as a powerful approach for enterprise integration and data exchange, its adoption in the healthcare domain requires meeting difficult requirements related to accessibility in light of the fragmentation of patients' information in heterogeneous proprietary systems.

Blockchain is another approach that is widely used in healthcare information systems due to its decentralization and security features [50]. The adoption of blockchain technology is becoming a widespread trend in distributed computing. Many researchers considered the use of it for sharing information across healthcare information systems [51].

In [52], the authors discuss the development and user experience of a personal health record (PHR) application that incorporates blockchain technology. By leveraging blockchain's decentralized and immutable nature, the application ensures secure storage and management of health-related information, granting users control over their data. The study highlights positive user feedback regarding the application's usability and effectiveness in managing health records. It also addresses challenges such as scalability and regulatory compliance. An example of a suggested blockchain-based application is found in [53]. The author proposed a distributed, smart, and secure healthcare system using blockchain and edge computing for sharing medical information across different institutions. The proposed system shares data by separating medical data processing, access control, and data sharing into local and blockchain networks. The work also presents a data-sharing security algorithm based on the value of the shared data.

The proposed system has significant drawbacks related to quality of service and information privacy. In terms of quality of service, it is difficult to guarantee consistent service quality because various validation times depend on the security level, while the move of data through edge nodes in the local network may lead to centralization and privacy concerns especially because data may not be encrypted.

Another blockchain-based work for sharing healthcare information is presented in [54]. The authors presented an attribute-based encryption system for authorization and dynamic authentication of medical on-demand services in remote medical systems. Blockchain technology in the system was exploited along with distributed database technologies to protect the integrity of information. However, the approach suffered from limitations related to centralization and security.

The authors in [55] conducted a review of blockchain-based secure sharing of healthcare data. The review included an evaluation of the development of blockchain in healthcare from various perspectives. It also analyzed the approaches to blockchain in different application scenarios. The results show that blockchain technology has an advantage in the field of healthcare, but the technology is suffering from issues, including low throughput and low scalability, which limit its adoption in the healthcare industry. Users can store information on a decentralized platform using unforgeable ledgers. Digital encryption can ensure data security and individual privacy. This technology has the potential to reduce operating costs and increase synergies while preserving the integrity of data [56]. In [57], a systematic review was conducted to summarize how blockchain technology has been used to address supply chain challenges in healthcare. The findings of the review indicated a significant but immature interest in the topic, with diverse ideas and methodologies but without effective real-life applications. The study highlights the security and privacy challenges that need to be overcome before achieving the best of what this technology can offer in the healthcare domain. Table 1 presents a summary of the literature.

Table 1. Summary of the literature.

Reference	Approach	Purpose	Challenges Highlighted
[42–45]	UML Profile for Messaging Patterns	Facilitate communication among systems	Healthcare information systems complexity
[47–49]	Enterprise Service Bus (ESB)	Applications integration to facilitate communications between applications	Fragmentation of patients' information in heterogeneous proprietary systems
[51]	A survey of the state-of-the-art schemes on secure and privacy-preserving medical data sharing of the past decade with a focus on blockchain-based approaches.	Analyze advantages and disadvantages of blockchain privacy-preserving medical data sharing.	Query efficiency, scalability, and Information management and security.
[52]	A mixed-method study using a questionnaire, in-depth interviews, and usability evaluation.	Introducing a novel blockchain-applied personal health records (PHR) application and validate its user experience.	Users trust in terms of the security of their information when shared using Blockchain.
[53]	Blockchain approach that separates medical data processing, access control, and data sharing into local and blockchain networks.	Sharing data among different local blockchain networks (different institutions).	Quality of Service and Information privacy
[54]	Proposed blockchain-based ABE scheme with multi-Authority for medical on demand in telemedicine system.	To achieve dynamic authentication and authorization with higher flexibility and efficiency.	Privacy and unauthorized access from insiders due to centralization.
[55]	A review of blockchain-based secure sharing of healthcare data.	Evaluating the development of blockchain in healthcare from various perspectives, and analyzing the approaches to blockchain in different application scenarios.	Issues related to low throughput and low scalability, which limit its adoption in the healthcare industry
[57]	A systematic review	Summarizing how blockchain technology has been used to address supply chain challenges in healthcare.	security and privacy challenges.

3. Proposed Architecture

Considering the complexity of the existing healthcare structure, where patients' health information is distributed to multiple entities such as hospitals, healthcare centers, and cloud servers, a centralized architectural design of information systems for the healthcare domain would not be suitable, especially when interoperability remains a challenging obstacle among the vast majority of healthcare information systems. A non-centralized architectural design would be the most suitable option for the healthcare sector so that disparate entities can collaborate through sharing information related to patients and their health.

In this work, a new cloud-based architecture is proposed for storing and sharing healthcare information in a privacy-preserving manner. There are two sources of information that informed the design of the proposed cloud architecture: case study findings and a literature review [58]. The characteristics of the proposed cloud architecture are the following:

- **Just-enough information disclosure**: disclosing only the right information according to the context in which it is required.
- **Accessible location of information**: Storing patients' information in one place for easy access whenever information is required.

- **Unified platform**: Accessing information through a unified platform is a key characteristic for improving healthcare services.
- **Adherence to the legal privacy-related frameworks**: The architecture should adhere to privacy-related regulations and policies such as HIPPA and the Information Privacy Act when using information.
- **Patients control**: Patients should have a means of controlling who can access their information.
- **Cloud provider blindness**: The cloud provider should not be able to read or access patients' information that is stored on the cloud.

3.1. Architecture Fundamental Aspects

There are two fundamental aspects of the proposed architecture design that enable it to store and share patient health-related information in a privacy-preserving manner. The first fundamental aspect is structuring patient information into categories. This aims to eliminate the exposure of information that is not needed during instances of medical treatment.

Structuring patient information also contributes to allowing patients to have means of control over who can access their information while it is stored on the cloud. The second fundamental aspect is the use of a searchable symmetric encryption scheme (SSE) [59], which enables searching through encrypted information without decrypting it. The objective of the searchable encryption scheme is to store patient information on the cloud without allowing the cloud provider to learn the content of the stored information.

3.1.1. Structuring Patients Information

A fundamental aspect of the proposed cloud architectural design is the accommodation of patients' health information under four main categories, which were identified in the case study findings [58]. These categories are: Information that is constantly required in every patient's visit (All_V); information that is required in patients' emergency visits (Em_V); information that is required in out-patients' clinical visits (OutP_V); and information required for research purposes (R). This paper focuses on information categories that are used for medical treatment purposes; therefore, the (R) category is explained in other work.

The main goal of structuring patients' health information is two-fold: first, to limit the exposure of information in instances when it is not needed. Second, limiting the exposure of information leads to better means of privacy protection that patients desire for their health information. The proposed system design stores patients' information in three groups referred to as documents. Each document has identifying tags and contains files.

Each file has the name of the patient, the name of the document, and a sub-tag used by the application system to identify and locate it. The system's identifying tags are used to technically facilitate access to documents and do not indicate their content.

For example, and for simplification purposes, the tags used for the documents are 1, 2, and 3. All patients registered in the proposed system have their information organized into doc-1, doc-2, and doc-3. In the practical implementation of the proposed system, the information stored in each document is subject to change according to the medical treatment's changing needs. The information categories comprise information contained in different documents, therefore, accessing a category of information is a result of accessing one document or more. For example, when a user has the right to access information about a patient in an emergency setting (Em_V), doc-1 and doc-2 are released to the user, while a combination of doc-1, doc-2, and doc-3 are released for users who have access to all information related to patients' health (OutP_V). Table 2 illustrates the information categories and their comprising documents.

Table 2. Information categories and their comprising documents.

Doc-1	Doc-2	Doc-3
Full Name Date of Birth Gender Ethnicity Significant Conditions NHI Number Current Medication	Drug Allergies Discharge, Summaries Blood Type Laboratory Results Next Kin	All the information that is not contained in doc-1 and doc-2

3.1.2. Searchable Symmetric Encryption (SSE)

Searchable symmetric encryption is a cornerstone of the proposed system architecture. The main objective of the proposed system architecture is to store patients' information on the cloud without allowing the cloud provider to read it. Achieving this is considered easy but not practical without a mechanism that enables searching through encrypted information without decrypting it. The proposed system employs a searchable symmetric encryption (SSE) approach. The SSE approach enables outsourcing data storage while preserving the ability to selectively search over it. There are three models for searching on encrypted data identified in the literature: searching on public-key encrypted data [60], single-database private information retrieval (PIR) [61], and finally searching on private-key encrypted data [59] which is the approach employed in the proposed cloud architecture. For consistency purposes, the private key is denoted by secret key (Sk) throughout the paper. In the secret-key-encrypted data model, the data are encrypted by the user and are organized in an arbitrary way prior to encrypting it. The data are stored on a server in encrypted form, and decrypting it can only happen with the S_k. In this model, the initial work for the user is large when the data are large, while subsequent work, such as accessing the data are small. The user's work is large because data pre-processing requires performing a number of processes to facilitate searchability while it is encrypted. Structuring data as part of the pre-processing allows for efficient access to relevant data. In this proposed system, information is partitioned into portions denoted by documents, as explained earlier. For every patient, there is a root secret key (S_{kR}) that is used to encrypt three secret keys (S_k). Secret keys are used to encrypt patients' documents (doc-1, doc-2, and doc-3). Each document is encrypted with its corresponding S_k. Indexes and trapdoors—explained further in this section—are generated to identify and decrypt documents, respectively. An important property of the secret-key-encryption approach is that anyone who can decrypt information for a document can also decrypt any file in that document. This means anyone who has access to a document can have access to all files within that document.

The main goal of employing the SSE approach is to store patients' health information on the cloud in a searchable manner so that only authorized parties can access it. Moreover, the cloud provider can never learn anything about the information stored; it receives encrypted information to store and releases it without decrypting it.

The decryption of each document under the secret root key requires the secret key for it, which is released upon authenticated and authorized user requests. The cloud provider is not informed about the content of any document; therefore, the challenge remains in identifying encrypted document(s) without decrypting them. The searching capability of the SSE approach is achieved using a secure index mechanism [62]. The secure index is a structure of data that stores document collections while supporting efficient keyword search. For example, given a keyword (w), the index returns a pointer to the documents that contain it. The secure index works by searching for an exact string match in encrypted documents. Every document contains a collection of encrypted strings, and a string is chosen to be the searching keyword for the document that contains it. The selected keyword is computed using the secret key with which the entire document is encrypted. The resulting ciphertext is then used to search for an exact match in documents. For example, a keyword in a document is "Basic-Information". This keyword is computed using the secret key of

the encrypted document, and the resulting ciphertext is, e.g., "JK^78Uo8361KL$#VWL". The combination of the keyword and its corresponding ciphertext is then used to identify the document that contains the keyword "Basic-Information". However, a keyword may appear in different documents, therefore, a number of keywords and their corresponding ciphertexts are put together in an encrypted index and corresponding trapdoor to assure the accuracy of document identification. Alternatively, a document's unique name can be used to achieve the same outcome accurately, such as "doc 1". Table 3 demonstrates an example of an encrypted index generated for a document listed under a secret root key and its corresponding trapdoor.

Table 3. Encrypted index and Trapdoor for a document.

Doc-1 Encrypted Index		**Doc-1 Trapdoor**	
Basic Info	JK^78Uo8361KL$#VWL	**S_k**	JK^78Uo8361KL$#VWL
Significant	RM*#%H)GIDU784K2%		RM*#%H)GIDU784K2%
Medication	B&0*9QOVPI(068B%#O		B&0*9QOVPI(068B%#O
Doc-1	APV*89&@JE)<I@DO$		APV*89&@JE)<I@DO$

To achieve the properties of the SSE approach, authors in [59] proposed the below five algorithms: which are the key generation algorithm (**KeyGen**), the key derivation algorithm (**KeyDer**), index generation algorithm (**IndexGen**), trapdoor generation algorithm (**Trap**), and the search algorithm (**Search**). Below is the description of these algorithms:

KeyGen Algorithm: A probabilistic algorithm that sets up the searchable encryption scheme. It is responsible for generating a secret root key for the patient's documents as a collection. It takes a security parameter k and generates a secret root key (S_{kR}) for the patient S_{kR}. This key is used for wrapping and unwrapping the secret keys of all documents that belong to the patient.

$$\text{KeyGen}\left(1^{k}\right) \rightarrow S_{KR}$$

KeyDer Algorithm: Employed for generating a secret key (S_k) for each document listed under the secret root key (S_{kR}). It takes the document name and secret root key (S_{kR}) as input and generates a secret key (S_k) for the document. This secret key will be used to encrypt and decrypt the information contained in its corresponding document.

$$\text{KeyDer}\left(sk_{(i_1 \ldots . i_{n-1})},\ (i_1 \ldots . i_n)\right) \rightarrow\ sk_{(i_1 \ldots .\ i_n)}$$

$$Fsk_{(i_1 \ldots . i_{n-1})}(i_n)$$

IndexGen Algorithm: Responsible for generating an encrypted index (I) for every document. It takes a number of keywords in a document, such as the name of the document or its title, and encrypts them using the document secret key (S_k). The output of the IndexGen algorithm is an encrypted searchable index I for every document to be used for searching.

$$\text{IndexGen}\left(sk_{(i_1 \ldots . i_n)},\ (i_1 \ldots . i_n),\ word_{w_1 \ldots .\ w_n}\right) \rightarrow\ sk_{(i_1 \ldots .\ i_n)},\ I$$

$$\text{Enc}\left(sk_{(i_1 \ldots .\ i_n)},\ I\right) \rightarrow\ C_1$$

Trap Algorithm: Responsible for generating trapdoors for documents. It takes the secret key of a document and keywords' ciphertexts as input and outputs a corresponding trapdoor (T), which is used for decrypting the document.

$$\text{Trap}\left(sk_{(i_1 \ldots . i_n)},\ (i_1 \ldots . i_n),\ word_{(w_1 \ldots .\ w_n)}\right) \rightarrow\ T$$

Search Algorithm: Uses the decrypted index and the trapdoor for one document to find it. It takes the decrypted index and the trapdoor as inputs and identifies the encrypted document as an output.

$$\text{Search } (C_1, T_1) \rightarrow \text{ Encrypted ciphertexts}$$

Similarly, in the proposed system, the process of preparing patients' information for storage involves five steps:

1. Generating a secret root key (S_{KR}) for the patient.
2. Generating a secret key (S_K) for every document of patient information and choosing a keyword for each document.
3. Keywords are encrypted using their corresponding secret keys, and the resulting ciphertexts are listed to form an encrypted index.
4. Trapdoors are then created, which involve combining the secret keys with the ciphertexts. Trapdoors will be used to identify and decrypt documents.
5. Documents are encrypted using their corresponding secret keys.

By following the above five steps, it becomes feasible to search for patients' documents while they are encrypted without having to perform decryption operations on them. Further explanation of how information is obtained from the cloud and decrypted is provided in the following section.

3.2. Architectural Design and Components

The proposed architecture comprises five architectural components that are required for storing healthcare information on the cloud and collaboratively using it in a privacy-preserving manner. These components are Requesting Agent, User Application, Cloud Service Registry, Secret Key Server, and Cloud Service Provider. Each component is responsible for accomplishing certain tasks as a contribution to achieving the main objectives of the proposed architecture. Figure 1 illustrates the architectural design and the relationship among its comprising components.

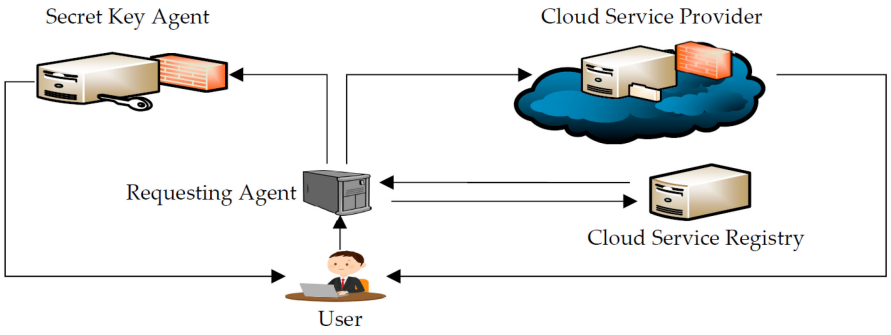

Figure 1. Proposed architectural design.

Requesting Agent

The Requesting Agent (RA) is a server that is responsible for receiving requests from users and forwarding them to both the Cloud Service Provider and the Secret Key Agent after authenticating users. It is the point of contact through which users send requests to store or access information stored on the cloud. Users are authenticated and their access rights are identified before requests are forwarded by the RA. In other words, it plays the role of the gatekeeper who does not allow unauthorized users to access the system. The RA has a limited communication channel with the users, a one-way communication channel with both the Cloud Service Provider (CSP) and the Secret Key Agent (SKA), and a two-way communication channel with the Cloud Service Registry (CSR) for users' authorization. The RA receives requests from users and only responds with information that is limited to confirmation of authentication.

The RA stores the required information for identifying patients who are registered in the system. Information stored on the RA is important for facilitating secure access to patients' information that is stored on the cloud. Every patient has a unique code referred to as a "system ID", which is generated by the RA and used for searching purposes.

When a user requests to access patient information, the patient system ID is used to identify the patient's information that is stored on the cloud. The main role of the RA in the proposed design includes receiving requests from users, authenticating users, and forwarding user requests to both the CSP and the SKA. The information stored on the RA for every patient is organized into three sections; every section contains information that is important to facilitate access to patients' information in a secure and privacy-preserving manner. Table 4 illustrates the information stored in the RA.

Table 4. Information stored on the Requesting Agent for every patient.

Full Name	D.O. Birth	NHI Number
ID	S_{KR}	doc-1 Index doc-2 Index doc-3 Index
User ID User ID User ID	☐ Email ☐ Phone Number	

Section 1 includes information that is required to identify patients in the system. Users request to access patients' information by using basic identification information such as name, date of birth, and NHI number. Section 2 includes the patient's system ID, S_{kR}, and indexes. The system ID is required to identify patients' information that is stored on both; the cloud and the Secret Key Agent (SKA). The S_{kR} is required to decrypt the trapdoors that are stored on the SKA, and the indexes are needed to identify the documents stored on the CSP. Section 3 includes information that is required by the Cloud Service Registry (CSR) for authorizing users to access patients' information. It includes a list of users who have permanent consent to access the patient's information, as well as the patient's contact details for requesting and obtaining patient consent. There are two types of consents that patients grant to users for accessing their information: permanent consents which are granted by patients to their local GPs or pre-determined medical institutions/practitioners, and temporary consents which patients grant to medical practitioners/institutions for casual incidents or clinical visits. Patients optionally grant permanent consent to users to access their information. Temporary consent is granted to a user who does not have permanent consent and requires access to a patient's information. There are two methods of requesting and obtaining temporary consent: via mobile phone in the form of a text message or via email confirmation. More information about patient consent is provided further in the paper.

StandardUser Application

The proposed system architecture requires a standard application that is installed and runs locally on users' machines. Accessing patients' information stored on the cloud can only happen through a standard user application (UA). Having a unified platform to access patients' information was identified in the case study findings as a desired characteristic of healthcare information systems; therefore, the proposed architecture design employs a standard UA through which users can access information that is stored on the cloud.

The UA plays a key role in the proposed system architecture; it facilitates means of standardization in the process of storing, accessing, categorizing, and structuring information. There are three main functions that UA is responsible for: storing, accessing, and updating patient information on the cloud. These functions are performed using buttons that are available on the UA interface; these buttons are ENROL, REQUEST, UPDATE, and RESEARCH. Further explanation about these functions is presented further in the paper.

There are a number of characteristics that UA has that enable it to store, access, and update patients' information on the system. The following are the main characteristics of the UA employed in the proposed system design:

- Standard presentation and categorization of information

Categorizing information is one of the UA's functionalities. The application organizes patient information files into three documents (doc-1, doc-2, and doc-3) before they are stored on the cloud. The UA has a standard user interface for all users. Information is accessible when it appears in predetermined fields on the user interface. Information is presented in its associated fields only when it is decrypted. Information fields remain blank when their corresponding files are not decrypted. For example, if a field on the application interface is predetermined for information related to patient mental health, this field remains blank when the logged-in user is not authorized to access the document in which the mental health file exists.

- Information pre-processing, encrypting, and decrypting

The properties of the searchable symmetric encryption (SSE) approach employed in the proposed architectural design are achieved by operations, performed by the UA. The pre-processing operations, together with encryption and decryption operations, are all performed by the UA. The UA is responsible for pre-processing the information by organizing it and encrypting it following the SSE approach before it is sent for storage. It is also responsible for requesting access information and decrypting it when it is received.

- Characteristics related to accessing patients' information for research purposes

The UA has important characteristics that are related to accessing patients' information for research purposes in a privacy-preserving manner. Entering kiosk mode, disabling certain functionalities such as copy-paste functionality, and allowing and prohibiting communication channels are all important characteristics of the UA. These characteristics aim to ensure the privacy of patients' information when used for research purposes. Further details about the characteristics related to using patients' information for research purposes are provided further in this paper.

Cloud ServiceRegistry

The proposed cloud architecture in this research employs the concept of the user identity management protocol for the cloud computing paradigm (U-IDM) proposed in [63]. U-IDM was initially proposed for cloud computing customers and cloud service providers. The main objectives of U-IDM were to achieve a set of global security objectives in cloud computing environments, which include user authentication, authorization, and accounting. It aimed to protect customers and cloud providers' infrastructures by preventing unauthorized users from gaining access to services or facilities delivered by cloud providers.

The main component of the U-IDM paradigm is the Cloud Service Registry (CSR). The CSR plays a vital role in the proposed architecture. CSR provisions access information according to users' privileges in the form of service level agreements (SLAs). Services in the context of the proposed architecture include the provision of access to patients' information that is stored on the cloud. There are three information categories that require access rights from the CSR, which are All_V, Em_V, and OutP_V. As discussed earlier, each information category contains one or more documents. The CSR grants access to information categories by providing access to the documents that form these categories. Repeating the example of the Em_V category, it is a combination of doc-1 and doc-2. Therefore, granting access to the Em_V category requires the CSR to include the name of documents or their identifying tags with the user authentication confirmation. The CSR stores the names of categories and their comprising documents' tags. A list of registered users is stored on the CSR. Each user has a record of information related to the information that they can access. Users are listed under the names of their organizations. Searching for a user requires knowing the organization he/she belongs to. Table 5 shows an example of users' lists who are affiliated with an organization.

Table 5. Example of users list in stored on the CSR.

ID#	Role	Access Privilege
29930894	Nurse	doc-1 | doc-2
29930804	Doctor	doc-1 | doc-2 | doc-3
29930832	Receptionist	doc-1
29930930	Doctor	doc-1 | doc-2 | doc-3

Nevertheless, an important task of the CSR in the proposed architecture is to obtain patients' consent for accessing their information. The CSR does not authorize users to access patients' information without the patients' consent. As mentioned earlier, when a user requests to access patient information, the RA authenticates the user and forwards the request to the CSR for authorization. Part of the information included in the RA's forwarded request includes a list of permanently authorized users to access the patient's information. This list enables the CSR to find out whether the user is granted permanent consent to access the patient's information or not. If the user is not included in the list, the CSR promptly sends a request for temporary consent to the patient, and the patient can promptly grant consent or reject it.

SecretKey Agent

The Secret Key Agent (SKA) resides on a server that stores the required information for decrypting information stored in the cloud. As explained earlier, for every patient, there are three secret keys (S_k) listed under a secret root key (S_{kR}) which are used to decrypt three documents. All secret keys are stored together with trapdoors for all documents related to one patient (under one S_{kR}). The main functionality of the SKA is to receive requests from the RA and send the required trapdoors directly to the user. SKA has a one-way communication channel with the RA, which is to receive requests, and a one-way communication channel with users to send secret keys, encrypted indexes, and trapdoors.

CloudService Provider

The cloud service provider (CSP) holds information related to patients' health. The main goal of the proposed architecture is to store all patients' information in one place, which is the cloud. The CSP serves by storing and releasing encrypted information related to patients upon users' requests. The CSP has a one-way communication channel with the RA, and a one-way communication channel with users. It receives requests from authenticated and authorized users through the RA and releases the required information in its encrypted form to users. Information stored in the cloud is contained in encrypted documents. The CSP cannot learn anything about the content of the documents stored. The cloud receives encrypted documents to store and releases them to users without performing any decryption process on the documents. The CSP employs a string match algorithm that aims to identify documents. Every patient has three encrypted documents that are labeled by the patient's system ID. Further explanation about the role of the string match algorithm is provided in the following section.

3.3. System Instantiation

The proposed architecture design aims to store healthcare data on the cloud and access it for legitimate purposes while protecting privacy. This section uses a scenario to show how the suggested architecture accomplishes this goal. The instantiation is provided in two parts: the first shows a patient enrolling in the system and storing their information on the cloud, while the second section shows how this information can be accessed for medical treatment.

Storing patient information on the cloud—Scenario

Let us assume that Bob visits a system authorized doctor and requests to enroll in the system. The doctor enters Bob's information through the user application (UA). The doctor

clicks on the **ENROL** button. The process of storing Bob's information on the cloud involves three stages: information preparation, authentication and authorization, and storage.

Stage 1: Information Preparation (Searchable Symmetric Encryption): Prior to forwarding the information to the RA, Bob's information undergoes a number of pre-processing algorithmic operations performed by the UA in preparation for storage. The operations aim to encrypt Bob's information for storage on the cloud in a searchable manner.

1. A random secret root key (S_{kR}) is generated for Bob's information using the KeyGen algorithm.
2. A number of keywords from each document are selected, and a secret key (S_k) for encrypting them is generated using the KeyDer algorithm.
3. The IndexGen algorithm encrypts selected keywords for every document using their corresponding $S_{k.}$ The goal in this step is to create an encrypted index for each document to identify it while it is encrypted.
4. After encrypted indexes are generated for all documents, the ciphertexts of keywords with their corresponding S_k for each document are grouped to be the documents' trapdoors.
5. The last step in the information preparation process involves encrypting the patient's documents and their corresponding trapdoors. Each document is encrypted using the S_k that is included in its corresponding trapdoor, and trapdoors are encrypted using the S_{kR} that was generated in the first step.

Stage 2: Authentication and Authorization: When Bob's information is pre-processed, it is forwarded to the Requesting Agent (RA) in the form of a Request of Enroll (ROE). Table 6 presents information contained in the request to enroll (ROE).

Table 6. Information included in the ROE.

Section 1	Section 2		Section 3
Organization ID Practitioner ID Patient Name Patient DOB Patient NHI Consent Method: Phone#	**S_{KR}**		
	doc-1 Index	doc-1 Trapdoor	Encrypted information doc-1 doc-2 doc-3
	doc-2 Index	doc-1 Trapdoor	
	doc-3 Index	doc-1 Trapdoor	

The ROE includes three sections that include different information, such as the following: Section 1 includes information that is required to identify (a) the doctor (organization ID and user ID), (b) Bob and practitioners who have permanent consent to access Bob's information, and (c) the method of obtaining Bob's consent to access his information. Section 2 includes information required to identify and decrypt Bob's information, and Section 3 includes Bob's encrypted information (3 encrypted documents).

When the RA receives the request, it authenticates the doctor and forwards his information (contained in the first section of the ROE) and Bob's phone number to the cloud service registry (CSR) for authorization. The CSR then sends a text message to Bob requesting consent to store his information on the cloud. The content of the message includes:

Please reply "**Yes**" to authorize **(doctor name**) from (**organization name**) to enroll you and store your health information on the system. Upon receiving a "**Yes**" reply from Bob, the CSR sends a confirmation of authorization to the RA.

Stage 3: Information Storage

When the RA receives confirmation of authorization from the CSR, it takes the following actions:

1. Generates a unique code for the patient, referred to as (System ID).
2. Sends Bob's encrypted information labeled by Bob's system ID to the cloud service provider (CSP).

3. Sends the encrypted trapdoors to the secret key agent (SKA) for storage. Information sent to the SKA is also labeled by Bob's system ID.
4. The information sent to both CSP and SKA is deleted from the RA.
5. The RA stores the following information:
    - Bob's identification information
    - Bob's system ID,
    - Bob's S_{kR},
    - Document indexes,
    - Names of users who have permanent consent to access Bob's information (if Bob has provided any),
    - Information required for obtaining Bob's temporary consent.

Below is the state-of-the-art of Bob's information while stored in the cloud:

1. Bob's information is stored in encrypted form and labeled by Bob's system-generated ID. The cloud provider is not able to learn the content of the information.
2. The trapdoors are encrypted using Bob's S_{kR} and stored on the SKA, labeled by Bob's system ID. The SKA is unable to learn the content of the trapdoors without Bob's S_{kR} which is stored on the RA.
3. The RA is the only entity in the system that can identify Bob and his S_{kR}. The RA stores all the information that is required to access Bob's information, as presented in Figure 2.

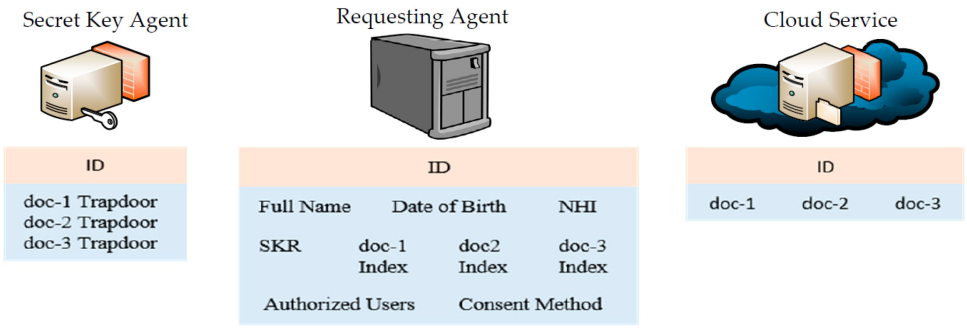

Figure 2. Bob's information stored in the proposed architecture.

Therefore, accessing Bob's information can only happen through collaborative interactions among the CSP, SKA, and RA. Compromising 1 or 2 of these architectural components will be fruitless for any disparate party in terms of accessing Bob's information.

Having discussed the process of storing Bob's information on the cloud, the following subsection presents the process of accessing Bob's information for genuine reasons, such as providing healthcare to a patient. For this reason, the scenario presented in the following subsection involves the same patient (Bob) requiring healthcare assistance from a different medical practitioner who also has access to the system.

- Accessing stored patient information—Scenario

Bob visits the hospital for urgent medical treatment. He walks into the emergency department and meets one of the nurses in charge. The nurse requires accessing Bob's information and updating his records to include information about his visit, his medical condition, and other information related to his visit.

3.3.1. Protocol to Access Information Stored in the Cloud

The process of accessing Bob's information comprises the following four stages:

Stage 1: Generating user request

The nurse enters Bob's basic information into the user application and clicks on the **REQUEST** button to generate a user request that is forwarded to the RA. The user request includes information about both Bob and the nurse.

Stage 2: Authentication and Authorization

When the RA receives the request from the user (nurse), it authenticates the user and forwards the request to the CSR for authorization. For this, the RA takes the following actions:

1. It searches for Bob's information using his basic information and finds his system ID.
2. It sends a request for authorization to the CSR. The request includes the following information:
   - Information that is required to identify the nurse (user ID);
   - List of users who have permanent consent to access Bob's information;
   - Bob's mobile number for requesting his consent if required in this particular instance.

When the CSR receives the request from the RA, it takes the following actions:

1. It searches for the nurse's information to identify her access rights to patient information. This happens by searching through the list of users that is stored locally on the CSR.
2. It checks if the nurse is permanently consented to access Bob's information using the list of users who have permanent consent to access Bob's information.
3. The CSR finds out that the nurse is allowed access doc-1 and doc-2 (Em_V) of patients' information, but she is not permanently consented to access Bob's information, therefore, Bob's consent is required.
4. The SCR sends a request of consent to Bob in the form of a text message. The content of the message includes:

*Please reply "**Yes**" to temporarily authorize (**nurse name**) at (**organization name**) to access your health information.*

Upon receiving a YES from Bob, the nurse becomes temporarily authorized to access Bob's information. The CSR sends a confirmation of authorization to the RA. The confirmation of authorization includes the information categories that the nurse can access (doc-1 and doc-2), and confirmation of obtaining Bob's consent to access his information. The nurse is then temporarily added to the list of authorized users (stored on the RA) as a temporarily authorized user. However, any authorization granted by the CSR remains valid for 1 h; after that, it is automatically deleted from the list of authorized users.

Stage 3: Releasing Information

Upon receiving confirmation of authorization from the CSR, the RA forwards requests to both the CSP and the SKA to send Bob's information to the nurse. As illustrated in Figure 3, the request to the CSP includes:

- Bob's system ID
- Indexes of doc-1 and doc-2
- The nurse's application address

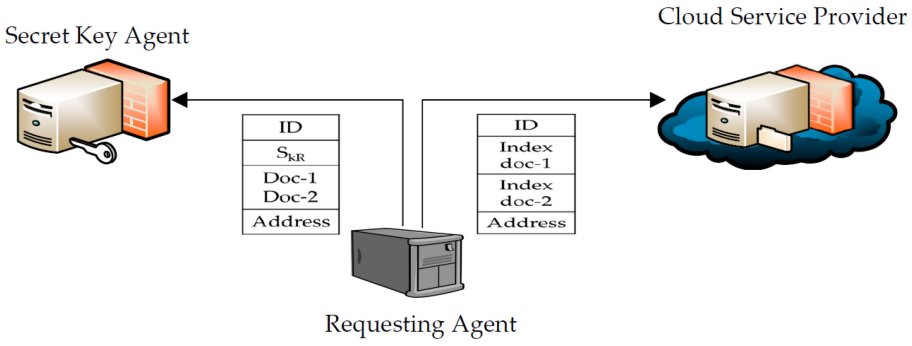

Figure 3. Requesting information from CSP and SKA.

While the information included in the request to the SKA includes:

- Bob's system ID and S_{kR}

- Trapdoor-1 and trapdoor-2 tags
- The nurse's application address

As illustrated in Figure 4, when the RA requests are received by the CSP and the SKA, they take the following actions:

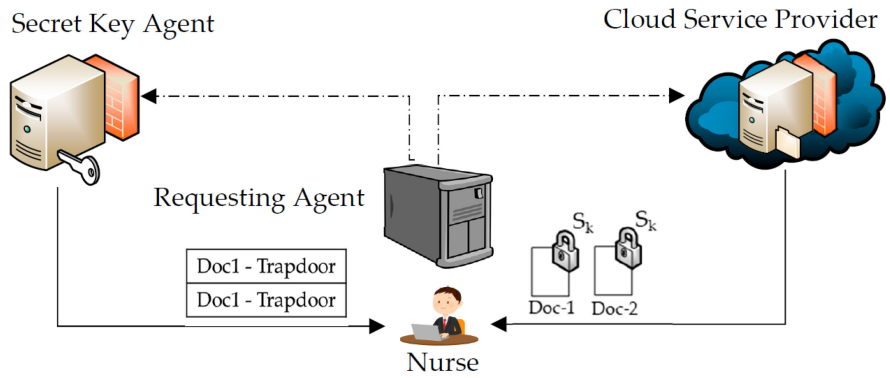

Figure 4. Releasing information to the nurse application.

The CSP:

1. Searches for Bob's information using Bob's system ID.
2. Searches for the doc-1 and doc-2 using their indexes.
3. Sends the identified documents (doc-1 and doc-2) to the nurse using her application physical address.

The SKA:

1. Searches for the encrypted trapdoors using Bob's system ID.
2. Decrypts the trapdoors using Bob's S_{kR}.
3. Sends trapdoors for doc-1 and doc-2 to the nurse application using her physical address.
4. Re-encrypts the trapdoors using the same S_{kR} and drops the S_{kR} (deletes it).

Sage 4: Decrypting Information

When the information from CSP and SKA is received by the nurse's application, doc-1 and doc-2 are identified and decrypted using their corresponding trapdoors. When information is decrypted, files in each document appear in their predetermined fields on the nurse's UA. Fields that belong to the files contained in doc-3 remain blank. The nurse application stores the trapdoors temporarily to be used for re-encrypting the information, which is further explained in the following subsection.

3.3.2. Updating Patient Information

Assuming that the nurse has made an update on Bob's information, such as information related to current medication. The nurse clicks on the **UPDATE** button on her UA interface. As illustrated in Figure 5:

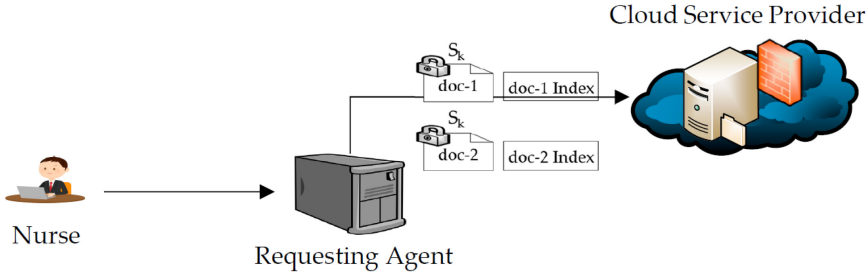

Figure 5. Updating patient information.

Stage 1: The nurse's UA encrypts doc-1 and doc-2 using their secret keys obtained from the trapdoors. The encrypted information (doc-1 and doc-2) is forwarded to the RA.

Stage 2: The RA receives the request from the nurse and does the following:

1. It searches for Bob's information to identify him.
2. It searches through the list of authorized users to access Bob's information and finds the nurse listed as a temporarily authorized user to access doc-1 and doc-2 of patients' information.
3. It forwards the encrypted information, indexes for doc-1 and doc-2, and Bob's system ID to the CSP.

Stage 3: When the CSP receives the information from the RA, it takes the following actions:

1. It searches for Bob's encrypted documents using indexes and the system ID.
2. It identifies the documents using the indexes and replaces them with the new ones.
3. It deletes the indexes received from the RA.

3.4. Architecture Implementation

Having discussed the components and protocol of the proposed architecture, the architecture is further implemented and adapted for data sharing. The proposed architecture was built using Amazon Web Services (AWS) which provides cost-effective cloud computing solutions [64]. The AWS Software Development Kit (SDK) was used with Java to implement and test the proposed architecture design and validate its concept. The implementation of the proposed architecture aimed to elaborate on how the proposed architecture enables the collaborative use of patients' information in a privacy-preserving manner. Figure 6 illustrates the diagram of the architecture implementation on AWS.

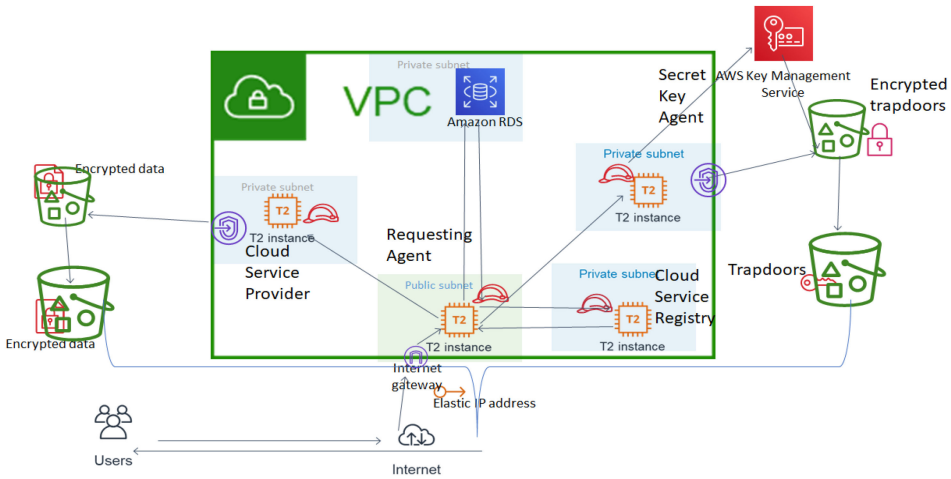

Figure 6. AWS implementation diagram.

The main objective of the implementation was two-fold: first, to elaborate on how patients' information can be collaboratively shared and used in the proposed architecture with assurance of its privacy protection, and second, to illustrate how patients' information is protected from a number of privacy-related threats, including confidentiality and unauthorized access.

The elaboration is presented in two parts: the first part presents a scenario that involves a patient who walks into a hospital for urgent medical treatment and is seen by a nurse. The goal of this part is to show how a user (a nurse) can access a patient's information according to certain access rights without questioning the privacy of the information. The elaboration also aims to validate the concept of information separation in real cloud-based application contexts.

The second part of the elaboration presents the results of tests that have been performed on the implemented architecture. The architecture was tested in terms of its ability to preserve the privacy of information while it is stored on the cloud. Four tests were performed, which covered the following aspects:

- The ability of the cloud provider to access and read the information that is stored in the cloud;
- The ability of unauthorized users to access patients' information stored in the cloud.

The following section presents information about the implementation of the cloud architecture in terms of its architectural components and privacy-preserving techniques. The implementation aims to mirror the proposed architectural design in terms of the employed components and the privacy-preserving techniques, for the goal of validating the concept of the proposed architectural design.

3.4.1. Implementation Setup

For the implementation of the proposed architecture, a virtual private cloud (VPC) was created using AWS, as seen in Figure 7. The VPC represents an important visualization tool in our research, showcasing valuable insights; it represents the entire proposed system architecture, and it is completely isolated from the Internet.

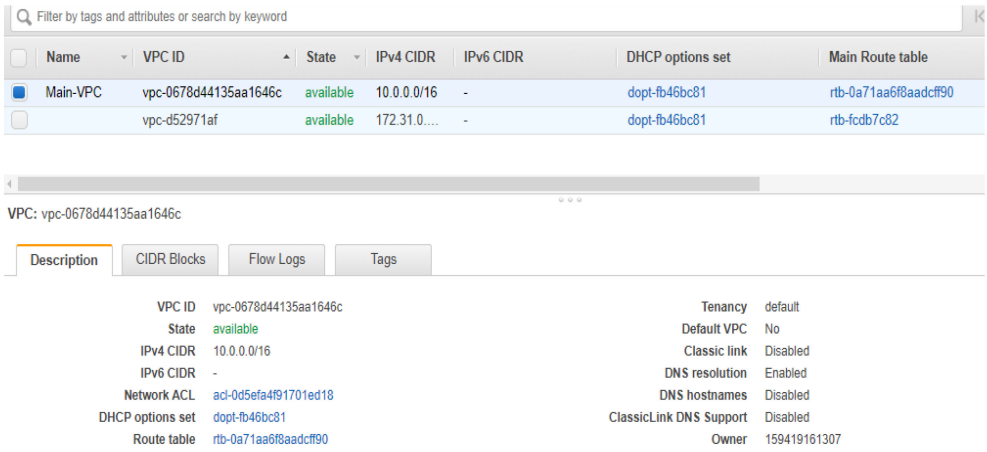

Figure 7. Isolated virtual private cloud.

The VPC has two subnets: the first subnet does not allow access from the Internet; it is only accessed locally. The second subnet enables Internet access to allow communication with the user application.

The public subnet has limited access and could be communicated through the Internet; however, only registered IP addresses can communicate with it. The EC2 represents the RA in the proposed design. Access to it happens only through a particular port.

Moreover, the database in the implemented design only accepts SQL traffic. Patients' documents and their associated trapdoors are stored in two different places that are not accessible through the Internet, as seen in Figure 8. Access to them can only happen locally.

Access to patients' documents can only happen through requests from the CSR, while the trapdoors can only be accessed through the SKA. Documents stored on the cloud are all encrypted. Figure 9 presents doc-1, which was downloaded directly from the cloud in the implemented system, highlighting the inability to learn the content of the document without going through SKA and CSR.

Accessing Patientinformation—Scenario

For the purpose of testing the architecture, dummy information about three patients was used. Each patient had three documents in the system. Documents were all encrypted and stored in the cloud database. Encrypted documents were stored on one database (CSP) and their trapdoors were stored on a different database (SKA). The illustration

involved a patient (Bob) who walked into the hospital for urgent medical treatment. The nurse wishes to access Bob's information to update information regarding his visit and current medication.

The nurse needs to login to the system for authentication purposes. The user is required to have username and password which are entered through the standard user application. Access to the system only happens through the user application. When the RA receives the user credentials (username and password), it searches for the user information in the list of registered users. The RA in the implemented architecture had a database that contains names of registered users, this database was used for authenticating users. When the user is found, authentication is confirmed.

User authorization

When the user is authenticated by the RA, another window pops up on the user application for entering Bob's basic information. Bob's information is used by the RA to identify him in the system. When information is entered by the nurse and forwarded to the RA, the RA searches for the patient in the list of the registered patients in the system. The RA in the implemented system had another database that contains information about all patients enrolled in the system. When the patient is found, the CSR is called for authorization. The RA sends Bob's information to the CSR along with the users' information. Figure 10 presents the code which was used in the implementation of the authorization process.

The CSR searches for the user in the list and finds out that the user is a nurse and is allowed to access Doc-1 and Doc-2 of patients' information. Table 7 is the table used by CSR to authorize users. The CSR confirms to the RA that the user is allowed to access Doc-1 and Doc-2 of Bob's information. The assumption made here was that Bob has received a text message from the CSR and has granted consent for the nurse to access his information.

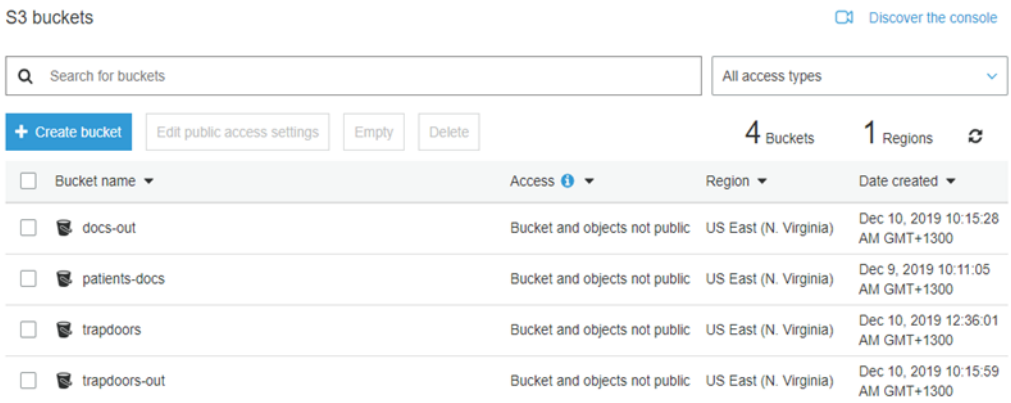

Figure 8. Separation of information stored on the cloud.

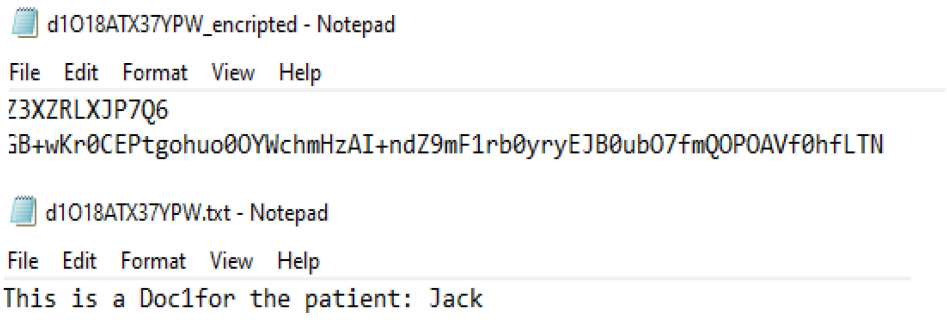

Figure 9. Encrypted document downloaded directly from the cloud.

```
//Calling Cloud Service Registry

    CloudServiceRegistry csr = new CloudServiceRegistry();
    String authorisedDocs=csr.getAuthorisedDocumentList(userId, SysId);
    System.out.println(authorisedDocs);
    String [] outputDocIndex= new String[3];
    String [] outputKeyIndex= new String [3];
    String[] docsList=null;
    String ClientReference="";
    int numberOfAuthorisedDocs=0;
    if(authorisedDocs==null || authorisedDocs.equals("None"))
        {
        dout.writeUTF("No Access");
        dout.flush();
        }
    else {
        docsList = authorisedDocs.split(",");
```

Figure 10. The implementation of the authorization process.

Table 7. The users' table used by CSR to authorize users.

SysId#	UserId	Role	Access Privilege
29930894	7777	Receptionist	doc-1
29930804	8888	Nurse	doc-1 | doc-2
29930832	9999	Specialist	doc-1 | doc-2 | doc-3
29930930	6666	Receptionist	doc-1

ReleasingInformation

When the RA receives confirmation of authorization from the CSR to access Doc-1 and Doc-2 of Bob's information, it does the following:

1. It sends Bob's system ID and indexes of Doc-1 and Doc-2 to the CSP. This was implemented using the code presented in Figure 11a.
2. It sends Bob's system ID and trapdoor tags to the SKA. This was implemented using the code presented in Figure 11b.

DecryptingInformation

In response to the requests received from the RA, the CSP searches for Doc-1 and Doc-2 using their indexes and send them to the user. And the SKA does the same for the trapdoors and sends them to the user. The user then has two encrypted documents and two trapdoors. The user application associates trapdoors to their corresponding documents using the string exact match mechanism. The secret keys in the trapdoors are then used to decrypt the documents.

3.4.2. Testing the Architecture

One of the main requirements of storing healthcare information on the cloud is the protection from unauthorized users. The cloud architecture was tested in terms of its ability to prevent unauthorized cloud users from accessing the information. Four tests were performed against the implemented system, summary of the tests and results are presented in Table 8.

```java
public class CloudServiceProvider {
    public CloudServiceProvider() {

    }
    public boolean sendDocsToClient(int numberOfDocs, String[] docIndexes, String[] outputDocNames) {

        try {
            S3Sample s3 = new S3Sample();
            File f=null;
            for(int i=0;i<numberOfDocs;i++)
            {
                f=s3.getDocument(docIndexes[i]);
                //Thread.sleep(500);
                s3.PutDocumentToOut(outputDocNames[i], f);

            }

            s3.listObjectInBuckt("docs-out");
            return true;
        } catch (Exception e) {
            // TODO: handle exception
            System.out.println("CloudServiceProvider.sendDocsToClient()");
            System.out.println(e.getMessage());
            return false;
        }


    }

}
```

(**a**)

```java
public class SecretKeyAgent {
    public SecretKeyAgent() {

    }

    public boolean sendKeysToClient(int numberOfKeys, String[] keyIndexes, String[] outputKeyNames) {

        try {
            S3Sample s3 = new S3Sample();
            File f=null;
            for(int i=0;i<numberOfKeys;i++)
            {
                f=s3.getTrapdoor(keyIndexes[i]);
                //Thread.sleep(500);
            s3.PutTrapdoorToOut(outputKeyNames[i],f );

            }

            s3.listObjectInBuckt("trapdoors-out");
            return true;
        } catch (Exception e) {
            // TODO: handle exception
            System.out.println("SecretKeyAgent.sendKeysToClient()"+e.getMessage());
            return false;
        }


    }
```

(**b**)

Figure 11. (**a**) The code used for the CSP in the implementation of the system. (**b**) The code used for SKA in the implementation of the system.

Table 8. Summary of architecture test results.

Test	Description	Result	Pass/Fail
Unauthorized System Access	Unauthorized user made request to download a trapdoor that is stored on the cloud	Access Denied	Pass
Getting access to documents stored on the cloud	A document downloaded without having the secret key.	Document was viewed in its encrypted form	Pass
Unauthorized operations	Attempt made to access the server or the database by sending queries	Access Denied	Pass
Unknown users	Unregistered user attempted to log in to the system	Login failed	Pass

4. Discussion

The adoption of cloud computing for healthcare information systems is a major improvement in the healthcare domain due to the significant benefits that cloud computing technology offers. The proposed architecture enables storing and sharing patient information in a privacy-preserving manner. The proposed cloud architecture will serve the healthcare domain by storing all patients' health information in one place (cloud) so that genuine users can access it regardless of their locations. The system instantiation of how the designed architecture works in terms of sharing healthcare information was presented in a scenario-based fashion. The feasibility and usability of the proposed architecture were confirmed, and the validity of the architecture in terms of preserving the privacy of information was successfully tested and proven.

4.1. Privacy-Preservation

Information in the proposed architecture is encrypted before it is sent to the cloud for storage. The cloud stores encrypted data without decryption details (secret keys). This separation of information (encrypted information and secret keys) makes it difficult for cloud providers to learn the content of the information by decrypting it. Exploiting the searchable symmetric encryption scheme (SSE) ensures that cloud providers can fulfill users' requests by releasing the needed information without decrypting it. The decryption of information can only happen between genuine parties.

Dividing patients' information into many divisions according to the need for it overcomes the issue of unnecessary disclosure of information. The designed architecture categorizes patients' information into four categories, of which three are for medical treatment purposes, namely All_V, Em_V, and OutP_V, and one is for research purposes (R). Medical practitioners do not always require accessing the entire patient's information every time medical treatment is needed. For example, information about a sexual disease may not be needed for urgent medical treatments such as car accidents or minor incidents such as skin wounds and cuts. This was derived from the findings of the case study data analysis and supported by the literature.

Nevertheless, the designed architecture requires obtaining patients' consent whenever accessing their information is required. The user identity management protocol (U-IDM) preserves the confidentiality of the information that is stored on the cloud and grants patients a means of control over who can access their information.

4.2. Security Analysis

The proposed system design prioritizes privacy and security when storing and sharing healthcare data on the cloud. The proposed system design includes several security approaches at different levels of the architecture.

User Application Level

The user application in the real implementation of the system design allows only certain operations to be performed by the user. Users in the proposed system design are given accessibility to the system that is controlled by the enabled features and functions of the user application. For example, a nurse's log-in credentials enable certain functions on the user application to access the system, meaning that a nurse cannot perform operations to modify the way information is stored on the system. Moreover, the encryption and decryption processes are not controlled by the user; they are performed internally by the user application. The user application in the real implementation of the system may have the characteristic of hiding all information that is related to the encryption and decryption of information.

Access control

The proposed system design employs the concept of user identity management (U-IDM), which is the Cloud Service Registry (CSR). The CSR is a component that is not located at the user's side, meaning that users cannot attempt to add or modify access rights to the system. Moreover, accessing the system can happen through requests that are sent from the user application to the Requesting Agent (RA), who authenticates and authorizes users before their requests are processed further in the system.

The collaboration between the RA and the CSR is the only way to forward users' requests to access the system. Therefore, there are three security stations in the system that the user must go through to access the system, as presented in Figure 12.

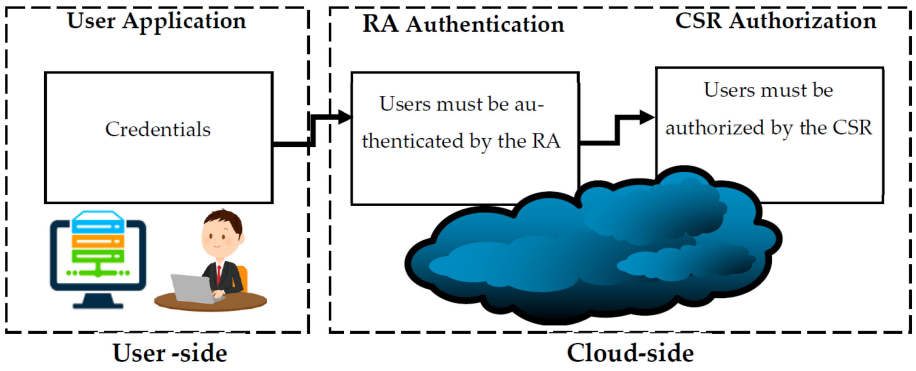

Figure 12. Security stations in the proposed system.

The first station is on the user side, which involves entering the user's credentials to access the system. Users' credentials in the real implementation can be through information entered by the system (user ID and password) or other forms of credentials such as figure prints. The second station is on the cloud side, which involves authenticating the user. The user must be authenticated by the RA component before the authorization process takes place. The third station involves the authorization process. Once a user is authenticated by the RA, the authorization process happens through the CSR. The CSR is not accessible by the user; it only communicates with the RA component.

The separation of information

The separation of information while it is stored in the system makes it difficult to perform any unauthorized actions that could lead to reading the information that is stored on the cloud, especially because the cloud provider cannot learn the content of the information that is stored on the cloud. The proposed system design provides means of security for the information in its simplest implementation due to: (1) Storing encrypted information and decryption keys on different components of the architecture; and (2) requiring the

collaboration of three different components of the architecture to gain access to the information that is stored in the system. The information required to identify patients and their information is stored on the RA. Therefore, compromising any of these three components (CSP, SKA, or RA) will be fruitless to any party in terms of reading the information stored on the cloud.

4.3. Architecture Limitation

In the practical implementation of the proposed architecture, there are some bandwidth-related issues that need to be addressed before obtaining the best of what the proposed architecture can offer in terms of information sharing. Bandwidth can be a significant issue when adopting cloud computing in healthcare, especially when it comes to accessing large amounts of data, such as medical images or electronic health records (EHRs) [65]. These issues may include:

1. Network congestion: Healthcare organizations may experience network congestion when multiple users are accessing cloud-based services simultaneously. This can cause slowdowns, latency, or even a complete loss of service.
2. Geographic location: Healthcare providers located in remote or rural areas may not have access to high-speed Internet, which can make accessing cloud-based services difficult.
3. Data-intensive applications: Certain healthcare applications, such as EHRs or medical imaging systems, generate large amounts of data that need to be transferred over the Internet. This can be a bandwidth-intensive process, which can lead to slowdowns and performance issues.
4. Security: High-bandwidth applications may require additional security measures to protect patient data. This can further slow-down the data transfer process and add to the bandwidth requirements.
5. Cost: Higher bandwidth requirements can result in increased costs for healthcare organizations, which may make cloud adoption less feasible.

To mitigate these bandwidth issues, healthcare organizations can take several steps to improve their network infrastructure and better support the bandwidth requirements, enabling them to benefit from the advantage that cloud computing offers in terms of sharing information. These steps include:

1. Assessing their current bandwidth requirements and planning for future needs. This can help ensure that their network infrastructure can handle the bandwidth requirements of cloud-based applications.
2. Investing in high-speed Internet connections and upgrading network infrastructure, including routers and switches.
3. Considering cloud providers that offer content delivery networks (CDNs) to minimize latency and speed up data transfer times.
4. Implementing data compression and deduplication techniques to reduce the amount of data that needs to be transferred over the Internet.
5. Prioritizing network traffic to ensure that bandwidth-intensive applications are given priority over less critical applications.

5. Conclusions

In conclusion, information about patients' health generates special value when it is exchanged and collaboratively used among different parties involved in the healthcare domain. Cloud computing technology appears to be the dreamed-of vision of the healthcare industry because it matches the need for healthcare information sharing directly with various healthcare-related parties over the Internet, regardless of their location and the amount of information being shared. However, the adoption of cloud computing in the healthcare domain has always been hindered due to many challenges, of which information privacy is a major one. In this work, a cloud architecture was proposed for healthcare information systems to collaboratively share and use information in a privacy-preserving

manner. The proposed architecture was implemented and tested in terms of its ability to share information in a privacy-preserving manner. Potential challenges that may arise in the practical implementation of the proposed architecture were highlighted, and a recommended set of actions was provided in response to these challenges. The research findings and outcomes provide multiple directions for extending and expanding upon the scope and focus of the present research.

First, getting feedback from medical practitioners on the prototype of the designed architecture is an important direction for future research. The proposed architecture has satisfied the need for sharing healthcare information in a privacy-preserving manner; however, getting feedback from medical practitioners and experts in the healthcare domain may further validate and improve the design of the architecture to best serve the domain.

Second, patients' information in the present research has been categorized into four categories, of which three were for medical treatment purposes; however, a research direction would refine these categories to further limit the exposure of information when it is needed for medical treatment purposes. This direction would require deeper knowledge in the medical field to allow for feeding the research with more technical data related to what and when patients' health information is needed.

Moreover, the proposed architecture allows for manually enrolling patients and storing their information on the cloud; however, healthcare data are collected today using various advanced methods such as mobile devices, wearable sensors, and home wireless networks that can automatically transmit and receive data. Researchers have proven that utilizing the data collected through these methods contributes significantly to healthcare service improvement. Therefore, a research direction can be to expand the proposed architecture design to accommodate patient-generated information that is collected by these data collection methods.

Author Contributions: F.A.—System architecture design, system implementation and validation. J.G. provided expertise that greatly assisted and guided the research. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gupta, M.; Thirumalaisamy, M.; Shamsher, S.; Pandey, A.; Muthiah, D.; Suvarna, N. Patient health monitoring using feed forward neural network with cloud based internet of things. In Proceedings of the 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 28–29 April 2022.
2. Mamlin, B.W.; Tierney, W.M. The Promise of Information and Communication Technology in Healthcare: Extracting Value From the Chaos. *Am. J. Med. Sci.* **2016**, *351*, 59–68. [CrossRef] [PubMed]
3. Neves, A.L.; Carter, A.W.; Freise, L.; Laranjo, L.; Darzi, A.; Mayer, E.K. Impact of sharing electronic health records with patients on the quality and safety of care: A systematic review and narrative synthesis protocol. *BMJ Open* **2018**, *8*, e020387. [CrossRef] [PubMed]
4. Kalkman, S.; van Delden, J.; Banerjee, A.; Tyl, B.; Mostert, M.; van Thiel, G. Patients' and public views and attitudes towards the sharing of health data for research: A narrative review of the empirical evidence. *J. Med. Ethics* **2019**, *48*, 3–13. [CrossRef]
5. Kim, E.; Rubinstein, S.M.; Nead, K.T.; Wojcieszynski, A.P.; Gabriel, P.E.; Warner, J.L. The Evolving Use of Electronic Health Records (EHR) for Research. *Semin. Radiat. Oncol.* **2019**, *29*, 354–361. [CrossRef] [PubMed]
6. Kitamura, T.; Kiyohara, K.; Matsuyama, T.; Hatakeyama, T.; Shimamoto, T.; Izawa, J.; Nishiyama, C.; Iwami, T. Is Survival After Out-of-Hospital Cardiac Arrests Worse During Days of National Academic Meetings in Japan? A Population-Based Study. *J. Epidemiol.* **2016**, *26*, 155–162. [CrossRef] [PubMed]
7. Gray, C.S.; Barnsley, J.; Gagnon, D.; Belzile, L.; Kenealy, T.; Shaw, J.; Sheridan, N.; Nji, P.W.; Wodchis, W.P. Using information communication technology in models of integrated community-based primary health care: Learning from the iCOACH case studies. *Implement. Sci.* **2018**, *13*, 87. [CrossRef]

8. Oude, W.; Velsen, L.V.; Huygens, M.; Hermens, H. Requirements for and Barriers towards Interoperable eHealth Technology in Primary Care. *IEEE Internet Comput.* **2015**, *19*, 10–19.
9. Bélanger, F.; Crossler, R.E. Privacy in the Digital Age: A Review of Information Privacy Research in Information Systems. *MIS Q.* **2011**, *35*, 1017–1041. [CrossRef]
10. White, T.; Blok, E.; Calhoun, V.D. Data sharing and privacy issues in neuroimaging research: Opportunities, obstacles, challenges, and monsters under the bed. *Hum. Brain Mapp.* **2022**, *43*, 278–291. [CrossRef]
11. Wirth, F.N.; Meurers, T.; Johns, M.; Prasser, F. Privacy-preserving data sharing infrastructures for medical research: Systematization and comparison. *BMC Med. Inform. Decis. Mak.* **2021**, *21*, 1–13. [CrossRef]
12. Public Law. Health Insurance Portability And Accountability Act Of 1996," Public Law 104-191, 104th Congress, 1996. Available online: https://www.govinfo.gov/app/details/PLAW-104publ191 (accessed on 20 May 2023).
13. Gunasekara, G.; Dillon, E. Data Protection Litigation in New Zealand: Processes and Outcomes. *Vic. Univ. Wellingt. Law Rev.* **2008**, *39*, 457–486. [CrossRef]
14. Gkoulalas-Divanis, A.; Loukides, G. Introduction to medical data privacy. In *Medical Data Privacy Handbook*; Springer International Publishing: Cham, Switzerland, 2015; pp. 1–14. [CrossRef]
15. Meng, S.; Fan, S.; Li, Q.; Wang, X.; Zhang, J.; Xu, X.; Qi, L.; Alam Bhuiyan, Z. Privacy-Aware Factorization-Based Hybrid Recommendation Method for Healthcare Services. *IEEE Trans. Ind. Inform.* **2022**, *18*, 5637–5647. [CrossRef]
16. Gürsoy, G.; Li, T.; Liu, S.; Ni, E.; Brannon, C.M.; Gerstein, M.B. Functional genomics data: Privacy risk assessment and technological mitigation. *Nat. Rev. Genet.* **2021**, *23*, 245–258. [CrossRef] [PubMed]
17. Tanriverdi, M. A Systematic Review of Privacy Preserving Healthcare Data Sharing on Blockchain. *J. Cybersecur. Inf. Manag.* **2020**, *4*, 31–37. [CrossRef]
18. Blackman, S. Towards a conceptual framework for persistent use: A technical plan to achieve semantic interoperability within electronic health record systems. In Proceedings of the 50th Hawaii International Conference on System Sciences, Waikoloa Village, HI, USA, 4–7 January 2017.
19. Shahid, J.; Ahmad, R.; Kiani, A.K.; Ahmad, T.; Saeed, S.; Almuhaideb, A.M. Data Protection and Privacy of the Internet of Healthcare Things (IoHTs). *Appl. Sci.* **2022**, *12*, 1927. [CrossRef]
20. Priyanga, P.; Muthu Kumar, V.P. Cloud computing for healthcare organisation. *Int. J. Multidiscip. Res. Development.* **2015**, *2*, 487–493.
21. Casola, V.; Castiglione, A.; Choo, K.-K.R.; Esposito, C. Healthcare-Related Data in the Cloud: Challenges and Opportunities. *IEEE Cloud Comput.* **2016**, *3*, 10–14. [CrossRef]
22. Svensson, A. Challenges in Using IT Systems for Collaboration in Healthcare Services. *Int. J. Environ. Res. Public Health* **2019**, *16*, 1773. [CrossRef]
23. Griffith, E. What Is Cloud Computing? Available online: http://au.pcmag.com/networking-communications-software-products/29902/feature/what-is-cloud-computing (accessed on 21 March 2016).
24. Cresswell, K.; Hernández, A.D.; Williams, R.; Sheikh, A. Key Challenges and Opportunities for Cloud Technology in Health Care: Semistructured Interview Study. *JMIR Hum. Factors* **2022**, *9*, e31246. [CrossRef]
25. Mell, P.; Grance, T. The NIST Definition of Cloud Computing. *Natl. Inst. Stand. Technol.* **2011**, 2. [CrossRef]
26. Doukas, C.; Pliakas, T.; Maglogiannis, I. Mobile healthcare information management utilizing cloud computing and android OS. In Proceedings of the Engineering in Medicine and Biology Society (EMBC), Annual International Conference of the IEEE, Glasgow, UK, 11–15 July 2010. [CrossRef]
27. Griebel, L.; Prokosch, H.-U.; Köpcke, F.; Toddenroth, D.; Christoph, J.; Leb, I.; Engel, I.; Sedlmayr, M. A scoping review of cloud computing in healthcare. *BMC Med. Inform. Decis. Mak.* **2015**, *15*, 1–16. [CrossRef] [PubMed]
28. Zhang, R.; Liu, L. Security models and requirements for healthcare application clouds. In Proceedings of the IEEE 3rd International Conference on Cloud Computing (CLOUD), Miami, FL, USA, 5–10 July 2010. [CrossRef]
29. Sharma, M.; Sehrawat, R. A hybrid multi-criteria decision-making method for cloud adoption: Evidence from the healthcare sector. *Technol. Soc.* **2020**, *61*, 101258. [CrossRef]
30. Yüksel, B.; Küpçü, A.; Özkasap, Ö. Research issues for privacy and security of electronic health services. *Future Gener. Comput. Syst.* **2017**, *68*, 1–17. [CrossRef]
31. Raj, H.; Kumar, M.; Kumar, P.; Singh, A.; Verma, O.P. Issues and Challenges Related to Privacy and security in healthcare Using IoT, Fog, and cloud computing. In *Advanced Healthcare Systems: Empowering Physicians with IoT-Enabled Technologies*; Scrivener Publishing LLC: Beverly, MA, USA, 2022; pp. 21–32.
32. Aziz, H.; Guled, A. Cloud Computing and Healthcare Services. 2016. Available online: http://hdl.handle.net/10576/4714 (accessed on 20 May 2023).
33. Chenthara, S.; Ahmed, K.; Wang, H.; Whittaker, F. Security and Privacy-Preserving Challenges of e-Health Solutions in Cloud Computing. *IEEE Access* **2019**, *7*, 74361–74382. [CrossRef]
34. Weir, C.R.; Hammond, K.W.; Embi, P.J.; Efthimiadis, E.N.; Thielke, S.M.; Hedeen, A.N. An exploration of the impact of computerized patient documentation on clinical collaboration. *Int. J. Med. Inform.* **2011**, *80*, e62–e71. [CrossRef]
35. Dixon, B.E.; Embi, P.J.; Haggstrom, D.A. Information technologies that facilitate care coordination: Provider and patient perspectives. *Transl. Behav. Med.* **2018**, *8*, 522–525. [CrossRef]

36. Bertagnolli, M.M.; Anderson, B.; Quina, A.; Piantadosi, S. The electronic health record as a clinical trials tool: Opportunities and challenges. *Clin. Trials* **2020**, *17*, 237–242. [CrossRef]
37. Liu, X.; Wang, Z.; Jin, C.; Li, F.; Li, G.A. Blockchain-Based Medical Data Sharing and Protection Scheme. *IEEE Access* **2019**, *7*, 118943–118953. [CrossRef]
38. Cordovano, G.; Shah, S.N. Requesting Medical Records, Health Data. 2020. Available online: https://journal.ahima.org/page/requesting-medical-records (accessed on 20 May 2023).
39. Gupta, D.; Malik, S.; Rana, A. Adopting Semantic Interoperability for Improved Healthcare (29 April 2022). Proceedings of the International Conference on Innovative Computing & Communication (ICICC) 2022. 2022. Available online: https://ssrn.com/abstract=4096399 (accessed on 20 May 2023).
40. Rajkumar, N.; Muzoora, M.; Thun, S. Dentistry and Interoperability. *J. Dent. Res.* **2022**, *101*, 1258–1262. [CrossRef]
41. Devadass, L.; Sekaran, S.S.; Thinakaran, R. Cloud Computing in Healthcare. *Int. J. Stud. Res. Technol. Manag.* **2017**, *5*, 25–31. [CrossRef]
42. Quatrani, T.; Evangelist, U.M.L. Introduction to the Unified Modeling Language. *IBM* **2003**, *6*, 3.
43. Górski, T. Profile for Messaging Patterns in Service-Oriented Architecture, Microservices, and Internet of Things. *Appl. Sci.* **2022**, *12*, 12790. [CrossRef]
44. Thramboulidis, K.; Christoulakis, F. UML4IoT—A UML-based approach to exploit IoT in cyber-physical manufacturing systems. *Comput. Ind.* **2016**, *82*, 259–272. [CrossRef]
45. Petrasch, R.J.; Petrasch, R.R. Data Integration and Interoperability: Towards a Model-Driven and Pattern-Oriented Approach. *Modelling* **2022**, *3*, 105–126. [CrossRef]
46. Pufahl, L.; Zerbato, F.; Weber, B.; Weber, I. BPMN in healthcare: Challenges and best practices. *Inf. Syst.* **2022**, *107*, 102013. [CrossRef]
47. Schmidt, M.-T.; Hutchison, B.; Lambros, P.; Phippen, R. The Enterprise Service Bus: Making service-oriented architecture real. *IBM Syst. J.* **2005**, *44*, 781–797. [CrossRef]
48. Niknejad, N.; Ismail, W.; Ghani, I.; Nazari, B.; Bahari, M.; Hussin, A.R.B.C. Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation. *Inf. Syst.* **2020**, *91*, 101491. [CrossRef]
49. Aziz, O.; Farooq, M.S.; Abid, A.; Saher, R.; Aslam, N. Research Trends in Enterprise Service Bus (ESB) Applications: A Systematic Mapping Study. *IEEE Access* **2020**, *8*, 31180–31197. [CrossRef]
50. Agbo, C.C.; Mahmoud, Q.H.; Eklund, J.M. Blockchain Technology in Healthcare: A Systematic Review. *Healthcare* **2019**, *7*, 56. [CrossRef]
51. Jin, H.; Luo, Y.; Li, P.; Mathew, J. A Review of Secure and Privacy-Preserving Medical Data Sharing. *IEEE Access* **2019**, *7*, 61656–61669. [CrossRef]
52. Kim, J.W.; Kim, S.J.; Cha, W.C.; Kim, T. A Blockchain-Applied Personal Health Record Application: Development and User Experience. *Appl. Sci.* **2022**, *12*, 1847. [CrossRef]
53. Abdellatif, A.A.; Al-Marridi, A.Z.; Mohamed, A.; Erbad, A.; Chiasserini, C.F.; Refaey, A. ssHealth: Toward Secure, Blockchain-Enabled Healthcare Systems. *IEEE Netw.* **2020**, *34*, 312–319. [CrossRef]
54. Guo, R.; Shi, H.; Zheng, D.; Jing, C.; Zhuang, C.; Wang, Z. Flexible and Efficient Blockchain-Based ABE Scheme with Multi-Authority for Medical on Demand in Telemedicine System. *IEEE Access* **2019**, *7*, 88012–88025. [CrossRef]
55. Xi, P.; Zhang, X.; Wang, L.; Liu, W.; Peng, S. A Review of Blockchain-Based Secure Sharing of Healthcare Data. *Appl. Sci.* **2022**, *12*, 7912. [CrossRef]
56. Qu, J. Blockchain in medical informatics. *J. Ind. Inf. Integr.* **2022**, *25*, 100258. [CrossRef]
57. Fiore, M.; Capodici, A.; Rucci, P.; Bianconi, A.; Longo, G.; Ricci, M.; Sanmarchi, F.; Golinelli, D. Blockchain for the Healthcare Supply Chain: A Systematic Literature Review. *Appl. Sci.* **2023**, *13*, 686. [CrossRef]
58. Alhaddadin, F.; Gutiérrez, J.A.; Liu, W. Privacy-Aware Cloud-Based Architecture for Sharing Healthcare Information. Ph.D. Thesis, Auckland University of Technology, Auckland, New Zealand, 2020.
59. Curtmola, R.; Garay, J.; Kamara, S.; Ostrovsky, R. Searchable symmetric encryption: Improved definitions and efficient constructions. In Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, VI, USA, 30 October–3 November 2006.
60. Boneh, D.; Crescenzo, G.D.; Ostrovsky, R.; Persiano, G. Public key encryption with keyword search. In Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, 11–15 May 2004.
61. Chang, Y.-C. Single database private information retrieval with logarithmic communication. In Proceedings of the 9th Australasian Conference on Information Security and Privacy, Sydney, Australia, 13–15 July 2004. [CrossRef]
62. Goh, E. Secure Indexes; IACR ePrint Cryptography Archive: 2003. Available online: http://crypto.stanford.edu/~eujin/papers/secureindex/ (accessed on 20 May 2023).
63. Eludiora, S.; Abiona, O.; Oluwatope, A.; Oluwaranti, A.; Onime, C.; Kehinde, L. A User Identity Management Protocol for Cloud Computing Paradigm. *Int. J. Commun. Netw. Syst. Sci.* **2011**, *4*, 152–163. [CrossRef]

64. Amazon, "aws,". 2019. Available online: https://aws.amazon.com (accessed on 5 April 2023).
65. Marwaha, J.S.; Landman, A.B.; Brat, G.A.; Dunn, T.; Gordon, W.J. Deploying digital health tools within large, complex health systems: Key considerations for adoption and implementation. *NPJ Digit. Med.* **2022**, *5*, 1–7. [CrossRef]