CC ASSIGNMENT - 2

Name: Anirudh Jakhotia
Roll No: S20190010007

Indian Institute of Information Technology, Sri City.

CLOUD COMPUTING

Report of two Research Papers:

Research Paper - 1: Summary

Link: https://ieeexplore.ieee.org/document/9381214
Name of the paper: A Scalable and Efficient Multi-Agent Architecture for Malware Protection in Data Sharing Over Mobile Cloud
Date of publication: 18 March 2021
Publisher: IEEE
Journal Name: IEEE Access (Volume: 9)
Authors: Hussian Qaisar, H.Almotiri, Al Ghamdi, Ali Nagra, Gulam Ali.

Motivation: Generally used Access control based on Attribute-Based Encryption (ABE) is a technique to ensure data shared security and may suffer from scalability and performance issues as they do not permit for addition or removal of computing nodes at run time. Another problem is existing approaches suffer from single-point-of-failure (SPoF).

Thus, to overcome this, a scalable multi-agent system architecture based on CP-ABE(Cypher-text Policy based on Attribute-Based Encryption) is proposed to ensure data sharing on public cloud storage and reliability in our proposed work. This methodology aims to protect the cloud from malware.

Objective: A Scalable and Efficient Multi-Agent Architecture is designed to preserve security, privacy, and malware screening using regress analysis by the graph embedding technique.

Major Contributions :

- Implement a CP-ABE mechanism that achieves better performance, is dynamically scalable, and takes less time to encrypt or decrypt a message and generate keys while achieving better security than KP-ABE.

- Provides security, privacy from malware by a proposed malware protection approach.

- We solve SPoF issue by running multiple agents of the same authority on different hosts.

- Introduce regress analysis which provides better malware protection.

- Compared Xiao, the Trustav, and Dey techniques for malware and encryption, the results advance in performance, scalability, and security.

This includes what are all the techniques the author performed and which were different from other general-based approaches.

Description of the proposed approach:
The authors approach this through an agent-based MA-CP-ABE architecture as system architecture. We define agents for the main entities (CA, AA, and CS). We also define some additional agents in order to achieve our stated goals such as scalability, availability, and efficiency. Here are some important agents in architecture.

- **Attribute Authority Agent (AA):** It maintains the same set of attributes and hence, the same encryption/decryption credentials.

- **High-Level Certificate Agent (HLCA)**: It distributes the incoming request for certificates from end-users in the round-robin fashion to the available CAs to improve system utilization.

- **Daemon Agent (DA):** The DA is kept in a separate cloud node. This agent works as an eye-keeper and is responsible for keeping track of all the agents of CAs.

- **Pair of Client-Side and Server-Side Agents:** A pair of Client-Side Agents (CSA) and Server-Side Agents (SSA) has been introduced on the client and cloud server-side respectively for each end-user connection.

The main entities of the proposed MA-CP-ABE model are as follows:

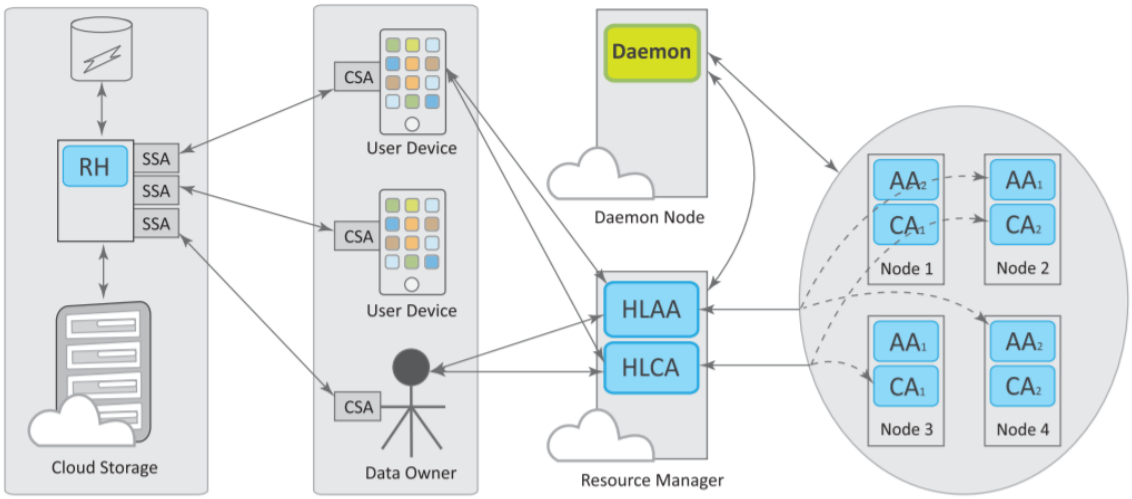

FIGURE 1. Architecture of the proposed system.

Method to the approach:

The Certificate Authority (CA) entity is responsible for verifying the user and issuing a certificate. Then the users generate partial credentials for encryption and decryption based on the user's provided attributes provided by Attribute Authority (AA). Now to access the shared data from anywhere, anytime we use Cloud Server (CS) which is responsible for storing the organizations' data. To ensure a scalable, efficient, and reliable

CP-ABE-based access control for MCC, we define agents for each entity and provide the respective services.

Discussion of the experimental testbed and results:
This section is divided into 3 components namely Implementation, Validation of the approach, and finally the results.

Implementation :

- In the proposed agent-based system, the first step is verifying the user's legitimacy. If it is successfully verified, CA issues a certificate and encrypts the certificate using its private key.

- The user then requests each AA for the corresponding partial credentials. This mechanism allows distributing the user requests to multiple hosts, consequently improving system performance.

- All this is achieved in four steps in our proposed system using system initialization, key generation, encryption, and decryption.

Validation of the approach:
In this, when a new app is considered for evaluation, it uses that knowledge base to check the current app's similarity under evaluation with the existing malicious patterns based on a graph similarity measure. In case the graph similarity is more than the threshold, it considers it as similar and categorizes it into malicious. It uses a graph convolution-based approach for malware. Our approach uses the preexisting knowledge base of malicious app patterns.

Results:
Evaluation parameters like precision, accuracy, recall and F-measure are used for comparison.

Approach	Accuracy	Precision	Recall	F-Measure
MA-CP-ABE	98	97	96	96
Xiao2017	95	93	92	91
Trustav2020	96	94	88	90
Dey2019	94	92	80	89

We see that while comparing the proposed with already existing approaches, our model performs better in all terms

Strengths :

- This paper's proposed approach is MA-CP-ABE which is a very efficient algorithm that showed significant results.
- The internal architecture is designed such that it can handle system availability in case of failure, have backup Daemon Nodes to create copies, handle the situation if the host machine fails, etc.
- The results were compared with various malware detection techniques in the cloud and it outperformed them.
- It overcomes the deficiencies of scalability and efficiency.
- Improve system performance and avoid SPoF.

Limitations:

- More complex to implement.
- Uses too many resources as having intermediary cloud agents in cloud nodes.
- It does not use feature extraction for the selection of relevant attributes instead considers all of them.

- It suffers from user collusion problems where more than one user can pool their partial private credentials to obtain the full credentials and higher illegal privileges.

- Due to the lack of confidentiality in the proposed architecture, this paper can be easily extended, and blockchain technology can easily be integrated to ensure data confidentiality.

- The cloud server is managed by a third-party service provider and is always assumed to be online. If the cloud service provider goes offline, then there would be a lot of trouble for the daemon and backup nodes to handle the situation.

- While starting a backup agent from the backup state, the system will not provide its services. So during this period, the system will be unavailable to process incoming user requests.

Research Paper - 2: Summary

Link: https://ieeexplore.ieee.org/document/9113446
Name of the paper: Disrupting Healthcare Silos: Addressing Data Volume, Velocity, and Variety with a Cloud-Native Healthcare Data Ingestion Service
Date of publication: 10 June 2020
Publisher: IEEE
Journal Name: IEEE Journal of Biomedical and Health Informatics
Volume: 24
Authors: Rohit Ranchal, Paul Bastide, Xu Wang, Aris Gkoulalas-Divanis, Maneesh Mehra, Senthil Bakthavachalam, Hui Lei, and Ajay Mohindra.

Motivation:
The main motivation behind this is the increased volume, velocity, and variety of healthcare data, and the need to facilitate data correlation and large-scale analysis.

This paper highlights the need for providing healthcare data acquisition using cloud infrastructures and presents the challenges, requirements, use-cases, and best practices for building a state-of-the-art healthcare data ingestion service on the cloud.

Objective:
The objective behind a cloud-native Healthcare Data Ingestion (HDI) service is to provide organizations a standard, interoperable, secure, compliant solution to acquire, store, consolidate, and use healthcare data in the cloud.

Major Contributions :

- The introduction of design patterns for building a cloud service with a standard and interoperable mechanism for healthcare data ingestion and storage.

- The data isolation presents a major challenge for healthcare entities and prevents them from using the latest IT innovations, such as the data processing and analytics capabilities offered by cloud computing, which can help improve care while significantly reducing costs

- Laying out the major functional and nonfunctional requirements necessary for operational and regulatory controls.

- Demonstrating the practicality of the HDI service in a realistic healthcare scenario.

- The paper focuses on identifying the requirements and presents a design for building a state-of-the-art cloud-native healthcare data ingestion service irrespective of data volume, variety, or velocity.

This includes what are all the techniques the author performed and which were different from other general-based approaches.

Description of the proposed approach:

- The author approaches this through a proposed Healthcare Data Ingestion (HDI) service that provides a standard interface and a repository to store various forms of healthcare data in its raw form in the cloud.

- This can be used in addition to other services in the cloud ecosystem, such as services for monitoring incoming data, triggering alerts and processing actions, running analytics pipelines to predict events.

Design of the proposed solution :

We discuss the design considerations of the HDI service and present a generic architecture. The three main functions include:

1) **Data acquisition:** The clients interact with this layer to upload and download various types of healthcare data in the cloud. These interfaces are responsible for authenticating the client applications, authorizing their request.

2) **Data Parsing & Validation:** We must understand various types of incoming healthcare data, parse data formats, and then validate them.

3) **Data Storage & Auditing:** This layer captures metadata by combining the data attributes extracted. Finally, it stores the metadata, audit log, and raw data content in the designated storage subsystem.

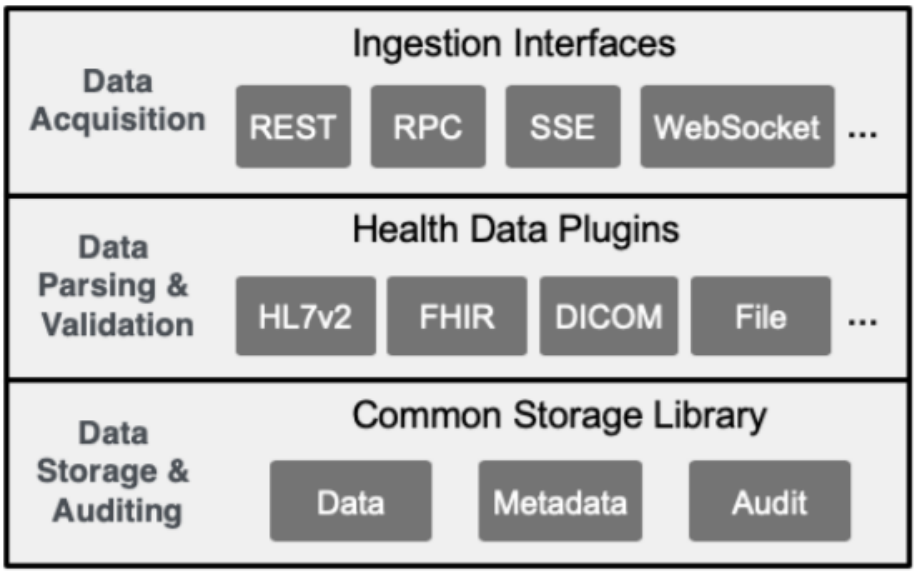

Fig. 1: Logical view of HDI service design

Method to the approach:

The HDI service can be used to develop a cloud solution for patient care managers to monitor patients with chronic diseases. Using the HDI service, this solution retrieves patient data and calculates key metrics from the data to study patient behavior.

The patient care managers continuously keep track of their assigned patient accounts by evaluating the derived behaviors of the patients. A cloud solution is developed that runs stream processing jobs to process the ingested data as it becomes available.

It generates real-time alerts for medical professionals. secondary use-cases can be developed to study population health and derive insights using a combination of de-identification and analytics services.

Discussion of the experimental testbed and results:
This section is divided into 3 components namely Implementation, Validation of the approach, and finally the results.We present a realistic healthcare scenario and discuss the implementation in its context.

Implementation :

- Big data services, like HDI, depends on solid implementations of nonfunctional requirements to support the volume, velocity, variety, and veracity of healthcare data.

- The HDI service must ensure auditing, logging, disaster recovery, high availability, security, and performance. The FHIR format is a good fit for data storage because it uses standard web and mobile-friendly protocols, such as REST APIs and JSON data exchange format.

- Each patient's inhalation event generates data that is recorded on their mobile device. The HDI service implements both storage-level encryption and secure endpoints consumer-to-service as the data ingress and egress happens.

- The HDI service must be validated to ensure that healthcare data is treated properly following regulatory requirements.

Discussion of Validation:
The HDI service plays a critical role by providing an accurate and efficient data acquisition mechanism in building an end-to-end healthcare solution.

Highlighted the challenges of large-scale healthcare data acquisition from multiple sources and provide guidance on leveraging the cloud for building an HDI service for regulated environments.

The paper provided a reference implementation of the HDI service without making it specific to a healthcare solution, cloud provider, or technology platform.

It also realizes multiple healthcare customer's use-cases for developing complex solutions on the cloud.

Results:
In our solution, data is buffered into optimal data blocks, such that a bulk of data is written to the data storage block balancing storage, performance, analysis.

Strengths:

- This paper is based on years of experience building such services for major customers in the healthcare industry.

- It focuses on identifying the requirements and presents a design for building a state-of-the-art cloud-native healthcare data ingestion service irrespective of data volume, variety, or velocity.

- Continuous scanning occurs within the cloud at the system and data levels.

- This service is a prerequisite for any healthcare cloud migration and key to promoting interoperability, sharing, and integration of the healthcare data leading to new advancements.

- Healthcare clients have strong requirements for malware and virus scanning for every data segment that is uploaded to the cloud.

- Small messages are scanned at entry into the platform using ICAP, while large files are scanned in a staging area before storage.

Limitations:

- Many secondary use-cases should be considered and are not considered in this paper where de-identified healthcare data from multiple sources are securely linked into a cloud-based global data repository.

- Migration of healthcare workloads to the cloud is not straightforward due to the lack of healthcare data standards, heterogeneity and sensitive nature of healthcare data, and many regulations that govern its usage.

- Healthcare organizations have a major concern about the availability and sharing of information across the continuum of medical care due to the healthcare data being typically dispersed.

- For instance, healthcare data can be searched and queried using metadata. For this, the metadata needs to be indexed and, if the queries need to be flexible which is not present in most cases.

- Instances must not be running in the same rack and on the same physical machine, to avoid service failures when a rack or a physical instance or a pod dies.

- As accurate data linkage is essential for correctly capturing records, the lack of standard terminology or use of different coding such as ICD, LOINC, SNOMED in different systems delays data linkage.