CAPITAL UNIVERSITY OF SCIENCE AND
TECHNOLOGY, ISLAMABAD

Application of Attribute-Based Encryption in Fog Infrastructure for Securing Health Related Data

by

Sahibzadi Annum Shaheen

A thesis submitted in partial fulfillment for the
degree of Master of Science

in the

Faculty of Computing
Department of Computer Science

2022

My work is devoted to My Parents, My Teachers, My Family, and My Friends. I have a special feeling of gratitude for My Parents and siblings. Special thanks to my supervisor whose support make me able to reach this milestone.

Certificate of Approval

Application of Attribute-Based Encryption In Fog Infrastructure For Securing Health Related Data

by

Sahibzadi Annum Shaheen

(MCS201013)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Muhammad Wasif Nisar	CUI, Wah Cantt
(b)	Internal Examiner	Dr. Qamar Mahmood	CUST, Islamabad
(c)	Supervisor	Dr. Nayyer Masood	CUST, Islamabad

Dr. Nayyer Masood
Thesis Supervisor
September, 2022

Dr. Abdul Basit Siddiqui
Head
Dept. of Computer Science
September, 2022

Dr. M. Abdul Qadir
Dean
Faculty of Computing
September, 2022

Author's Declaration

I, **Sahibzadi Annum Shaheen** hereby state that my MS thesis titled "**Application of Attribute-Based Encryption In Fog Infrastructure For Securing Health Related Data**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MS Degree.

(Sahibzadi Annum Shaheen)
Registration No: MCS201013

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**Application of Attribute-Based Encryption In Fog Infrastructure For Securing Health Related Data**" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS Degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Sahibzadi Annum Shaheen)
Registration No: MCS201013

Acknowledgement

I praise and worship my Allah who is all in all. He is perfect source of strength in my life. I want to show gratitude to my family including parents and siblings for their support and love. I would not be able to achieve anything without my family. I am thankful to my teachers. They give me a lot of knowledge. Specially, I want to thank my supervisor Dr. Nayyer Masood for this work. He has guided me in a very good way. I have learned a lot from him.

(Sahibzadi Annum Shaheen)

Abstract

Remote health monitoring is gaining more interest with the availability of improved health related devices, sensors and better connectivity services. The cloud related and the mobile technologies have made it easier to monitor the patient's health conditions by health care teams such as doctors, nurses and specialists. SAFE-RURAL HEALTH (SAFE-RH) is a European Union funded program that intends to create a digital healthcare community. In order to help patients deal with fewer challenges and expenses, such as transportation problems when travelling for specialty care, this programme aims to deploy remote medical monitoring within Pakistan's rural areas. Data security is becoming a bigger more critical challenge as the canvas of computing is increasing. From single desktop PC to client-server to internet to cloud computing, the paradigm is getting wider and wider. Every next step poses new challenges in the data security and specially in the healthcare domain. The presence of data on the cloud/fog units raises many security and privacy concerns for individuals and healthcare providers. Attribute-based encryption (ABE) technology is critical for achieving fine granularity and scalability in access control systems. Transmitting unprotected patient's data over the Internet carries the risk that someone can monitor and alter this data, whether accidentally or intentionally. However, attribute-based encryption systems still need to solve difficulties with data protection and user privacy. It also presents difficulties like high administrative costs and end-user attribute privacy. In order to achieve data protection and user privacy for health-related data in the SAFE-RH Health Program, we adopted ciphertext policy attribute-based encryption (CP-ABE), a sort of attribute-based encryption. Moreover, we have also defined attributes of different users as well as access policies for accessing the data in secured way from the fog environment. Furthermore, we have provided the formal security analysis and proof of concept that our proposed scheme is light-weight yet effective against Chosen Ciphertext attack. Our future work includes but not limited to the key's generation from multiple key authority centre, increase in speed of encryption, data retrieval from multiple facilities and multiple fog nodes.

Contents

List of Figures

List of Tables

Abbreviations

ABE	Attribute Based Encryption
CPABE	Ciphertext Policy Attribute Based Encryption
CSP	Cloud Service Provider
CT	Cipher Text
CCA	Chosen CipherText Attack
DU	Data User
DO	Data Owner
ETSI	European Telecommunications Standards Institute
EU	European Union
ECC	Elliptic Curve Cryptography
FSP	Fog Service Provider
FN	Fog Node
IaaS	Infrastructure as a Service
KPABE	Key Policy Attribute Based Encryption
KeyGen	Key Generation
MPK	Master Public Key
MSK	Master Secret Key
QoS	Quality of Service
SaaS	Software as a Service
SAFE-RH	Safe Rural Health
SK	Secret Key
SSD	Solid State Drive

Symbols

λ	Setup Input Parameters
MPK	Master Public Key
SK	Secret Key
L	Attribute List
M	Message/Data
CT	Cipher Text
G_1, G_2	Bilinear Map
g_1, g_2	Generators of Bilinear Map
α, β	Random Integers for the Generator
Z_p	Prime Order Integer
P	Prime Order

Chapter 1

Introduction

1.1 Introduction

Data security is becoming a bigger more critical challenge as the canvas of computing is increasing. From single desktop PC to client-server to internet to cloud computing, the paradigm is getting wider and wider. Every next step poses new challenges in the data security and specially in the healthcare domain. The centralization of data on the cloud/fog servers raises many security and privacy concerns for individuals and healthcare providers.

In order to achieve data security and user privacy for health-related data in the SAFE-RH Health Program, we adopted ciphertext policy attribute-based encryption (CP-ABE), a sort of attribute-based encryption. Additionally, we have specified the qualities of different users as well as accessibility guidelines for securing data access from the fog computing environment.

We have divided this chapter into three sections: Section 1.1: Describes encryption, Section 1.2: Describes attribute-based encryption, Section 1.3: Describes cloud computing, Section 1.4: Describes edge/fog computing, Section 1.5: Describes SAFE-RH project, Section 1.6: Describes rationale and significance of research, Section 1.7: Describes problem statement, Section 1.8: Is about research

questions, Section 1.9: Is about our contributions to the research, Section 1.10: Is about thesis organization.

1.2 Encryption

Digital data can be protected via encryption [1], which encrypts it using any or more mathematical methods and requires a password or "key" to decipher. Information is translated during the encryption process using a technique that renders the original data unintelligible. For instance, the procedure can change a plaintext original text into ciphertext, a different form. When a person with permission wishes to access the data, they can use a secret key to decrypt it. Thus, the original data can be accessed by the authorised user by converting ciphertext back to plaintext.

Sensitive information should always be encrypted to prevent hackers from accessing it. For instance, websites that send bank account and credit card details should always encrypt sensitive data to guard against fraud and identity theft. Cryptography is the study and use of encryption in mathematics.

Symmetrical and asymmetrical encryption are the two most popular encryption techniques [2]. Whether or not same key to encrypt the data and decryption is indicated by the names:

- Symmetric encryption keys: Private key encryption is another name for symmetric cryptographic keys. Due to the fact that the key required for encoding and decoding is the same, it functions best for lone users and closed situations. If not, the key must be given to the receiver.
- Asymmetric encryption keys: These employ two distinct keys—public and private—that are mathematically connected to one another. Since the keys are merely pairs of large numbers which aren't precisely the same, the word "asymmetric" relates to this. While sharing or making public key publicly available or only allowed recipients, the owner takes the secret key a secret.

Depending on whether they are made for data in transit or data at rest, data encryption solutions like application and cloud encrypting data are frequently categorised:

1.2.1 Data Encryption in Transit

When information is travelling between devices over a public network or the internet, it is referred to as being in transit. Data is more vulnerable during transfer because to the necessity of decryption prior transfer and the flaws in the transfer technique itself. End-to-end encryption, which protects data privacy even if it is intercepted, encrypts data as it is being transferred [3].

1.2.2 Data Encryption at Rest

When data is stored on a storage medium but is not being used or transported, it is said to be at rest. Since, device security mechanisms restrict access, data at rest is frequently less insecure than when it is in transit, but it isn't immune. It is also frequently more valuable in terms of information, making it a more desirable target for thieves.

1.3 Attribute Based Encryption

For access control systems to achieve fine granularity and scalability, attribute-based cryptography (ABE) [4] technology is essential. As seen in the ABE-enabled data access, the flexible attributes are contained in the ciphertext as well as the Data Owner (DO) is not required to be aware of the identities of individual Data Users (DU) prior to encryption. Whenever a new DU enters the system, DOs are unchanged and are not required to take any action. As a result, the access control system with ABE support is flexible and scalable. Sahai and Waters [5] established the ABE concept. ABE is separated into two categories and has attracted a lot

of academic attention as an insight into future primitive. The first is known as ciphertext-policy ABE, or CP-ABE, and the latter is known as key-policy ABE.

In CP-ABE, an attribute list and a user's attribute secret key are linked, and an access policy that spans the system's attribute array is described by a ciphertext. Therefore, a user could only read ciphertext if the attribute list complies with the access policy of the ciphertext. In KP-ABE, an attribute list is created as ciphertext and an access policy specified well over system's attribute universe is used to encode a user's attribute secret key. Only when the associated attribute list matches the access policy linked to the user's secret key attribute will a user be able to decode a ciphertext. Because ABE-enabled access management on outsourced cloud services is so expressive and scalable, a lot of academic attention has been paid to it. On the other side, CP-ABE has gotten a lot of attention than KP-ABE. This is due to the fact that with CP-ABE, it is up to the data owner to decide on the access policy. Based on the demands of data access in different application contexts, CP-ABE systems are further divided into numerous types.

For the remote patient monitoring domain, we suggest an encryption method based on ciphertext-policy attributes. There are numerous improved versions of CP-ABE. The enhanced CP-ABE additionally realizes other cryptographically useful components. These characteristics can enhance computation effectiveness, attribute keys, access control, and attribute authorities. Based on their characteristics, improved CP-ABE schemes are divided into eight types:

1. CP-ABE that can be revoked. The revocable CP-ABE provides revocation functionality. Revocation procedures are divided into two categories based on their graininess: Revocation of both users and attributes. Based on the effects on non-revoked users, revocation methods are divided into implicit revocation and direct revocation.

2. CP-ABE that is accountable. The capability of accountability is provided in accountable CP-ABE. User tracking and attribute authority accountability

both are important aspects of responsible CP-ABE. Depending on the situation, accountability techniques are categorized as either "white" or "black-box".

3. CP-ABE with policy concealing. Access policy privacy is better protected in policy-hiding CP-ABE setups.

4. CP-ABE with Policy Updating is the fourth option. In basic CP-ABE, it is not feasible to modify the access control policies of ciphertext. CP-ABE with policies update can then be used to modify the access control policies in a complex ciphertext for emergency access control.

5. CP-ABE with several authorities. This kind of CP-ABE structure can be used to achieve distributed access privileges. Based on whether or not a central authority exists, multi-authority CP-ABE schemes are divided into centralized and decentralized multi-authority CP-ABE structures.

6. CP-ABE with a hierarchical structure. In hierarchical CP-ABE architectures, the allocation of privileged access is categorized hierarchically.

7. CP-ABE (online/offline). By enabling offline encrypted communications or key generation, the online/offline CP-ABE is intended to reduce the computational strain on system users and attribute authorities.

8. Outsourced CP-ABE. CP-ABE is outsourced. In order to assist data users (otherwise, data owners and the authority) with limited computation resources, the outsourcing CP-ABE is suggested to outsource laborious computations in decryption (in both, encrypting and key generation) to third-party servers.

1.4 Cloud Computing

Cloud computing [7] includes both the software and hardware that underpin the data centers that provide these services as well as the applications which are

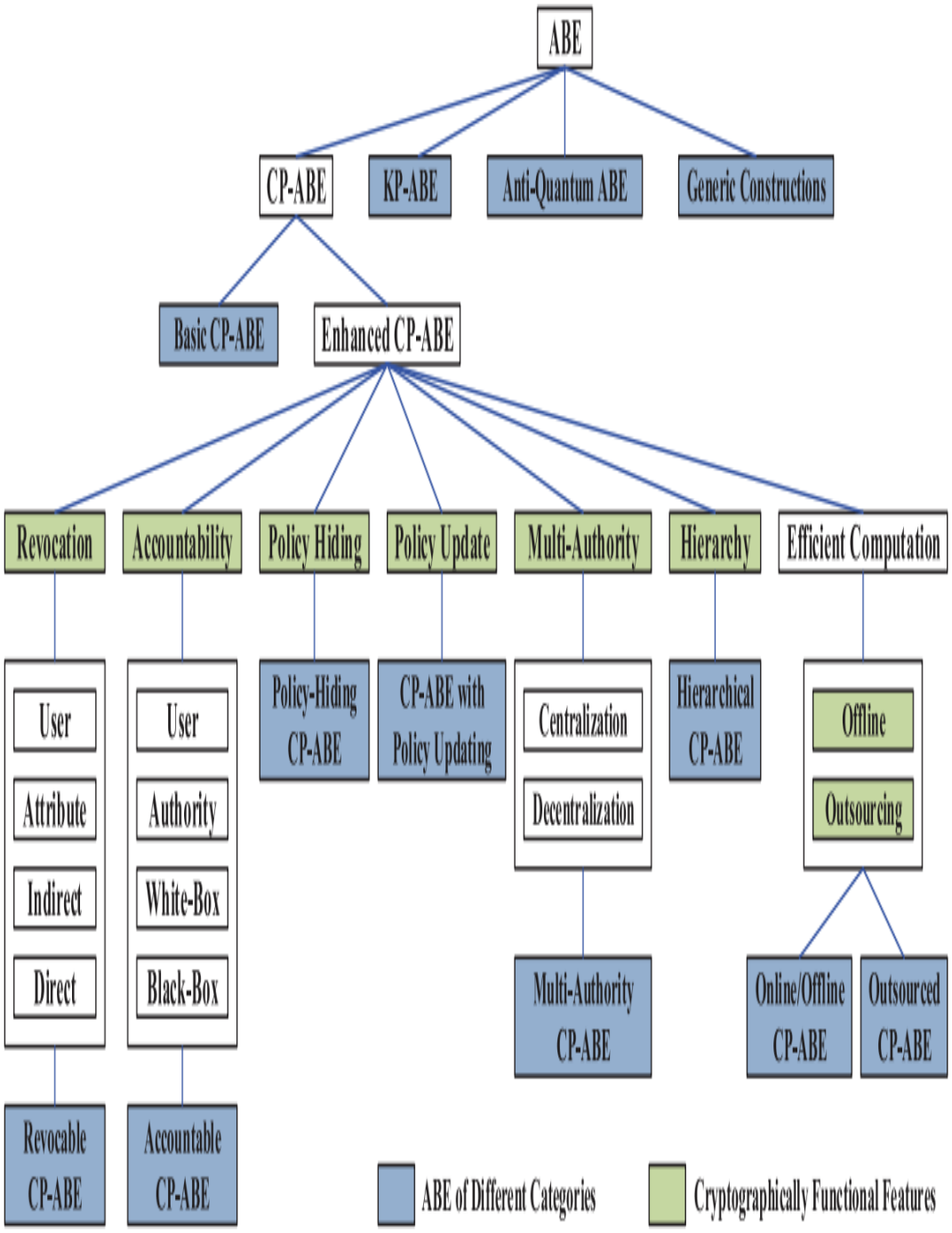


FIGURE 1.1: Overall Taxonomy of ABE [6]

offered as services via the Internet. For a very long time, the phrase "software as a service" (SaaS) has been used to refer to services (SaaS). Some businesses refer to their solutions as IaaS (Infrastructure as a Service) and PaaS (Platform as a

Service). The term "Cloud computing," which also originates from of the powerful computational community, relates the protocols that permit shared processing and storage across great distances; however, those protocols aren't generally used. The general public can use a public cloud on the payment basis, and utility computing is the service that is being promoted. When internal data centers of a company or other organization are sizable enough to reap the rewards of cloud computing, this is considered to as a cloud infrastructure.

SaaS and utilities computing are therefore included in the term "cloud computing," while small to medium-sized data centers are not, even if they operate using virtualization. Users and providers of SaaS as well as clients and providers of utility computing are both possible. The limitations of private clouds in virtualization are a hot topic of discussion online. Excluding exceptionally large data centers with hundreds of thousands of computers, such as those run by Google or Microsoft, the majority of data center only benefit from a small percentage of the potential benefits of the public cloud. Since the idea of cloud computing encompasses traditional data center, this can result in exaggerated claims for smaller, so-called private clouds. [8]

1.5 Edge and Fog Computing

The term "edge computing" [9] is a comparatively recent one in the field of computing. It is distinguished by quick processing and application reaction times and puts cloud - based services that utilities closer to the end users. Real-time traffic monitoring, virtual reality, and surveillance are a few internet-enabled applications that require quick processing and reaction times. These applications often run on the resource-constrained mobile devices of end users, with cloud servers handling the processing and core service. Edge computing addresses the above-mentioned application requires by moving processing to a network's edge, which eliminates the mobility-related issues and severe latency that come with using cloud services on mobile devices. Three Edge computing techniques that can assist with

cloud computing difficulties include cloud resources, fog nodes, and mobile edge computing.

Mobile Edge Computer is a term coined by the European Telecommunications Standards Institute (ETSI) to describe how mobile users consume computing resources from a base station [10]. The fog computing idea was put forth by Cisco and enables programs run directly now at edge networks using billions of intelligently linked devices. Cloudlets are a concept that Satyanarayanan et al. [11] proposed to overcome the latency issue with cloud access by employing local computer resources. On the other side, mobile edge computing puts storage, application services, and off-load processing closer to end users. Edge computing may have the ability to offer mobility, location awareness, super bandwidth, and closeness to the user. These features make edge computing suitable for a range of potential applications, such as data analytics, industrial automation, vr technology, vehicular monitoring, smart homes, and smart maritime monitoring. Edge devices including router, access points, and base stations host a variety of services (e.g., QoS, VPN, Voice over IP, etc). These Edge devices act as a bridge between the cloud and modern mobile devices.

In a decentralised computing environment known as fog computing, data, compute, storage, and applications are distributed between the data source and the cloud. Fog computing brings the benefits and power of the cloud closer to where data is created and used. Instead of creating in-cloud channels for usage and storage, users can aggregate traffic at access points like routers by placing these closer to the devices. As a result, less data must be transported from data centres across long distances and through various cloud routes, which lowers the overall bandwidth requirement.

The distinction between edge computing and cloud computing is that the former reduces latency by moving resources directly to end users [12]. By distributing services and resources within the Edge network, edge computing reduces the burden on a cloud. Moreover, Fog computing allows users to submit data to

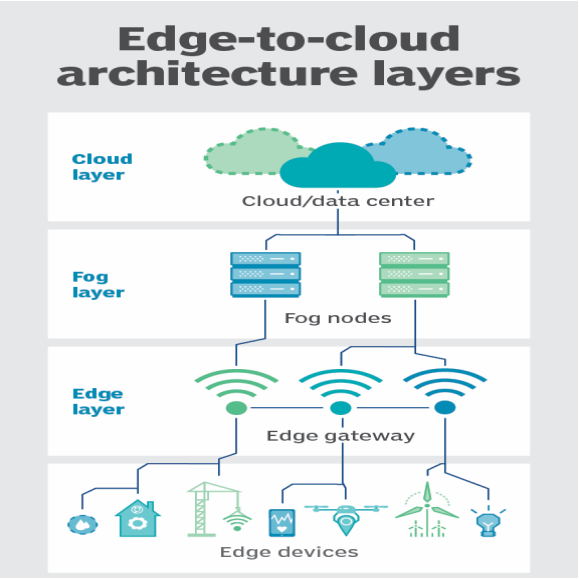


FIGURE 1.2: Architecture layers from edge to cloud [13]

strategic compiling and distribution rules aimed to increase efficiency and lower costs because less data requires immediate cloud storage.

On the other side, edge computing works in conjunction with cloud computing to enhance end-user service for time-sensitive tasks. Like cloud providers, edge service providers give end users access to application, data processing, and storage resources. These two new computer models are comparable on a fundamental level, but they also have some important differences. The main difference among both Edge computing and cloud-based computing is the position of the servers. While cloud - based services are found on the Internet, edge computing solutions are found on the edge network.

1.6 Safe-RH Project

The SAFE-RURAL HEALTH [14] program intends to create a digital healthcare community that bridges the gap by providing digital health solutions. The goal of SAFE-RURAL HEALTH in Pakistan is to promote the delivery of healthcare services to reduce maternal and child mortality, address maternity-related issues in distant locations and monitor the elderly. Patients may experience less hassles

and burdens as a result of the using remote health monitoring in Pakistan's rural areas for the delivery and assistance of healthcare services, such as transportation issues when travelling for specialty care. Remote health monitoring will enhance communications, monitoring, and timeliness within the healthcare system. The overall goal of establishing a Smart Healthcare system in Pakistan is to aid in the delivery of health care services. A high mortality rate in newborns and women is caused by a shortage of professional opinion and health services, and monitoring of elderly patients is also necessary to provide appropriate health care. As a result, a system must be developed that can give digital health services to patients in Pakistan's distant and frequently difficult-to-reach places.

- Providing patients in rural locations with access to competent doctors.
- In the case of chronic conditions, reducing repeated visits and re-admissions.
- Reduce the number of people who die as a result of a lack of or delay in receiving medical treatment.
- Keeping consultation costs and time to a minimum
- Providing patients with convenience while seeking medical advice

The usage of smart technologies in health in the target country, Pakistan, is far less widespread than in the EU. Emerging technologies, as well as technical expertise, will greatly improve Pakistan's health sector, ensuring high productivity on a worldwide scale. The following are the partner country Pakistan's most common healthcare challenges:

- To lower the mortality rate of mothers and children
- To handle maternity-related issues in rural locations promptly
- To keep track of elderly persons
- Data Generation by Body Sensors

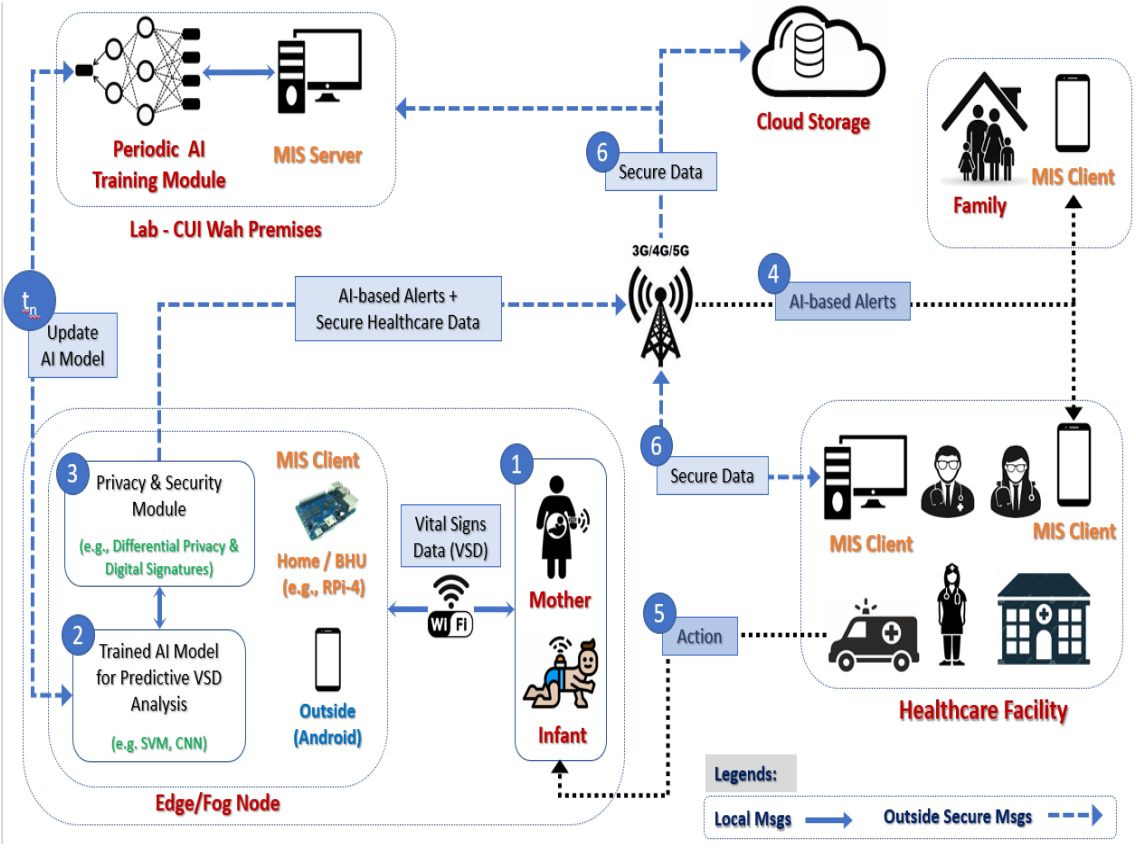


FIGURE 1.3: Overall Architecture of SAFE-RH Project

  - For example, wrist-watch sends data to Sender Edge Device (SED)
  - SED sends data/alerts to Receiver End Devices (RED) and data to Cloud

- Data Processing at SED
  - Machine Learning at the Edge/Fog node (e.g. SVM, CNN, ...)
  - (type-01) SED At Home

- Data Privacy/Security from SED to RED
  - At SED and RED (techniques like Differential Privacy and Digital Signatures)
  - ML-based Alerts to RED devices of Hospital, Doctor, Family and Caretakers

- Data Storage

– From SED to RED devices of Hospital and/or 3rd party cloud servers

1.7 Rationale and Significance of Research Topic

Ciphertext Attribute Based Encryption (CP-ABE) can be used by an edge computing network to implement access control and protect the security and privacy of data. Because of the increased security of the CP-ABE scheme's mechanism, this observation leads us to suggest it. Additionally, the main advantages of using CP-ABE are that:

- The data holder shall secure the data in line with the attributes; the number or identity of users need not be taken into consideration.
- Users can decrypt ciphertext if they match the attribute requirements.
- The user collusion attacks can be prevented by the ABE key's connection to a string of integers.
- Encourage adaptable, fine-grained access control.

1.8 Problem Statement

The SAFE-RH is a remote healthcare monitoring program for Pakistan's rural areas, to deliver and assist for provision of healthcare services. In SAFE-RH patient's critical data is generated through wearable sensors, and is uploaded to Fog nodes. Data users, by using their private keys to decrypt this confidential data, can access this data directly from fog nodes. Fog servers have limited computational capabilities as compared to cloud servers. The centralization of data on the fog servers raises many security and privacy concerns for individuals and healthcare providers. This type of data can easily be leaked due to weak encryption algorithms and policies defined for accessing it. For the purpose of maintaining

the security and integrity of individualised data over fog nodes, this thesis suggests an attribute-based encryption approach.

1.9 Research Questions

The above problem statement raises following research questions: -

1. How can we build a light weight yet effective encryption system for securing data in SAFE-RH project?
2. What CP-ABE policies can be made to secure data at fog nodes in the context of SAFE-RH?

1.10 Our Contributions

Following are our contributions to the literature related to securing health data:

1. We have defined policies for accessing patient's data from fog
2. We have defined attributes for accessing patient's data from fog
3. Defined attributes and policies are implemented at Fog node which avoid transmission delay

1.11 Thesis Organization

The format of this thesis is as follows:

- Chapter 2 is about literature review.
- Chapter 3 is about proposed methodology.

- Chapter 4 is about base paper, policies, implementation and results.
- Chapter 5 is about test case studies
- Chapter 6 is about conclusion and future work.

Chapter 2

Literature Review

2.1 Introduction

To provide a solid solution of the problem statement discussed in chapter 1 we need to answer the following questions during our literature review. 1: Is attribute based encryption adopted for securing health related data? 2: What are the strength of the different types/ flavors of attribute based encryption adopted by different authors? 3: What are the drawbacks / weakness in current techniques?

This section provides a comprehensive literature review of the research conduct in this area and provides critical reviews of all the proposed approaches.

We have divided this chapter into three sections: Section 2.1: Shows the related work, Section 2.2: Shows the analysis of related work, Section 2.3: Shows the gap in the literature from the related work.

2.2 Related Work

Numerous research articles dealing with data security and privacy issues in the e-health field have been published. The following research has been done by researchers: Using attribute-based encryption and data deduplication, the authors

Kumar et.al [15] suggested a solution to the problems with storage and privacy in patient care for such cloud computing system. They asserted that the best way to prevent privacy issues is to use their suggested system. The integration of cloud internet and mobile computing for healthcare sector is the topic of Paper by Sreenivasa et.al [16]. When using mobile computing, users face a number of difficulties, including the secrecy of data transferred to the cloud, the accuracy of data that has been saved, WAN latency delays, and resource limitations on mobile devices. To solve the aforementioned issues, writers have suggested a Cloudlet-Based eHealth Big Data System with Outsourced Decryption (CBe-BDS-OD). They safely contracted out the server's decryption function to them. Security testing reveals that their plan is safe. Additionally, their plan can be implemented on mobile devices with limited resources because theory analysis and empirical simulation show a significant increase in computation efficiency of 99%.

In a publication of Shynu et.al [17], a novel patient-centric paradigm and a set of procedures for controlling data access to patient health records stored on sub servers were proposed. The authors partition the users of the PHR system into different security domains and use attribute-based encryption (ABE) techniques to safeguard each patient's PHR data centre, which greatly simplifies key management for both owners and users. This method offers a high level of patient privacy through the simultaneous generation of multi-authority ABE and CC-MAABE.

Kibiwott et.al [18] makes a fresh suggestion on how to deal with the secure and fine-grained access control problems for cloud-stored e-health data. For E-health clouds, they specified a secured access control system. The E-health clouds' effective design is presented first, and afterwards the access control model is explained. The cloud provider and the data user must first perform three-factor authentication. Additionally, the suggested method uses attribute-based searchable encryption with a trapdoor function to effectively block unauthorized access to cloud data. Their experiment, which makes use of charm crypto, yields result that are noticeably superior to those of the already used methods.

Zhang et.al [19] describes a modified form of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) dubbed CESCR, a CP-ABE enabling quick and effective attribute/user revocation while sharing health data in collaborative e-health systems. This is due to the fact that the traditional CP-ABE technique cannot be directly adopted in a collaborative eHealth environment. For starters, because it is built on a monotonic access structure, its expressiveness is constrained. Second, there is no mechanism for revocation of attributes or users. Third, as the number of features in the ciphertext increases, so does the computational cost on the data controller and users. The CESCR scheme is unbounded, i.e., it does not bind the dimensions of a attribute universe to a security parameter, and it securely outsources the computationally demanding attribute procedures of both data encryption and decryption without requiring a dummy attribute. It is based primarily on expressive as well as non-restrictive ordered binary decision diagram (OBDD) access framework. They demonstrated the CESCR system's expressiveness and efficacy.

The claim of the Ciphertext-Policy Attribute-Based Signcryption (CP-ABSC) system (Liu et al. 2015) is that it provides secrecy against specific ciphertext assaults in a selective-predicate architecture. However, the authors Prakash et.al [20] circumvent this claim. The authors of this work also suggest a CP-ABSC technique that can simultaneously offer fine-grained user access, secrecy, authenticity, signcryptor privacy, and public verifiability for cloud-based PHR sharing systems. Their system achieves security in the conventional paradigm by using expressive monotonic binary functions as predicates for signing and encryption. When compared to other schemes in the field, their construction has a small ciphertext size and uses fewer pairing computations.

A certificate-based progressive proxy re-encryption system (CB-PReS) was proposed in paper by Jianfei et.al [21] for the sharing of e-health data in fog computing. The updated, deleted, and inserted file actions are all improved by the suggested technique. The iFogSim simulator is used to test the proposed method. The iFogSim simulator makes it easier to create scenarios of fog and IoT scenarios and evaluates how resource management strategies affect network

latency and congestion. The proposed approach is superior than existing strategies based upon expensive bilinear pairing and elliptic curve techniques, according to experiments. The key generation and data modification times of the suggested scheme have significantly improved.

Using Attribute Based Encryption (ABE) and Semantic Web technologies, paper of Edemacu et.al [22] explains our innovative cloud-based EHR system that enables differential access to an EHR and ensures that only Users with acceptable attributes can access a specific field of the EHR. The system also has searchable encryption, which enables EHR fields to be queried without having to decrypt the full patient record. By giving the Cloud Platform Provider control over the modification of the secret key plus ciphertext, the attribute revocation functionality in their EHR is effectively controlled (CSP). Their approach contains cutting-edge security elements that prevent malicious use EHR data and make a substantial contribution to maintaining safe cloud-based digital health systems.

Yiyuan et.al [23] propose an optimised vector transformation approach to effectively convert the access control policies ciphertext policy attribute-based encryption (CP-ABE) methods, such as their high running costs and end-user attribute privacy for the Internet of Things (IoT) and smart health, by encoding user attribute set into matching vectors of shorter length (s-health). By using their transformation technique, the costly overhead of the key production, encryption, and decryption processes can be greatly minimised. The authors then offer a simpler policy-hiding CP-ABE solution on the transformation technique and offline/online computing technology for this kind of IoT targeted s-health applications. They proposed a solution that would enable data users in the s-health system to execute basic decryption and encryption operations without revealing any personal information about the user's features.

The Han Yu et.al of paper [24] suggest an effective ABE technique that delegates some cryptographic operations to edge nodes and enables attribute modifications, providing flexible right control. Their system is confirmed to be secure under the Decisional Bilinear Diffie-Hellman assumption by a formal security proof. The

effectiveness of the authors' scheme is assessed at different security grades, and the test findings show that their method is more effective for devices with limited resources than the conventional ABE.

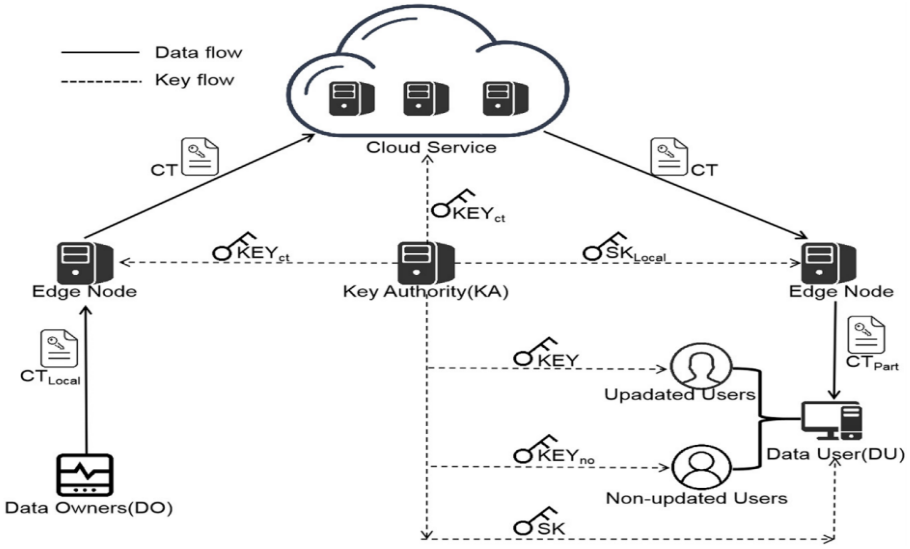


FIGURE 2.1: Proposed Architecture of [23]

A new multi-user CP-ABE system with keyword search feature is proposed in paper by Nidhya, R and Shanthi [25], allowing data users to look for certain ciphertext on a cloud server by utilizing a specified term. Additionally, they use a separate proxy in the suggested system design to sever client and cloud server contact, preventing direct attacks on cloud servers and lowering the computational strain on cloud servers. Their suggested encryption algorithm runs faster and produces relatively short ciphertext. Additionally, only one exponentiation computation is needed for the pre-decryption and re-encryption processes, respectively.

A secured framework for viewing and accessing health data acquired by WBAN is suggested by Zhishuo and Xiong [26]. The Improved RSA (E-RSA) authentication method is made to offer high security. Some user attributes are taken into account when generating secret keys for users in addition to master keys in order to increase the security of the process of transmitting health data. Random values enabling secret key generation are also generated using the bilinear mapping approach. Their simulation model shows how effectively the suggested solution

protects the privacy and confidentiality of patient health information in remote patient monitoring.

The attributes inside the access structures would be given to users in cleartext along with the ciphertext in order to address the issues with traditional ABE [27], which are the first issue. A potential attacker can thus use the plaintext access policy to acquire some of the private data. Another problem is that the conventional ABE scheme is unable to effectively revoke users' illegal keys. To combat this, writers created a large ciphertext-policy ABE (CP-ABE) method that enables both very efficient key revocation and concealed access structures (PHAS). Their access structure is solely based on expressive linear secret sharing (LSSS), that supports both AND/OR gates in access formulas. Their scheme is developed from prime-order bilinear pairing groups. The comparability with other relevant papers demonstrates how much more thorough and efficient the author's methodology is.

A compact CP-ABE scheme using ECC was devised in paper [28] for an IoT-based medical system. The CP-ABE scheme's suggested key management technique is key-escrow free and considerably lowers the data receiver's decryption overhead. The peculiarity of this technique is that it cannot decrypt any text using the secret keys generated by the nearly fully authority unless it also obtains the extra secret key of the recipient. The performance analysis showed that the proposed scheme is superior to the currently used competing schemes.

For PHR, a HAP-CP-ABE-based encryption method and HAP-based authentication scheme are suggested by Zhiguang et.al [29]. The intended work is divided into three phases: (A) the authentication stage, during which verification takes place before the user transfers the PHR files. (B) Protect upload phase: The PHR files' features are all originally extracted. The features are then reduced using ENT-LDA based on the retrieved features. HAP will then be produced utilizing SHA-512. Lastly, use HAP-CP-ABE to safely upload PHR to the cloud. (C) The system confirms the HAP for the request user during the secure download phase.

If they match, they are given access; otherwise, the system denies permission. The system allows the desired user to receive the file after authentication.

Hassan et.al [30], the concealed policy CP-ABE security flaw of a novel ciphertext-policy attribute-based encryption (CP-ABE) variant is examined. (HP-CP-ABE). One issue is that numerous HP-CP-ABE schemes' access regulations include attribute values that an attacker can find through attacks that guess attribute values (AVGA). Another issue is that if the HP-CP-ABE schemes use "Linear Secret Sharing Schemes (LSSS)" as their own access structures, the calculation cost of a decryption testing technique would rise sharply as the rows of an LSSS matrix grow, considerably increasing the computing strain placed on the user. They recommended a partially HP-CP-ABE (PHP-CP-ABE) method that can withstand attacks that aim to accurately predict attribute values (AVGA). Writers have also developed an online private information decryption checking method that enables customers to outsource the decryption testing procedure to a cloud server quietly and securely. In conclusion, it is clear that from the perspective of functionality and efficiency, the proposed scheme is superior to and highly efficient when compared to the most cutting-edge HP-CP-ABE methods.

Introduces PASH [31], an anonymity s-health access control system. A huge universe CP-ABE with partially hidden access restrictions serves as the system's main component. In PASH, just the attribute names are made public while the parameters of access privileges are concealed in encrypted SHRs. In actuality, sensitive information is carried by attribute values rather than by generic attribute names. In particular, PASH implements an effective SHR decryption test that only requires a few bilinear pairs. While the size of public parameters is limited and consistent, the attribute universe has an exponentially wide range. PASH is more expressive and efficient than earlier methods, according to evaluating performance and experimental findings.

Authors created and developed a safe, fine-grained access control mechanism with simple access policy updates for outsourced PHRs in the study [32]. Their

suggested method is based on proxy re-encryption and ciphertext policy attribute-based encryption (CP-ABE) (PRE). In order to facilitate the complete traceability of policy changes, they also devised a policy versioning technique. They also carried out a performance evaluation to show how effective the idea was.

The paper by Rao et.al [33] recommends using Multi-Authority Ciphertext Policy Attribute Based Encryption with Elliptic Curve Cryptography (MA-CPABE-ECC) for web data sharing since it has a smaller key size, more security, and needs less calculation time. The suggested approach uses user information to produce the keys. To increase security, these keys are encrypted and treated as dots on an elliptical curve. The system was subjected to a security and performance analysis, and the results showed that the suggested scheme might provide stronger protection for data transmission to the cloud.

For web patient health record data, the authors Pillai et.al [34] produced a CP-ABE plot for proficient decryption, in which both the size of control of design and the cost of data encryption are constant. The authors further showed that the proposed method achieves full safety inside the standard version by employing the dual systems scheme.

New user secure data sharing architecture for web PHR systems is described in paper [35]. The goal of the study is to develop effective, safe health care information systems (HCIs). HCIs may be kept on a different party's cloud. The patients who have registered as users in the system establish their own HCIs by describing their illness, symptoms, and other pertinent information. Doctors and are also a member of the system respond to patients' updates in the cloud and their questions. The system administrator updates the doctor's prescription, but the cloud administrator does not have access to personal records.

The Internet of Medical Things (IoMT) ecosystem is used to present an effective, outsourced online/offline revocable cipher - text policy attribute-based encryption technique[36]. Their solution satisfies the requirements of quick encryption, ciphertext verification, user revocation, outsourced decryption, and fine-grained access control. It is interesting that they built the data user's private key

with collision resistant, semantic security, and key-exposure freedom relying on the chameleon hashing algorithm in order to achieve revocation. It is demonstrated through formal analysis to be secure in a game of selective replayable chosen-ciphertext attack (RCCA). Their simulation findings show that it supports the IoMT ecosystem with great reliability, high efficiency, and practicality.

For the Internet of Things (IoT), a novel CP-ABE framework elliptic curve cryptography (ECC) is put forth in [37]. It substitutes a straightforward scalar multiplication for the bilinear pairing operation and offloads the majority of the decryption work to advanced practice. The suggested approach also combines time and location features, allowing users to only access data within the time and location parameters established by data owners to create a more granular access control function. The technique simultaneously addresses the bottleneck problem of using one authority by using many authorities to handle attributes. An evaluation of the suggested scheme's performance shows that it is efficient and appropriate for electricity IoT.

An innovative collaborative method for protecting privacy built on top of Ciphertext-Policy Attribute-Based Encryption is presented in Paper [38] as SHARE-ABE (CP-ABE). The most time-consuming decryption tasks are delegated to Fog nodes using their method, which makes use of fog computing. They work together to decode the data partially utilising a unique and effective chained architecture. A new construction of a cooperation attribute has also been included, enabling users within the same group to combine their attributes while still adhering to the access rules. Experiments and security property assessments show that the suggested system is efficient and secure, especially for IoT devices with limited resources.

An efficient ABE strategy is presented in [39]for edge-enabled digital health-care systems which send and store a substantial amount of medical data. Flexible right control is made possible by the proposed system, which permits attribute updates and offloads a part of decryption and encryption to the edge nodes. A

formal security demonstration demonstrates that the authors' method is safe under Decisional Bilinear Diffie-Hellman assumption. When the effectiveness of the suggested scheme is assessed at different security levels, the experimental results demonstrate that their new scheme is more efficient for devices with low resources than the traditional ABE.

2.3 Related Work Analysis

We have performed an analysis on the basis of literature review and compared the work proposed by different authors on the basis of above questions and then find a research gap See Table 2.1.

2.4 Gap in the Literature

From this critical analysis it has been figured out that all the attribute-based encryption techniques are applied on patients' sensitive data for its protection in different ways. Some authors have combined the different flavors of ABE algorithm for the better results. Some of the authors have considered different techniques other than ABE. However, we have revealed that researched techniques have not used any of the original technique that can encrypt the data from the edge/fog node. Sharing of keys is also between encryption and decryption is big issue. It is because for sharing the keys we have to involve third party namely the key generation center. That's why solution will become costly.

Also, none of the authors have defined any attribute neither for the data owners nor for the data users which will be used for encryption. Our research has shown that in order to secure the patient's sensitive data, the CP-ABE, a type of ABE, needs to be given more thought. Therefore, we have proposed a technique which is lightweight and is secure enough that can be used in SAFE-RH project. After experimentation, we will compare the results with base paper for the cadre of confidentiality and performance.

TABLE 2.1: Depicts the Summary of Related Work

Paper Reference	Year of Publication	Domain Area	Proposed Scheme	Data Storage	For Resource Constraint Devices	Outsourced Partial Encryption/ Decryption	Access Structure	Proxy	Achievement by Proposed Scheme
[15]	2013	Personal Health Record	Multi Authority ABE	Cloud	-	-	AND-OR-NOT	-	Security & Scalability Confidentiality Authentication
[16]	2017	E-Health	Three-factor Authentication with ABE	Cloud	Yes	-	-	-	Privacy
[17]	2017	Personal Health Record	Ciphertext-Policy Attribute-Based Signcryption	Cloud	-	-	LSSS	-	Confidentiality
[18]	2018	E-Health Big Data Systems	CP-ABE with Outsourced Decryption	Cloud & Mobile	Yes	Cloudlet/Fog	AND Gates Multi Valued Attributes	Yes	Confidentiality & Integrity
[19]	2018	Smart Health	PH – CP – ABE with Large Ciphertext	Cloud	-	-	LSSS	-	Confidentiality Privacy
[20]	2019	Health Care System	ABE with Data Deduplication	Cloud	-	-	-	-	& Data Redundancy
[21]	2020	Smart Health	Policy Hiding CP-ABE	Cloud	Yes	-	AND-Gate and Wildcard Ordered	-	Security
[22]	2021	Healthcare Systems	CP-ABE	Cloud	-	Cloud	Binary Decision Diagram (OBDD)	-	Security
[23]	2021	Smart Healthcare	Outsourced ABE	Cloud & Edge	Yes	Edge Node	AND -Gates	-	Security
[24]	2021	Medical Cloud System	KwS-CP-ABE	Cloud	-	Proxy Server	LSSS	Yes	Security Confidentiality Authentication
[25]	2021	Patient Health Data	E-RSA	Cloud	-	-	-	-	Integrity
[26]	2021	Healthcare	Partially Hidden CP-ABE	Cloud	-	-	LSSS AND/OR Gates	-	Confidentiality
[27]	2021	IoT healthcare systems	ECC based CP-ABE	Cloud	-	-	LSSS	-	Confidentiality Confidentiality
[28]	2021	Patient Health Record	HAP-CP-ABE	Cloud	-	-	Hashed Access Policy (HAP)	-	Authentication Verification
[29]	2021	IoT	PHP-CP-ABE	Clond & Edge Cloud	Yes	Cloud	LSSS	-	Confidentiality
[30]	2021	E-Health Care	CB-PReS	& Fog	-	-	-	Yes	Confidentiality Security
[31]	2021	Personal Health Record	PRE Proxy-CP-ABE	Cloud	-	-	LSSS	Yes	And privacy Security
[32]	2021	Generic	ECC based MA-CP-ABE	Cloud	-	Cloud	LSSS	-	and Authentication
[33]	2021	Personal Health Record	CP-ABE with proficient decryption	Cloud	-	-	AND/ OR Gates	-	Confidentiality
[34]	2021	Personal Health Record	Weighted CP-ABE	Cloud	-	-	-	-	Confidentiality Privacy
[35]	2021	Internet of Medical Things (IoMT)	CP-ABE Outsourced with Revocation	Cloud & Edge Cloud	Yes	Cloud	LSSS	-	Authorization Security
[36]	2021	Power IoT	ECC based CPABE	& Edge Cloud	Yes	Edge	LSSS	-	Security
[38]	2021	Smart Health	HP-CP-ABE with Collaboration	& Fog Cloud	Yes	Fog Nodes	AND/OR Gates	-	Confidentiality Privacy
[39]	2021	Smart Healthcare	Outsourced ABE scheme	& Edge	Yes	Edge Nodes	AND/OR Gates	-	Confidentiality

Surely, this approach will increase the confidentiality, integrity level of the patient as well as the IT health-based sector. It will also open a new concept of protection on health care sector in which edge/fog is used.

2.5 Conclusion

In total we have reviewed 24 research papers related to attribute-based encryption and its type in health-related domain. In our literature, we have found that different flavors of CP-ABE have been used for different scenarios to secure the data in terms of cloud computing. Some authors have used Multi-Authority CP-ABE for key generation whereas other authors have used proxy servers so that attacks cannot be directed to the main server. This shows that very little work has been done by authors on securing sensitive data which comes directly from the fog node using CP-ABE.

Chapter 3

Proposed Methodology

3.1 Introduction

Based on gap in the literature review, this chapter describes in detail practically our proposed system for encryption of the patient's sensitive data (the owner) as well as decryption of that data by different data users.

Chapter 3 is divided into different sections. Section 3.1 describes the proposed system, Section 3.2 describes the CP-ABE algorithm, Section 3.3 describes formal construction of our proposed system.

3.2 Proposed System

Patients' data in our suggested architecture is split into two categories: updated data and live data. Patients' wearable sensor devices generate real-time data, which is uploaded to the edge/fog nodes closest to the patients. While the processed data, which is uploaded from the edge/fog node to the cloud server, is the updated data. Data users can access secret data from edge/fog nodes as well as cloud servers by using their private keys to decrypt it. This is shown diagramatically in Figure 3.1

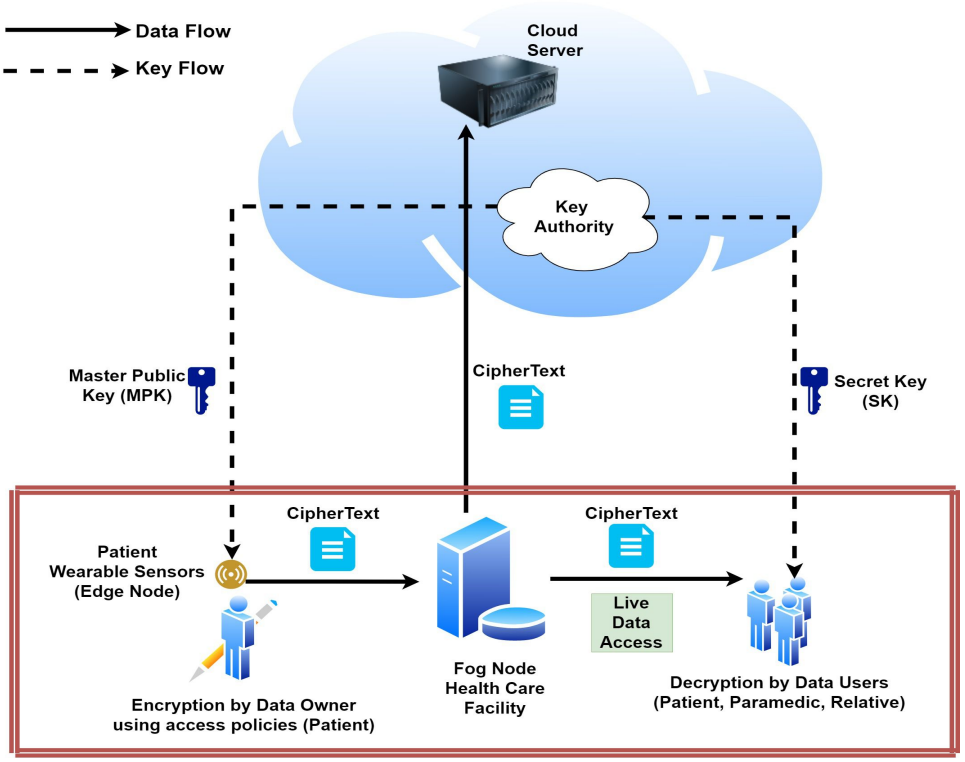


FIGURE 3.1: An Architecture of our Proposed Work

Key Authority (KA): The system setup, also known as "Setup," and the generation of public parameters as well as the distribution of attribute keys (ABE-Keys) linked to DU's attributes would be handled by the certificate authority (CA).

Data Owner (DO): The DO is a patient who establishes the PHR data or medical records' access policies.

Cloud Server (CS): The CS offers data retrieval and storage services. Users of the data can request the matching ciphertext, and it will then provide it back.

Edge/Fog Server (EFS): The EFS offers storage and data retrieval services. It will keep encrypted data that only authorized users, such as doctors and paramedic staff, will have access to.

Data User (DU): A doctor, paramedic, or patient's relative who has access to the patient's encrypted data on the CS is known as the DU. The ABE-key, which must be used to decrypt ciphertexts, is specifically owned by the DU.

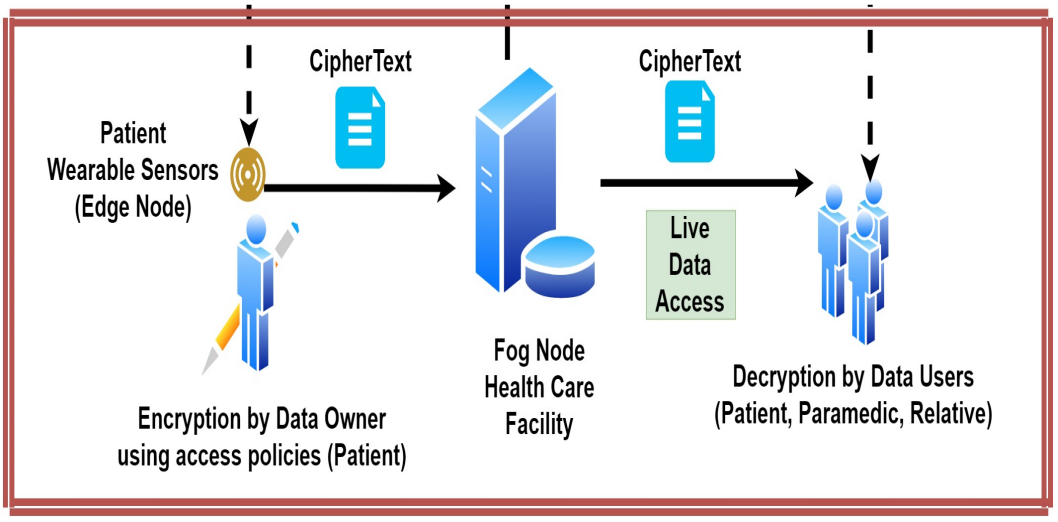


FIGURE 3.2: Patient retrieving data directly from Fog Node

A light weight CP-ABE variant is proposed for live data due limited resources of edge nodes as shown in Figure 3.2.

3.3 CP-ABE Algorithm

The four (4) algorithms that make up CP-ABE are listed below:

Setup: KeyGen Authority runs this algorithm to start up the system. It generates the master secret key (MSK) and the master public key (MPK).

KeyGen: KeyGen Authority executes this algorithm. Considering the inputs of the master secret key (MSK), the master public key (MPK), and the attributes list (L). It provides users with a secret key (SK).

Encryption: This algorithm is run by the owner of the data (Patient). It produces the ciphertext after receiving the plain message, access policy, and master public key as inputs (CT).

Decryption: The data user (paramedic/relative) runs this algorithm. If the policy is met, it outputs the message (M) using the user's secret key set (SK) and the ciphertext (CT).

3.4 Proposed Architecture Construction

Setup (λ)ßPK, MK: The implicit argument must first be input. Define the set of characteristics $P = m1, m2, ..., mn$. Let g be a generator of $G1$, and let $G1$ be a bilinear group of the prime order q. [40]

Let bilinear map

$e : G1 \times G1 \rightarrow G2$

The method then chooses two integers at random $\alpha, \beta \in Z_p$, and for each $m_j \in P$, selects a random $n_j \in Z_p$, then let $PK_j = g_n j$. The algorithm finally produces:

$PK = (G1, H(.), g, h = g\beta, e(g, g)\alpha, (PK_j = gn_j | m_j \in P))$ and the master key MK $= (\beta, g\alpha, (n_j | m_j \in P))$

KeyGeneration $(S, MK) \rightarrow SK$: This algorithm chooses two random $r, t \in RZ_q$, then chooses $r_j \in RZ_q$ for each property $j \in S$. It generates the user's local private key.

$SK_l = (d = g\alpha + r\beta, t)$

Encryption $(M, PK, T) \rightarrow CT$: The message M is encrypted using this technique with the access tree T, where $M \in G2$. The technique chooses a random number $s \in RZ_q$ for the root node R. T is a subtree of T_1, and $T = T1 \bigcup T2$ indicates that. The algorithm selects $s_1, s_2 \in RZ_q$, lets $qR(1) = s1, qR(2) = s2$, and chooses a 1-degree polynomial $q()$ that $qR(0) = s$. There is only one virtual attribute in the T_2. The algorithm then generates a portion of the ciphertext: y

$C = (C \simeq= Me(g, g)\alpha s, C = hs, \vee y \in Y2 : Cy = gq_y(0)n - 1j, C'y = H(att(y))q_y(0)n_j)$.

$Decryption(CT, SK) \rightarrow M$: The recipient uses his SK in conjunction with the attribute list S to attempt to decrypt CT without knowing the access policy W as follows:

$$\sim C(FR)1te(C,d) = Me(g,g)\alpha se(hs, g\alpha + r\beta)e(g,g)rs = M$$

3.5 CP-ABE in Proposed Scenario

Setup: The KeyGen Authority creates the Master Public Key (MPK) and Master Secret Key (MSK).

FIGURE 3.3: Setup Algorithm of CP-ABE

The CP-setup ABE's phase is the first one. CPABE is initialized in SETUP. The system generates Master Public Key (PK) and Master Secret Key using random security parameters (MSK). It is operating on a cloud server in our hypothetical situation.

Key Generation: Secret Key is generated for data users based on attribute lists of data owner by KeyGen Authority.

This is KEYGEN, the second procedure of CP-ABE. Secret keys are generated via KEYGEN. KEYGEN produces secret keys by using as inputs the Master Public Key, the Master Secret Key, and characteristics. It will also be operating on the cloud server in our example.

Encryption: Encryption is done by data owner results in output of ciphertext and is stored to the fog server.

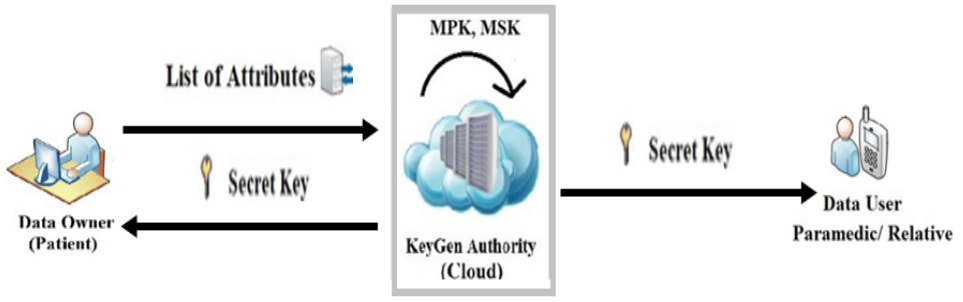


FIGURE 3.4: Key Generation Algorithm of CP-ABE

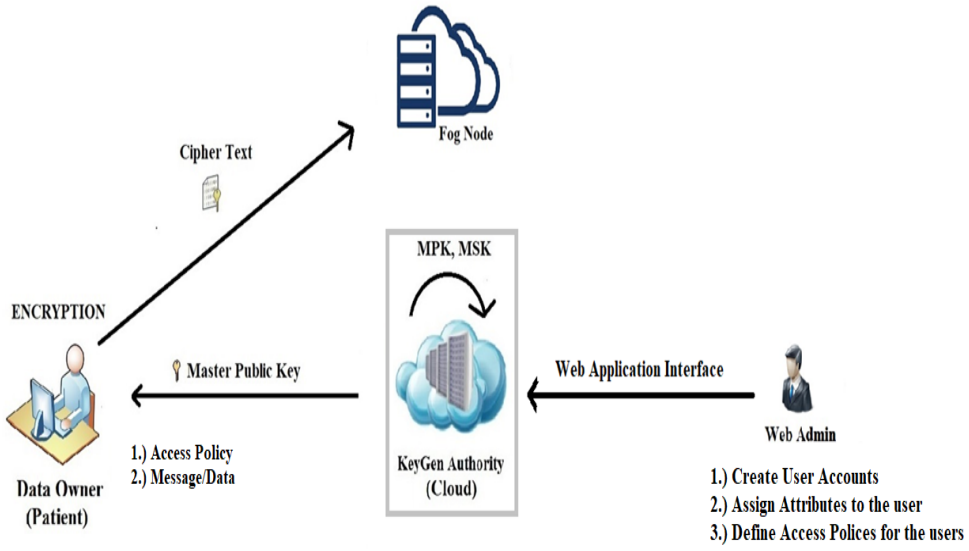


FIGURE 3.5: Encryption Algorithm of CP-ABE

This third procedure in the CP-ABE is called ENCRYPT. Data is converted from plain text to encrypted message using the ENCRYPT function. Master Public Key, Policy, and Message/Data are the three inputs that ENCRYPT uses to produce the appropriate cypher text. Access policies is set once Web Administrator has authenticated in to the system. The data owner (Patient) will encrypt its data and transfer it to the fog server in accordance with those established policies.

Decryption: Decryption is done by data user secret key after satisfying the policy.

The DECRYPT procedure is the last one in CP-ABE. Data is converted from encrypted message to plain text in DECRYPT. DECRYPT outputs the desired

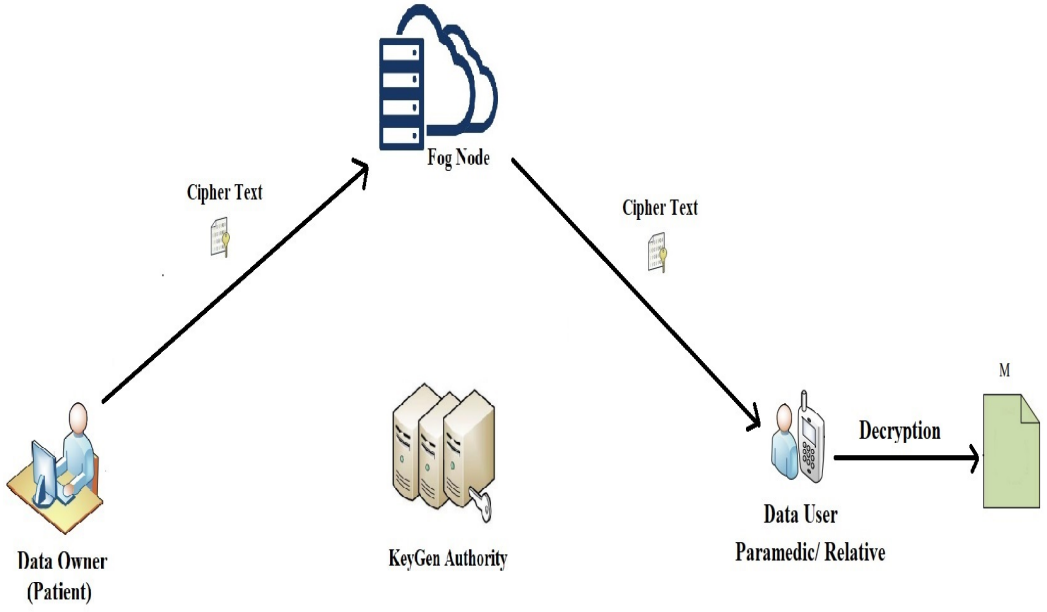


FIGURE 3.6: Decryption Algorithm of CP-ABE

plain text after receiving the Secret Key and encrypted Message/Data as inputs. Data users can decode encrypted messages if the access policy is met with the user attributes, and vice versa.

3.6 Conclusion

In this chapter, we have described our proposed architecture. Moreover, we have also described that how we can effectively secure our data using the original CP-ABE for viewing data directly from the fog node/server. Also, we have justified our proposed methodology using formal language.

Chapter 4

Evaluation, Implementation and Results

4.1 Introduction

This chapter is about the analysis and experimental set up of the proposed system. This system focuses on encryption and decryption methods that how we protect the sensitive information of the patients. We have developed a system in which data owner will encrypt the sensitive data based on its own attributes and policies. Data users that will satisfy the policy can only decrypt the data.

We have divided this chapter into different sections. Section 4.1 describes the security model , Section 4.2 describes the base paper we have selected, Section 4.3 describes the policies on which data owner will encrypt the data, Section 4.4 describes the implementation and experimentation of our proposed scheme, Section 4.5 describes the performance results of our proposed scheme which is compared with the base paper proposed scheme

4.2 Security Model

Based on a semantic security game, it was first presented by Zhou and Huang [41]. In this paradigm, a challenge's "C" cannot be immediately decrypted using any secret key obtained through an attacker's "A" query of an attribute set's "S" secret key.

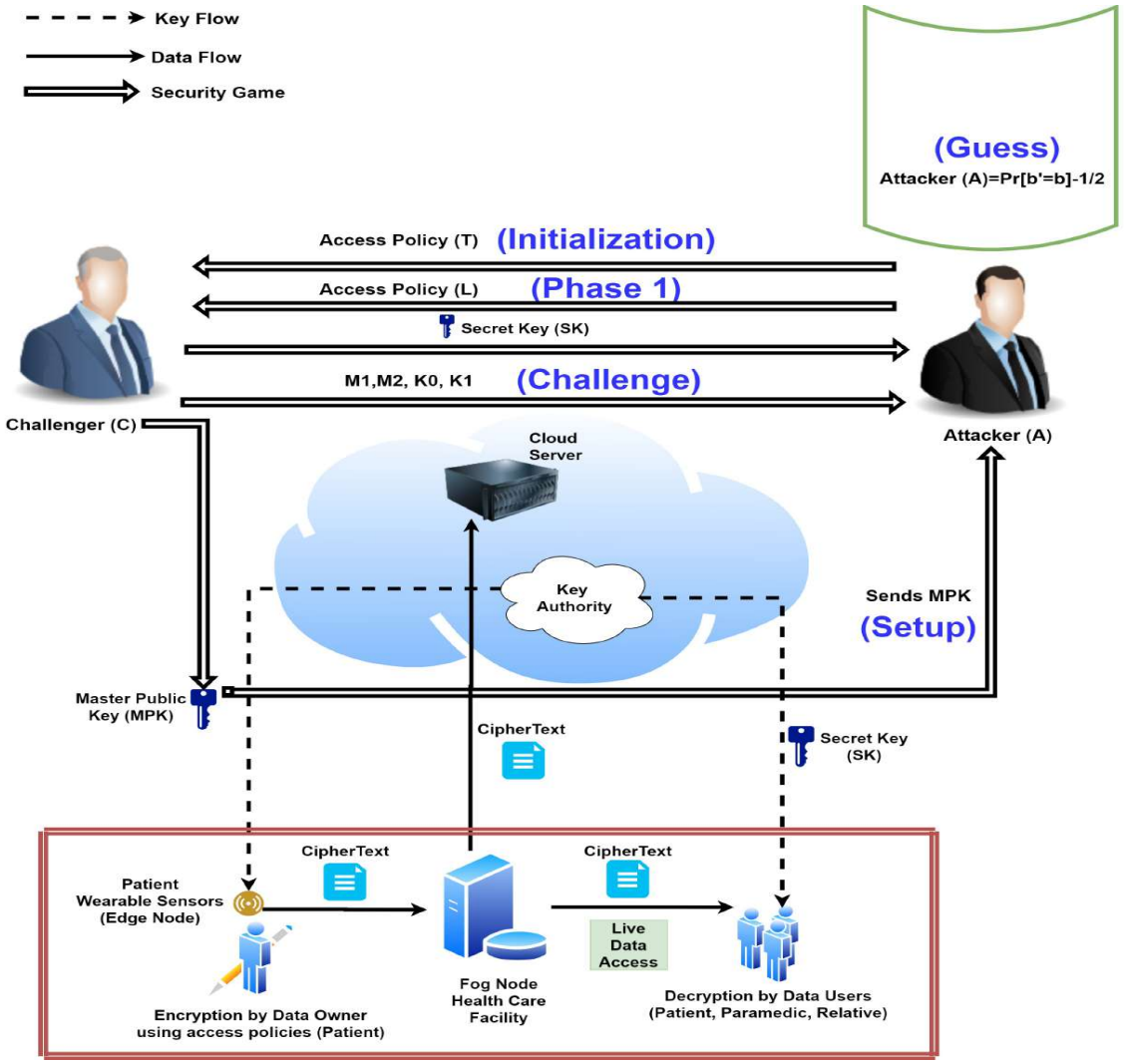


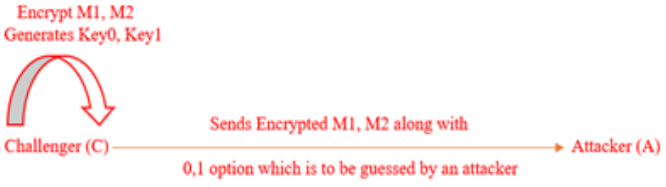
FIGURE 4.1: Security Model of Our Proposed Architecture

Initialization: The challenger receives the attacker's choice of the challenger access structure policy, "T".

Setup: The public key "PK" is generated during the Setup step by the challenger "C" who then sends it to the attacker.

Phase 1: The challenger is asked by the attacker "A" for the secret key for the access structure policy "L". At this point, if "L" does not fulfil "T" the challenger "C" transmits the attacker the secret key "SK" that corresponds to "L". If necessary, the attacker can go back to Phase 1.

Challenge: The challenger "C" uses the encryption procedure to create Key_0 and Key_1 for the messages "$M1$" and "$M2$". The challenger creates a random key, Key_1, and then sets $Key_0 = Key$ for the original Key. The two keys, Key_0 and Key_1, currently have the same sizes. The challenger sends the attacker "A" the commands "$M1, M2, Key_b$" using the random number $b_{0,1}$.



Guess: The ciphertext is assumed to be $b'_{0,1}$ by the attacker. If $b' = b$ and L does not fulfil T, the attacker is said to have won. Finally, we define the probability of an attacker winning in this game as follows:

"$\backslash Attacker_A = P_r[b' = b] - 1/2$"

As:

- b' is the guess value of key by an attacker.
- b is the actual value set by the challenger.
- $1/2$ is the half probability.

It is clear and confirmed with this security model that our proposed architecture for fog computing is secured against Chosen Ciphertext Attack (CCA).

4.3 Evaluation with Base Paper

KDC will conduct the Setup procedure in their system [42], create the master key MK and the public key PK, and then deliver the PK to DO. The local private key SKlocal and private key SK are then generated by KA using the KeyGeneration method. The edge node receives SK, and the DU receives SK. Then, DO generates a portion of the ciphertext CT using the Encryption: encryption technique and sends the CT to the edge/fog node.

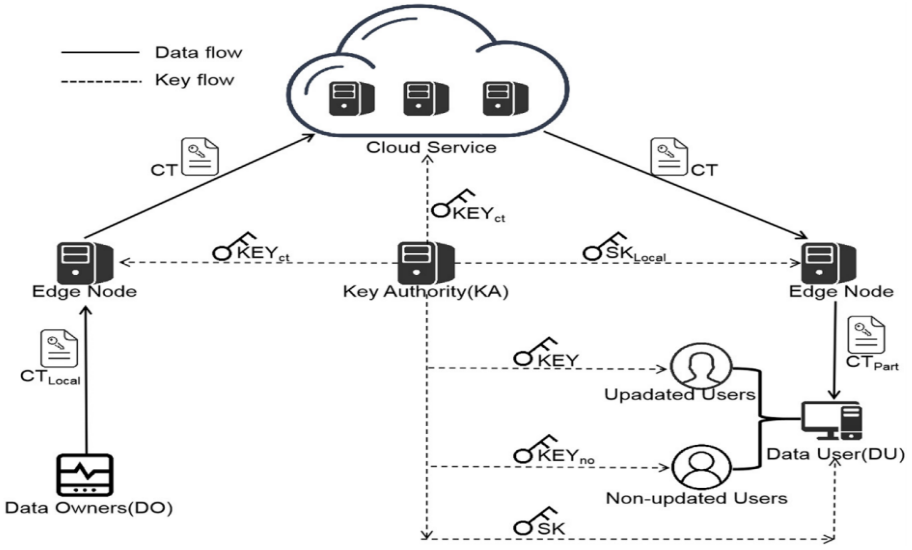


FIGURE 4.2: Base Paper System Architecture

The ciphertext CT is produced by the edge/fog node, which also executes the encryption algorithm and sends the ciphertext CT to the cloud service. the CT is sent by the cloud service to a different edge node. The CT is decrypted, output,

and sent to DU by the edge node using the Decrypt method. For message M, DU uses the Decrypt algorithm.

4.3.1 Base Paper Architecture Vs Proposed Architecture

The key difference between the our proposed architecture and base paper architecture is as below:

- The primary distinction between the proposed architecture and the base paper is that the data produced by various body sensors in the proposed architecture is first transferred to the fog node for transient storage. If the data consumers meet the policy requirements, they can then retrieve this almost real-time data. That data is eventually transmitted to cloud storage afterwards for permanent storage. While in the case of the base paper, the data can only be retrieved from cloud storage by authorized individuals.

- The encryption/decryption technique is another variation between the proposed architecture and the basis paper. The encryption and decryption processes are only performed once in the proposed architecture since we have defined the limited attributes for the data owner as well as the data user for data access that don't burden the resource-constrained edge devices. In the case of the base paper, some of the encryption and decryption is carried out on the edge node and some of it at the edge devices. This requires an additional step and takes time.

- In proposed architecture, data owner which is the patient and can encrypt the data using some access policy is also the data user. This means that data owner can also decrypt the data. In base paper, data owner can only encrypt the data.

- In base paper architecture, user's attributes are not fixed and can be added or removed (Updated). In our proposed architecture, we have defined the attributes both for the data owner and the data user.

- Unlike the base paper design, which has no such restrictions and allows data to be obtained from the cloud, the proposed architecture requires that the data owner and the data user be connected to the same fog node.

- Lastly, the key generating authority in base paper architecture is a separate entity and not a component of a cloud. While under the proposed architecture, the cloud's key generation authority is in charge of issuing keys to data owners and users.

4.4 Policies

An encryption technique called ciphertext policy attribute-based encryption (CP-ABE) [43] connects a group of attributes used in the encryption process to logical access structures, also called attribute policies.

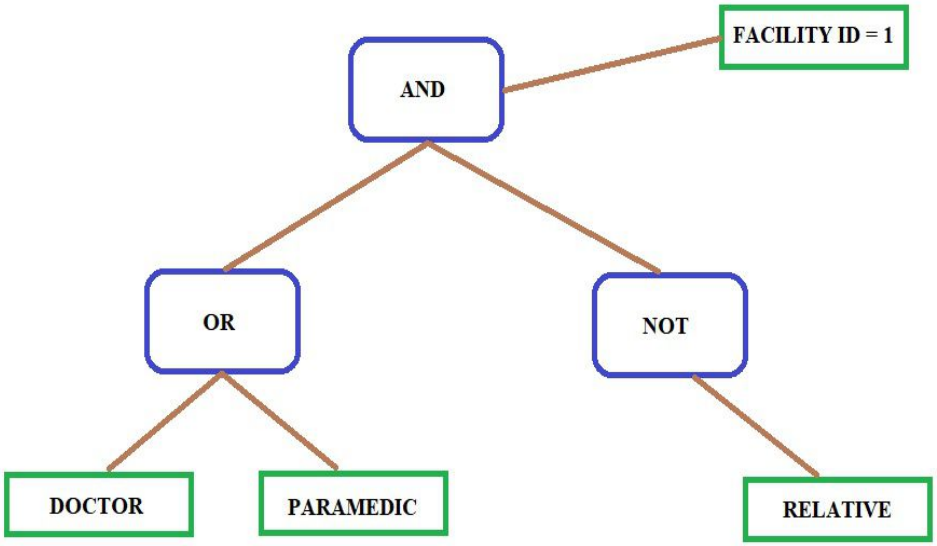


FIGURE 4.3: Flowchart of AND/OR Gates in Policies

As a result, the encryption method needs two inputs: an attribute policy and a message. Once the method encrypts the message and generates a ciphertext, only a receiver with a specific set of qualities that abide by the attribute policy can decode the message. We'll presume that the policy is already present in the ciphertext for the remainder of this section.

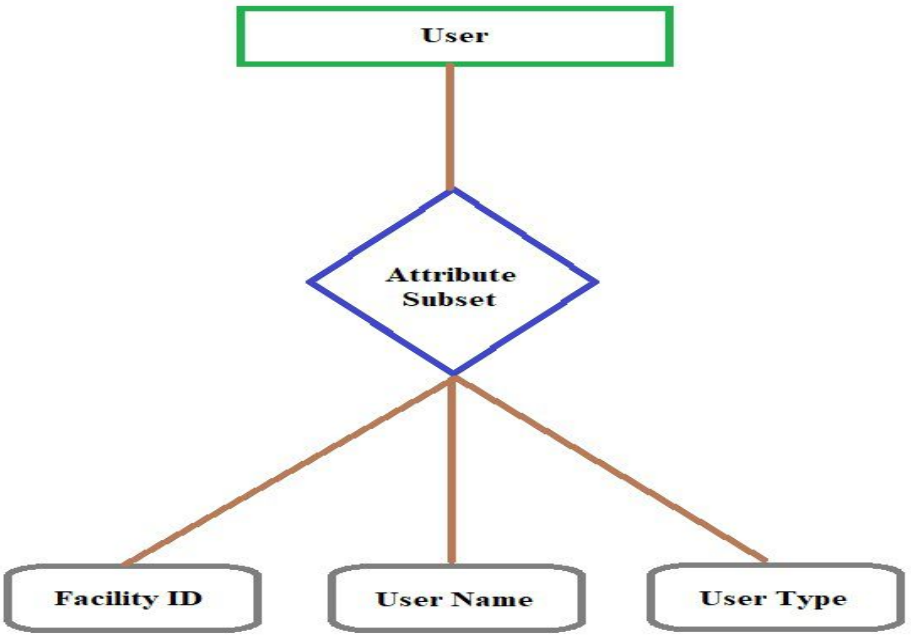


FIGURE 4.4: Flowchart of Attribute in Policies

CP-ABE is used as hybrid encryption in real-world contexts, encrypting the message with a secret key that is produced at random. After then, only this key pair is CP-AB encrypted in line with a policy. The examples of a CP-ABE policy using logical AND/OR gates and a subset of properties derived from our case are shown in Figures 4.4 and 4.5, respectively.

4.4.1 Crypto Library for Policies

A user-friendly application programming interface, several attribute-based encryption (ABE) techniques, and frequently used cryptographic tools are all included in the cryptographic library known as OpenABE [44]. (API). With OpenABE, developers will be able to easily implement ABE technology into programmes that stand to gain from its ability to secure and restrict user access to private information. Developers do not need to be specialists in encryption to use OpenABE because of its user-friendly design. Several attribute-based encryption (ABE) schemes are provided by the OpenABE C/C++ software library, along with other essential cryptographic features like Key derivation functions, digital signatures, X.509 certificate handling, authenticated symmetric-key encryption, public key encryption,

and more are all included. For ABE to work, application developers shouldn't need to be cryptography experts. To keep OpenABE as secure and convenient as possible, the following features are provided by default:

1. Collusion-Resistant: This prevents the common mistake of Alice and Bob combining their private keys to decrypt ciphertexts that none of them can decrypt on their own. Notably, this attack usually succeeds in any attempt to "engineer" ABE using conventional public key encryption.

2. Chosen Ciphertext Attack (CCA) Secure: Prevents severe and practical tampering attacks; the majority of current systems in the academic literature only adhere to a weaker security principle (CPA-security).

3. Unrestricted Attributes: Attributes can be represented by any string and can be used as many times as necessary in a policy. The alternative is that all current and future attributes must be enumerated at system initialization.

4.4.2 Dataset

TABLE 4.1: Sample of our Dataset

F_ID (Facility)	**Location**	**FN_ID (Fog Node)**	**EN_ID (Edge Node)**	**U_ID (User)**	**U_Name**	**U_Type**	**P_ID**
001	Capital Hospital	100	101	1001	Annum	Paramedic	Nil
001	Capital Hospital	100	102	1002	Dr Qamar	Relative	1003
001	Capital Hospital	100	103	1003	Sir Salman	Patient	Nil
001	Capital Hospital	100	104	1004	Dr Nayyer	Relative	3003
001	Capital Hospital	100	201	2001	Ali	Patient	Nil
002	Capital Hospital	200	202	2002	Nadeem	Paramedic	Nil
002	Capital Hospital	200	301	3001	Qadir	Relative	5002
002	Capital Hospital	200	302	3002	Amir	Paramedic	Nil
002	Capital Hospital	200	303	3003	Mahmood	Patient	Nil
002	Capital Hospital	200	401	4001	Masood	Patient	Nil
002	Capital Hospital	200	402	4002	Ayesha	Relative	6001
002	Capital Hospital	200	501	5001	Ishtiaq	Relative‘	2001
002	Capital Hospital	200	502	5002	Najma	Patient	Nil
002	Capital Hospital	200	503	5003	Ejaz	Relative	4001
002	Capital Hospital	200	601	6001	Sherbano	Patient	Nil

4.4.3 Proposed Policies

[POLICY-1 - Paramedics can view records of patients in Single Facility]

"oabe-enc -s CP -p safe-rh -e "((FID == 001) and Paramedic)" -i input.txt -o output.cpabe"

[POLICY-2 - Paramedics can view records of patients in Multiple Facilities]

"oabe-enc -s CP -p safe-rh -e "((FID ¿= 001) and (FID ¡= 003) and Paramedic)" -i input.txt -o output.cpabe"

[POLICY-3 – Patient can view his/her own record]

"oabe-enc -s CP -p safe-rh -e "((FID == 001) and (Capital Hospital) and (UID == 1003) and Patientwithrelative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 001) and (Capital Hospital) and (UID == 2001) and Patientwithrelative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 3003) and Patientwithrelative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 4001) and Patientwithrelative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 5002) and Patientwithrelative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (MaroofHospital) and (UID == 6001) and Patientwithrelative)" -i input.txt -o output.cpabe"

[POLICY-4 - Only concerned Relative can view his/her patient record in the same facility]

"oabe-enc -s CP -p safe-rh -e "((FID == 001) and (Capital Hospital) and (UID == 1002) and (PID == 1003) and Relative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 001) and (Capital Hospital) and (UID == 1004) and (PID == 3003) and Relative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 3001) and (PID == 5002) and Relative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 4002) and (PID == 6001) and Relative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 5001) and (PID == 2001) and Relative)" -i input.txt -o output.cpabe"

"oabe-enc -s CP -p safe-rh -e "((FID == 002) and (Capital Hospital) and (UID == 5003) and (PID == 4001) and Relative)" -i input.txt -o output.cpabe"

Decryption

"oabe-dec -s CP -p safe-rh -k Annum.key -i output.cpabe -o Annum-input.txt"

"oabe-dec -s CP -p safe-rh -k DrQamar.key -i output.cpabe -o DrQamar-input.txt"

"oabe-dec -s CP -p safe-rh -k SirSalman.key -i output.cpabe -o SirSalman-input.txt"

"oabe-dec -s CP -p safe-rh -k DrNayyer.key -i output.cpabe -o DrNayyer-input.txt"

"oabe-dec -s CP -p safe-rh -k Ali.key -i output.cpabe -o Ali-input.txt"

"oabe-dec -s CP -p safe-rh -k Nadeem.key -i output.cpabe -o Nadeem-input.txt"

4.5 Implementation

The system environment is set up with a 2.30 GHz Intel i7 CPU, 16 GB of RAM, 64-bit linux OS, and a 1 TB solid state drive to implement CP-ABE encryption for patient data integrity, data privacy, user privacy, and fine-grained access control. Pairing-based cryptography library is used to implement inner product encryption [45]. Our approach makes use of an open-source C library for pairing-based encryption that is based on the GNU libraries for multiple precision arithmetic [46]. Our strategy is compared to other pertinent plans through simulation and implementation.

4.5.1 Crypto Library For Implementation

The design of sophisticated cryptosystems can be completed quickly using a framework called Charm [47]. Based on Python, it was designed from the ground up to reduce code complexity and development time while promoting component reuse. Charm is a hybrid design where the cryptosystems themselves are designed in a legible, high-level language and the performance-demanding mathematical processes are carried out in native C modules. Charm also offers a large selection of unique components that speed up the creation of novel schemes and protocols. Charm includes the following features:

- Support for a variety of mathematical contexts, including integer rings, fields, and bilinear and non-bilinear elliptic curve groups
- Common APIs for commitments, encryption, and digital signature constructions are included in the base crypto library together with symmetric encryption techniques, hashing algorithms, and PRNGs.
- A "protocol engine" to enable creating multi-party protocols easier; an integrated benchmarking capability; an integrated compiler for interactive and non-interactive ZK proofs;

4.5.2 Hardware Requirements

Ubuntu 20.04.2 was used as the operating system, and it was paired with an Intel Core i7-7500 processor running at 3.40 GHz, 16GB of virtual memory, and a 1TB NVME drive. Large integer operations are implemented using the GMP package (version 6.1.2). Pairing calculations are implemented using the PBC library (version 0.5.14).

4.5.3 Code of Proposed Architecture Implementation using Crypto Library

"This is the first process in CP-ABE namely SETUP. In SETUP, CPABE is initialized. System takes random security parameters and outputs Master Public Key (PK) and Master Secret Key (MSK). It will be running on the cloud server"
def setup(self):
g1, g2 = group.random(G1), group.random(G2)
alpha, a = group.random(), group.random()
egg alpha = pair(g1,g2) ** alpha
msk = ['g1 ⋆ alpha':g1 ** alpha, 'g2 ⋆ alpha':g2 ** alpha]
pk = ['g1':g1, 'g2':g2, 'e(gg) ⋆ alpha':egg alpha, 'g1 ⋆ a':g1 ** a, 'g2 ⋆ a':g2 ** a]
return (msk, pk)"
"This is the second process in CP-ABE namely KEYGEN. In KEYGEN, Secret keys are generated. KEYGEN takes Master Public Key, Master Secret Key and attributes as an input and outputs secret keys. It will also be running on the cloud server" "def keygen(self, pk, msk, attributes):
t = group.random()
K = msk['g2 ⋆ alpha'] * (pk['g2 ⋆ a'] ** t)
L = pk['g2'] ** t
kx = [group.hash(s, G1) ** t for s in attributes]
Kx = []
key = ['K':K, 'L':L, 'Kx':Kx, 'attributes':attributes]
return key"
"This is the third process in CP-ABE namely ENCRYPT. In ENCRYPT, data is encrypted from plain text to cipher text. ENCRYPT takes Master Public Key, Policy and Message/Data as an input and outputs the desired cipher text. Web Administrator will first logged into the system after which it will define access policies. Based on those defined policies, data owner (Patient) will encrypt its data and send it to the fog server" "def encrypt(self, pk, M, policystr):
policy = util.createPolicy(policystr)

plist = util.getAttributeList(policy)

s = group.random()

Ctilde = (pk['e(gg) ⋆ alpha'] ** s) * M

C0 = pk['g1'] ** s

C, D = [], []

secret = s

shares = util.calculateSharesList(secret, policy)

for i in range(len(plist)):

attr = shares[i][0].getAttribute()

C[plist[i]] = ((pk['g1 ⋆ a'] ** shares[i][1]) * (group.hash(attr, G1) ** -r))

D[plist[i]] = (pk['g2'] ** r)

if debug: print("SessionKey: s" Ctilde)

return ['C0':C0, 'C':C, 'D':D , 'Ctilde':Ctilde, 'policy':policystr, 'attribute':plist]"

"This is the final process in CP-ABE namely DECRYPT. In DECRYPT, data is decrypted from cipher text to plain text. DECRYPT takes Secret Key and encrypted Message/Data as an input and outputs the desired plain text. If the access policy is satisfied with the user attributes then data user can decrypt the encrypted message and vice versa" "def decrypt(self, pk, sk, ct):

policy = util.createPolicy(ct['policy'])

pruned = util.prune(policy, sk['attributes'])

if pruned == False:

return False

coeffs = util.getCoefficients(policy)

numerator = pair(ct['C0'], sk['K'])

kx, wi = [],[]

denominator = 1

for i in pruned:

j = i.getAttributeAndIndex()

denominator *= (pair(C[j] ** wi[j], sk['L']) * pair(kx[j] ** wi[j], D[j]))

return ct['Ctilde'] / (numerator / denominator)"

```
“def main(): (msk, pk) = cpabe.setup()”
'Policy - 1: All PARAMEDIC CAN VIEW PATIENTS RECORD'
pol = '((FID) and (PARAMEDIC))'
attrlist = ['FID', 'FNID', 'ENID', 'UID', 'UNAME', 'PARAMEDIC', 'PID']
'Policy - 2: PATIENT CAN VIEW HIS/HER OWN RECORD'
pol = '((FID and LOCATION) and (UID and PATIENT))'
attrlist = ['FID', 'FNID', 'LOCATION' , 'ENID', 'UID', 'UNAME', 'PATIENT',
'PID']
'Policy - 3: RELATIVE CAN ONLY VIEW HIS/HER PATIENT RECORD'
pol = '((FID and LOCATION) and (PID and UID and RELATIVE))'
attrlist = ['FID', 'FNID', 'LOCATION' , 'ENID', 'UID', 'UNAME', 'RELATIVE',
'PID']
m = groupObj.random(GT)
cpkey = cpabe.keygen(pk, msk, attrlist)
if debug:groupObj.debug(cpkey)
cipher = cpabe.encrypt(pk, m, pol)
if debug: print(”nCiphertext...”)
origm = cpabe.decrypt(pk, cpkey, cipher)
assert m == origm, 'FAILED Decryption!!!'
if debug: print('Successful Decryption!')
```

4.6 Performance Results

We have compared our execution time results of different processes of algorithm with the results of base paper and other similar proposed architectures. Table 4.2 depicts the mentioned results.

Table 4.2: Comparative Results with the Base Paper and Similar Techniques

S.No	Processes/Steps	Base Paper	OABE-E	OABE-D	OABE-ED	Our Proposed Method
1	Key Generation	241.37 ms	383.22 ms	297.41 ms	364.99 ms	179.17 ms
2	Encryption	1403.01 ms	2333.79 ms	1659.21 ms	1682.94 ms	192.65 ms
3	Decryption	150.33 ms	172.99 ms	183.31 ms	204.65 ms	105.34 ms

FIGURE 4.5: Comparison of Key Generation Execution Time

4.6.1 Discussion

The data owner's encryption process and the key generation technique account for the majority of our computing expenses. Table 4.2 compares our system with the computation costs of strategies by the base paper, which uses outsourced encryption and proxy servers, to make things easier to comprehend. Table 4.2 shows that our proposed strategy has a nearly 50 percent lower computational burden than the proposed scheme in the base study. This is so because we just defined the primary properties for the data owner and the user, not any extra attributes. Our system is identical to the original CP-ABE system; however, data is obtained directly from the fog node rather than the cloud node. This is due to our scheme's usage of numerous private keys to guarantee data security. Because the patient's data is of a critical nature, the data owner utilises their own device to encrypt the data during the encryption phase. Additionally, the encryption procedure takes longer than it should because more data is created as a result of the utilisation of various sensors. Key creation required some time after

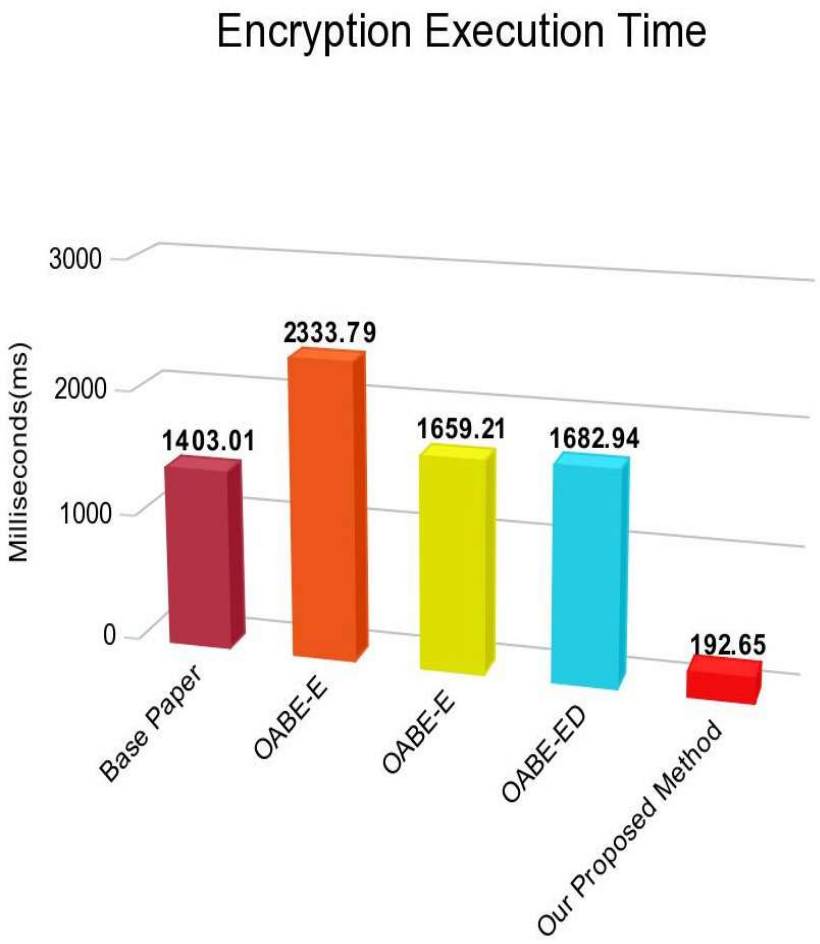


FIGURE 4.6: Comparison of Encryption Execution Time

encryption, although this totally depended on the characteristics of the users and the user base itself. It will take longer to produce the keys for various users and attributes if there are more attributes and users. Data user devices independently decipher the ciphertext produced by the data owner during the decryption phase because our approach does not include an outsourcing decryption process. Because of this, our suggested solution is more secure. As a result, our plan is generally more effective than the plan presented in the original paper.

4.7 Conclusion

In this chapter, we have used Zhou and Huang security model and concluded that our proposed scheme to get data directly from fog node is secured against

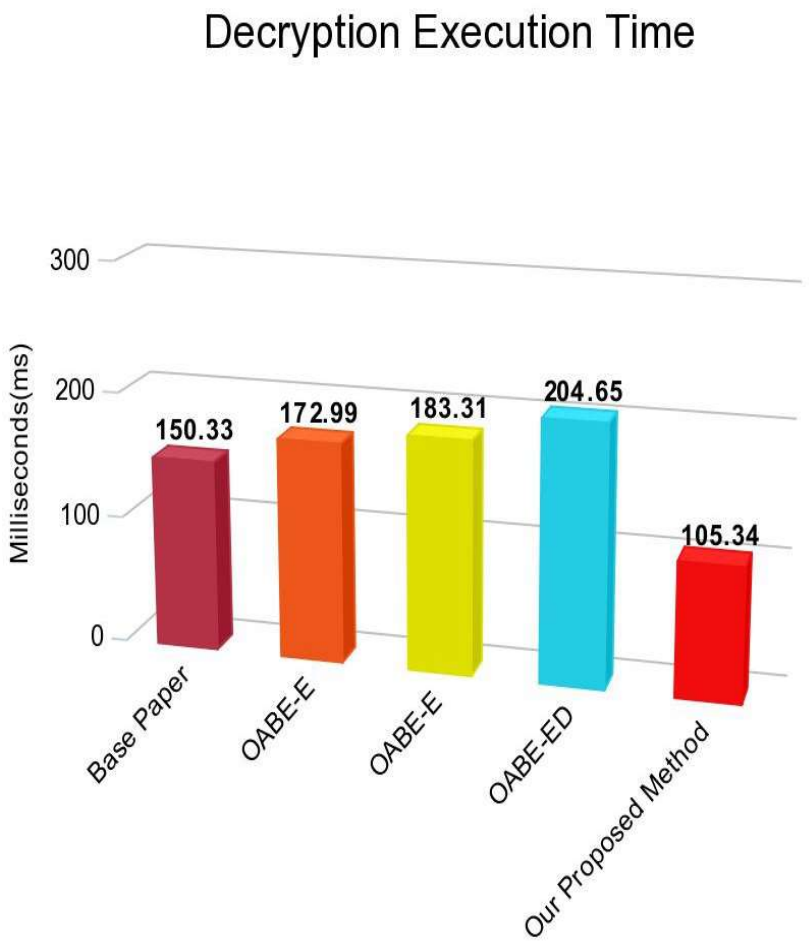


FIGURE 4.7: Comparison of Decryption Execution Time

Chosen CipherText Attack. Also, we have selected the base paper and described our dataset. We have also specified our policies, which the data owner will use to encrypt the information with. If the data complies with those policies, users can access the encrypted data. Also, we have described the tools/libraries and system requirements by using which we have implemented our proposed scheme for our scenario. At the end, we have compared our results with the base paper and claim that our scheme works efficiently and is less time consuming as compared to the selected base papers work by the authors.

Chapter 5

Test Case Studies

5.1 Introduction

This chapter describes in detail the different scenario-based test case studies as per our policies that which user type can decrypt the data aur view the results.

Chapter 5 is divided into different section. Section 5.1 describes the scenario-based test case studies; Section 5.2 describes the security requirements and security evaluation of CP-ABE.

5.2 Scenario

Take the hospital CAPITAL HOSPITAL as an example, which outsources patient shared data to both a fog and cloud server. The fog node is a short-term storage where data is kept for a while before being sent to the cloud platform. Data is accessible from both a fog server and a cloud server. Data is encrypted using CPABE, and access control is implemented. Suppose that a MIS application allows a patient to store all of his health-related data including the blood pressure (BP), temperature (TEMP), oxygen level (SPO2) and heart rate Level (HRL) by storing

them Fog Service Provider (FSP) and in a Cloud Service Provider (CSP). This scenario is depicted as follows:

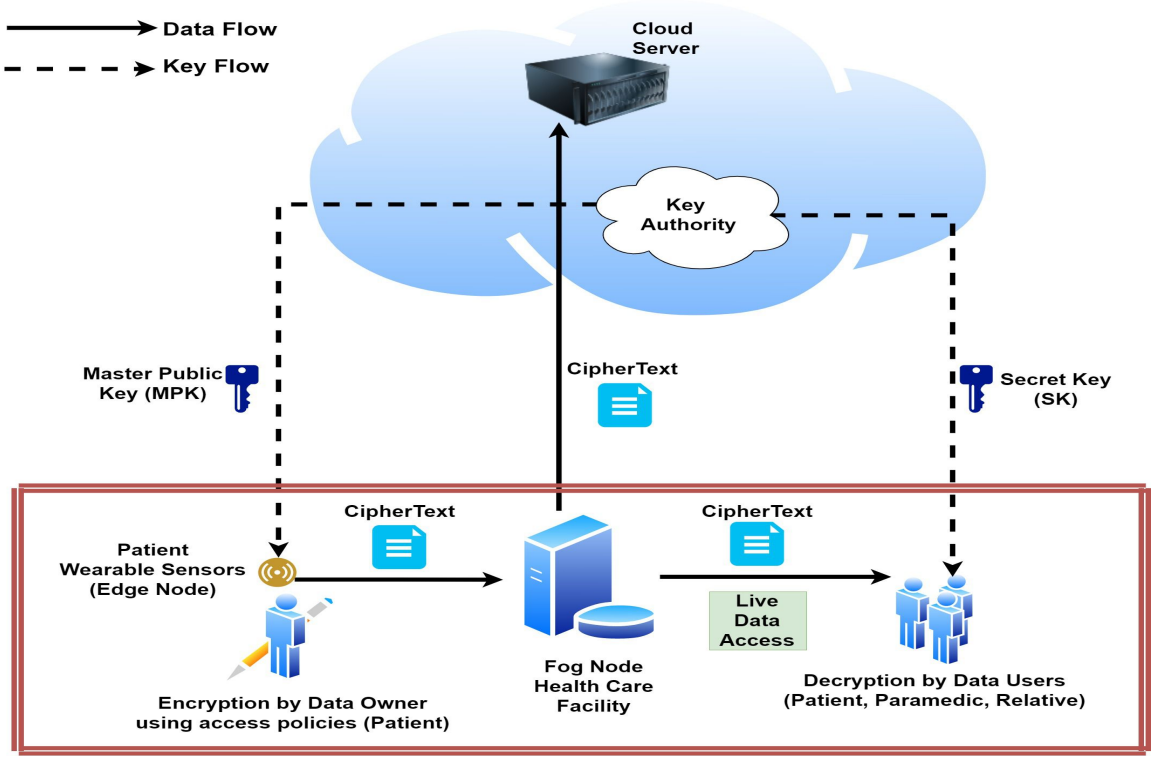


FIGURE 5.1: Our Scenario of Accessing Data

Patient's data that is shared and is send to the Fog Service Provider (FSP) can be viewed by the patient itself, his/her relative and all of the Paramedic/Doctor Staff that is connected to the same fog node only.

TABLE 5.1: Depicts all of the User Attributes and its Values

F_ID (Facility)	Location	FN_ID (Fog Node)	EN_ID (Edge Node)	U_ID (User)	U_Name	U_Type	P_ID (Patient)
001	Capital Hospital	100	101	1001	Annum	Paramedic	Nil
001	Capital Hospital	100	102	1002	Qamar	Relative	1003
001	Capital Hospital	100	103	1003	Salman	Patient	Nil
001	Capital Hospital	100	104	1004	Nayyer	Relative	3003
001	Capital Hospital	100	201	2001	Ali	Patient	Nil
002	Capital Hospital	200	202	2002	Nadeem	Paramedic	Nil
002	Capital Hospital	200	301	3001	Qadir	Relative	Nil
002	Capital Hospital	200	302	3002	Amir	Paramedic	Nil
002	Capital Hospital	200	303	3003	Mahmood	Patient	Nil

A patient encrypts his/her health-related with a specified access policy and then sends to the FSP and then to CSP. The patient's health-related data are encrypted, but only a user with attributes that meet the access control policies

can decipher them while also having access to read them. In our scenario, we have taken two fog service providers (FSP) and defined its attribute as (FN.ID) whose values are '100' and '200' respectively. There are total '3' patients Salman, Ali and Mahmood. Annum, Nadeem and Amir are Paramedic/Doctors whereas Qamar, Nayyer and Qadir are relatives of the patients. All of this data is shown with the relevant attributes and its values in table 5.1

5.2.1 Access Policies for Our Scenario

As mentioned above, there are three different types of users who can decrypt and access the data using CP-ABE. These users can be patient itself, the paramedic/-doctor staff and the patient's relative. The patient who wants to share its health data to individual or to a group of users should satisfy one of the following conditions:

- The patient shall decrypt and access its own health data. No other patient can view someone's else patient data.
- All of the paramedic and doctor staff can view data of all the patients that are connected to the same fog node.
- Relatives can only view data of their own patient and not of any other patient.

Based on the above-mentioned access policies, we have derived several test cases that is highlighted in section 4.1.2 of this chapter.

5.2.2 Test Case Studies

5.2.2.1 Patient wants to view its own data

In our data which is shown in table 5.1 there are three patients namely Salman, Ali and Mahmood. Salman is curious about his health and wanted to view his data

that is generated by different sensors attached to his body. The data is encrypted by salman himself using an access policy and the data is now available in fog server. Moreover, Salman's attributes are F.ID (Facility), Location, FN.ID (Fog Node), EN.ID (Edge Node), U.ID (User), U.Name, U.Type, P.ID(Patient). Some of these attributes might have different values. As Salman is patient himself, he will easily decrypt and access his data as his access policy is satisfied with the attributes and its corresponding values.

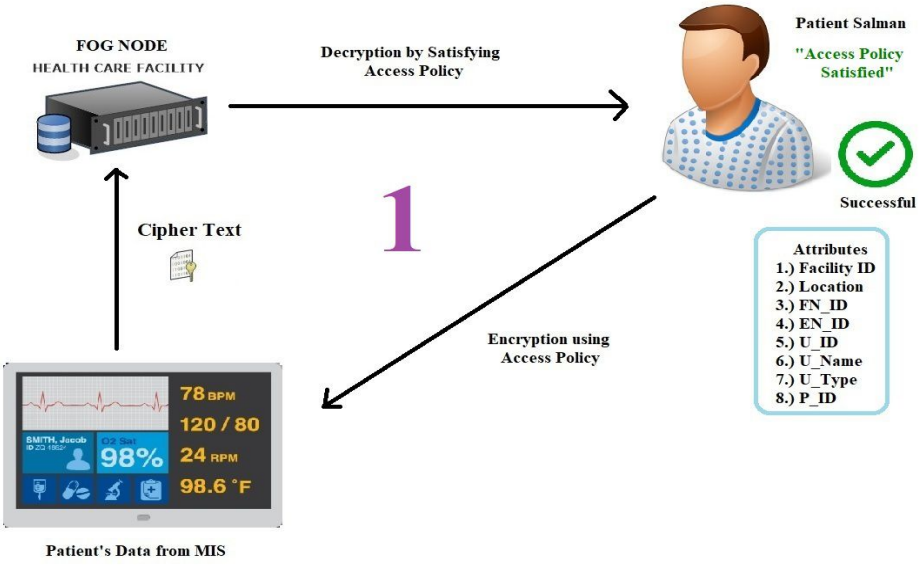


FIGURE 5.2: Patient wants to view its own data

5.2.2.2 Patient wants to view others patient data

In our data which is shown in table 5.1 there are three patients namely Salman, Ali and Mahmood. Salman is now interested to view another patient's data namely Mahmood that is generated by different sensors attached to his body. The data is encrypted by Mahmood using an access policy and the data is now available in fog server. Moreover, Mahmood's attributes are F.ID (Facility), Location, FN.ID (Fog Node), EN.ID (Edge Node), U.ID (User), U.Name, U.Type, P.ID(Patient). Some of these attributes might have different values. As Mahmood is patient different patient, Salman cannot decrypt and access his data as his access policy is not be satisfied with the attributes and its corresponding values.

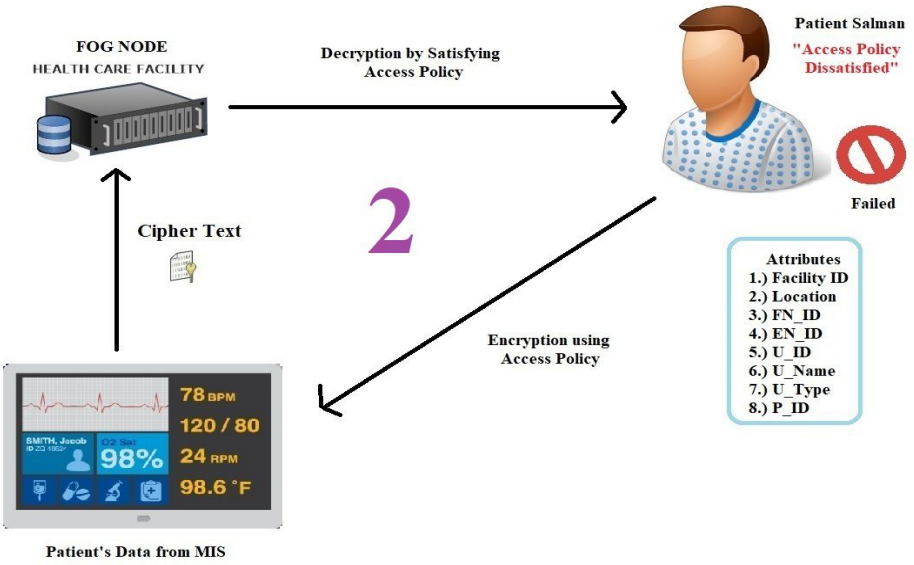


FIGURE 5.3: Patient wants to view others patient data

5.2.2.3 Relative want to view His/Her Patient Data

In our data which is shown in table 5.1 there are three relatives namely Qamar, Nayyer and Qadir. Qamar is the relative of patient Salman. He wants to view his data that is generated by different sensors attached to his body. The data is encrypted by salman using an access policy and the data is now available in fog server. Moreover, salman's attributes are F.ID (Facility), Location, FN.ID (Fog Node), EN.ID (Edge Node), U.ID (User), U.Name, U.Type, P.ID(Patient). Some of these attributes might have different values. As Qamar is the relative of Salman, he will easily decrypt and access his data as his access policy is satisfied with the attributes and its corresponding values.

5.2.2.4 Relative want to view Others Patient Data

In our data which is shown in table 5.1 there are three relatives namely Qamar, Nayyer and Qadir. Qamar is the relative of patient Salman. He now wants to view data of patient Ali that is generated by different sensors attached to his body. The data is encrypted by Ali using an access policy and the data is now available in fog server. Moreover, Ali's attributes are F.ID (Facility), Location, FN.ID (Fog Node), EN.ID (Edge Node), U.ID (User), U.Name, U.Type, P.ID(Patient). Some

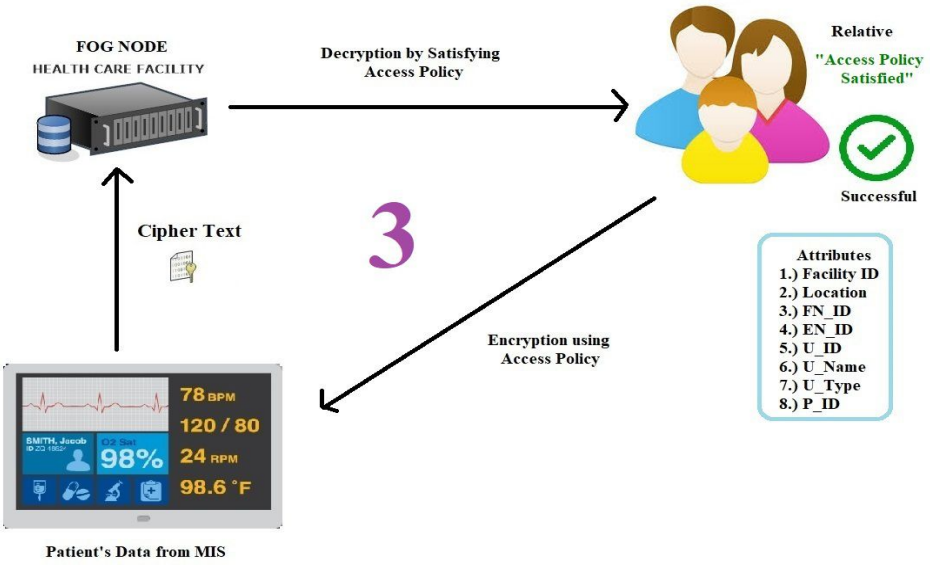


FIGURE 5.4: Relative want to view His/Her Patient Data

of these attributes might have different values. As Qamar is the relative of Salman, he cannot decrypt and access Ali's data as his access policy is not be satisfied with the attributes and its corresponding values.

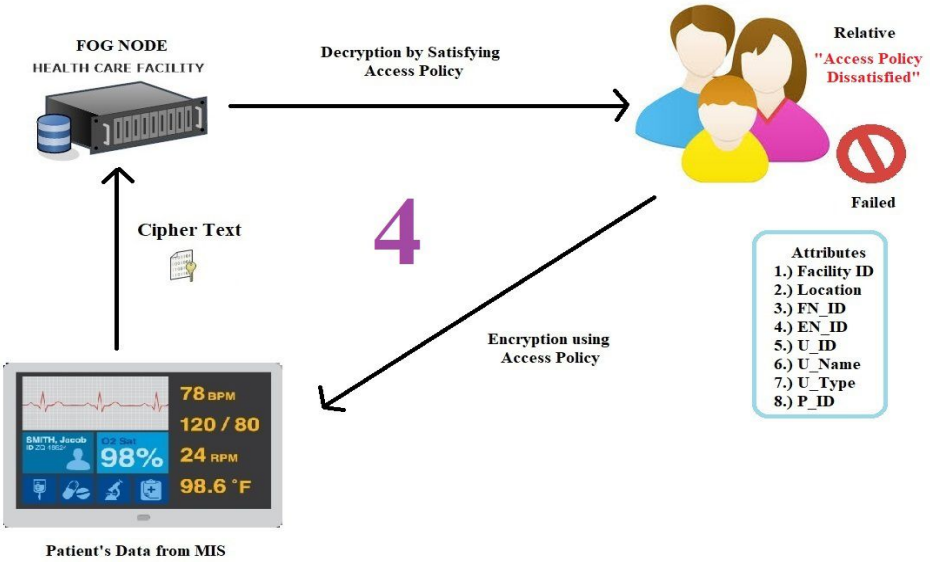


FIGURE 5.5: Relative want to view Others Patient Data

5.2.2.5 Paramedic/Doctor wants to view Patients Data in Same Facility

In our data which is shown in table 5.1 there are three paramedic/ doctor staff namely Annum, Nadeem and Amir. Annum wanted to view patient's data several times a day that is generated by multiple patients using different sensors attached to their body. The data is encrypted by patient themself using an access policy and the data is now available in fog server. Moreover, patient's attributes are F.ID (Facility), Location, FN.ID (Fog Node), EN.ID (Edge Node), U.ID (User), U.Name, U.Type, P.ID (Patient). All of the patients as well as the paramedic/-doctors are connected to the same facility/fog node. The paramedic/doctor staff will easily decrypt and access all patient's data as their access policy is satisfied with the attributes and its corresponding values.

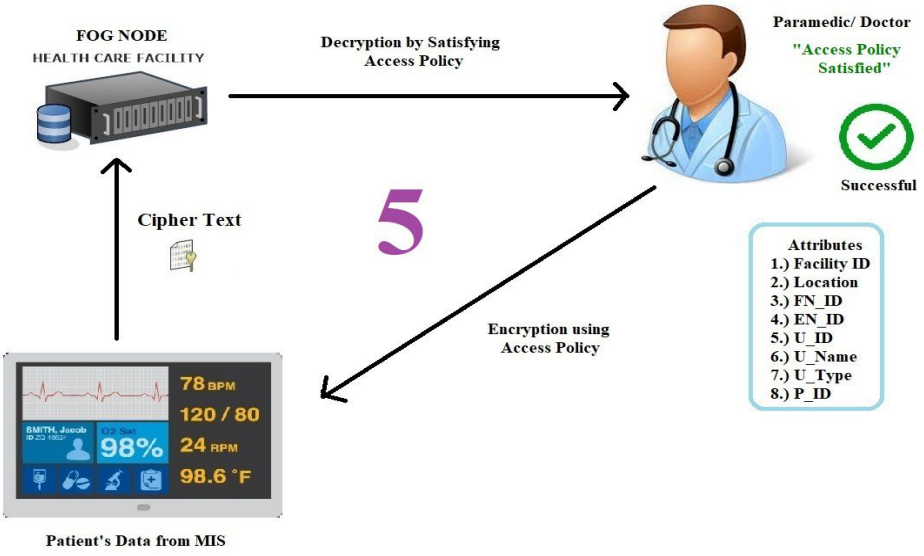


FIGURE 5.6: Paramedic/Doctor wants to view Patients Data in Same Facility

5.2.2.6 Paramedic/Doctor wants to view Patients Data in Different Facility

In our data which is shown in table 5.1 there are three paramedic/ doctor staff namely Annum, Nadeem and Amir. Annum wanted to view patient's data several times a day that is generated by multiple patients using different sensors attached

to their body. The data is encrypted by patient themself using an access policy and the data is now available in fog server. Moreover, patient's attributes are F.ID (Facility), Location, FN.ID (Fog Node), EN.ID (Edge Node), U.ID (User), U.Name, U.Type, P.ID (Patient). All of the patients as well as the paramedic/-doctors are connected to the different facility/fog node. The paramedic/doctor staff cannot decrypt and access all patient's data as their access policy is not be satisfied with the attributes and its corresponding values.

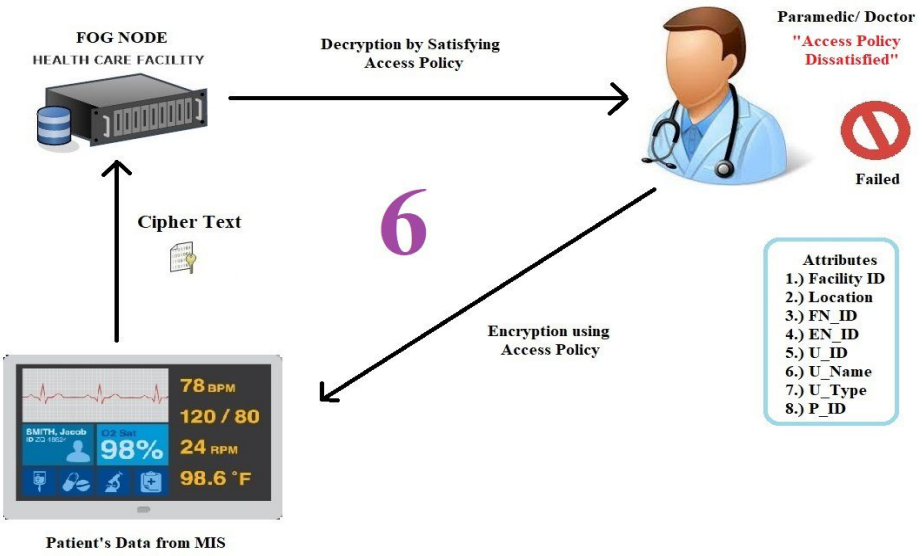


FIGURE 5.7: Paramedic/Doctor wants to view Patients Data in Different Facility

5.3 Security Requirements

This section discusses the issues that could arise while employing CP-ABE and explores the IoMT environment's security in the cloud. The important security factors [48] that must be addressed are listed below:

- Collusion attacks: Through mutual collusion, users can infer the qualities of other users and utilise the learned attributes to construct the secret key of another user. As a result, in addition to using the user's attributes, an AA must also apply other variables while generating a secret key. Additionally,

users may divulge information by conspiring with service providers. Therefore, data security technology is necessary so that only authorised users may decrypt and see it, even if data is leaked through a collusion assault.

- Unauthorized user access control: Since the cloud is a widely accessible public environment, anyone can access the data that is stored there, posing a number of security risks. Access control technology and security technologies are thus necessary for accessing stored data. Only users with the qualities that the data owner has previously defined can access the stored data if attribute-based encryption is employed among security solutions. As a result, attribute-based encryption must be used in order to ensure the data integrity and confidentiality.

- Tracking users through a distributed key: Using the fundamental CP-ABE strategy, it is difficult to pinpoint the user who initially received a distributed key because there is no data that can pinpoint the person who received a key. In the case that the key is misused, investigation is necessary to establish the authenticity of the individual that issued it. Through the use of a tracking system, this is possible.

- User privacy protection: Attribute-based encryption safeguards privacy since data owners and users both encrypt and decrypt information based on their own attributes. However, if the attribute validation agency issues the key, the user's privacy can be infringed by giving the AA access to the user's identification value in order to provide traceability. Therefore, research is necessary to protect users' confidentiality in the cloud.

- Verify data integrity as uploaded by data owner: In current CP-ABE methods, if a user accesses the encrypted message that data owner has transmitted and decrypt the data it to get the message, it is assumed that the message is authentic. Since the outsourcing server is believed to be trustworthy in the schemes, it is further assumed that the message produced by the partially decrypted ciphertext is legitimate. Nevertheless, these presumptions are true. Because the message sent to the web can be faked, it is unknown if

the value computed by the outsourced server, is the right value. Therefore, it is necessary to verify that the user's final decrypt value matches the original message of the data owner.

- Efficiency: In several of the known CP-ABE algorithms, the size of an encrypted message is proportional to the amount of attributes provided in the access control mechanism [49]. As a reason, the ciphertext's size increases linearly as the number of features increases, requiring costly cloud storage space. The amount of processing required by the decrypting user is similarly increased by the length of the ciphertext. To solve this issue, a server that supports outsourcing and can manage a part of the computational load must be implemented. It is also necessary to look into ways to reduce burden and computation requirements.

5.4 Conclusion

In this chapter, we have first considered a scenario of a hospital in which data owner will use our defined policies to secure its data which is being generated through different sensors attached to the patient's body. Data user can only decrypt that sensitive data if and only if it satisfies our defined policy. For said reason, we have described six '6' different test cases in which '3' test cases satisfy the policy whereas rest of the test cases are related to those policies which cannot be satisfied by the users. Then we have described the security requirements for CP-ABE mainly for our scenario. At the end, we have evaluated CP-ABE security and concluded that it is secured against Chosen Ciphertext Attack.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Patient sensitive data protection is very challenging task because of security concern. From the comprehensive review of literature, we have studied currently which techniques have been introduced. Finding new technique was also very challenging. We have found that CP-ABE is the good choice for securing patients data in SAFE-RH project. CP-ABE for Healthcare Data in Fog Environment is implemented on patient data and effects are analysed. Following advantages are achieved from suggested techniques are:

- Data is secured by using light-weight CP-ABE technique
- Additionally, encryption time is reduced using resource constraint devices
- Confidentiality level is increase on fog computing
- Patients gain confidence
- Cost effective

6.2 Future Work

Below targets will be our future work:

- Keys generation from multiple key authority center.
- More secured algorithms will be added for our proposed scenario
- Speed of encryption will be increased.
- Data will be fetched from multiple facilities and multiple fog nodes.
- Implementation of different flavors of CP-ABE for the SAFE-RH project.

Bibliography

[1] K. Stine and Q. Dang, "Encryption basics," *Journal of AHIMA*, vol. 82, no. 5, pp. 44–46, 2011.

[2] "https://www.ssl2buy.com/wiki/symmetric-vs-asymmetric-encryption-what-are-differences." (Date last accessed 14-December-2021).

[3] "https://www.kaspersky.com/resource-center/definitions/encryption." (Date last accessed 14-December-2021).

[4] A. Mittal, "Attribute based encryption for secure data access in cloud," 2017.

[5] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007 IEEE symposium on security and privacy (SP'07)*, pp. 321–334, IEEE, 2007.

[6] Y. Zhang, D. Zheng, and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2130–2145, 2018.

[7] "https://www.investopedia.com/terms/c/cloud-computing.asp." (Date last accessed 14-December-2021).

[8] "https://learn.saylor.org/mod/page/view.php?id=27582forceview=1." (Date last accessed 14-December-2021).

[9] "https://www.techtarget.com/searchdatacenter/definition/edge-computing." (Date last accessed 14-December-2021).

[10] Y. Ai, M. Peng, and K. Zhang, "Edge computing technologies for internet of things: a primer," *Digital Communications and Networks*, vol. 4, no. 2, pp. 77–86, 2018.

[11] M. Satyanarayanan, Z. Chen, K. Ha, W. Hu, W. Richter, and P. Pillai, "Cloudlets: at the leading edge of mobile-cloud convergence," in *6th international conference on mobile computing, applications and services*, pp. 1–9, IEEE, 2014.

[12] "https://www.macrometa.com/topics/edge-computing-vs-cloud-computing." (Date last accessed 14-December-2021).

[13] "https://www.techtarget.com/iotagenda/feature/fog-nodes-simplify-edge-vs-cloud-computing-choice." (Date last accessed 14-December-2021.

[14] "https://safe-rh.eu/." (Date last accessed 14-December-2021).

[15] M. R. Kumar, M. D. Fathima, and M. Mahendran, "Personal health data storage protection on cloud using ma-abe," *International Journal of Computer Applications*, vol. 75, no. 8, 2013.

[16] Y. S. Rao, "A secure and efficient ciphertext-policy attribute-based signcryption for personal health records sharing in cloud computing," *Future Generation Computer Systems*, vol. 67, pp. 133–151, 2017.

[17] P. G. Shynu and K. J. Singh, "An enhanced abe based secure access control scheme for e-health clouds," *International Journal of Intelligent Engineering and Systems*, vol. 10, no. 5, pp. 29–37, 2017.

[18] K. P. Kibiwott, F. Zhang, A. A. Omala, and D. Adu-Gyamfi, "Secure cloudlet-based ehealth big data system with fine-grained access control and outsourcing decryption from abe.," *Int. J. Netw. Secur.*, vol. 20, no. 6, pp. 1149–1162, 2018.

[19] Y. Zhang, D. Zheng, and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2130–2145, 2018.

[20] A. Pandey and G. Prakash, "Deduplication with attribute based encryption in e-health care systems," *International Journal of MC Square Scientific Research*, vol. 11, no. 4, pp. 16–24, 2019.

[21] J. Sun, H. Xiong, X. Liu, Y. Zhang, X. Nie, and R. H. Deng, "Lightweight and privacy-aware fine-grained access control for iot-oriented smart health," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6566–6575, 2020.

[22] K. Edemacu, B. Jang, and J. W. Kim, "Cescr: Cp-abe for efficient and secure sharing of data in collaborative ehealth with revocation and no dummy attribute," *PloS one*, vol. 16, no. 5, p. e0250992, 2021.

[23] H. Zhong, Y. Zhou, Q. Zhang, Y. Xu, and J. Cui, "An efficient and outsourcing-supported attribute-based access control scheme for edge-enabled smart healthcare," *Future Generation Computer Systems*, vol. 115, pp. 486–496, 2021.

[24] H.-Y. Lin and Y.-R. Jiang, "A multi-user ciphertext policy attribute-based encryption scheme with keyword search for medical cloud system," *Applied Sciences*, vol. 11, no. 1, p. 63, 2020.

[25] R. Nidhya, S. Shanthi, and M. Kumar, "A novel encryption design for wireless body area network in remote healthcare system using enhanced rsa algorithm," in *Intelligent system design*, pp. 255–263, Springer, 2021.

[26] W. Zhang, Z. Zhang, H. Xiong, and Z. Qin, "Phas-hekr-cp-abe: partially policy-hidden cp-abe with highly efficient key revocation in cloud data sharing system," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 1, pp. 613–627, 2022.

[27] K. Sowjanya, M. Dasgupta, and S. Ray, "A lightweight key management scheme for key-escrow-free ecc-based cp-abe for iot healthcare systems," *Journal of Systems Architecture*, vol. 117, p. 102108, 2021.

[28] N. Saravanan and A. Umamakeswari, "Hap-cp-abe based encryption technique with hashed access policy based authentication scheme for privacy preserving of phr," *Microprocessors and Microsystems*, vol. 80, p. 103540, 2021.

[29] Z. Zhang, W. Zhang, H. Zhuang, Y. Sun, and Z. Qin, "Efficient partially policy-hidden cp-abe for iot assisted smart health," in *International Conference on Artificial Intelligence and Security*, pp. 619–636, Springer, 2021.

[30] J. Hassan, D. Shehzad, I. Ullah, F. Algarni, M. U. Aftab, M. Asghar Khan, and M. I. Uddin, "A lightweight proxy re-encryption approach with certificate-based and incremental cryptography for fog-enabled e-healthcare," *Security and Communication Networks*, vol. 2021, 2021.

[31] S. Fugkeaw, "A lightweight policy update scheme for outsourced personal health records sharing," *IEEE Access*, vol. 9, pp. 54862–54871, 2021.

[32] G. Sandhia and S. K. Raja, "Secure sharing of data in cloud using ma-cpabe with elliptic curve cryptography," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 8, pp. 3893–3902, 2022.

[33] R. V. Rao, J. K. Raja, C. R. Raman, and P. N. Kishore, "Attribute based encrypted and secured cloud based personal health record system," *European Journal of Molecular & Clinical Medicine*, vol. 8, no. 2, pp. 1501–1507, 2021.

[34] H. Pillai, "Secure health care based encryption schemes on cloud computing storage," 2021.

[35] Z. Zhang, W. Zhang, and Z. Qin, "A partially hidden policy cp-abe scheme against attribute values guessing attacks with online privacy-protective decryption testing in iot assisted cloud computing," *Future Generation Computer Systems*, vol. 123, pp. 181–195, 2021.

[36] R. Cheng, K. Wu, Y. Su, W. Li, W. Cui, and J. Tong, "An efficient ecc-based cp-abe scheme for power iot," *Processes*, vol. 9, no. 7, p. 1176, 2021.

[37] N. Attrapadung, J. Herranz, F. Laguillaumie, B. Libert, E. De Panafieu, and C. Ràfols, "Attribute-based encryption schemes with constant-size ciphertexts," *Theoretical computer science*, vol. 422, pp. 15–38, 2012.

[38] A. Saidi, O. Nouali, and A. Amira, "Share-abe: an efficient and secure data sharing framework based on ciphertext-policy attribute-based encryption and fog computing," *Cluster Computing*, vol. 25, no. 1, pp. 167–185, 2022.

[39] H. Zhong, Y. Zhou, Q. Zhang, Y. Xu, and J. Cui, "An efficient and outsourcing-supported attribute-based access control scheme for edge-enabled smart healthcare," *Future Generation Computer Systems*, vol. 115, pp. 486–496, 2021.

[40] A. Balu and K. Kuppusamy, "Ciphertext policy attribute based encryption with anonymous access policy," *arXiv preprint arXiv:1011.0527*, 2010.

[41] Z. Zhou and D. Huang, "On efficient ciphertext-policy attribute based encryption and broadcast encryption," in *Proceedings of the 17th ACM conference on Computer and communications security*, pp. 753–755, 2010.

[42] H. Zhong, Y. Zhou, Q. Zhang, Y. Xu, and J. Cui, "An efficient and outsourcing-supported attribute-based access control scheme for edge-enabled smart healthcare," *Future Generation Computer Systems*, vol. 115, pp. 486–496, 2021.

[43] A. Balu and K. Kuppusamy, "Ciphertext policy attribute based encryption with anonymous access policy," *arXiv preprint arXiv:1011.0527*, 2010.

[44] "https://github.com/zeutro/openabe." (Date last accessed 8-February-2022).

[45] "https://crypto.stanford.edu/pbc/." (Date last accessed 8-February-2022).

[46] "https://www.gnu.org/software/libc/." (Date last accessed 8-February-2022).

[47] J. A. Akinyele, C. Garman, I. Miers, M. W. Pagano, M. Rushanan, M. Green, and A. D. Rubin, "Charm: a framework for rapidly prototyping cryptosystems," *Journal of Cryptographic Engineering*, vol. 3, no. 2, pp. 111–128, 2013.

[48] "https://www.ncbi.nlm.nih.gov/pmc/articles/pmc7506716/." (Date last accessed 9-February-2022).

[49] S. Ding, C. Li, and H. Li, "A novel efficient pairing-free cp-abe based on elliptic curve cryptography for iot," *IEEE Access*, vol. 6, pp. 27336–27345, 2018.