New Hidden Policy CP-ABE for Big Data Access Control with Privacy-preserving Policy in Cloud Computing

A Dissertation Submitted to the University of Hyderabad
in Partial Fulfillment of the Degree of

Master of Technology
in
Information Technology

by

Sucharita Khuntia

16mcmb15

School of Computer and Information Sciences
University of Hyderabad
Gachibowli, Hyderabad - 500 046
Telangana, India

August 2, 2018

CERTIFICATE

This is to certify that the dissertation titled, "**New Hidden Policy CP-ABE for Big Data Access Control with Privacy-preserving Policy in Cloud Computing**" submitted by **Sucharita Khuntia** , bearing Regd. No. **16mcmb15**, in partial fulfillment of the requirements for the award of Master of Technology in Information Technology is a bonafide work carried out by her under my supervision and guidance.

The dissertation has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Dr. P. Syam Kumar
Project Supervisor
Institute for Development and
Research in Banking Technology,
Hyderabad

Dr. Arun Agarwal
Dean
School of Computer and
Information Sciences
University of Hyderabad.

DECLARATION

I, **Sucharita Khuntia** , hereby declare that this dissertation titled, "**New Hidden Policy CP-ABE for Big Data Access Control with Privacy-preserving Policy in Cloud Computing**", submitted by me under the guidance and supervision of Dr. P. Syam Kumar is a bonafide work which is also free from plagiarism. I also declare that it has not been submitted previously in part or in full to this University or other University or Institution for the award of any degree or diploma. I hereby agree that my dissertation can be deposited in Shodganga/INFLIBNET.

A report on plagiarism statistics from the University Librarian is enclosed.

Date:

Name: Sucharita Khuntia

Regd. No. 16mcmb15

Signature of the student

//Countersigned//

Signature of the Supervisor:

(Dr. P. Syam Kumar)

Acknowledgement

I take this opportunity to express my sincere thanks to Dr.P.Syam Kumar, for his valuable suggestions, guidance, motivation throughout the year of my project. His constant guidance helped me to a lot to develop the progress of my project work.

I am conveying my extreme gratitude to Dr. A. S. Ramasastry Director of IDRBT for providing me infrastructural facilities to work such an environment to exploring my knowledge.

I also wish to extend my sincere thanks to Dr. Arun Agarwal, Dean of SCIS, University of Hyderabad, FOR providing facilities to accomplish my project successfully.

Special thanks to IDRBT faculties for their constant support and encouragement during project work. I would like to thank to University of Hyderabad, for giving me opportunity to do M.Tech course.

Last but not least, I wish to convey my heartfull thanks to my beloved parents and my dear friends for their gracious solicitude, which helped me to accomplish my project successfully.

With sincere regards

Sucharita Khuntia

Publication

1. Sucharita Khuntia, P. Syam Kumar, "New Hidden Policy CP-ABE for Big Data Access Control with Privacy-preserving Policy in Cloud Computing " , Accepted for IEEE Explore, 9th International Conference on Computing Communication and Networking Technologies(ICCCNT-2018), July 10-12, 2018, IISC Bengaluru, India.

Abstract

Cloud offers flexible and cost effective storage for big data but the major challenge is access control of big data processing. CP-ABE is a desirable solution for data access control in cloud, but CP-ABE has some drawbacks that in CP-ABE, the access policy may leak users private information as the access policy attached with ciphertext with plain text format. So, there is possibility that adversary can obtain user's personal information. To address this issue, several Hidden Policy CP-ABE schemes has been proposed but those schemes still causing data leakage problem because the access policies are partially hidden and create more computational cost.

→ In this project, we propose a New Hidden Policy Ciphertext Policy Attribute Based Encryption (HP-CP-ABE) to ensure Big Data Access Control with Privacy-preserving Policy in Cloud. In our method, We applied mask technique on each attribute in access policy and embed the access policy in ciphertext to protect users private information from access policy. We used Multi Secret Sharing Scheme(MSSS) to reduce the computational overhead, while encryption and decryption process. Thus, our scheme provides fully hidden policy and preserves privacy from access policy. The security analysis shows that HP-CP-ABE is more secure and preserve the access policy privacy. Performance evaluation shows that our schemes takes less computational cost for encryption and decryption than other existing schemes.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Big Data

A large amount of exponentially increasing data sets refereed as Big Data. In daily life data generated from different sources like social media, sensor devices, machines, digital videos, pictures etc. Big data is a large volume of data collected to store and analyze for the benefit of the organization to make better decision making and increase efficiency. The voluminous Big Data also refereed the collection of structured and unstructured data. Big data sets are voluminous with complex in nature. These data set analyzed and computed according to human behavior, interactions to reveal the associations among data sets. Big data technology makes economically and technically feasible for big data management to store and collect large data sets along with analyzing them to get the valuable sagacity between data sets. This big data term frequently used to describe the procedure of serious computing power that is trending in machine learning and artificial intelligence. Presently big data used technologies like Hadoop, NoSQL, Mongo, Cassandra etc. Big data is not only about analytics, it is about the driving experience for customers to develop a real-time application.

1.1.1 Big Data Characteristics

Before data scientists described big data with three characteristics, volume, velocity, and veracity, later more Vs added to the list that is variety and value. Now Big data characteristics referred with 5Vs. Big data characteristics described as follows:

- Volume: The huge amount of data generated in each and every second defines a volume. Big data itself understand the large volume of data. These data created from different sources like social media platforms, machines and sensor networks etc and that data analyzed in massive. It is difficult for huge amounts of data process in the traditional database system, storing of huge data also difficult in the traditional database.

- Velocity: Velocity of big data is the speed at which data generated from different sources like social media, machine, sensor networks etc and collects and analyze. Big data technology can analyze data at the time of generation, not putting the data into a database.

- Veracity: Veracity says the quality and accuracy of data. This referred to as the abnormality, noise in data. This huge amount of data have less control over accuracy and quality. For example just take one scenario, the hashtags, abbreviations etc, collecting a large amount of data no use if there is no accuracy. Now big data technology allow us to work with these type of data. Many times, the large volume makes lack of accuracy or quality.

- Variety: It refers to the different type of data can be used. Now, we are no longer using only structured data like phone no, contact information, name, address etc, which can fit into tables or traditional database. Now a day's world full of unstructured data like digital images and video, text, voice data etc.

- Value: Accessing a large amount of data no use unless we can be turned it into value. The important part is to understand the

actual cost and benefit of collected and analyzed data that ensures ultimately the data, which is gather can be converted to the desired value.

1.1.2 Benefits and Issues

Big data has lots of benefits in the Business world. It does not bother that how much a company collecting data, it is important the way data used. Increasing efficiency depends on the way data used efficiently.
Benefits:

- **Cost Reducing: Big data tools like Hadoop and some cloud-based analytics. These tools bring advantage in cost while storing a large amount of data and with the help of these tools companies can identifying the efficient ways for business.**
- **Time Reductions: Because of using big data tools, it's easy to know data sources. That helps a business organization to analyze data and to take quick decision.**
- **New Product Development: You can know the customer needs and satisfaction by using Big data analytics. So that, products can be created according to users need.**
- **Understand the market conditions: By analyzing on big data, you can get the current market condition. Like analyzing customer's purchase transaction, a company can know, which products having trends and according to that they can plan and can get ahead of competitors.**
- **Control online reputation: In Big data analytics, you can get feedback that that who is saying what about your company. Big data tools can help to improve your business presence in online through sentiment analysis.**

Issues:

- **Big data itself understands the huge amount of data. Traditional database systems do not enough huge data set processing and also has a problem in storing.**
- **One measure issue in big data is the privacy breach. Where data can be accessible to unauthorized peoples. Weather, it can be done mistakenly or intentionally.**
- **Big data does not ensure 100 percent level of accuracy analysis. Because of the involvement of a large amount of data, checking analysis manually not possible. So, can ensure that your data analytics can not give inaccurate data for that can use analytics tools, which are most trusted and can ensure best level accuracy.**
- **With big data now E-discovery becomes expensive. E-discovery is the search for electronic data for the use in legal activities like Court, Government orders for E-discovery for critical evidence. But because of lots of data, now it is difficult to search digital evidence. Hence difficult to take action with legal restriction.**

1.2 Cloud Computing

Cloud Computing deliveries different resources and services such as applications, storage, network over the internet. In cloud computing, users can access resources and services from anywhere and any time. Delivery of services and resources cost relies on pay per use basis. Vitalization is related closely to cloud computing improvements according to NIST(National Institute of Standards Technology) cloud model consist of five essential characteristics,three service models and four deployment models.

1.2.1 Essential Characteristics

1. **On demand self service: In cloud computing, whatever consumer needed either server time or network storage, they have the capa-**

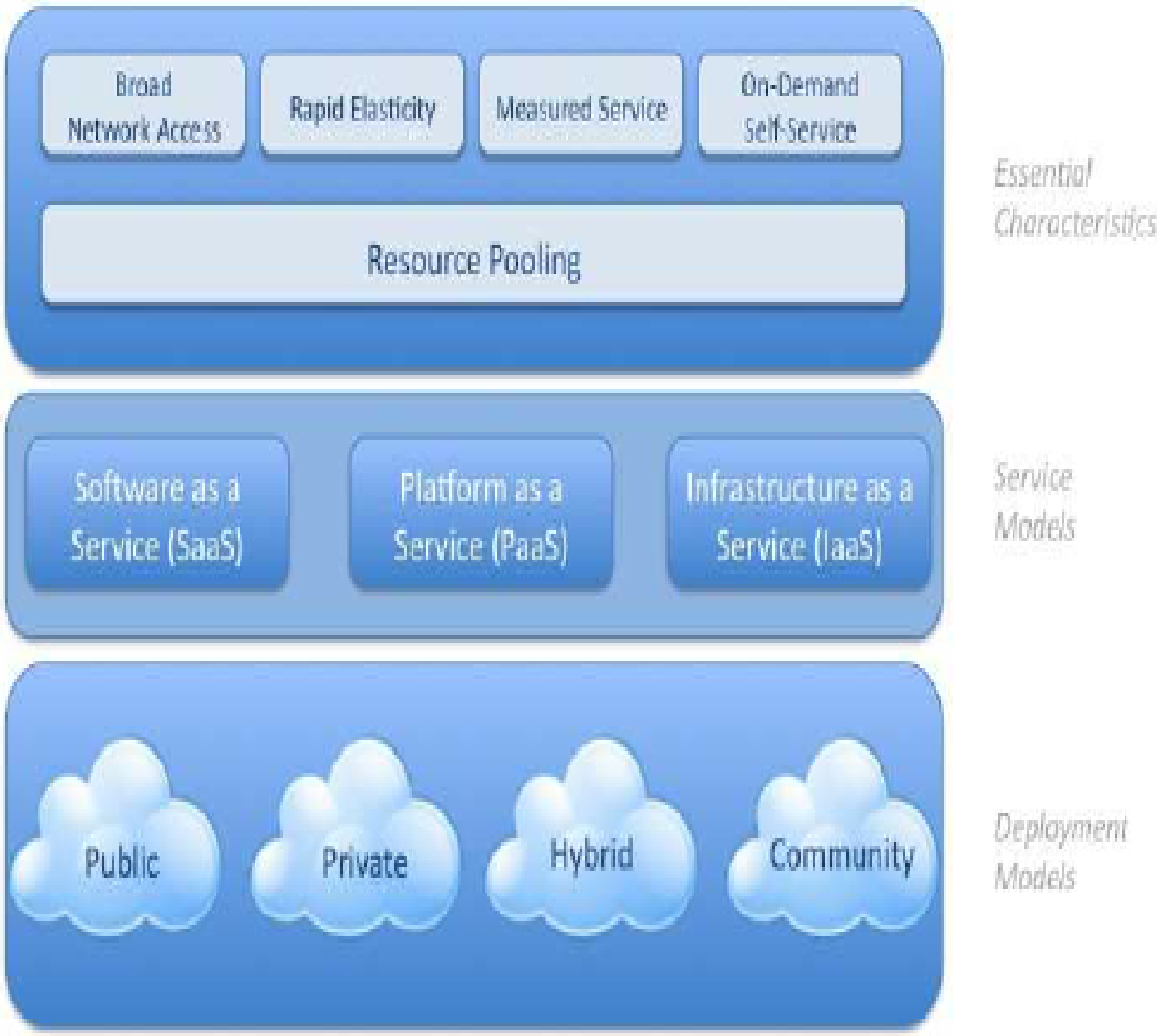

Figure 1.1: Cloud Computing Architecture

bilities to get it without their interaction needed with the service provider.

2. **Broad network access: Cloud Computing has broad network access that consumer can access cloud by thin or thick platforms for Ex. different android phones, tablets, PCs etc.**

3. **Resource pooling: Cloud Computing has characteristic of resource pooling, where the provided computing resources pools together. So that, multiple consumers can use the resource through different physical and virtual assigned resources based on the consumer's demand.**

4. **Rapid elasticity: Cloud computing has characteristics of scale up and scales down. The resources can be scale up or scale down rapidly based on consumer demand. There are capabilities that consumers can get quantity needed resources and services at any point of time.**

5. **Pay per use: Cloud system follows measured service that the resources used like electricity. Cloud can monitor resource usage, and according to the usage, the consumer pays. Hence, the use of the resource can be a monitor. Control and the report provide transparency for both consumer and provider.**

1.2.2 Service Models

1. **Software as a Service: Saas provides the facility for consumers to run the application of provider on a cloud platform. Users can use these applications at any time from anywhere through the different interface of clients like the web browser, UPI etc.**

2. **Platform as a Service: PaaS has the facility for consumers can build, test or deploy an application on cloud infrastructure. Consumer need not manage or control network, server, storage and operating system, which are underlined in cloud infrastructure.**

Still, the consumers have control over their application. Also, consumers have control over the configuration of settings of their application environment.

3. **Infrastructure as a Service: IaaS gives facility to users for storage networks along with resources. So that consumer can able for deployment and running the software, which also includes applications along with OS.**

1.2.3 Deployment Models

1. **Public Cloud: The General public can access the public cloud. This cloud environment may be managed, operates by any third party like an educational academy, business corporate or government organization or combination of them. Public cloud IT resources provides to consumers with chipper cost. Public cloud placed a cloud provider on premises.**

2. **Private cloud: Private cloud exclusively used by only one organization. This cloud manages, operates and owns by only one organization. The IT resources of private cloud can be accessible by different departments, parts of the organization. This cloud may be placed on or off premises.**

3. **Hybrid cloud: This cloud is a combination of more than one cloud infrastructure. Like the mix of private and public or private and community etc. Although, every cloud have their own qualities mix of both, provides portability of data and application.**

4. **Community cloud: This cloud is similar to a public cloud but it has limited access to an only particular community of an organization. This cloud environment manages, operates and owns by more than one organization belongs to the same community. It may be placed on or off premises. Membership in the community does not ensure**

that can control all IT cloud resources. Outside parties not granted access unless allowed by the community.

1.3 Benefits and Challenges

Benefits:

1. **Flexibility: Cloud can scale up and scale down depending on the user's need. It gives opportunities for users to choose different cloud infrastructure according to their need like if they concern about security or any other need.**

2. **Efficiency: Cloud-based applications and data can be accessed from anywhere through an Internet connection having the device. Cloud gives speed quite faster to access any application or data. As this uses remote recurses, so it saves the equipment and server cost. Also provides facility to pay for cloud services as per use.**

Challenges:

1. **Security: The main concern in cloud computing is the security concern. To protect the data from attackers and hackers because even if one data got affected by attacking them, other clients also got affected by that loss.**

2. **Interoperability and portability: Cloud computing has a drawback that when providers switching from on-premises IT to cloud infrastructure, there will be a problem of having the lock-in period. So the migration should be smooth when it is moving from on-premises IT to cloud infrastructure.**

1.4 Motivation

Due to the huge size of big data, the traditional database systems cannot handle the big data. Cloud is the perfect solution to store and

process of an extensive large amount of data because of its elasticity and flexibility. In the cloud computing environment, users can store and share their data with other users. User loses control over their data, and once they stored data in the cloud, then the access control scheme becomes challenging. CP-ABE provides efficient big data access control, where the user encrypted data under their defined access policy and stored the encrypted data along with access policy in the cloud. If the attributes satisfied by access policy of any user, then only the user can able to access data from the cloud. However, CP-ABE gives efficient access control mechanism for big data. But it has data leakage problem from access policy. If access policy attached to ciphertext in normal text format, then there is a possibility that attacker may obtain user's private information from access policy. In addressing this problem, there are several hidden policy schemes [5]-[19] has been proposed. In [5]-[17], where they partially hide attributes on access policies rather fully hide every attribute on access policies. The attribute has two-part, attribute value and attributes name. These schemes hide attribute values only, still the name of attribute visible for the public. Also, This every partially hidden policy scheme not gives full privacy, as well as many of those schemes, support only specific type access policy structure. Later, fully hidden access policy scheme [18]-[19] described, where access policies are fully hidden with using an Attribute-bloom filter, also used Attribute-localization algorithm still these schemes have the drawback of having more computation cost. Another fully hidden policy scheme is [19], in which the attribute is fully hidden but also has the problem of more computational cost.

1.5 Problem Definition

Our project proposes a new hidden policy CP-ABE to provide big data access control with privacy-preserving policy, which provides Privacy Preservation Policy along with efficient Big-data Access Control. Our

contribution summarized as follows:

- **Our contribution follows → First encryption carries by the user on data under CP-ABE with (t,n) threshold multi-secret sharing scheme with tree access policy. After encrypting the data, then we apply a mask technique to each attribute of the access policy for hiding attributes in the access policy, which embed with ciphertext.**
- **When the user wants for access data, user applied de-mask technique on masked access policy for getting the original attributes from access policies, then authorizes users can reconstruct the secret by combining their share.**
- **Security Analysis proves our method is fully hiding the access policies and provides privacy from access policies.**
- **Performance Analysis explains our scheme has efficiency than existing schemes.**

1.6 System Architecture

1.7 Framework of Proposed System

Proposed HP-CP-ABE framework has following :
SETUP $(1^{\alpha}) \rightarrow (PK, MK)$**: Input is security parameter** α**, outputs Public Key (**PK**) as well as Master Key (**MK**).**
KEYGEN $(PK, MK, A) \rightarrow SK$**: It has Master secret key(MK) , attributes set (**A**) as input, generates Secret Key(**SK**) associate with attributes set.**
ENCRYPTION $(PK, M, \tau i = 1..k) \rightarrow (CT)$**: Inputs of this algorithm are public key with message M along with access policy** $\tau_{i=1..k.}$ **outputs Ciphertext CT with masked policy** $m_{i=1..k}$ **,** k**= no of secrets.**
DECRYPTION $(CT, SK) \rightarrow M$**: Inputs are Secret key** SK**, Ciphertext** CT **with masked access policy** $m_{i=1..k}$ **and outputs plaintext** M **as output.**

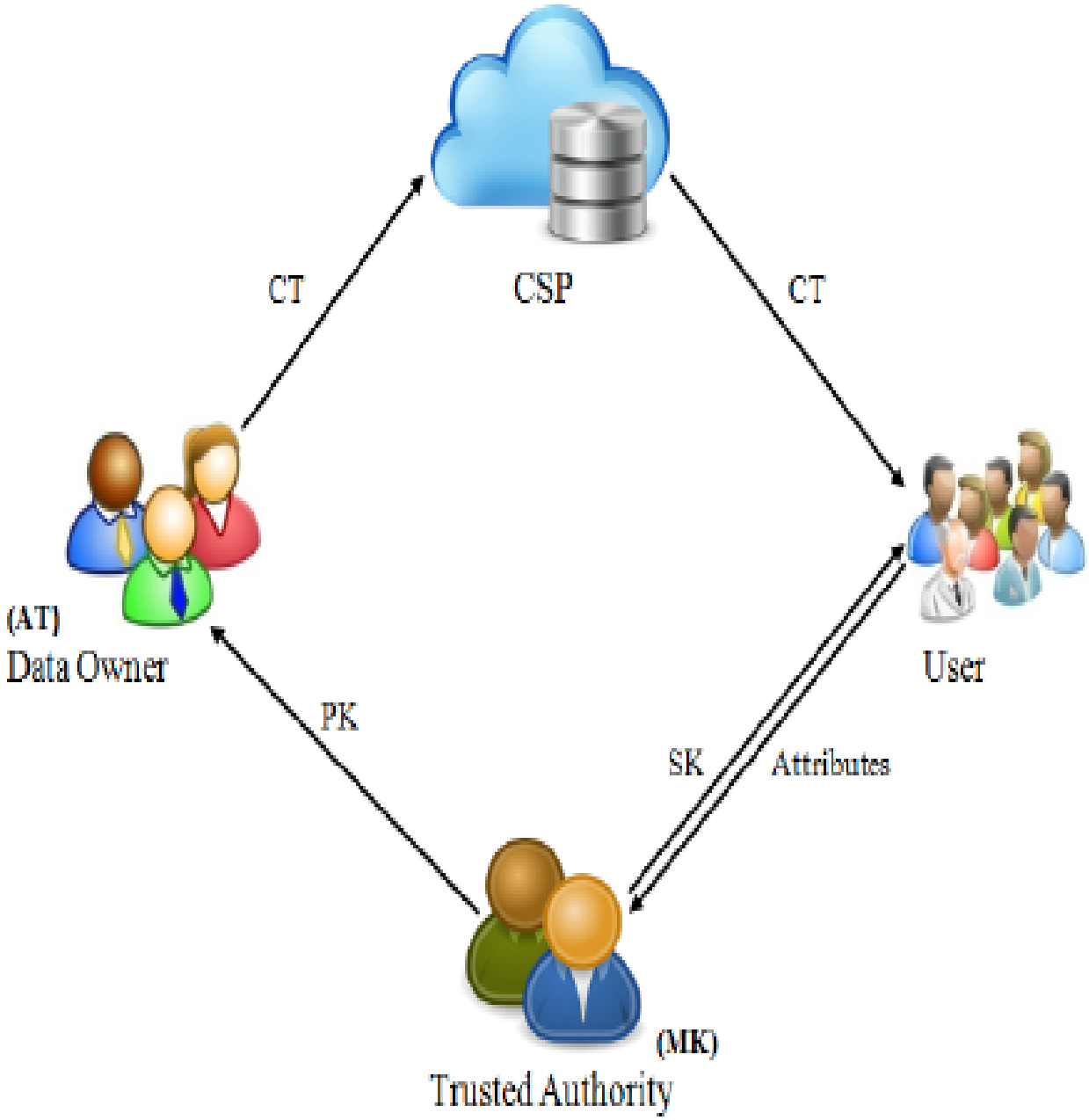

Figure 1.2: Proposed Scheme Architecture

1.8 Organization of the Thesis

The remaining organization of my thesis as follows:

In Chapter 2, we did Literature Survey on various Hidden Policy CP-ABE schemes.

In Chapter 3, we cover Preliminaries, which describes various tech-

niques, terms, and concepts used in our scheme.

In Chapter 4, the proposed construction of our scheme describes in this section. We give a description process of our idea in a details manner.

In Chapter 5, covers a brief description about Security analysis. We explain security game that shows, how our proposed system is secure against adversary attacks

In Chapter 6, we described a performance analysis along with an experimental result of our proposed scheme.

Finally, we conclude our work in chapter 7 with future work.

Chapter 2

Literature Survey

Recently, several Hidden-Policy CP-ABE schemes [5]-[19] has been proposed. We can consider details of these schemes with two type of hidden policy schemes. Some partially hidden policy schemes [5]-[17], that partially hides access structure in the ciphertext and few are fully hidden policy schemes [18]-[19], provides fully hidden access structure.

2.1 Partially Hidden Access Policy Schemes

The details description of partially hidden policy schemes [5]-[17] follows: Wang et al. [5] describes CP-ABE hidden-policy scheme, used bilinear groups of composite order with LSSS access structure. This scheme used LSSS access structure to realize access policy and groups that is bilinear in the nature of the composite order to hide the access policy. This scheme realizes water's CP-ABE scheme with composite order groups. This scheme used three groups in ciphertext and 2 groups in secret key. Ciphertext and access policy hide by group elements. The dual encryption mechanism provides security proof to this scheme and gives CPA security based on 4 static assumptions.

Lai et al. [6] described CP-ABE scheme for hidden policy, where the attribute value hides in access policy that only hides the attribute value but the attribute name and information about access structure are not hidden. This scheme used the method of adding the redundant com-

ponent in cipher-text so that user can know, which attribute satisfies the access policy through ciphertext redundant component and can be used LSSS access structure but along with access structure complexity of ciphertext size also increases. This model handles expressive access structure and provides security by using dual system encryption methodology.

Runhua et al. [7] defines a CP-ABE hidden-policy, used tree-based access structure for the greater expressive ability of access policy. This scheme has the capability to protect the access structure as well as policy. This paper proposed two models, 1st proposed a CP-ABE hidden policy model that is used tree-based access structure and also has a CPA attack under DBDH assumption. Still, this scheme has the disadvantage of data leakage from access policy.

Phuong et al. [8] proposed two CP-ABE scheme with privacy preserving. The design used AND-gate with a wildcard and inner product encryption where the first schemes have the drawback of not able to hide the access policy and can hide access policy against with legitimate decryptors. The construction of this paper used different symbol positions to find the matching among access policy and attributes. The main policy of our scheme to change the access policy of user attributes in 2 vectors after that applies inner product encryption to hide the access policy. But still, scheme suffering from data leakage of access policy.

In [9] Helil et al. described CP-ABE access control scheme with a hidden policy for sensitive data set constraint, which used additional artificial attributes to handle SDS constraints through additional entity participation. The design of this scheme used tree-based access structure and This scheme used duty principle, due to this principle the access policy and constraint policy divided into two different entities for the better security. This model also provides flexibility to data owners that they are able to change the sensitive data set. also suffers from data leakage from access policy.

Liu et al. [10] described CP-ABE has access structure that partially

hides. This method used that added redundantly to ciphertext that is the cause the access structure satisfied with private key attributes. The design of the scheme used a linear secret sharing scheme and dual system encryption methodology. The ciphertext size depends on the complexity of the access structure. This scheme provides more flexibility, expressiveness, and full security. This model also appropriate for EMR for privacy-preserving but has a disadvantage of hides attribute value in access policy and other information still public.

In [11] Li et al. describe one ABE for privacy-aware used tracing algorithm that has user accountability and also has security against collision attacks. By using black box model accountability of user can be obtained by adding particular user information to private key attributes respective to the user whom it issued. This scheme used wildcards that hides attribute value in the access policy. The model is probably secure but suffered from leaking data from access policy.

Lai et al. [12] defines CP-ABE scheme used inner product encryption for attribute hiding, which is fully secure and also can support access structure in a wide range like conjunctive normal form or disjunctive normal form but partially hides the access policy

Yu et al. [13] described a hidden policy scheme with content-distribution for SDNs that is in distributive in nature using cryptographic primitives and designed used attributes and implementation with CP-ABE and hierarchical key. This method provides flexible access control, thus only authorized user able to decrypt the contents. The protocols of this model designed the way to provide management efficiency of the key. This protocol follows that the user can keep track of

Nishide et al. [14] described a CP-ABE, that design using on the partial hidden-policy scheme used wildcard but has the disadvantage that defined access policy restricts with specific attributes. ciphertext reveal ciphertext policy.

Boneh et al. [15] described a hidden policy scheme, hidden-vector encryption only hides attribute values in the access policy. This scheme

selectively secure also generates ciphertexts that size is 0(n).

Katz et al. [16] defined CP-ABE scheme that provides hidden policy and used anonymous IBE and attributes hiding scheme of bilinear groups. Hidden vector encryption uses to hide attributes values on access policy.

Xu et al. [17] described a CP-ABE hidden policy, that used a tree-based access structure, which protects access policy privacy as well as provides flexible access control. All above-described hidden-policy schemes partial hide access policy, only hide attribute values but still attribute name not hidden. The below-described schemes support fully hidden access policy,

2.2 Fully Hidden Access Policy Schemes

Yang et al. [18] described a fully-hidden policy CP-ABE. scheme design used an attribute-bloom filter to provide fully-hidden attributes in access policy and LSSS. In access policy the whole attribute is hidden. The design of the scheme based on an attribute bloom filter. The attribute-bloom filter used to evaluate the attribute in the access policy or not and it also defines the proper location in access policy by using attribute localization algorithm. For the increase of efficiency in this model used attribute bloom filter, which shows in the access matrix the exact row number of an attribute in LSSS access structure. This scheme provides less overhead in computation and preserves privacy from LSSS access structure, the access policy does not leak user's private sensitive information. But the scheme has the disadvantage that the attribute bloom filter is inefficient and the attribute bloom filter takes more computational overhead and false positing.

Khan et al. [19] also described Hidden policy scheme which is expressive, which completely hides the access policy in the ciphertext. This scheme used LSSS access structure and Hidden Vector Encryption (HVE) to check the particular location of attributes in access policy.

In the idea of this scheme to not send the access matrix with the ciphertext. To finding the user attributes satisfies access policy or not, for that purpose Hidden Vector Encryption used to check the condition of a subset. The performance analysis of this scheme showed by charm simulator and also proves that under DBDH assumption and DLIN assumption the scheme selectively secure. Although, these schemes hide the whole attribute in access policy, still has the disadvantage of more computation cost.

Chapter 3

Preliminaries

Various techniques and concepts used in our project described below.

3.1 CP-ABE

CP-ABE is Public key Encryption type. CP-ABE has each private key of every user has attributes, which shows the permission of that user to access their data to the only authorized user. The user performs encryption as well as decryption on their data according to the user attribute. Data owners define the access policy for data, access policy shows with access-tree with an attribute. where user performing encryption with data with kept access policy. Then store the encrypted data in the cloud. Users, the access policy satisfied by which user, they only eligible for data decryption. For example: University Vice-Chancellor shares one file, which only the SCIS department faculties and administration staffs can access it. So, the University Vice-Chancellor defines the access policy for the file:
SCIS AND (Faculties OR Administration)
the access policy satisfied by which user attribute, that person only able to access the file or we can say, who belongs to SCIS department and they are either faculties of this department or they are administration staffs can able to access the file, which shared by University Vice-Chancellor. CP-ABE not requiring any storage like trusted authority

or any other storage. CP-ABE has 4 algorithms: Setup, Keygen, Encryption along with Decryption.

Setup: The security parameter is input for Setup algorithm, after taking input parameter public key along with master secret key generates as output. The Trusted Authority(TA) performs set up algorithm, then retains master key itself and passed the public key to the data owner. Data owner encrypts their data with the public key.

Keygen: Public key, attributes set and master secret key are the inputs in keygen algorithm. The combination of the inputs makes the result corresponding secret key and sends generated key to costumers corresponding to their attributes. Through the secret key, they able to decrypt the data.

Encryption: Encryption algorithm takes inputs are public key, plain text message, and access policy. The user performs encryption on their data then during encryption mask technique applied to access policy to hides the access policy. So, that it does not leak the user's private information. After performing encryption operation, it generates outputs as ciphertext with the access policy, that is in masked form. Then the ciphertext stored in cloud

Decryption: Decryption algorithm takes input as ciphertext with masked access policy and secret key. Decryption starts with applying de-mask operation on masked access policy which embeds with the ciphertext to get the original access policy. User's attributes satisfy by the original access policy, that user performs decryption operation and generates plain text message as output.

3.2 Access Structure

Our project used tree access structure. Tree access structure is defined access policy, which denotes with access tree. On that tree, threshold-gate like AND, OR defines the interior node and every leaf nodes have attributes. Tree access structure is more expressive, any access policy

can be realized with tree-based access structure. This access structure has the ability of more expressiveness.

Figure 3.1 denotes the access structure, where leaf nodes are attributes and root nodes are the threshold gate.

Example:

Attributes:

User 1: University, Administration

User 2: University, Faculty

User 3: University, Student

We can observe from Figure 3.1, the above User 2 and User 3 satisfies the access structure. So, only User 2 and User 3 can able decrypt the data that means the User 2 and User 3 only can access the file, User 1 can not able to access the file. That defined access policy represent Figure 3.1 access structure.

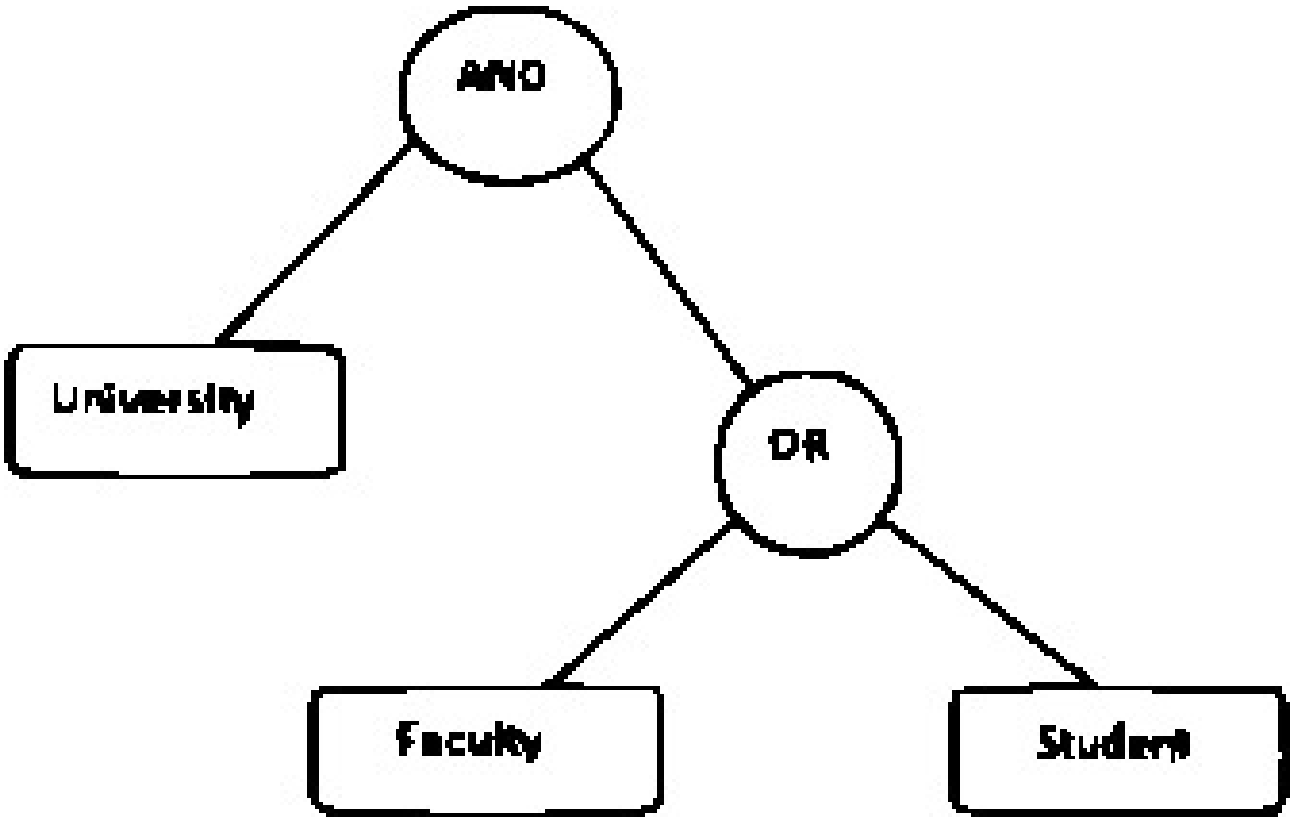

Figure 3.1: Access Structure

3.3 Bilinear Pairing

G_1 and G_2 two cyclic groups are multiplicative in nature groups has prime order that p with generator h of G_1. $e : G_1 * G_1 \rightarrow G_2$ having 3 property

Linearity→ $e(U^b, V^x) = e(U, V)^{bx}$ for all $U \in G_1$, $V \in G_2$ where $b, x \in Z_P$

Degeneracy: $e(h, h) \neq 1$

Computability: e is efficiently computes.

3.4 Decisional BDHE Assumption:

As per security parameter Γ, a cyclic group G_1 chooses. Which has a prime order p and generator h. $m, y \in Z_p$ chosen randomly. Let h^i denoted as h^{m^i} and $\vec{b} = (h, h_1 ... h_q, , h_{h+2}, ..., h_{2q}, h^y)$. From random element R in G_T, the adversary must have distinguish $e(h, h)^{m^{q+1}y} \in G_T$.

If BDHE problem solved by an algorithm B in:

$$|pr[B(\vec{b}, M = e(h, h)^{m^{q|1}y}) = 0] - pr[B(\vec{b}, M = R) = 0]| \geq \upsilon$$

For solving BDHE problem in group G_1, B has advantage υ. We can also say that, for solving BDHE assumption if no algorithm with polynomial time has a non-negligible advantage, then BDHE assumption holds.

3.5 Multi-Secret Sharing Scheme (MSSS)

Multi-secret sharing scheme is an extension of the secret-sharing scheme. The secret-sharing scheme has the process of one secret can be shared among a set of attributes (nodes) of the access tree. Dealer keeps private value to each node that is share. Secret recovers only if a subset of authorized attributes uses their share. This work only with a single secret. Every time system updates shares assign new share to every node whenever secret changes to the new secret. Multi-secret sharing scheme has a process that more than one secrets can able to share but

all attributes receive one share only for reconstructing other secrets, no need for reassigning of share for every secret. Share size same with any one secret. Our method used multi-secret sharing scheme [20]-[21] reconstruction of secrets based on without particular order restriction [20] and 1-way function with 2 variable technique along with public shift technique in [20] method used. MSSS share multiple secrets but secrets reconstruct independently not have order restriction.

We used multi-secret sharing scheme with tree access structure. So far our scheme is first to used MSSS with tree access structure in CP-ABE for big data access control. By using MSSS with tree-based access structure gives the result as less computational overhead than other existing schemes.

Chapter 4

Proposed HP-CP-ABE Construction

We proposed a new hidden policy CP-ABE scheme for efficient big data access control with privacy-preserving policy. Our construction based on ciphertext-policy attribute-based encryption with (t,n) threshold multi-secret sharing scheme(MSSS). With tree-based access structure and mask, de-mask technique provides fully hidden access policy. By applying mask function during encryption, we fully hide the access policy. So that, there will be no possibility of leaking user's sensitive information. Our HP-CP-ABE has
4 algorithms: Setup, Keygen, Encryption, and Decryption

4.1 Setup

Setup algorithm takes Security parameter Λ and attribute universe U as input. This algorithm generates output as the Public key(PK) and master key(MK). This algorithm performs by the trusted authority(TA).

Trusted authority passes public key $PK = (G, h, r, \forall b_{ij} \epsilon U : (M_{ij}, p_j), n)$ to data owner and retains the master key $MK = (\alpha; n; \forall b_{ij} \epsilon U : (c_{ij1}, c_{ij2}))$. The set up algorithm describes in Algorithm 1.

Algorithm 1: Setup

G generated one bilinear group with prime order p and h generator
Choose random α, n
Calculate $r = e(h, h)^{\alpha}$
for $i = 1, 2..k$, $k =$ no of secrets **do**
  for each attribute $b_{ij} \in U$ **do**
    Choose random secretshare: $p_j \in Z_p$
    Choose random : $c_{ij1}, c_{ij2} \in Z_p$
    Compute $M_{ij} = h^{c_{ij1}c_{ij2}(c_{ij1}+c_{ij2})}$
  end for
end for
Return PK=$(G, h, r, \forall b_{ij} \in U : (M_{ij}, P_j), n), MK = (\alpha, n, \forall b_{ij} \in U : (c_{ij1}, c_{ij2})$

4.2 Keygen

Keygen algorithm takes Master Key MK with attribute set A as input and generates Secret Key SK associate with attribute set A as output. This algorithm generates the Secret key, which is associated with an attribute set generates by the trusted authority for each authorized user according to attributes. Secret key $SK = (E_{i1}, \forall b_{ij} \epsilon A : (E_{ij}, E'_{ij}), n)$ passes to authorised user by trusted authority. Then, user with help of secret key decrypting their data. Algorithm 2: describes keygen algorithm

Algorithm 2: Keygen

for $i = 1...K$, $K=$ no of secrets **do**
  Choose random $r \in Z_p$
  Compute $E_{i1} = h^{\alpha - y}$
  for each attribute $b_{ij} \in A$ **do**
    Compute $E_{ij} = h^{y/c_{ij1}}$ and $E'_{ij} = h^{y/c_{ij2}}$
  end for
end for
Return SK=$(E_{i1}, \forall b_{ij} \in A : (E_{ij}, E'_{ij}), n)$

4.3 Encryption

This algorithm takes input as Public Key PK, massage M and access policy $\tau_{i \in 1..k}$ and generates output Ciphertext CT embed with masked

access policy $m_{i\in 1..k}$. Data owner encrypts message under defined access policy $\tau_{i\epsilon i..n}$ that is in plaintext format. During encryption applied mask technique on defined access policy, so that the attributes of access policy completely hides. Algorithm generates the ciphertext CT = $(H_{i1}, H_{i2}, \forall b_{ij}\ A : H_{ij}; m_i)$ embed with masked access policy and stored into cloud. Algorithm 3: explains encryption algorithm.

Algorithm 3: Encryption

for for $i = 1...k$, k= no of secrets **do**
  Choose random secret $v_i \in Z_p$
  Compute $H_{i1} = h^{v_i}$ and $H_{i2} = Me(h,h)^{\alpha v_i}$
  for each node b_i in τ_i (top down manner) **do**
    if b_i root node and b_{ij}= $att(b_i)$, choose random s_{b_i} from Z_p and make polynomial that has degree $d_i - 1$, if $s_{b_i} = v_i$ shift value : $v_{ij} = s_{b_i} - s_j$ and child node r_i secretshare $s_{r_i} = v_{ij}(index(r_i))$ **then**
      else, choose random s_{b_i} and form polynomial with degree $d_i - 1$, $s_{b_i} = v_i = s_{parent(b_i)}(index(b_i))$, again assign child node $s_{r_i} = v_{ij}(index(r_i))$ to non-leaf node.
    end if
  end for
  Let A leaf node set
  for leaf node $b_i \in A$ in τ_i , $b_{ij} = atti(b_i)$ **do**
    Calculate $H_{ij} = {M_{ij}}^{(v_{ij}+p_j)} = h^{(v_{ij}+p_j)c_{ij1}c_{ij2}(c_{ij1}+c_{ij2})}$
  end for
  if $\tau_i = K$,where K= any real number **then**
    Compute mask $m_i = \tau_i * n$
    else, compute m_i = ASCII $(\tau_i) * n$
  end if
end for
Return CT=$(H_{i1}, H_{i2}, \forall b_{ij} \in A : H_{ij}, m_i)$

4.4 Decryption

Decryption algorithm takes Ciphertext CT , Secret Key SK as input and generates plain text Message M as output. In decryption algorithm, demask technique performed on masked access policy to get the original attributes of access policy. Which user's attribute satisfies the original access policy, that user only able to decrypt the ciphertext using the

secret key. Algorithm 4: describes the decryption algorithm.

Algorithm 4: Decryption

if $\tau_i = k$,where k= real number **then**
Compute demask $\tau_{i\in 1..k} = m_{i\in 1..k}/n$
else, compute ASCII $(\tau_{i\in 1..k}) = m_{i\in 1..k}/n$
end if
for every leaf node r in τ_i, where $j = atti(r)$,$j \in A$ and $i = 1..k$ no of secrets **do**
Compute $F_r = e(H_{ij}, E_{ij}E'_{ij})$
$= e(h^{(w_{ij}+p_j)l_{ij1}l_{ij2}(l_{ij1}+l_{ij2})}, h^{y/l_{ij1}}h^{y/l_{ij2}})$
$= e(h,h)^{y(w_{ij}+p_j)})$
$= e(h,h)^{ys_{b_i}}$
$= e(h,h)^{yv_i}$
Else $F_r = \gamma$
end for
for every nonleaf node r in $\tau_{i\in 1..k}$, **do**
S_r denoted as n'_r size child node set a $F_a \neq \gamma$, If no that type set exist then node not satisfies by attributes set A and return function γ.
Otherwise,
Compute $F_r = \prod_{a\in S_r} F_a^{\Delta, S'_r}$
$= \prod_{a\in S_r} (e(h,h)^{ys_{b_i}})^{\Delta_t, S'_r}$
$= \prod_{a\in S_r} (e(h,h)^{y.s_{parent(a)}(index(a))^{\Delta_t, S'_r)}}$
$= \prod_{a\in S_r} (e(h,h)^{y.v_i\Delta_t, S'_r)}$
$= e(h,h)^{y.v_i}$
Where $t = (index(a)), S'_r = (\forall a \in S_r : index(a))$ and lagrange co-efficient is $\Delta_t.S'_r$
end for

Chapter 5

Security analysis

In this, we analyze the security of our scheme by following two properties following properties

→ Correctness

→ Security Strength of HP-CP-ABE

5.1 Correctness

The user, whose attributes satisfies the access policy can able to obtain the data from the cloud server. A subset of t attributes by combining their assigned true share can recover a secret.

PROOF:

if b_i is root node in $\tau_{i\in 1..n}$

$F_{b_i} = \gamma$ **then $\tau_{i\in 1..k}$ not satisfies with A and return γ**

B= $e(H_{i1}, E_{i1})F_{b_i}$

$= s_{ij}e(h,h)^{ys_{b_i}}$

$= s_{ij}e(h,h)^{yv_i}$

then,

Compute $\frac{H_{i2}}{e(H_{i1},E_{i1})/B}$

$= \frac{Me(h,h)^{\alpha v_i}}{e(h^{v_i},h^{\alpha-y})/e(h,h)^{yv_i}}$

$= M$

5.2 Security of HP-CP-ABE

In this section, we described the security strength of hidden policy CP-ABE through a security game. This game described in section (III-2). In IND-CPA game, adversary A won game with a non-negligible advantage υ then simulator B able to solve DBDH assumption with advantage $\upsilon/2$. Then, simulator B sets two bilinear cyclic groups G_y, G_p of distinct prime orders y, p. Respective generators are h_y, h_p. Challenger sets Z_w , where $w = \{0, 1\}$

$Z_w = e(h_y, h_p)^{\theta}, w = 0$

$Z_w = e(h_y, h_p)^{abc}, w = 1$

Challenger sends DBDH challenges (h_y, D, E, F, Z_w) to simulator B . In IND-CPA game, simulator B plays as challenger.

- **<u>INITIAL:</u> Adversary A chooses a challenge access structure Z^* and sends it to challenger**

- **<u>SETUP:</u> Challenger chooses one random element $r' \in Z_p^*$ and set $\alpha' = bx + y', r' = e(h_y, h_p)^{\alpha'} = e(h_y, h_p)^{bx} e(h_y, h_p)^{y'}$**

  Randomly select following elements $c'_{j1}, c'_{j2}, p'_j \in Z_s^*$, $(j = 1..n)$ and sets below parameter

  $M'_j = h_y^{c_{j1}c_{j2}/(c_{j1}+c_{j2})}, b'_j \in Z^*$

  The challenger sends $PK^*(h_y, r', \forall b'_j \in Z^* : (M'_j, p'_j))$, $(1 \leq j \leq k)$ to adversary

- **<u>PHASE1:</u> Adversary sends a request for user private key to challenger over any attribute set, let the attribute set is S'_j=$\{b'_j / b'_j \in \Omega$, $(b'_j \notin Z^*)\}$**

  Challenger selects random $q' \in Z_s^*$ element for adversary's each request and sets $y' = bx + q'x$, $d'_1 = h_t^{\alpha-(bx+q'x)} d'_j = h^{y/c_{j1}}, d'_{j'} = h^{y/c_{j2}}$

  As restriction $b'_j \in Z^*$, adversary obtained result and the challenger send private key $SK^* = (d'_1, \forall a'_j \in S'_j : d'_j, d'_{j'})$ to adversary

  <u>Challenge:</u> Adversary submits two message m_0 and m_1. Then challenger selects any random massage m_x, where $x \in \{0, 1\}$ and encrypt

message with challenge access policy $z*$

Following message encryption :

$H'_1 = h_r^{c'}, c' \in z_s^*$

$H'_2 = m_b e(h_y, h_y)^{bxc'} e(h_y, h_y)^{y'c'}$

If b'_j **is root node in access policy, then take one random polynomial number** $s'_j \in z_s^*$ **of degree** $d'_j - 1$**. Let** $s'_j = c'$**, then by applying MSSS assign secret share to each child node.**

Else, for every non-leaf node take random polynomial no $s'_j \in Z^*$**,** $s'_j = c' = s'_{parent(b'_j)}(index(b'_j))$ **and again assign secret share to each child node.**

For each leaf node : $H'_j = M_j^{(w'_j + p'_j)}$**.**

Then Challenger gives the Ciphertext $CT^*(H'_1, H'_2, \forall b'_j : H'_j)$ **to adversary.**

PHASE2: With same restriction as PHASE1, adversary continues the secret key request to challenger and challenger also does the same as in PHASE1.

GUESS: Adversary guess a output $b' \in \{0, 1\}$ **. If** $b' = b$ **then simulator guesses** $w = 1$**,** $Z_w = e(h_y, h_p)^{bxc}$**, then adversary found the valid ciphertext. Therefore, simulator can solve the assumption of DBDH with following advantage is**

$$pr[b = b/Z_w = e(h_y, h_p)^{bxc}] = \frac{1}{2} + \upsilon$$

otherwise challenger guess $w = 0$**,** $Z_w = e(h_y, h_p)^{\theta}$**,**
$e(h_y, h_p)^{\theta}$ **is a random ciphertext, now adversary can not get any information about** m_b**. So, the disadvantage is**

$$pr[b' \neq b/Z_w = e(h_y, h_p)^{\theta}] = \frac{1}{2}$$

The conclusion is for any guesses simulator solves DBDH assumption with following advantage

$$\frac{1}{2} pr[w' = w/w = 0] + \frac{1}{2} pr[w' = \frac{w}{w} = 1] - \frac{1}{2} = \upsilon/2$$

If to wining IND-CPA game adversary has the above advantage υ then, DBDH problem solves by simulator with advantage $\upsilon/2$. However, according to DBDH assumption, no polynomial time algorithm can able to solve DBDH assumption with non-negligible advantage. Hence, adversary also not able to win IND-CPA game with advantage υ and adversary A have no advantage to break our system.

Chapter 6

Performance Analysis

In performance analysis, computation cost of our scheme analyzed through Theoretical and Experimental Results

6.1 Theoretical Analysis

Schemes	Access Structure	Encryption Time	Decryption Time	Hidden Policy
Lie et al. [6]	LSSS Access Structure	$(8n + 4)exponent$	$(4n + 2)e + (2n + 3)exponent$	Partial
Phuong et al. [8]	AND-gate Access Structure	$\|G_T\| + (4n + 2)\|G\|$	$(4n + 2)p$	Partial
Helil et al. [9]	Tree Based Access Structure	$(3n + 4)exponent + (t + 2)e$	$e + 3exponent$	Partial
Yang et al. [18]	LSSS Access Structure	$(2n_l + 2)exponent$	$(2n_l + 1)p + n_l exponent$	Fully
Our Scheme	Tree Based Access Structure	$(2n)exponent$	$(n + 1)exponent$	Fully

Table 6.1: Comparison of computation overhead

In Table 6.1, we measure the computation cost of the proposed scheme along with other existing hidden policy schemes. The given Encryption time and Decryption time in Table 6.1 derived from respective Encryption and Decryption algorithm of their method. From Table 6.1, we can see that HP-CP-ABE takes less encryption and decryption computational time as compared to other given existing schemes.

Therefore, our HP-CP-ABE scheme is efficient enough. Our scheme gives less computational overhead for both encryption and decryption, because we used MSSS with tree based access structure

6.2 Experimental Results

Our experiment carried on Linux system with an Intel i5 CPU and 8 GB RAM. For implementation, we used python-3.2, pairing-based cryptography library version 0.5.14 and charm crypto library 0.43.
In our experiment, we compare the computational cost of encryption and decryption of our proposed scheme with the existing scheme Yang et al. [18]. As shown in Figure 6.1 and Figure 6.2, our proposed scheme takes less computation overhead than the existing scheme Yang et al. [18]. From Figure 6.1, it observed our proposed method takes lesser time than already exist scheme Yang et al. [18], due to the use of multi secret share scheme on tree-based access structure. Mask technique applies to access policy in place of existing method's linear secret sharing scheme (LSSS) with Attribute Bloom Filter. Encryption time contains data encryption time and the time taken for mask technique. Figure 6.2. shows decryption computation time verse number of attributes in the access policy. From Figure 6.2, it observes, our proposed scheme takes lesser decryption time, due to the use of de-mask technique during decryption method on access policy in place of the existing method's attribute Bloom filter query algorithm. The decryption includes the time for de-mask technique and the data decryption time. In Yang et al. [18]. used attribute bloom filter that takes more overhead because of an array used in attribute bloom filter(ABF). Our scheme takes less computational overhead for both encryption and decryption while preserving access policy privacy. In Experimental result, we compare our result with only Yang et al. [18]. because only this scheme works for big data privacy.

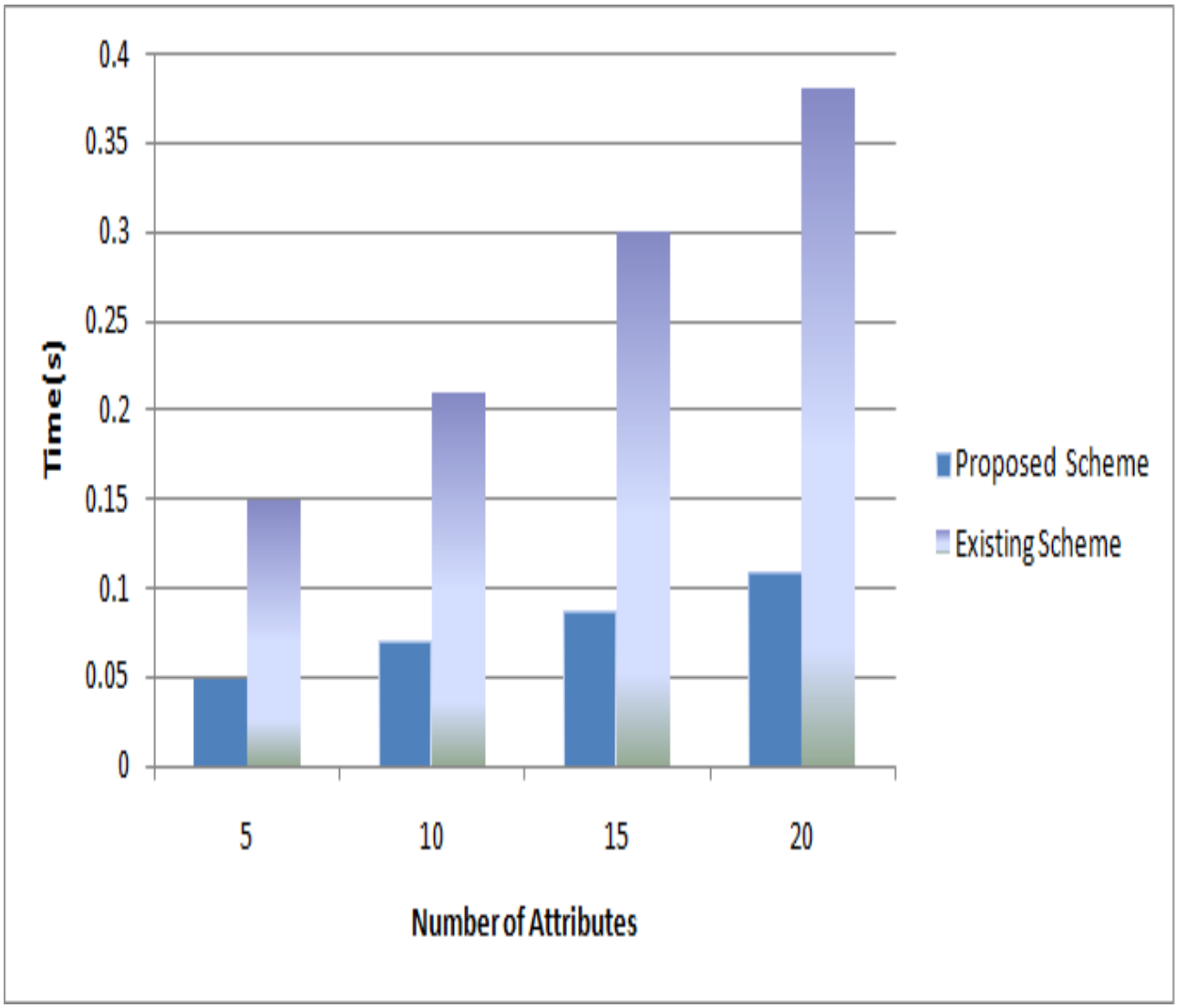

Figure 6.1: Computation cost for encryption

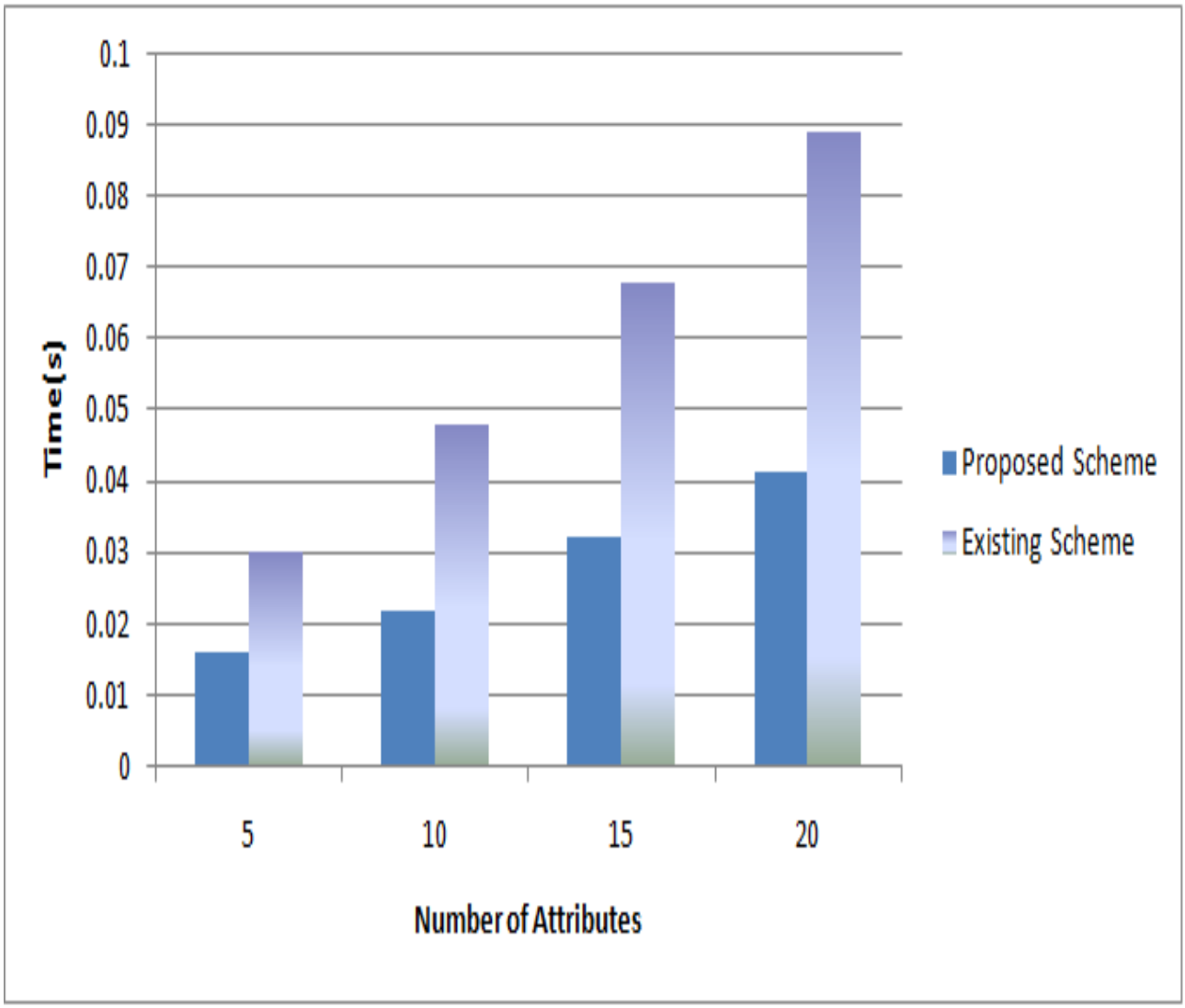

Figure 6.2: Computation cost for decryption

Chapter 7

Conclusion

In our project, we proposed a new HP-CP-ABE for big data access control with privacy-preserving policy using multi-secret sharing scheme. Mask technique applied to access policy during encryption, so that it fully hides the attributes in access policy and during decryption applied the de-mask technique on the masked access policy for getting original attributes in the access policy. Hence, our scheme completely hides the access policy that means there is no chance of leaking the user's any sensitive information. Because of the multi-secret sharing scheme with tree-based access structure in CP-ABE, our scheme takes the less computational cost for encryption as well as decryption. Security analysis shows that our scheme is selectively secure against plain-text attack through IND-CPA game and can preserve users stored information effectively that attacker cannot obtain any sensitive user's information. Experimental results show that, our scheme is efficient due to the less computational overhead of encryption and decryption than existing hidden policy schemes.

In future work, we plan to extend this work to Dynamic update policy. If new users want to access the data, the access policy can change according to users.

Bibliography

[1] P. Mell and T. Grance, The NIST definition of cloud computing, Recommendations of the National Institute of Standards and Technology- Special Publication 800-145], 2011.

[2] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, Toward efficient and privacy-preserving computing in big data era, IEEE Network, vol. 28, no. 4, pp. 4650, 2014.

[3] John Bethencourt, Amit Sahai and Brent Waters"Ciphertext Policy-Hiding Attribute-Based Encryption" 2007 IEEE Symposium on Security and Privacy(SP'07).

[4] Heng He, Ruixuan Li, Xinhua Dong, and Zhao Zhang "Secure, Efficient and fine -grained data access control mechanism for p2p storage cloud" IEEE TRANSACTIONS ON CLOUD COMPUTING, VOL. 2, NO. 4, OCTOBER-DECEMBER 2014.

[5] Zhiwei Wang and Mingjun He "CP-ABE with Hidden Policy from Waters Efficient Construction" Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2016, Article ID 3257029, 8 pages http://dx.doi.org/10.1155/2016/3257029.

[6] J. Lai, R. H. Deng, and Y. Li, "Expressive cp-abe with partially hidden access structures," in Proc. of ASIACCS12. ACM, 2012, pp. 1819.

[7] **Xu, R. and Lang, B. (2015) A CP-ABE scheme with hidden policy and its application in cloud computing, Int. J. Cloud Computing, Vol. 4, No. 4, pp.279298.**

[8] **Tran Viet Xuan Phuong, Guomin Yang and Willy Susilo, "Hidden Ciphertext Policy Attribute-Based Encryption Under Standard Assumptions" IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 11, NO. 1, JANUARY 2016**

[9] **Nurmamat Helil and Kaysar Rahman, "CP-ABE Access Control Scheme for Sensitive Data Set Constraint with Hidden Access Policy and Constraint Policy" Hindawi Security and Communication Networks Volume 2017, Article ID 2713595, 13 pages https://doi.org/10.1155/2017/2713595**

[10] **Lixian Liu, Junzuo Lai, Robert H. Deng and Yingjiu Li, "Ciphertext-policy attribute-based encryption with partially hidden access structure and its application to privacy-preserving electronic medical record system in cloud environment "SECURITY AND COMMUNICATION NETWORKS Security Comm. Networks 2016; 9:48974913 Published online 8 November 2016 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1663**

[11] **J. Li, K. Ren, B. Zhu, and Z. Wan, Privacy-aware attribute-based encryption with user accountability, in Information Security. Springer, 2009, pp. 347362.**

[12] **J. Lai, R. H. Deng, and Y. Li, Fully secure cipertext-policy hiding cpabe, in Information Security Practice and Experience. Springer, 2011, pp. 2439 IEEE Transl. J. Magn.Jpn., vol. 2, Aug. 1987, pp. 740741 [Dig. 9th Annu. Conf. Magnetics Japan, 1982, p. 301].**

[13] **S. Yu, K. Ren, and W. Lou, Attribute-based content distribution with hidden policy, in Secure Network Protocols (NPSec08 Workshop). IEEE, 2008, pp. 3944.**

[14] **T. Nishide, K. Yoneyama, and K. Ohta, Attribute-based encryption with partially hidden encryptor-specified access structures, in Applied cryptography and network security. Springer, 2008, pp. 111129.**

[15] **D. Boneh and B. Waters, Conjunctive, subset, and range queries on encrypted data, in Theory of cryptography. Springer, 2007, pp. 535554.**

[16] **J. Katz, A. Sahai, and B. Waters, Predicate encryption supporting disjunctions, polynomial equations, and inner products, in Advances in CryptologyEUROCRYPT08. Springer, 2008, pp. 146162.**

[17] **Runhua Xu, Yang Wang and Bo Lang"A Tree-based CP-ABE Scheme with Hidden Policy Supporting Secure Data Sharing in Cloud Computing" 2013 International Conference on Advanced Cloud and Big Data.**

[18] **Kan Yang, Qi Han, Hui Li, Kan Zheng, Zhou Su, and Xuemin (Sherman) Shen, "An Efficient and Fine-grained Big Data Access Control Scheme with Privacy preserving Policy"Citationinformation:DOI 10.1109/JIOT.2016.2571718, IEEE Internet of Things Journal.**

[19] **Fawad Khan, Hui Li, Liangxuan Zhang and Jian Shen, "An Expressive Hidden Access Policy CP-ABE"2017 IEEE Second International Conference on Data Science in Cyberspace**

[20] **He, J., Dawson, E.: Multisecret-sharing scheme based on one-way function, Electron. Lett., 1995, 31, (2), pp. 9395.**

[21] **Harn, L.: Comment on: multistage secret sharing ased on one-way function, Electron. Lett., 1995, 31, (4), pp. 262.**