Scalable revocation in CP-ABE with constant ciphertext length in storage constrained IoT devices

M.Tech Major -II Project Report

Submitted in partial fulfillment of
the requirements for the award of the degree
of
Master of Technology
in
Software Engineering
by

Anadi Shakya
(2K17/SWE/02)

under the guidance of

Ms Divyashikha Sethia
Assistant Professor
COE Department

DEPT. OF COMPUTER SCIENCE AND ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITy, DELHI
JUNE 2019

DELHI TECHNOLOGICAL UNIVERSITY
(Formaly Delhi College of Engineering)
Bawana Road, Delhi-110042

DECLARATION

I, Anadi Shakya, 2K17/SWE/02 student of MTech Software Engineering, hereby declare that the project Dissertation titled "Scalable revocation in CP-ABE with constant ciphertext length in storage constrained IoT devices" which is submitted by me to the Department of Computer Science, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **ANADI SHAKYA**
Date:

DELHI TECHNOLOGICAL UNIVERSITY
(Formaly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the project Dissertation titled "Scalable revocation in CP-ABE with constant ciphertext length in storage constrained IoT device" which is submitted by Anadi Shakya, 2K17/SWE/02 Department of Computer Science and Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the Degree of master of technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge, this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi
Date:

DIVYASHIKHA SETHIA
SUPERVISOR
Assistant Professor
Department of Computer Science and Engineering
Delhi Technological University
Bhawan Road, Delhi-110042

ACKNOWLEDGEMENT

I would like to express my gratitude and appreciation to all those who gave me the support to complete this project.

A special thanks to my mentor and project guide, **Ms Divyashikha Sethia** , whose help, stimulating suggestions and encouragement, helped me to make our ideas come into reality. I would like to take this opportunity to express my profound gratitude not only for her academic guidance but also for her personal interest in my project and constant support coupled with confidence boosting and motivating sessions which proved very fruitful and were instrumental in infusing self-assurance and trust within me.

The crucial role of the staff of Computer Science & Software Engineering is also acknowledged with much appreciation, which helped me throughout the process of the development of this project by giving appropriate suggestions and assistance.

Also, I am obliged to mention the support provided by my parents and my peer group. Also, I would like to appreciate the contribution and help provided to me by the seniors and staff working in LANS Lab.

ANADI SHAKYA
2K17/SWE/02

ABSTRACT

Portable devices such as a smartphone or an IoT device can be used to selectively share secure data with several users. A Ciphertext-Policy Attribute based Encryption (CP-ABE) is a fine-grained encryption technique, which can serve as selective access control mechanism. Due to the resource constraints and battery limitation in mobile devices, there is a requirement of an efficient CP-ABE Scheme.

In this work, we refer to a RSA-based CP-ABE scheme which does not use costly bilinear maps with efficient storage. Irrespective of the total number of attributes defined, the length ciphertext and secret keys remains constant. However, it is also important to support efficient revocation to protect from malicious users and allow valid users for uninterrupted access. The RSA-based CP-ABE scheme lacks support for revocation.

Contents

List of Tables

List of Figures

LIST OF SYMBOLS, ABBREVIATION AND ACRONYMS

- IBE: Identity Based Encryption is a scheme where the identity of the decryptor cannot be revealed.
- ABE: Attribute Based Encryption scheme is an encryption techniques where multiple user can decrypt a data if they have legit set of attributes.
- KP-ABE: Key Policy ABE, in these schemes, each users private key is associated with an access structure.
- CP-ABE: Ciphertext- Policy, here each ciphertext is associated with an access structure.
- PES: A Predicate Encryption Scheme (PES), is a variant of ABE, in which predicates are associated with the secret keys and ciphertext are associated with the attributes.
- PIRATTE: Proxy based Immediate Revocation of ATTribute-based Encryption; it uses a trusted third-party called proxy server to enhance the CP-ABE scheme.
- CCA: Chosen Ciphertext Attack, the attacker is able to obtain the decryption of any ciphertext of his choosing, except the challenge. It models the case where tricking an enemy into decrypting many ciphertext for you will not help you into breaking any others.

Chapter 1: INTRODUCTION

1.1 OVERVIEW

With an increase in digitalization, the focus has shifted from man-work to making machines do the work. Researchers and developers are continually working on building machines that are intelligent which lessens the efforts made by human beings. Smartphones and Internet of Things (IoT) devices are some examples of such kind. Sharing data among ourselves and others, has become one of prominent role that is carried out most frequently and readily through such devices due to its handiness and availability of data. The data distributed can be some regular mails, sms, images or highly sensitive data, such as passwords that are exchanged among friends, relatives or used at commercial, official, or at academic level in any workplace. This data can either be stored in ones personal device or at the cloud. Data stored at cloud can be manipulated openly, same as, data stored in personal device is secured to a limit which can also be exposed after breaching. Thus, we require some mechanism to safeguard our data from any adversary that can manipulate the data for his/her own good. In the recent times, more sophisticated and effective mobile devices are designed for faster sharing of data. However, these devices comes with a disadvantage of constrained storage space and security [1], which needs to addressed along with the fast delivery of data. Devices can be configured for the users using encryption schemes that are effective in security and utilizes less storage space. [1].

Attribute Based Encryption Scheme
Numerous identity-based encryption scheme [2], [3], [4] have been proposed with constant length ciphertext and secret keys. The Attribute Based Encryption scheme (ABE), an extension of identity-based encryption scheme, is a fine-grained encryption technique, in which, a user is having a legit set of attributes that can decrypt a given ciphertext if it holds with the access policy associated with the given attributes. A user characteristic like name, bio-metric, contact of address, email-id, date of birth or any relevant information can be used as attributes. The ABE schemes have two variants namely Key-Policy Attribute Based Encryption scheme (KP-ABE) and Ciphertext Policy Attribute Based Encryption scheme (CP-ABE) [5]. In KP-ABE [5] [6] schemes, every user has a private key which is associated with an access structure. On the other-hand, int CP-ABE schemes every ciphertext is associated with an access structure; this implies that an encryptor has the power to decide which decryptor is allowed to gain access to the ciphertext and which do not. Since, in CP-ABE scheme, an encryptor is able to decides who is allowed to access the data and chooses an access policy, therefore, it fits effectively

in the situations where access control is major concern within the applications, as compared to KP-ABE.

In current CP-ABE schemes [7] [8], it is quite noticeable that the length of ciphertext is dependent upon the number of attributes present in the access structure. Also, the number of pairing computation gets increased as the number of attributes increases. The length of ciphertext plays a vital role in any CP-ABE system. Cloud storage systems are capable of storing such long ciphertexts, but for mobile devices having limited space, it can be a bane. Emura et. al [9] proposed a CP-ABE scheme where the length of ciphertext and secret keys remains constant irrespective of the increase and decrease in the number of attributes. It incorporates bilinear pairing in it. Studies have found that, bilinear pairing takes long time for computations. Odelu et. al [10], is a CP-ABE which is based upon RSA computations and overcomes overhead of bilinear maps here. This makes Odelu et. al [10] an effective CP-ABE scheme that can be used in battery-limited mobile devices.

Revocation

Along with sharing data, there comes revocation. That is, giving authority to access information only to specific people and then, being able to revoke a person, when required. In any system, a revoked user should not be able to decrypt the data. Hence, revocation becomes an integral part of CP-ABE schemes. Revocation can be achieved in CP-ABE schemes by direct and indirect methods [11]. For memory constrained devices re-encryption becomes inefficient in terms of time and cost, also leads to interruption of services for genuine users.

1.2 MOTIVATION

In recent times, it is noteworthy that, there is a growing demand of the battery and storage constrained mobile devices that are easily available and have become popular amongst us. Due to this increasing demand, it has become necessary to develop applications that are light-weight, design efficient and secured. Thus, CP-ABE seems a suitable option for cloud computing environment, where a data owner has the rights to write and authorize access depending upon the polices, drawn by themselves. Seeing that, most of the mobile devices are battery-constrained, thus, it is necessary to develop and design such CP-ABE schemes are having constant length ciphertext and secret keys, keeping in mind that the cost to the mechanism is encryption and decryption efficient.

In the literature, several CP-ABE schemes have been developed. Some have constant size ciphertext [9], [12], [13], [14] and other having constant size secret keys [14], [9]. Among the above mentioned encryption schemes, Emura et. al [9] is the first scheme to have the length of both secret keys and ciphertext as constants, irrespective of increase and decrease in attribute set. It uses AND-gate multivalued attributes in its access structure. In inclusion, above scheme are based upon bilinear mapping, that are expensive as compared to the present conventional schemes [2], [15].

Thus, it becomes necessary to develop a system that is less expensive plus supports constant length ciphertext and secret keys. Odelu et. al [10] proposed a CP-ABE scheme, that is based upon RSA and AND-gate access structure. It provides constant size ciphertext and secret keys with efficient encryption and decryption technique. It does not utilizes bilinear mapping, therefore, it makes it better than other CP-ABE schemes as well as cost efficient for resource constrained battery operated devices.

As discussed in 1.1, we know that revocation plays a vital role n CP-ABE. Once given access, it is necessary to be able to revoke a user or an attribute. Several revocation CP-ABE scheme have been proposed [16], [17]. However, revocation of attributes, that is, adding or subtracting attributes from the attribute list results into re-generation and re-distribution of secret keys. This makes the process time-consuming and, also affecting other legitimate user. Thus, it is necessary to develop revocation mechanism in CP-ABE scheme, such that, there is no-regeneration or re-distribution of secret key. Also, allowing users that are legitimate, to work uninterruptedly. Thus, according to Setia et. al [17], the identified requirements for efficient scalable revocation in a CP-ABE scheme can be listed as follows:

1. **Absence of prior knowledge of revocation list.**
   Having no prior information regarding revocation list help in sharing the ciphertext across a large group of users. If we maintain a revocation list prior to encryption then we limit our sources only to those users present in list, thus result in lack of scalability.

2. **Ciphertext should not be re-encrypted.**
   A ciphertext should not be re-encrypted after revocation thus maintain uninterrupted environment for owner and other authorised users.

3. **Secret keys should not be re-generated and re-distributed**
   There should not be re-generation and re-distribution of the secret keys after revocation so that non-revoked legitimate users can continue to access the encrypted message uninterruptedly.

4. **Scalable revocation of users.**
   A CP-ABE scheme should give authority to owner to share data among multiple users. Also, it should be able to revoke the malicious users.

5. **Independence from ciphertext information**
   The system should not store any ciphertext specific information that can be used in user revocation to lessen revocation and storage overhead.

ProSRCC [18] is extension of Emura et. al [9], that introduces revocation feature in [9] and is selectively secured against Choosen Plaintext Attack (CPA). It revokes a user without re-distributing and re-generating secret keys. The drawback for ProSRCC [18] is, it is vulnerable against a Choosen Ciphertext Attack (CCA). However, Odelu et. al [10], is more cost effective as compared to Emura et. al [9] and selectively secured againt Chooosen Ciphertext Attack, although, it lacks revocation feature. This thesis seek to bridge this gap between Odelu et. al [10] and revocation mechanism.

1.3 PROBLEM STATEMENT

The main objective of this research work is to find the latest CP-ABE scheme which is excellent for light-weight, storage contrained and battery-limited devices and then integrate revocation feature in it. This work is the extension the existing scheme Odelu et. al [10] into a CP-ABE scheme that supports revocation mechanism. It combines RSA along with AND-gate access structure. It provides constant length ciphertext and secret keys, along with effective encryption and decryption scheme.

1.3.1 Proposed Solution

- It requires a trusted proxy server that should be online all the time, to ensure safety against a malicious user. The server keeps a revocation list, partial secret-key for a user to itself whereas the other half is with the user. The decryption takes place in two stages. In the first stage, the proxy server calculates the partial decryption of ciphertext and then forwards it to all the users such that the user who is not having legit another half of the secret-key is not able to decrypt the ciphertext since they are revoked and legitimate users can decrypt it without any interference.

- Performances is analysed using experiment results of the proposed scheme, and comparing it with the existing CP-ABE schemes to check for the efficient and effectiveness of the work.

Chapter 2: RELATED WORK

2.1 CP-ABE SCHEMES SUPPORTING AND-GATE ACCESS POLICY

Cheung et al. [8] introduced a new CP-ABE scheme, which supports AND gate access policy with two types of attributes, positive and negative attributes. It termed the attributes, which participate in the access policy as positive terms. For those attributes, which are not a part of the access structure, it uses a wildcard element. The scheme is Chosen Ciphertext Attack (CPA) secure under the Decisional Bilinear Diffie-Hellman (DBDH) assumption.

Moreover, it improves the security proof in Bethencourt et al. [7] scheme. It is less proficient as compared to Bethencourt et al. 's CP-ABE scheme [7] because it is not flexible enough. It supports access policies that are consists logical conjunction only, and the size of the ciphertext and the secret key linearly increase as the number of attributes gets increased in this scheme. Based on Cheung et al.'s scheme [8] and Emura et al. [9] scheme further improved the efficiency and provided hidden access policies. Emura et al. [9] scheme uses similar access policy and further improves the scheme to achieve a constant number of bilinear pairing operations along with a constant length of ciphertext.

Another novel CP-ABE scheme for storage constrained devices, is given by Odelu et. al [10] scheme. It defines lightweight security protocol for IoT devices. The scheme is a combination of RSA and CP-ABE that results into constant ciphertext length and security keys. It is selectively secures against key recover, collision attack, and CCA attack under DBDH assumtion. It does not uses bilinear maps, which requires high computation time, instead it uses XORs. It uses AND-gates access structure. It calculates RSA modulus $N = pq$, where p and q are very large primes. It takes into account integer factorization problem which is gives as computationally hard problem. Therefore, deducing p and q from N is infeasible. Then it follows the CP-ABE scheme integrating RSA primes and modulus. Further it defines one-way collision hash functions and XORs plaintext message and signature element S_m. S_m acts as a verification key which is used to identify users authenticity. It gives away the CCA secure element in the CP-ABE scheme. In decryption stage, the scheme XORs the result to generate the original message, and also to verify the user.

2.2 CP-ABE WITH REVOCATION

For encryption systems it is essential to have revocation feature that can deal with the malicious behavior of users. However, the adding a revocation feature in any existing CP-ABE schemes is much more cumbersome than any public key cryptosystem or Identity Based Encryption (IBE) schemes.

According to Pang et. al [11], there are two methods to realize revocation indirect revocation method and direct revocation method. A revocation method, where the owner can delegate the authority to execute the revocation function, which then releases a key-update material after every delegation performed. Only the non-revoked users will be able to update their keys, this method is known as indirect revocation method. In the direct revocation method, revocation is performed by the users directly. In this method the revocation list is specified while encrypting the ciphertext. The design of revocation mechanisms in previous CP-ABE schemes was difficult as users with same attributes might have been holding the same user secret key. In ProSRCC [18], a novel revocation scheme, extends Emura et al.'s [9] CP-ABE scheme by providing scalable user revocation and non-revoked user can continue to do their work uninterruptedly. Hence, it can be used for direct selective access to information. However, it is only secure against CPA attacks.

Jahid et al. [16] proposed another such scheme for revocation, named Proxy based Immediate Revocation of ATTribute-based Encryption (PIRATTE). Their scheme uses trusted proxy server and enhances the Bethencourt et al. 's CP-ABE scheme [7]. However, both the schemes suffer from the increasing ciphertext size problem. Such schemes divide the user secret-key into two parts. The proxy server keeps a revocation list, and one part of the user secret-key to itself and the user keeps the other part. Whenever the Trusted Computing Authority (TCA) discovers a malicious user or some attributes to be revoked, it lists them in the revocation list held by the proxy server. The Key Authority generates a polynomial P of degree $t+1$ in the master key where t is the number of users that can be revoked. The trusted server divides the secret $P(0)$ into portions and provides a share to each user. During decryption, each user seeks a proxy key and t shares of the secret from the proxy-based server. It uses Lagrange's interpolation to combine the t secret portions with the user portion to generate the secret $P(0)$. If the user is non-revoked, the proxy-based server sends valid secret portions. Otherwise, it sends invalid secret portions, so that the user cannot generate the secret $P(0)$ not allowing them to unblind their secret key. Thus decryption fails. PIRATTE [16] does not need any prior knowledge for either the list of revoked user or the re-encrypted ciphertext or re-distributed secret keys generated after revocation. Yet, PIRATTE [16] is only able to revoke a limitied noumber of users.

Sethia et al. [19] presented another novel scheme Scalable Proxy-based Immediate Revocation for CP-ABE (SPIRC) scheme for user revocation. It is an improvement over previously discussed PIRATTE scheme [16] for scalable user revocation. Unlike PIRATTE [16], for master key MK, there is no generation of polynomial P of degree $t+1$. Instead the proxy-based server maintains a random set S_i for each user along with a revocation list. For the completion of decryption, $user_i$ seeks proxy data PXD from the proxy-based server which is unique for every user. $PXD = \lambda_i$. The trusted server sends proxy data PXD to $user_i$, who also sends $C_x^{'}$ to the proxy-based server to return Convert $C_x^{''}$.The user secret SK is blinded by $(\lambda_i a_i + b_i)$ and needs $C_x^{''}$ along withC_x and $C_x^{'}$. The proxy can revoke the user by updating the λ_i and b_i for $user_i$ in PXD and $C_x^{''}$. SPIRC [19] like PIRATTE [16] does not require any prior knowledge for either the list of revoked user or the re-encrypted ciphertext or re-distributed secret keys generated after revocation. The owner can to share ciphertext with multiple users as well as revoke a scalable number of malicious users. However, since it is based on Bethencourt's CP-ABE scheme [7] the length of the ciphertext is not constant.

The Proxy-based Scalable Revocation for Constant Ciphertext Length (ProSRCC) [18] scheme improves the Emura et al.'s scheme [9] for scalable revocation. The ProSRCC [18] scheme supports two types of revocation schemes attribute-based and user-based revocation. The ProSRCC does not requires any re-encryption or re-generation and re-distribution of secret keys. It incorporates proxy server. Proxy sever works as an essential part of ProSRCC scheme. It holds the proxy elements, list of revoked users, revoked attributes and revoked attributes corresponding to a specific user. For a user to decrypt a ciphertext, it first have to ask the proxy server for the proxy elements. Depending upon the information in the list, the proxy server computes the proxy elements and replies to the user. The user than can decrypt the ciphertext. The proxy server is self-sufficient to handle access control and revocation of attributes as well as users. It support AND-gate multi-valued attributes. It is also secure again CPA attacks.

Below a TABLE 2.1 is shown, which compares the existing CP-ABE scheme with our scheme on the basis of the functions a scheme performs. We can observe that our scheme has constant length ciphertext, along with revocation, and is also secure against both CPA and CCA attacks.

Table 2.1: Comparison of schemes by their functionality

Scheme Name	Constant Ciphertext	Revocation Feature	CPA Secure	CCA Secure
Odelu et. al [10]	√	×	×	√
Emura et al. [9]	√	×	√	√
PIRATTE [16]	×	√	√	×
SPIRC [19]	×	√	√	×
ProSRCC [18]	√	√	√	×

Chapter 3: PRELIMINARY REQUIREMENTS

In this section, we discuss the computationally hard problem, types of attributes and definition of access structure, CCA security game and define CP-ABE scheme. We define a list of notations that we will use throughout the thesis in the table 3.1.

Table 3.1: List of Symbols

Notations	**Meaning**
(k, x)	Private key for the system
$N = pq$	Modulo used in RSA, with 2 distinct large primes, p and q
Z_a	Congruence class set for a
$\phi(.)$	Totient function, for N (product of primes p and q), equals $\phi(p)\phi(q)$, where $\phi(prime) = (prime - 1)$.
H_i	i^{th} hash function (one way and collision resistant).
U	Complete attribute set (or universe) $\{A_i\}_{i=1}^n$
A	Attribute set used for the user, subset of U
P	Access Policy
$\|V\|$	Cardinality of set V
C_{u_i}	A proxy element calculated by the proxy server for i_th non-revoked user that is used in decryption
C_{a_i}	A proxy element calculated by the proxy server for i_th non-revoked user that is used in decryption
MPK, MSK	Master Public Key, and Master Secret Key
C, M, RL	Ciphertext, Message, Revocation List

3.1 ATTRIBUTES AND ACCESS STRUCTURE

3.1.1 Definition of attribute and access policy

Let U be a set of n attributes $attr_1, attr_2, ..., attr_n$ denoted as $U = \{attr_1, ..., attr_n\}$. For every user, we denote an attribute set A such that, $A \subseteq U$. Also, a bit string of length n, that is, $a_1, a_2, ..., a_n$ is associated with A, defined as: $a_i = 1$, if $attr_i \in A$, otherwise $a_i = 0$, if $attr_i \notin A$. For example, let $n = 4$ and $A = \{attr_1, attr_3\}$, then the 4-bit string associated with A will result in 1010. Similarly, we define access policy P, such that, it haves the attribute that are present in U, $P \subseteq U$. Also, a bit string of length n, that is, $b_1, b_2, ..., b_n$ is associated with P, defined as: $b_i = 1$, if $attr_i \in P$, otherwise $b_i = 1$, if $attr_i \notin P$. For example, let $n = 4$ and $A = \{attr_1, attr_2, attr_4\}$, then the 4-bit string associated with P will result in 1101.

3.1.2 Access Structure Defined

We use AND-gate access structure represented in terms of attributes present in U. Let U be a universal bet having n attributes. $a_1a_2...a_n$ be the $n-$bit string associated with A and $b_1b_2...b_n$ be the $n-$bit string associated with P. Then, $P \subseteq A$, if and only if, $a_i \geq b_i$, $\forall i = 1, 2, ..., n$. If $P \subseteq A$, then we call it as, attribute set A satisfies the access policy P, denoted as, $A \models P$ and the user associated with A, can access the data associated with access policy P.

3.2 COMPUTATIONALLY HARD PROBLEMS

3.2.1 Integer Factorization Problem (IFP)

Let p and q be large prime numbers, with $\rho-$bit length. Computer N, as $N = pq$. Assume GEN_F be a probabilistic polynomial time algorithm, which takes an input 1^ρ and outputs (N, p, q). Then factoring assumption related to GEN_F states as follows: given N it is computationally infeasible to generates the factors of p and q, such that $N = pq$, hence the value of p and q except with a negligible probability of ρ. Therefore , it is categorized as computationally hard problem.

According to to Hofheinz and Kiltz [20], the above definition can be given by:
Let η be probabilistic polynomial time algorithm, then the advantage is defined as:

$$Adv^{IFP}_{GEN_F,\eta}(\rho) = Prob[(N, p, q) \leftarrow GEN_F(1^\rho) : \eta(N) = \{p, q\}] \tag{3.1}$$

Thus, the factoring assumption with respect to GEN_F, says that, $Adv^I FP_{GEN_F,\eta}(\rho)$ is negligible in ρ for every probabilistic polynomial time algorithm η.

If t_{IFP} is the running time, then we say that, $(t_{IFP}, \epsilon_{IFP}) - IFP$ assumption holds only if, $Adv^{IFP}_{GEN_F,\eta}(\rho) \leq \epsilon_{IFP}$, for very small $\epsilon_{IFP} > 0$ and its time is utmost t_{IFP}.

3.2.2 Diffie-Hellman Problem (DHP)

Let g be a generator, and $a, b, c, z \in Z_p$. $\langle g^a \rangle, \langle g^b \rangle$ be the two cyclic group in Z_p, generated using generator g. Then,

$$DHP(N, g, X, Y) : \langle g_N \rangle \times \langle g_N \rangle \to \langle g_N \rangle \tag{3.2}$$

used to decide whether

$$DHP(N, g, g^a, g^b) = g^{ab}\ (modN) \tag{3.3}$$

This is knows as Diffie-Hellman (DHP) problem which uses RSA modulus $N = pq$ and base g [21].

If t_{DHP} is the running time, then we say that an adversary ß has an advantage $Adv^{DHP}_{Z_n,ß}(\rho)$ is given by,

$$Adv^{DHP}_{Z_n,ß}(\rho) = Prob[ß(N, g, g^a, g^b) = g^{ab}] \tag{3.4}$$

The $(t_{DHP}, \epsilon_{DHP}) - DHP$ assumption holds only if, $Adv^{DHP}_{Z_N,ß}(\rho) \leq \epsilon_{DHP}$, for very small $\epsilon_{DHP} > 0$ and its time is utmost t_{DHP}.

(t, ϵ)-hard n-IF-DHP

The DHP assumption holds when a t-polynomial time algorithm, suppose ß, which output $\gamma \in \{0, 1\}$, has an advantage ϵ in solving the DHP, if

$$\begin{aligned} Adv^{DHP}_{Z_n,ß}(\rho) :=& [P[ß(g^{rd}N, p_1, ..., p_n, g, g^k, g^x, g^{kr}, g^{xr}, g^d, g^{rd}) = 0] - \\ & [P[ß(g^{rd}N, p_1, ..., p_n, g, g^k, g^x, g^{kr}, g^{xr}, g^d, T) = 0]] \geq \epsilon \end{aligned} \tag{3.5}$$

3.3 CP-ABE SCHEME

It is defined using four algorithms, Setup, Key-Gen, Encrypt and Decrypt [9].

Definition of CP-ABE

- **Setup:** This inputs a security parameter κ along with universal set $U = \{attr_1, ..., attr_n\}$ and outputs a master public key MPK and a master secret key MSK.

- **KeyGen:** This algorithms inputs MPK, MSK, and a set of user attributes A and outputs a secret key k_u associated with A.

- **Encrypt:** It loads with MPK, a message M, and an access structure P. It returns a ciphertext C with the property that a user having k_u can decrypt C if and only if $A \models P$.

- **Decrypt:** It takes MPK, ciphertext C, which was encrypted using access policy P, and user secret key k_u corresponding to attribute set A. It returns originial messsage M, if k_u is associated with A, such that, $A \models P$, otherwise it return null $(\perp)$.

3.4 SELECTIVE GAME FOR CCA SECURE CP-ABE

The security of the proposed algorithm will be tested by the Indistinguishability - Chosen Ciphertext Attack (IND-CCA) security model. This is a game between a challenger (holding the security key) and an attacker (wants to break the algorithm), where the challenger generates a set of MPK and MSK on basis of some security bounds and hands over MPK to attacker retains the MSK. The attacker can get as many decryptions of chosen ciphertexts as he wants. The attacker submits two same length plaintexts $M0$ and $M1$ to the challenger and the challenger encrypts and returns the encryption of one of the messages at random, and the task of the attacker is to determine that if the encryption is of $M0$ or $M1$. There are two variants of IND-CCA, under the rst one the attacker can not ask for further decryptions and in the second one the attacker can ask for any number of decryptions but with the limitation that the challenge ciphertext should not be asked to be decrypted. The aim is to show mathematically that the attacker ßcan win this game (given that the underlying problem of the algorithm cannot be broken). Authenticated encryption primarily implies security against CCA. The theorem says: Let (E, D) be a cipher that provides authenticated encryption (AE). Then (E, D) is choosen ciphertext attack secure. Particularly, if we have a $q-$query are made by adversary ß, then we have efcient $b1$ and $b2$, such that :

$$Adv_{CCA}[A, E] \leq 2q * Adv_{CI}[b1, E] + Adv_{CPA}[b2, E] \tag{3.6}$$

$Adv_{CPA}[b2, E]$ is negligible as it is CPA secure [18] [9] [14], $2q * Adv_{CI}[b1, E]$ is also negligible since ciphertext integrity is maintained by the encryption scheme. Therefore $Adv_{CCA}[A, E]$ is also negligible, which implies the chance of winning of adversary also becomes negligible.

Chapter 4: PROPOSED ALGORITHM

4.1 KEY MANAGEMENT IN DEFINED ACCESS STRUCTURE

Based upon Harn et. al [22], which is an adaptation of Akl-Taylor's scheme [23], in this given section, we present the key management in the access structure used in this work. Also, it is proven that the given Harn et. al [22] scheme is secured against the key recovery attack.
Suppose,

$$Z_n \Rightarrow Congruence\, class's\, set\, for\, N$$

where N = pq, and congruence class contains integers, that give same modulo when divided by N.
p and q will be two sufficiently large enough primes chosen according to the RSA standards, this will follow that p will not be so close or too far from q, in order to make it computationally hard to guess p or q by factorizing N.

For a positive integer belonging to Z_n, we say that a & N are coprime, i.e. gcd(N,a)= 1, iff we can use the extended euclidean for finding out the multiplicative inverse of a (mod N). More formally, we can find out a value b which satisfies :

$$ab \equiv 1(modN) \tag{4.1}$$

where, 1 is the identity element under multiplication in Z_n.

The process for key management is :

1. For each i, Arbitrarily choose p_i (Part of RSA public key) in a way that ensures

   $$GCD(\phi(N), p_i) = 1$$

2. Calculate q_i so that,

   - $p_i q_i = 1(mod(\phi(N)))$ respective to each $A_i \in U$ and
   - $p_i \neq p_j \longleftrightarrow i \neq j$

3. Let,
$$\{\phi(N), q_1, \ldots, q_n\} \Rightarrow Secret\, Parameters$$
$$\{N, p_1, \ldots, p_n\} \Rightarrow Public\, Parameters$$
As, for the computation of $\phi(N)$, that is the totient function of N, we need to know the factors of N (i.e. p & q), that in-turn will give us $(p-1)(q-1)$. As factorization of N is hard so, we can't calculate $\phi(N)$ in this manner. And calculation of $\phi(N)$, without any information about p and q will be impracticable. So, this can be directly inferred that, finding q_i that satisfies the equation :
$$p_i q_i = 1(mod(\phi(N)))$$
will also be computationally hard.

4. Select a random g∈(2, N-1) and such that g & N are coprimes i.e. GCD(N,g) is 1.

5. Computation of K_A (secret key related to A) and K_P (secret key related to P) is done as :
$$K_A = g^{d_A}(modN) \tag{4.2}$$
$$K_P = g^{d_P}(modN) \tag{4.3}$$
Here, A corresponds to the Attributes. P corresponds to the Access Policy being used and $d_A = \prod_{i=1}^{n} q_i^{a_i},\ a_i \in A$ & $d_P = \prod_{i=1}^{n} q_i^{b_i},\ b_i \in P$.

Proposition : If we define :
$$e_A = \prod_{i=1}^{n} p_i^{a_i}$$
$$e_P = \prod_{i=1}^{n} p_i^{b_i}$$
and also,
$$K_P = K_A^{\frac{e_A}{e_P}}$$
We can say that A (Attribute set) fulfills P (Access Policy) iff $\frac{e_A}{e_P}$ (which equates to $\prod_{i=1}^{n} p_i^{a_i - b_i}$) is integral.

Proof :
Let's assume that the attribute set doesn't satisfy the access policy. As a_i and b_i can take only values 0 and 1, $a_i - b_i$ can only values from -1, 0 and 1. This further indicates that in $\frac{e_A}{e_P}$, we will encounter atleast one term of the form p_k^{-1}, (that is the inverse term), and computation of this inverse term will be computationally hard without the factorization of N into p and q. And hence, when the access policy is not satisfied, the fraction can not be integral.
Considering the other scenario, when the access policy is satisfied, the computation

of K_P goes as :

$$
\begin{aligned}
K_P &= K_A^{\frac{e_A}{e_P}} \,(mod\, N) \\
&= \left(g^{d_A} \,(mod\, N) \right)^{\frac{\prod_{i=1}^{n} p_i^{a_i}}{\prod_{i=1}^{n} p_i^{b_i}}} \,(mod\, N) \\
&= g^{d_A(\prod_{i=1}^{n} p_i^{a_i-b_i})}(mod\, N) \\
&= g^{(\prod_{i=1}^{n} q_i^{a_i})(\prod_{i=1}^{n} p_i^{a_i-b_i})}(mod\, N) \\
&= g^{(\prod_{i=1}^{n} q_i^{a_i-b_i+b_i})(\prod_{i=1}^{n} p_i^{a_i-b_i})}(mod\, N) \\
&= g^{(\prod_{i=1}^{n} q_i^{b_i})(\prod_{i=1}^{n} q_i^{a_i-b_i} p_i^{a_i-b_i})}(mod\, N) \\
&= g^{(\prod_{i=1}^{n} q_i^{b_i})(\prod_{i=1}^{n} (q_i p_i)^{a_i-b_i})}(mod\, N) \\
&= g^{\prod_{i=1}^{n} q_i^{b_i}} \,(mod\, N) \\
&= g^{d_P}(mod\, N)
\end{aligned}
$$

Hence, the result stands.

4.2 PROPOSED CP-ABE SCHEME

In this work, we have proposed an algorithm based upon scalable revocation using Chosen Ciphertext Attack (CCA)-secure CP-ABE scheme. It is an extension to Odelu et. al [10]. It achieves revocation with the assistance of a trusted proxy server. It allows user-based revocation.

4.2.1 Role of Proxy Server

A trusted proxy server assists in partial decryption by providing two proxy terms required to complete the decryption process. The proxy server contains a revocation list RL containing the list of revoked users, a list of revoked attributes and corresponding users from whom attributes have been revoked. The proxy server uses the RL and the user's secret key to compute two components named as C_{u_i} and C_{a_i}. It modifies the two components for revocation for a revoked user so that decryption fails. The non-revoked users can continue to access the ciphertext uninterruptedly without re-encryption of the ciphertext or re-distribution of the their keys. The setup(), keygen() and encrypt() phases are similar to the. Odelu et. al's [10] scheme.

4.2.2 Setup Stage

Inputs ⇒ ρ and U

Steps ⇒

I1. RSA Setup

- Select p & q such that p $\neq$ q

- Calculate the value of N as : $N = pq$
- Arbitrarily choose p_i (Part of RSA public key) in way that ensures

$$GCD(\phi(N), p_i) = 1$$

- Calculate q_i so that $p_i q_i = 1(mod(\phi(N)))$ respective to each $A_i \in U$
- Choose two private keys for the system, k & x in manner that following properties are satisfied :

$$GCD(\phi(N), k) = 1$$
$$GCD(k, q_i) = 1$$
$$GCD(x, q_i) = 1$$

- Choose a random g, such that $2 < g < N - 1$ & GCD(g,N)=1

I2. Hash Functions

- Choose 3 one-way collision free hash functions H_1, H_2 and H_3.

$$H_1 : \{0,1\}^* \to \{0,1\}^\rho$$
$$H_2 : \{0,1\}^* \to \{0,1\}^{l_\sigma}$$
$$H_3 : \{0,1\}^* \to \{0,1\}^{l_m}$$

where l_σ is the length of the arbitrary string under security parameters and l_m is length of plaintext message M.

I3. Public Parameters

- Public parameters are computed as follows:

$$D_U = g^{d_U} \tag{4.4}$$
$$Y = g^x \tag{4.5}$$
$$R = g^k \tag{4.6}$$

Here, d_U is $\prod_{A_i \in U} q_i$.

I4. Output

- Output the following master keys (secret and public)

$$MSK = \{k, x, p, q, q_1, \ldots, q_n\} \tag{4.7}$$
$$MSK = \{N, D_U, Y, R, H_1, H_2, H_3, p_1, \ldots, p_n\} \tag{4.8}$$

4.2.3 Encryption

Steps ⇒

Enc1. Pick a random $\sigma_m \in \{0,1\}^{l_\sigma}$ and compute r_m as :

$$r_m = H_1(P, M, \sigma_m)$$

Enc2. K_m computation

$$\begin{aligned} K_m &= D_U^{r_m \frac{e_U}{e_P}} \\ &= (g^{d_U})^{r_m \frac{e_U}{e_P}} \\ &= (g^{\prod_{A_i \in U} q_i})^{r_m \frac{\prod_{A_i \in U} p_i}{\prod_{A_i \in P} p_i}} \\ &= g^{r_m d_P} \end{aligned} \tag{4.9}$$

Enc3. Computation of Y_m, R_m. C_{σ_m}, C_m and S_m

$$Y_m = g^{xr_m} \tag{4.10}$$

$$R_m = g^{kr_m} \tag{4.11}$$

$$C_{\sigma_m} = H_2(K_m) \oplus \sigma_m \tag{4.12}$$

$$C_m = H_3(\sigma_m) \oplus M \tag{4.13}$$

$$S_m = H_1(\sigma_m, M) \tag{4.14}$$

Ultimately, the ciphertext will be :

$$C = \{P, Y_m, R_m, C_{\sigma_m}, C_m, S\}$$

4.2.4 KeyGen

Inputs ⇒ A (User's Attributes), MPK and MSK

Output ⇒ k_u

Steps ⇒

KG1. Computation of d_A

$$d_A = \prod_{i=1}^{n} q_i^{a_i} \tag{4.15}$$

where

$$a_i = \begin{cases} 1, & \text{if } A_i \in A \\ 0, & \text{if } A_i \notin A \end{cases}$$

KG2. Computation of K_u

- Pick r_u & t_u randomly.
- Get to a value s_u, in a manner that it satisfies :

$$d_A = ks_u + r_u x(mod\phi(N)$$

- Calculate k_1 and k_2 as :

$$k_1 = s_u + xt_u(mod\phi(N)) \tag{4.16}$$

$$k_2 = r_u + kt_u(mod\phi(N)) \tag{4.17}$$

KG3. Output

$$k_u = (k_1, k_2)$$

4.2.5 CASE 1: No Revocation

A. Proxy Server

Inputs ⇒ k_u and RL

Output ⇒ $PC = (C_{a_i}, C_{u_i})$

Steps ⇒

PS1. Proxy server checks the revocation list RL to see if the current user is revoked or not. If it is found to be legitimate user, then the following process takes place.

PS2. Computation of proxy components:

$$C_{u_i} = \lambda, \lambda \in \text{Random Number} \tag{4.18}$$

$$C_{a_i} = k_2 \times \lambda \tag{4.19}$$

PS3. Output

$$PC = (C_{a_i}, C_{u_i})$$

B. Decryption

Inputs ⇒ k_u, C and PC

Steps ⇒

Dec1. According to Proposition 4.1, if $P \subseteq A$, then only $\frac{e_A}{e_P}$ will result into an integer. Thus,

$$
\begin{aligned}
K_m &= \left(Y_m^{(\frac{C_{a_i}}{C_{u_i}})} R_m^{k_1} \right)^{\frac{e_A}{e_P}} \\
&= \left(Y_m^{(\frac{k_2 \times \lambda}{\lambda})} R_m^{k_1} \right)^{\frac{e_A}{e_P}} \\
&= \left(g^{x r_m (r_u - k t_u)} g^{k r_m (s_u + x t_u)} \right)^{\frac{e_A}{e_P}} \\
&= \left(g^{r_m (x r_u + k s_u)} g^{x r_m (-k t_u) + k r_m (x t_u)} \right)^{\frac{e_A}{e_P}} \\
&= (g^{r_m d_A})^{\frac{e_A}{e_P}} \\
&= g^{r_m d_P}
\end{aligned}
\tag{4.20}
$$

Otherwise, if $P \nsubseteq A$, then $\frac{e_A}{e_P}$ will not be an integer, and therefore computing K_m becomes computationally infeasible.

Dec2. Computation of σ'_m and M':

$$
\begin{aligned}
\sigma'_m &= H_2(K_m) \oplus C_{\sigma_m} \\
&= H_2(K_m) \oplus (H_2(K_m) \oplus \sigma_m)
\end{aligned}
\tag{4.21}
$$

$$
\begin{aligned}
M' &= C_m \oplus H_3(\sigma'_m) \\
&= (H_3(\sigma_m) \oplus M) \oplus H_3(\sigma'_m) \\
&= M
\end{aligned}
\tag{4.22}
$$

Des3. Computation of signature S_m:

$$
S_m = H_1(\sigma'_m, M') \tag{4.23}
$$

Des4. Verifying Signature:
If the equation 4.23 hold true than output the message M. Otherwise, if the equation 4.23 does not hold, then output null ($\perp$).

4.2.6 CASE 2: User Revocation

A. Proxy Server

Inputs ⇒ k_u and RL

Output ⇒ $PC = (C_{a_i}, C_{u_i})$

Steps ⇒

PS1. Proxy server checks the revocation list RL to see if the current user is revoked or not. If it is found to be revoked user, then the following process takes place.

PS2. Computation of proxy components:

$$C_{u_i} = \lambda_1 \tag{4.24}$$

$$C_{a_i} = k_2 \times \lambda_2 \tag{4.25}$$

where, $\lambda_1, \lambda_2 \in$ Random Number and $\lambda_1 \neq \lambda_2$

PS3. Output

$$PC = (C_{a_i}, C_{u_i})$$

B. Decryption

Inputs $\Rightarrow$ k_u, C and PC

Steps $\Rightarrow$

Dec1. According to Proposition 4.1, if $P \subseteq A$, then only $\frac{e_A}{e_P}$ will result into an integer. Thus,

$$\begin{aligned} K_m &= \left(Y_m^{(\frac{C_{a_i}}{C_{u_i}})} R_m^{k_1} \right)^{\frac{e_A}{e_P}} \\ &= \left(Y_m^{(\frac{k_2 \times \lambda_2}{\lambda_1})} R_m^{k_1} \right)^{\frac{e_A}{e_P}} \\ &\neq g^{r_m d_P} \end{aligned} \tag{4.26}$$

Dec2. Since $K_m \neq g^{r_m d_P}$, due to λ_1 does not cancel out λ_2. Thus, a revoked user will not be able to fetch the message M.

Chapter 5: EXPERIMENTAL RESULTS AND ANALYSIS

5.1 EXPERIMENTAL ENVIRONMENT

In this work, we first implemented Odelu et. al [10] CP-ABE scheme using CP-ABE toolkit [24]. Then we added the revocation phases into it. The large number are handled using GMP GNU Bignum library [25]. We have implemented Pro-SRCC [18] using PBC Library [26] and GMP Library [25]. The PBC Library [26] and GMP Library [25] works as a backbone for all pairing based crypto-systems. The GMP Library works efficiently with signed and floating point numbers.

Table 5.1 gives the setup requirements used to run the above CP-ABE schemes.

Table 5.1: Experiment Environment

Hardware Requirements	• RAM with 2048 MB or more • Intel Dual Core Processor with 1.7GHz or faster processor • Disk Space with 2MB or above
Software Requirements	• 64-bit Ubuntu • GMP Library [25] • PBC Library [26] • CP-ABE toolkit [24]

5.2 SIMULATION AND OUTPUT

5.2.1 PBC Library

The PBC (Pairing-Based Cryptography) library is a C library (released under the GNU Lesser General Public License) built on the GMP library that performs the mathematical operations underlying pairing-based cryptosystems.

This library forms the base for building any cryptosystems as it enables us to work on pairing and arithmetic functions more efficiently.

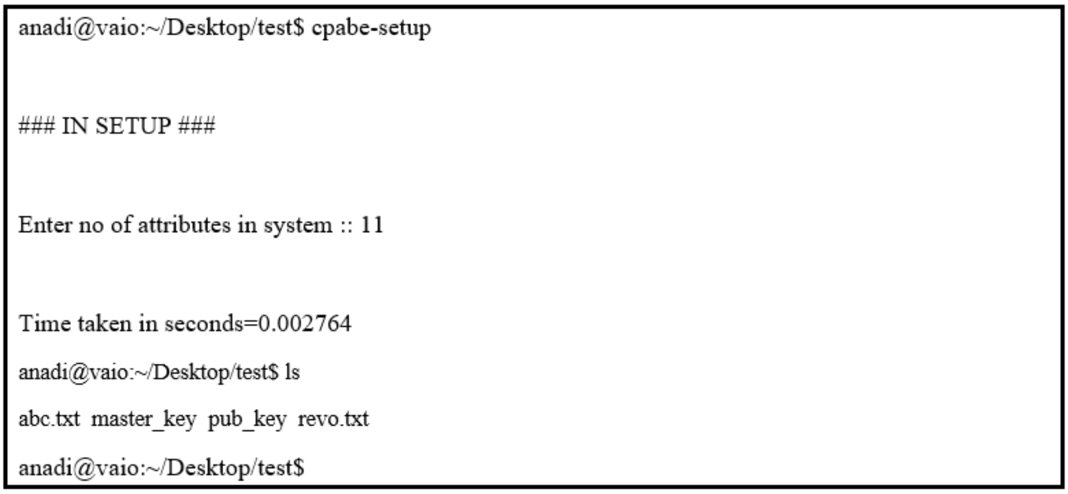
5.2.2 Setup

It takes the number of attributes as an input and produces two keys, a master public key MPK, and a master secret key MSK and also an empty revocation list.

Syntax: $cpabe−setup

Enter the number of attributes in system:: n

Enter the number of attributes on prompt. Suppose number of attributes is 11. After above command run, it will create master public key and master secret key for 11-attributes and then save it in serialize for in file as shown in Fig 5.1. Also, it create a new text file named revo.txt which contains the revocation list which is initially empty.



```
anadi@vaio:~/Desktop/test$ cpabe-setup


### IN SETUP ###


Enter no of attributes in system :: 11


Time taken in seconds=0.002764
anadi@vaio:~/Desktop/test$ ls
abc.txt  master_key  pub_key  revo.txt
anadi@vaio:~/Desktop/test$
```

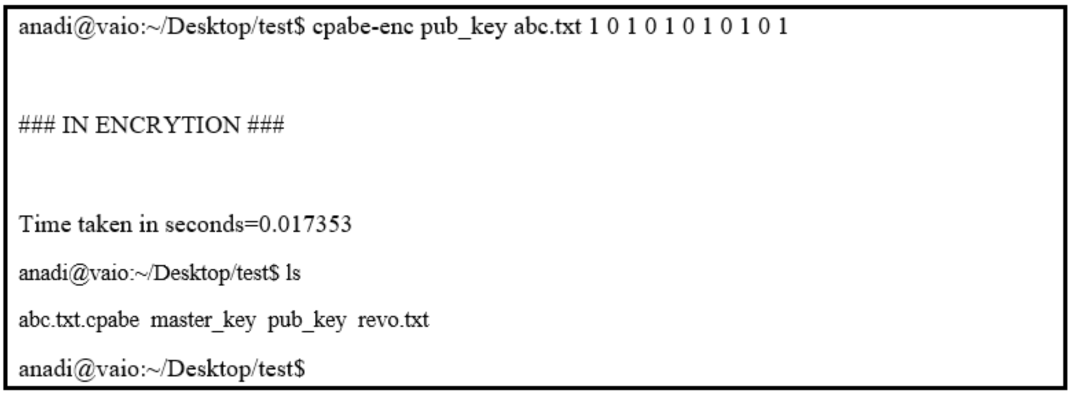
Figure 5.1: Setup

5.2.3 Encrypt

It takes an input MPK, message text file and access structure P generates ciphertext according to the algorithm provided above and encrypts the file according to the access structure.

Syntax : $cpabe-enc pub_key plaintext_file_name [Policy]

Policy are in the form of 1s (ON) or 0s(OFF) as shown in Fig 5.2.



```
anadi@vaio:~/Desktop/test$ cpabe-enc pub_key abc.txt 1 0 1 0 1 0 1 0 1 0 1


### IN ENCRYTION ###


Time taken in seconds=0.017353

anadi@vaio:~/Desktop/test$ ls

abc.txt.cpabe  master_key  pub_key  revo.txt

anadi@vaio:~/Desktop/test$
```

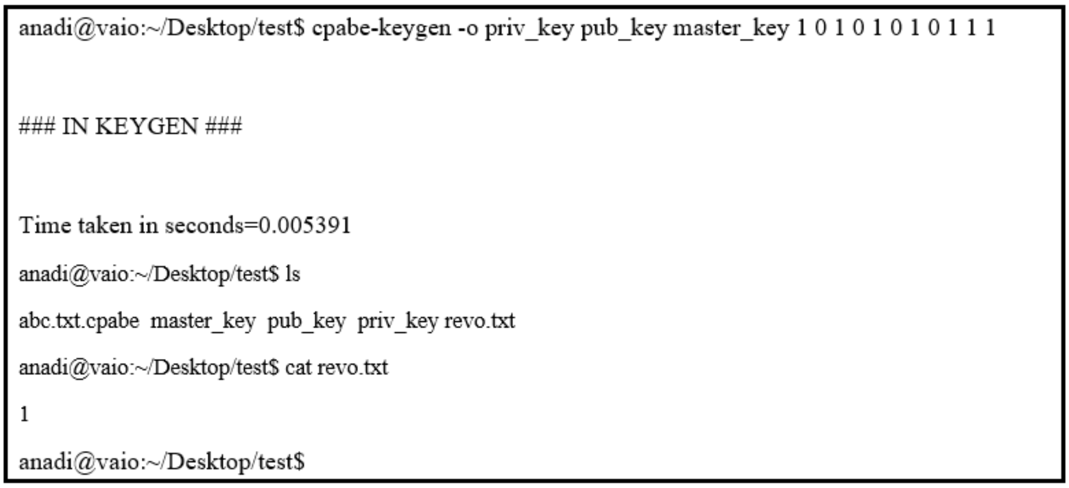
Figure 5.2: Encryption

5.2.4 Key Generation

It generates a user secret key k_u using master public key MPK, master secret key MSK and attributes. Also, it assigns a user id to the user calling keygen and writes to the revocation list in revo.txt.

Syntax : $cpabe-keygen −o priv_key pub_key master_key [Attributes]

Attributes are in the form of 1s(ON) or 0s(OFF) as shown in Fig 5.8.



```
anadi@vaio:~/Desktop/test$ cpabe-keygen -o priv_key pub_key master_key 1 0 1 0 1 0 1 0 1 1 1


### IN KEYGEN ###


Time taken in seconds=0.005391

anadi@vaio:~/Desktop/test$ ls

abc.txt.cpabe  master_key  pub_key  priv_key revo.txt

anadi@vaio:~/Desktop/test$ cat revo.txt

1

anadi@vaio:~/Desktop/test$
```

Figure 5.3: Key Generation

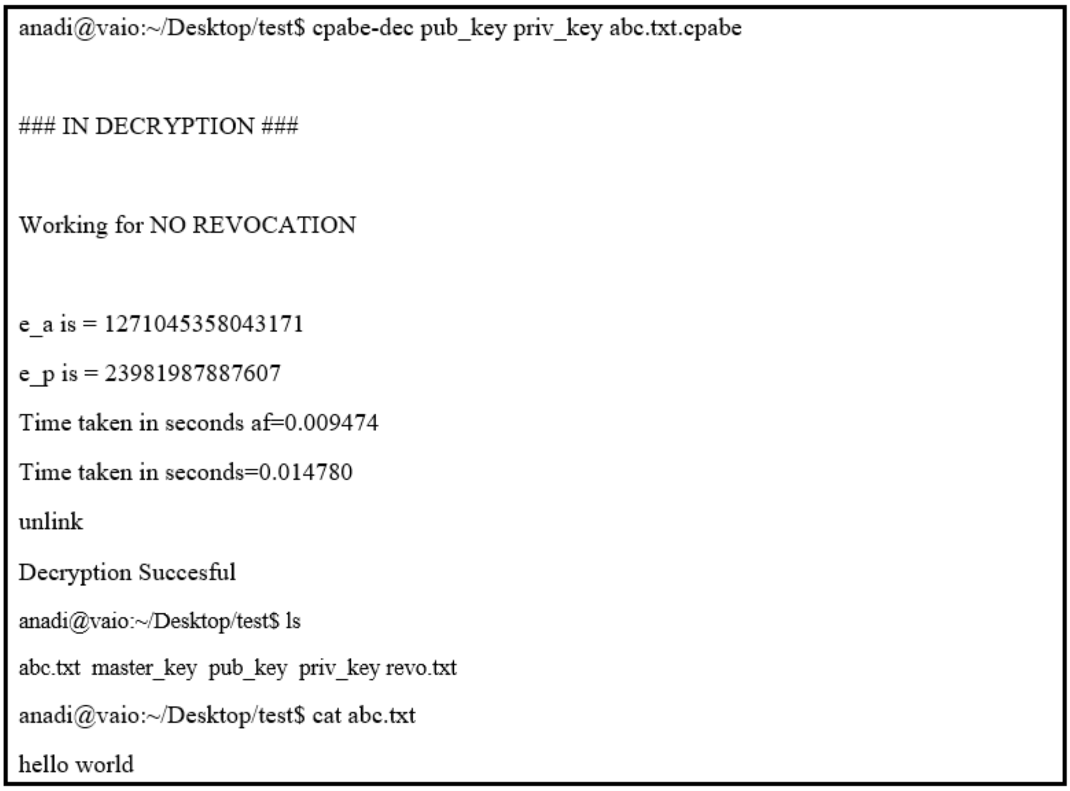
5.2.5 Decrypt

It takes MPK, ciphertext C, which is encrypted by access structure P as inputs. It returns plaintext message M if user attribute list A, satisfies $P \subseteq A$.

Initially it request proxy server for the proxy component PC. Proxy server checks if the access structure of the user and access structure related to the ciphertext are equal or not at the proxy server. Even if the revoked user tries to decrypt message, he cannot complete the full decryption by-self since he/she will not get the proxy components from the proxy server.

In this scheme, proxy server provides two proxy elements $PC = (C_{u_i}, C_{a_i})$ to complete the decryption process. It performs partial decryption. It consist of a revocation list which includes a list of users that are genuine. This list is represented by RL, Revocation List. If a user is present in the revocation list and policy is subset of the user's attribute than the ciphertext is decrypted and original message is produced as shown in Fig 5.4.

Syntax : \$ cpabe-dec pub_key priv_key file.cpabe



```
anadi@vaio:~/Desktop/test$ cpabe-dec pub_key priv_key abc.txt.cpabe


### IN DECRYPTION ###


Working for NO REVOCATION


e_a is = 1271045358043171
e_p is = 23981987887607
Time taken in seconds af=0.009474
Time taken in seconds=0.014780
unlink
Decryption Succesful
anadi@vaio:~/Desktop/test$ ls
abc.txt  master_key  pub_key  priv_key revo.txt
anadi@vaio:~/Desktop/test$ cat abc.txt
hello world
```

Figure 5.4: Decryption

5.2.6 Potential Scenarios

There can be 3 probable scenarios that can take place in the revocation environment which are:

1. **No Revocation:** Normal decryption where no revocation takes place for a genuine user who is trying to decode the original message.

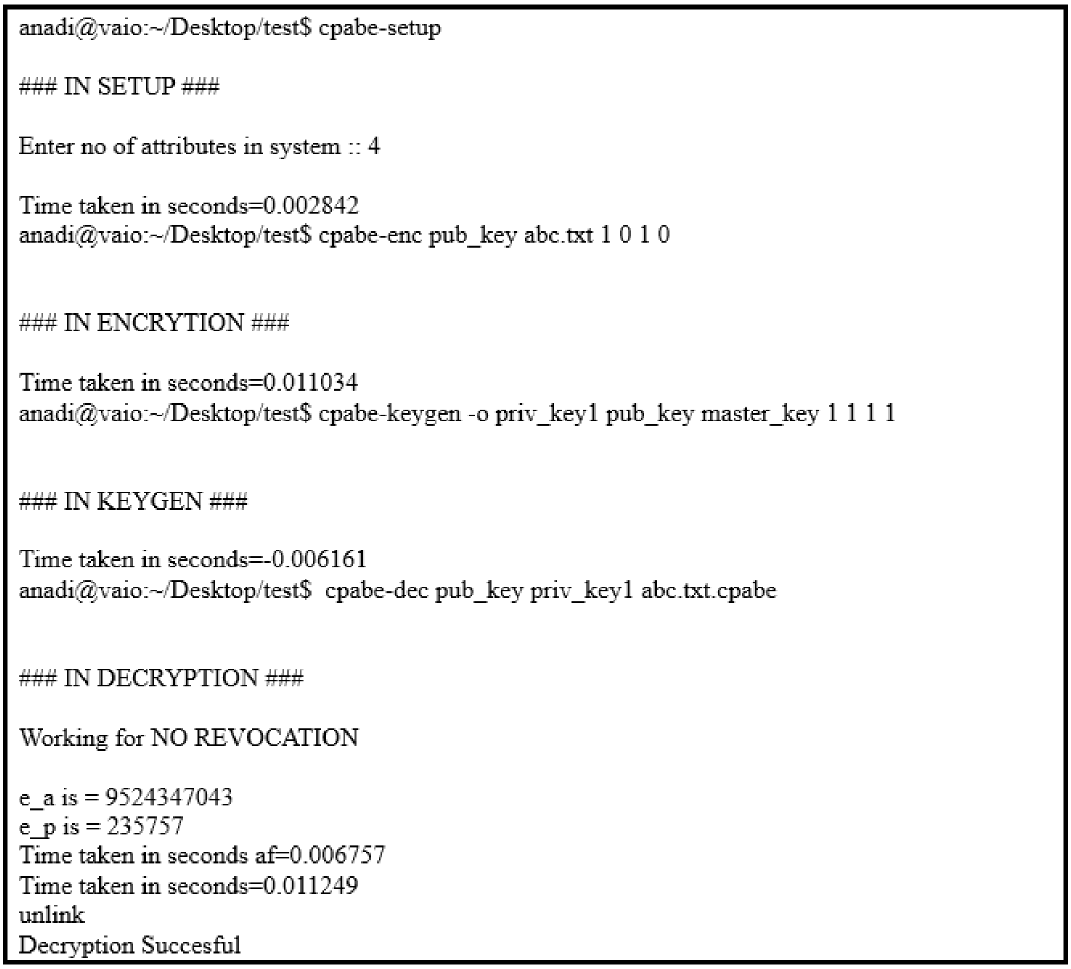
2. **User Revocation:** A user that is revoked is trying to decrypt the decoded original message.

3. **Invalid user:** An adversary is trying to decode the original message.

Our proposed scheme can handle all the above cases efficiently and effectively as given below.

Scenario 1: No Revocation

If the user, is a legitimate user and if user's attribute list A, satisfies $P \subseteq A$, then ciphertext is decrypted successfully and results into M, as shown in Fig 5.5

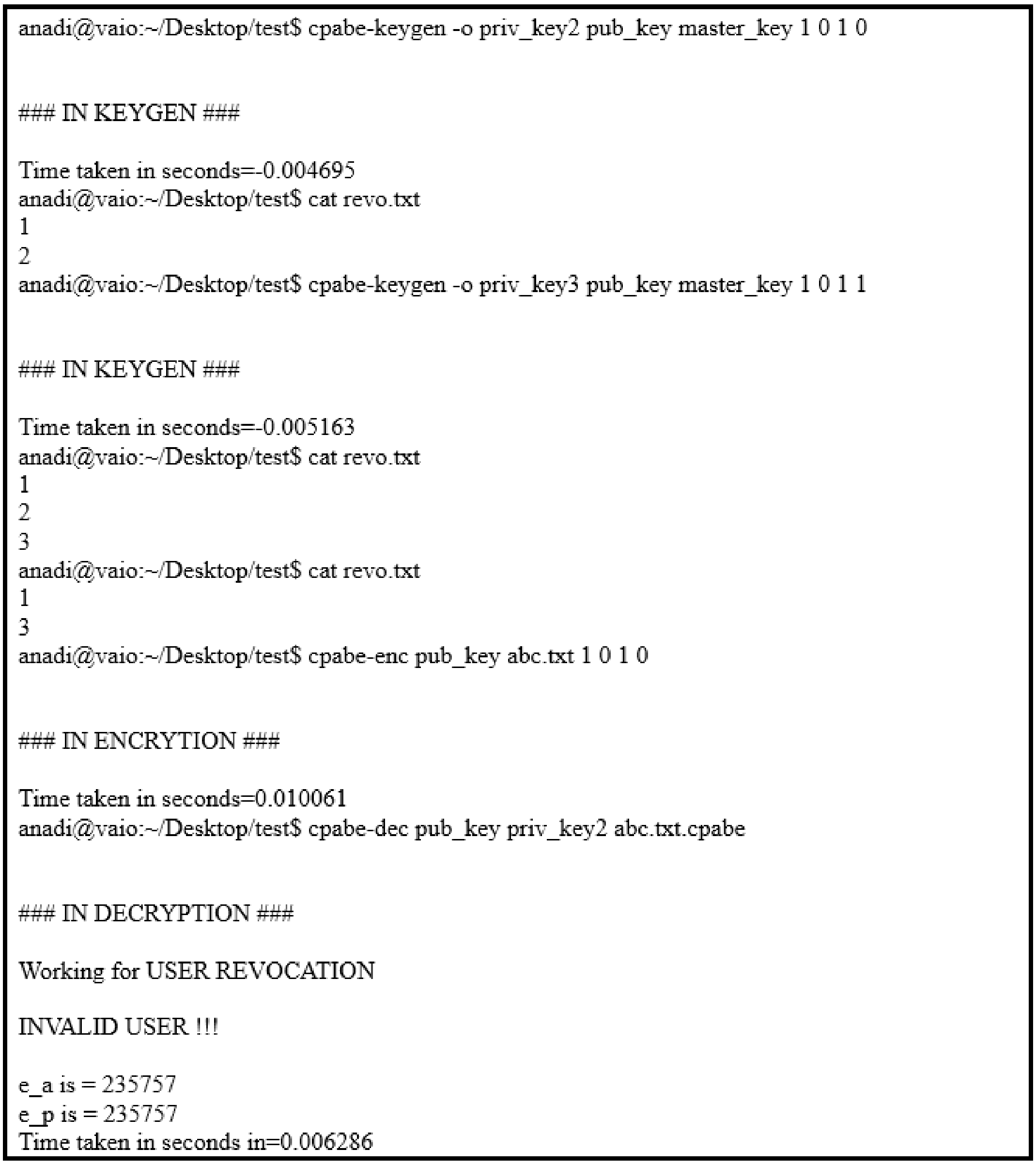


```
anadi@vaio:~/Desktop/test$ cpabe-setup

### IN SETUP ###

Enter no of attributes in system :: 4

Time taken in seconds=0.002842
anadi@vaio:~/Desktop/test$ cpabe-enc pub_key abc.txt 1 0 1 0


### IN ENCRYTION ###

Time taken in seconds=0.011034
anadi@vaio:~/Desktop/test$ cpabe-keygen -o priv_key1 pub_key master_key 1 1 1 1


### IN KEYGEN ###

Time taken in seconds=-0.006161
anadi@vaio:~/Desktop/test$  cpabe-dec pub_key priv_key1 abc.txt.cpabe


### IN DECRYPTION ###

Working for NO REVOCATION

e_a is = 9524347043
e_p is = 235757
Time taken in seconds af=0.006757
Time taken in seconds=0.011249
unlink
Decryption Succesful
```

Figure 5.5: Scenario 1: No revocation

Scenario 2: User Revocation

If the user, is a revoked user who has genuine secret key with user's attribute list A, satisfying $P \subseteq A$ but whose user id is not present in the revocation list tries to decode the original message then, ciphertext is not decrypted and results into error message, as shown in Fig 5.6

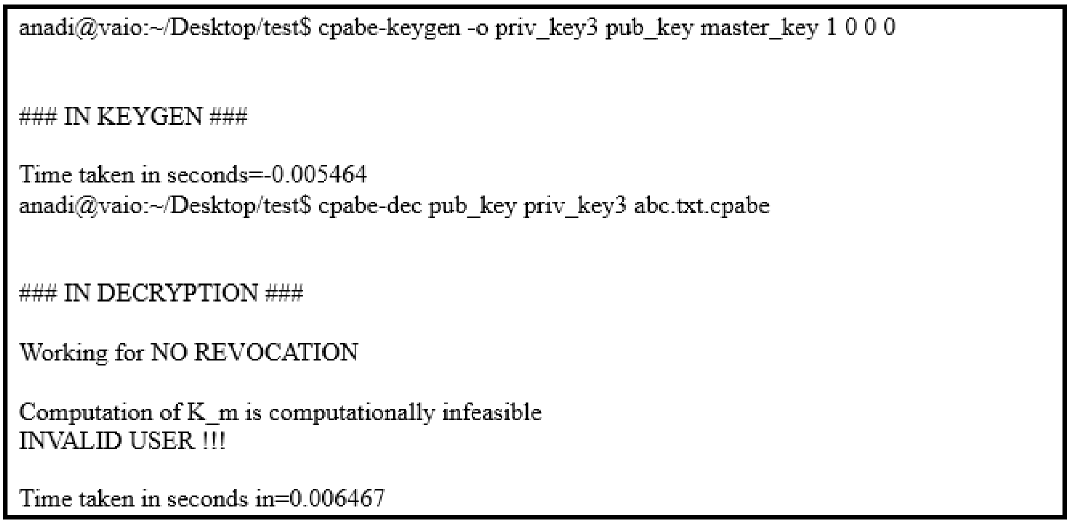


```
anadi@vaio:~/Desktop/test$ cpabe-keygen -o priv_key2 pub_key master_key 1 0 1 0


### IN KEYGEN ###

Time taken in seconds=-0.004695
anadi@vaio:~/Desktop/test$ cat revo.txt
1
2
anadi@vaio:~/Desktop/test$ cpabe-keygen -o priv_key3 pub_key master_key 1 0 1 1


### IN KEYGEN ###

Time taken in seconds=-0.005163
anadi@vaio:~/Desktop/test$ cat revo.txt
1
2
3
anadi@vaio:~/Desktop/test$ cat revo.txt
1
3
anadi@vaio:~/Desktop/test$ cpabe-enc pub_key abc.txt 1 0 1 0


### IN ENCRYTION ###

Time taken in seconds=0.010061
anadi@vaio:~/Desktop/test$ cpabe-dec pub_key priv_key2 abc.txt.cpabe


### IN DECRYPTION ###

Working for USER REVOCATION

INVALID USER !!!

e_a is = 235757
e_p is = 235757
Time taken in seconds in=0.006286
```

Figure 5.6: Scenario 2: User revocation

Scenario 3: Invalid user

If the user, is an invalid user/ adversary who does not have the genuine secret key,that is, if user's attribute list A, does not satisfies $P \nsubseteq A$ and also, whose user id is not present in the revocation list tries to decode the original message then, ciphertext is not decrypted and results into error message, as shown in Fig 5.7



```
anadi@vaio:~/Desktop/test$ cpabe-keygen -o priv_key3 pub_key master_key 1 0 0 0


### IN KEYGEN ###

Time taken in seconds=-0.005464
anadi@vaio:~/Desktop/test$ cpabe-dec pub_key priv_key3 abc.txt.cpabe


### IN DECRYPTION ###

Working for NO REVOCATION

Computation of K_m is computationally infeasible
INVALID USER !!!

Time taken in seconds in=0.006467
```

Figure 5.7: Scenario 3: Invalid user

5.3 COMPARATIVE RESULT

In this section we compare our scheme with the existing CP-ABE schemes, such as, Emura et.al [9], SPIRC [19], and ProSRCC [18]. The comparisons are made on the following grounds:

5.3.1 Access Policy

Table 5.2 depicts the access policies defined in exsisting scheme, such as, Emura et.al [9], SPIRC [19], ProSRCC [18] and our scheme. Access policy is defined as a set of attributes that allows decryption of ciphertext.

Table 5.2: Access policies

Scheme Name	**Access Policy**
Emura [9]	AND-gates on multivalued attributes
SPIRC [19]	Tree-based Access Structure
ProSRCC [18]	AND-gates on multivalued attributes
Our Scheme	AND-gate

5.3.2 Size of each entity

We can compare the size of the different scheme by elements of the bilinear group used in different entities such as public key, master key, secret key, and ciphertext. n denotes the number of attributes. S are the number of least interior nodes that are satisfying the access structure, including root node in tree like access structure. m denotes the size of verification key. G_1, G_2, and G_T are three multiplicative cyclic groups of order p. G prime order pairing Z_N. N is RSA modulus such that $N = pq$. L is the length of plaintext. Table 5.3 provides us with such analysis theoretically. From table 5.3 we can see that, the length of ciphertext and secret key is independent of the number of attributes, and also, the length is constant.

Table 5.3: Comparison of storage

Scheme Name	**Public Key**	**Master Key**	**Secret Key**	**Ciphertext**
Emura [9]	$(2n+3)G_1 + G_T$	$(n+1)Z_p$	$2G_1$	$2G_1 + G_T$
SPIRC [19]	$2G_1+G_2+G_T$	$G_1 + Z_p$	$G_2+(a+G_1+2G_2)n$	$(2n+1)G_1 + G_2$
ProSRCC [18]	$(2n+3)G_1 + G_T$	$(n+1)Z_p$	$2G_1$	$2G_1 + G_T$
Our Scheme	$(3+n)G$	$(4+n)G$	$2G$	$3G + L$

5.3.3 Computation Overhead

The theoretical comparison of the encryption, as well as decryption times for the same schemes, can be found in Table 5.4. T_{G_1}, T_{G_T} are time required to excute an exponential in group G_1 and G_T that are two multiplicative cyclic groups of order p. T_{Z_N} is the time taken to execute exponential function in multiplicative cyclic group Z_N. e is the time required to compute bilinear map operation.

Table 5.4: Comparison for computation time

Scheme Name	**Encryption Time**	**Decryption Time**
Emura [9]	$(n+1)T_{G_1} + 2T_{G_T}$	$2e + 2T_{G_T}$
SPIRC [19]	$(2n+1)T_{G_1} + T_{G_T}$	$3ne + (2\|S\|+2)T_{G_T}$
ProSRCC [18]	$(n+1)T_{G_1} + T_{G_T}$	$2e + 2T_{G_T}$
Our Scheme	$3T_{Z_N}$	$3T_{Z_N}$

5.3.4 Performance Graph

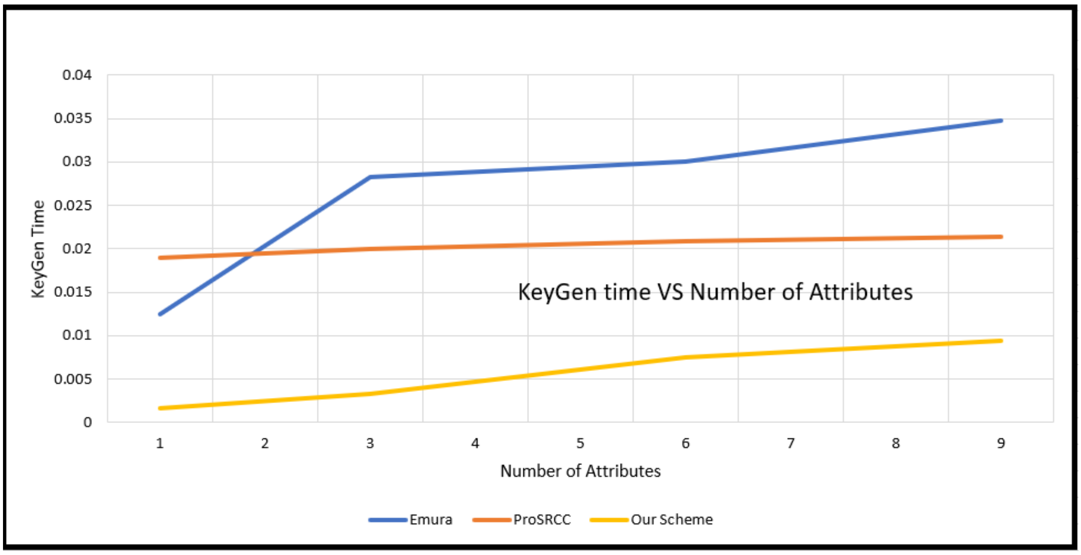


Figure 5.8: KeyGen time vs number of attributes

The performance graphs in Fig 5.8, 5.9, and 5.10 illustrate, the time required by the different schemes Emura et.al [9], ProSRCC [18], and our proposed revocation scheme with constant ciphertext length for key generation, encryption, and decryption respectively. The respective graphs compare the respective key generation process time variation with respect to change in the number of attributes. It is clearly evident from Fig. 5.8 that our scheme takes very less computation overhead as compared to rest CP-ABE scheme.

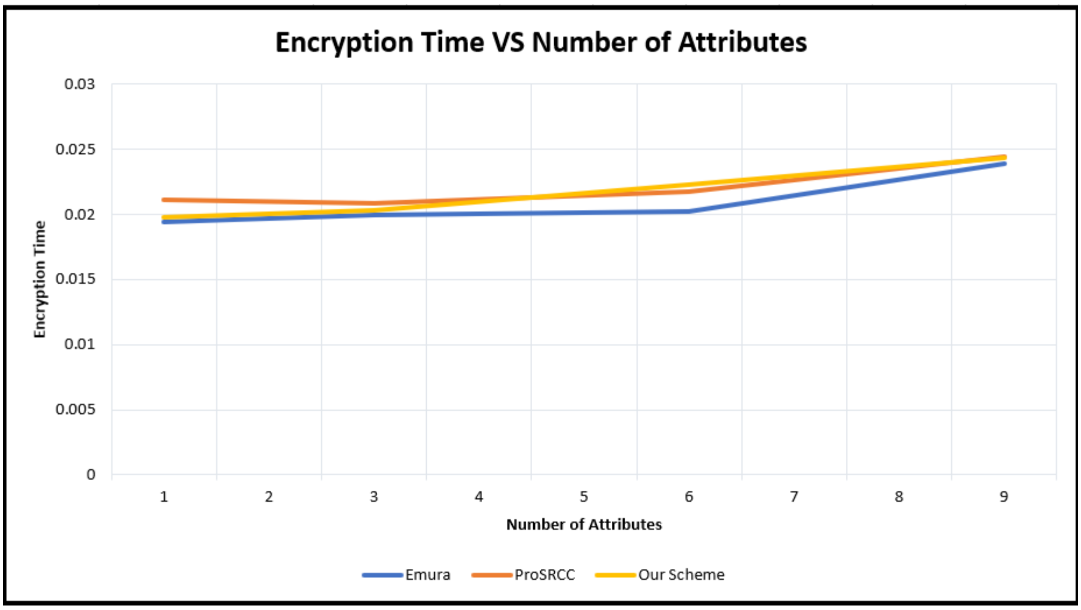


Figure 5.9: Encryption time vs number of attributes

We find that our scheme takes more time to encrypt compared to ProSRCC [18], EMURA et al. [9]. Since our scheme also computes S_m used to verify a digital signature, this makes it more time comsuming than the other two schemes.

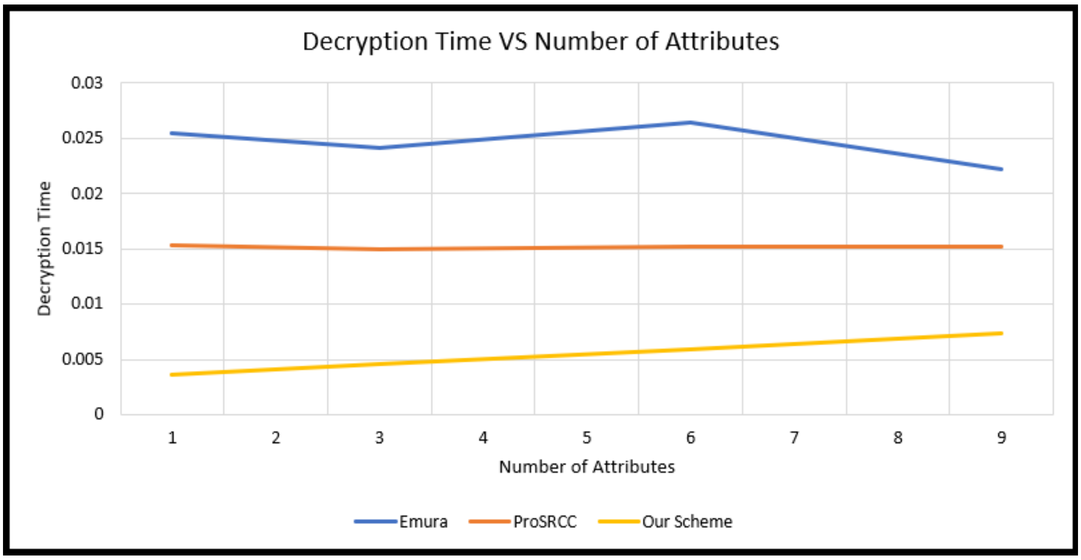


Figure 5.10: Decryption time vs number of attributes

Our scheme is much faster than the Emura et al. [9] scheme, and ProSRCC [18] scheme in terms of dercyption. Since it does not uses any bilinear pairing overhead thus the cost of computation of the system is evidently less than other three schemes.

Thus, overall our scheme proves to have an expressive access policy that results in efficient key generation, encryption, and decryption for battery-limited mobile devices without using bilinear pairing.

5.3.5 Performance comparison.

In this section, we compare our proposed scheme with CP-ABE scheme given by Odelu et. al [10]. For the comparative environment the parameters are given in table 5.5.

Table 5.5: Parameters used comparative environment

Parameter	**Value**
n	1000
$\|P\|$	500
$\|A\|$	600

Table 5.6 gives the comparison of the execution time for encryption and decryption in both the scheme. Our proposed scheme takes a slightly longer than Odelu et. al [10] because of the revocation cases it need to check.

Table 5.6: Execution time comparison

Scheme	**Encryption (in sec)**	**Decryption (in sec)**
Odelu et. al [10]	0.011742	0.009761
Our Scheme	0.010587	0.016133

5.4 SECURITY ANALYSIS

This section gives the security analysis for the proposed work against the different possible attacks. The CP-ABE scheme focuses more on the indistinguishability of message and security from the collision attack, that is, the attackers are not able to develop a new secret key by combining multiple secret keys. We also take into account the basics of linear equations, to prove our theories. We also use the assumption of computationally hard problems, such as RSA modulus $N = pq$, Deffie-Hellman Problem discussed in section 3.2.2 and integer factorization problem 3.2.1 for the same.

Theorem 1: Our work is secure against the collision attack for deriving the system private (x, k)

Proof: Let $user^i$, where $i = 1, ..., m$, be a group of user, each having their own attribute set A^i. The group of user, $user^i$ collaborates together using their secret key pair $k^i_{user} = (k^i_1, k^i_2)$, to derive the system secret pair (x, k), where

$$k^i_1 = s_{user^i} + xt_{user^i} \ (mod\phi(N)) \tag{5.1a}$$

$$k^i_2 = r_{user^i} + kt_{user^i} \ (mod\phi(N)) \tag{5.1b}$$

From *KeyGen* defined in section 4.2.4, we have

$$d_{A^i} = ks_{user^i} + xr_{user^i} \ (mod\phi(N)) \tag{5.2}$$

If s_{user^i} and r_{user^i} is known in equation 5.2, then the value of above equations are solvable and hence we can calculate the value of (x, k).

However, in-order to solve for x in equation 5.1a we need to guess two unknowns s_{user^i} and t_{user^i}, and in-order to solve k in equation 5.1b, we need to guess two unknowns r_{user^i} and t_{user^i}. Thus after randomly guessing both equation 5.1a and 5.1b are solvable but can have infinitely many solution. Therefore, also after collision of secret keys, the attacker will be unable to fetch system secret key pair (x, k).

Theorem 2: Our work is secure against an adversary trying to derive a valid user secret key $k_u = (k_1, k_2)$ **which corresponds to an attribute set** A^i

Proof: The adversary ß can randomly choose the values for r_{u^i} and t_{u^i} and then calculate the value of s_{u^i}, such that it satisfies the equation $d_{A^i} = ks_{u^i} + xr_{u^i} \ (mod\phi(N))$. However, adversary ß needs to know the key pair (x, k) and the RSA modulus d_A

for the calculation of the value of s_{u^i}.

From **Theorem 1**, we can say that, it is computationally infeasible for an adversary ß to generate system secret key pair (x, k). Thus, this follows that is its impossible to compute the valid secret key pair $k_u = (k_1, k_2)$. Moreover, it is also impossible to compute the value of k_u because of the RSA modulus d_A, which states the computationally infeasibility of Integer Factorization problem 3.2.1 since it depends upon the solving of the Euler's totient function $\phi(N) = (p-1)(q-1)$.

Theorem 3: Given ciphertext $C = \{P, Y_m, R_m, C_{\sigma_m}, C_m, S_m\}$, our work is secure against an adversary and also a valid user trying to derive key K_m which corresponds to an attribute set A_i, such that $P \nsubseteq A$

Proof: Given ciphertext $C = \{P, Y_m, R_m, C_{\sigma_m}, C_m, S_m\}$ to decrypt, such that $P \nsubseteq A$ and $k_{user} = (k_1, k_2)$ be the secret key pair associates with the attribute set A_i. Then,

$$\begin{aligned} Y_m^{k_2} R_m^{k_1} &= g^{xr_m(r_{user} - kt_{user})} g^{kr_m(u_{user} + xt_{user})} \\ &= g^{r_m(xr_{user} + ks_{user})} g^{xr_m(-kt_{user}) + kr_m(xt_{user})} \\ &= g^{r_m d_A} \end{aligned} \tag{5.3}$$

However, it is computationally impossible to calculate the value of K_m, where $K_m = (g^{r_m d_A})^{\frac{e_A}{e_P}}$ as discussed in the Proposition given in section 4.1, without solving Integer Factorization Problem (IFP) 3.2.1 since in obove case, the value of $\frac{e_A}{e_P}$ will not be an integer value.

Therefore, our work is secured against the decrypting unauthorized ciphertext by a user *user* or an adversary ß.

Theorem 4: Given ciphertext $C = \{P, Y_m, R_m, C_{\sigma_m}, C_m, S_m\}$, our work is secure against a group of unauthorized users $user^i$ trying to derive key K_m which corresponds to an attribute sets A^i, such that $P \nsubseteq A^i, i = 1, 2, ..., m$

Proof: We start to prove the theorem for two users and then it can be extended to any number of users. Let $user_1$ and $user_2$ be the two unauthorized users, associated with the access structure R and S, respectively, such that $P \nsubseteq R$, $P \nsubseteq S$ but $p \subseteq (R \text{ OR } S) = Q$. According to **Theorem 2**, we can say that both the unauthorized user $user_1$ and $user_2$, will not be able to derive a valid secret key pair k_u associated with access policy structure Q, such that $P \subseteq Q$. But using their own secret keys k_{user_1} and k_{user_2}, they can produce $g^{r_m d_R}$ and $g^{r_m d_S}$.

Suppose $g_1 = g^{r_m} = (g^{r_m d_R})^{e}_{R}$, then we have,

$$g_1^{d_R} = g^{r_m d_R} \tag{5.4a}$$

$$g_1^{d_S} = g^{r_m d_S} \tag{5.4b}$$

A adversary ß can calculate the value of K_m, if he/she can solve the DHP problem, as follow:

$$g_1^{d_R d_S} \leftarrow DHP(g_1, g_1^{d_R}, g_1^{d_S}) \tag{5.5}$$

$$K_m = \left((g_1^{d_R d_S})^{e_T} \right)^{\frac{e_Q}{e_P}} \tag{5.6}$$

where $T = (R \text{ AND } S)$

For better understanding lets take an example. Let us defines a universal attribute set $U = A_1, A_2, A_3, A_4$ with four attribute A_1, A_2, A_3, A_4. Suppose $R = 0110, S = 1010$ and $P = 1100$. Thus, $Q = (R \text{ OR } S) = 1110$ and $T = (R \text{ AND } S) = 0010$. We see that, $P \nsubseteq R$, $P \nsubseteq S$, and $P \subseteq Q$.Then, above equation 5.5 becomes:

$$g_1^{(q_2q_3)(q_1q_3)} \leftarrow DHP(g_1, g_1^{(q_2q_3)}, g_1^{(q_1q_3)})$$

And, from equation 5.6 the value of K_m is derived as:

$$\begin{aligned} K_m &= \left((g_1^{(q_2q_3)(q_1q_3)})^{p_3} \right)^{\frac{p_1p_2p_3}{p_1p_2}} \\ &= ((g_1^{(q_1q_2q_3)})^{p_3}) \\ &= g_1^{(q_1q_2)} \\ &= g_1^{d_P} \\ &= g^{r_m d_P} \end{aligned}$$

Solving DHP problem in group Z_N is same as solving IFP for RSA modulus $N = pq$ which is computationally hard problem, thus the group of unauthorized users collaborated together cannot derive K_m, such that $P \nsubseteq R$ and $P \nsubseteq S$ under the DHP assumption.

Theorem 5: Our work satisfies the indistinguishablity of messages under the chosen ciphertext attack (CCA) and the n-IF-DHP assumption

Proof: Let us assume that the adversary ß wins the selective CCA game for our proposed work (t, q_e, q_c, ϵ) with an advantage ϵ^* in time t^*.

Here,

$$t_* = t + \mathcal{O}(q_c t_c + q_e t_{inv} + q_{H_1} t_{exp}) \tag{5.7}$$

$$\epsilon_* = \frac{1}{q_c + g_{H_2}} \left(\epsilon - \frac{q_{H_1}}{N} \right) \tag{5.8}$$

$$n = |U| \tag{5.9}$$

where,

$$t_c = \text{time taken to respond to a decryption query}$$

$$t_{inv} = \text{average time taken for group inverse}$$

$$t_{exp} = \text{average time taken to compute exponential operations}$$

$$q_{H_1} = \text{total number of queries addressed to Oracle } H_1$$

$$q_{H_2} = \text{total number of queries addressed to Oracle } H_2$$

$$|U| = \text{total number of attributes in universal set } U$$

We proceed with contradiction proof method in-order to prove our work [2], [14], [27]. We construct an algorithm κ, such that it can break the DHP assumption with the advantage $\epsilon^* = \frac{1}{q_c + g_{H_2}} \left(\epsilon - \frac{q_{H_1}}{N} \right)$, where $N = \prod_{i=1}^{n} n_i$ which is the total number of access structure that the adversay ß can express.

Stage 1: Given below is the definition of the three random hash Oracles used by the adversary ß :

$$\begin{aligned} H_1 := \text{query} &=> H_1(P_i, M_i, t_i) \\ \text{response} &=> d_i \in \{0,1\}^{\rho} \end{aligned} \tag{5.10a}$$

$$\begin{aligned} H_2 := \text{query} &=> H_2(K_m) \\ \text{response} &=> J_i \in \{0,1\}^{l_{\sigma m}} \end{aligned} \tag{5.10b}$$

$$\begin{aligned} H_3 := \text{query} &=> H_3(t_i) \\ \text{response} &=> K_i \in \{0,1\}^{l_m} \end{aligned} \tag{5.10c}$$

Stage 2: The adversary ß makes some queries for fetching the secret key pair and in response he gets the valid secret key pair. Let d_i be a random number used to generate ciphertext, if for any decrypt query on $E[P_i, M_i]$, $\exists(P_i, M_i, t_i, J_i, K_i)$ in the list of query, than, the output of the decrypt query will be M_i, otherwise it will return null ($\perp$). An assumption is made, that no decrypt query will return $\perp$ (gets aborted) because all the response of the queries will consists d_i, and also, because of hash oracles that, need to send responses to all the valid encryptions.

Stage 3: The adversary ß produces two challenge messages $\{M0, M1\}$ and sends it to challenger. The challenger responds with a challenge ciphertext C_{c^*} associated with the access structure P^*, such that P^* cannot be satisfied with any secret key queries previously made. Mathematically it can be explained as:

- Select

$$J^* \in \{0,1\}^{l_{\sigma m}}$$

$$K^* \in \{0,1\}^{l_m}$$

$$L^* \in \{0,1\}^{\rho}$$

- Select a random number r_m^* such that,

$$r_m^* \in \{0,1\}^{\rho}$$

- Calculate challenge ciphertext as,

$$C_{c^*} = \{P^*, Y^*_m, R^*_m, C^*_{\sigma_m}, C^*_m, S^*_m\} \tag{5.11}$$

where,

$$\begin{aligned} Y^*_m &= g^{xr^*_m} \\ R^*_m &= g^{kr^*_m} \\ C_{\sigma_m} &= J^* \\ C^*_m &= K^* \\ S^*_m &= L^* \end{aligned}$$

for access structure P^* this gives a valid encryption.

Stage 4: The adversary ß outputs a guess c^*_g of c^*. In the above case the challenge ciphertext C_{c^*} is indistinguiable from the real ciphertext. Thus, the adversary ß wins the Choosen Ciphertext Attack (CCA) game if $c^*_g = c^*$. Otherwise, a random group element is generated, that is, $I = g^{r^*_m d_P}$.

Let the algorithm κ used in solving the Deffie-Hellman Problem (DHP) in the RSA modulus Z_N has an advantage, denoted by $Adv^{DHP}_{Z_N,\kappa}$. Let $Prob[Abort]$ denotes the probability of algorithm κ to abort. Then the probability can be gives as:

$$Prob[Abort] \leq \frac{q_{H_1}}{N} \tag{5.12}$$

The adversary ß view will become identical to its view in real attack, if the algorithm κ does not aborts. Then the probability using equation 5.12 can be gives as:

$$\left[Prob[c^*_g = c^*] - Prob[c^*_g \neq c^*] \right] \geq \epsilon - \frac{q_{H_1}}{N} \tag{5.13}$$

The adversary ß queries the Oracle H_2 at the random group element generated in **Stage 4**, $I = g^{r^*_m d_P}$, let this event be denotes as S. Then the probability can be gives as:

$$Prob[S] \geq \left[Prob[c^*_g = c^*] - Prob[c^*_g \neq c^*] \right] \tag{5.14}$$

The probability of the tuple chosen by algorithm κ from the H_2 query list that is same as $g^{r_m d_P}$ is $\frac{1}{q_c + q_{H_2}}$.Thus, using the equations 5.13, 5.14 the final advantage can be computed as:

$$Adv^{DHP}_{Z_N,\kappa} = \frac{1}{q_c + q_{H_2}} Prob[S] \geq \frac{1}{q_c + q_{H_2}} \left(\epsilon - \frac{q_{H_1}}{N} \right)$$

Each secret key takes time $\mathcal{O}(1)$ for computing group inverse operations and each decryption takes time $\mathcal{O}(1)$ for computing exponential operations. Thus, the time taken by the algorithm ß to solve the DHP in RSA modulus Z_N, is:

$$t_* = t + \mathcal{O}(q_c t_c + q_e t_{inv} + q_{H_1} t_{exp})$$

.

According to **Theorem 4**, the algorithm κ can deduce the system secret key pair that is associated with the challenge ciphertext C_{c^*}. Thus, $g^{r_m d_P}$ can only be calculated if and only if the DHP problem in RSA modulus Z_N can be solved. But since we have proved that it is a computationally hard (infeasible) problem, therefore, it contradicts the assumption made that algorithm κ is able to break the indistinguishablity of messages under the chosen ciphertext attack (CCA) and the n-IF-DHP assumption, with the advantage:

$$\epsilon^* = Adv_{Z_N,\kappa}^{DHP} = \frac{1}{q_c + q_{H_2}}\left(\epsilon - \frac{q_{H_1}}{N}\right)$$

Hence proved.

Chapter 6: CASE STUDY FOR SELECTIVE ACCESS FROM A PORTABLE DEVICE

6.1 SELECTIVE ACCESS SMART HOME

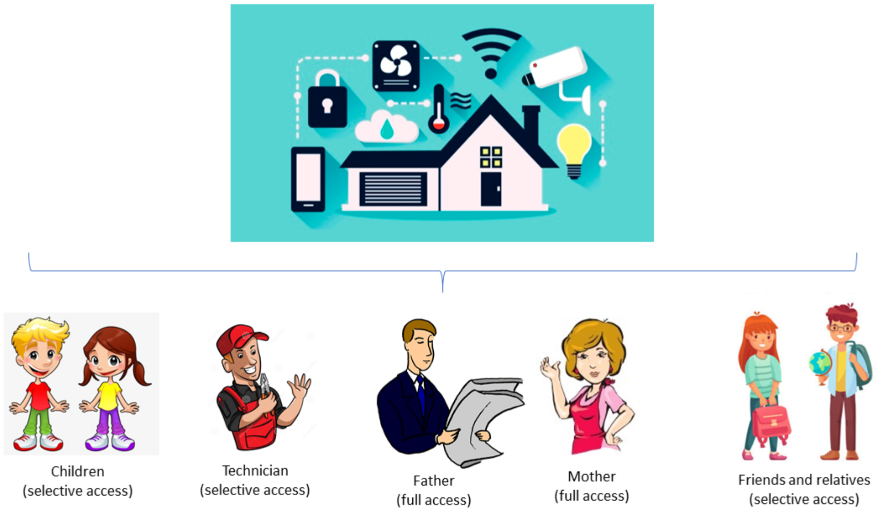


Figure 6.1: Selective access smart home

In this section, we consider a case study of artificial intelligence equipped smart home [1] which provides selective access to multiple users. Here the users refer to the members of a family, which include the father and the mother who are also the owners of the home and have full access of the system, and their son and daughter that have selective access for the same. Some other users to this system can include friends or a technician.

Each user possesses some attributes, which define the access rights of that user. The users are also issued unique keys by the system, which helps in identifying a particular user. The access rights vary according to the users, like the elders of the family are given complete access, whereas the children, friends and the technician are given only partial access to the system. The environment of room or home modifies itself according to the preference of the user accessing it. For example, the door unlocks/locks itself, ac temperature is set according to user, wifi turns ON/OFF, lights turns ON/OFF.

In present time, sensitive data are transmitted and received by the IoT devices [1]. This makes them vulnerable to security breaches. A viable solution for this problem is the attribute-based encryption. The smart home should satisfy the requirements listed below for the effective and efficient functioning of the system.

- **Scalable user revocation:** The system can revoke an infinite number of users. If some user violates any terms of the agreement or leaves the system, his/her access must be revoked. For example in the case of the smart home, the people other than the family members, friend, and technician, must not be given access. Also, owner should be able to revoke access of another user when required.

- **An efficient delegation of user:** A user can delegate his access rights to another user who can than access the system on user's behalf. For example, if some family member wishes to allow some relative or friend to have access to the home, then that member can delegate his access rights.

- **No re-generation and re-distribution of keys:** The non-revoked user should be able to continue their work uninterruptedly, whenever a specific user is revoked. This feature prevents the overhead involved in the re-generation and the re-distribution of the keys associated with the non-revoked users. The access rights of other users should remain the same and need not be modified when some particular user is revoked.

- **No re-encryption of ciphertext:** No re-encryption of ciphertext should be required during revocation of a specific user. The authorize non-revoked user should be able to continue their work uninterruptedly.

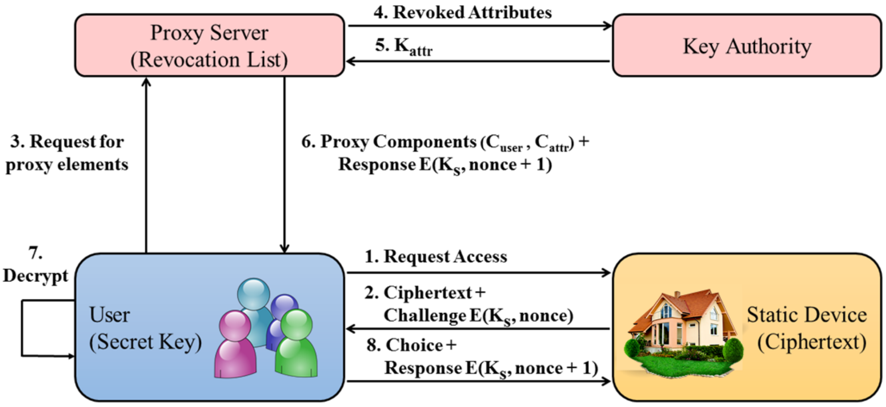

Proxy Server
(Revocation List)
4. Revoked Attributes
5. K$_{attr}$
Key Authority
3. Request for proxy elements
6. Proxy Components (C$_{user}$, C$_{attr}$) + Response E(K$_S$, nonce + 1)
7. Decrypt
User
(Secret Key)
1. Request Access
2. Ciphertext + Challenge E(K$_S$, nonce)
8. Choice + Response E(K$_S$, nonce + 1)
Static Device
(Ciphertext)

Figure 6.2: Replay Attack

- **Immune to replay attack:** An adversary will not be able to access the asset having an old proxy elements. From Fig 6.2, we can see that after requesting for access for home, the static devices replies with the ciphertext along with a challenge having a nonce. Using this nonce and ciphertext, the user than asks for the proxy elements from the proxy server. Proxy server checks with the revocation list for user authenticity. Proxy server than replies user with the proxy elements and response to challenge nonce, if the user is found to be a legitimate non-revoked user. The user than decrypts the ciphertext. Finally, user sends the choice and response to nonce to the system. Smart home features activates depending upon the choice of user.

Through this case study, we understand the need and implementation of our work, how it can improve the functioning of any simple day-to-day event.

Some other examples are Car Aggregation Company [17], NFC Secure Element-based Mutual Authentication and Attestation for IoT access [1], Selective Access Mobile-based Healthfolder [19].

Chapter 7: CONCLUSION AND FUTURE WORK

The current era revolves around society connected via Internet in which mobile devices such as IoT, sensors plays a vital role. Mobile devices being portable has also disadvantage of battery and storage limitation. It is also crucial to look into security and privacy of data in these devices. The security solutions need to take into account the efficiency and light-weight phenomenon of these devices [28], [29].

Ciphertext Policy Based Attribute Encryption (CP-ABE) has become one of the most attractive research topics in recent times. It has various properties and applications in various fields, which makes it the choice of study that one researcher can make. It allows non-interactive access control of encrypted data. In this work, we expound the emergence and development of the CP-ABE schemes.

Revocation is an essential feature in any encryption mechanism to monitor the malicious activity of the user. The proposed algorithm ensures selective access to the user based on its attributes. It is found to effective over Emura et al. [9] extension ProSRCC [18] that is CPA secure. Our work is extension of Odelu et. al [10], that is introducing revocation in it. Also, the ciphertext length remains constant even if the number of attributes is gets increased or decreased. From the comparison study it is evident that our scheme takes less time as compared to existing CP-ABE schemes. It is selectively secure against CCA attacks as well as CPA attacks and is collusion resistant.

The drawback of our scheme is that we integrate XORs with hasing funtions. Since XORs are not found to be that effectively secure. Therefore, for future work we can improve our scheme by replacing XORs with some more secure functions. Future it can be enhanced for user attribute revocation. We used AND-gate access structure, that is single valued attribute. But at present, the attribute can have multiple values. Thus, we can extend out scheme to support multivalued attributes. In our scheme, the owner and users are at single level of authority. Therefore, for future work we can extend out scheme into multi-tier authority access control mechanism in portable devices. Also, we can look into other latest CP-ABE scheme such as Zhang et. al [13], Teng et. al [30], Aijung et. al [31] and intergrate revocation in it, or search for CP-ABE scheme with sclabale revocation and compare it with our scheme.

BIBLIOGRAPHY

[1] D. Sethia, D. Gupta, H. Saran, H. Dabas, and P. Nagar, "NFC Secure Element-Based Mutual Authentication and Attestation for IoT Access," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 4, pp. 470–479, 2018.

[2] M. Zheng, Y. Xiang, and H. Zhou, "A strong provably secure ibe scheme without bilinear map," *Journal of Computer and System Sciences*, vol. 81, no. 1, pp. 125–131, 2015.

[3] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *International Conference on the Theory and Application of Cryptology and Information Security.* Springer, 2007, pp. 200–215.

[4] F. Guo, Y. Mu, and W. Susilo, "Identity-based traitor tracing with short private key and short ciphertext," in *European Symposium on Research in Computer Security.* Springer, 2012, pp. 609–626.

[5] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," in *Springer Proc. Int. Conf. Annual international cryptology*, 2001, pp. 213–229.

[6] R. Canetti, S. Halevi, and J. Katz, "Chosen-Ciphertext Security from Identity-Based Encryption," in *Springer Proc. Int. Conf. Theory and Applications of Cryptographic Techniques*, 2004, pp. 207–222.

[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy Attribute-Based Encryption," in *IEEE Proc. Int. Symp. Security and Privacy*, 2007, pp. 321–334.

[8] L. Cheung and C. Newport, "Provably Secure Ciphertext Policy ABE," in *ACM Proc. Int. Conf. Computer and communications security*, 2007, pp. 456–465.

[9] K. Emura, A. Miyaji, A. Nomura, K. Omote, and M. Soshi, "A Ciphertext-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length," in *Springer Proc. Int. Conf. Information Security Practice and Experience*, 2009, pp. 13–23.

[10] V. Odelu, A. K. Das, M. K. Khan, K.-K. R. Choo, and M. Jo, "Expressive cp-abe scheme for mobile devices in iot satisfying constant-size keys and ciphertexts," *IEEE Access*, vol. 5, pp. 3273–3283, 2017.

[11] L. Pang, J. Yang, and Z. Jiang, "A survey of research progress and development tendency of attribute-based encryption," *The Scientific World Journal*, vol. 2014, 2014.

[12] 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.* ACM, 2010, pp. 753–755.

[13] Y. Zhang, D. Zheng, X. Chen, J. Li, and H. Li, "Computationally efficient ciphertext-policy attribute-based encryption with constant-size ciphertexts," in *International Conference on Provable Security.* Springer, 2014, pp. 259–273.

[14] F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "Cp-abe with constant-size keys for lightweight devices," *IEEE transactions on information forensics and security*, vol. 9, no. 5, pp. 763–771, 2014.

[15] H. Li, X. Lin, H. Yang, X. Liang, R. Lu, and X. Shen, "Eppdr: An efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2053–2064, 2013.

[16] S. Jahid and N. Borisov, "PIRATTE: Proxy-based Immediate Revocation of ATTribute-based Encryption," *arXiv preprint arXiv:1208.4877*, 2012.

[17] D. Sethia, D. Gupta, H. Saran, H. Dabas, and P. Nagar, "Selective IoT Access with Scalable CP-ABE Revocation and Delegation," in *IEEE Proc. Int. Conf. Computational Science and Computational Intelligence*, 2017, pp. 703–708.

[18] D. S. Zeya Umayya, "Pro-SRCC: Proxy-based Scalable Revocation for Constant Ciphertext Length," *Proc. Int. Conf. SECURWARE-The Twelfth International Conference on Emerging Security Information, Systems and Technologies*, pp. 58–66, 2018.

[19] D. Sethia, H. Saran, and D. Gupta, "CP-ABE for Selective Access with Scalable Revocation: A Case Study for Mobile-based Healthfolder," *Int. Journal Network Security*, vol. 20, no. 4, pp. 689–701, 2018.

[20] D. Hofheinz and E. Kiltz, "Practical chosen ciphertext secure encryption from factoring," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques.* Springer, 2009, pp. 313–332.

[21] K. S. McCurley, "A key distribution system equivalent to factoring," *Journal of cryptology*, vol. 1, no. 2, pp. 95–105, 1988.

[22] L. Harn and H.-Y. Lin, "A cryptographic key generation scheme for multilevel data security," *Computers & Security*, vol. 9, no. 6, pp. 539–546, 1990.

[23] S. G. Akl and P. D. Taylor, "Cryptographic solution to a problem of access control in a hierarchy," *ACM Transactions on Computer Systems (TOCS)*, vol. 1, no. 3, pp. 239–248, 1983.

[24] J. Bethencourt, A. Sahai, and B. Waters, "Advanced crypto software collection," http://acsc.cs.utexas.edu/cpabe/, 2006.

[25] "The gnu multiple precision arithmetic library," https://gmplib.org/.

[26] "The pairing-based cryptography library," https://crypto.stanford.edu/pbc/.

[27] J. Li, Q. Wang, C. Wang, and K. Ren, "Enhancing attribute-based encryption with attribute hierarchy," *Mobile networks and applications*, vol. 16, no. 5, pp. 553–561, 2011.

[28] Y. Yang, H. Cai, Z. Wei, H. Lu, and K.-K. R. Choo, "Towards lightweight anonymous entity authentication for iot applications," in *Australasian Conference on Information Security and Privacy.* Springer, 2016, pp. 265–280.

[29] Y. Yang, J. Lu, K.-K. R. Choo, and J. K. Liu, "On lightweight security enforcement in cyber-physical systems," in *Lightweight Cryptography for Security and Privacy.* Springer, 2015, pp. 97–112.

[30] W. Teng, G. Yang, Y. Xiang, T. Zhang, and D. Wang, "Attribute-based access control with constant-size ciphertext in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 5, no. 4, pp. 617–627, 2015.

[31] A. Ge, R. Zhang, C. Chen, C. Ma, and Z. Zhang, "Threshold ciphertext policy attribute-based encryption with constant size ciphertexts," in *Australasian Conference on Information Security and Privacy.* Springer, 2012, pp. 336–349.