

# Proposal of a Revocable Multi-Authority CP-ABE Scheme for Secure Data Access Control in Cloud Storage

Kowde Ramesh<sup>1</sup>; A.Shoba Rani<sup>2</sup>& Prof.Dr.G.Manoj Someswar<sup>3</sup>

<sup>1</sup>M.Tech.(CSE) from Narasimha Reddy Engineering College, Affiliated to JNTUH, Hyderabad, Telangana, India

<sup>2</sup>M.Tech. (CSE), Associate Professor, Department of CSE, Narasimha Reddy Engineering College, Affiliated to JNTUH, Hyderabad, Telangana, India

<sup>3</sup>B.Tech., M.S.(USA), M.C.A., Ph.D., Principal & Professor, Department Of CSE, Anwar-ul-uloom College of Engineering & Technology, Affiliated to JNTUH, Vikarabad, Telangana, India

## ABSTRACT

Data access control is an effective way to ensure the data security in the cloud. Due to data outsourcing and untrusted cloud servers, the data access control becomes a challenging issue in cloud storage systems. Ciphertext-Policy Attributebased Encryption (CP-ABE) is regarded as one of the most suitable technologies for data access control in cloud storage, because it gives data owners more direct control on access policies. However, it is difficult to directly apply existing CP-ABE schemes to data access control for cloud storage systems because of the attribute revocation problem. In this paper, we design an expressive, efficient and revocable data access control scheme for multi-authority cloud storage systems, where there are multiple authorities co-exist and each authority is able to issue attributes independently. Specifically, we propose a revocable multi-authority CP-ABE scheme, and apply it as the underlying techniques to design the data access control scheme. Our attribute revocation method can efficiently achieve both forward security and backward security. The analysis and simulation results show that our proposed data access control scheme is secure in the random oracle model and is more efficient than previous works.

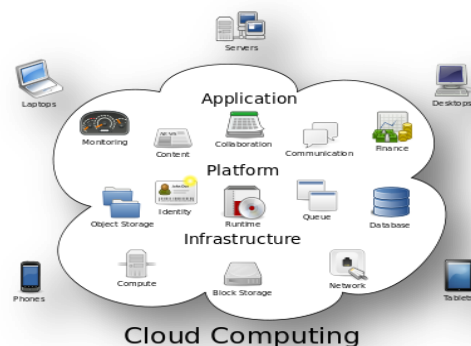
**Keywords:** Ciphertext-Policy Attribute Based Encryption; Broad Network Access; Resource Pooling; Central Authority; Global Identifiers; Acceptance Testing

## INTRODUCTION

**Cloud computing** is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)



**Figure 1: Structure of cloud computing**

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and

research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games.[1]

The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

### Characteristics and Services Models:

The salient characteristics of cloud computing based on the definitions provided by the National Institute of Standards and Terminology (NIST) are outlined below:

- **On-demand self-service:** A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.
- **Broad network access:** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs). [2]
- **Resource pooling:** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location-independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or data center). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)

- **Rapid elasticity:** Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.[3]
- **Measured service:** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be managed, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

### 5 Essential Characteristics of Cloud Computing

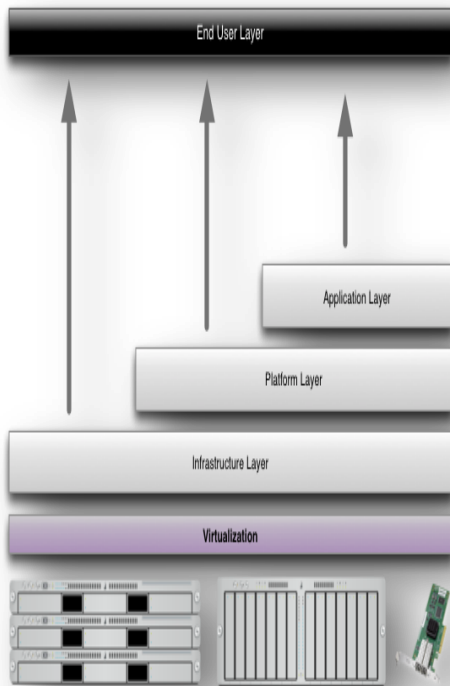


jpinfotech.org

### Figure 2: Characteristics of cloud computing Services Models:

Cloud Computing comprises three different service models, namely Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). The three service models or layer are completed by an end user layer that encapsulates the end user perspective on cloud services. The model is shown in figure below. If a cloud user accesses services on the infrastructure layer, for instance, she can run her own applications on the resources of a cloud infrastructure and remain responsible for the support, maintenance, and security of these applications herself. If she accesses a service on the application layer, these

tasks are normally taken care of by the cloud service provider.[4]



**Figure 3: Structure of service models**

#### Benefits of cloud computing:

1. **Achieve economies of scale** – increase volume output or productivity with fewer people. Your cost per unit, project or product plummets.
2. **Reduce spending on technology infrastructure.** Maintain easy access to your information with minimal upfront spending. Pay as you go (weekly, quarterly or yearly), based on demand.
3. **Globalize your workforce on the cheap.** People worldwide can access the cloud, provided they have an Internet connection.
4. **Streamline processes.** Get more work done in less time with less people.
5. **Reduce capital costs.** There's no need to spend big money on hardware, software or licensing fees.

6. **Improve accessibility.** You have access anytime, anywhere, making your life so much easier!
7. **Monitor projects more effectively.** Stay within budget and ahead of completion cycle times.
8. **Less personnel training is needed.** It takes fewer people to do more work on a cloud, with a minimal learning curve on hardware and software issues.
9. **Minimize licensing new software.** Stretch and grow without the need to buy expensive software licenses or programs.
10. **Improve flexibility.** You can change direction without serious “people” or “financial” issues at stake.

#### Advantages:

1. **Price:** Pay for only the resources used.
2. **Security:** Cloud instances are isolated in the network from other instances for improved security.
3. **Performance:** Instances can be added instantly for improved performance. Clients have access to the total resources of the Cloud's core hardware.
4. **Scalability:** Auto-deploy cloud instances when needed.
5. **Uptime:** Uses multiple servers for maximum redundancies. In case of server failure, instances can be automatically created on another server.
6. **Control:** Able to login from any location. Server snapshot and a software library lets you deploy custom instances.
7. **Traffic:** Deals with spike in traffic with quick deployment of additional instances to handle the load.

#### LITERATURE SURVEY

In case of the research study on “Multi-Authority Attribute Based Encryption by M. Chase, an identity based encryption scheme, wherein each user is

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)



identified by a unique identity string. An attribute based encryption scheme (ABE), in contrast, is a scheme in which each user is identified by a set of attributes, and some function of those attributes is used to determine decryption ability for each ciphertext. Sahai and Waters introduced a single authority attribute encryption scheme and left open the question of whether a scheme could be constructed in which multiple authorities were allowed to distribute attributes [5]. We answer this question in the affirmative. Our scheme allows any polynomial number of independent authorities to monitor attributes and distribute secret keys. An encryptor can choose, for each authority, a number  $dk$  and a set of attributes; he can then encrypt a message such that a user can only decrypt if he has at least  $dk$  of the given attributes from each authority  $k$ . Our scheme can tolerate an arbitrary number of corrupt authorities. We also show how to apply our techniques to achieve a multiauthority version of the large universe fine grained access control ABE presented by Gopal et al.

In the case of “Improving Privacy and Security in Multi-Authority Attribute-Based Encryption,” by M. Chase and S.S.M. Chow, Attribute based encryption (ABE) [6] determines decryption ability based on a user's attributes. In a multi-authority ABE scheme, multiple attribute-authorities monitor different sets of attributes and issue corresponding decryption keys to users and encryptors can require that a user obtain keys for appropriate attributes from each authority before decrypting a message. Chase [5] gave a multi-authority ABE scheme using the concepts of a trusted central authority (CA) and global identifiers (GID). However, the CA in that construction has the power to decrypt every ciphertext, which seems somehow contradictory to the original goal of distributing control over many potentially untrusted authorities. Moreover, in that construction, the use of a consistent GID allowed the authorities to combine their information to build a full profile with all of a user's attributes, which unnecessarily compromises the privacy of the user. In this paper, we propose a solution which removes the trusted central authority, and protects the users' privacy

by preventing the authorities from pooling their information on particular users, thus making ABE more usable in practice.

Regarding “Decentralizing Attribute-Based Encryption,” by A.B. Lewko and B. Waters, we propose a Multi-Authority Attribute-Based Encryption (ABE) system. In our system, any party can become an authority and there is no requirement for any global coordination other than the creation of an initial set of common reference parameters. A party can simply act as an ABE authority by creating a public key and issuing private keys to different users that reflect their attributes. A user can encrypt data in terms of any Boolean formula over attributes issued from any chosen set of authorities. Finally, our system does not require any central authority.[7]

In constructing our system, our largest technical hurdle is to make it collusion resistant. Prior Attribute-Based Encryption systems achieved collusion resistance when the ABE system authority “tied” together different components (representing different attributes) of a user's private key by randomizing the key. However, in our system each component will come from a potentially different authority, where we assume no coordination between such authorities. We create new techniques to tie key components together and prevent collusion attacks between users with different global identifiers.[8] We prove our system secure using the recent dual system encryption methodology where the security proof works by first converting the challenge ciphertext and private keys to a semi-functional form and then arguing security. We follow a recent variant of the dual system proof technique due to Lewko and Waters and build our system using bilinear groups of Composite order. We prove security under similar static assumptions to the LW paper in the random oracle model.

In the case of “Attribute Based Data Sharing with Attribute Revocation,” by S. Yu, C. Wang, K. Ren, and W. Lou, Ciphertext-Policy Attribute Based Encryption (CP-ABE) is a promising cryptographic primitive for fine-grained access control of shared data.

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)



In CP-ABE, each user is associated with a set of attributes and data are encrypted with access structures on attributes. A user is able to decrypt a ciphertext if and only if his attributes satisfy the ciphertext access structure. Beside this basic property, practical applications usually have other requirements. In this paper we focus on an important issue of attribute revocation which is cumbersome for CP-ABE schemes. In particular, we resolve this challenging issue by considering more practical scenarios in which semi-trusted on-line proxy servers are available. As compared to existing schemes, our proposed solution enables the authority to revoke user attributes with minimal effort. We achieve this by uniquely integrating the technique of proxy re-encryption with CP-ABE, and enable the authority to delegate most of laborious tasks to proxy servers.[9] Formal analysis shows that our proposed scheme is provably secure against chosen ciphertext attacks. In addition, we show that our technique can also be applicable to the Key-Policy Attribute Based Encryption (KP-ABE) counterpart.

“Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption,” by M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, Personal health record (PHR) is an emerging patient-centric model of health information exchange, which is often outsourced to be stored at a third party, such as cloud providers. However, there have been wide privacy concerns as personal health information could be exposed to those third party servers and to unauthorized parties. To assure the patients’ control over access to their own PHRs, it is a promising method to encrypt the PHRs before outsourcing. Yet, issues such as risks of privacy exposure, scalability in key management, flexible access and efficient user revocation, have remained the most important challenges toward achieving fine-grained, cryptographically enforced data access control. In this paper, we propose a novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control

**Conference Chair:** Prof. Dr. G. Manoj Someswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)

for PHRs, we leverage attribute based encryption (ABE) techniques to encrypt each patient’s PHR file. Different from previous works in secure data outsourcing, we focus on the multiple data owner scenario, and divide the users in the PHR system into multiple security domains that greatly reduces the key management complexity for owners and users.[10] A high degree of patient privacy is guaranteed simultaneously by exploiting multi-authority ABE. Our scheme also enables dynamic modification of access policies or file attributes, supports efficient on-demand user/attribute revocation and break-glass access under emergency scenarios. Extensive analytical and experimental results are presented which show the security, scalability and efficiency of our proposed scheme.

### **SYSTEM ANALYSIS**

#### **EXISTING SYSTEM:**

This new paradigm of data hosting and data access services introduces a great challenge to data access control. Because the cloud server cannot be fully trusted by data owners, they can no longer rely on servers to do access control.[11] Ciphertext-Policy Attribute-based Encryption (CP-ABE) is regarded as one of the most suitable technologies for data access control in cloud storage systems, because it gives the data owner more direct control on access policies. In CP-ABE scheme, there is an authority that is responsible for attribute management and key distribution.

#### **DISADVANTAGES OF EXISTING SYSTEM:**

- Chase’s multi-authority CP-ABE protocol allows the central authority to decrypt all the ciphertexts, since it holds the master key of the system.
- Chase’s protocol does not support attribute revocation.

#### **PROPOSED SYSTEM:**

In this research paper, we first propose a revocable multi-authority CP-ABE scheme, where an efficient and secure revocation method is proposed to solve the attribute revocation problem in the system.

Our attribute revocation method is efficient in the sense that it incurs less communication cost and computation cost, and is secure in the sense that it can achieve both backward security (The revoked user cannot decrypt any new ciphertext that requires the revoked attribute to decrypt) and forward security (The newly joined user can also decrypt the previously published ciphertexts, if it has sufficient attributes). Our scheme does not require the server to be fully trusted, because the key update is enforced by each attribute authority not the server. Even if the server is not semi-trusted in some scenarios, our scheme can still guarantee the backward security. Then, we apply our proposed revocable multi-authority CP-ABE scheme as the underlying techniques to construct the expressive and secure data access control scheme for multi-authority cloud storage systems.[12]

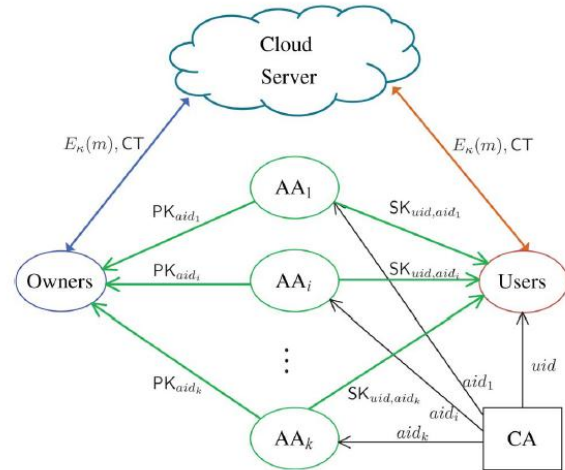
#### ADVANTAGES OF PROPOSED SYSTEM:

- We modify the framework of the scheme and make it more practical to cloud storage systems, in which data owners are not involved in the key generation.
- We greatly improve the efficiency of the attribute revocation method.
- We also highly improve the expressiveness of our access control scheme, where we remove the limitation that each attribute can only appear at most once in a ciphertext.

#### SYSTEM DESIGN

##### SYSTEM ARCHITECTURE:

**Figure 1: System Architecture**



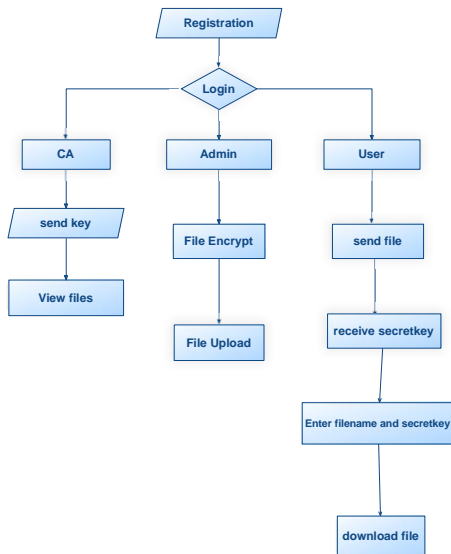
#### DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)

Figure 2: Data Flow Diagram

**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extensibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

**UML DIAGRAMS**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.[13]

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)

Figure 3: Use Case Diagram

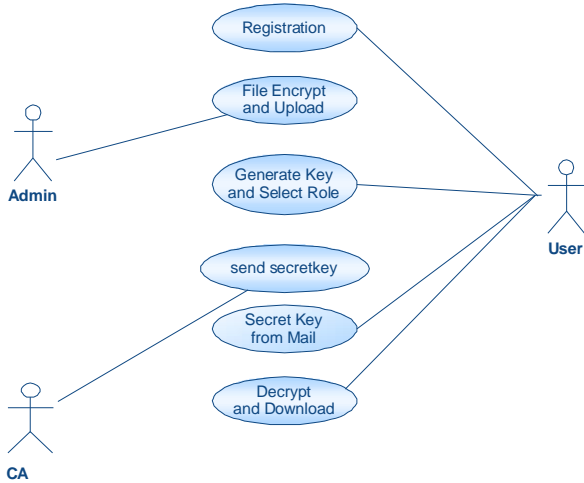
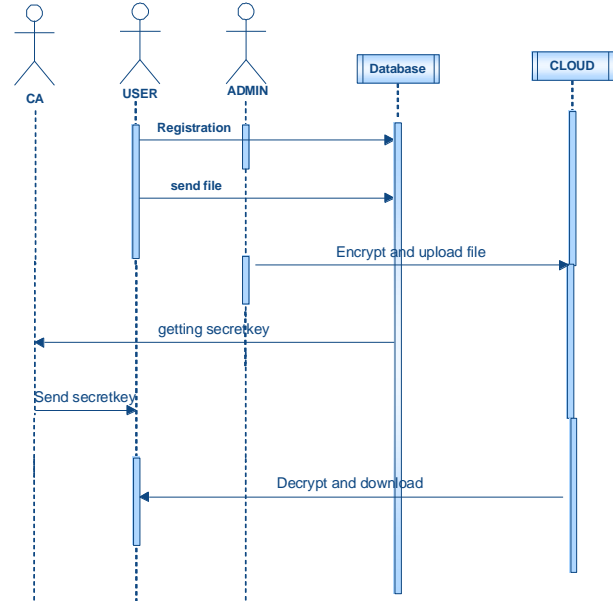


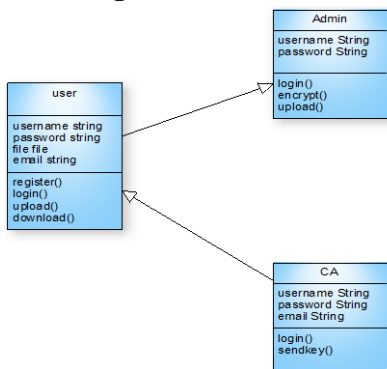
Figure 5: Sequence Diagram



**CLASS DIAGRAM:**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

Figure 4: Class Diagram



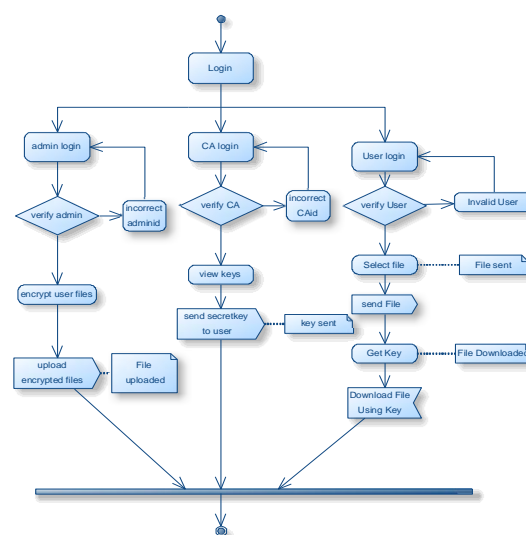
**SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

**ACTIVITY DIAGRAM:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

Figure 6: Activity Diagram



**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)





## **SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

### **TYPES OF TESTS**

#### **Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

#### **Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields.[14] Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

#### **Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.[15]

#### **System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

#### **White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose.[16] It is purpose. It is used to test areas that cannot be reached from a black box level.

#### **Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)



definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.[16]

#### Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

#### Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

#### Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

#### Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

#### Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:**All the test cases mentioned above passed successfully. No defects encountered.

#### Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the

end user. It also ensures that the system meets the functional requirements.

**Test Results:**All the test cases mentioned above passed successfully. No defects encountered.

#### RESULTS & CONCLUSION

In this research paper, we proposed a revocable multi-authority CPABE scheme that can support efficient attribute revocation. Then, we constructed an effective data access control scheme for multi-authority cloud storage systems. We also proved that our scheme was provable secure in the random oracle model. The revocable multi-authority CPABE is a promising technique, which can be applied in any remote storage systems and online social networks etc.

#### REFERENCES

- [1] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” National Institute of Standards and Technology, Gaithersburg, MD, USA, Tech. Rep., 2009.
- [2] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-Policy Attribute-Based Encryption,” in Proc. IEEE Symp. Security and privacy (S&P’07), 2007, pp. 321-334.
- [3] B. Waters, “Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization,” in Proc. 4th Int’l Conf. Practice and Theory in Public Key Cryptography (PKC’11), 2011, pp. 53-70.
- [4] V. Goyal, A. Jain, O. Pandey, and A. Sahai, “Bounded Ciphertext Policy Attribute Based Encryption,” in Proc. 35th Int’l Colloquium on Automata, Languages, and Programming (ICALP’08), 2008, pp. 579-591.
- [5] A.B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, “Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption,” in Proc. Advances in Cryptology-EUROCRYPT’10, 2010, pp. 62-91.

**Conference Chair:** Prof. Dr. G. Manoj Someswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)



[6] M. Chase, “Multi-Authority Attribute Based Encryption,” in Proc. 4th Theory of Cryptography Conf. Theory of Cryptography (TCC’07), 2007, pp. 515-534.

[7] M. Chase and S.S.M. Chow, “Improving Privacy and Security in Multi-Authority Attribute-Based Encryption,” in Proc. 16<sup>th</sup> ACM Conf. Computer and Comm. Security (CCS’09), 2009, pp. 121-130.

[8] A.B. Lewko and B. Waters, “Decentralizing Attribute-Based Encryption,” in Proc. Advances in Cryptology-EUROCRYPT’11, 2011, pp. 568-588.

[9] S. Yu, C. Wang, K. Ren, and W. Lou, “Attribute Based Data Sharing with Attribute Revocation,” in Proc. 5th ACM Symp. Information, Computer and Comm. Security (ASIACCS’10), 2010, pp. 261-270.

[10] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, “Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption,” IEEE Trans. Parallel Distributed Systems, vol. 24, no. 1, pp. 131-143, Jan. 2013.

[11] J. Hur and D.K. Noh, “Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems,” IEEE Trans. Parallel Distributed Systems, vol. 22, no. 7, pp. 1214-1221, July 2011.

[12] S. Jahid, P. Mittal, and N. Borisov, “Easier: Encryption-Based Access Control in Social Networks with Efficient Revocation,” in Proc. 6th ACM Symp. Information, Computer and Comm. Security (ASIACCS’11), 2011, pp. 411-415.

[13] S. Ruj, A. Nayak, and I. Stojmenovic, “DACC: Distributed Access Control in Clouds,” in Proc. 10th IEEE Int’l Conf. TrustCom, 2011, pp. 91-98.

[14] K. Yang and X. Jia, “Attribute-Based Access Control for Multi-Authority Systems in Cloud

Storage,” in Proc. 32th IEEE Int’l Conf. Distributed Computing Systems (ICDCS’12), 2012, pp. 1-10.

[15] D. Boneh and M.K. Franklin, “Identity-Based Encryption from the Weil Pairing,” in Proc. 21st Ann. Int’l Cryptology Conf.: Advances in Cryptology - CRYPTO’01, 2001, pp. 213-229.

[16] A.B. Lewko and B. Waters, “New Proof Methods for Attribute- Based Encryption: Achieving Full Security through Selective Techniques,” in Proc. 32st Ann. Int’l Cryptology Conf.: Advances in Cryptology - CRYPTO’12, 2012, pp. 180-198.

**Conference Chair:** Prof.Dr.G.ManojSomeswar, Director General, Global Research Academy, Hyderabad, Telangana, India.

Papers presented in Conference can be accessed from [www.edupediapublications.org/journals](http://www.edupediapublications.org/journals)