

A
Major Project Report
On
**Combining Data-Owner Side and Cloud-Side Access Control for
Encrypted Cloud Storage**

Submitted to

Jawaharlal Nehru Technological University, Hyderabad

In partial fulfillment of the requirements for the award of Degree of

Bachelor of Technology

In

Computer Science & Engineering

By

SABA BEGUM
SUMAIYA NAZNEEN
P.RAVEENA
SADIA AFREEN

(166Y1A0586)
(166Y1A0596)
(166Y1A0583)
(166Y1A0588)

Under the guidance
Of
T.RAVI KUMAR
Asst. Professor



Department of Computer Science & Engineering

SUMATHI REDDY INSTITUTE OF TECHNOLOGY for WOMEN

(Approved by AICTE, New Delhi; Affiliated to JNTU, Hyderabad)

Ananthasagar(Vill), Hasanparthy(M), Warangal – 506 371(T.S.), Website : www.sritw.org

2019-2020

SUMATHI REDDY INSTITUTE OF TECHNOLOGY for WOMEN

(Approved by AICTE, New Delhi; Affiliated to JNTU, Hyderabad)

Ananthasagar(Vill), Hasanparthy(M), Warangal – 506 371 (T.S.), Website : www.sritw.org

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

This is to certify that the project entitled “*Combining Data Owner Side and Cloud Side Access Control for Encrypted Cloud Storage*” is submitted by *Saba begum (166Y1A0586)*, *Sumaiya nazneen (166Y1A0596)*, *P.Raveena (166Y1A0583)* and *Sadia afreen (166Y1A0588)* in the partial fulfillment of requirement for the award of Degree of Bachelor of Technology in Computer Science and Engineering during academic year 2019-20.

MR T.RAVI KUMAR
Project Guide

Dr.E.SUDARSHAN
Head of the Department

External Examiner

Acknowledgement

We wish to take this opportunity to express our deep gratitude to all the people who have extended their cooperation in various ways during my project work. It is our pleasure to acknowledge the help of all those individuals.

We would like to thank our project guide **MR T.RAVIKUMAR**, Asst. Prof Department of Computer Science and Engineering for her guidance and help throughout the development of this project work by providing us with required information. With her guidance, cooperation and encouragement we had learnt many new things during our project tenure.

We would like to thank our project coordinator **Mr.V.SRINIVAS**, for his continuous coordination throughout the project tenure.

We specially thank **Dr.E.SUDARSHAN**, Head, department of Computer Science and Engineering for his continuous encouragement and valuable guidance in bringing shape to this dissertation.

We specially thank **Dr. I.RAJASRI REDDY**, Principal, and Sumathi Reddy Institute of Technology for Women for her encouragement and support.

In completing this project successfully all our faculty members have given an excellent cooperation by guiding us in every aspect. We also thank our lab faculty and librarians.

SABA BEGUM (166Y1A0586)

SUMAIYA NAZNEEN (166Y1A0596)

P.RAVEENA (166Y1A0583)

SADIA AFREEN (166Y1A0588)

ABSTRACT

People endorse the great power of cloud computing, but cannot fully trust the cloud providers to host privacy-sensitive data, due to the absence of user-to-cloud controllability. To ensure confidentiality, data owners outsource encrypted data instead of plaintexts. To share the encrypted files with other users, Cipher text-Policy Attribute-based Encryption (CP-ABE) can be utilized to conduct fine-grained and owner-centric access control. But this does not sufficiently become secure against other attacks. Many previous schemes did not grant the cloud provides the capability to verify whether a downloader can decrypt. Therefore, these files should be available to everyone accessible to the cloud storage. A malicious attacker can download thousands of files to launch Economic Denial of Sustainability (EDOS) attacks, which will largely consume the cloud resource. The payer of the cloud service bears the expense. Besides, the cloud provider serves both as the accountant and the payee of resource consumption fee, lacking the transparency to data owners. These concerns should be resolved in real-world public cloud storage. In this paper, we propose a solution to secure encrypted cloud storages from EDOS attacks and provide resource consumption accountability. It uses CP-ABE schemes in a black-box manner and complies with arbitrary access policy of CP-ABE. We present two protocols for different settings, followed by performance and security analysis.

TABLE OF CONTENTS

TITLES	Page No
1. INTRODUCTION	1-5
1.1 Introduction to Cloud	1-4
1.2 Goals and Objectives	5
2. LITERATURE SURVEY	06
3. ANALYSIS	08-09
3.1 Existing System	08
3.2 Proposed System	08
3.3 System Requirement Specification	09
4. DESIGNING	10-18
4.1 System Architecture	10
4.2 Class diagram	11
4.3 State diagram	12
4.4 Data flow diagram	13
4.5 Activity diagram	14-16
4.6 Sequence diagram	17
4.7 Use case diagram	18
5. IMPLEMENTATION	19
5.1 Modules	19
5.2 Module Description	19
6. SYSTEM TESTING	20-21
6.1 Unit Testing	20
6.2 Integration Testing	20
6.3 Acceptance testing	21
7. OUTPUT SCREENS	22-31
8. CONCLUSION	32
9. BIBLIOGRAPHY	33

TABLE OF FIGURES

Fig4.1: system architecture

Fig 4.2: class diagram

Fig 4.3: state diagram

Fig 4.4: data flow diagram

Fig 4.5(a) data owner diagram

Fig 4.5: (b) user

Fig 4.5: (6) cloud provider

Fig 4.6: sequence diagram

Fig 4.7: use case diagram

1. INTRODUCTION

What is cloud?

In the simplest terms, cloud means storing and accessing data and programs over the Internet instead of your computer's hard drive. The cloud is just a metaphor for the Internet.

1.1 INTRODUCTION TO CLOUD:

Cloud storage has many benefits, such as always-online, pay-as-you-go, and cheap. During these years, more data are outsourced to public cloud for persistent storage, including personal and business documents. It brings a security concern to data owners the public cloud is not trusted, and the outsourced data should not be leaked to the cloud provider without the permission from data owners. Many storage systems use server-dominated access control, like password-based and certificate-based authentication. They overly trust the cloud provider to protect their sensitive data. The cloud providers and their employees can read any document regardless of data owners' access policy. Besides, the cloud provider can exaggerate the resource consumption of the file storage and charge the payers more without providing verifiable records, since we lack a system for verifiable computation of the resource usage.

Relying on the existing server-dominated access control is not secure. Data owners who store files on cloud servers still want to control the access on their own hands and keep the data confidential against the cloud provider and malicious users. Encryption is not sufficient. To add the confidentiality guarantee, data owners can encrypt the files and set an access policy so that only qualified users can decrypt the document. With Cipher text-Policy Attribute-based Encryption (CP-ABE), we can have both fine-grained access control and strong confidentiality.

.To include the confidentiality guarantee, information proprietors can scramble the records and set an access policy with the goal that just qualified clients can decode the document. With Cipher text-Policy Attribute-based Encryption (CP-ABE), we can have both fine-grained get to control and strong secrecy. In any case, this entrance control is accessible for information proprietors, which swings out to be insufficient. On the off chance that the cloud supplier can't verify users before downloading, as in many existing CP-ABE cloud storage frameworks, the cloud needs to permit everyone to download to guarantee accessibility. This makes the storage system defenseless against the asset depletion assaults

However, this access control is only available for data owners, which turn out to be insufficient. If the cloud provider cannot authenticate users before downloading, like in many existing CP-ABE cloud storage systems, the cloud has to allow everyone to download to ensure availability. This makes the storage system vulnerable to the resource-exhaustion attacks. If we resolve this problem

by having data owners authenticate the downloader's before allowing them to download, we lose the flexibility of access control from CP-ABE.

Here lists the two problems should be addressed in our work:

Problem I: Resource-exhaustion attack.

If the cloud cannot do cloud-side access control, it has to allow anyone, including malicious attackers, to freely download, although only some users can decrypt. The server is vulnerable to resource-exhaustion attacks. When malicious users launch the DOS/DDOS attacks to the cloud storage, the resource consumption will increase. Payers (in pay-as-you-go model) have to pay for the increased consumption contributed by those attacks, which is a considerable and unreasonable financial burden. The attack has been introduced as Economic Denial of Sustainability (EDOS), which means payers are financially attacked eventually. In addition, even files are encrypted, unauthorized downloads can reduce security by bringing convenience to offline analysis and leaking information like file length or update frequency.

Problem II: Resource consumption accountability.

In the pay-as-you-go model, users pay money to the cloud provider for storage services. The fee is decided by resource usage. However, CP-ABE based schemes for cloud storage access control does not make online confirmations to the data owner before download. It is needed for the cloud service provider to prove to the payers about the actual resource usage. Otherwise, the cloud provider can charge more without being discovered

A Summary of Challenges and Approaches:

- **Challenge I: modeling the cloud provider:** Many existing CP-ABE based schemes model the cloud providers (like Google, Amazon, Microsoft Azure) as semi honest adversaries or passive attackers. However, such a definition is restricted and it excludes some possible attacks in the real world, such as exaggerated resource usage. To model such attacks, we consider a less restricted security model, covert adversary, for the cloud provider. In practice, the cloud services are usually provided by some big IT enterprises like Google, Amazon, and Microsoft. This model, covert security, has been used in many secure systems. Note that the covert security model is different with the semi-honest model. The semi-honest model, which is widely used in proxies and cloud providers, is a model that resides between “malicious” and “trusted”. It models a party that observes all data, but it never executes the wrong program.

Approach: model cloud providers as covert adversaries, and design protocols resilient to a covert adversary.

- **Challenge II: compatible with existing systems:** There are many constructions and variants for CP-ABE. We don't design a new variant of CP-ABE to resolve the first challenge, as it is hard to achieve all the functionalities in these systems and also it's not necessary. Besides the functionalities, some variants provide additional security and privacy guarantee. For example, the literature hides the access policy. If the cloud-side access control makes the cloud provider knowing the access policy, it is not considered secure and compatible. It requires the cloud side access control to be zero-knowledge for arbitrary CP-ABE schemes. **Approach:** use CP-ABE in a syntactical and black-box way and ensure the construction not leaking policy and attributes. The system only learns whether the user is legitimate or not, and nothing else.

Approach: use CP-ABE in a syntactical and black-box way and ensure the construction not leaking policy and attributes. The system only learns whether the user is legitimate or not, and nothing else.

- **Challenge III: minimal performance overhead:** To protect the cloud storage effectively against the resource-exhaustion attack, the cloud-side access control needs to be efficient and lightweight, otherwise if the cloud server spends, for example 20ms, executing the cloud-side access control, it will become a computational resource exhaustion attacks, which can be used by malicious attackers for DDoS and EDoS.

Approach: design an efficient access control for the cloud provider which should not add too much overhead.

- 1) We propose a general solution to secure encrypted cloud storage to prevent the EDoS attacks
- 2) For different data owner online patterns and performance concern, we provide two protocols for authentication and resource consumption accounting. We also introduce the bloom filter and the probabilistic check to improve the efficiency but still guarantee the security

Preliminary:

We provide the preliminary information on the cryptographic tools, the underlying CP-ABE and the encryption with integrity guarantee AEAD. We present the key encapsulated mechanism to avoid double cost and ensure the integrity between the data owner and data users and we introduce the probabilistic check tool, bloom filter.

A. CP-ABE: Cipher text-Policy Attribute-based-Encryption

CP-ABE is a public key encryption scheme with fine-grained access control. In CP-ABE, each user has some attributes and data owners encrypt their files with an access policy over attributes. Users in the system hold their own secret keys associated with their attribute sets. If and only if the user satisfies the access policy, the user can decrypt. Some useful definitions in CP-ABE are as follows:

Attributes: Attributes depict the party's properties relevant to access control.

Policy: A policy is a predicate over the attributes

Correctness, security and the construction: The definitions and formal proofs of correctness and security, and the construction of CP-ABE can be found. CP-ABE achieves indistinguishability under chosen-plaintext attacks.

B. Authenticated Encryption with Associated Data:

Authenticated Encryption with Associated Data (AEAD) is a symmetric-key encryption that provides both confidentiality and integrity

C. Digital Signature:

The system uses a public-key signature scheme for message integrity. Assuming the secure distribution of public keys, any data recipient can verify the message integrity. For succinctness of signatures

D. Hybrid Encryption for CP-ABE:

We illustrate the usage of hybrid encryption with the example of two CP-ABE cipher texts with the same access policy and from the same data owner. The cost can be reduced by encrypting an ephemeral key for both cipher texts.

E. Bloom Filter:

Bloom filter is an m -bit sequence for membership test that is reasonably accurate and space-efficient

1.2 GOALS AND OBJECTIVES:

Goals: Malicious user is to convince the cloud server that he is a legitimate data owner. □

1. To design a dynamic collaboration environment utilizing the benefits of cloud storage while ensuring strong data security and fine-grained data access.
2. To improve our skills and knowledge with regards to designing systems that leverage the benefits of the cloud, improve our ability to research a scientific subject from different perspectives, and to contribute to the scientific community.

Objectives: Improving data confidentiality in cloud storage environments while enhancing dynamic sharing between users.

1. Indeed, the proposed security mechanisms should ensure both robustness and efficiency, namely the support of flexible access control, efficient user revocation and performances.
2. Addressing the issue of provable data possession in cloud storage environments for data integrity verification support, following three substantial aspects: security level, public verifiability, and performance, and considering the limited storage and processing capacities of user devices.

2. LITERATURE SURVEY

1: Efficient k-NN Query over Encrypted Data in Cloud with Limited Key-disclosure and Offline Data Owner

Lu Zhouc, Youwen Zhua, Aniello Castiglione

In this paper, we propose a new scheme to perform k-NN query over encrypted data in cloud while protecting the privacy of both data owner and query users from cloud. Our new method just reveals limited information about data owner's key to query users, and has no need of an online data owner. For gaining the properties, we present a new scalar product protocol, then the new protocol and some other transformation approaches are merged into our secure k-NN query system. Additionally, we confirm our security and efficiency through theoretical analysis and extensive simulation experiments.

2: OPass: A User Authentication Protocol Resistant to Password Stealing and Password Reuse Attacks

Hung-Min Sun, Yao-Hsin Chen, and Yue-Hsun Lin

Text password is the most popular form of user authentication on websites due to its convenience and simplicity. However, users' passwords are prone to be stolen and compromised under different threats and vulnerabilities. Firstly, users often select weak passwords and reuse the same passwords across different websites. Routinely reusing passwords causes a domino effect; when an adversary compromises one password, she will exploit it to gain access to more websites. Second, typing passwords into untrusted computers suffers password thief threat. An adversary can launch several password stealing attacks to snatch passwords, such as phishing, key loggers and malware. In this paper, we design a user authentication protocol named oPass which leverages a user's cellphone and short message service to thwart password stealing and password reuse attacks. OPass only requires each participating website possesses a unique phone number, and involves a telecommunication service provider in registration and recovery phases. Through oPass, users only need to remember a long-term password for login on all websites. After evaluating the oPass prototype, we believe oPass is efficient and affordable compared with the conventional web authentication mechanisms

3. Securing SIFT: Privacy-preserving Outsourcing Computation of Feature Extractions over Encrypted Image Data

Shengshan Hu, Qian Wang, Jingjun Wang, Zhan Qin, and Kui Ren,

This observation has recently aroused new research interest on privacy-preserving computations over outsourced multimedia data. In this paper, we propose an effective and practical privacy-preserving computation outsourcing protocol for the prevailing scale-invariant feature transform (SIFT) over massive encrypted image data. We first show that previous solutions to this problem have either efficiency/security or practicality issues, or none can well preserve the important characteristics of the original SIFT in terms of distinctiveness and robustness. We then present a new scheme design that achieves efficiency and security requirements simultaneously with the preservation of its key characteristics, by randomly splitting the original image data, designing two novel efficient protocols for secure multiplication and comparison, and carefully distributing the feature extraction computations onto two independent cloud servers. We both carefully analyze and extensively evaluate the security and effectiveness of our design. The results show that our solution is practically secure, outperforms the state-of-the-art, and performs comparably to the original SIFT in terms of various characteristics, including rotation invariance, image scale invariance, robust matching across affine distortion, addition of noise and change in 3D viewpoint and illumination

3. ANALYSIS

3.1 EXISTING SYSTEM:

In the existing system, R. K. Koet.al., the authors discussed key issues and challenges about how to achieve accountability in cloud computing. In the literature, D. O'Coileá'et.al., the authors surveyed existing accounting and accountability in content distribution architectures. ⊞ V. Sekaret. Al. and C. Chen et. al., the authors respectively proposed a systematic approach for verifiable resource accounting in cloud computing.

DISADVANTAGES OF EXISTING SYSTEM:

- The accounting approach involves changes to the system model, and requires the anonymous verification of users, which is not supported in previous systems.
- The access control is only available for data owners, which turns out to be insufficient.
- This makes the storage system vulnerable to the resource exhaustion attacks. ⊞ It loses the flexibility of access control from CP-ABE.

3.2 PROPOSED SYSTEM:

In this paper, we combine the cloud side access control and the existing data owner-side CP-ABE based access control, to resolve the aforementioned security problems in privacy preserving cloud storage. Our method can prevent the EDOS attacks by providing the cloud server with the ability to check whether the user is authorized in CP-ABE based scheme, without leaking other information. ⊞ For our cloud-side access control, we use CP-ABE encryption/ decryption game as challenge-response. While upload an encrypted file, the data owner firstly generates some random challenge plaintexts and the corresponding cipher texts. The cipher texts are related to the same access policy with the specific file. For an incoming data user, the cloud server asks him/her to decrypt randomly selected challenge cipher text. ⊞ If the user shows a correct result, which means he/she is authorized in CP-ABE, the cloud-side access control allows the file download. To make our solution secure and efficient in real world applications, we provide two protocols of cloud side and data owner-side combined access control.

ADVANTAGES OF PROPOSED SYSTEM:

- We propose a general solution to secure encrypted cloud storage to prevent the EDOS attacks, as well as have fine-grained access control and resource consumption accountability. To the best of our knowledge, this is the first work to claim that insufficient cloud-side access control in encrypted cloud storage will lead to EDOS attacks and provides a practical solution. The solution can be compatible with many CP-ABE schemes.

- For different data owner online patterns and performance concern, we provide two protocols for authentication and resource consumption accounting. We also introduce the bloom filter and the probabilistic check to improve the efficiency but still guarantee the security.
- Compared with many state-of-arts constructions of encrypted cloud storage that assume the existence of a semi-honest cloud provider, we use a more practical threat model where we assume the cloud provider to be a covert adversary, which provides higher security guarantee. Compared with relevant schemes, our approach works on the protocol level to provide the resource verifiability that relies on authorized users who satisfy the CP-ABE

As appeared in Fig. 1, we have three controls among three entities in our framework:

Control I. Data proprietors dole out an entrance arrangement in the document, which controls the arrangement of information clients who have the benefits to decode the substance.

Control II. Data proprietors confirms the asset consumption from the cloud supplier, which controls the cloud provider not to overstate the asset use.

Control III. The cloud supplier confirms whether the user can unscramble before the download, which controls the capacity of a malevolent client who dispatches DDoS /EDoS attacks.

3.3 SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System: Pentium Dual Core.
- Hard Disk: 120 GB.
- Monitor : 15’’ LED
- Input Devices : Keyboard, Mouse
- Ram : 1 GB

SOFTWARE REQUIREMENTS:

- Operating system: Windows 7.
- Coding Language : JAVA/J2EE
- Database : MYSQL
- Web Server : Tomcat
- IDE : Eclipse

4. DESIGNING

4.1 System Architecture:

Besides, our framework varies from past cloud storage constructions, as we consider the asset consumption. Practically speaking, the cloud administrations are generally charged according to the asset utilization, which incorporates the resource spent on aggressors. The DDoS/EDoS assaults will invariably succeed and raise the overhead, which is controlled in our framework because of the presentation of the cloud-side access control

System model of the encrypted cloud storage with mitigation of EDoSattacks and transparency of resource consumption accounting

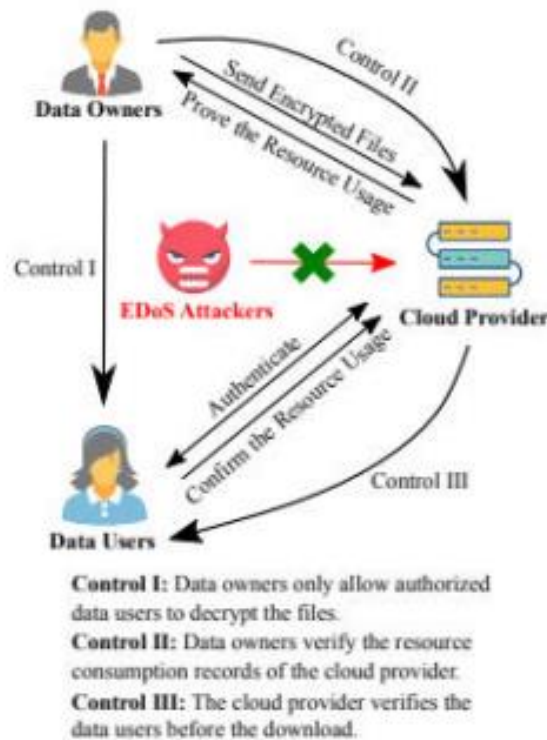
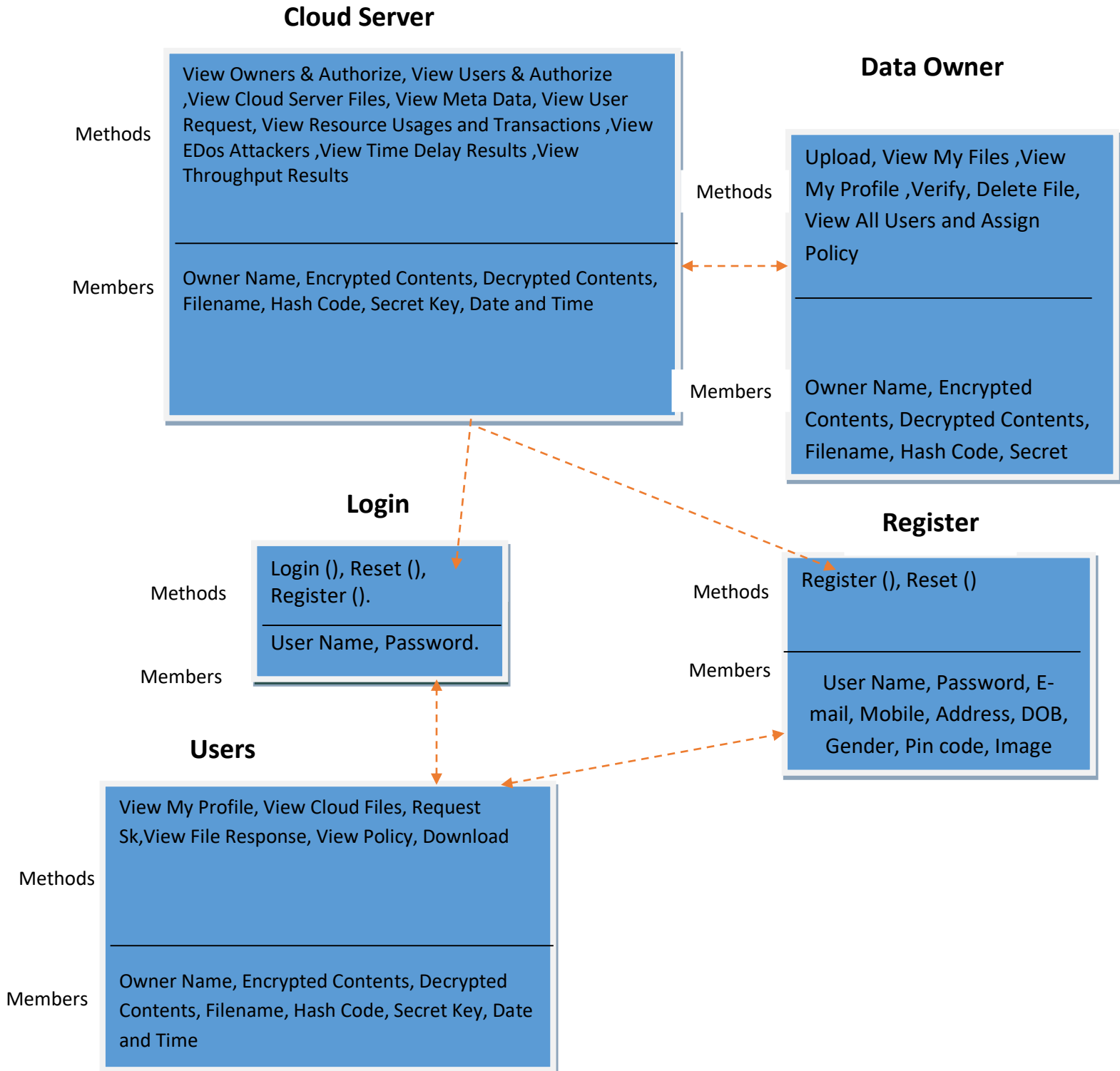


Fig: 4.1 system architecture

4.2 Class Diagram: class diagram is a type of static structure diagram that describes the structure of a system by showing system's classes, their attributes, operations and the relationship among objects.



4.3 State Diagram:

A state diagram is used to represent the condition of the system and also to describe the behavior of system.

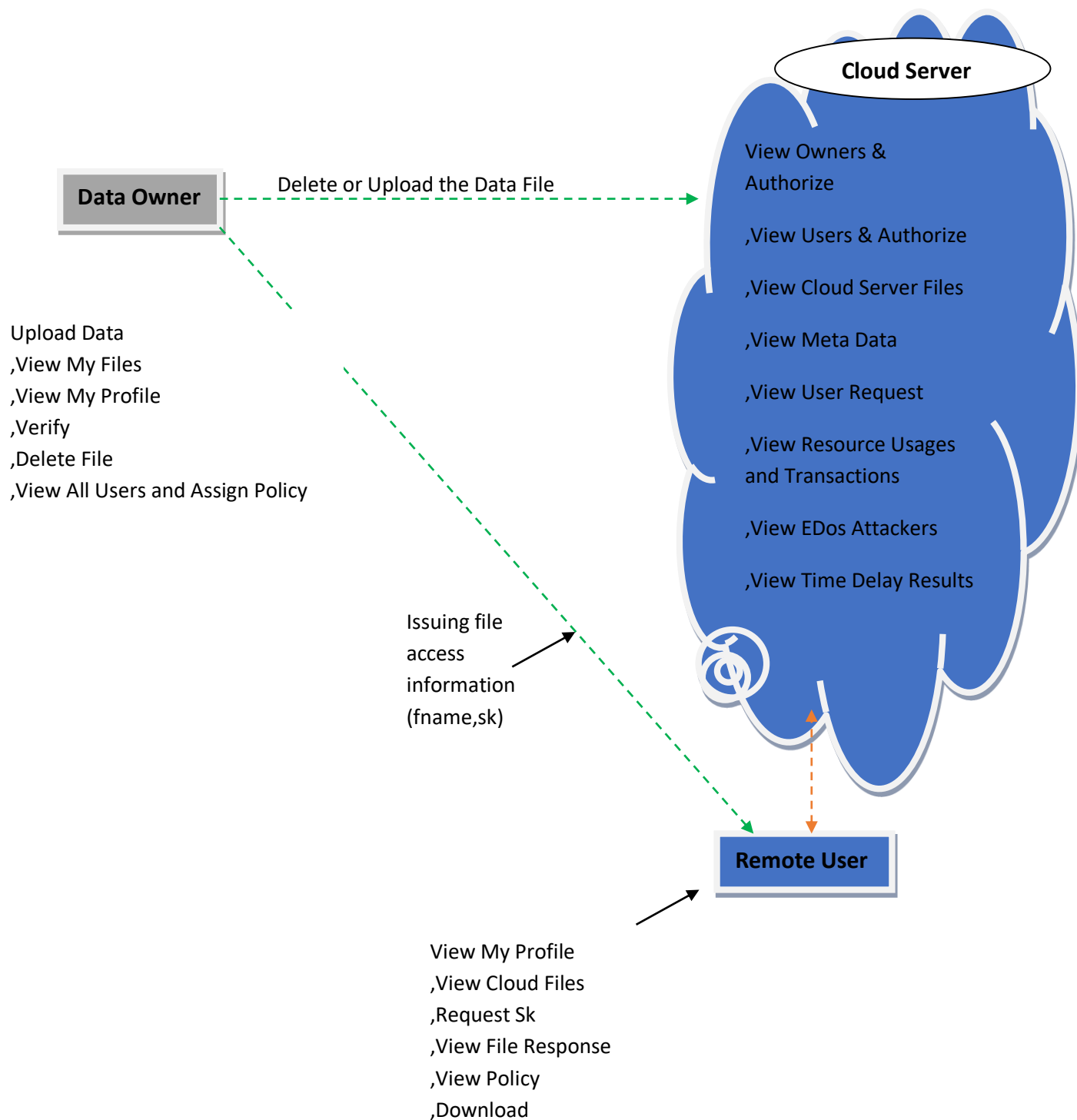


Fig 4.3: State Diagram

4.4 DATA FLOW DIAGRAM:

A Data Flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The later is usually indicated however by two separate arrows since these happen at different type.

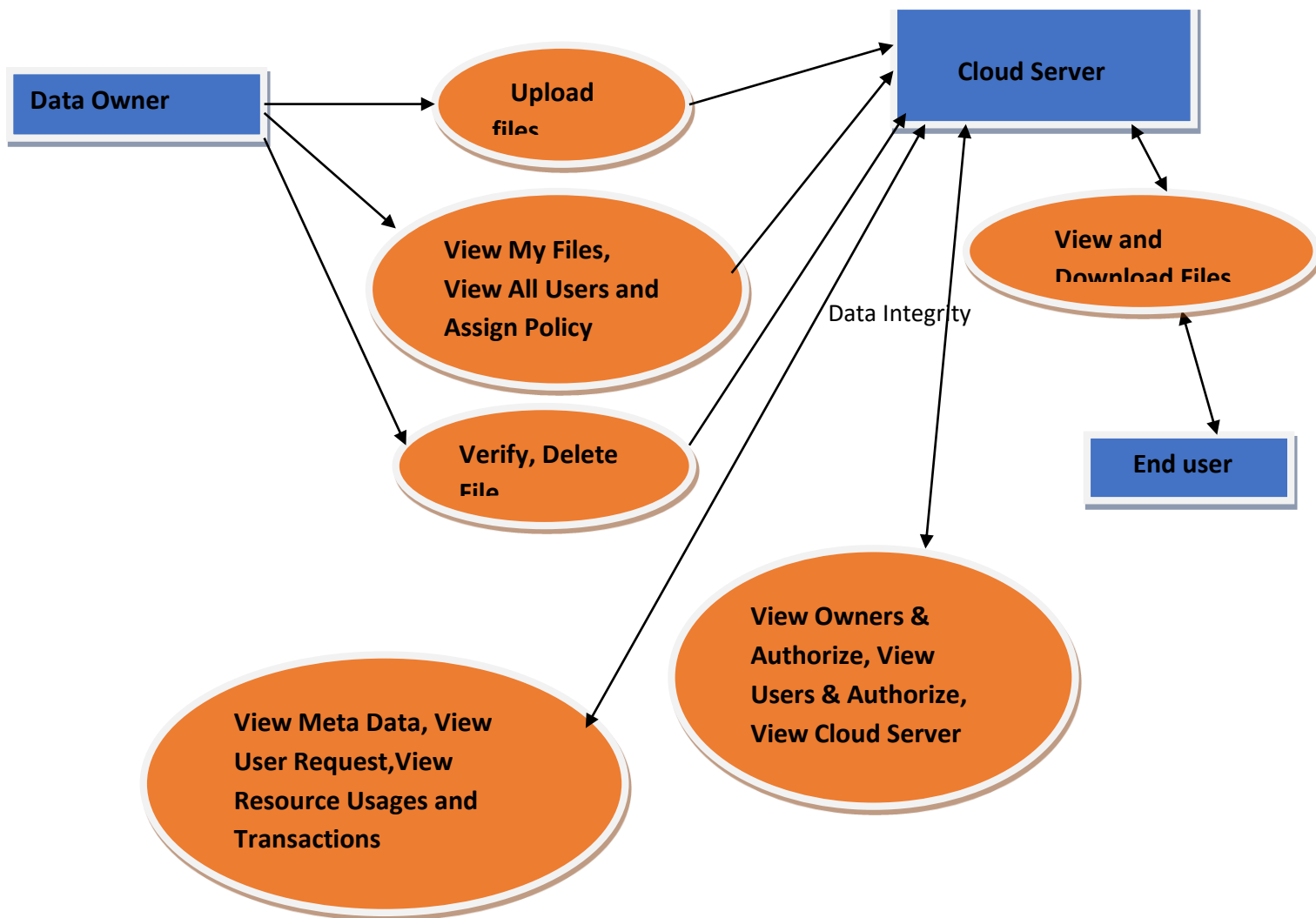


Fig 4.4: data flow diagram

4.5 ACTIVITY DIAGRAM: A State diagram/Activity diagram is a specification of the sequences of states that an object or an interaction goes through in response to events during its life, together with its responsive action

(a) Data owner

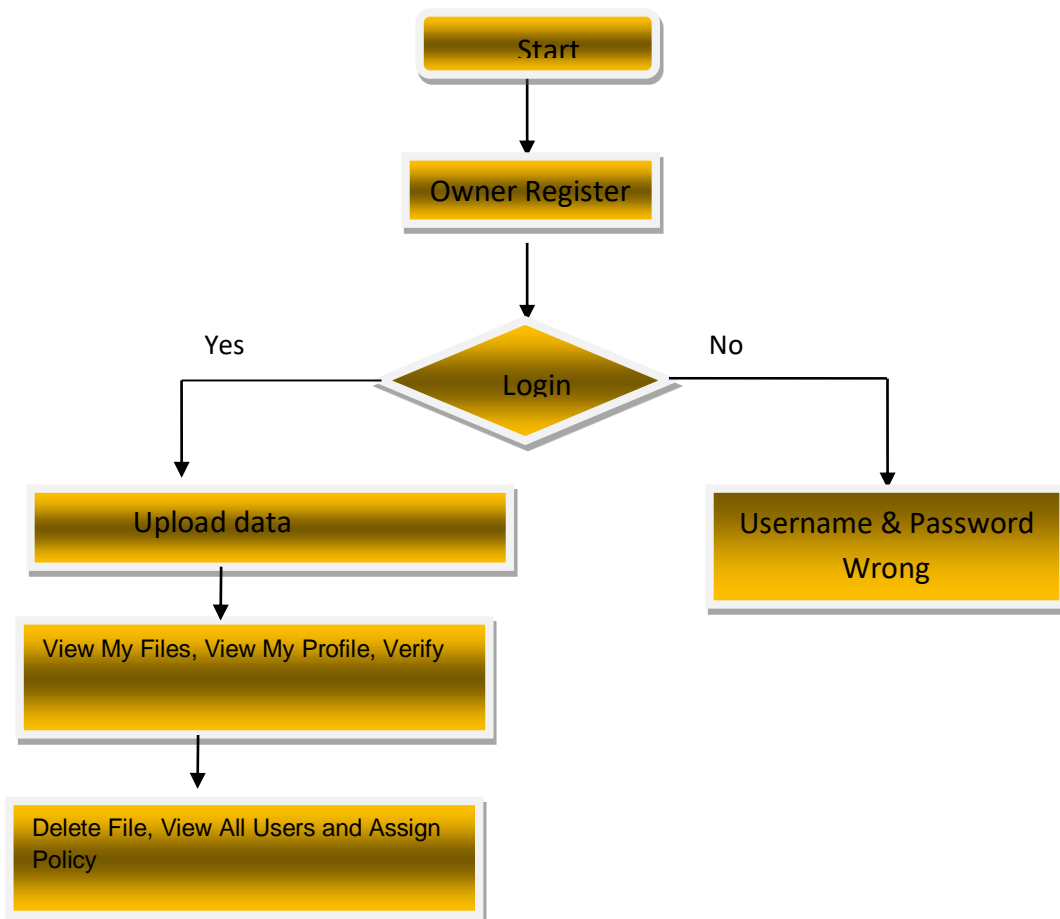


Fig 4.5: (a) data owner

4.5.1 Flow Chart: (b) User

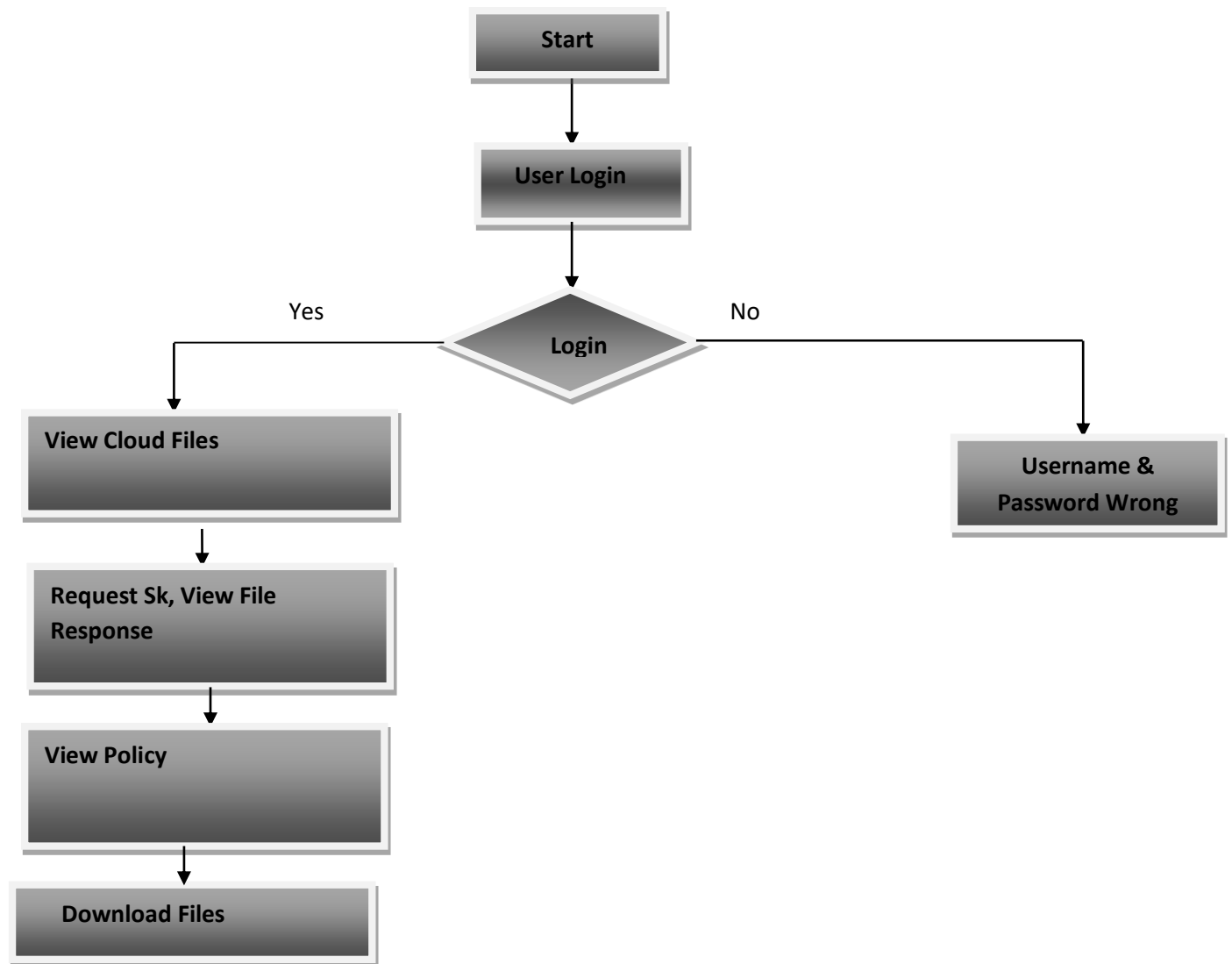


Fig 4.5.1: (b) user

4.5.2 Flow Chart: (c) Cloud Server:

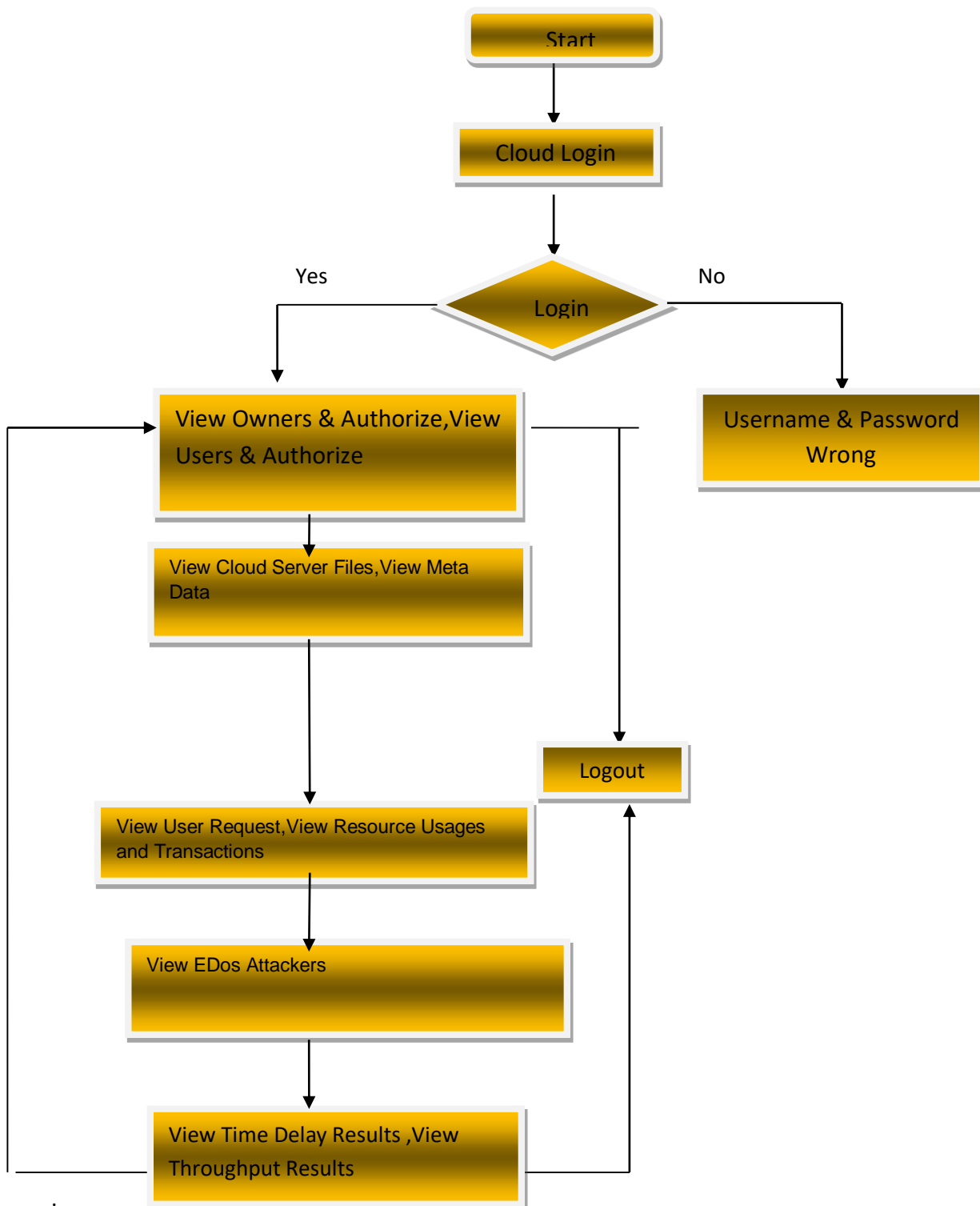


Fig 4.5.2 :(c) cloud server

4.6 Sequence Diagram:

A sequence diagram is a graphical view of a scenario that shows object interaction in a time-based sequence what happens first, what happens next. Sequence diagrams establish the roles of objects and help provide essential information to determine class responsibilities and interfaces

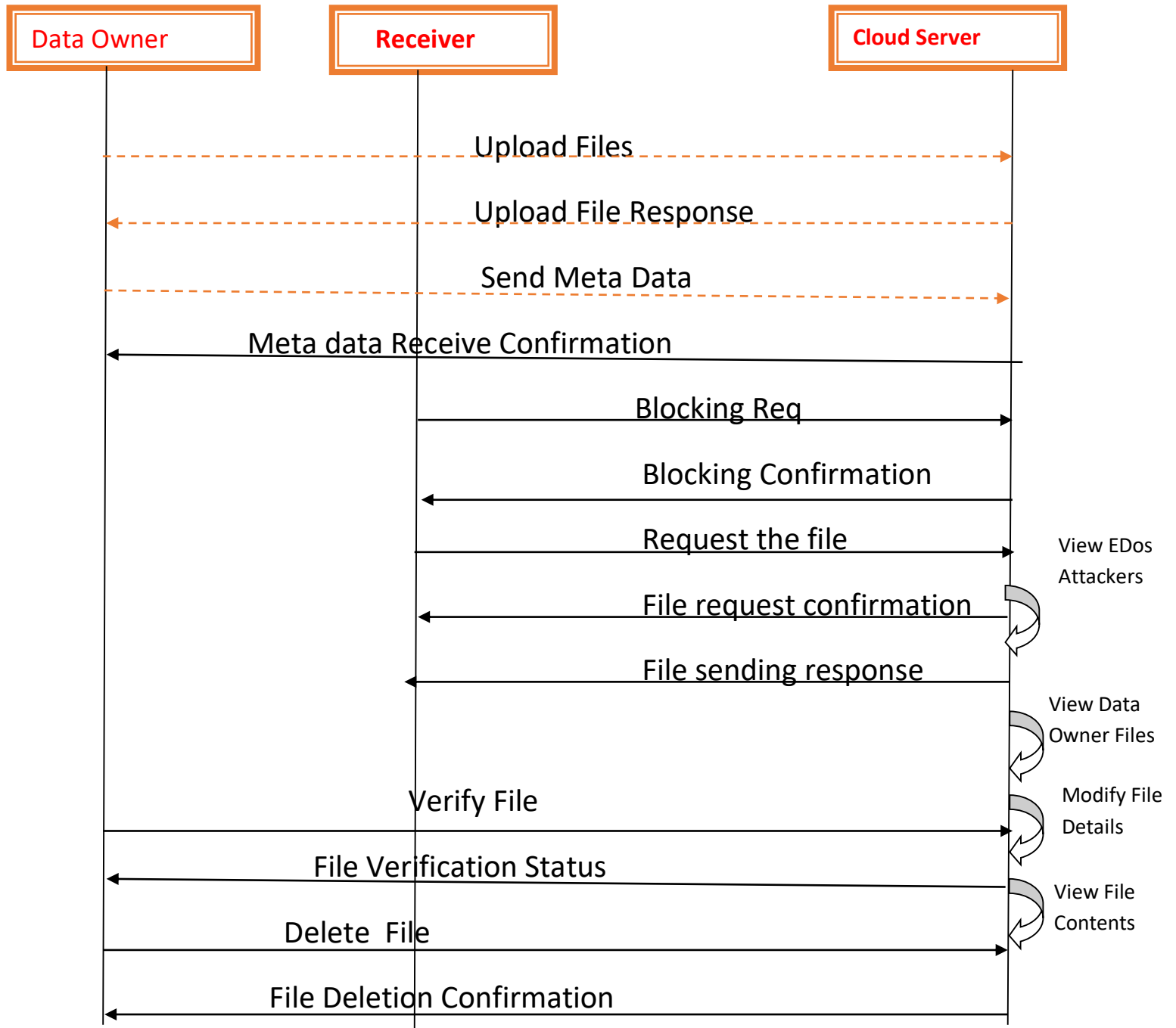


Fig 4.6: sequence diagram

4.7 Use Case Diagram:

Use case is a description of a set of sequence of actions that a system performs that yields an observable result of value to a particular things in a model. User is an actor and these are use cases are login, view work details, assign work, approval link, view voter request details, view ward member and helper details.

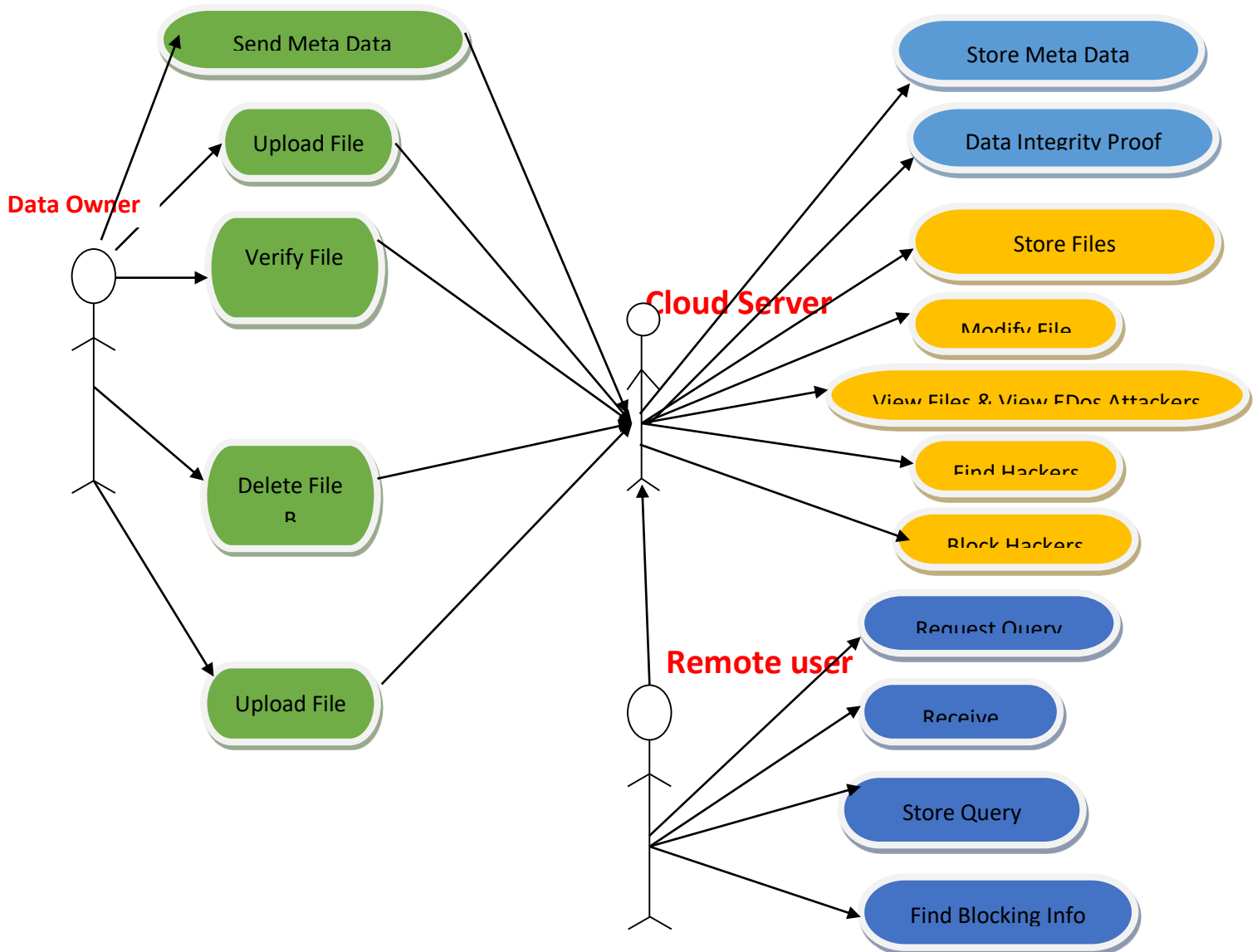


Fig 4.7: Use case diagram

5. IMPLEMENTATION

5.1 Modules:

- Data owners
- Data users
- Cloud servers

5.2 MODULE DESCRIPTION:

(a)Data Owners:

Data owners are the owner and publisher of files and pay for the resource consumption on file sharing. As the payers for cloud services, the data owners want the transparency of resource consumption to ensure fair billing. The data owners require the cloud provider to justify the resource usage. In our system, the data owner is not always online.

(b)Data Users:

Data users want to obtain some files from the cloud provider stored on the cloud storage. They need to be authenticated by the cloud provider before the download (to thwart EDOS attacks). The authorized users then confirm (and sign for) the resource consumption for this download to the cloud provider.

(c)Cloud Server:

Cloud provider hosts the encrypted storage and is always online. It records the resource consumption and charges data owners based on that record. The cloud is not public-accessible in our system as it has an authentication based access control. Only data users satisfying the access policy can download the corresponding files. The cloud provider also collects the proof of the resource consumption to justify the billing.

6. 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:

6.1 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.

6.2 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.

6.3 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.

SYSTEM SECURITY MODEL:

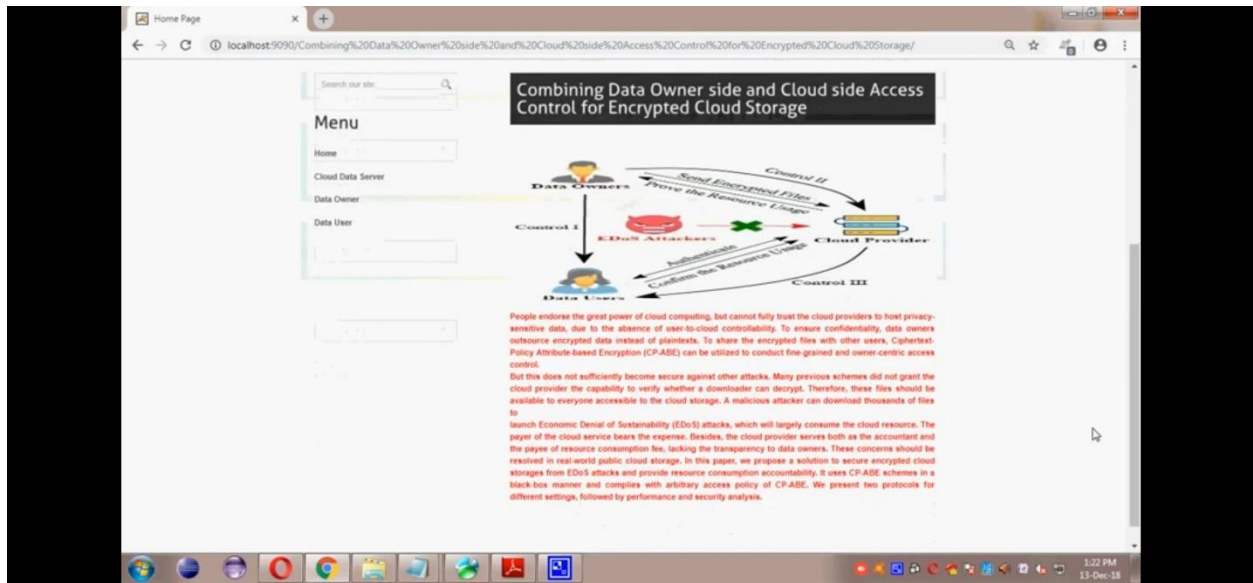
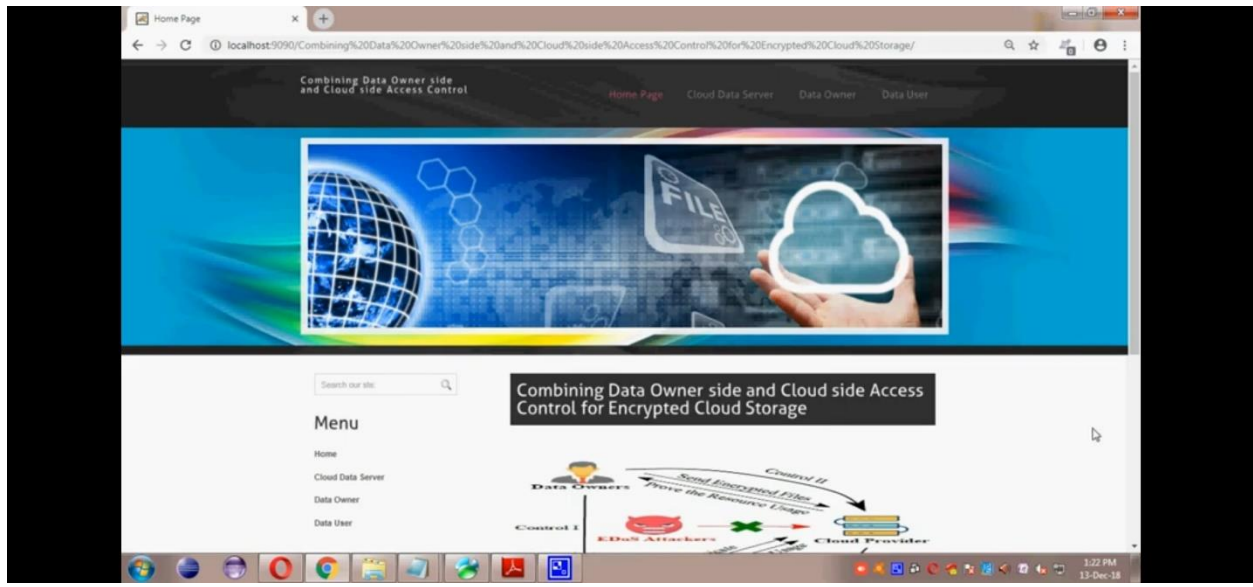
Security Assumptions and Requirements Data owners are trusted and data users can be considered as malicious adversaries. Users may try to cheat for files and launch the EDoS attacks. But authorized users are assumed not to collude with unauthorized users which is impossible to thwart and beyond the scope of this paper.

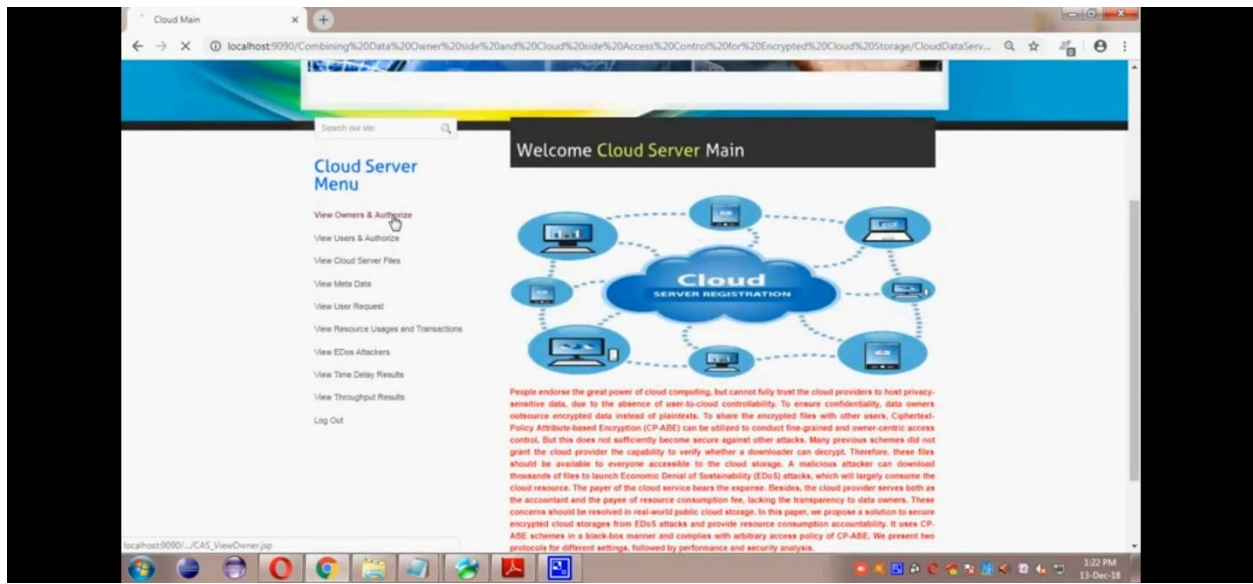
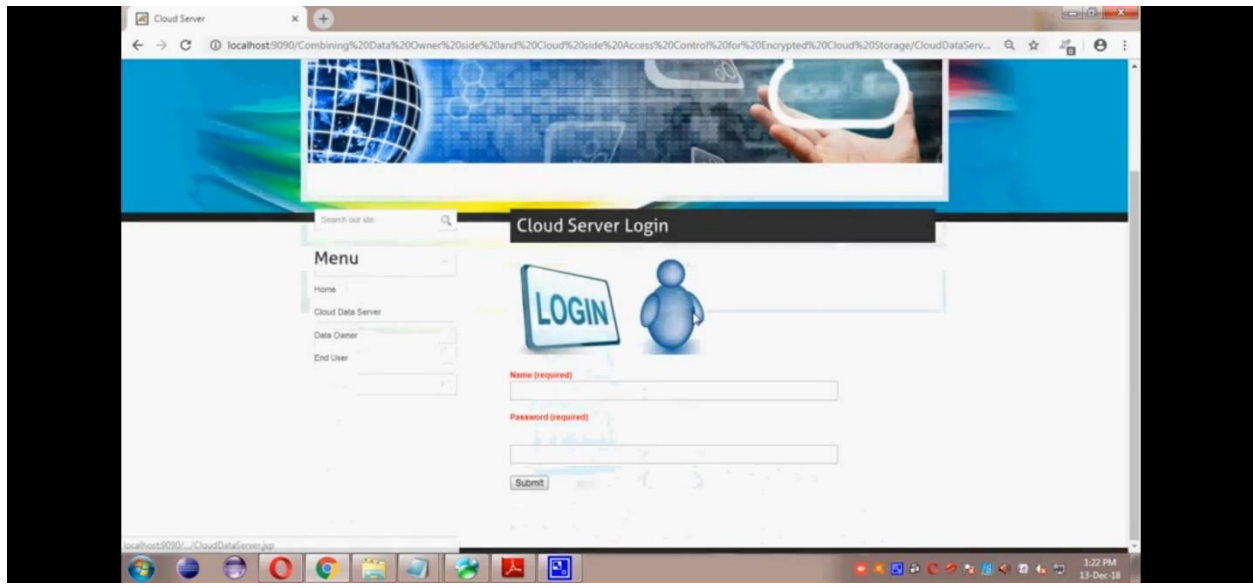
A. Security against EDoS Attacks: EDoS attackers are those that do not satisfy the access policy (i.e., unauthorized users) but want to trigger the cloud provider to send something through the network, as a result the resource consumption increases. To the wart such attacks, the cloud provider uses authentication.

B. Resource Consumption Accounting: for a user whose attribute set A_i does not satisfy the access policy

- 1) The user cannot output a valid pre image in POP;
- 2) The user cannot obtain the signing key (SK) in FOP. Without the loss of generality, we assume that the cloud provider has never been granted any attributes and doesn't collude with any authorized users. The result above can also be applied to the cloud provider

7. OUTPUT SCREENS








View Files

localhost:9090/Combining%20Data%20Owner%20side%20and%20Cloud%20side%20Access%20Control%20for%20Encrypted%20Cloud%20Storage/CDS_ViewFiles.j...



View Cloud Data Server Files !!!


Owner Name	Encrypted Contents	Decrypted Contents	FileName	Hash Code	Secret Key	Date
Gokul	PhRp0u1PeF10h1bnp72P0ai 9uF8h2U0L3p0d1Pg0PCuA 101uP2u100p1c1cT81V5u0n V5dC5e:3413T4KJd1QC8uVud1 101tc0y001aef2Y5510811k 8uadu1T4eC-u100g1C1u18y m0n101nauU8uWvduVzdc5nZk RQvK3huW0Z1o1nV12K1u2C1p Q1Ag1G1g1h1C1g1Vh1nu1u1y BuY0uPK11c0V1c3Qz2V0uGfY H11u0Vv1C3uW0u11k70Qg1C 1agm75uWu0Cp100u1C0u1k1c3 CvH0cu1u1B1cuW011MFF0V0C Acl12510g03k1c1u10P0F58J aduV2581c2VvWf1270n1tu1u	<title>Authentication Page</title> <% include file="connect.jsp"%> <% page import="java.util.Date"%> <% String name=request.getParameter("userid"); String pass=request.getParameter("pass"); try{	CloudAuth.jsp	7b07299a70a00b74fe02d7310d41449f0ba22b	08f1ca0d99	12/12/2018 18:19:35

[Go Back](#)

View Meta

localhost:9090/Combining%20Data%20Owner%20side%20and%20Cloud%20side%20Access%20Control%20for%20Encrypted%20Cloud%20Storage/Tpa_ViewMDat...

Home Page Cloud Data Server Data Owner Data User



View Meta Data !!!

Owner Name	FileName	MAC	Secret Key	Date
CloudAuth.jsp	Gokul	7b07299a70a00b74fe02d7310d41449f0ba22b	08f1ca0d99	12/12/2018 18:19:35

[Back](#)

View Transactions

localhost:9090/Combining%20Data%20Owner%20side%20and%20Cloud%20side%20Access%20Control%20for%20Encrypted%20Cloud%20Storage/CDS_ViewTrans...



View All **Resource Usage** and Transaction Details !!!

Transaction Id	Transacted User	File Name	Task	Date
1	admin	CloudBath.jpg	Upload	13/12/2018 14:13:38
2	admin	CloudBath.jpg	Retrieved	13/12/2018 12:40:33
3	admin	CloudBath.jpg	Retrieved	13/12/2018 12:40:33

[Go Back](#)

1:22 PM
13-Dec-18

View EDos Attackers

localhost:9090/Combining%20Data%20Owner%20side%20and%20Cloud%20side%20Access%20Control%20for%20Encrypted%20Cloud%20Storage/CDS_ViewAttac...

Home Page Cloud Data Server Data Owner Data User



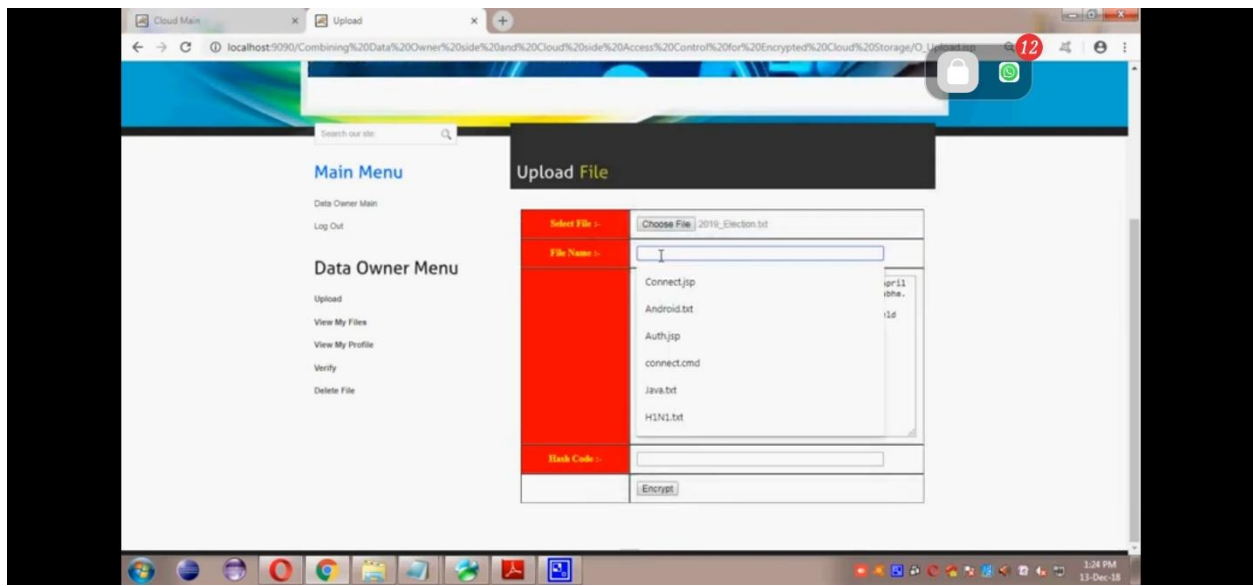
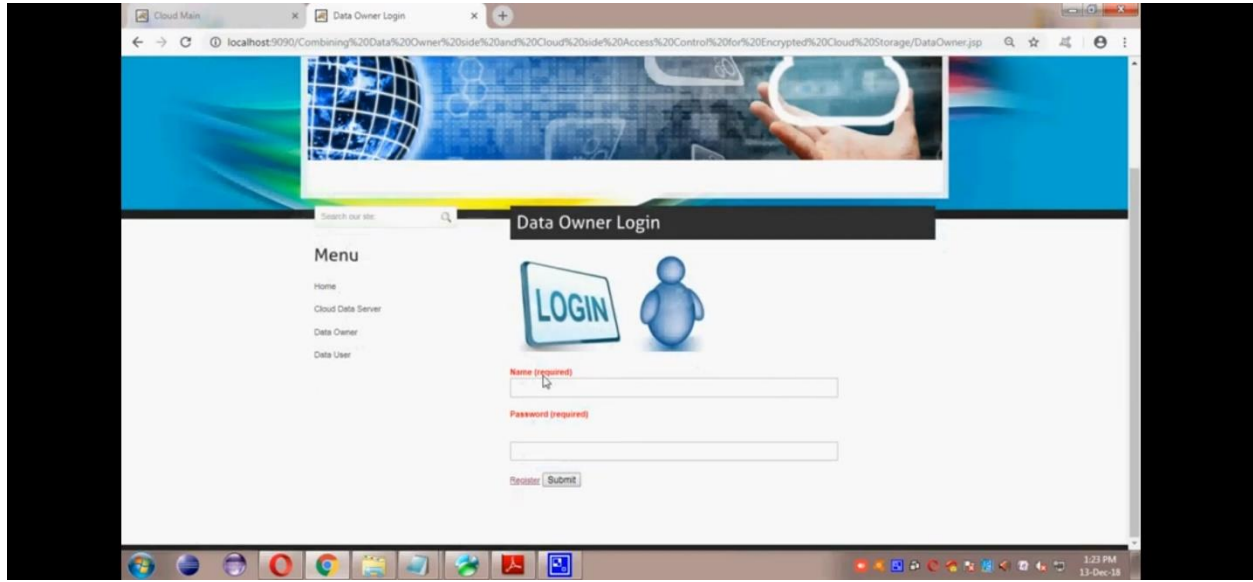
View All **EDos Attackers** Details !!!

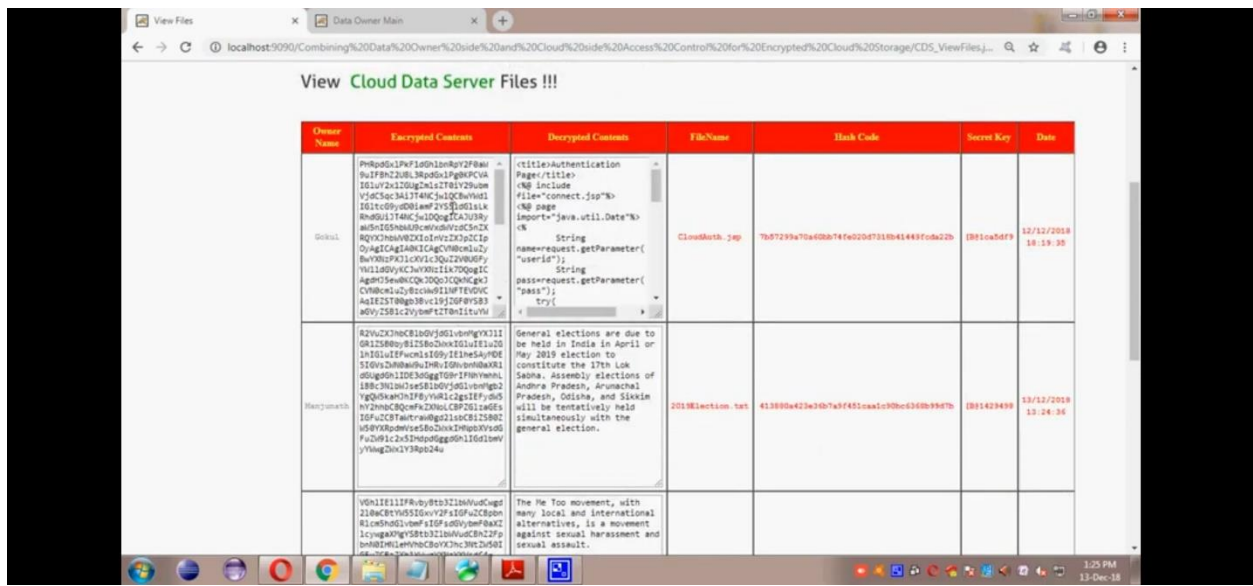
Attacker Id	Attacked User Name	File	Type	Date
1	hacker	CloudBath.jpg	Malicious Data Attack	13/12/2018 12:54:43

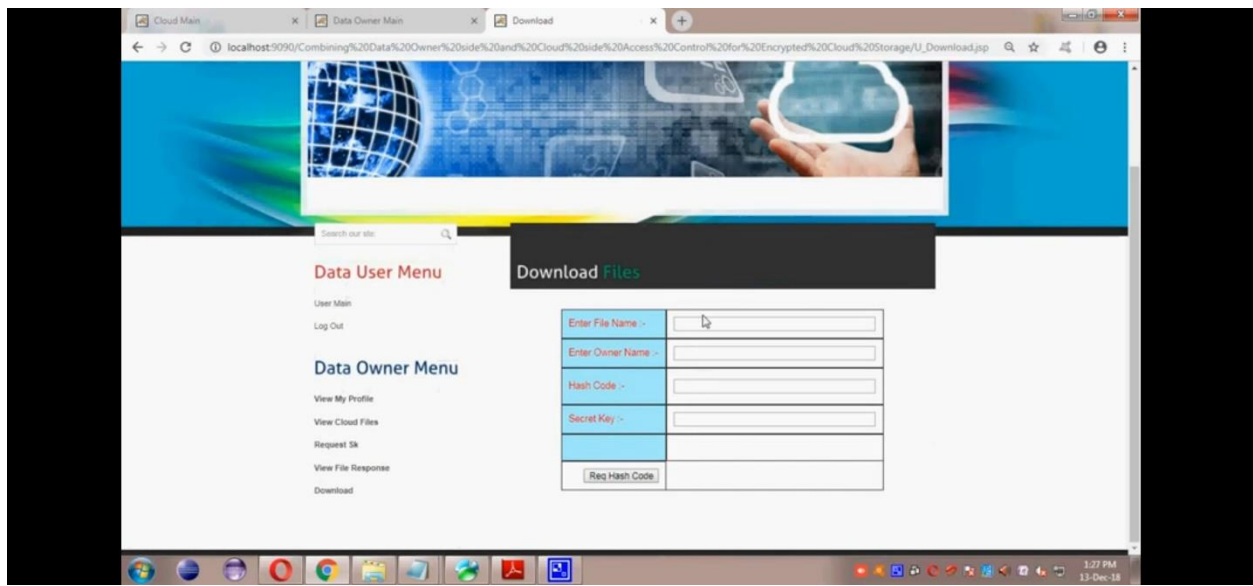
[Go Back](#)

localhost:9090/~/CDS_ViewAttackers.jpg

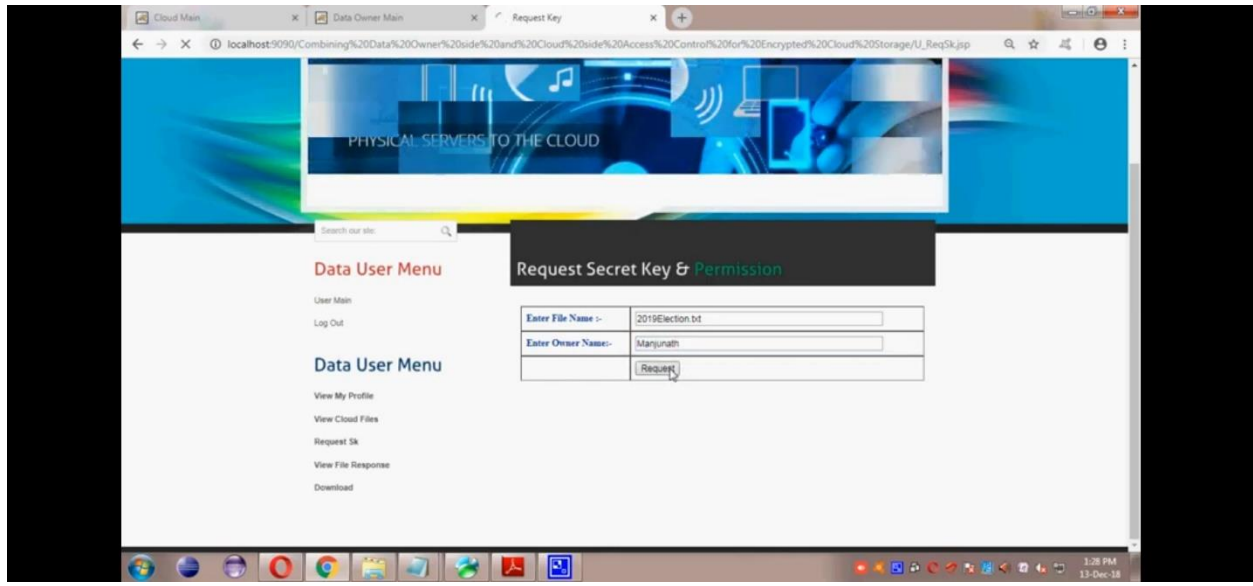
1:22 PM
13-Dec-18











8. CONCLUSION

This, we propose a combined the cloud-side and data owner-side access control in encrypted cloud storage, which is resistant to DDOS/EDOS attacks and provides resource consumption accounting. Our system supports arbitrary CP-ABE constructions. The construction is secure against malicious data users and a covert cloud provider. We relax the security requirement of the cloud provider to covert adversaries, which is a more practical and relaxed notion than that with semi-honest adversaries. To make use of the covert security, we use bloom filter and probabilistic check in the resource consumption accounting to reduce the overhead. Performance analysis shows that the overhead of our construction is small over existing systems

9. BIBLIOGRAPHY

Q. Zhang, L. Cheng, and R. Boutaba, “Cloud computing: state-of-the-art and research challenges,” *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, 2010.

K. Ren, C. Wang, and Q. Wang, “Security challenges for the public cloud,” *IEEE Internet Computing*, no. 1, pp. 69–73, 2012.

L. Zhou, Y. Zhu, and A. Castiglione, “Efficient k-NN query over encrypted data in cloud with limited key-disclosure and offline data owner,” *Computers & Security*, vol. 69, pp. 84–96, 2017.

S. Hu, Q. Wang, J. Wang, Z. Qin, and K. Ren, “Securing SIFT: Privacy-preserving outsourcing computation of feature extractions over encrypted image data,” *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3411–3425, 2016.

H.-M. Sun, Y.-H. Chen, and Y.-H. Lin, “oPass: A user authentication protocol resistant to password stealing and password reuse attacks,” *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 651–663, 2012.

L. Harn and J. Ren, “Generalized digital certificate for user authentication and key establishment for secure communications,” *IEEE Transactions on Wireless Communications*, vol. 10, no. 7, pp. 2372–2379, 2011.

Sekar and P. Maniatis, “Verifiable resource accounting for cloud computing services,” in *Proceedings of the 3rd ACM workshop on Cloud computing security workshop*. ACM, 2011, pp. 21–26.