“Preserving Privacy Of Data Over Cloud”

Bachelor of Engineering
in
Computer Engineering

Submitted by

Mr. Sanket	Laddha	15CE1053
Ms. Himadri	Patil	16CE5004
Ms. Asha	Karkera	16CE5007
Mr. Aditya	Chaudhari	15CE2015

Guided by

(Mrs. Sheetal Ahir)

Department of Computer Engineering

Ramrao Adik Institute Of Technology

Dr. D. Y. Patil Vidyanagar, Sector-7, Nerul, Navi Mumbai-400706.
(Affiliated to University of Mumbai)

April 2019

“Preserving Privacy Of Data Over Cloud”
B.E. Project Report

Submitted in partial fulfillment of the requirements

For the degree of

Bachelor of Engineering
in
Computer Engineering

Submitted by

Mr. Sanket	Laddha	15CE1053
Ms. Himadri	Patil	16CE5004
Ms. Asha	Karkera	16CE5007
Mr. Aditya	Chaudhari	15CE2015

Guided by

(Mrs.Sheetal Ahir)

D Y PATIL
RAMRAO ADIK
INSTITUTE OF
TECHNOLOGY
NAVI MUMBAI

Department of Computer Engineering

Ramrao Adik Institute Of Technology

Dr. D. Y. Patil Vidyanagar, Sector-7, Nerul, Navi Mumbai-400706.

(Affiliated to University of Mumbai)

April 2019

Ramrao Adik Institute of Technology

(Affiliated to the University of Mumbai)
Dr. D. Y. Patil Vidyanagar, Sector-7, Nerul, Navi Mumbai-400706.

CERTIFICATE

This is to certify that, the dissertation titled
“ Preserving Privacy Of Data Over Cloud ”
is a bonafide work done by

Mr. Sanket	Laddha	15CE1053
Ms. Himadri	Patil	16CE5004
Ms. Asha	Karkera	16CE5007
Mr. Aditya	Chaudhari	15CE2015

and is submitted in the partial fulfillment of the requirement for the degree of

Bachelor of Engineering
in
Computer Engineering
to the
University of Mumbai

Supervisor
(**Mrs. Sheetal Ahir**)

Project Coordinator
(**Mrs. Smita Bharne**)

Head Of Department
(**Dr. Leena Ragha**)

Principal
(**Dr. Ramesh Vasappanavara**)

Project Report Approval for B.E

This is to certify that the project ‘B’ entitled ***“ Preserving Privacy Of Data Over Cloud*** ” is a bonafide work done by ***Mr. Sanket Laddha*** , ***Ms. Himadri Patil*** , ***Ms. Asha Karkera , Mr. Aditya Chaudhari*** under the supervision of Mrs. Sheetal Ahir. This dissertation has been approved for the award of Bachelor’s Degree in Computer Engineering, University of Mumbai.

Examiners :

1. . . . .

2. . . . .

Supervisors :

1. . . . .

2. . . . .

Principal :

. . . .

Date : . . . /. . . /. . . . . .

Place : . . . .

Declaration

We declare that this written submission represents my ideas in my own words and where other's ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Mr Sanket Laddha	15CE1053	____________________
Ms Himadri Patil	16CE5007	____________________
Ms. Asha Karkera	16CE5004	____________________
Mr. Aditya Chaudhari	15CE2015	____________________

Date : . . . /. . . /. . . . . .

Abstract

Cloud Data Services are not only used to store data on cloud at one place for single user but also to share among multiple users. The integrity of cloud data is mistrustful or doubtful due to the failures in hardware and software and human errors. Data owners and public verifiers have been enabled to efficiently audit cloud data integrity using various mechanisms. This disables the recovering of the entire data from cloud server. Public auditing on integrity of shared data ,with this existing mechanism will consequently reveal restricted information identity privacy to public verifiers .We propose a system that supports public auditing on shared data stored over the cloud. Specifically we take the advantage of the ring signatures to compute verification of metadata needed to audit the correctness of shared data. With our mechanism the identity of user in shared data is kept private from public verifiers .Moreover, this mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one but one. Our results shows the effectiveness and efficiency of this mechanism when auditing shared data integrity.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Cloud computing has been envisioned as the next generation information technology (IT) architecture for enterprises, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk. As a disruptive technology with profound implications, cloud computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data are being centralized or outsourced to the cloud. From users' perspective, including both individuals and IT enterprises, storing data remotely to the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with location independence, and avoidance of capital expenditure on hardware, software, and personnel maintenances, etc.

1.1.1 Cloud computing

Cloud computing, or something being in the cloud, is an expression used to describe a variety of different types of computing concepts that involve a large number of computers connected through a real-time communication network such as the Internet. In science, cloud computing is a synonym for distributed computing over a network and means the ability to run a program on many connected computers at the same time. The phrase is also more commonly used to refer to network-based services which appear to be provided by real server hardware, which in fact are served up by virtual hardware, simulated by software running on one or more real machines. Such virtual servers do not physically exist and can therefore be moved around and scaled up (or down) on the fly without affecting the end user arguably, rather like a cloud. The popularity of the term can be attributed to its use in marketing to sell hosted services in the sense of application service provisioning that run client server software on a remote location.

Advantages

Cloud computing relies on sharing of resources to achieve coherence and economies of scale similar to a utility (like the electricity grid) over a network. At the foundation of cloud computing is the broader concept of converged infrastructure and shared services. The cloud also focuses on maximizing the effectiveness of the shared resources. Cloud resources are usually not only shared by multiple users but are also dynamically re-allocated per demand. This can work for allocating resources to users. For example, a cloud computer facility, which serves European users during European business hours with a specific application (e.g. email) while the same resources are getting reallocated and serve North American users during North America's business hours with another application (e.g. web server). This approach should maximize the use of computing powers thus reducing environmental damage as well since less power, air conditioning, rack space, etc. is required for a variety of functions.

The term "moving to cloud" also refers to an organization moving away from a traditional CAPEX model (buy the dedicated hardware and depreciate it over a period of time) to the OPEX model (use a shared cloud infrastructure and pay as you use it).

Proponents claim that cloud computing allows companies to avoid upfront infrastructure costs, and focus on projects that differentiate their businesses instead of infrastructure. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and enables IT to more rapidly adjust resources to meet fluctuating and unpredictable business demand.

1.2 Motivation

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.

Following figure 1.1 depicts various services of Cloud computing.

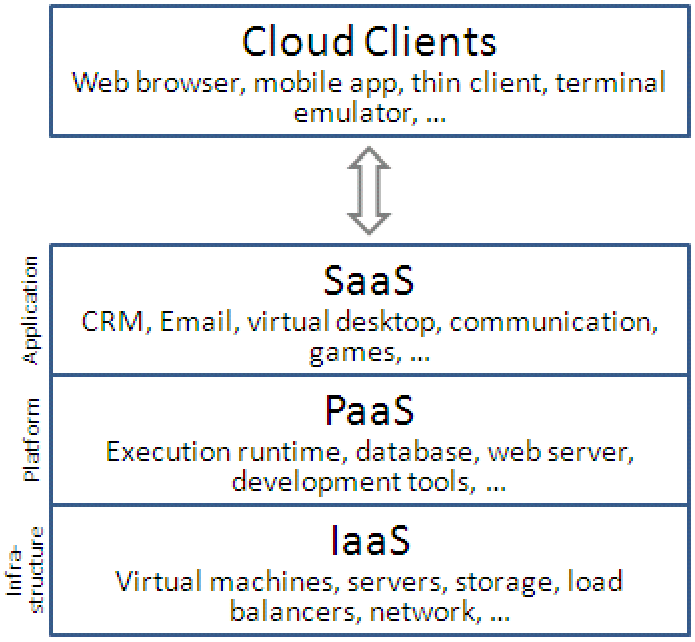

Cloud Clients
Web browser, mobile app, thin client, terminal emulator, ...
SaaS
CRM, Email, virtual desktop, communication, games, ...
PaaS
Execution runtime, database, web server, development tools, ...
IaaS
Virtual machines, servers, storage, load balancers, network, ...
Application
Platform
Infra-structure

Figure 1.1.Different services of cloud computing [8]

1.3 Problem Definition

The aim is to design a system to audit the integrity of shared data in the cloud with static groups.

- Integrity Threats
- Security Threats
- Our mechanism should be designed to achieve Public auditing, correctness identity privacy, traceability

1.4 Objectives

The main objective of this project is to propose a novel privacy-preserving mechanism that supports public auditing on shared data stored in the cloud using Group Oriented Ring Signature.

- **Public Auditing**: The third party auditor is able to verify the integrity of shared data for a group of users without retrieving the entire data

- **Correctness**: The third party auditor is able to correctly detect whether there is any corrupted block in shared data

- **Identity Privacy**: During auditing, the TPA cannot distinguish the identity of the signer on each block in shared data.

1.5 Organization of report

This part explains the organization of the project.

- **CHAPTER 1 :**

  This chapter contains the brief introduction of Cloud computing and its services.

- **CHAPTER 2 :**

  This chapter contains the Literature survey of more than five papers.

- **CHAPTER 3:**

  This chapter contains the existing system and its drawbacks. It also contains the details of proposed system, modules.

- **CHAPTER 4 :**

This chapter contains the schedule for the project and detail execution of the plan.

- **CHAPTER 5 :**

This chapter contains the design and explanation of the system.

- **CHAPTER 6 :**

This chapter contains the implementation details, result and cost benefit analysis

- **CHAPTER 7:**

This chapter contains the conclusion and future work.

Chapter 2

Literature Survey

2.1 Research Papers Survey

Kui Ren , Cong Wang and Qian Wang presented Security Challenges for the Public Cloud. Security and privacy is one fundamental obstacle to cloud computing's success. One possible approach to enforce data access without relying on cloud servers could be to encrypt data in a differentiated manner and disclose the corresponding decryption keys only to authorized users. This approach usually suffers from severe performance issues. Multitenancy security and privacy is one of the critical challenges for the public cloud, and finding solutions is pivotal if the cloud is to be widely adopted.

Dawn Song, Elaine Shi, and Ian Fischer presented a paper on Cloud Data Protection for the Masses. Although cloud-computing promises lower costs, rapid scaling, easier maintenance, and service availability anywhere, anytime, a key challenge is how to ensure and build confidence that the cloud can handle user data securely. It's impossible to develop a single data-protection solution for the cloud because the term means too many different things. A new cloud computing paradigm, data protection as a service is proposed. PaaS is a suite of security primitives offered by a cloud platform, which enforces data security and privacy and offers evidence of privacy to data owners, even in the presence of potentially compromised or malicious applications.

Ernie Brickell, Jan Camenisch, Liqun Chen described the direct anonymous attestation scheme (DAA). Trusted Computing Group (TCG) is an industry standardization body that aims to develop and promote an open industry standard for trusted computing hardware and software building blocks to enable more secure data storage. It draws on techniques that have been developed for group signatures, identity escrow, and credential systems. In fact, our scheme can be seen as a group signature scheme without the capability to open signatures (or anonymity revocation) but with a mechanism to detect rogue members.

Dan Boneh, Ben Lynn and Hovav Shacham introduced Short signatures from the Weil pairing. Short digital signatures are needed in environments where a human is asked to manually key in the signature. So, a short signature scheme based on the Computational Diffie-Hellman assumption on certain elliptic and hyper-elliptic curves is introduced. The signature length is half the size of a DSA signature for a similar level of security. Our short signature scheme is designed for systems where signatures are typed in by a human or signatures are sent over a low-bandwidth channel. The signature scheme is secure against existential forgery under a chosen message attack (in the random oracle model) assuming the Computational Diffie-Hellman problem (CDH) is hard on certain elliptic curves over a finite field of characteristic three.

Xuefeng Liu, Yuqing Zhang, Boyang Wang and Jingbo Yan proposed Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud. A secure data sharing scheme, Mona, for dynamic groups in an untrusted cloud is designed. In Mona, a user is able to share data with others in the group without revealing identity privacy to the cloud. By leveraging group signature and dynamic broadcast encryption techniques, any cloud user can anonymously share data with others. Meanwhile, the storage overhead and encryption computation cost of our scheme are independent with the number of revoked users. Mona supports efficient user revocation and new user joining. Efficient user revocation can be achieved through a public revocation list without updating the private keys of the remaining users, and new users can directly decrypt files stored in the cloud before their participation.

2.2 Outcome of Literature Survey

These various papers addresses the challenging open issue by, on one hand, defining and enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine-grained data access control to untrusted cloud servers without disclosing the underlying data contents. We achieve this goal by exploiting and uniquely combining techniques of attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. Our proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability .Extensive analysis shows that our proposed scheme is highly efficient and provably secure under existing security models.

Table 1 Comparison between existing privacy preserving mechanisms [1]

INFO	PDP[2]	WWRL[3]	ORUTA[1]
Public Auditing	YES	YES	YES
Data Privacy	NO	YES	YES
Identity Privacy	NO	NO	YES

Chapter 3

Project Proposal

3.1 Proposed Work

We only considered how to audit the integrity of shared data in the cloud with static groups. It means the group is pre-defined before shared data is created in the cloud and the membership of users in the group is not changed during data sharing. The original user is responsible for deciding who is able to share her data before outsourcing data to the cloud. Another interesting problem is how to audit the integrity of shared data in the cloud with dynamic groups— a new user can be added into the group and an existing group member can be revoked during data sharing — while still preserving identity privacy. To enable the TPA efficiently and securely verify shared data for a group of users, Oruta should be designed to achieve following properties:

(1) Public Auditing: The third party auditor is able to verify the integrity of shared data for a group of users without retrieving the entire data.

(2) Correctness: The third party auditor is able to correctly detect whether there is any corrupted block in shared data.

(3) Unforgetability: Only a user in the group can generate valid verification information on shared data.

(4) Identity Privacy: During auditing, the TPA cannot distinguish the identity of the signer on each block in shared data. We utilize ring signatures to construct group authenticators in Oruta, so that a public verifier is able to verify the integrity of shared data without retrieving the entire data while the identity of the signer on each block in

shared data is kept private from the public verifier. One Ring to Rule Them All. of the group. Group members are allowed to access and modify shared data created by the original user based on access control polices. Shared data and its verification information (i.e. signatures) are both stored in the cloud server. The third party auditor is able to verify the integrity of shared data in the cloud server on behalf of group members. Users Cloud Server Third Party Auditor (TPA) Shared Data Flow Auditing Request Auditing Report, Auditing Message, Auditing Proof.

3.2 System Requirement

Hardware Requirement

Table 2: Hardware requirements

System	Intel Processor
Hard disk	40 GB
Input	Keyboard and mouse
Ram	512 MB

Software Requirement

Table 3: Software requirement

Operating System	Windows 10
Coding language	Java, JSP, Servlet
IDE	NetBeans 7.2.1
Database	MySQL
Web server	Apache Tomcat 5.5

3.3 Proposed Methodology/Techniques

The ring signatures generated for not only able to preserve identity privacy but also able to support blockless verifiability. A public verifier is able to correctly verify shared data integrity. A public verifier cannot distinguish the identity of the signer on each block in shared data during the process of auditing.

3.3.1 MODULE DESCRIPTION

LIST OF MODULES

3.3.1.1 Ring Signature Generation

3.3.1.2 Oruta Processing

3.3.1.3 Construction of Oruta

3.3.1.1 Ring Signature Generation

With ring signatures, a verifier is convinced that a signature is computed using one of group members' private keys, but the verifier is not able to determine which one. More concretely, given a ring signature and a group of users, a verifier cannot distinguish the signer's identity with a probability more than 1=d. This property can be used to preserve the identity of the signer from a verifier.

3.3.1.2 Oruta Processing

Since the computation of a ring signature includes an identifier of a block (as presented in HARS), traditional methods, which only use the index of a block as its identifier (i.e., the index of block mj is j), are not suitable for supporting dynamic operations on shared data efficiently. The reason is that, when a user modifies a single block in shared data by performing an insert or delete operation, the indices of blocks that after the modified block are all changed, and the changes of these indices require users, who are sharing the data, to re-compute the signatures of these blocks, even though the content of these blocks are not modified.

3.3.1.3 Construction of Oruta

Now, we present the details of our public auditing mechanism. It includes five algorithms:KeyGen,SigGen,Modify,ProofGenandProofVerify.InKeyGen,users generate their own public/private key pairs. In SigGen, a user (either the original user or a group user) is able to compute ring signatures on blocks in shared data by using its own private key and all the group members' public keys. Each user in the group is able to perform an insert, delete or update operation on a block, and compute the new ring signature on this new block in Modify.ProofGenis operated by a public verifier and the cloud server together to interactively generate proof of possession of shared data. In Proof Verify, the public verifier audits the integrity of shared data by verifying the proof. Note that for the ease of understanding, we first assume the group is static, which means the group is pre-defined before shared data is created in the cloud and the membership of the group is not changed during data sharing. Specifically, before the original user outsources shared data to the cloud, he/she decides all the group members.

3.3.2 ALGORITHM

Group-oriented ring signature

Designated verifier signature first proposed by Jakobsson, Sako and Impagliazzin 1996. It is very useful in controlling the message transmission range. In this kind of signature schemes, nobody besides the designated person can verify the signature. Ma et al. extended the notion, and proposed the group-oriented encryption. In group-oriented signature, nobody besides the designated group can verify the signature. Obviously, in a PKI authentication frame, each person should have his own key pair. So the core issue of group-oriented signature is how to design a scheme in which each group member is allowed to verify the signature independently. As we have mentioned above, a ring signature with limited verification range is necessary in some instances. A signer can perform following steps to produce a group-oriented ring signature.

Step1. The ring signer *i P* chooses a random number *y* and computes value *Q* . Here $0 \leq v^{d^*} < y$.

$$\sum_{i=0}^{n} v^{d^*} y^{\prod_{j=0}^{n} e'_{j-i}} = Q$$ [4]

We define the set of verifiers as (V1,V2,…. ,Vn). Here e_i is the public key of the designated verifier *V1* . Without loss of generality, we suppose that V_i will verify the signature produced by P_i.

Step2. Compute and publish value *W* .

$$y^{\prod_{i=0}^{n} e'_i} = W$$ [4]

Step3. Perform the ***Generation Algorithm*** in Section 4, and output a group-oriented ring signature

$$(m, P_0, P_1, P_2, \cdots, P_n, Q, \Omega, c, W, y, U, e^*)$$ [4]

To verify the signature the verifier V_i takes following steps.

Step1. Compute x_o = H(m||c)

Step2. Compute the value *v* .

$$\left\lfloor \left(\frac{Q}{W^{d'_i}} \right)^{e^*} \right\rfloor \bmod y \equiv \left\lfloor \left(\frac{\sum_{i=0}^{n} Z^{d^*} y^{\prod_{j=0}^{n} e'_{j-i}}}{y^{d'_i \prod_{j=0}^{n} e'_{j-i}}} \right)^{e^*} \right\rfloor \bmod y = v$$ [4]

Step3. Perform the ***Verification Algorithm*** and verify if the following equation holds.

$$x_0^{e_0} + x_1^{e_1} + x_2^{e_2} + \cdots + x_i^{e_i} + \cdots + x_n^{e_n} = v$$ [4]

If above equation holds, it shows that the ring signature is valid.

Chapter 4

Planning And Formulation

4.1 Schedule for Project

Software requirement specification is a fundamental document, which forms the foundation of the software development process. It not only lists the requirements of a system but also has a description of its major feature. An SRS is basically an organizations understanding of a customer or potential clients system requirements and dependencies at a particular point in time prior to any actual design or development work. It’s a two way insurance policy that assures that both the client and the organization understand the others requirements from that perspective at a given point of time. The SRS also functions as a blueprint for completing a project with as little cost growth as possible. The SRS is often referred to as the parent document because all subsequent project management documents, such as design specifications, statements of work, software architecture specifications, testing and validation plans, and documentation plans, are related to it. It is important to note that and SRS contains functional and non-functional requirements only; it doesn’t offer design suggestions, possible solutions to technology or business issues, or any other information other than what the development team understands the customers system requirements to be.

Functional Requirements : Functional requirements characterize a component of a software framework and how the frame- work must carry on when given particular inputs or conditions. These may incorporate estimations, data manipulation and handling and other particular functionality. The details of our public auditing mechanism in Oruta includes: Key-Gen, Sig-Gen, Modify, Proof-Gen and Proof Verify. In Key-Gen, users generate their own public/private key pairs. In Sig-Gen, a user is able to compute ring signatures on blocks in shared data. Each user is able to perform an insert, delete or update operation on a block, and compute the new ring signature on this new block in Modify. Proof Gen is operated by the TPA and the cloud server together to generate a proof of possession of shared data. In Proof-Verify, the TPA verifies the proof and sends an auditing report to the user.

4.2 Detail Plan of Execution

In this project we made use of iterative model for the development of the overall project. We divided the project into different modules and developed each those modules in a specific time period. During the Development we got through requirement, design, implementation, testing, and analysis of each module. Developing a public auditor is a tedious task. It takes long time and big efforts to develop a privacy preserving public auditor. Lots of aspects are required to be fulfilled while designing the system .The aspects included while developing the system are:

- Creating an online file storage system
- Creating a user login system
- Creating a file digital signature algorithm using Ring Signatures
- Creating a Database for handling the queries and keep records.
- Creating the Third Party Auditor

To manage all these critical components of project planning is must.

Firstly, we plan to develop the cloud storage system of the shared data. The data will be uploaded and broken down into chunks. These chunks will be stored separately

A user login system must be created that allows users to login and upload and download their files. Each user will have their own login information such as username and password. Moreover, a system must be made that allows users to perform insert, update and delete operations on their own files. Furthermore, the system must be extended to allow users to form groups.

Following this, a ring signature generation algorithm must be created. This algorithm should create a signature for data shared between a group. Each shared data chunk must be signed using this algorithm every time it is modified by a user in the group

The next step is to create a database system to keep records and handle each query. Every time a file is uploaded and broken into chunks, the locations and metadata for each chunk must be stored on the database. Moreover, the database should hold records for ever operation per- formed on the data by any user.

Now the third party auditor must be created. The auditor should be able to take verification requests from a user, and then request the server for metadata. The TPA then verifies the actual data chunk and notifies the user of its authenticity. If any modifications to the file are required, then they are made and the server is notified. The server then notifies the administrator, and the file owner in turn. After this, all other steps must be merged and all entities and their communication channels should be finalized. The User Interface will now be created and modified for the cloud application

Chapter 5

Design of System

5.1 System Design

The system comprises of three nodes: The third party verifier i.e. TPA, group of users, and the cloud. The group encompasses two types of users: the owner of a data and a few group users. The data owner and group users are both group members .Members in a group can access and modify data shared between them which is created by the owner of data based on access control polices. Cloud server stores both shared data and verification metadata. Instead of the overhead of integrity check of data on group users, a third party auditor check the data integrity for them. The user is responsible for whether his/her data should be shared among a particular group of users, before outsourcing data to the cloud. When a user needs to plaid the data integrity, He/she directs an auditing call to the TPA. After reception of the auditing request, the TPA generates an auditing message to the cloud, and retrieves an auditing proof of shared data from the cloud server. Then the TPA verifies that the auditing proof is accurate. Lastly, the TPA sends an auditing report to the user grounded on the result of the verification. System Architecture

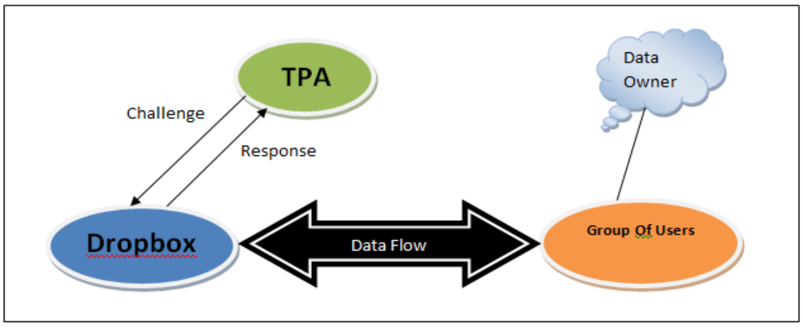


Figure 5.1 System Architecture [1]S

5.2 Input and Output Design

5.2.1 Input Design

There is a link between a user and the information system it is called input design. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things: What data should be given as input? How the data should be arranged or coded? The dialog to guide the operating personnel in providing input. Methods for preparing input validations and steps to follow when error occur.

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

5.2.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the in- formation clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the systems relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis de- sign computer output, they should identify the specific output that is needed to meet the requirements.
2. Select methods for presenting information.
3. Create document, report, or other formats that contain information produced by the system. The output form of an information system should accomplish one or more of the following objectives.

•Convey information about past activities, current status or projections of the Future.
•Signal important events, opportunities, problems, or warnings.
•Trigger an action.
•Confirm an action.

5.3 Data Flow Diagram

A Data Flow diagram (DFD) is a graphical representation of the flow of data through an information system, modelling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

5.3.1 Level 0 Data Flow Diagram

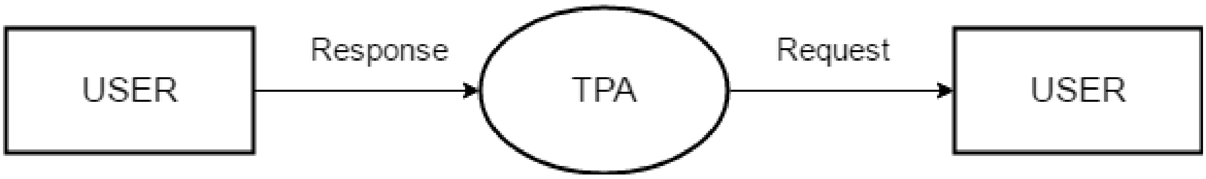


Figure 5.2 Level 0 data flow diagram

The step by step taken in the phase of input to output processing of data is shown in this Fig. 5.2.

This includes user input and corresponding output the user receives.

5.3.2 Level 1 Data Flow Diagram-Admin

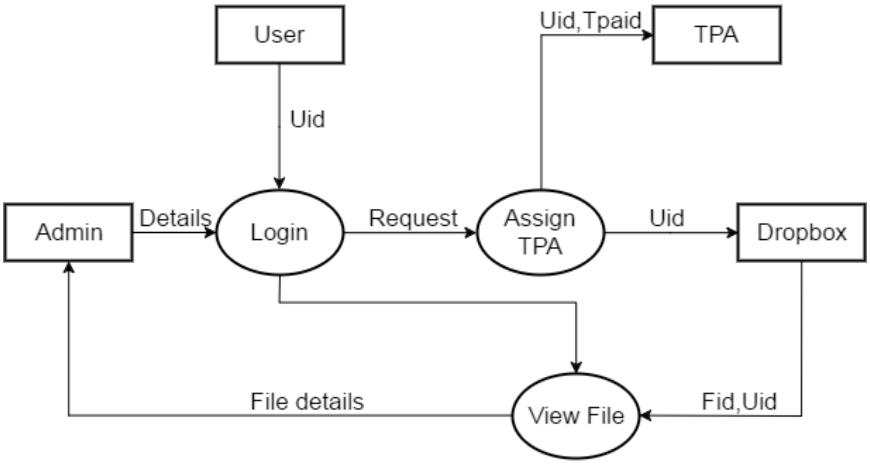


Figure 5.3 Level 1 Data Flow - Admin

The Admin logins with valid username and password to the application. The admin has access to the files verified and other details pertaining to the file. The admin overall has all the options like viewing of the files verified by the TPA and the changes made to the existing files. The data flow diagram is as shown in Fig. 5.3.

5.3.3 Level 1 Data Flow Diagram-Owner

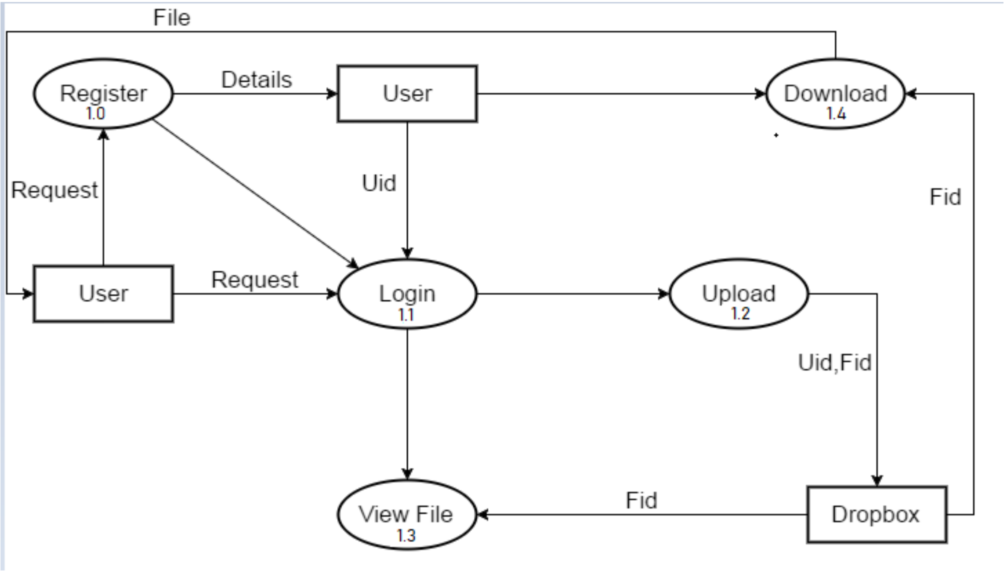


Figure 5.4 Level 1 Data Flow – Owner

Above figure shows level 1 data flow diagram of the owner. A owner is a person who can access resources from cloud. The owner would first register to the interface to get the services with valid user name an password .In order to correctly audit the integrity of the entire data the public verifier needs to choose the appropriate public key for. There will be a third party auditor who performs integrity checking of the data before providing it to the owner or the user. This is done by first splitting the data into blocks and then performing the integrity check. The owner has the option of downloading the verified file.

5.3.4 Level 2 Data Flow Diagram-TPA

The TPA registers to the application with a valid username and password. TPA logins to the application and verifies the integrity of data. TPA views all the list of files uploaded by the owner without the key. Has the privilege of encrypting the data and save it on cloud. TPA also view data which is uploaded by various owner. Level 3 Data Flow is as

Figure 5.4 Level 2 Data Flow – Owner shown in Fig. 5.5.

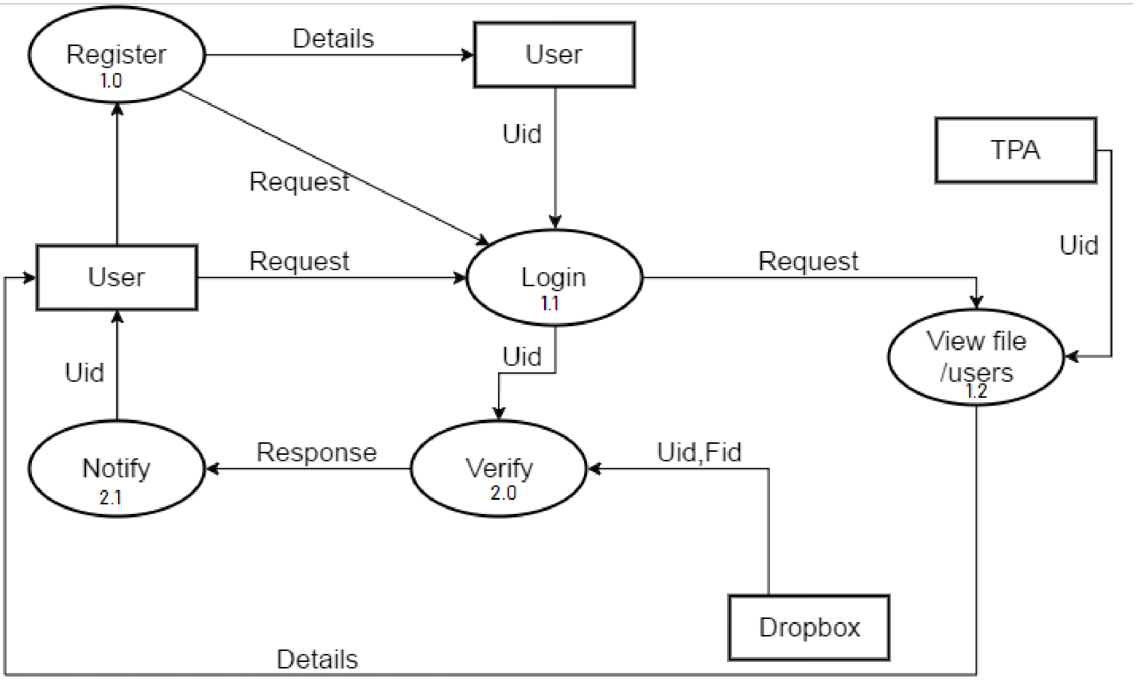


Figure 5.5 Level 2 Data Flow – TPA

5.4 Use Case Diagram of the System

A use case captures the interaction that occurs between the developers and users of information and the system itself .The use case diagram shows a number of external actors and their connections to the use cases that the system provides. A use case is a description of a functionality that the system provides. The use case diagram presents an outside view of the system. The use-case model consists of use-case diagrams consists of the following:

1. The use-case diagrams illustrate the actors, the use cases and their relationships.
2. Use cases also require a textual description (use case specification), as the visual diagrams can't contain all of the information that is necessary.
3. The customers, the end-users, the domain experts, and the developers all have an input into the development of the use-case model. The step by step taken in the phase of input to output processing of data is shown in following figure 5.6. This includes user input and corresponding output the user receives. The Admin logins with valid username and password to the application. The admin has access to the files verified and other details pertaining to the file.

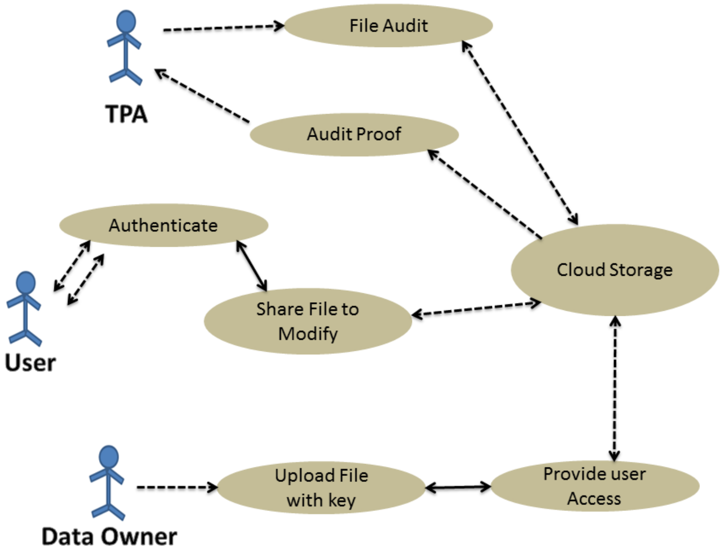


Figure 5.6 Use Case of the system

5.5 Sequence Diagram

A sequence diagram models a dynamic view of the interaction between model elements at run time. Sequence diagrams are commonly used as explanatory models for use case scenario. The figure 5.7 depicts the sequence diagram of the proposed system which is structured representation of behaviour as a series of sequential steps over time to achieve a result.

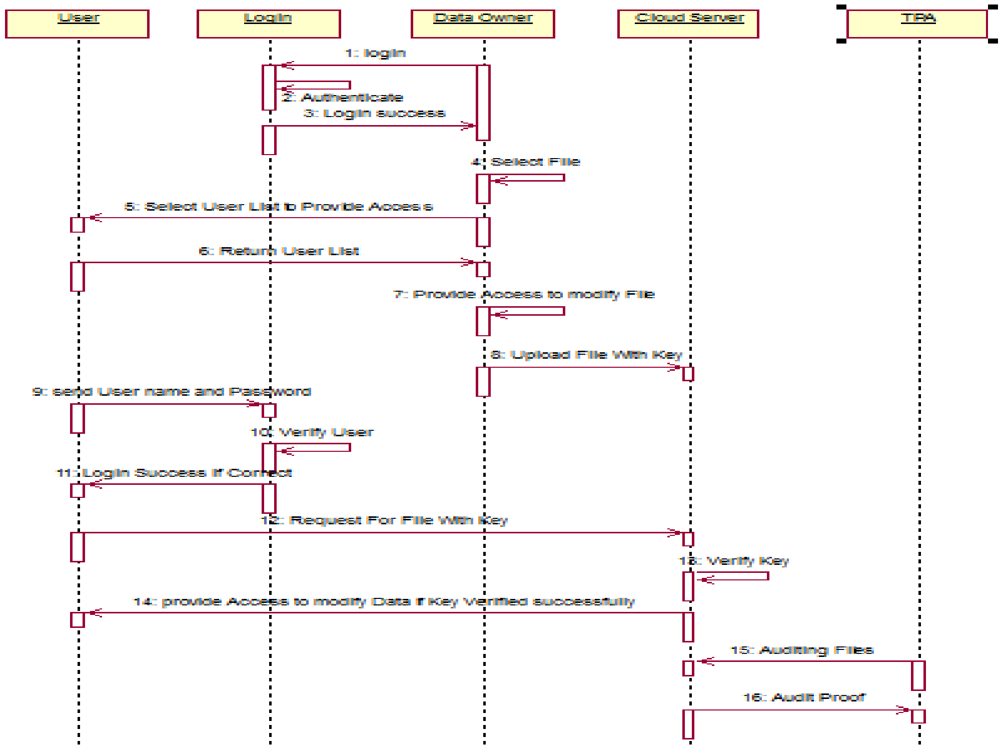

User
Login
Data Owner
Cloud Server
TPA
1: login
2: Authenticate
3: Login success
4: Select File
5: Select User List to Provide Access
6: Return User List
7: Provide Access to modify File
8: Upload File With Key
9: send User name and Password
10: Verify User
11: Login Success If Correct
12: Request For File With Key
13: Verify Key
14: provide Access to modify Data if Key Verified successfully
15: Auditing Files
16: Audit Proof

Figure 5.7 Sequence Diagram

5.6 Activity Diagram

A owner is a person who can access resources from the cloud. The owner would first register to the interface to get the services with the valid username and password. The figure 5.8 depicts the activity diagram of the proposed system which shows owner, user and admin. In order to correctly audit the integrity of the entire data, a public verifier needs to choose the appropriate public key for each block. Then they can request for the file to the cloud service admin. There will be a third party auditor who performs the integrity checking of the data before providing it to the owner or the users. This is done by 1st splitting the data into blocks and then performing integrity check. The owner has the option of downloading the verified file and also uploads new files. The TPA registers to the application with a valid username and password. TPA logins to the application and verifies the integrity of data.

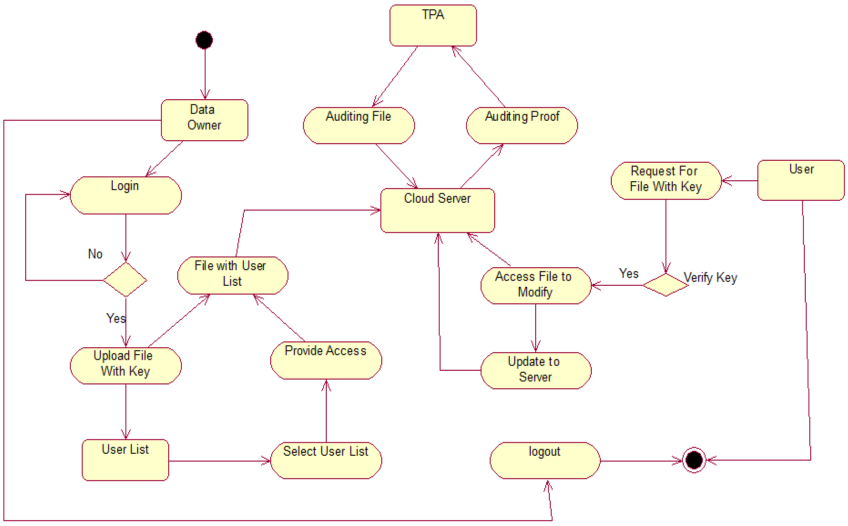


Figure 5.8 Activity Diagram

Chapter 6

Results And Discussion

6.1 Implementation

JSP – FRONT END

Java Server Pages (**JSP**) is a Java technology that allows software developers to dynamically generate HTML, XML or other types of documents in response to a Web client request. The technology allows Java code and certain pre-defined actions to be embedded into static content. The JSP syntax adds additional XML-like tags, called JSP actions, to be used to invoke built-in functionality. Additionally, the technology allows for the creation of JSP tag libraries that act as extensions to the standard HTML or XML tags. Tag libraries provide a platform independent way of extending the capabilities of a Web server. JSPs are compiled into Java Servlets by a JSP compiler. A JSP compiler may generate a servlet in Java code that is then compiled by the Java compiler, or it may generate byte code for the servlet directly. JSPs can also be interpreted on-the-fly reducing the time taken to reload changes Java Server Pages (JSP) technology provides a simplified, fast way to create dynamic web content. JSP technology enables rapid development of web-based applications that are server- and platform-independent.

SERVLETS – FRONT END

Servlets are Java technology's answer to CGI programming. They are programs that run on a Web server and build Web pages. Building Web pages on the fly is useful (and commonly done) for a number of reasons:

- The Web page is based on data submitted by the user. For example the results pages from search engines are generated this way, and programs that process orders for e-commerce sites do this as well.
- The data changes frequently. For example, a weather-report or news headlines page might build the page dynamically, perhaps returning a previously built page if it is still up to date.
- The Web page uses information from corporate databases or other such sources. For example, you would use this for making a Web page at an on-line store that lists current prices and number of items in stock.

MYSQL – BACK END

The MySQL Reference Manual covers most areas of MySQL use. This manual is for both MySQL Community Server and MySQL Enterprise Server. If you cannot find the answer(s) from the manual, you can get support by purchasing MySQL Enterprise, which provides comprehensive support and services. MySQL Enterprise also provides a comprehensive knowledge base library that includes hundreds of technical articles resolving difficult problems on popular database topics such as performance, replication, and migration.

MySQL AB develops and supports a family of high-performance, affordable database products. The company's flagship offering is 'MySQL Enterprise', a comprehensive set of production-tested software, proactive monitoring tools, and premium support services. MySQL is the world's most popular open source database software. Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, business-critical systems and packaged software -- including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube and Booking.com. With headquarters in the United States and Sweden -- and operations around the world -- MySQL AB supports both open source values and corporate customers' needs.

6.2 Result and Analysis

Results

Case 1: Ideal working condition

On successful user registration in a group, the user is authenticated by the group manager. The user can upload files using his key. The file uploaded over cloud is encrypted. The group users can view files pertaining to their group only. To download a particular file a user will have to specify his key. The public verifier audits the files by comparing the hash values of the uploaded and downloaded file. The TPA should be able to detect whether file has been deleted. This lets users know they have the genuine file or not.

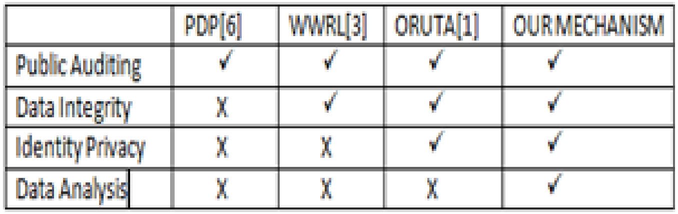
Case 2: Malicious activity by users

Consider five static groups each including ten group users. A user can knowingly modify the files belonging

to his group. This can affect the data integrity and confidential data may be compromised. There could be more than one malicious user in group. Since our system provides identity privacy, the identity of user involved in malicious activity is not revealed.

Case 3:Risk management by TPA

It is important to maintain any unauthorized modification of data in groups. The download details of a file including the timestamp and group id is recorded by the public verifier. The TPA can use this information to analyze the activities taking place in any group. TPA can then find out the malicious users belong to which group. This will give an overall view of integrity maintained in the group.



	PDP[6]	WWRL[3]	ORUTA[1]	OUR MECHANISM
Public Auditing	✓	✓	✓	✓
Data Integrity	X	✓	✓	✓
Identity Privacy	X	X	✓	✓
Data Analysis	X	X	X	✓

Table 4 Comparison of our system with existing system

Performance analysis

The server’s computation cost spent in user revocation to evaluate the system performance of user revocation. Especially, the lazy-revocation method greatly reduces the cost of revocation, because it aggregates multiple cipher text/key update operations, which amortizes the computations over time.

Experimental Results:

Performance of Signature Generation:

The generation time of a ring signature on a block is determined by the number of users in the group and the number of elements in each block. As illustrated in Fig.6.1, when k is fixed, the generation time of a ring signature is linearly increasing with the size of the group; when d is fixed, the generation time of a ring signature is linearly increasing with the number of elements in each block. Specifically, when d ¼ 10 and k ¼ 100, a user in the group requires about 37 milliseconds to compute a ring signature on a block in shared data .

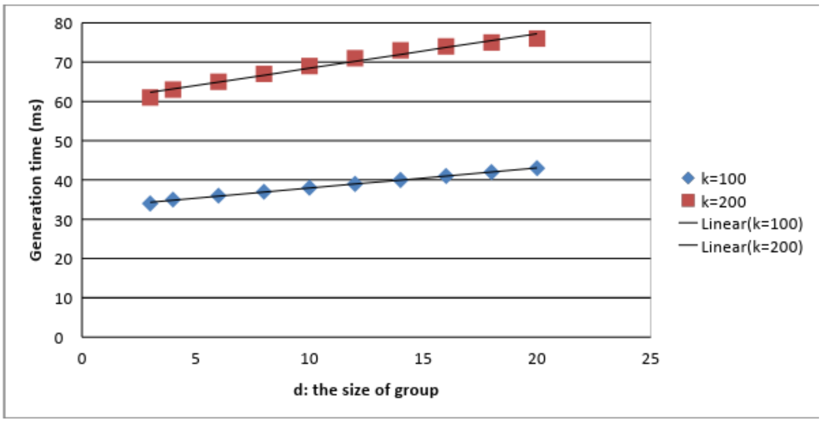


Fig.6.1 Performance of Signature Generation.

Performance of Communication Cost:

The communication cost of an auditing task under different parameters is presented in Fig 6.2. Compared to the size of entire shared data, the communication cost that a public verifier consumes in an auditing task is very small. When maintaining a higher detection probability, a TPA needs to consume more computation and communication overhead to finish the auditing task.

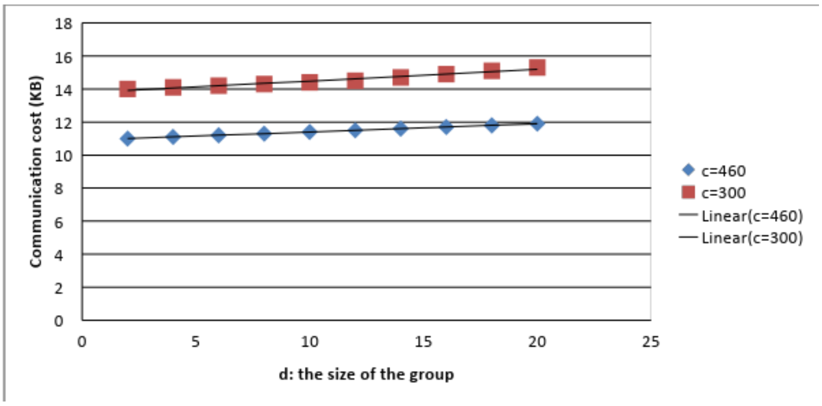


Fig 6.2 Performance of Communication Cost.

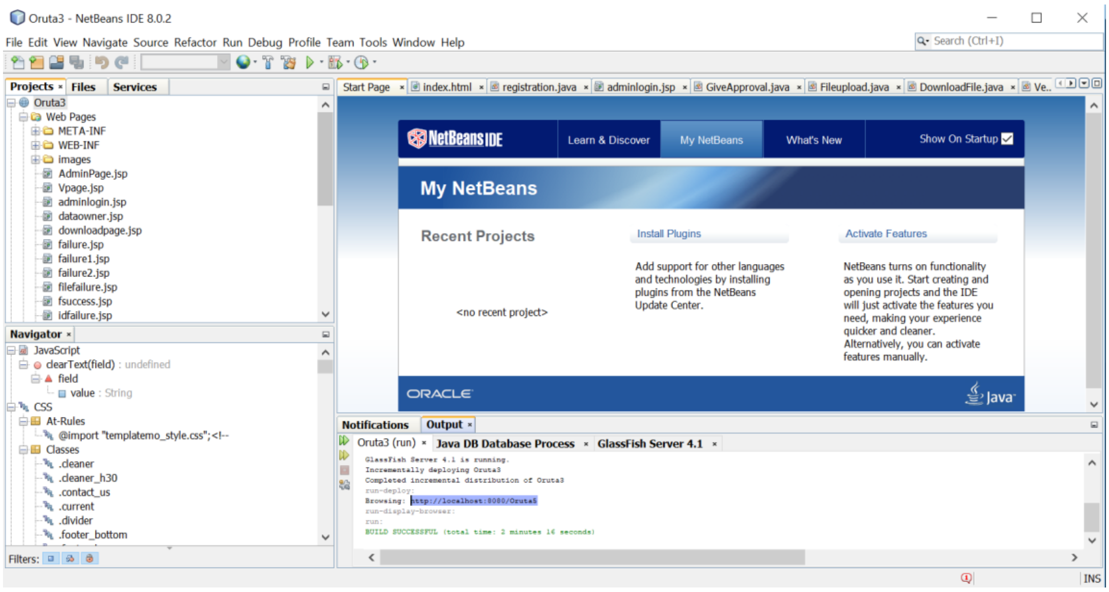
6.3 Cost and Benefit Analysis

Computational Cost and Benefit Analysis

During an auditing task, the public verifier first generates some random values to construct an auditing challenge, which only introduces a small cost in computation.

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

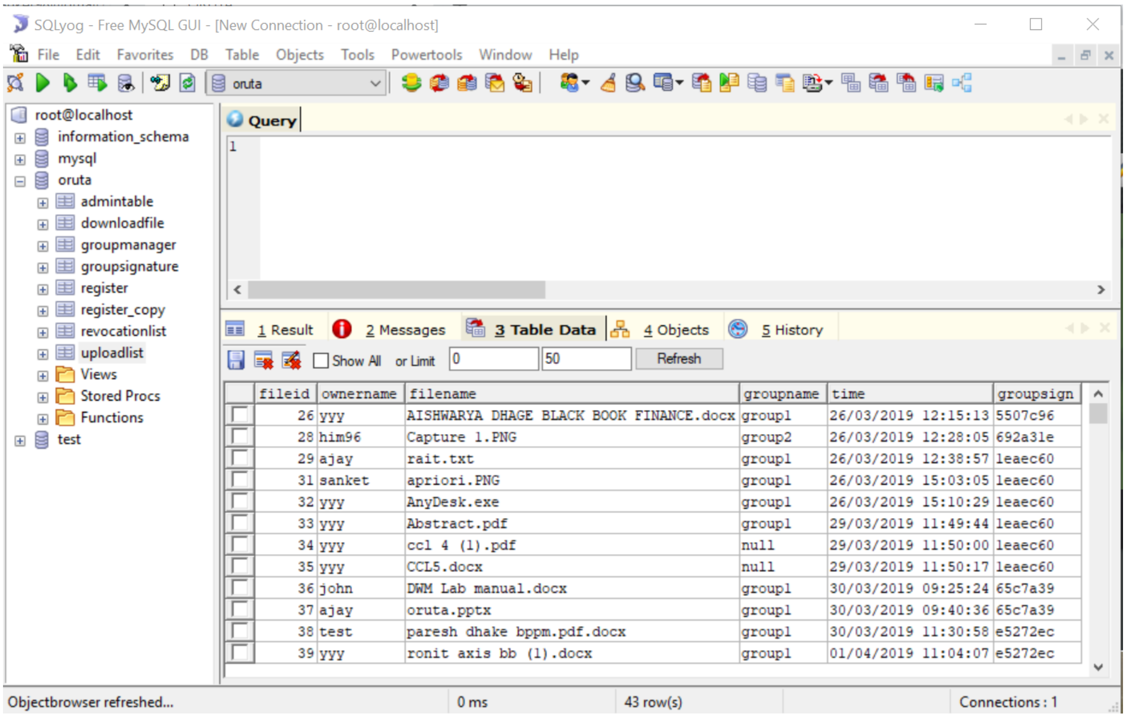
Screenshots



Add support for other languages and technologies by installing plugins from the NetBeans Update Center.

NetBeans turns on functionality as you use it. Start creating and opening projects and the IDE will just activate the features you need, making your experience quicker and cleaner. Alternatively, you can activate features manually.

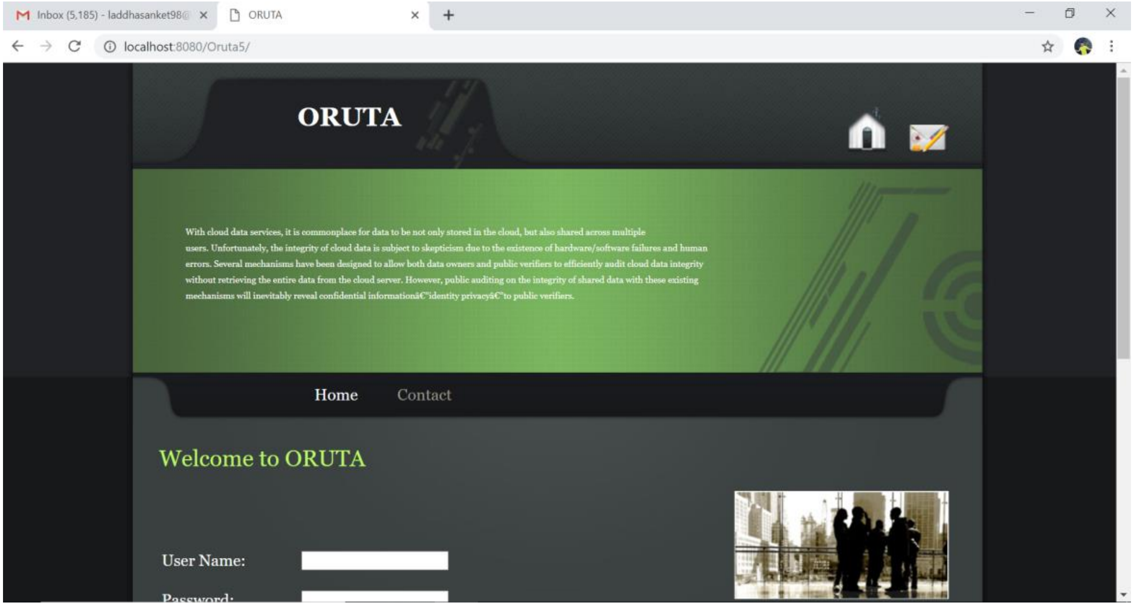
```
GlassFish Server 4.1 is running.
Incrementally deploying Oruta3
Completed incremental distribution of Oruta3
run-deploy:
Browsing: http://localhost:8080/Oruta3
run-display-browser:
run:
BUILD SUCCESSFUL (total time: 2 minutes 16 seconds)
```



0 | 50

	fileid	ownername	filename	groupname	time	groupsign
□	26	yyy	AISHWARYA DHAGE BLACK BOOK FINANCE.docx	group1	26/03/2019 12:15:13	5507c96
□	28	him96	Capture 1.PNG	group2	26/03/2019 12:28:05	692a31e
□	29	ajay	rait.txt	group1	26/03/2019 12:38:57	1eaec60
□	31	sanket	apriori.PNG	group1	26/03/2019 15:03:05	1eaec60
□	32	yyy	AnyDesk.exe	group1	26/03/2019 15:10:29	1eaec60
□	33	yyy	Abstract.pdf	group1	29/03/2019 11:49:44	1eaec60
□	34	yyy	ccl 4 (1).pdf	null	29/03/2019 11:50:00	1eaec60
□	35	yyy	CCL5.docx	null	29/03/2019 11:50:17	1eaec60
□	36	john	DWM Lab manual.docx	group1	30/03/2019 09:25:24	65c7a39
□	37	ajay	oruta.pptx	group1	30/03/2019 09:40:36	65c7a39
□	38	test	paresh dhake bppm.pdf.docx	group1	30/03/2019 11:30:58	e5272ec
□	39	yyy	ronit axis bb (1).docx	group1	01/04/2019 11:04:07	e5272ec

Objectbrowser refreshed... | 0 ms | 43 row(s) | Connections : 1



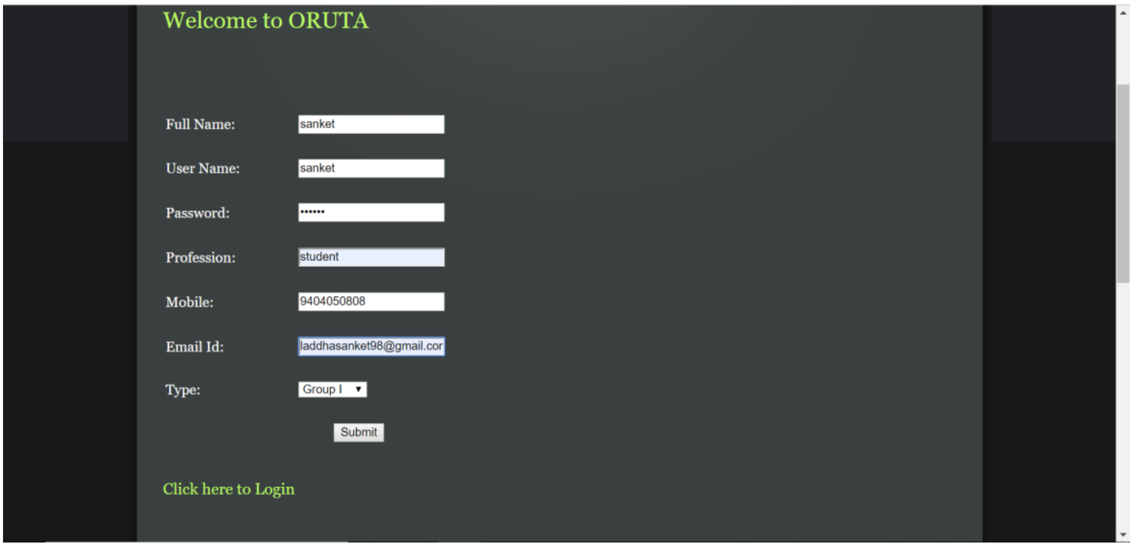
ORUTA

With cloud data services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. Unfortunately, the integrity of cloud data is subject to skepticism due to the existence of hardware/software failures and human errors. Several mechanisms have been designed to allow both data owners and public verifiers to efficiently audit cloud data integrity without retrieving the entire data from the cloud server. However, public auditing on the integrity of shared data with these existing mechanisms will inevitably reveal confidential informationâ€"identity privacyâ€"to public verifiers.

Welcome to ORUTA

User Name:

Password:



Welcome to ORUTA

Full Name: sanket

User Name: sanket

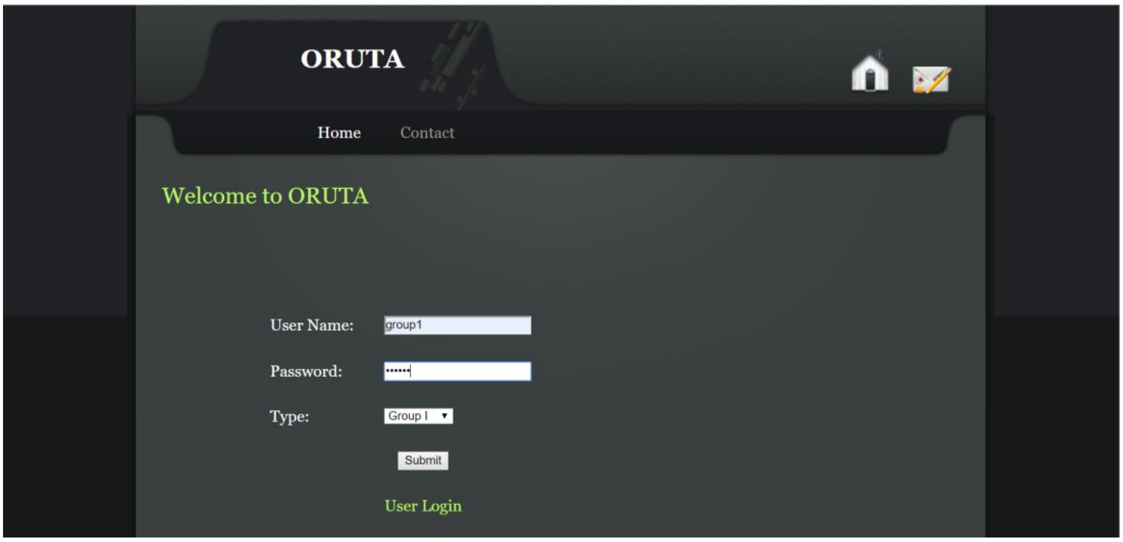
Password: ••••••

Profession: student

Mobile: 9404050808

Email Id: laddhasanket98@gmail.cor

Type: Group I



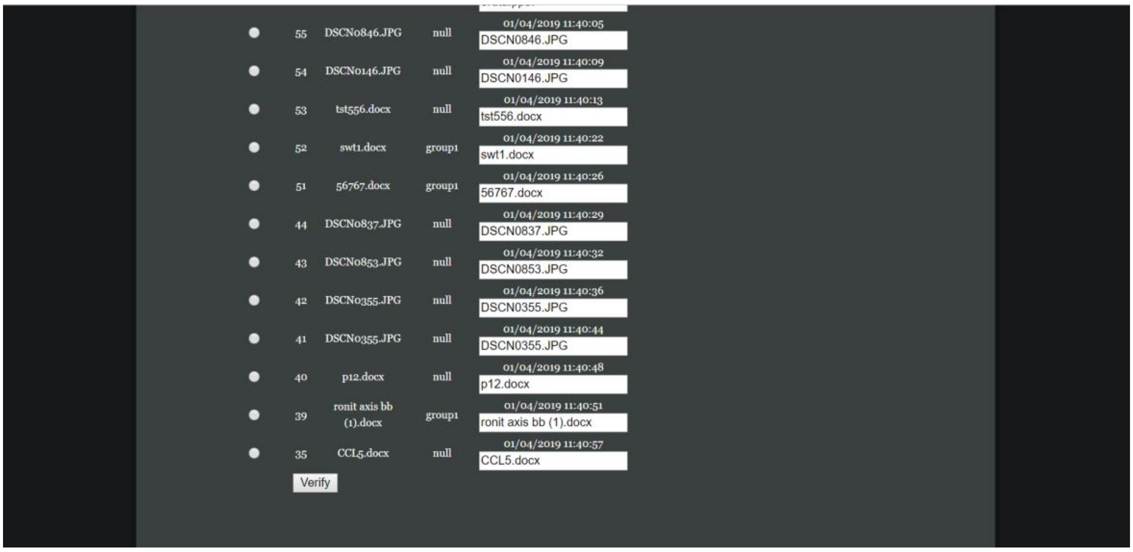
ORUTA

Welcome to ORUTA

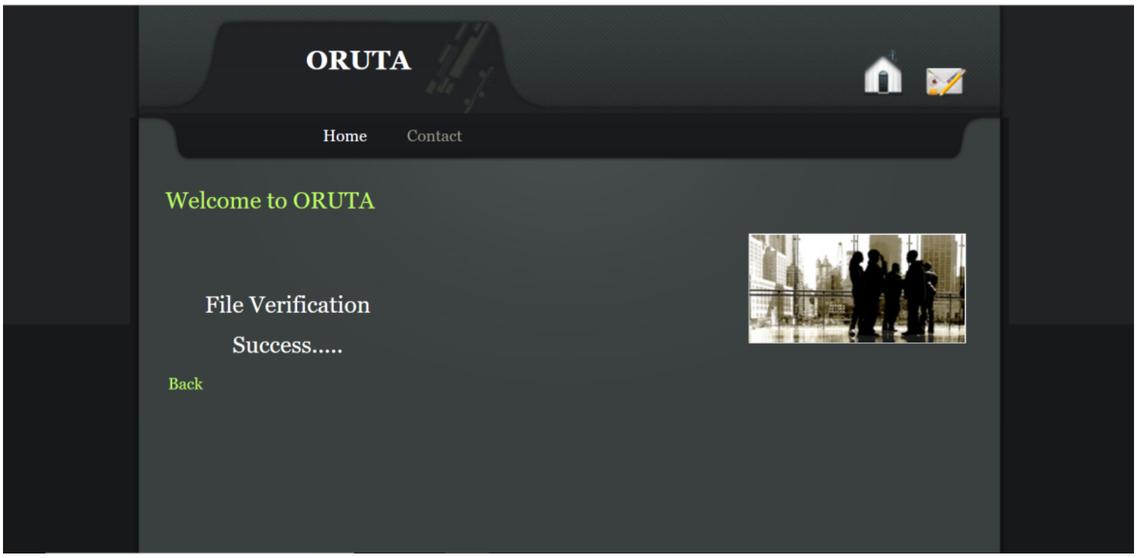
User Name: group1

Password: ••••••

Type: Group I



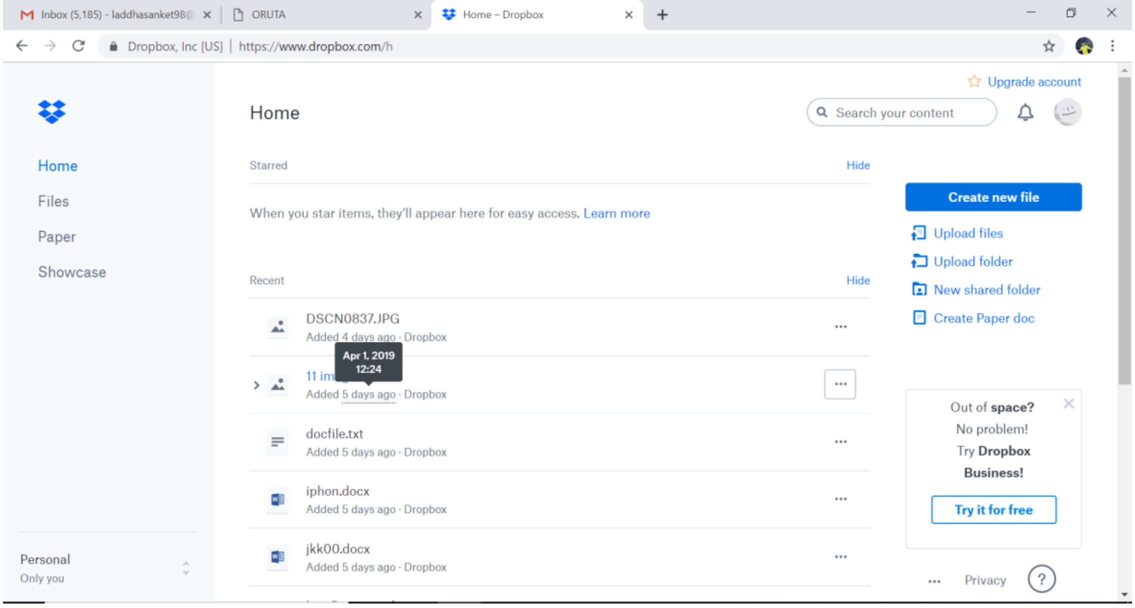
○	55	DSCN0846.JPG	null	01/04/2019 11:40:05 DSCN0846.JPG
○	54	DSCN0146.JPG	null	01/04/2019 11:40:09 DSCN0146.JPG
○	53	tst556.docx	null	01/04/2019 11:40:13 tst556.docx
○	52	swt1.docx	group1	01/04/2019 11:40:22 swt1.docx
○	51	56767.docx	group1	01/04/2019 11:40:26 56767.docx
○	44	DSCN0837.JPG	null	01/04/2019 11:40:29 DSCN0837.JPG
○	43	DSCN0853.JPG	null	01/04/2019 11:40:32 DSCN0853.JPG
○	42	DSCN0355.JPG	null	01/04/2019 11:40:36 DSCN0355.JPG
○	41	DSCN0355.JPG	null	01/04/2019 11:40:44 DSCN0355.JPG
○	40	p12.docx	null	01/04/2019 11:40:48 p12.docx
○	39	ronit axis bb (1).docx	group1	01/04/2019 11:40:51 ronit axis bb (1).docx
○	35	CCL5.docx	null	01/04/2019 11:40:57 CCL5.docx



Welcome to ORUTA

File Verification

Success.....



Home

Starred — Hide

When you star items, they'll appear here for easy access. Learn more

Recent — Hide

DSCN0837.JPG
Added 4 days ago · Dropbox

Apr 1, 2019 12:24

11 im
Added 5 days ago · Dropbox

docfile.txt
Added 5 days ago · Dropbox

iphon.docx
Added 5 days ago · Dropbox

jkk00.docx
Added 5 days ago · Dropbox

Out of space? No problem! Try Dropbox Business!

Chapter 7

Conclusion and Future Works

7.1 Conclusion

First the privacy-preserving public auditing mechanism for shared data in the cloud is proposed. The TPA is able to efficiently audit the integrity of shared data, yet cannot distinguish who is the signer on each block, which can preserve identity privacy for users. The identity of the signer on each block in shared data is kept private from public verifiers, who are able to efficiently verify shared data integrity without retrieving the entire file. The mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one by one. The experimental results demonstrate the effectiveness and efficiency of our mechanism when auditing shared data integrity. We propose a group-oriented ring signature scheme. In this scheme, nobody besides the designated members can verify the validity of the ring signature. We utilize ring signatures to construct homomorphic authenticators so that a public verifier is able to audit shared data integrity without retrieving the entire data, yet it cannot distinguish who is the signer on each block. To improve the efficiency of verifying multiple auditing tasks, we further extend our mechanism to support batch auditing.

7.2 Future Work

To improve the efficiency of verifying multiple auditing tasks, we further extend our mechanism to support batch auditing. To the best of our knowledge, designing an efficient public auditing mechanism with the capabilities of preserving identity privacy and supporting traceability is still open. Another problem for our future work is how to prove data freshness (prove the cloud possesses the latest version of shared data) while still preserving identity privacy.

References

[1] B. Wang, B. Li, and H. Li, Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud, Proc. IEEE Fifth Intl Conf. Cloud Computing, pp. 295-302, 2012.

[2] C. Wang, S.S. Chow, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Secure Cloud Storage,” IEEE Trans.

[3] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Storage Security in Cloud Computing”

[4] Chunbo Ma and Jun Ao, “Group-oriented ring signature”,School of Information and Communication, Guilin University of Electronic Technology, Guilin, Guangxi, 541004, P. R. China.

[5] The MD5 Message-Digest Algorithm (RFC1321).
https://tools. ietf.org/html/rfc1321 2014.

[6]The Dropboxapi
https://www.dropbox.com/developers/documentation/http/overview

[7] "What is Cloud Computing?". Amazon Web Services. 2013-03-19. Retrieved 2013-03-20.

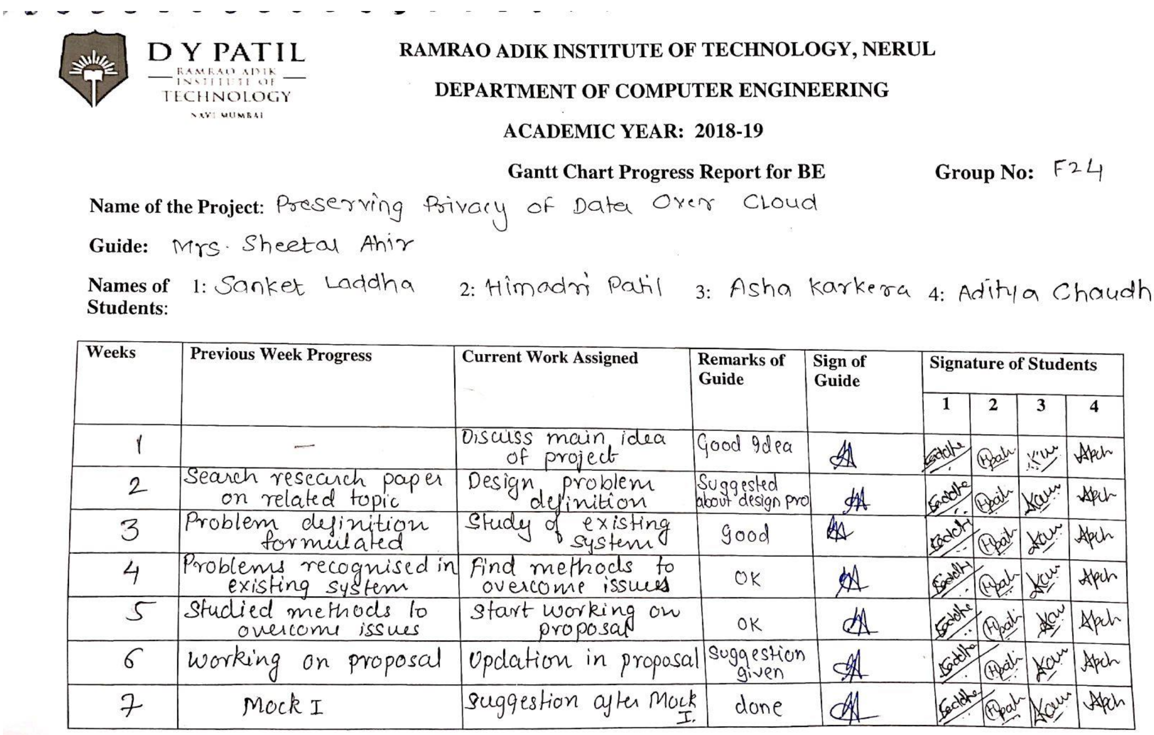
[8] Peter Mell, Timothy Grance, "The NIST Definition of Cloud Computing".

[9] Swapnali More, Sangita Chaudhari, "Third Party Public Auditing scheme for Cloud Storage", 7th International Conference on Communication, Computing and Virtualization 2016.

Appendices

Appendix A

Weekly Progress Report



D Y PATIL
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

RAMRAO ADIK INSTITUTE OF TECHNOLOGY, NERUL

DEPARTMENT OF COMPUTER ENGINEERING

ACADEMIC YEAR: 2018-19

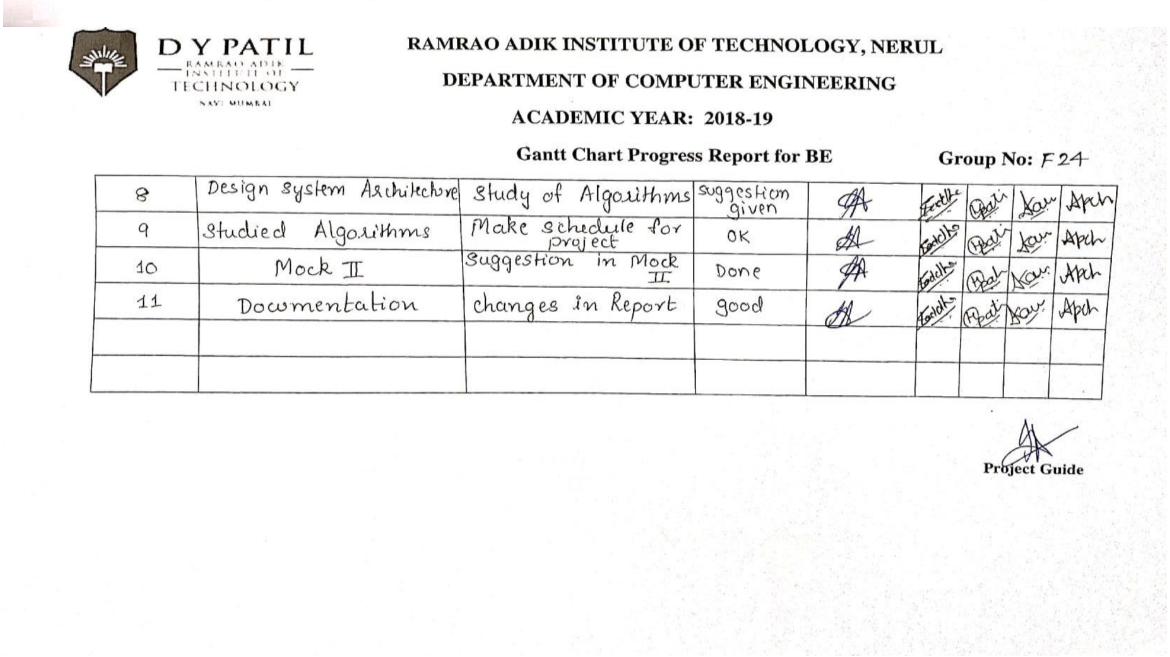
Gantt Chart Progress Report for BE Group No: F24

Name of the Project: Preserving Privacy of Data Over Cloud

Guide: Mrs. Sheetal Ahir

Names of Students: 1: Sanket Laddha 2: Himadri Patil 3: Asha Karkera 4: Aditya Chaudhari

Weeks	Previous Week Progress	Current Work Assigned	Remarks of Guide	Sign of Guide	Signature of Students			
					1	2	3	4
1	—	Discuss main idea of project	Good Idea					
2	Search research paper on related topic	Design problem definition	Suggested about design pro					
3	Problem definition formulated	Study of existing system	good					
4	Problems recognised in existing system	Find methods to overcome issues	OK					
5	Studied methods to overcome issues	start working on proposal	OK					
6	working on proposal	Updation in proposal	suggestion given					
7	Mock I	suggestion after Mock I.	done					



D Y PATIL
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

RAMRAO ADIK INSTITUTE OF TECHNOLOGY, NERUL

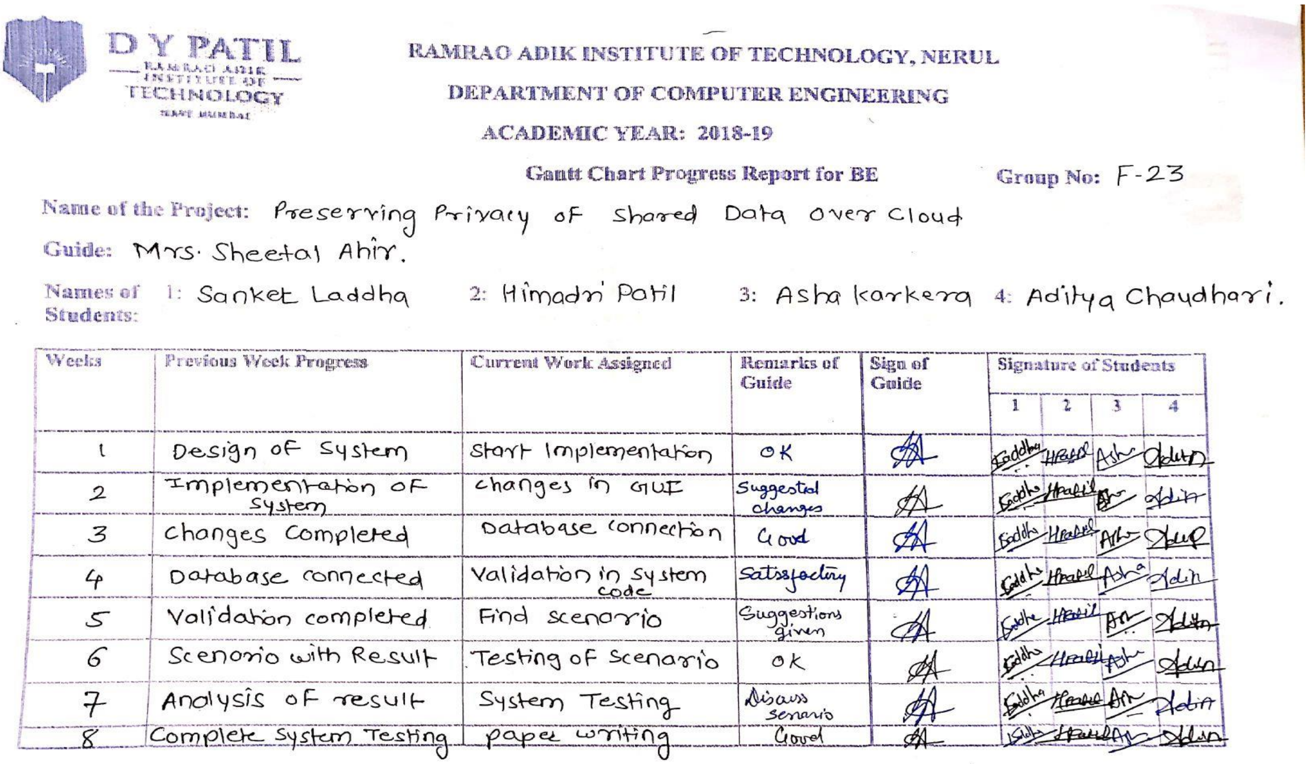
DEPARTMENT OF COMPUTER ENGINEERING

ACADEMIC YEAR: 2018-19

Gantt Chart Progress Report for BE Group No: F24

8	Design system Architecture	Study of Algorithms	suggestion given					
9	Studied Algorithms	Make schedule for project	OK					
10	Mock II	Suggestion in Mock II	Done					
11	Documentation	changes in Report	good					

Project Guide



D Y PATIL
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

RAMRAO ADIK INSTITUTE OF TECHNOLOGY, NERUL
DEPARTMENT OF COMPUTER ENGINEERING
ACADEMIC YEAR: 2018-19

Gantt Chart Progress Report for BE

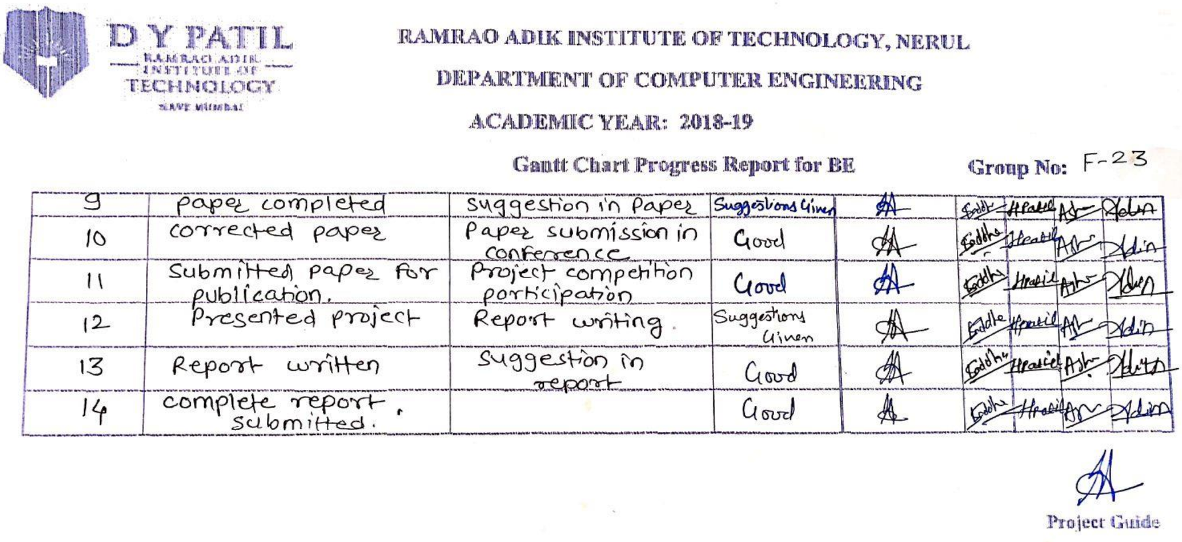
Group No: F-23

Name of the Project: Preserving Privacy of Shared Data over Cloud

Guide: Mrs. Sheetal Ahir.

Names of Students: 1: Sanket Laddha 2: Himadri Patil 3: Asha karkera 4: Aditya Chaudhari.

Weeks	Previous Week Progress	Current Work Assigned	Remarks of Guide	Sign of Guide	Signature of Students			
					1	2	3	4
1	Design of System	start Implementation	OK		Laddha	HPatil	Asha	Aditya
2	Implementation of System	changes in GUI	Suggested changes		Laddha	HPatil	Asha	Aditya
3	changes Completed	Database connection	Good		Laddha	HPatil	Asha	Aditya
4	Database connected	Validation in system code	Satisfactory		Laddha	HPatil	Asha	Aditya
5	Validation completed	Find scenario	Suggestions given		Laddha	HPatil	Asha	Aditya
6	Scenario with Result	Testing of Scenario	OK		Laddha	HPatil	Asha	Aditya
7	Analysis of result	System Testing	Discuss scenario		Laddha	HPatil	Asha	Aditya
8	Complete System Testing	paper writing	Good		Laddha	HPatil	Asha	Aditya



D Y PATIL
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

RAMRAO ADIK INSTITUTE OF TECHNOLOGY, NERUL
DEPARTMENT OF COMPUTER ENGINEERING
ACADEMIC YEAR: 2018-19

Gantt Chart Progress Report for BE

Group No: F-23

9	paper completed	suggestion in paper	Suggestions Given		Laddha	HPatil	Asha	Aditya
10	corrected paper	Paper submission in conference	Good		Laddha	HPatil	Asha	Aditya
11	Submitted paper for publication.	Project competition participation	Good		Laddha	HPatil	Asha	Aditya
12	Presented project	Report writing.	Suggestions Given		Laddha	HPatil	Asha	Aditya
13	Report written	Suggestion in report	Good		Laddha	HPatil	Asha	Aditya
14	complete report, submitted.		Good		Laddha	HPatil	Asha	Aditya

Project Guide

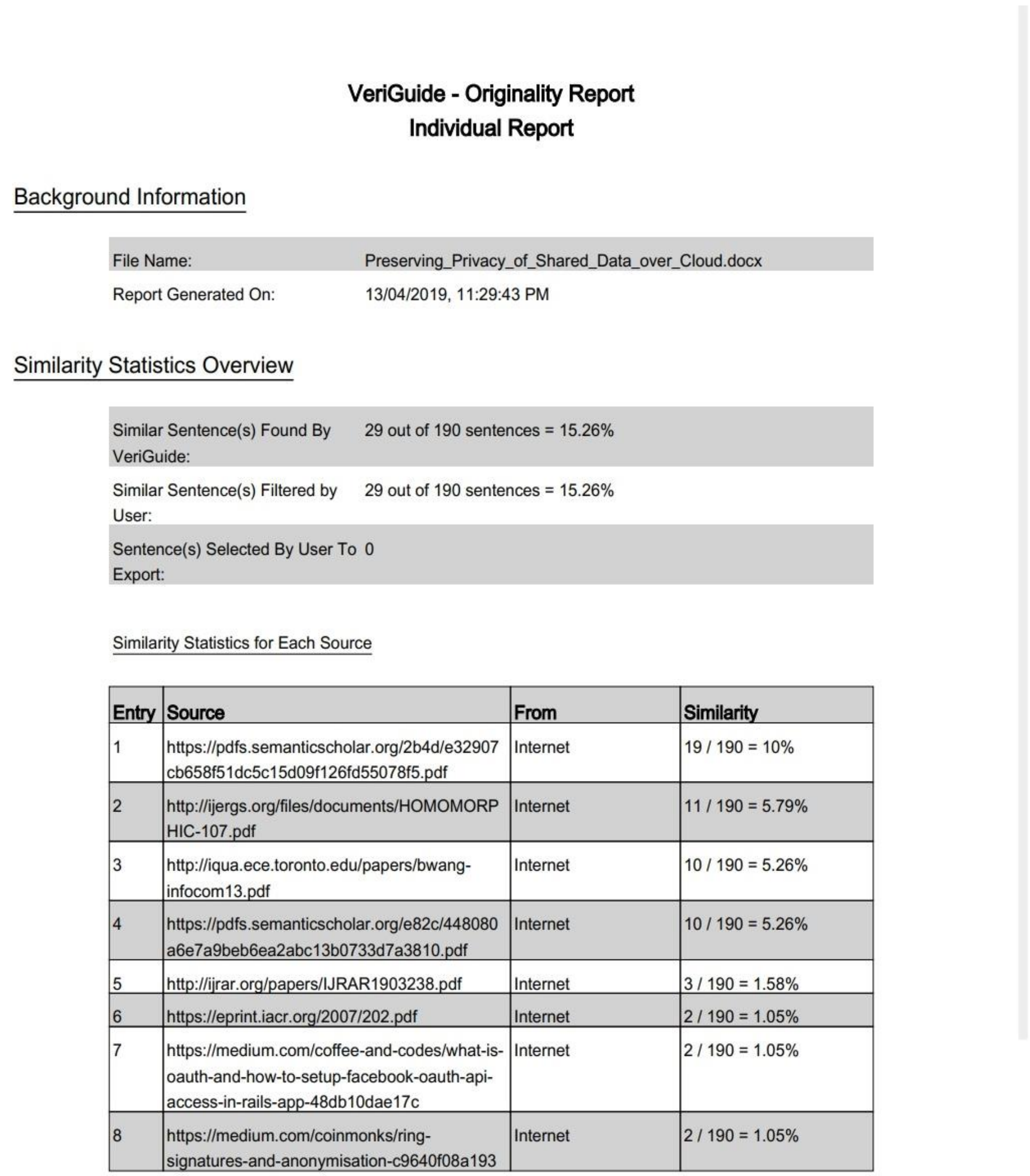
Appendix B

Paper Publication

Paper is submitted to the conference IEEE BOMBAY SECTION SIGNATURE CONFERENCE (IBSSC-2019).

https://easychair.org/conferences/?conf=ibssc2019

submitted by Sanket Laddha , Himadri Patil , Asha Karkera , Aditya Chaudhari and Mrs.Sheetal Ahir .



VeriGuide - Originality Report
Individual Report

Background Information

File Name:	Preserving_Privacy_of_Shared_Data_over_Cloud.docx
Report Generated On:	13/04/2019, 11:29:43 PM

Similarity Statistics Overview

Similar Sentence(s) Found By VeriGuide:	29 out of 190 sentences = 15.26%
Similar Sentence(s) Filtered by User:	29 out of 190 sentences = 15.26%
Sentence(s) Selected By User To Export:	0

Similarity Statistics for Each Source

Entry	Source	From	Similarity
1	https://pdfs.semanticscholar.org/2b4d/e32907cb658f51dc5c15d09f126fd55078f5.pdf	Internet	19 / 190 = 10%
2	http://ijergs.org/files/documents/HOMOMORPHIC-107.pdf	Internet	11 / 190 = 5.79%
3	http://iqua.ece.toronto.edu/papers/bwang-infocom13.pdf	Internet	10 / 190 = 5.26%
4	https://pdfs.semanticscholar.org/e82c/448080a6e7a9beb6ea2abc13b0733d7a3810.pdf	Internet	10 / 190 = 5.26%
5	http://ijrar.org/papers/IJRAR1903238.pdf	Internet	3 / 190 = 1.58%
6	https://eprint.iacr.org/2007/202.pdf	Internet	2 / 190 = 1.05%
7	https://medium.com/coffee-and-codes/what-is-oauth-and-how-to-setup-facebook-oauth-api-access-in-rails-app-48db10dae17c	Internet	2 / 190 = 1.05%
8	https://medium.com/coinmonks/ring-signatures-and-anonymisation-c9640f08a193	Internet	2 / 190 = 1.05%

Preserving Privacy of Shared Data over Cloud

Sanket Laddha
Department of Computer Engineering
Ramrao Adik Institute of Technology
Navi Mumbai, India
laddhasanket98@gmail.com

Himadri Patil
Department of Computer Engineering
Ramrao Adik Institute of Technology
Navi Mumbai, India
himadripatil27@gmail.com

Asha Karkera
Department of Computer Engineering
Ramrao Adik Institute of Technology
Navi Mumbai, India
ashauk2010@gmail.com

Aditya Chaudhari
Department of Computer Engineering
Ramrao Adik Institute of Technology
Navi Mumbai, India
adityac1305@gmail.com

Mrs Sheetal Ahir
Department of Computer Engineering
Ramrao Adik Institute of Technology
Navi Mumbai, India
sheetal.ahir@rait.ac.in

Abstract**--** ***Cloud Data Services are not just used to store information on cloud at one spot for single client, yet in addition to share among numerous clients. The honesty of cloud information is suspicious or dicey because of the disappointments in equipment and programming and human mistakes. Information proprietors and open verifiers have been empowered to proficiently review cloud information trustworthiness utilizing different systems. This cripples the recouping of the whole information from cloud server. Open examining on honesty of shared information, with this current component will thusly uncover confined data character protection to open verifiers. In this task, we propose a framework that underpins open inspecting on shared information put away over the cloud. Explicitly we exploit the ring marks to figure check of metadata expected to review the rightness of shared information. With our system the personality of client in shared information is kept private from open verifiers .Moreover, this component can play out different inspecting assignments at the same time as opposed to checking them one by one. Our outcomes demonstrates the adequacy and effectiveness of this instrument while inspecting shared information honesty.***

Keywords:—Auditing, Ring signature, Third Party Auditor

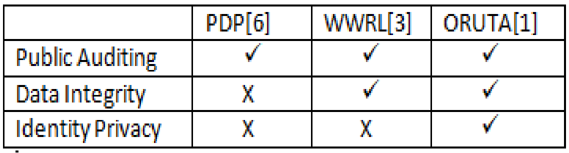
I. Introduction

Applications conveyed as administrations through the web and the framework programming and equipment in the server farms that give those administrations, is the thing that distributed computing alludes to. Because of the sharing idea of assets Cloud Service Providers deal with an endeavor class foundation that offers a versatile and solid condition for clients, at a much lower negligible expense. It is normal for clients to utilize distributed storage administrations to impart information to others in a group, as information sharing turns into a standard component in most distributed storage contributions, including DropBox and Google Docs. The trustworthiness of information in distributed storage, in any case, is liable to skeptism and examination, as information put away in an endowed cloud can without much of a stretch be lost or defiled, because of equipment disappointments and human blunders. The most ideal way of secure the data honesty over cloud is to perform open reviewing by presenting an outsider reviewer (TPA), who offers its examining administration with more dominant calculation and correspondence capacities than normal clients.

Literature Survey

In [1], the identity of the signer on each block in shared data is kept private from a third party auditor (TPA), who is still able to publicly verify the integrity of shared data without retrieving the entire file. In [2], various security challenges were proposed which the user may face in the cloud environment. In [3], a privacy-preserving public auditing system was proposed for data storage security in Cloud Computing utilizing homomorphic authenticator and random masking to guarantee that TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process. In [4], presents a survey on various data retrieval techniques in cloud using multiple keys. detailed review about the data retrieval process, functionalities, methodologies and the evaluation of various data retrieval techniques using multiple keys in cloud was provided. In [5] the authors have suggested a method for ensuring scalable, secure and fine-grained Data Access Control in Cloud Computing. Ateniese et al. [6] are the first to consider public auditability in their defined “provable data possession” (PDP) model for ensuring possession of data files on untrusted storages. It provided data format independence, which is a relevant feature in practical deployments and put no restriction on the number of times the client can challenge the server to prove data possession along with public verifiability.

TABLE I



	PDP[6]	WWRL[3]	ORUTA[1]
Public Auditing	✓	✓	✓
Data Integrity	X	✓	✓
Identity Privacy	X	X	✓

Fig. 1. Comparison of existing mechanisms

II. Security issues in public cloud

In [2] In spite of the fact that distributed computing's points of interest are monstrous, security and protection concerns are the principle deterrents to wide reception. Since cloud specialist organizations (CSPs) are isolated managerial elements, moving to the business open cloud denies clients of direct authority over the frameworks where their information is put away. Regardless of whether CSPs' framework and the executives capacities are substantially more dominant and trustworthy than those of individualized computing gadgets, the cloud stage still faces both interior and outer security and protection dangers, including media disappointments, programming bugs, malware, manager blunders and noxious insiders

Since clients don't approach the cloud's inside operational subtleties, CSPs may likewise deliberately look at clients' information for different reasons without recognition. Likewise, attributable to equipment virtualization, numerous clients would now be able to have the equivalent physical foundation, which runs their unmistakable application occasions all the while. Despite the fact that it builds asset use, this interesting multitenancy highlight likewise exhibits new security and protection vulnerabilities for client connections. Subsequently, we contend that the cloud is characteristically shaky from a client's perspective. It can't be normal that the clients turn control of their information and figuring applications over to the cloud dependent on financial reserve funds and administration adaptability without giving a solid security and protection ensure.

Proposed Scheme

To determine the issue of protection issue on aggregate information, Oruta (One Ring To Rule Them All), a unique security safeguarding open reviewing system has been executed. Ring marks in Oruta is used, so an open verifier can confirm the uprightness of shared information without recovering the whole information while the character of the endorser on each square in shared information is kept undisclosed from the open verifier .The entities in the system are:

A. *Cloud Server:*

- The cloud server serves as the storage module and is responsible for storage and maintenance of user files.
- It also holds user data to validate and authorize users.
- Metadata on the files stored must also be generated and kept by the server.

B. *Group of users:*

- The original user and a number of group users. The original user initially creates a file on the cloud, and shares it with group users.
- Both the original user and group users are members of the group.
- Every member of the group is allowed to access and modify shared data.
- Shared data and its verification metadata (i.e., signatures) are both stored in the cloud server.
- A public verifier, such as a TPA providing expert data auditing services or a data user outside the group intending to utilize shared data, is able to publicly verify the integrity of shared data stored in the cloud server.

C. *Public verifier:*

- When a public verifier wants to verify the authenticity of shared data, it sends an auditing challenge to the cloud server.
- After receiving the auditing challenge, the Cloud server responds to the public verifier with an auditing proof of the possession of shared data.
- Then the public verifier checks the correctness of the entire data by verifying the auditing proof. Essentially, the process of public auditing is a challenge and response protocol between public verifier and cloud server.

D. *DropBox*

- We utilise Dropbox as the storage system on cloud. This is done by using the Dropbox API using the OAuth method, which allows interaction between Dropbox and Application after initial authorization.
- OAuth (Open Authorization) is an open standard for token-based authentication and

authorization on the Internet. OAuth, which is pronounced "oh-auth," allows an end user's account information to be used by third-party services, such as Facebook, without exposing the user's password.

III. System Architecture

As illustrated in Fig. 2, the system model in this paper involves three parties: the cloud server, users and a public verifier. There are two types of users in a group: the original user and group users. The shared data is created by the original user in the cloud, and shares it with group users. Both the original user and group users are members of the group. Every member of the group is able to access and modify shared data. Shared data and its verification metadata (i.e., signatures) are both stored in the cloud server. A public verifier, such as a third-party auditor providing data auditing services, is able to publicly verify the correctness of shared data stored in the cloud server. When a public verifier wishes to check the integrity of shared data, it first sends an auditing challenge to the cloud server. The cloud server responds to the public verifier with an auditing proof of the possession of shared data on receiving the auditing challenge. Then, this public verifier checks the integrity of the entire data by verifying the correctness of the auditing proof. Essentially, the process of public auditing is a challenge-and-response protocol between a public verifier and the cloud server.

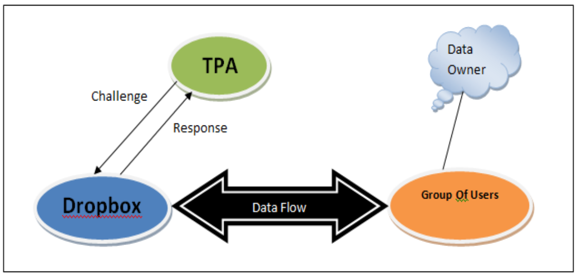


Fig. 2. System model including TPA, DropBox and group of users

A. Ring Signatures

A ring mark is a computerized mark that is made by an individual from a gathering which each have their own keys. It is then not be conceivable to decide the individual in the gathering who has made the mark. The idea of ring marks was first proposed by Rivest et al. in 2001. With ring marks, a verifier is sure that a mark is figured utilizing one of gathering individuals' private keys, however the verifier is unfit to discover which one. All the more solidly, given a ring mark and a gathering of d clients, a verifier can't separate the underwriter's personality with a likelihood more than 1=d. This property can be utilized to save the personality of the endorser from a verifier. The ring signature scheme introduced by Boneh et al. [8] (referred to as BGLS in this paper) is constructed on bilinear maps. We will extend this ring signature scheme to construct our public auditing mechanism.

B. Ring Creation

In a ring signature we define a group of entities who each have their own public/private key pairs of (P1, S1), (P2, S2), …, (Pn, Sn). If we want an entity i to sign a message , they use their own secret key (si), but the public keys of the other group members (m,si,P1…Pn).

IV. RESULT ANALYSIS

A. Case 1: Ideal working condition

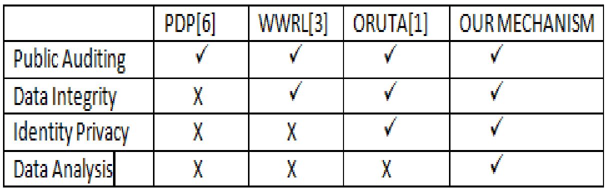
On successful user registration in a group, the user is authenticated by the group manager. The user can upload files using his key. The file uploaded over cloud is encrypted. The group users can view files pertaining to their group only. To download a particular file a user will have to specify his key. The public verifier audits the files by comparing the hash values of the uploaded and downloaded file. The TPA should be able to detect whether file has been deleted. This lets users know they have the genuine file or not.

B. Case 2: Malicious activity by users

Consider five static groups each including ten group users. A user can knowingly modify the files belonging to his group. This can affect the data integrity and confidential data may be compromised. There could be more than one malicious user in group. Since our system provides identity privacy, the identity of user involved in malicious activity is not revealed.

C. Case 3:Risk management by TPA

It is important to maintain any unauthorized modification of data in groups. The download details of a file including the timestamp and group id is recorded by the public verifier. The TPA can use this information to analyze the activities taking place in any group. TPA can then find out the malicious users belong to which group. This will give an overall view of integrity maintained iin the group.



	PDP[6]	WWRL[3]	ORUTA[1]	OUR MECHANISM
Public Auditing	√	√	√	√
Data Integrity	X	√	√	√
Identity Privacy	X	X	√	√
Data Analysis	X	X	X	√

Fig 3. Comparison of our system with existing systems

D. Experimental results

Performance of Signature Generation: The age time of a ring mark on a square is dictated by the quantity of clients in the gathering and the quantity of components in each square. As delineated in Fig.6.1, when k is fixed, the age time of a ring mark is directly expanding with the span of the gathering; when d is fixed, the age time of a ring mark is straightly expanding with the quantity of components in each square. In particular, when d 1⁄4 10 and k 1⁄4 100, a client in the gathering requires around 37 milliseconds to figure a ring mark on a square in shared information

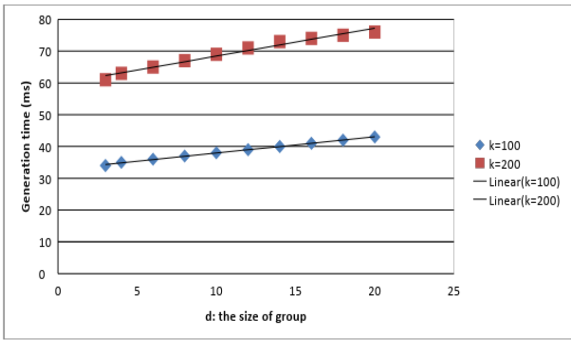


Fig.4 Performance of Signature Generation.

Performance of Communication Cost: The correspondence cost of an examining task under various parameters is introduced in Fig 6.2. Contrasted with the measure of whole shared information, the correspondence cost that an open verifier expends in a reviewing task is little. While keeping up a higher recognition likelihood, a TPA needs to devour more calculation and correspondence overhead to complete the reviewing task.

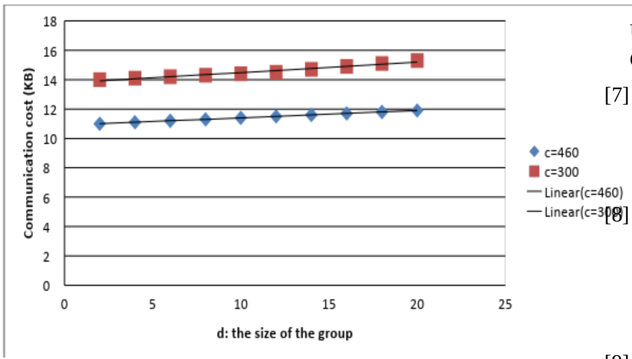


Fig 5 Performance of Communication Cost

V. CONCLUSION

In this paper, privacy-preserving public auditing mechanism for shared data in the cloud is proposed. With our mechanism, the TPA is able to efficiently audit the integrity of shared data, yet cannot distinguish who is the signer on each block, which can preserve identity privacy for users. An interesting problem in our future work is how to efficiently audit the integrity of shared data with dynamic groups while still preserving the identity of the signer on each block from the third party auditor. The identity of the signer on each block in shared data is kept private from public verifiers, who are able to efficiently verify shared data integrity without retrieving the entire file. The mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one by one. The experimental results demonstrate the effectiveness and efficiency of our mechanism when auditing shared data integrity. we propose a group-oriented ring signature scheme. In this scheme, nobody besides the designated members can verify the validity of the ring signature.To improve the efficiency of verifying multiple auditing tasks, we further extend our mechanism to support batch auditing.

REFERENCES

[1] B. Wang, B. Li, and H. Li, “Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud,” Proc. EEE Fifth Int’l Conf. Cloud Computing, pp. 295-302, 2012.

[2] K. Ren, C. Wang, and Q. Wang, “Security Challenges for the Public Cloud,” IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.

[3] C. Wang, Q. Wang, K. Ren, and W. Lou, Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing, Proc. IEEE INFOCOM, pp. 525-533, 2010

[4] B. Wang, M. Li, S.S. Chow, and H. Li, Computing Encrypted Cloud Data Efficiently under Multiple Keys, Proc. IEEE Conf. Comm. and Network Security (CNS 13), pp. 90-99, 2013..

[5] R. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public Key Cryptosystems,” Comm. ACM, vol. 21, no. 2, pp. 120-126, 1978.

[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner,Z. Peterson, and D. Song,“Provable Data Possession at Untrusted Stores,” Proc. 14th ACM Conf. Computer and Comm.Security (CCS ’07), pp. 598-610, 2007.

[7] B. Wang, Student Member, IEEE , B. Li, Senior Member, IEEE, and Hui Li, Member , IEEE ”Oruta: Privacy Preserving Public Auditing “ IEEE Transaction On Cloud Computing Vol.2, No.1, January-March 2014.

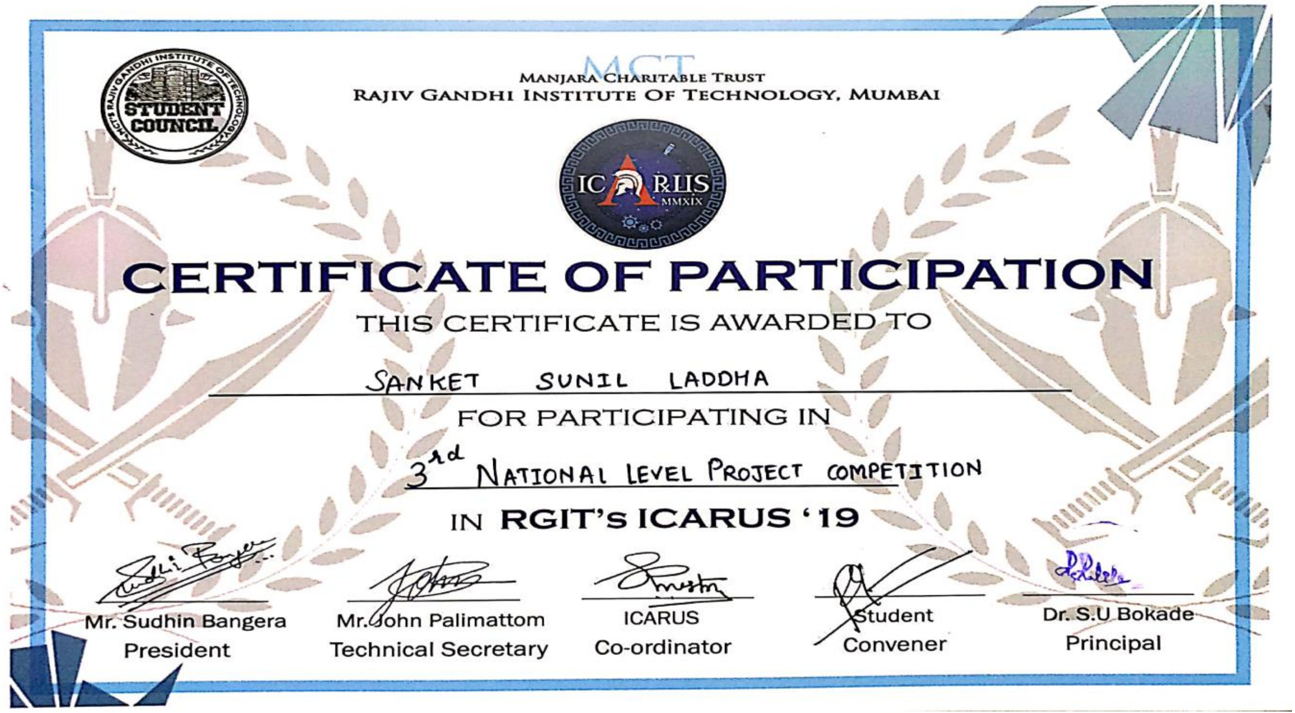
[8] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, “Aggregate and Verifiably Encrypted Signatures from Bilinear Maps,” Proc . 22nd Int’l Conf. Theory and Applications of Cryptographic Techniques:Advances in Cryptology (EUROCRYPT’03), pp. 416-432, 2003.

[9] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,” Proc. IEEE INFOCOM, pp. 525-533, 2010.

Appendix C

Project Competition

We participated at National Level Project Competition, at Rajiv Gandhi Institute of Technology, Andheri.



MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI

CERTIFICATE OF PARTICIPATION

THIS CERTIFICATE IS AWARDED TO

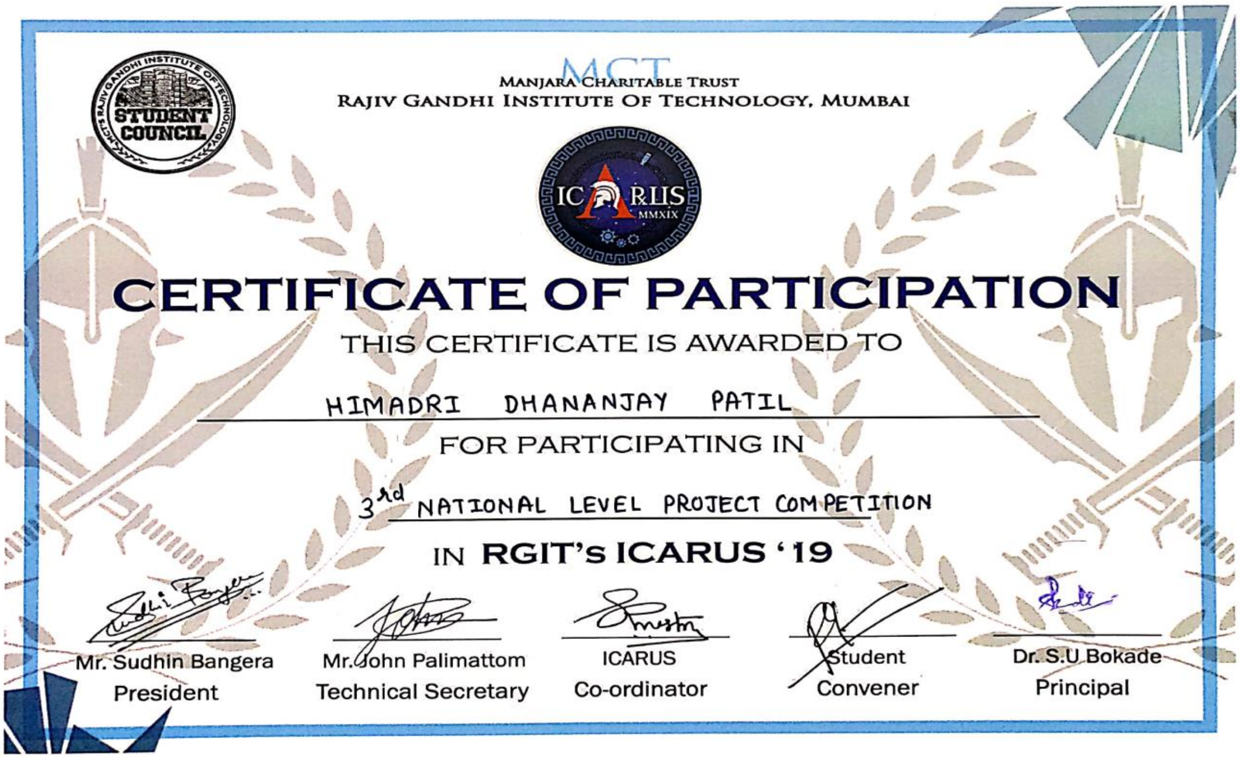
SANKET SUNIL LADDHA

FOR PARTICIPATING IN

3rd NATIONAL LEVEL PROJECT COMPETITION

IN **RGIT's ICARUS '19**

Mr. Sudhin Bangera President	Mr. John Palimattom Technical Secretary	ICARUS Co-ordinator	Student Convener	Dr. S.U Bokade Principal



MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI

CERTIFICATE OF PARTICIPATION

THIS CERTIFICATE IS AWARDED TO

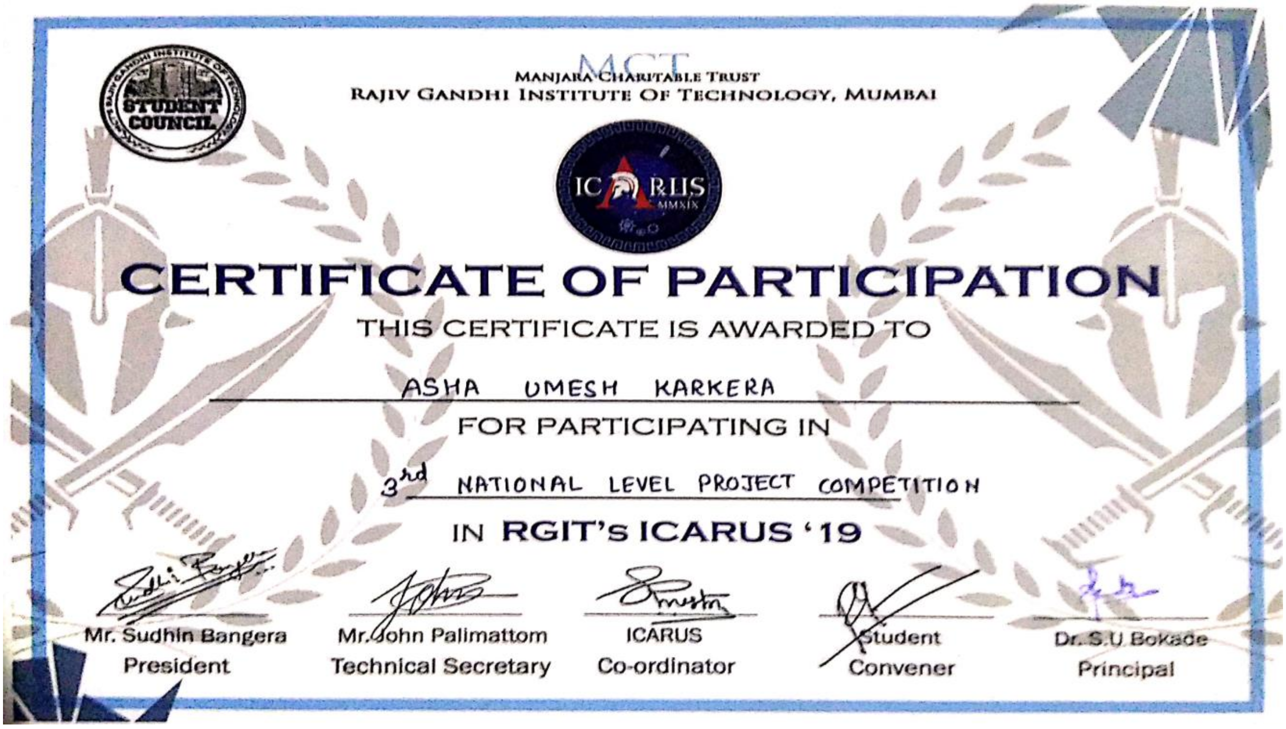
HIMADRI DHANANJAY PATIL

FOR PARTICIPATING IN

3rd NATIONAL LEVEL PROJECT COMPETITION

IN **RGIT's ICARUS '19**

Mr. Sudhin Bangera President	Mr. John Palimattom Technical Secretary	ICARUS Co-ordinator	Student Convener	Dr. S.U Bokade Principal



MCT
MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI

STUDENT COUNCIL

ICARUS MMXIX

CERTIFICATE OF PARTICIPATION

THIS CERTIFICATE IS AWARDED TO

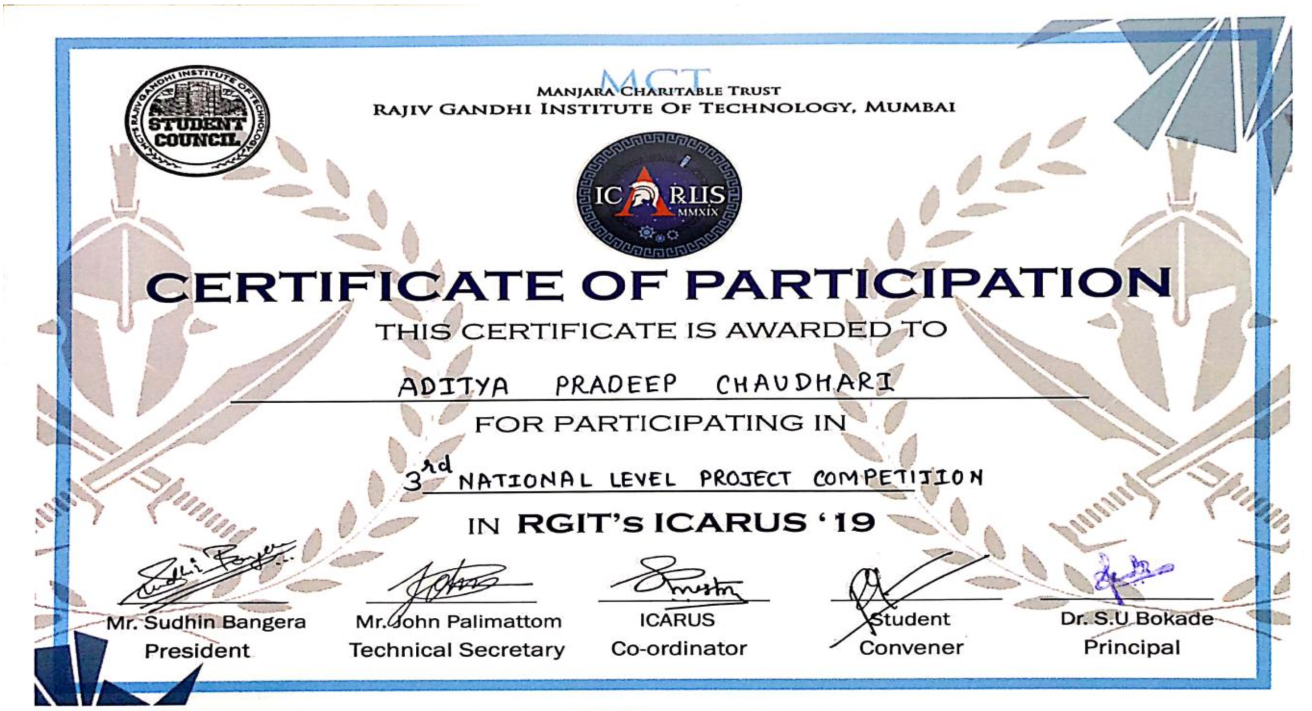
ASHA UMESH KARKERA

FOR PARTICIPATING IN

3rd NATIONAL LEVEL PROJECT COMPETITION

IN **RGIT's ICARUS '19**

Mr. Sudhin Bangera President	Mr. John Palimattom Technical Secretary	ICARUS Co-ordinator	Student Convener	Dr. S.U Bokade Principal



MCT
MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI

STUDENT COUNCIL

ICARUS MMXIX

CERTIFICATE OF PARTICIPATION

THIS CERTIFICATE IS AWARDED TO

ADITYA PRADEEP CHAUDHARI

FOR PARTICIPATING IN

3rd NATIONAL LEVEL PROJECT COMPETITION

IN **RGIT's ICARUS '19**

Mr. Sudhin Bangera President	Mr. John Palimattom Technical Secretary	ICARUS Co-ordinator	Student Convener	Dr. S.U Bokade Principal

Acknowledgement

We take this opportunity to express my profound gratitude and deep regards to our guide **Mrs. Sheetal Ahir** for her exemplary guidance, monitoring and constant encouragement throughout the completion of this report. We truly grateful to her efforts to improve our understanding towards various concepts and technical skills required in our project. The blessing, help and guidance given by him/her time to time shall carry us a long way in the journey of life on which we are about to embark.

We take this privilege to express our sincere thanks to **Dr. Ramesh Vasappanavara**, Principal, RAIT for providing the much necessary facilities. We are also thankful to **Dr. Leena Ragha**, Head of Department of Computer Engineering, Project Co-ordinator **Mrs. Smita Bharne**, Department of Computer Engineering, RAIT, Nerul Navi Mumbai for their generous support.

Last but not the least we would also like to thank all those who have directly or indirectly helped us in completion of this thesis.

Mr Sanket Laddha
Ms Himadri Patil
Ms.Asha Karkera
Mr Aditya Chaudhari