PROJECT NAME



Disclaimer: The content is curated for educational purposes only.



Group Data Sharing in cloud computing

On Identity Basesd Encryption

**Team Members:**

THELLAMEKALA REDDY PRAKASH

**Guide:**

Uma Maheshwari R

MEENIGE KULLAI SWAMI

PONNAGANTI KULADEEP CHOWDARY

MANCHURI LAKSHMI VARA PRASAD

BUDDALA REDDY SHEKAR REDDY

# OUTLINE

* Abstract
* Problem Statement
* Aims, Objective & Proposed System/Solution
* System Design/Architecture
* System Development Approach (Technology Used)
* Algorithm & Deployment
* Conclusion
* Future Scope
* References
* Images of the Project

# Abstract

Identity Based Encryption (IBE) is an extension of public-key encryption (PKE) that can use the user’s identity string as a public key. The first IBE scheme was proposed earlier, and they constructed their IBE scheme by using a bilinear map and proved its security in the random oracle model. Since then, various IBE schemes have been proposed in bilinear maps. In order to use an IBE scheme in a real application environment, an Revocable IBE scheme that provides the functionality of effectively revoking a user’s private key is required. Various RIBE schemes have been proposed to enhance the security or improve the performance.

Generic methods for designing RIBE schemes using binary trees has been proposed . A key principle that enables the generic RIBE design is to make the path of a binary tree associated with a ciphertext instead of associated with a private key. However, since the generic RIBE design requires a larger binary tree compared to the existing direct RIBE design method, there is a problem of inefficiency in terms of update key size. In order to reduce the computational load of a key generation center in RIBE, an RIBE scheme that delegates the generation of update keys to a cloud server was proposed, but the proposed RIBE scheme has a problem that the size of an update key increases in proportion to the number of users.

we newly introduce the concept of delegated RIBE (DRIBE) that can delegate the generation of update keys to the semi-trusted cloud server and define the security models of DRIBE. Next, we propose a DRIBE scheme by generically combining a hierarchical IBE (HIBE) scheme, an identitybased broadcast encryption (IBBE) scheme, and a collision-resistant hash function. In addition, we propose a DRIBE-INC scheme that generates an occasional base update key and a periodic incremental update key to reduce the size of update keys in our DRIBE scheme.

## ProblemStatement

In the real of cloud computing, data sharing among groups demands robust security mechanisms to safeguard sensitive information. Identity-Based Encryption (IBE) presents a promising approach for securing data in cloud environments, offering convenience and flexibility in key management. However, leveraging IBE for group data sharing in cloud computing introduces several challenges and complexities that require thorough examination.

# Aim and Objective

**Aim:**

The aim of the study is to enhance the security and efficiency of group data sharing in cloud computing environments through the implementation of Identity-Based Encryption (IBE) mechanisms.

**Objectives:**

* **Develop Secure Group Data Sharing Framework**: Design and develop a robust framework for secure group data sharing in cloud computing environments leveraging Identity-Based Encryption. This framework should incorporate mechanisms for efficient key management, access control, and secure communication among group members.
* **Investigate Scalable Key Management Solutions**: Explore scalable key management solutions tailored to group data sharing scenarios in cloud computing. Investigate techniques for efficient key generation, distribution, and revocation to accommodate dynamic group membership and evolving security requirements.
* **Enhance Access Control Mechanisms**: Enhance access control mechanisms to support fine-grained access policies and dynamic group membership changes. Develop algorithms and protocols for enforcing access policies while maintaining the confidentiality and integrity of shared data.

# Proposed Solution

Implement Identity-Based Encryption (IBE) to simplify key management and access control.Develop a system for managing encryption keys tailored to group dynamics.

Establish access control rules to determine who can access shared data within the group.

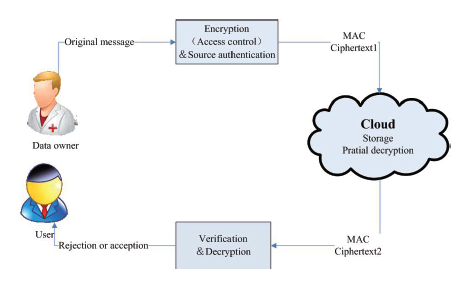
Create secure channels for communication to protect data during transmission.Implement procedures for key revocation and updates to maintain security.

Optimize system performance to handle large groups and data volumes.

Ensure interoperability with various cloud services and platforms.

Design the system to be user-friendly, enabling easy data sharing among group members.

# System Architecture



# System Deployment Approach

Technologies are used:

* **Frontend:** HTML, CSS, JavaScript.
* **Backend:** Java, Servlets, JSP for dynamic content generation.
* **Database:** Mysql
* **Web Server:** Tomcat Server.

# Algorithm & Deployment

**Algorithm:**

**Input:**

**GMem**-> Group Member

**GMan**-> Group Manager

**CS->** Cloud Server

**Output:**

Result-> R

Step 1: **GMem** register, login and upload files

Step 2: **GMem**view files which uploaded

Step 3: **GMem**-> (req.) GMkey from **GMan**

Step 4: **GMan**generates key **->GMem**

Step 5: **GMem**<-(reci.)**GMan**key **->**(req.) **CS**

Step 6: **CS** -> (send) cloud key **->GMem**

Step 7: KAC encrypt all files

Step 8: **GMem**access user files

Step 9: **GMem**decrypt files using **KAC**

Step 10: **GMem**get **R**

**Deployment Steps:**

1.Planning and Requirement Analysis:Define the specific requirements and objectives for group data sharing in the cloud using Identity-Based Encryption. Identify the target user groups, types of data to be shared, and access control policies.

2.Infrastructure Setup:Set up the cloud computing infrastructure required for deploying the group data sharing system. This includes provisioning servers, configuring network settings, and ensuring connectivity to cloud storage services.

3.Installation and Configuration:Install the necessary software components for implementing Identity-Based Encryption and group data sharing functionality. Configure the encryption algorithms, access control policies, and integration with cloud storage services.

4.Identity Management Configuration:Configure the identity management module to manage user identities and attributes effectively. Set up user registration processes, authentication mechanisms, and attribute-based access control policies.

5.Key Management Setup:Establish the key management and distribution module to generate, distribute, and manage encryption keys for group data sharing. Implement mechanisms for key generation, distribution, revocation, and updates based on group membership changes.

6.Access Control Configuration:Configure the access control module to enforce fine-grained access policies for shared data within the group. Define access control rules based on user roles, attributes, and group memberships, ensuring that only authorized users can access specific data.

7.Encryption and Decryption Configuration:Configure the encryption and decryption engine to encrypt data using Identity-Based Encryption techniques. Implement encryption algorithms to secure data-at-rest and in-transit, ensuring confidentiality and integrity during transmission and storage.

# Conclusion

In conclusion, delegating and verifying the update keys of revocable identity-based encryption (RIBE) systems is a critical aspect in ensuring the security and efficiency of these cryptographic schemes. By allowing authorized parties to delegate their update keys, RIBE systems can accommodate dynamic environments and evolving access control requirements.

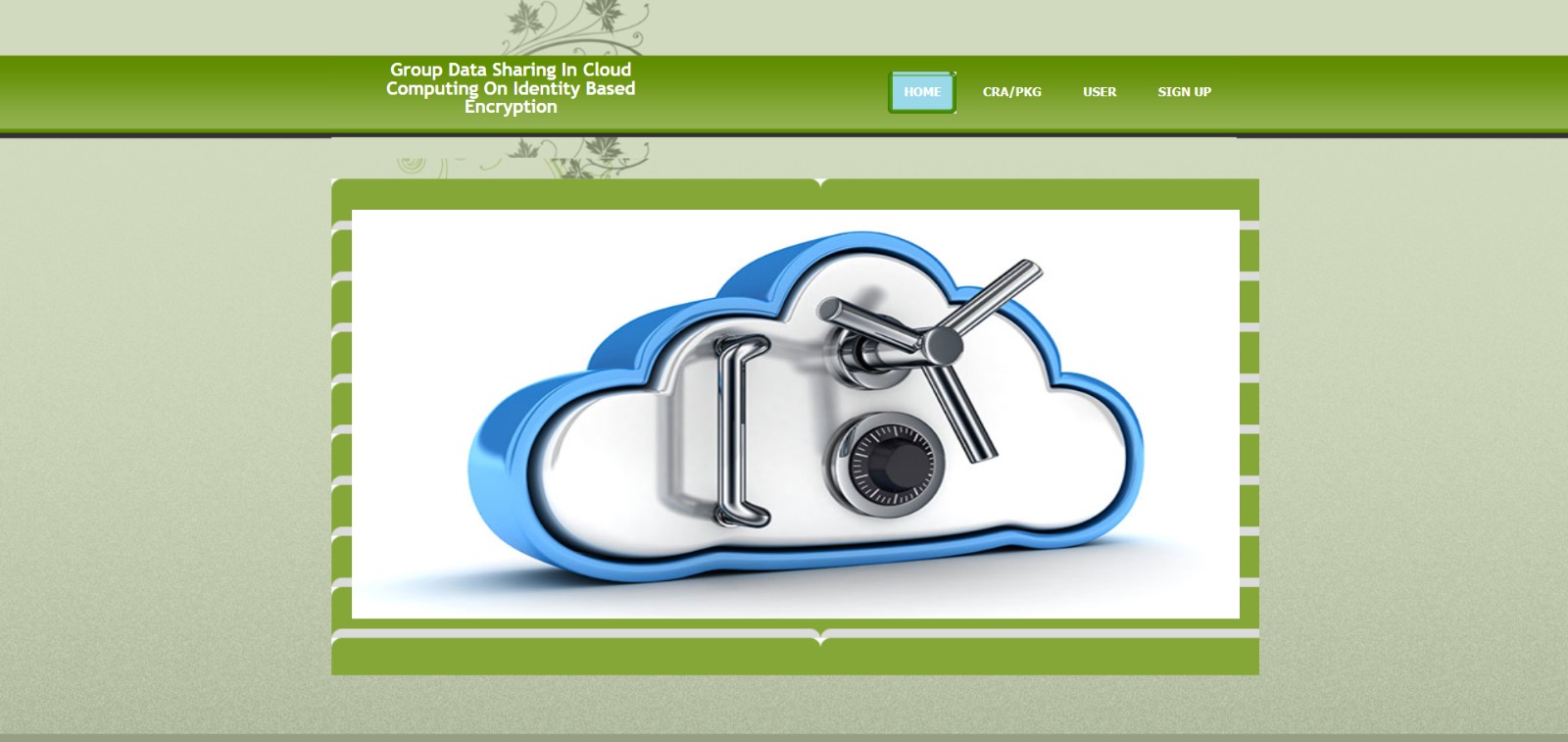
# Future Scope

The most effective method to ensure clients' information protection is a focal inquiry of distributed storage. With more scientific apparatuses, cryptographic plans are getting more adaptable and regularly include different keys for a solitary application. In this article, we consider how to "pack" mystery keys out in the open key cryptosystems which bolster assignment f mystery keys for distinctive cipher text classes in distributed storage. Regardless of which one among the force set of classes, the delegate can simply get a total key of consistent size..

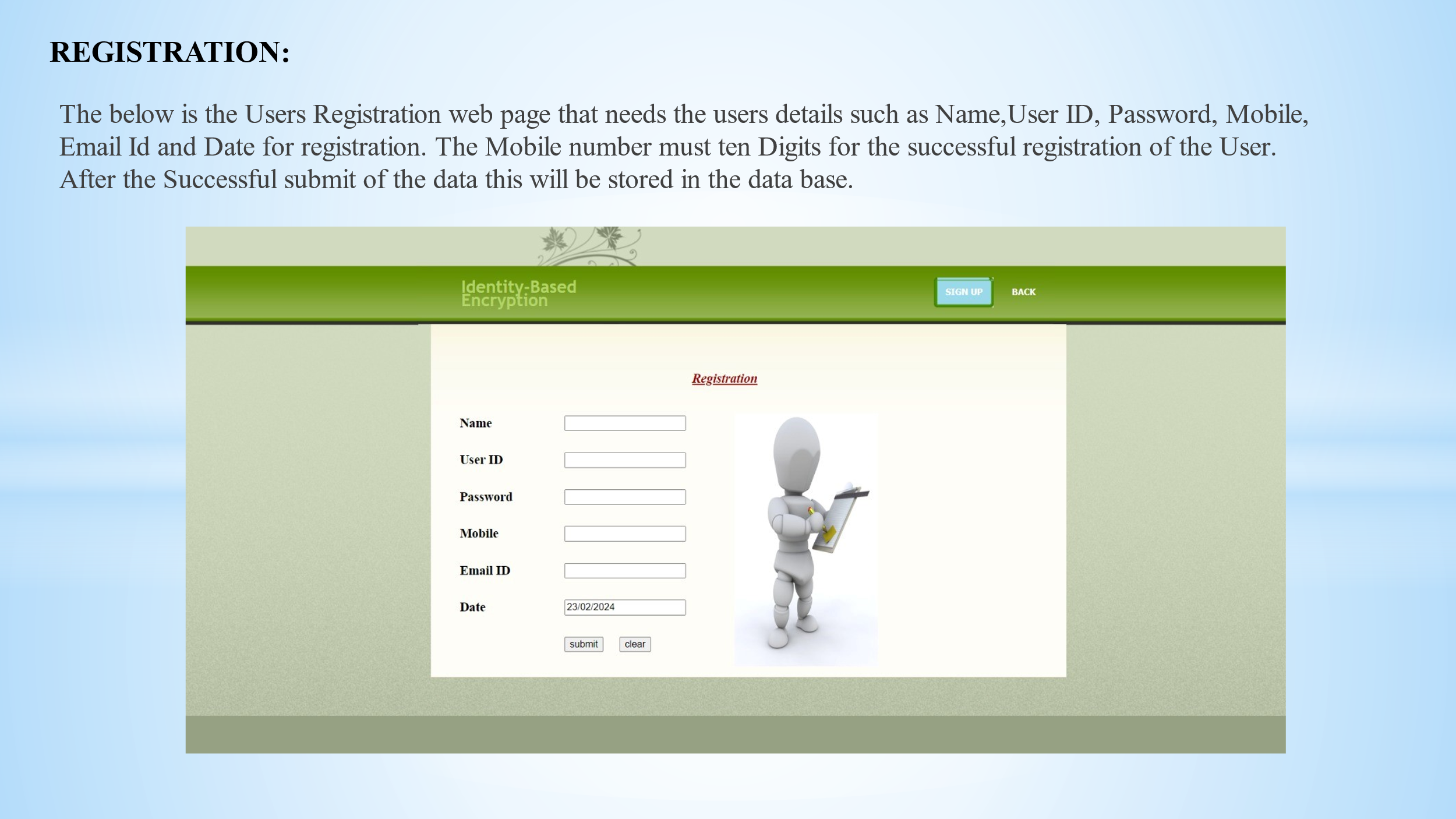
**Reference**

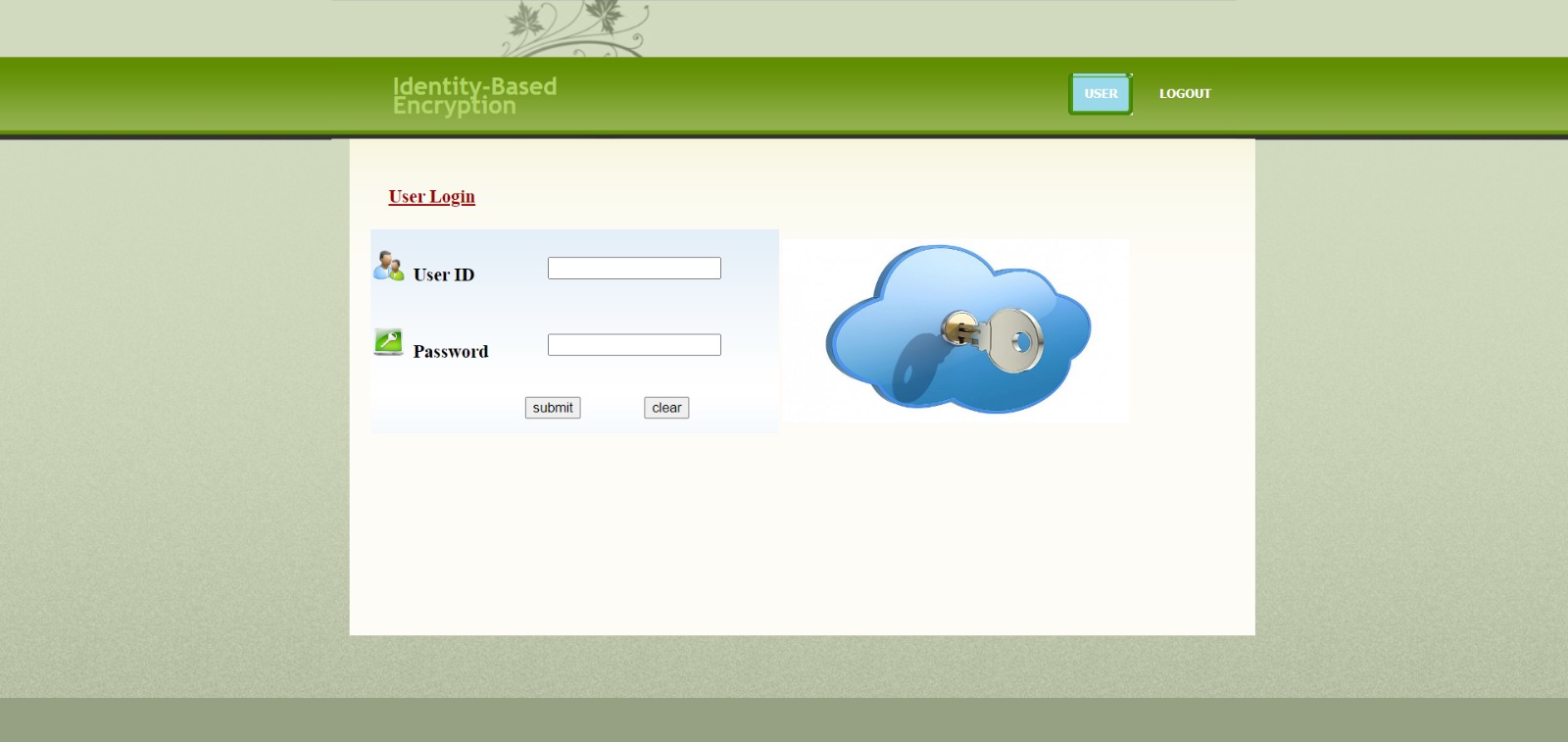
1. D. Boneh and M. K. Franklin, ‘‘Identity-based encryption from the Weil pairing,’’ in Advances in Cryptology— CRYPTO 2001, vol. 2139. Berlin, Germany: Springer, 2001, pp. 213–229.
2. A. Boldyreva, V. Goyal, and V. Kumar, ‘‘Identity-based encryption with efficient revocation,’’ in Proc. 15th ACM Conf. Comput. Commun. Secur., Oct. 2008, pp. 417–426.
3. B. Libert and D. Vergnaud, ‘‘Adaptive-ID secure revocable identity-based encryption,’’ in Topics in Cryptology—CT-RSA 2009, vol. 5473. Berlin, Germany: Springer, 2009, pp. 1–15.

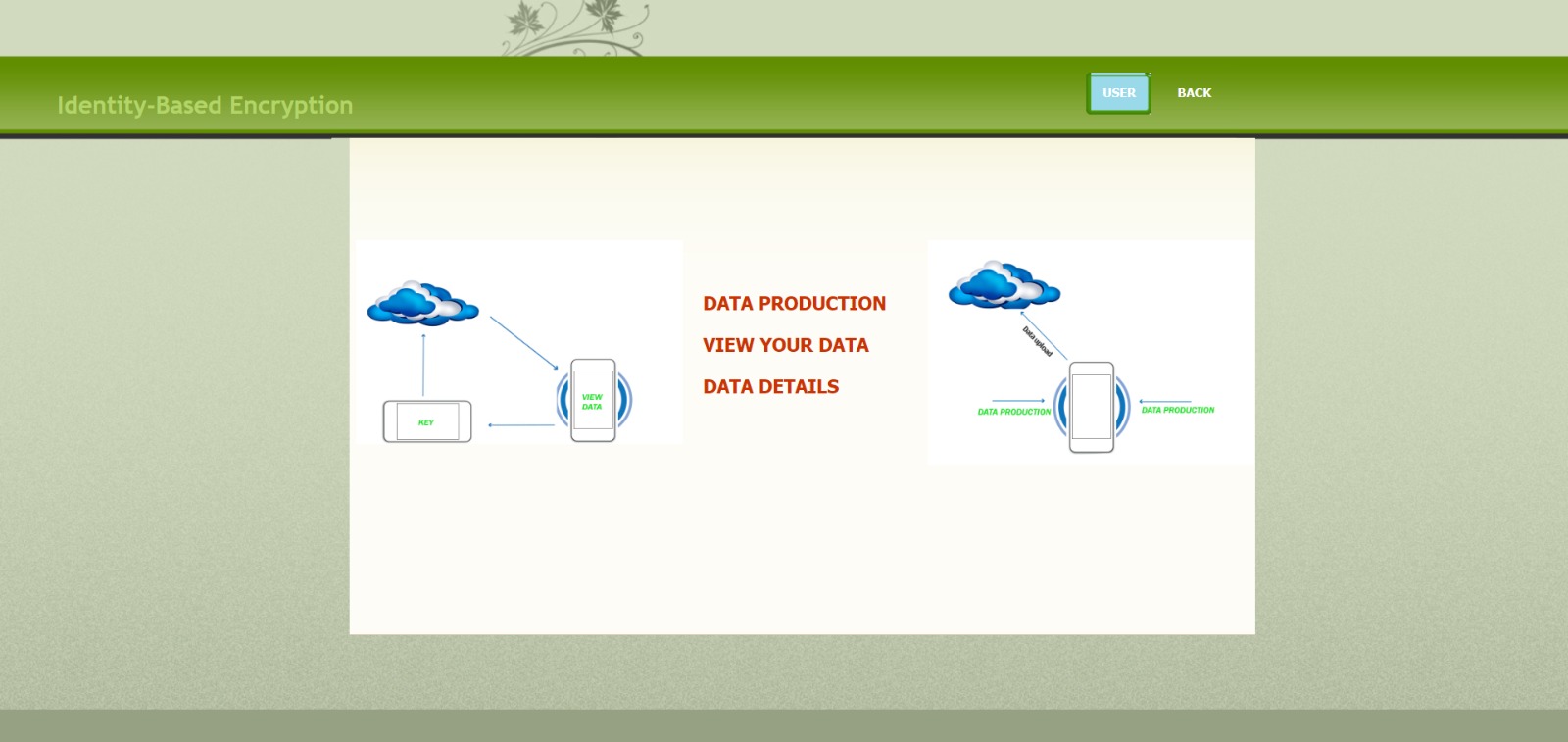
Outputs: HTML Page:



JSP Page:







**GITHUB**

<https://github.com/20AK1A0564/Identity-Based-Encryption.git>

**VIDEO LINK**

<https://drive.google.com/file/d/1OqCnpuI0t7qvB_H92w8y80SNcYi7JYPx/view?usp=sharing>

**Thank you!**