Project Report

**Group members:**

Harikrishna Sappidi (**Group Leader**)

Gayathri Pittu

Manojkumar Reddy Matli

Akhil Yerrabothu

Abhi Patibandla

**Project Title:**

Secure File Sharing using Cryptography

**Project Description:**

In our day-to-day life we share a lot of secret and sensitive information with our peers. We are proposing a system that mostly focusses on securing the files while sharing them with the help of cryptographic algorithms. Cryptographic algorithms facilitate an efficient way to store the files and to transmit the files without being captured by the third parties. We will make a web application to transfer the files between users. We use html, CSS, Django framework, SQL, and Python 3 programming language to implement our proposal.

**Entities:**

Below are the entities involved in our software

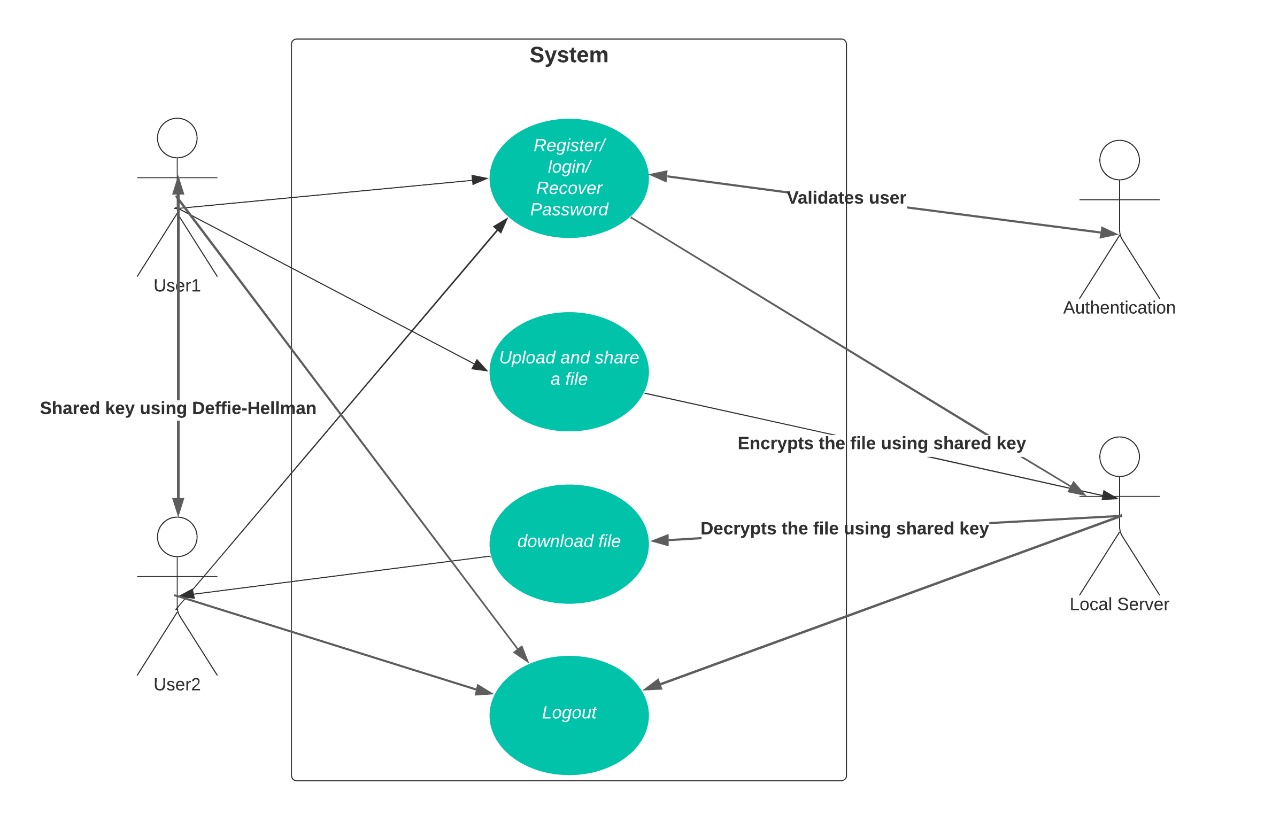
Users

Local Server

Files

**Functions:**

The functions of our software are represented using an UML diagram below.



Use case Diagram (High Level)

Users: Users can register their profile and should setup a password with respect to the secure password policies that we implemented, and they will be able to login to the system and can upload files from their local machine to the application and can transfer them to the desired users and receivers will be able to download by decrypting them using shared key.

**Functionality of System:**

To share files, first users must create an account in our website. We are using some password metrics, so if the user chooses a weak password, we suggest to them create a strong password.

Users’ login credentials will be stored in database in the form of hash. Even if database is compromised, then attackers will only be able to see the hash values which are highly impossible to crack the actual passwords. This is the first security feature we are providing to the users.

Diagram

Description automatically generated

Fig 1.1 Working of Hash algorithm

Once the authentication is done, user can be able to share or view the files. Suppose User1 wants to share a file with user2, they upload the file and give key value to encrypt the file. And file encryption is done using AES Algorithm. User2 will login into the system and user2 must provide key value to view the file shared by user1. If the key value is matching with user1 key value, then the file will be decrypted. Here the key sharing will be done using Diffie-Hellman algorithm. As per the Diffie-Hellman protocol, users share their computed values which are public to find shared key value.

Security Features:

Our software will have many security features like Confidentiality, Integrity, Authentication, Authorization.

***Confidentiality***: Information that should stay secret stays secret and only those authorized to access it may receive access.

***Integrity***: Trustworthiness, origin, completeness, and correctness of information as well as the prevention of improper or unauthorized modification of information.

***Authentication***: Verification of the identity who tries to access the resource.

**Authorization**: Verification of access rights to resources related to information security and computer security in general and to access control.

Potential Methods:

We will use the cryptographic algorithms like *Hashing, AES, Diffie – Hellman protocol.* SHA256 Hashing algorithm will be used to hash the passwords of users and to achieve Integrity. We use AES algorithm for file encryption and decryption to achieve Confidentiality. Validating the user login credentials for Authentication and authorization.

We chose AES algorithm because the main benefit of AES lies in its key length options. The time required to crack an encryption algorithm is directly related to the length of the key used to secure the communication

We chose Diffie – Hellman protocol because It allows the two parties to communicate over a potentially dangerous connection and still come up with a shared secret that they can use to make encryption keys for their future communications.

Roles:

Front End Developers and Designers:

* + Akhil Yerrabothu
  + Gayathri Pittu

Backend Developers and Testers:

* Manojkumar Reddy Matli
* Abhi Patibandla
* Harikrishna Sappidi

Akhil and Gayathri work on front end design. They will design User registration, login webpages and they mention password metrics while creating registration page. And they are providing the functionalities to the users to view or share the files using website. They will use HTML, CSS, DJANGO framework to create webpages.

Manoj Kumar, Abhi, and I work on the backend. We create a database to store user’s information. We work using SQL, Django frameworks for creating connection between frontend and backend and implementing the cryptography algorithms.

**Conclusion:**

The goal of this proposal is to create a secure channel for users to share the information in the form of files. Our main goal is to implement all the security features on our own because all the cloud servers (Ex: Azure, OneDrive) will be embedded with the security features by default. We chose to do it in our local host because we can implement the security features from scratch like hashing the password and storing in local database, Diffie-Hellman protocol for secure file sharing, and AES for Encryptions and decryptions.

**References:**

1. <https://codesigningstore.com/what-is-hashing-algorithm-how-it-works>
2. <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiMw7y-i_D6AhVCk4kEHbbdAXUQFnoECAsQAQ&url=https%3A%2F%2Fijcsmc.com%2Fdocs%2Fpapers%2FMay2021%2FV10I5202113.pdf&usg=AOvVaw2ukEFeUig3L5_Dxs81grf0>
3. <https://journals.resaim.com/ijramt/article/view/695/668>
4. <https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard#:~:text=The%20main%20benefit%20of%20AES,56%2Dbit%20key%20of%20DES>.
5. <https://www.comparitech.com/blog/information-security/diffie-hellman-key-exchange/#:~:text=This%20structure%20of%20the%20Diffie,keys%20for%20their%20future%20communications>.