**SECURE FILE STORAGE ON CLOUD USING CRYPTOGRAPHY**

**Abstract**

In this paper we aim to securely store information into the cloud, by splitting data into several chunks and storing parts of it on cloud in a manner that preserves data confidentiality, integrity and ensures availability. The rapidly increased use of cloud computing in the many organization and IT industries provides new software with low cost. Cloud computing is beneficial in terms of low cost and accessibility of data. Cloud computing gives lot of benefits with low cost and of data accessibility through Internet. Ensuring the security of cloud computing is a major factor in the cloud computing environment, as users often store sensitive information with cloud storage providers, but these providers may be untrusted. So sharing data in secure manner while preserving data from an untrusted cloud is still a challenging issue. Our approach ensures the security and privacy of client sensitive information by storing data across single cloud, using AES, DES and RC2 algorithm.

**SYSTEM ANALYSIS**

**EXISTING SYSTEM:**

* more data storage needs turning over to the cloud, finding a secure and efficient data access structure has become a major research issue
* Security techniques are not applied in the protection of offloaded data from attacks.
* Once uploaded and shared, the data owner inevitably loses control over the data, opening the door to unauthorized data access.

**PROPOSED SYSTEM:**

Selectively sharing data files on the cloud becomes a burden on the data owner as the hierarchy grows (the access privileges increase in number) and/or as the access restrictions become more complex due to an increase in the sensitivity of the file segments. A trivial solution involves the data owner to use public key encryption. This solution would require the data owner to encrypt the same part of the data file once for each data user being granted access then upload the resulting cipher texts to the cloud. The data users would then fetch their uniquely encrypted parts of the file from the cloud and utilize their private keys to decrypt them. This method ensures that no unprivileged data user will gain access to any part of the data file even if that user is able to download the cipher texts from the cloud. However, on a large scale, public key encryption becomes an inefficient solution due to the increase in the number of encryptions and large storage spaces required. Therefore, the challenge is to provide the data owners with an efficient, secure and privilege-based method that allows them to selectively share their data files among multiple data users while minimizing the required cloud storage space needed to store the encrypted data segments.

* Requiring less network communication.
* We present multiple data file partitioning techniques and propose a privilege-based access structure that facilitate data sharing in hierarchical settings.
* A new security layer is added to encrypt the data of the task before transferring to the cloud side by using AES encryption technique.

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System : Pentium IV 2.4 GHz.
* Hard Disk : 40 GB.
* Floppy Drive : 1.44 Mb.
* Monitor : 15 VGA Colour.
* Mouse : Logitech.
* Ram : 512 Mb.

**SOFTWARE REQUIREMENTS:**

* Operating system : - Windows XP/7.
* Coding Language : Python
* Data Base : MYSQL

**CONCLUSION**

The numerous beneﬁts provided by the cloud have driven many large multilevel organizations to store and share their data on it. This paper begins by pointing out major security concerns data owners have when sharing their data on the cloud. Next, the most widely implemented and researched data sharing schemes are brieﬂy discussed revealing points of weakness in each. To address the concerns, this paper proposes a Privilege-based Multilevel Organizational Data sharing scheme that allows data to be shared efficiently and securely on the cloud. partitions a data ﬁle into multiple segments based on user privileges and data sensitivity. Each segment of the data ﬁle is then shared depending on data user privileges. We formally prove that is secure against adaptively chosen plaintext attack assuming that the DBDH assumption holds. Our comprehensive performance and simulation comparisons with the three most representative schemes show that cans signiﬁcantly reduce the computational complexity while minimizing the storage space. Our proposed scheme lays a foundation for future attribute-based, secure data management and smart contract development.