AN ALGORITHM FOR ENHANCING THE SECURITY ON CLOUD DATA SHARING SYSTEM

PRARTHANA H R[1], YOGISH H K[2]

[1,2]SAPTHAGIRI COLLEGE OF ENGINEERING, BANGALORE -560057, INDIA

ABSTRACT

Cloud data Storage is a new innovative service model of data storage where data from the organizations are remotely maintaining, managing, and backup. This system will allow the users of different prospective and requirements to store their files online and access them from any location via the Internet The most critical aspect in cloud data storage environment will rise the different security issues based on the group data sharing which relates to both cloud centric and conventional insider treats. The focus of cloud data storage architecture is to achieve the goal of providing the security and privacy for the user data that is shared among the group of defined user. However there are many ways that the malicious users are attempting the get an access to the shared data and also the personal information stored on cloud servers. Thus there is a need of new techniques to solve the serious problems in the area of information security for group sharing concept. To highlight this aspect, in this paper we are proposing and developing an prototype system that will encrypt the user file to be shared with the group through the use of trusted cryptographic server to ensure the data sharing, confidentiality and access control.

Keywords: .NET framework, Cloud computing, C# language, Single key encryption, Information security.

INTRODUCTION

The new evolution in recent days of cloud computing is the cloud computing storage system which was not designed from scratch but it is an evolution of the many of the computer architectures such as web system, networking, application infrastructure, database management system. The Cloud computing system constitutes infrastructures, platforms and applications on demand. Now cloud storage system is an integrated cross platform system for mechanism of storage and retrieval of data. It is a data model which stores huge digital data in logical pools of physical storages. The physical storages will span over multiple servers (often located in different places of the world) and the physical data storage environment is owned by the hosting company.

Cloud computing is an growing architecture continues to provide an large savings for investment in IT Industries, the popularity for cloud is rising in many mission-critical areas which are very sensitive such as medical areas and energy areas. The Cloud computing always provides cost effective scalable services for deployment and infrastructure for these sectors with large computing power and increase in productivity. But, the availability of data, confidentiality and integrity of data are of great importance in these sectors. The chapter provides a brief introduction of cloud security. This

introduces the techniques that are presently adopted in securing the cloud from attacks and then introduces security algorithms that are popularly used to raise the data security for cloud processing.

This is most important issue in the cloud computing architecture has become as nightmare issue to be solved.

The biggest cloud computing treats that listed are,

- **The Data perforation** – This is one nightmare existing in the virtual machine and hypervisor cloud systems. In this a user who is logon in one virtual machine will listen the arrival of an encryption key in the form of activity signal of another virtual machine running on the same host. The attack is used to steal the personal and credit card information from the cloud storage.
- **The Data diddling** – the data diddling may occur due to the malicious attack in which the owner will lose the encryption key or the encrypted data.
- **Communication blocking and deviation** – this is due to the Denial of service attacks. But with the cloud infrastructure this may sound less but still remains as one of the software vulnerabilities because of the overflow the buffers.
- **Misuse Of Cloud Services -** The Cloud computing comes with a elastic, large-scale services to users and hackers protected. The encryption key used might take attacker years to crack using his own limited hardware. But, using an array of cloud servers, he might be able to crack it in minutes. Thus, the hackers uses cloud servers to transmit malware, perform DDoS attacks, or circulate pirated software.
- **Insufficient investigation and examination -** Enterprise systems jump into the cloud representation without knowing the full scope of the cloud system. Without knowing the service providers' environment and protections. Customers never have clarity regarding what to expect in the way of incident response, encryption use, and security monitoring. Not having enough knowledge about these factors means organizations are taking on unknown levels of risk by using the cloud system, but that are a far departure from their current risks of attacks.
- **Shared Technology -** In a multi-tenant environment, the hypervisor used by a single customer is exposes the customer who has compromised with the cloud system,rather than exposing the entire environment of the hypervisor. The same could be explained with other shared services, which includes CPU cache with shared technique, a database service with shared architecture, or storage with shared principle. The cloud system with shared infrastructure, and a misconfigured operating system or application will lead to compromises beyond their high priority surroundings.

Cloud computing treats indicate that without the proper level of security techniques may cause a tremendous social problem in the near future. This paper is to propose a new enhanced encryption prototype for the sharing of data that employs the new way of generating the secret key and performing the encryption of the data which is shared to a owner defined group of users. This method is for improving the some security issues that are listed above. The algorithm uses the cryptographic server (CS) which is a trusted server on the network for the generation of the encryption secret key. The owner of the data will do the encryption of his data and stores it on the cloud along with access control information for the data within the pre identified group.

If any malicious user attacks the cloud or the cryptographic server (CS) obtains the encrypted data using the means virtual machine or any transmission channel or server, encrypted data cannot

decrypted to access the entity of the information shore in it, because the secret key which is generated may not be known. The cryptographic server will not retain any generated key information of that user. Thus obtaining the key details by the malicious user is practical impossible. This implies that the proposed solution will avoid the malicious user access over the shared data which in turn raise the sharing security of the cloud data storage system.

MATERIALS AND METHODS

Storing data on cloud is gaining popularity recently in many application environments. In enterprise systems are converting the central application into cloud architecture, to increase in demand for data outsourcing, which assists in the planned management of business data. The data storing and sharing architectures are used as a basic technology behind many online services for personal applications. Now, it is easy to maintain the accounts for email, photo album, file sharing and/or remote access, with storage size more than 25GB. Mobile and WIFI technology of recent developments made the users to retrieve almost all of their files and emails by a cell phone in any side of the world. data Confidentiality, is a raising issue to ensure it is to rely on the server to enforce the access control after authentication.

Data from different clients can be present on separate virtual machines but reside on a single physical machine. Data in a destination VM could be stolen by instantiating another VM co-resident with the destination one. Regarding availability of files, there are a number of cryptographic schemes which go as far as allowing a third-person auditor to check the availability of files on behalf of the sender without leaking anything about the data, or without compromising the data owner's secrecy. Likewise, cloud users possibly will not hold the strong conviction that the cloud server is doing a good job in terms of secrecy.

A cryptographic solution, with stated security relied on number theoretic assumptions is more attractive whenever the user is not perfectly happy with trusting the security of the Virtual Machine or the honesty of the technical member. These users are encouraged to encrypt their files with their own keys before uploading them on to the cloud. Sharing of data is an vital functionality in cloud storage. The demanding problem is how to effectively share cipher text. So the users can download the cipher text from the storage, decrypt them, then upload them on to the cloud for sharing, but it loses the value of cloud computing. Users should be able to hand over the access rights of the sharing data to others so that they can access these data from the cloud directly.

Cryptographic Key for a Predefined Hierarchy Cryptographic key assignment schemes goal to reduce the cost in storing and managing secret keys for general cryptographic use. Using a tree structure, a key for a given branch can be used to get the keys of its descendant nodes. Just permitting the parent key implicitly grants all the keys of its descendant nodes. The method can be generalized from a tree to a graph. Advanced cryptographic key assignment concept support access policy that can be modeled by an connected graph or a disconnected graph. Most of these concepts produce keys for symmetric-key cryptosystems, even if the key derivations may need modular arithmetic as used in public key cryptosystems, which are generally more costly than symmetric-key operations such as pseudo random function

We Consider the tree structure as an example. A can first classify the ciphertext classes according to their subjects. Each node in the tree represents a secret key, while the child nodes representsthe

keys for single ciphertext classes. Filled circles represent the keys for the classes to be delegated and circles circumvented by dotted lines represent the keys to be permitted. Note every key of the non-child node can derive the keys of its descendant nodes.

Deyan Chen et. al,[1] has made an detail complete analysis for the data privacy protection and security problems. According to the analysis, there is an expectation to integrate and comprehensive security solution to meet the needs of defense in depth. The privacy protection, data identification and isolation of private data are the primary tasks that should be considered during the design of cloud-based applications. According to the analysis, the data security and privacy protection issues are above and is expected to have an integrated solution of security to meet the needs of defense. But, the work as left the important discussion related to the access control and authorization mechanisms should achieve a unified, reusable and scalable access control model and meet the need of fine-grained access authorization over the shared group of users. Attribute based data sharing on cloud promises the better security for the data. The CP-ABE defined in [2] a solution for fine-grained access control of data stored in cloud for sharing. In this algorithm, Every user are associated with a set of attributes. The data are encrypted with access structures based on the attribute set.

The Hierarchical attribute-based encryption (HABE) and scalable user revocation method for data sharing security prototype [3] clearly shows the new approach of handling the attributes based on the hierarchy for the encryption and scalable user revocation problems of data sharing in cloud system.

The model, discusses the property for generation of keys in hierarchical form in the HIBE system, and the property flexible access control in the CP-ABE system. These properties are more applicable to the environment of data sharing in the cloud enterprises . The scheme which as combined a hierarchical identity-based encryption system and a ciphertext-policy attribute-based encryption system, so as to obtain a fine-grained access control over the data secrecy. During Decryption the algorithm requires only a constant number of bilinear map operations, to provide high performance. The second solution in this method is scalable revocation scheme which is applied for proxy re-encryption and lazy re-encryption.

The improvement for CP-ABE algorithm is defined in [4], where, it concentrate on the key generation procedure of CP-ABE, proposed a work for improving the attribute based data sharing over the group of users. It is concentrating on the challenging issue of access policy enforcement and support of policy updates. In this prototype, the key escrow problem is solved using escrow-free key issuing protocol, which constructs using the secure two-party computation between the key generation center and the data-storing center.

In multi-authority Cloud storage system the data access control algorithm is shown in the paper of Yang et al. [8]. This work proposes a data access control technique for multi-authority cloud storage and it is called as DAC-MACS for an secure data access control scheme with efficient decryption and revocation. Specifically, their approach [8] aims at constructing a new multi-authority cipher-text policy for attribute based encryption (CP-ABE) [9] scheme in which there is an efficient decryption, and also a design for efficient revocation method based on attributes that shows both forward and backward security.

The multi-authority CP-ABE[9] schemes are not useful for the application to access control for multi-authority cloud storage systems because of the deficiency in representing the complete revocation. Thus, the main challenge in this method was to construct a new multi-authority CP-ABE[9] scheme that supports efficient decryption and revocation. The challenging issue considered

in the proposed work was to join different secret keys together and at the same time, to prevent the possibility of a collusion attack.

ARCHITECTURE

The system model for secure data sharing on cloud[6] is shown in figure1. The system model describes the algorithms for exchanging of secret key, preparing the group access policy, encrypting the data, decrypting the data. The system model of the paper shows the process of single user sharing the document to other users.

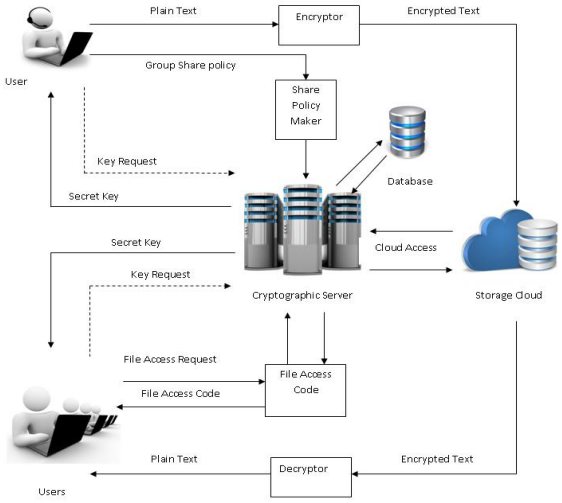

Figure1: System Architecture for Data Sharing on cloud.

The Cryptographic server will manage each user actions. The cloud is used to hold and distribute the encrypted data to all the registered and access permitted users. The cloud is accessed using the single user registration by the cryptographic server and all the users will utilize the same cloud access for their representations.

The cloud is defined as the storage cloud. The paper is illustrated using the Google Drive storage cloud for the purpose. The Google Drive API 2.0 is used to establish the connection and read or write the encrypted file to the server. The complete action is performed by the individual user without the intervention of Cryptographic server. The Cryptographic server as shown in figure2 will aid in the generation of keys, managing of user accounts and building the sharing model between the defined set of users. The Cryptographic server is a multithreaded TCP based program ever running program. The user activities such as new user creation, key management, user file sharing are maintained using the MySql Database system.

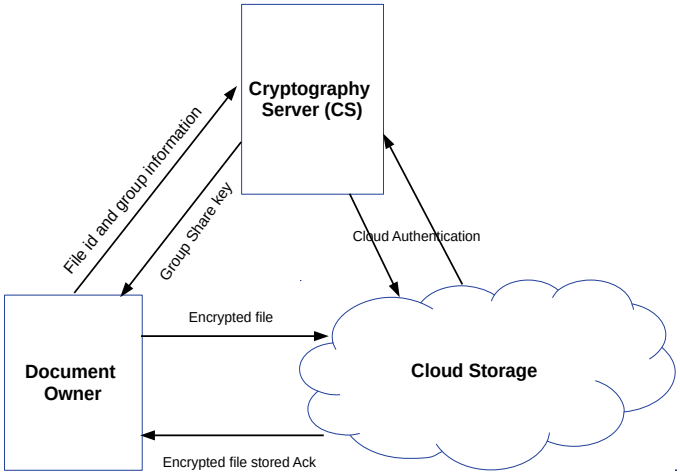

Figure2: Model for file encryption and saving.

The proposed system consists of the following algorithms that are developed.

1. Secret Key Generation
2. Group Policy for File Sharing
3. File Encryption process.
4. File Decryption process.
5. Cloud Authentication process.
6. Storing and Retrieving encrypted file from cloud.

The cloud used in this paper is Google Drive public cloud. The Google Drive will provide two strings such as CLIENT_ID and CLIENT_SECRET which is used to generate the service connection. DriveService class of GoogleDriveAPI package is used to obtain the connection service object. The process of obtaining the CLIENT_ID and CLIENT_SECRET is shown in the figure 3.

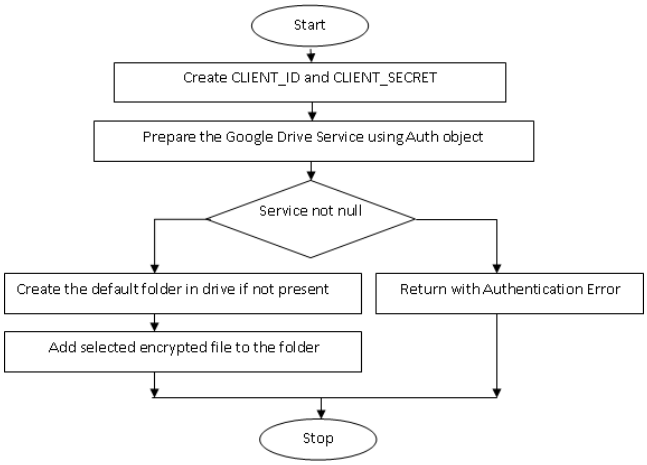

Figure 3: Flowchart for the Google Drive connect and saving of encrypted file.

The hash generator[5] is a supporting algorithm used in the random encryption to convert the given 256 byte biginteger prime number into 256 hash code. The algorithm is shown in the sequence diagram of figure 4.

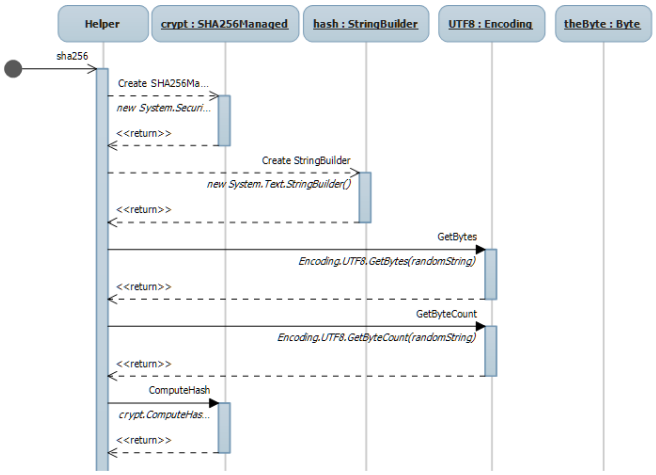

Figure 4: Computing the hashing.

IMPLEMENTATION AND RESULTS

RANDOM ENCRYPTION

The implementation detail of each algorithm is described using the prototype of the algorithm. The encryptstring function will considers the message in the form of text and a six character password as inputs. The algorithm converts the message by using the password and returns the cipher text as output. The steps of execution are shown in the encryptstring function.

```
string EncryptString(string text, string password){
        convert the 6 character password into hash key.
        convert the secret text into byte stream.
        Generate the random number.
        for each byte in secret text stream do
                add the random number into text byte.
        place the random number in the random text.
        set the block size as 128
        set the key size as 256
        for each block from the random text
                convert block into cipher block
        join all blocks to form cipher stream.
        convert cipher stream to cipher text.
        return cipher text.
}
```

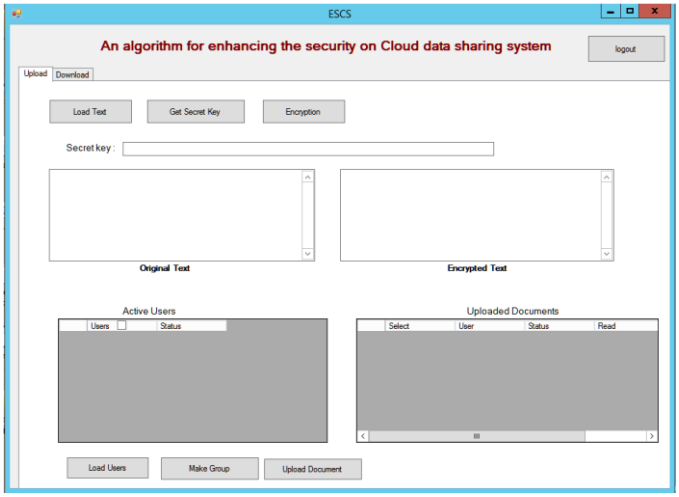

Figure5: User Main form for File Sharing.

The authenticated user is allowed into the main screen of the system on the client side dissipated in figure 5. The client side system consists of two tabs.

1. Upload tab.
2. Download tab.

The user works as the data owner in the upload tab. In upload tab user will select his document, generates the secret key and perform the encryption process to convert his document to cipher document. The owner will prepare the group policy for sharing the document with the set of users using the load user and make group options.

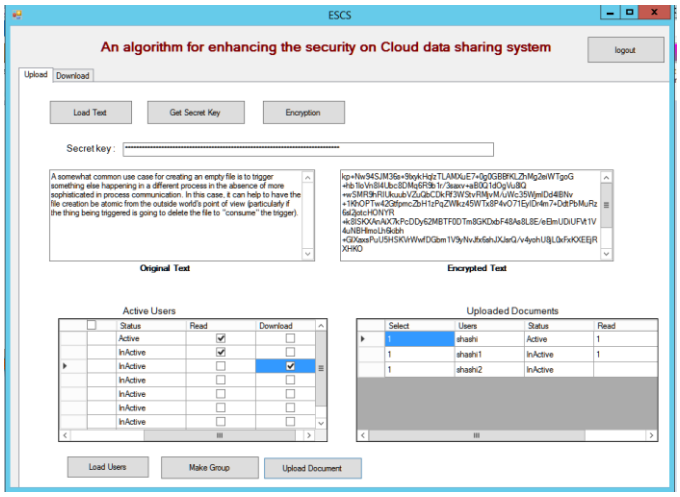

Figure6: User files Select, encrypts and group formation.

The user will upload the encrypted document into the cloud and stores the group policy in the cryptographic server database.

The download tab is used for viewing the shared document to the user by other owner users. In download tab the user will retrieve the document shared. Select the one document at a time to view or download and perform the decryption process.

The figure 6 shows the one illustration of the complete process of encryption and group policy preparation. The user has selected the text document for the uploading. The system is defined to work with the text document. This system can be extended for supporting with other document such as images, videos, word files, PDF documents etc.

The group formation with read and download attributes is also shown in the figure 6 by using the load users and make group options. The active users show all the registered users with their state of login. If the user is presently login to the system then the status is shown as active, otherwise; it displays the user as inactive status.

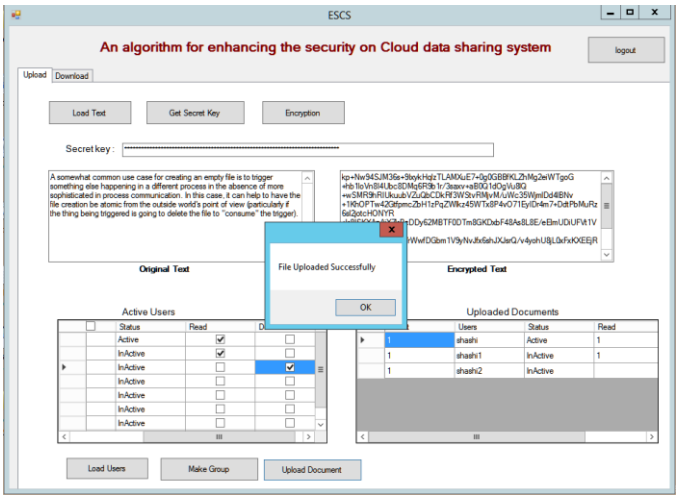

Figure 7: User files upload success by the owner.

The owner client upload process is demonstrated with the snapshot in the figure 7 after performing the encryption and creation of group user policy for sharing the file. The user will store the encrypted file into the Google Drive using the upload document option.

The process first makes authentication with Google Drive using the CLIENT_ID and CLIENT_SECRET fields and creates the drive service. Using the drive service, the process creates a default folder called ESCS folder in the Google drive and then uploads the encrypted document as filename.dat file. Any other user agent is not allowed by Google drive to read or update this document file.

CONCLUSION

This paper is an attempt to demonstrate the complete working model for the user data sharing over the group of users using the public cloud system. This application provides a way for improving the security in cloud system based on shared key technique. The application defines two modules. Firstly, the client module, which provides the provision for uploading and downloading of the document between the set of registered users. The second module is the Cryptographic module, which manages all the users, generates the secret keys and gets connected with Google Drive cloud for storing the documents.

In this paper, the complete prototype sharing system is designed and implemented using the random encryption and random decryption algorithms and the results obtained after the application of these algorithms is discussed. The user management, cloud connectivity and data sharing modules are implemented as the modules with the client and CS system.

The outcome shows the enhancement to the public cloud security and increases the trust with the end user, since the complete system will work independent of the public cloud security system.

REFERENCES

[1] Deyan Chen and Hong Zhao, "Data Security and Privacy Protection Issues in Cloud Computing,"International Conference on Computer Science and Electronics Engineering, pp647-651, 2012.

[2] Shucheng Yu, Cong Wang, Kui Ren, Wenjing Lou, "Attribute Based Data Sharing with Attribute Revocation," Proceeding ASIACCS '10 Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security Pages 261-270, 2010.

[3] Guojun Wang, Qin Liu , Jie Wu , Minyi Guo, "Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers," Journal of Advances in network and system security, Vol 30, Issue 5, pp 320–331, 2011.

[4] Junbeom Hur, "Improving security and efficiency in attribute-based data sharing, " IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 10, pp. 2271-2282, 2013.

[5] M.J.Atallah, MarinaBlanton, NellyFazio, Keith B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009.

[6] Mazhar Ali, Revathi Dhamotharan, Eraj Khan, Samee U. Khan, Athanasios V. Vasilakos, Keqin Li, Albert Y. Zomaya "SeDaSC: Secure Data Sharing in Clouds". IEEE Systems Journal Volume: Issue: 99), Page(s): 1 – 10, January 2015.

[7] K. A. Muthukumar, M Nandhini, "Modified secret sharing algorithm for secured medical data sharing in cloud environment", Second International Conference on Science Technology Engineering and Management (ICONSTEM), March 2016.

[8] K. Yang, X. Jia, K. Ren, B. Zhang, R. Xie, "Dac-macs: Effective data access control for multiauthority cloud storage systems," IEEE Transactions on Information Forensics and Security, vol. 8, no. 11, pp. 1790–1801, 2013.

[9] J. Bethencourt, A. Sahai, and B. Waters,"Ciphertext-policy attribute-based encryption," in Security and Privacy, 2007. SP'07. IEEE Symposium. IEEE, 2007, pp. 321–334.

[10] RF wireless Tutorial point, " What is cloud storage | definition of cloud storage ", http://www.rfwireless-world.com/Tutorials/what-is-cloud-storage.html