REVIEW PAPER ON IMPLEMENTATION OF TRIPLE DES USING OTP

[1]Prof. Atiya R. Kazi, [2]Mr. Jawwad A R. Kazi, [3]Ms. Sonali S. Ghogale, [4]Mr. Gunjan N. Kiratkar
[1]Assistant Professor, [2]Student, [3]Student, [4]Student
[1,2,3,4]Department of Information Technology,
[1,2,3,4]Finloex Academy of Management and Technology, Ratnagiri, Maharashtra, India

Abstract: This study has been undertaking to develop a system where we can securely and efficiently share files between admin and client. To maintain security between admin and client, authentication and authorization method is used. Admin can upload file to server which will be store in the database. Generation of OTP will be done by Triple DES. Client will download file using OTP received. After verification of OTP, file will be downloaded to user.

***Index Terms- Triple* DES, OTP, Cloud.**

I. INTRODUCTION

All we are aware of methods of sharing files between admins and clients. Most of these methods includes a system where there is a chance of attack. It has chance of data leak, data misuse, bully, etc. To make system more secure for data sharing it has to be strong to verify its audience. In case of any emergency or critical condition, system should automatically deny to user's request. Our aim is to develop a system where we can share files between admin and client securely and efficiently by maintaining confidentiality and integrity. Here, there will be an authenticated admin which will login to its system. It can upload new file to cloud which will be store in cloud database. After that admin can logout. This system also consists of a client which will login to system. It can search for the file it wants. It will request its cloud to download the file. The cloud will pass this request to admin's cloud. Admin's cloud will generate an OTP (One Time Password) for the file. This OTP will be generated using Triple DES. Triple DES is an encryption algorithm which uses three keys for encryption. As it uses three keys, it provides more and better security as compared to another encryption algorithm. After generating OTP, it will send OTP to client. Client will use that OTP to download the file. After OTP timeout, client have to request again to download the file.

II. LITERATURE SURVEY

2.1) Improved Key Generation Algorithm in Data Encryption Standard (DES)

2.1.1) Features:

This paper presents method of improved key generation in DES. This paper suggests key generation with the help of two arrays of size eight bit. It also suggests that using described method problems of weak and semi weak can be resolved completely. In this improved key method, they have enhanced security because of random number array without compromising the performance. Goal of designing an encryption algorithm is to provide better security and performance from attackers. Size of array can be enlarged to make it difficult to find [1].

2.1.2) Disadvantages:

1) As it uses 16 rounds for encryption and decryption, it makes process slow.
2) As it uses random numbers for key generation, sometimes it may lack confidentiality.
3) It consumes more storage as it generates 16 random keys.

2.2) Cloud Data Sharing Using Cipher Proxy Re-encryption and Ciphertext-Policy Attribute Based Encryption

2.2.1) Features:

The proposed system is ciphertext-policy attribute-based encryption scheme delegating attribute revocation process to cloud server by proxy re-encryption. The proposed scheme does not require secret sharing schemes (LSSS) access structure. Their proposed system is secure against attack by unauthorized users and cloud server. Sharing of the cloud storage has a risk of information leakage caused by service. In order the protect data, the data owner encrypts data shared on the cloud storage so that only authorized users can decrypt the cloud data [2].

2.2.2) Disadvantages:

1) Privacy and confidentiality issue of the cloud.
2) The Cloud provider usually has direct access to data and hence is more likely to steal data for illegal purposes.
3) Since data is stored "in the open", this provides a world of opportunities for malicious users to steal data.

2.3) Enhancing Cloud Data Security with Data Encryption & Tokenization

2.3.1) Features:

Cloud service provider develops necessary infrastructure to facilitate the service and cloud consumer uses the services through network connection. Cloud computing makes outsourcing of computing environment for an individual or an enterprise so that they can avoid committing large capital outlays when purchasing & managing software and hardware as well as dealing with the operational overhead therein. In this paper they have proposed one way to enhance the security of cloud data by combination of tokenization and encryption. Tokenization is a technology which replaces the sensitive data field with a surrogate value called a token. This helps in protecting sensitive data from data breach. The surrogate value i.e. token normally looks like the original data in terms of format, length etc. [3].

2.3.2) Disadvantages:

1) Although cloud computing's benefits are tremendous, data security and protection are one of the major concerns in Cloud Computing.
2) Security can be breached.
3) Only encryption do not provide 100% protection to highly sensitive data.

III. RESEARCH METHODOLOGY

Before designing the system, we have studied the core concepts which will be applicable to our system.

A. Triple DES:

Triple DES is a data encryption standard, which applies the DES cipher algorithm three times. It uses key size of 56 bits and block size of 64 bits. It involves 48 DES-equivalent rounds. A naïve approach to increase strength of a block encryption algorithm with short key length would be to use two keys (k1, k2) instead of one, and encrypt each block twice: $E_{k2}(E_{k1}(\text{plaintext}))$. If the original key length is n bits, one would hope this method provides security equivalent to using key 2n bits long. Triple DES uses a *key bundle* to avoid vulnerable meet-in-the-middle attack. It comprises three DES keys, k1, k2 and k3, each of 56 bits. The encryption algorithm is:

$\text{Ciphertext} = E_{k3}(D_{k2}(E_{k1}(\text{plaintext})))$

i.e., First it will encrypt with key k1, then will decrypt with key k2 and again will encrypt with key k3.

Decryption is the reverse:

$\text{Plaintext} = D_{k1}(E_{k2}(D_{k3}(\text{ciphertext})))$

i.e., First it will decrypt with key k3, then will encrypt with key k2 and again will decrypt with key k1.

B. OTP:

OTP stands for One Time Password. It is also known as one-time pin. It is a password that will be valid for only unique transaction. OTPs avoid a number of shortcomings that are associated with traditional (static) password-based authentication; a number of implementations also incorporate two-factor authentication by ensuring that the one-time password requires to verify one user's security code which it has with the code that user have. The most important advantage that is addressed by OTPs is that, in contrast to static password, they are not vulnerable to replay attacks. This means that an intentional intruder who manages to capture an OTP that were used to log into a product, will not be able to abuse, since it has been expired.

IV. PROPOSED SYSTEM

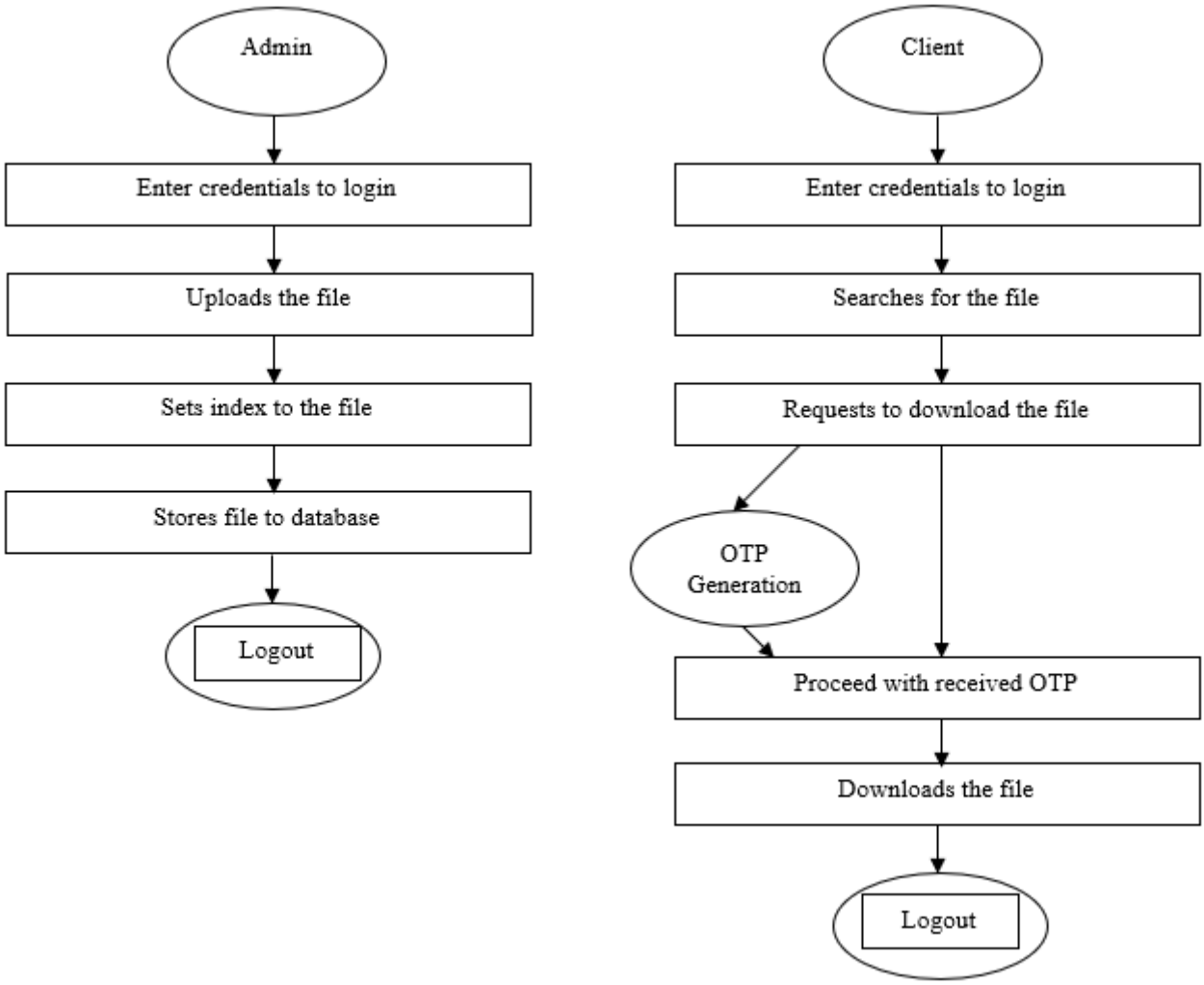

fig. 4.1: proposed system

In our proposed system there will be mainly two parts. One part works as admin part where admin will first login to the system and will upload new file. Admin will then set index to the file. This uploaded file will be stored in the database. Another part is of client which will login to the system using authenticated credentials. Client will then search for the file in the system. Client will request to download that file. This request will be forwarded to OTP generation server where OTP will be generated using Triple DES and generated OTP will be sent to client. Using received OTP client will download the file. Client will then logout from the system.

V. CONCLUSION

Cloud plays important role in today's world. File sharing using cloud is widely acceptable. To provide security to file sharing environment is crucial task. We have proposed a system with the help of which we can share files between admin and client securely and efficiently. Also, we can use encryption algorithm to secure files from being misused. It will provide high security and will also reduce cost spent on maintaining security of the system.

VI. ACKNOWLEDGMENT

We want to give our sincere thanks to Prof. Atiya R. Kazi, for her guidance throughout our research. Her deep knowledge and diligence helped us a lot. We also thank our family and friends for their kind support. Thanks to everyone who helped us directly or indirectly.

REFERENCES

[1] Deepika Rani Bansal, Preeti Thakur. 2016. Improved Key Generation Algorithm in Data Encryption Standard (DES). International Journal of Innovative Research and Advanced Studies, 3(2): 2394–4404.

[2] StanleyRaja S.J., Dr R. Subha. 1997. Cloud Data Sharing Using Cipher Proxy Re-encryption and Ciphertext-Policy Attribute Based Encryption. International Journal of Scientific Research in Science, Engineering and Technology, 2(3): 2394-4099.

[3] Shri. RK. Bigensana Singh, Dr. Lakshmi Prasad Saikia. 2016. Enhancing Cloud Data Security with Data Encryption & Tokenization. International Journal of Current Trends in Engineering & Research, 2(5): 2455-1392.