Final Report for Project

GALGOTIAS
UNIVERSITY

Secure File Storage System on Cloud Using hybrid Cryptography

Project Guide: Mr. Jay Prakash

Project S. No:661

Student Details:

Uttam Kumar(16SCSE101146)

<u>Table of Content</u>:

1. Abstract:

Data stored in the cloud is increasingly gaining popularity for all users including personal, institutions and business purposes. The data is usually highly protected, encrypted and replicated depending on the security and scalability needs. Despite the advances in technology, the practical usefulness and longevity of cloud storage is limited in today's systems. This project provides a solution to the problem of securely storing the client's data by maintaining the confidentiality and integrity of the data within the cloud. This project addresses the problem of ensuring data confidentiality against cloud and against accesses beyond authorized rights. To resolve these issues, we designed a data encryption model that is in charge of storing data in an encrypted format in the cloud. To improve the efficiency of the designed architecture, the service in form of the model designed allows the users to choose the level of security of the data and according to this level different encryption algorithms are used.

2. Introduction:

Cloud computing is defined as for enabling suitable, on demand network entrance to a shared pool of configurable calculating resources. Cloud computing everything is delivered as a service. There are three main service models used in the cloud namely.

- Platform as a Service (P a a S)
- Software as a Service (S a a S)
- Infrastructure as a service (I a a S)

(i) Security Issues: Cloud as a method of providing computing resources has many challenges based on design issues which affect the efficiency, security and performance of the entire system, these challenges could be:

a) **Data Storage:** Cloud storage providers manage the data in multiple copies across many independent locations

b) **Cloud Confidentiality:** Confidentiality can be defined as the sensitive data not being disclosed to unauthorized process, devices and person. A cloud service provider knows where the user's public or private data is located and who can/cannot access the data.

c) **Data Integrity:** Data Integrity is defined as the rightness of data stored in the cloud. The alterations between two updates of a record violate the data integrity.

d) **Data Security:** In the traditional file systems data was stored within boundaries, but cloud data is stored outside the boundaries of an organization, say, and third party storage using strong encryption techniques.

(ii) Security Solutions: To resolve the above listed challenges, cryptography can provide solutions such as reassuring the receiver/recipient that the message received has not been tampered with or altered – this can be defined as Integrity Checking. This can be achieved by generating a legitimate source and authentication. Securing the database can be a means of securing the cloud.

Cryptography techniques convert original data into Cipher text. So only legitimate users with the right key can access data from the cloud storage server. The main aim of cryptography is to keep the security of the data from hackers, online/software crackers, and any third party users. Non-legitimate user access to information results in loss of confidentiality. Security has the characteristics to block or stop this kind of unauthorized access or any other kind of malicious attacks on the data here by securing the users' trust. In the cloud computing environment, security is deemed to be a crucial aspect due to the significance of information stored on the cloud and the different services provided to the users. This data can be confidential and extremely sensitive. Hence, the data management and security should be completely reliable. It is necessary that the data in the cloud is protected from malicious attacks.

This can be achieved using different encryption/decryption algorithms which are classified as follows:

a) **Symmetric key:** this refers to encryption methods in which both the sender and receiver share the same key. Examples of such algorithms include:3DES, DES,BLOWFISH, and AES etc.

b) **Asymmetric key:** this is a public key cryptography that entails using different keys for encryption and decryption; this means that there is a key for private and another different one for public. Therefore, the private key is kept by the receiver and the public is kept by anyone(public)

Examples of such algorithms includes: RSA, DSA etc.

Hash Algorithms: this is where the input data(message) is recreated from the hash value(message digest/digest) Examples of such examples include:MD5, SHA, MD2, MD4, MD6.SHA-256, SHA-512, SHA-1, Whirlpool etc.

To ensure the security of data in the cloud we propose an effective way with the features of CIA (Confidentiality, Integrity and Authentication).

(iii)Scope: The future scope of cloud security is as follows:

a) **Cost-benefit analysis:** The business case for cloud computing requires a clear understanding of costs as compared to an organization's in-house solution. The key measure is that cloud must reduce capital and operational expenses without sacrificing user functionality, such as availability. The best delivery model for cloud functionality is a hardware-agnostic approach that embraces the commodity architectures in use by the world’s leading Internet and SaaS providers. This can be achieved through low-cost commodity servers and disks coupled with intelligent management software, providing true cloud-based economies of scale and efficiency.

b) **Robust security:** When you move to the cloud, you’re entrusting the organization’s intellectual property to a third party. Do their security standards meet the needs of your business? Even the smallest entry point can create an opening for unauthorized access and theft. Authentication and access controls are even more critical in a public cloud where cluster attacks aimed at a hypervisor can compromise multiple customers. Ideally, the cloud provider should offer a broad set of security solutions enabling an information-centric approach to securing critical interfaces – between services and end users, private and public services, as well as virtual and physical cloud infrastructures.

c) **Data availability:** As cloud places new demands on storage infrastructure, data availability, integrity, and confidentiality must be guaranteed. Often, these provisions come with vendors who offer massive scalability and elasticity in their clouds. To make this approach manageable for customers, cloud vendors must offer tools that provide visibility and control across heterogeneous storage platforms. The final test for cloud storage is interoperability with virtual infrastructures. This allows service providers to standardize on a single approach to data protection, de-duplication, assured availability, and disaster recovery across physical and virtual cloud server environments, including VMWare, MS Hyper-V and a variety of UNIX virtualization platforms.

d) **Regulatory compliance:** Cloud computing brings a host of new governance considerations. Organizations must evaluate the ability of the cloud provider to address the company’s own regulations, national and worldwide rules for conducting business in different regions, and customer needs. For example, many healthcare customers will require SOX and HIPAA compliance while financial customers must comply with Gramm-Leahy-Biley and Red Flags.

3. Existing System:

[1]Cloud supports large data storage, these results to lots of pressures in the cloud computing which are avoided in the cloud. The main threats in the cloud are confidentiality and data integrity in the cloud data storages. Calce et al introduces the use of a single box for putting everything in the cloud computing model, this only makes it easy for hackers thus lacking SECURITY.

[2]A. K. Shahade and V. S. Mahale [1] in their research introduced a Hybrid encryption algorithm which was a combination of RSA algorithm and AES algorithm. In their system, the user creates and stores the RSA private key with himself and also create an RSA public key while uploading the data. In the cloud, the server calls the RSA and AES algorithm for encryption of the file and then properly store the file on the server.

[3]Sravan Kumar et al proposed a method of proof by adopting the use of Meta Data. This data is created using randomly selected bits from original file and is appended in an encrypted form to be stored on the cloud, therefore whenever a person wants to check for integrity, he/she throws a challenge by specifying block number and its corresponding Meta Data and finally decrypt’s for proof of Correctness.

[4]Dalia et al also implemented a mechanism in which integrity is checked at 2 sides by cloud server (for the attacker who's at the inside)and by TPA(for the attacker at the outside) using a digital signature with MD5.

[5]Paresh D. Sharma et al. proposes the use of symmetric key technique using AES algorithm for stored data as well as for the data moving within the cloud or for the outside service provider, then this service provider cannot use these data if it didn't get the key cryptography. So the service provider in the cloud should use the key to get and use their data.

[6]S. Hesham in her research proposed an algorithm that increases the efficiency of the Advanced Encryption Algorithm. The proposed method reduces the critical path delay of the original algorithm. Compared to the original AES encryption/decryption algorithm the proposed algorithm provides an efficiency improvement of 61% and 29% respectively.

4. Proposed Model:

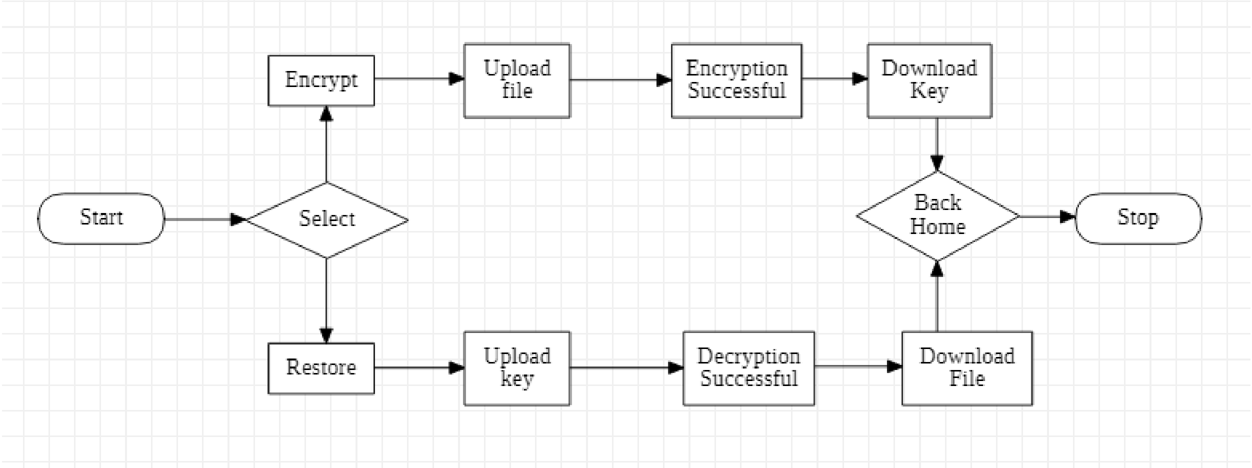


Fig. 1 Flow of system

The encryption and decryption of data is done by AES algorithm. The algorithm increases the run time for both encryption/decryption thus increasing the total time required for both processing time. AES is an iterative rather than Feistel cipher. It is based on ‘substitution–permutation network’. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix –

Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

The schematic of AES structure is given in the following Fig.2 illustration –

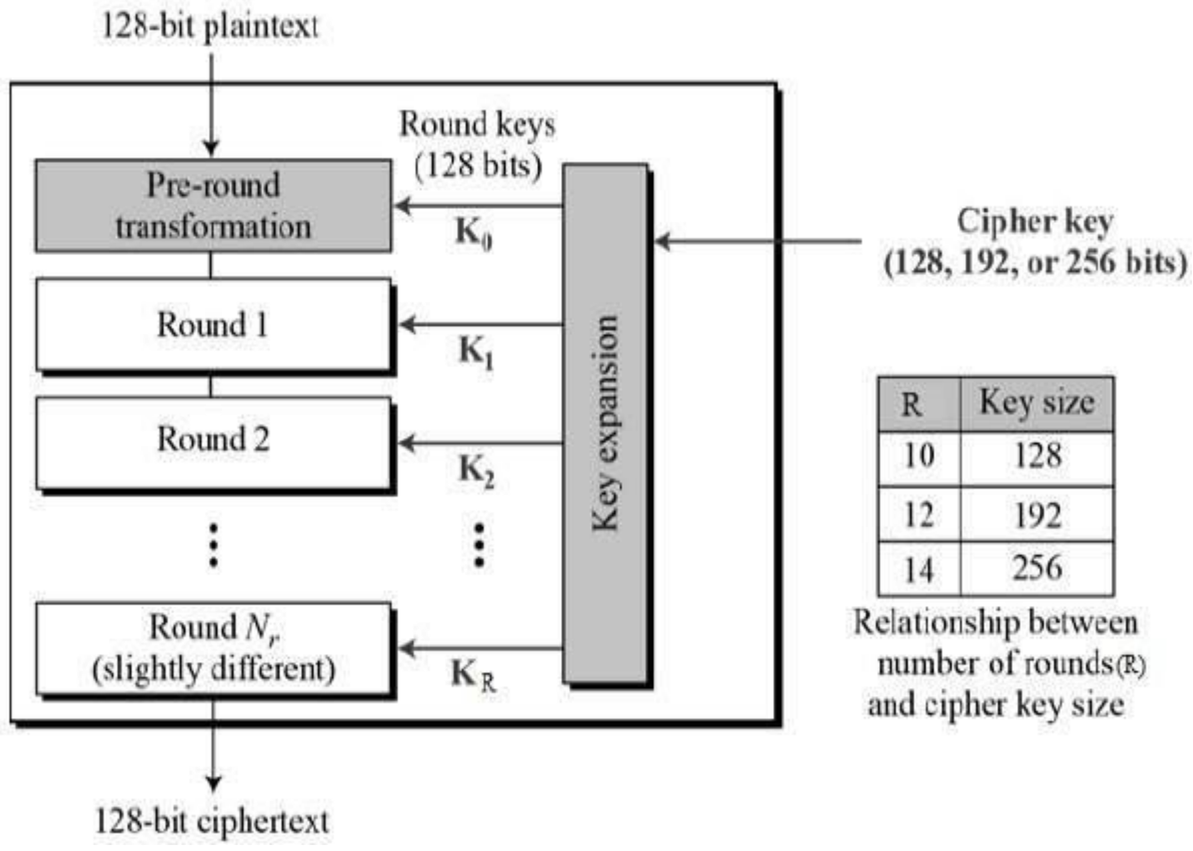


Fig. 2 AES Structure

5. Implementation or Architecture Diagrams:

a) Flowchart for Encryption:

Flowchart for encryption algorithm is as follows as in Fig.3:

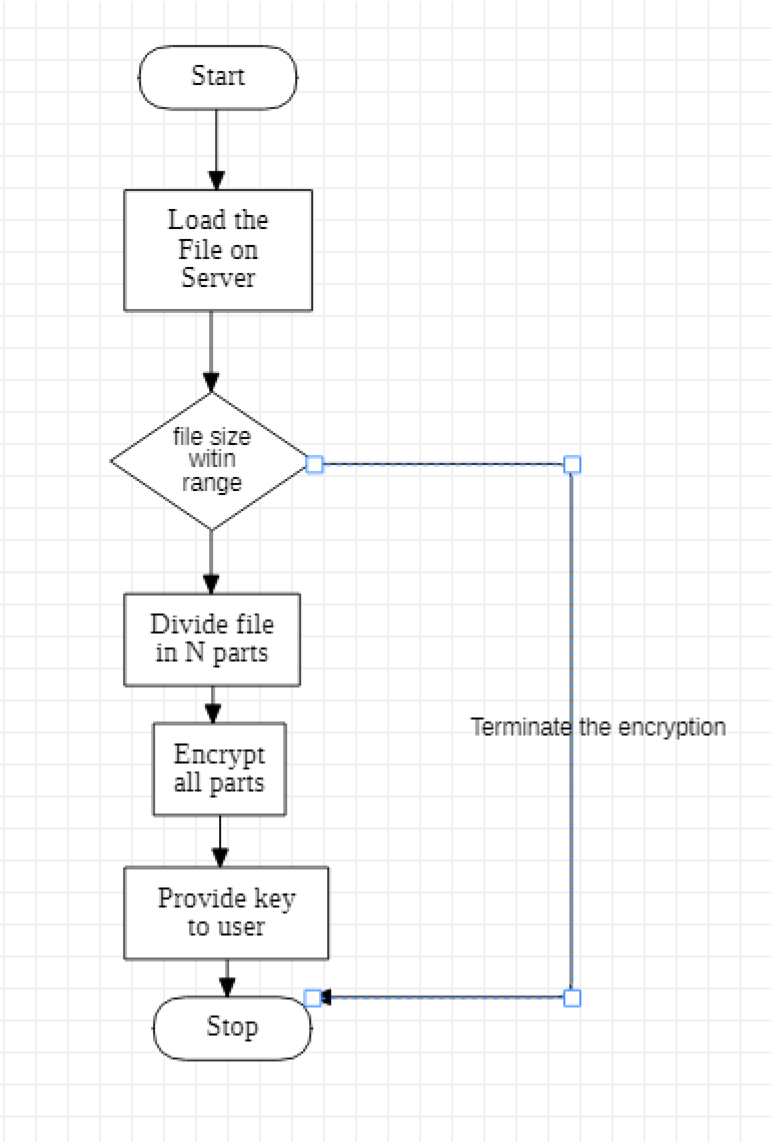


Fig.3 Encryption

b) Flowchart for Decryption:

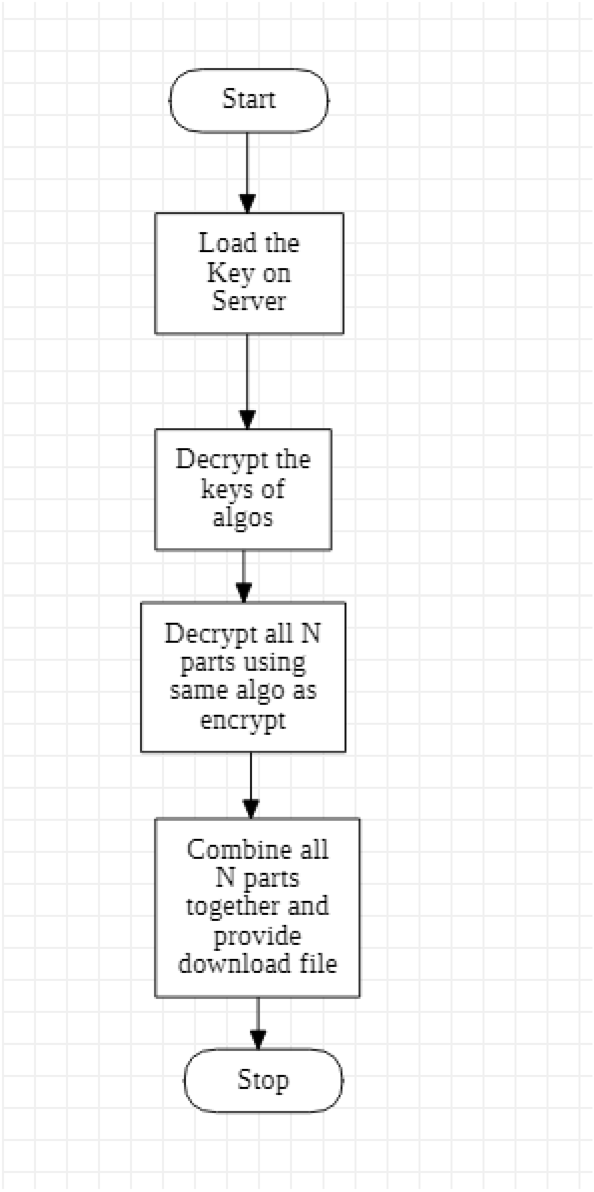

Start
Load the Key on Server
Decrypt the keys of algos
Decrypt all N parts using same algo as encrypt
Combine all N parts together and provide download file
Stop

Fig.4 Decryption

c) Algorithm for Encryption:

```
def Algo1(data, key):

        f = Fernet(key)

        target_file = open("raw_data/store_in_me.enc","wb")
```

```
    secret_data = f.encrypt(data)
    target_file.write(secret_data)
    target_file.close()


def Algo1_extented(filename, key1, key2):
    f = MultiFernet([Fernet(key1),Fernet(key2)])
    source_filename = 'files/' + filename
    target_filename = 'encrypted/' + filename
    file = open(source_filename,'rb')
    target_file = open(target_filename,'wb')
    raw = ""
    for line in file:
        raw = raw + line
    secret_data = f.encrypt(raw)
    target_file.write(secret_data)
    file.close()
    target_file.close()


def Algo2(filename, key, nonce):
    aad = "authenticated but unencrypted data"
    chacha = ChaCha20Poly1305(key)
    source_filename = 'files/' + filename
    target_filename = 'encrypted/' + filename
    file = open(source_filename,'rb')
    target_file = open(target_filename,'wb')
    raw = ""
    for line in file:
        raw = raw + line
    secret_data = chacha.encrypt(nonce, raw, aad)
    target_file.write(secret_data)
    file.close()
    target_file.close()
```

```
def Algo3(filename, key, nonce):

    aad = "authenticated but unencrypted data"

    aesgcm = AESGCM(key)

    source_filename = 'files/' + filename

    target_filename = 'encrypted/' + filename

    file = open(source_filename,'rb')

    target_file = open(target_filename,'wb')

    raw = ""

    for line in file:

        raw = raw + line

    secret_data = aesgcm.encrypt(nonce, raw, aad)

    target_file.write(secret_data)

    file.close()

    target_file.close()
```

```
def Algo4(filename, key, nonce):
    aad = "authenticated but unencrypted data"
    aesccm = AESCCM(key)
    source_filename = 'files/' + filename
    target_filename = 'encrypted/' + filename
    file = open(source_filename,'rb')
    target_file = open(target_filename,'wb')
    raw = ""
    for line in file:
        raw = raw + line
    secret_data = aesccm.encrypt(nonce, raw, aad)
    target_file.write(secret_data)
    file.close()
    target_file.close()
```

d) Algorithm for Decryption:

```
def Algo1(key):
    f = Fernet(key)

    target_file = open("raw_data/store_in_me.enc","rb")

    secret_data = ""

    for line in target_file:

        secret_data = secret_data + line

    data = f.decrypt(secret_data)

    target_file.close()

    return data



def Algo1_extented(filename, key1, key2):

    f = MultiFernet([Fernet(key1),Fernet(key2)])

    source_filename = 'encrypted/' + filename

    target_filename = 'files/' + filename

    file = open(source_filename,'rb')

    target_file = open(target_filename,'wb')

    raw = ""

    for line in file:
        raw = raw + line
    secret_data = f.decrypt(line)
    target_file.write(secret_data)
    file.close()
    target_file.close()


def Algo2(filename, key, nonce):
    aad = "authenticated but unencrypted data"
    chacha = ChaCha20Poly1305(key)
    source_filename = 'encrypted/' + filename
    target_filename = 'files/' + filename
    file = open(source_filename,'rb')
    target_file = open(target_filename,'wb')
    raw = ""
    for line in file:
        raw = raw + line
    secret_data = chacha.decrypt(nonce, raw, aad)
    target_file.write(secret_data)

    file.close()

    target_file.close()



def Algo3(filename, key, nonce):

    aad = "authenticated but unencrypted data"

    aesgcm = AESGCM(key)

    source_filename = 'encrypted/' + filename

    target_filename = 'files/' + filename

    file = open(source_filename,'rb')

    target_file = open(target_filename,'wb')

    raw = ""

    for line in file:

        raw = raw + line

    secret_data = aesgcm.decrypt(nonce, raw, aad)

    target_file.write(secret_data)

    file.close()
```

```
        target_file.close()



    def Algo4(filename, key, nonce):

        aad = "authenticated but unencrypted data"

        aesccm = AESCCM(key)

        source_filename = 'encrypted/' + filename

        target_filename = 'files/' + filename

        file = open(source_filename,'rb')

        target_file = open(target_filename,'wb')

        raw = ""

        for line in file:

            raw = raw + line

        secret_data = aesccm.decrypt(nonce, raw, aad)

        target_file.write(secret_data)

        file.close()

        target_file.close()
```

6. Results and Discussions:

An application has been designed and implemented in java language on the same network to achieve the functionalities of the client and the server. We have it that the cloud and the server are on the same network so that they can be in opposition to communication; we therefore put them on the same ip address for them to communicate without any hindrances

. This makes the client and the server which are on the same domain be subjected to the same parameters. Based on the experiment we have computed the parameters value for AES, Blowfish and the combination or the hybrid system(AES +Blowfish) for the same file size.

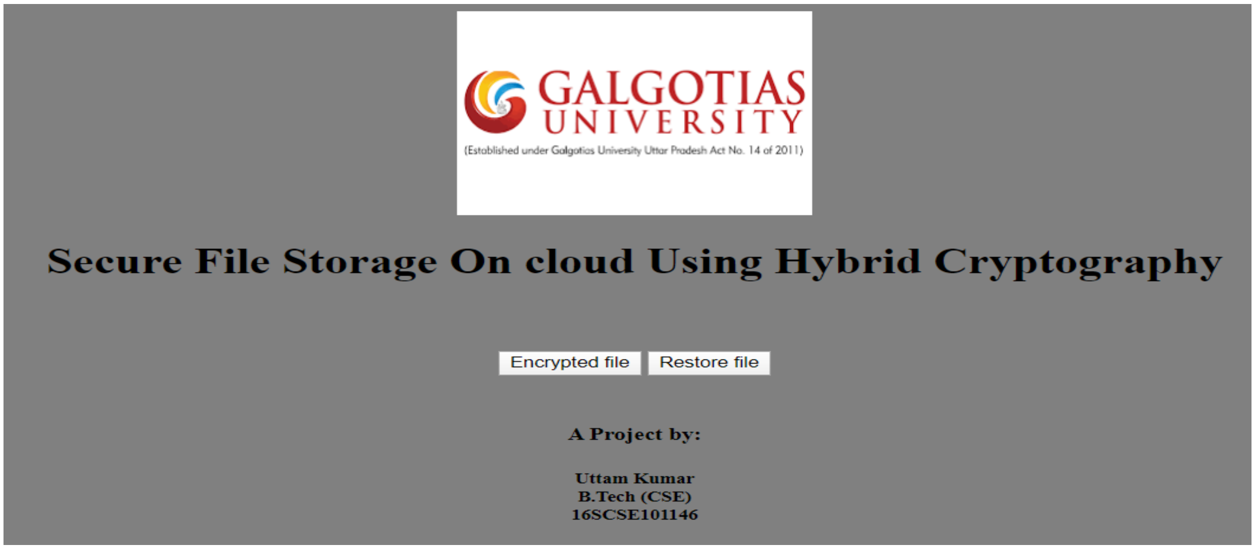
Results show that AES is the best algorithm of symmetric encryption technology. AES algorithm is more secure than the Blowfish algorithm but on the other hand Blowfish is more secure than other algorithms. Blowfish runs faster than other symmetric algorithms.

AES is the symmetrical based encryption standard by NIST. The hybrid algorithm is more secure since it has the characteristics of both algorithms and makes it more vulnerable to threats.

a) Welcome Screen:

First of all we have to run ‘app.py’ in command prompt using python2. When the ‘app.py’ run, it will create a cloud server of following address “http://127.0.0.1:8000/” copy the above link and open it in any browser using internet.

After accessing following web address the below page as in Fig.3 will open in the web browser:



GALGOTIAS UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)
Encrypted file
Restore file

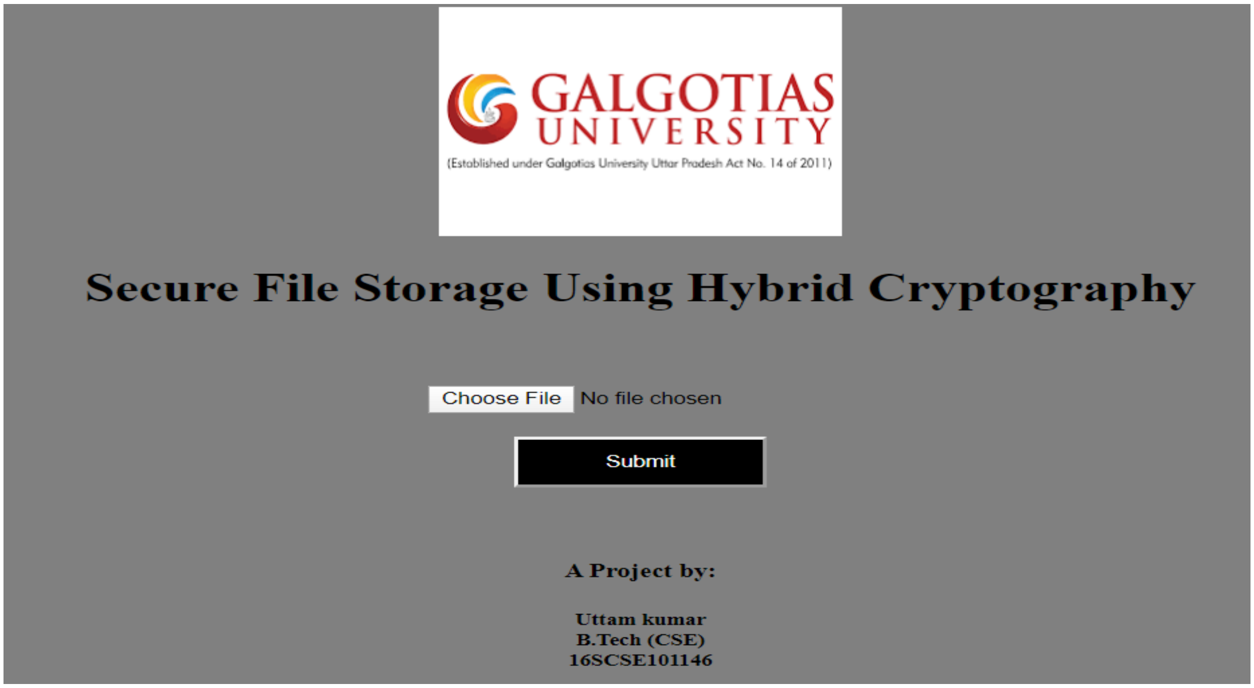
Secure File Storage On cloud Using Hybrid Cryptography

A Project by:

Uttam Kumar
B.Tech (CSE)
16SCSE101146

Fig.5 Welcome Window

b) Select File to Encrypt:



GALGOTIAS UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)
Choose File
No file chosen
Submit

Secure File Storage Using Hybrid Cryptography

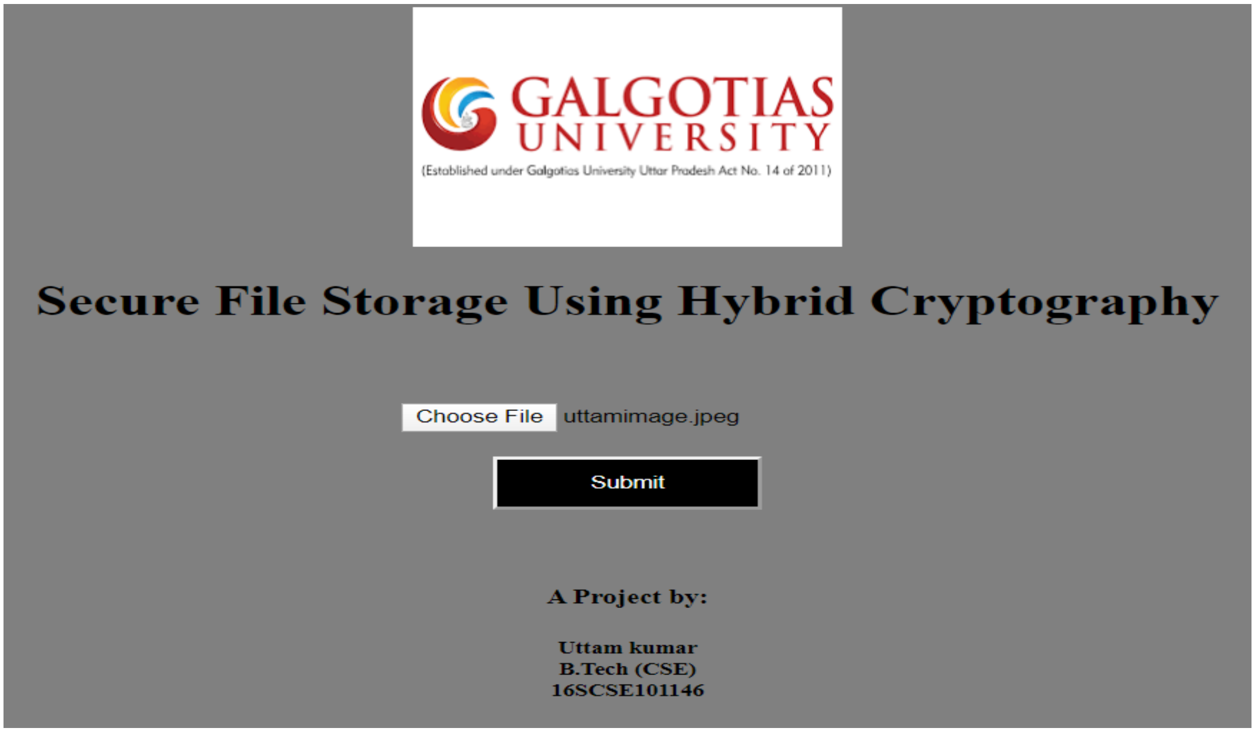
A Project by:

Uttam kumar
B.Tech (CSE)
16SCSE101146

Fig.6 Encryption window

c) Selected file for Encryption:

When we will select file it will show as in Fig.7 and the selected file is in the Fig.8. And when we click on “submit” button it will move to next window and will show “File Encrypted Successfully” as in the Fig.9.



GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Secure File Storage Using Hybrid Cryptography

A Project by:

Uttam kumar
B.Tech (CSE)
16SCSE101146

Fig.7 Select file for encryption

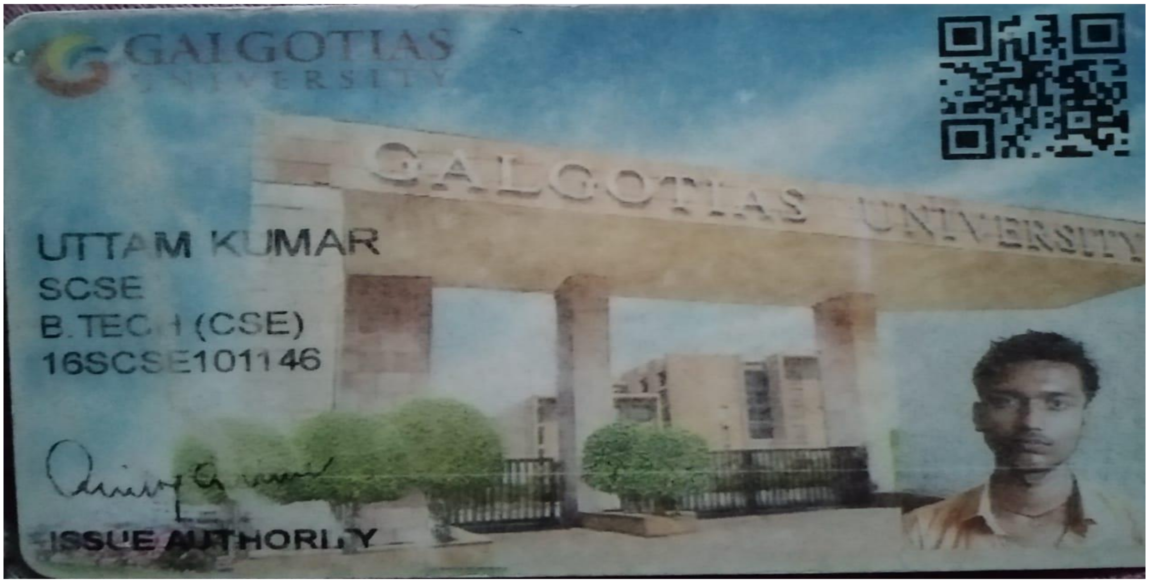
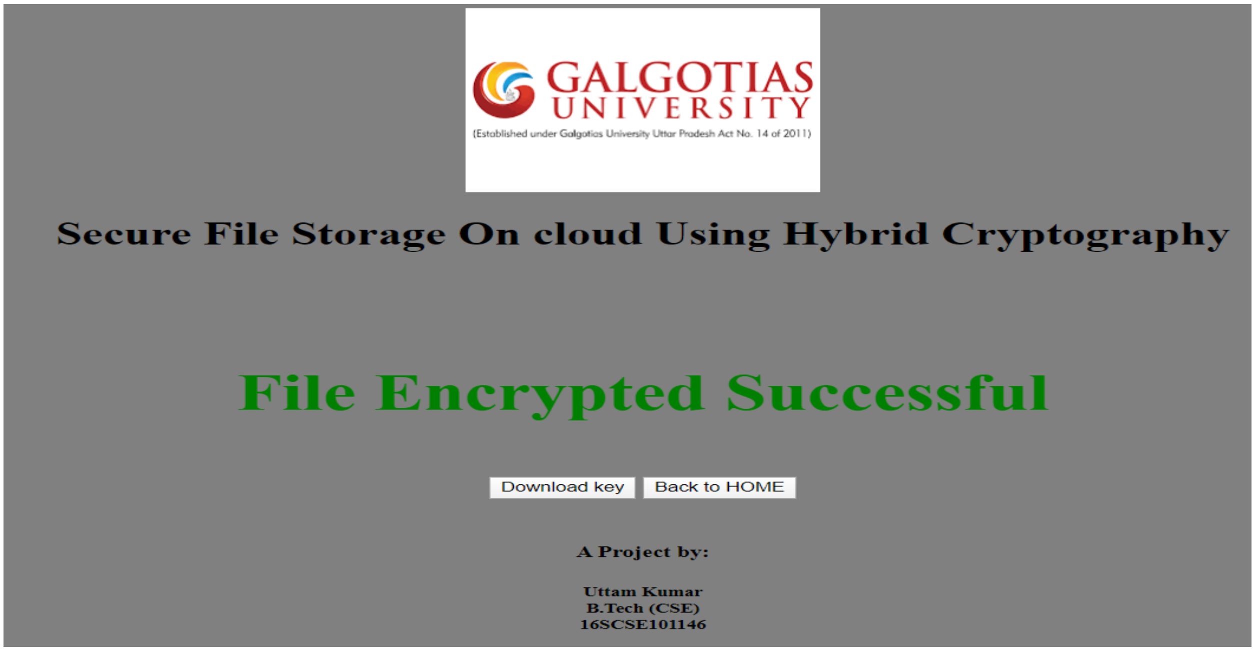


Fig.8 Selected File for Encryption



GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Secure File Storage On cloud Using Hybrid Cryptography

File Encrypted Successful

A Project by:

Uttam Kumar
B.Tech (CSE)
16SCSE101146

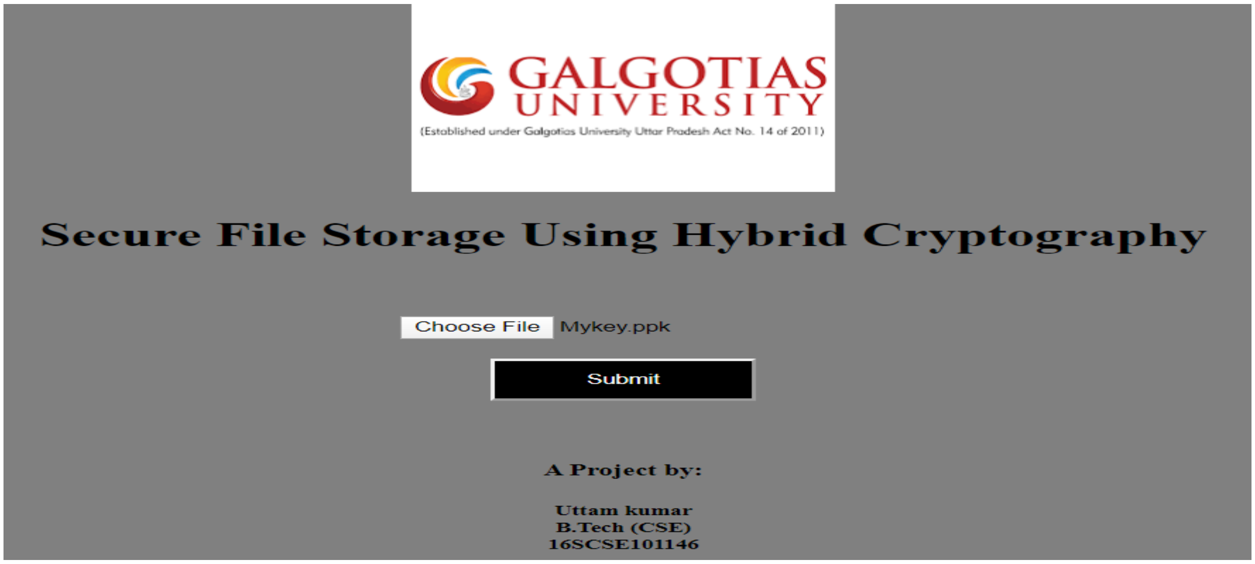
Fig.9 File Encryption Successful

After successfully encryption we have to download the key for the encrypted file. The key will download in a file format with '.pem' extension. The key is mandatory to restore the encrypted file/data.

d) **Decryption:**

For decryption we should have an encryption key which will be provided after encryption. The key is the only way to access the encrypted data.

We have to select the key from memory location in computer as shown in Fig.10:


GALGOTIAS UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)
Choose File
Submit

Secure File Storage Using Hybrid Cryptography

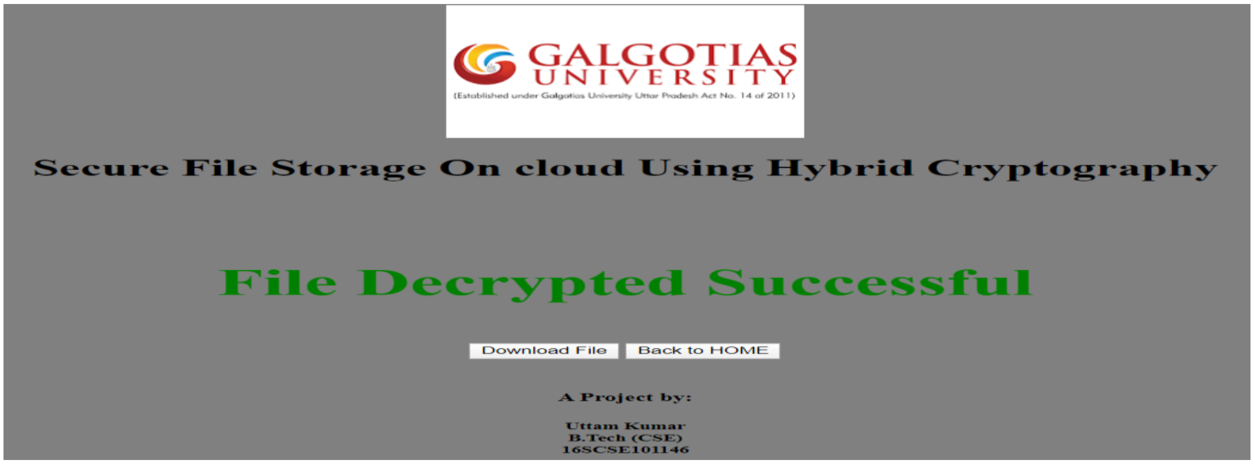
Mykey.ppk

A Project by:

Uttam kumar
B.Tech (CSE)
16SCSE101146

Fig.10 Select Downloaded Key

e) It will decrypt the data using the key and will show a message on successful decryption “File Decrypted Successfully” as in the Fig.11. And give you option to download the decrypted file. The decrypted file is as in the Fig.11 which is as same as in the Fig.8.


GALGOTIAS UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)
Download File
Back to HOME

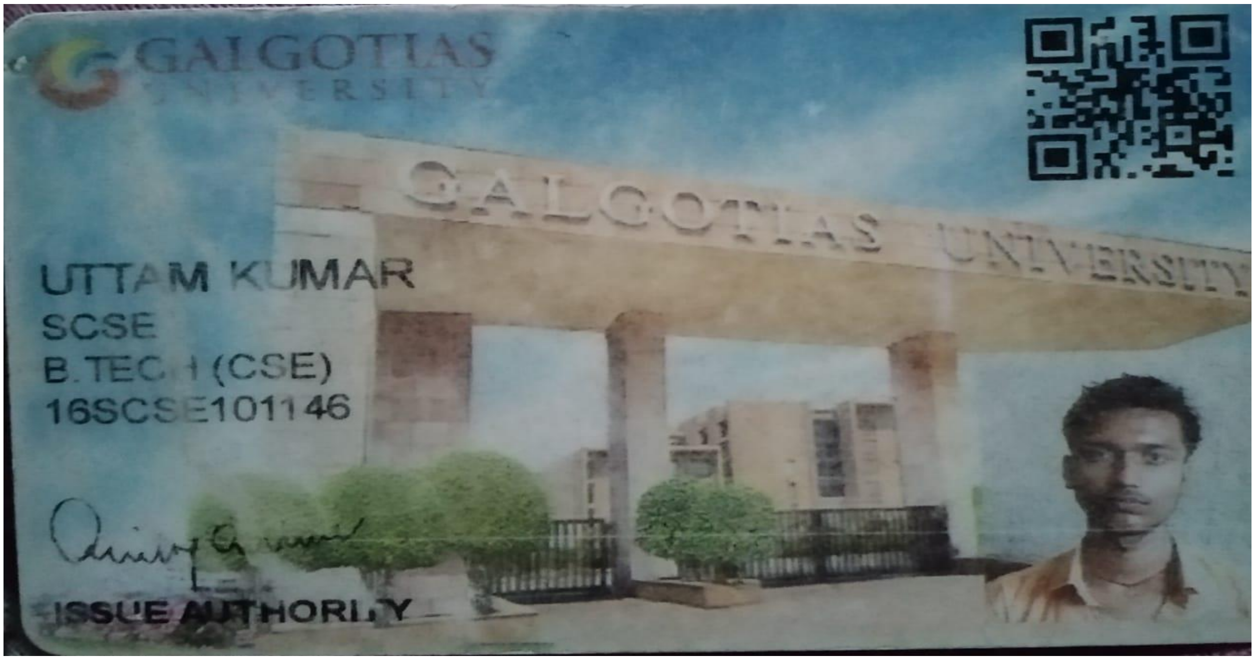
Secure File Storage On cloud Using Hybrid Cryptography

File Decrypted Successful

A Project by:

Uttam Kumar
B.Tech (CSE)
16SCSE101146

Fig.11 Decryption Successful



GALGOTIAS UNIVERSITY
GALGOTIAS UNIVERSITY

UTTAM KUMAR
SCSE
B.TECH (CSE)
16SCSE101146

ISSUE AUTHORITY

Fig.12 Downloaded file

7. Conclusions and Future Enhancement:

When the clients store data in the cloud, there’s always an issue whether or not cloud service provider stores the data securely. Security as earlier discussed is the main challenge faced while storing data in the cloud, the proposed system provides security for the data stored in the cloud computing model through the help of AES and Blowfish algorithms.

Results show that AES is the best symmetric encryption algorithm, it’s more secure than Blowfish though compared to other algorithms Blowfish is by far the best. Blowfish gives the highest throughput as compared to AES.

The hybrid of AES and Blowfish gives the properties of both algorithms thus making the formed hybrid algorithm much stronger to threats. This makes the formed hybrid system secure by increasingly adding the complexity functionalities.

The future scope of this work can be extended by:

- Performing the same experiments using audio and video as well.

- Compression algorithm can be performed for faster encryption.

- Performing the same experiments using some locking techniques for security mechanism

8. References

1) Retrievability for Large Files," In CCS '07: Proceedings ofthe 14th ACM Conference on Computer and Communications Security. New York, NY, USA: ACM, 584–597.

2) A.Nadeem and M.Y Javed.,"A PerformanceComparison of Data Encryption Algorithms,"IEEE Information and Communication Technologies, 2005.ICICT 2005.First International Conference, 2006, pp. 84-89.

3) J.Daeman and V.Rijmen,"AES submission document on Rijndael, Ver2", September 1999.

4) "Announcing the AdvanceEncryption standard", FIPS Publication, 2001

5) Gurpreet Kaur And Manish Mahajan (2013), ―Analyzing Data Security for Cloud Computing Using Cryptography Algorithms‖, International Journal of Engineering Research and Application, Vol.-3,782-786

6) H. Shacham And B. Waters (Dec.2008), ―Compact Proofs of Retrievability, ‖ In Proceedings. Of Asia Crypt‖08.

7) Kamak Ebadi, Victor Pena Etc.‖High Performance Implementation and Evaluation of Blowfish Cryptographic Algorithm on Single-Chip Cloud Computer: A Pipelined Approach ‖.

8) K.Govinda, E.Sathiyamoorthy (2012), ―Data Auditing in Cloud Environment using Message Authentication Code‖, International Conference on Emerging Trends on Advanced Engineering Research (ICETT).

9) Ms.Payal P.Kilor and Prof. Vijay B.Gadicha (2014) ―Data Integrity Proofs in Document Management System under Cloud with Multiple Storage‖, International Journal of Engineering &Computer Science, and vol.3.

10) Omer K.Jasim et. al.(2013) ―Efficiency of modern encryption algorithms in cloud computing‖, International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), vol. 2.

11) Paresh D.Sharma, Prof. Hitesh Gupta(February 2014) ―An Implementation for Conserving Privacy based on Encryption Process to Secured Cloud Computing Environment‖ IJESRT Sharma, 3(2).

12) P.Metri and G.Sarote (2013), ―Privacy Issues and Challenges in Cloud Computing‖, International Journal of Advanced Engineering Science and Technologies, vol.no.-5, 1-6.

13) R.Buyya, C.S.Yeo, S.Venugopal (2009), ―Cloud Computing and emerging IT platforms: vision, hype and reality for delivering computing" as 5th utility, Future Generartion Computer System, 25: 599-616.

14) Schneier, Bruce. (2014) "Cryptanalysis of MD5 and SHA: Time for a New Standard". Computerworld. Retrieved.

15) ShivShakti etc. (January-Febuary-2013).‖Encryption using different techniques:A Review international journal in Multidisciplinary and academic research (SSIJMAR) vol.2 No.1 (ISSN 2278-5973).

16) Sravan Kumar and Ashutosh Saxena (2011), ―Data Integrity Proofs in Cloud Storage", 978-1-4244-8953- 4/11/$26.00© IEEE.

17) S.Subashini and V.Kavitha (2011), ―A Survey on security issues in service delivery models of cloud computing". Journal of Network and Computer Applications 34, 1-11.

18) Zaigham Mahmood(2011), ―Data Location and Security Issues in Cloud Computing‖, Proceedings of International Conference on Emerging Intelligent Data and Web Technologies.