GALGOTIAS UNIVERSITY

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

SECURE FILE STORAGE ON CLOUD USING CRYPTOGRAPHY

A Report for the Evaluation 3 of Project 3

Submitted by

ARUN PRATAP SINGH

(1613101184)

In partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Under the supervision of

Mr. HIMANSHU PUNDIR

Professor

APRIL/MAY

GALGOTIAS
UNIVERSITY

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report **“SECURE FILE STORAGE ON CLOUD USING CRYPTOGRAPHY**” is the bonafide work of **“ARUN PRATAP SINGH(1613101184)”** who carried out the project work under my supervision.

SIGNATURE OF HEAD

DR.MUNISH SHABARWAL

PhD (Management), PhD (CS)

Professor & Dean

SIGNATURE OF SUPERVISOR

MR.HIMANSHU PUNDIR

Professor

School of Computer Science & Engineering

TABLE OF CONTENT

1. ABSTRACT

The proposed model is liable to meet the required security needs of data center of cloud. Blowfish used for the encryption of file slices takes minimum time and has maximum throughput for encryption and decryption from other symmetric algorithms. The idea of splitting and merging adds on to meet the principle of data security. The hybrid approach when deployed in cloud environment makes the remote server more secure and thus, helps the cloud providers to fetch more trust of their users. For data security and privacy protection issues, the fundamental challenge of separation of sensitive data and access control is fulfilled. Cryptography technique translates original data into unreadable form. Cryptography technique is divided into symmetric key cryptography and public key cryptography. This technique uses keys for translate data into unreadable form. So only authorized person can access data from cloud server. Cipher text data is visible for all people. we expect to safely store data into the cloud, by parting information into a few lumps and putting away pieces of it on cloud in a way that jelly information confidentiality, uprightness and guarantees accessibility. The quickly expanded utilization of distributed computing in the numerous association and IT businesses gives new programming minimal effort. Distributed computing is useful as far as ease and openness of information. Distributed computing gives parcel of advantages with ease and of information availability through Internet. Guaranteeing the security of distributed computing is a main consideration in the distributed computing condition, as clients regularly store delicate data with distributed storage suppliers, yet these suppliers might be untrusted.

So sharing information in secure way while safeguarding information from an untrusted cloud is as yet a difficult issue. Our methodology guarantees the security and protection of customer touchy data by putting away information across single cloud, utilizing AES, DES and RC2 calculation.

2. INTRODUCTION

Cryptography is the shielding method of information from the unapproved party by changing over into the non-comprehensible structure. The principle reason for cryptography is keeping up the security of the information from outsider. There are following two kinds of calculations, for example, (I) symmetric key based calculation, here and there known as traditional key calculation and (ii) topsy-turvy key based calculation, otherwise called open key calculation. Symmetric calculation can be additionally isolated into two kinds.

In the distributed computing condition, security is considered to be a urgent perspective because of the centrality of data put away in the cloud. The information can be private and incredibly touchy. Subsequently, the information the executives ought to be totally dependable. It is essential that the data in the cloud is shielded from malignant assaults. Security acquires worries for secrecy, uprightness and accessibility of information. Unapproved access to data brings about loss of information privacy. Information honesty and accessibility endures because of disappointment of cloud administrations. Security has the qualities of a supplement to unwavering quality.

The utility of this cloud and its administrations are not confined to an area or any premises. All the clients, for example, head, instructors and understudies are permitted to utilize this information at whatever point required The cloud can be gotten to through web from anyplace. The clients need to login to the cloud and give subtleties to get to the information from database. The cloud will likewise give security to all the information put away at our server.

Advantages

- The stored image file is completely secured, as the file is being encrypted not by just using one but three encryption algorithm which are AES, DES and RC6.
- The key is also safe as it embeds the key in image using LSB.
- The system is very secure and robust in nature.
- Data is kept secured on cloud server which avoids unauthorized access.

Disadvantages:

- Requires an active internet connection to connect with cloud server.

3. PROBLEM STATEMENT

Client's stores information at cloud specialist organizations is powerless against different dangers. In our work, we consider four kinds of risk models. First is the single purpose of disappointment, which will influence the information accessibility that could happen if a server at the cloud specialist organization fizzled or smashed, which makes it harder for the client to recover his put away information from the server. Accessibility of information is likewise a significant issue which could be influenced, if the cloud specialist organization (CSP) comes up short on administration.

Our subsequent danger is information trustworthiness. Uprightness is a degree certainty that the information in the cloud is what should be there, and is ensured against coincidental or purposeful adjustment without approval. Such concerns are not any more gainful issues; thusly, a cloud administration client can not so much depend upon a cloud specialist organization to guarantee the capacity of his essential information. Security is a vital help for wired system just as remote system correspondence to improve what was offered in cloud .Simply putting away the data on mists tackles the issue isn't about information accessibility, however about security. The solid purpose of this technique is that the mystery key must be consolidated by remaking.

The majority of the organizations that have kept away from receiving the cloud have done as such in the dread of having their information spilled. This accomplishment comes from the way that the cloud is a multi-client condition, wherein all the assets are shared. It is likewise an outsider help, which implies that information is possibly in danger of being seen or misused by the supplier. It is just human instinct to question the abilities of an outsider, which appears to be a much greater hazard with regards to organizations and delicate business information. There are additionally various outer dangers that can prompt information spillage, including malevolent hacks of cloud suppliers or bargains of cloud client accounts. The best procedure is to rely upon record encryption and more grounded passwords, rather than the cloud specialist co-op themselves.

Provisioning Multi-Cloud infrastructure for secured data storage This problem is divided into following sub-problems: • To Design an Efficient Multi-Cloud Data Storage Solution for Enterprise To explore the possibility of designing an efficient hybrid-Multi-Cloud framework for optimal utilization of storage space, improved data availability, promote in-premise processing of data by privacy preserved data distribution on multiple CSPs.

- To Generate Master SLA for Multi-Cloud Solutions To generate master SLA for storage and retrieval of different types of data like archive, medical records and Virtual computing data being stored on various CSPs.

- To Develop a Mathematical Model for Ranking CSP Combinations To identify the attributes required to draft the master service level agreement for Multi-Cloud framework to store the data and formulate the aggregate functions for each attribute. To apply a multi criteria decision making algorithm for selection of various CSP combinations by using a technique that 9 appropriately provides varying weights to different SLA parameters based on user requirements, expert ratings and past performance of the CSPs.

- To Provide Security for Data In-flight To enhance the data confidentiality by designing a light weight encryption algorithm and sending the encrypted data on to the cloud

. • To Provide Access Control Rights and enhance public verifiability To explore the possibility of enhancing the public verifiability scheme of the data stored on multiple cloud service provider and providing access control for the files stored on multiple clouds. This research work is carried out in three phases. In the first phase a hybrid Multicloud framework for secured file storage technology using open Zeta byte File System (ZFS) is designed for

(1) Storing data on multi-clouds in file format on the CSPs selected by the clients based on their requirements

(2) Enhancing the secrecy of the data stored by the proposed model to improve confidentiality, integrity, availability, and resolve vendor-lock-in issues.

(3) Providing an economical model based not only on the selection of CSPs but also on the amount of data being stored/transmitted on to CSPs by using the compression and de-duplication options .

(4) Offering computation at the enterprise in the private cloud to enhance security to avoid cross VM attacks. Various tests have been carried out using File IO tools [35]. Second phase of research presents a framework for drafting a Master SLA Generation for Multi-Cloud Solutions Using Erasure Coding Technique which

(a) Identifies attributes related to implementation method of multi-cloud environment and need to be part of master SLA.

(b) Establishes functions to calculate aggregate values for master SLA parameters in multi-cloud environment by considering implementation details of multi-cloud.

(c) Considers customer requirement, expert rating of measurement methods of SLA attributes and pricing models adopted by service provider, and past performance of individual service providers while ranking alternative combinations of service providers.

(d) Considers different normalization 10 methods in MCDM technique while ranking possible combinations of CSPs to form multi-cloud.

(e) Evaluation of the framework. In the third phase of research we present a solution on two major issues that are public verifiability and data accessibility simultaneously. An encryption algorithm is introduced as a part of a protocol that provides four layered security. Every layer adds to the time needed to break the cipher and gets the genuine information.

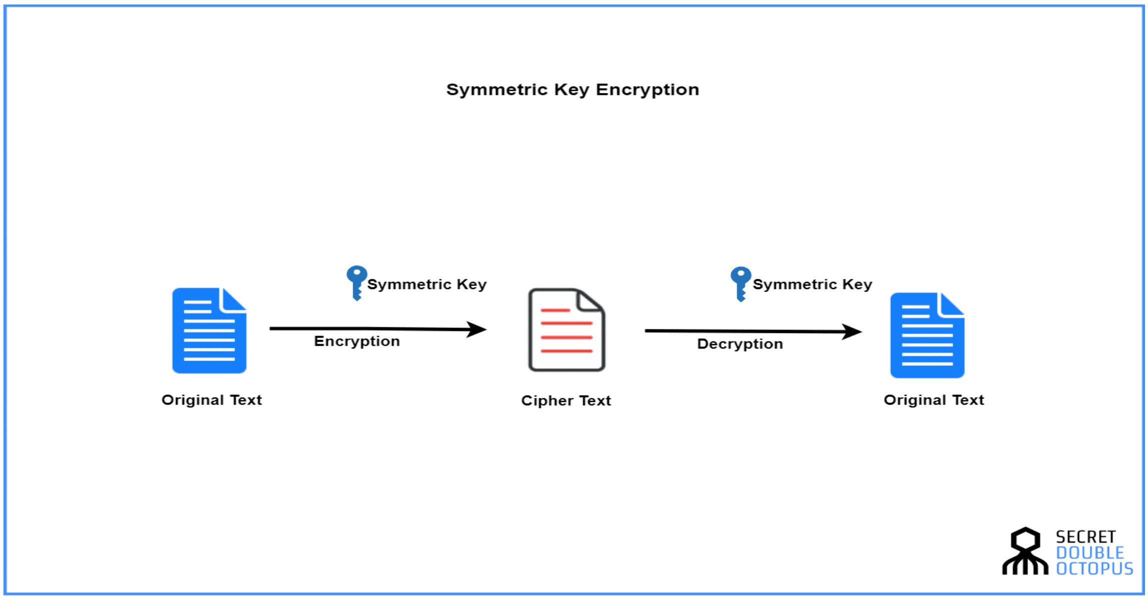
4. FRAMEWORK

Symmetric-key cryptography:

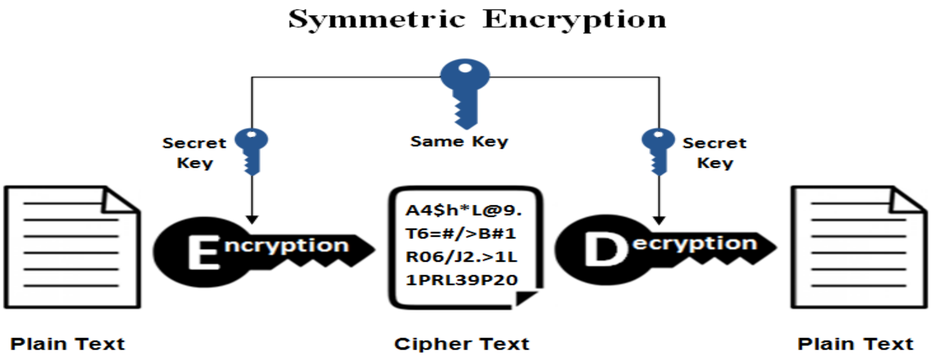
Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key (or, less commonly, in which their keys are different, but related in an easily computable way). This was the only kind of encryption publicly known until June 1976.

Symmetric key ciphers are implemented as either block ciphers or stream ciphers. A block cipher enciphers input in blocks of plaintext as opposed to individual characters, the input form used by a stream cipher.

The Data Encryption Standard (DES) and the Advanced Encryption Standard (AES) are block cipher designs that have been designated cryptography standards by the US government (though DES's designation was finally withdrawn after the AES was adopted).



One of the chief weaknesses of symmetric key cryptography lies in the use of one key. If the key is exposed beyond the sender and the receiver, it is possible for an attacker who has managed to intercept it to decrypt the message or, worse to decrypt the message, alter it, then encrypt it once more and pass it on to the receiver in place of the original message. Since such issues are present, symmetric key cryptography by itself provides only confidentiality, and not integrity, as we would not



be aware that the message in our example had been altered.

Block versus stream ciphers

Symmetric key cryptography makes use of two types of ciphers: block ciphers and stream ciphers. A block cipher takes a predetermined number of bits, known as a block, in the plaintext message and encrypts that block. Blocks are commonly composed of 64 bits but can be larger or smaller depending on the particular algorithm being used and the various modes in which the algorithm might be capable of operating. A stream cipher encrypts each bit in the plaintext message, 1 bit at a time. It is also possible for a block cipher to act as a stream cipher by setting a block size of 1 bit.

A large majority of the encryption algorithms in use at present are block ciphers. Although block ciphers are often slower than stream ciphers, they tend to be more efficient. Since block ciphers operate on larger blocks of the message at a time, they do tend to be more resource intensive and are more complex to implement in hardware or software. Block ciphers are also more sensitive to errors in the encryption process as they are working with more data. An error in the encryption process of a block cipher may render unusable a larger segment of data than what we would find in a stream cipher, as the stream cipher would only be working with 1 particular bit.

In general, several block modes can be used with an algorithm based on a block cipher to detect and compensate for such errors. We can see such modes in use with algorithms such as the Data Encryption Standard (DES) and Advanced Encryption Standard (AES), and we will look at some of these modes in the next section when we talk about the algorithms that use them.

Typically, block ciphers are better for use in situations where the size of the message is fixed or known in advance, such as when we are encrypting a file or have message sizes that are reported in protocol headers. Stream ciphers are often better for use in situations where we have data of an unknown size or the data is in a continuous stream, such as we might see moving over a network.

Symmetric key algorithms

Some of the cryptographic algorithms that are more recognizable to the general public are symmetric key algorithms. Several of these, such as DES, 3DES, and AES, are or have been in regular use by the US government and others as standard algorithms for protecting highly sensitive data.

DES first came into use in 1976 in the United States and has since been used by a variety of parties globally. DES is a block cipher based on symmetric key cryptography and uses a 56-bit key. Although DES was considered to be very secure for some period of time, it is no longer considered to be so. In 1999, a distributed computing project was launched to break a DES key by testing every possible key in the entire keyspace, and the project succeeded in doing so in a little more than 22 h.

This weakness brought about by the short key length was compensated for a period of time through the use of 3DES (pronounced triple DES), which is simply DES used to encrypt each block three times, each time with a different key. DES can operate in several different block modes, including Cipher Block Chaining (CBC), Electronic CodeBook (ECB), Cipher Feedback (CFB), Output Feedback (OFB), and Counter Mode (CTR). Each mode changes the way encryption functions and the way errors are handled.

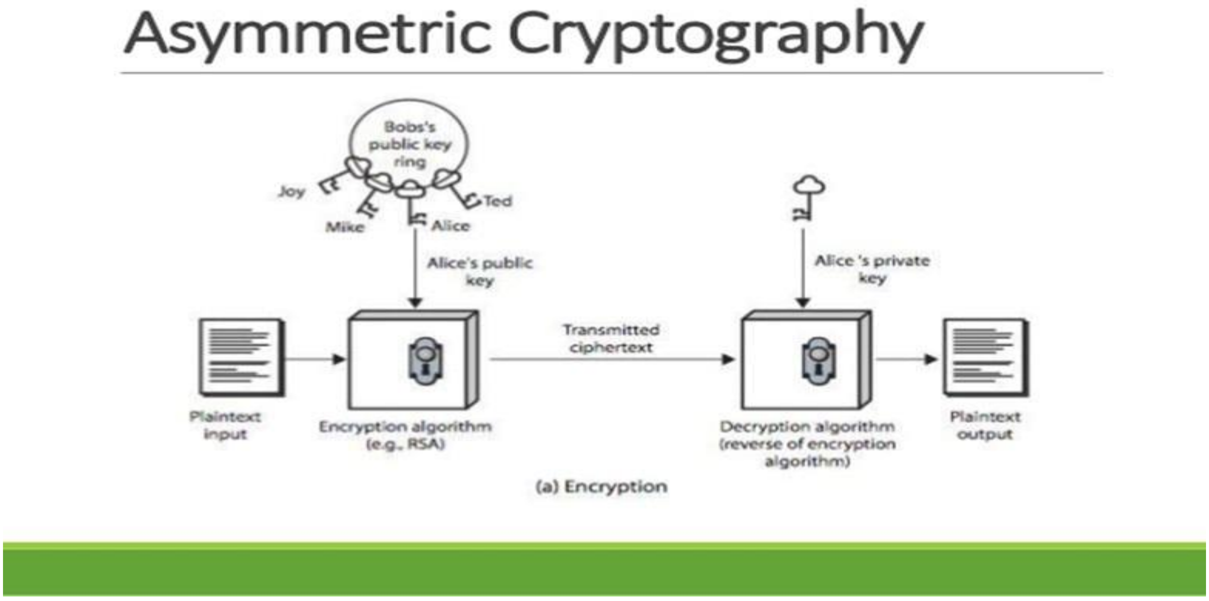
AES is a set of symmetric block ciphers endorsed by the US government through NIST, and now used by a variety of other organizations, and is the replacement for DES as the standard encryption algorithm for the US federal government. AES uses three different ciphers: one with a 128-bit key, one with a 192-bit key, and one with a 256-bit key, all having a block length of 128 bits. A variety of attacks have been attempted against AES, most of them against encryption using the 128-bit key, and most of them unsuccessful, partially successful, or questionable altogether.

At the time of this writing, the US government still considers AES to be secure. AES shares the same block modes that DES uses and also includes other modes such as XEX-based Tweaked CodeBook (TCB) mode.

There are a large number of other well-known symmetric block ciphers, including Twofish, Serpent, Blowfish, CAST5, RC6, and IDEA, as well as stream ciphers, such as RC4, ORYX, and SEAL.

Asymmetric Key Cyroptography*:*

Public-key algorithms are most often based on the computational complexity of "hard" problems, often from number theory. For example, the hardness of RSA is related to the integer factorization problem, while Diffie– Hellman and DSA are related to the discrete logarithm problem. More recently, elliptic curve cryptography has developed, a system in which security is based on number theoretic problems involving elliptic curves. Because of the difficulty of the underlying problems, most public-key algorithms involve operations such as modular multiplication and exponentiation, which are much more computationally expensive than the techniques used in most block ciphers, especially with typical key sizes.



When someone wants to send an encrypted message, they can pull the intended recipient's public key from a public directory and use it to encrypt the message before sending it. The recipient of the message can then decrypt the message using their related private key. On the other hand, if the sender encrypts the message using their private key, then the message can be decrypted only using that sender's public key, thus authenticating the sender. These encryption and decryption processes happen automatically; users do not need to physically lock and unlock the message.

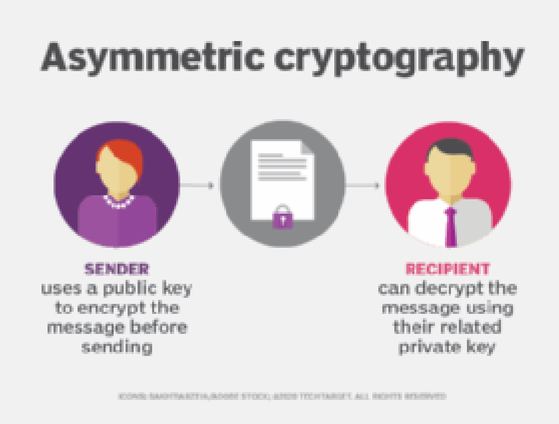
Many protocols rely on asymmetric cryptography, including the transport layer security (TLS) and secure sockets layer (SSL) protocols, which make HTTPS possible. The encryption process is also used in software programs -- such as browsers -- that need to establish a secure connection over an insecure network like the Internet or need to validate a digital signature.

Increased data security is the primary benefit of asymmetric cryptography. It is the most secure encryption process because users are never required to reveal or share their private keys, thus decreasing the chances of a cybercriminal discovering a user's private key during transmission.

How asymmetric cryptography works

Asymmetric encryption uses a mathematically related pair of keys for encryption and decryption: a public key and a private key. If the public key is used for encryption, then the related private key is used for decryption; if the private key is used for encryption, then the related public key is used for decryption.

The two participants in the asymmetric encryption workflow are the sender and the receiver; each has its own pair of public and private keys. First, the sender obtains the receiver's public key. Next, the plaintext -- or ordinary, readable text -- is encrypted by the sender using the receiver's public key; this creates ciphertext. The ciphertext is then sent to the receiver, who decrypts the ciphertext with their private key and returns it to legible plaintext.


Asymmetric cryptography
SENDER
uses a public key to encrypt the message before sending
RECIPIENT
can decrypt the message using their related private key

ICONS: SAADTIKAZETA/ADOBE STOCK; ©2020 TECHTARGET. ALL RIGHTS RESERVED

A visualization of how public and private keys are used in asymmetric cryptography

Because of the one-way nature of the encryption function, one sender is unable to read the messages of another sender, even though each has the public key of the receiver.

Uses of asymmetric cryptography

Asymmetric cryptography is typically used to authenticate data using digital signatures. A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document. It is the digital equivalent of a handwritten signature or stamped seal.

Based on asymmetric cryptography, digital signatures can provide assurances of evidence to the origin, identity and status of an electronic document, transaction or message, as well as acknowledge informed consent by the signer.

Asymmetric cryptography can also be applied to systems in which many users may need to encrypt and decrypt messages, including:

- Encrypted email - a public key can be used to encrypt a message and a private key can be used to decrypt it.
- The SSL/TSL cryptographic protocols - establishing encrypted links between websites and browsers also makes use of asymmetric encryption.
- Bitcoin and other cryptocurrencies rely on asymmetric cryptography as users have public keys that everyone can see and private keys that are kept secret. Bitcoin uses a cryptographic algorithm to ensure that only the legitimate owners can spend the funds.

In the case of the Bitcoin ledger, each unspent transaction output (UTXO) is typically associated with a public key. So if user X, who has an UTXO associated with his public key, wants to send the money to user Y, user X uses his private key to sign a transaction that spends the UTXO and creates a new UTXO that's associated with user Y's public key.

Benefits and disadvantages of asymmetric cryptography

The benefits of asymmetric cryptography include:

- the key distribution problem is eliminated because there's no need for exchanging keys.
- security is increased as the private keys don't ever have to be transmitted or revealed to anyone.

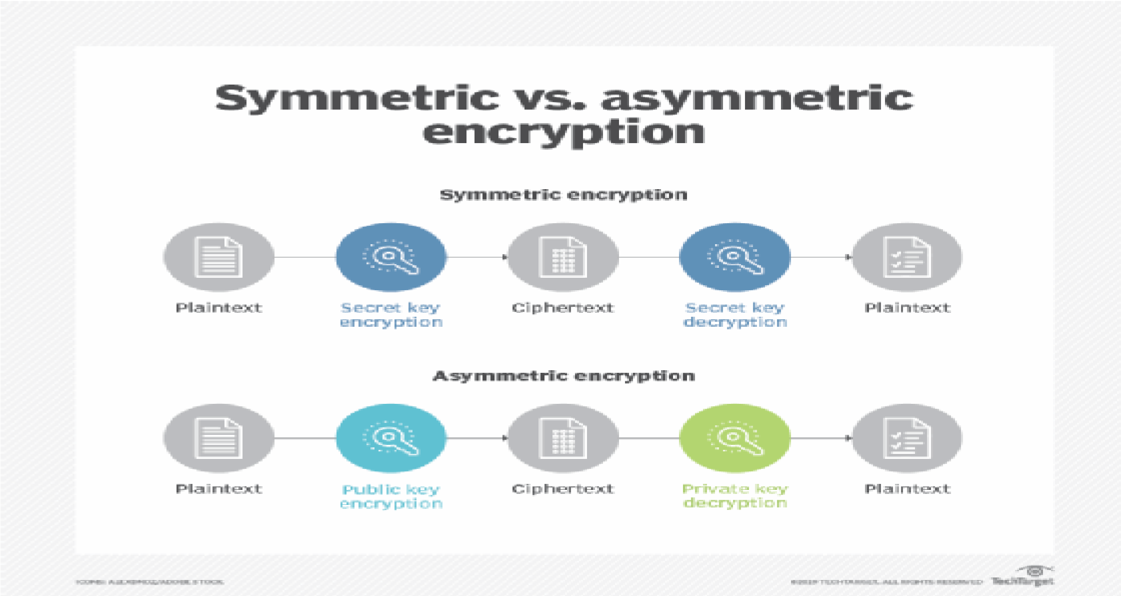
- the use of digital signatures is enabled so that a recipient can verify that a message comes from a particular sender.
- it allows for non-repudiation so the sender can't deny sending a message.

Disadvantages include:

- it's a slow process compared to symmetric cryptography, so it's not appropriate for decrypting bulk messages.
- if an individual loses his private key, he can't decrypt the messages he receives.
- since the public keys aren't authenticated, no one really knows if a public key belongs to the person specified. Consequently, users must verify that their public keys belong to them.
- if a hacker identifies a person's private key, the attacker can read all of that individual's messages.

Asymmetric vs. symmetric cryptography

The main difference between these two methods of encryption is that asymmetric encryption algorithms makes use of two different but related keys -- one key to encrypt the data and another key to decrypt it -- while symmetric encryption uses the same key to perform both the encryption and decryption functions.



This image displays how the asymmetric cryptography process differs from the symmetric cryptography process

Another difference between asymmetric and symmetric encryption is the length of the keys. In symmetric cryptography, the length of the keys -- which is randomly selected -- are typically set at 128-bits or 256-bits, depending on the level of security that's needed.

However, in asymmetric encryption, there must be a mathematical relationship between the public and private keys. Since hackers can potentially exploit this pattern to crack the encryption, asymmetric keys need to be much longer to offer the same level of security. The difference in the length of the keys is so pronounced that a 2048-bit asymmetric key and a 128-bit symmetric key provide just about an equivalent level of security.

Additionally, asymmetric encryption is slower than symmetric encryption, which has a faster execution speed.

Examples of asymmetric cryptography

The RSA algorithm -- the most widely used asymmetric algorithm -- is embedded in the SSL/TSL protocols, which are used to provide communications security over a computer network. RSA derives its security from the computational difficulty of factoring large integers that are the product of two large prime numbers.

Multiplying two large primes is easy, but the difficulty of determining the original numbers from the product -- factoring -- forms the basis of public key cryptography security. The time it takes to factor the product of two sufficiently large primes is considered to be beyond the capabilities of most attackers, excluding nation-state actors who may have access to sufficient computing power. RSA keys are typically 1024- or 2048-bits long, but experts believe that 1024-bit keys could be broken in the near future, which is why government and industry are moving to a minimum key length of 2048-bits.

Elliptic Curve Cryptography (ECC) is gaining favor with many security experts as an alternative to RSA for implementing public key cryptography. ECC is a public key encryption technique based on elliptic curve theory that can create faster, smaller and more efficient cryptographic keys. ECC generates keys through the properties of the elliptic curve equation.

To break ECC, one must compute an elliptic curve discrete logarithm, and it turns out that this is a significantly more difficult problem than factoring. As a result, ECC key sizes can be significantly smaller than those required by RSA yet deliver equivalent security with lower computing power and battery resource usage making it more suitable for mobile applications than RSA.

History of asymmetric cryptography

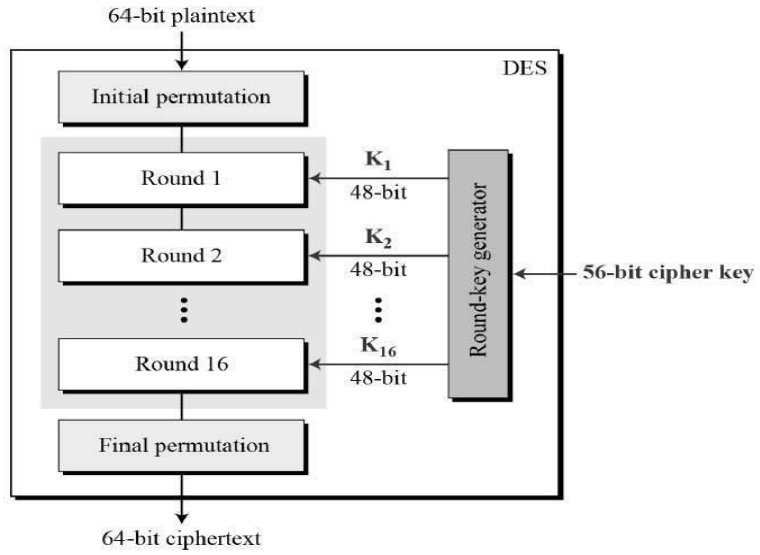
Whitfield Diffie and Martin Hellman, researchers at Stanford University, first publicly proposed asymmetric encryption in their 1977 paper, "New Directions in Cryptography." The concept had been independently and covertly proposed by James Ellis several years earlier, while he was working for the Government Communications Headquarters (GCHQ), the British intelligence and security organization.

Data Encryption Standard:

DES is the archetypal block cipher—an algorithm that takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another ciphertext bitstring of the same length. In the case of DES, the block size is 64 bits. DES also uses a key to customize the transformation, so that decryption can supposedly only be performed by those who know the particular key used to encrypt.

The key ostensibly consists of 64 bits; however, only 56 of these are actually used by the algorithm. Eight bits are used solely for checking parity, and are thereafter discarded. Hence the effective key length is 56 bits.The key is nominally stored or transmitted as 8 bytes, each with odd parity.

Before the main rounds, the block is divided into two 32-bit halves and processed alternately; this criss-crossing is known as the Feistel scheme. The Feistel structure ensures that decryption and encryption are very similar processes—the only difference is that the subkeys are applied in the reverse order when decrypting.



Advanced Encryption Standard:

AES is a subset of the Rijndael cipher developed by Belgian cryptographers, VincentRijmen and Joan Daemen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes.For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.

A. Byte Substitution (SubBytes)

The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

B. Shiftrows

Each of the four rows of the matrix is shifted to the left. Any entries that ‘fall off’ are re-inserted on the right side of row. Shift is carried out as follows −

- First row is not shifted.
- Second row is shifted one (byte) position to the left.
- Third row is shifted two positions to the left.
- Fourth row is shifted three positions to the left.
- The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

C. MixColumns

Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

D. Addroundkey

The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

E. Decryption Process

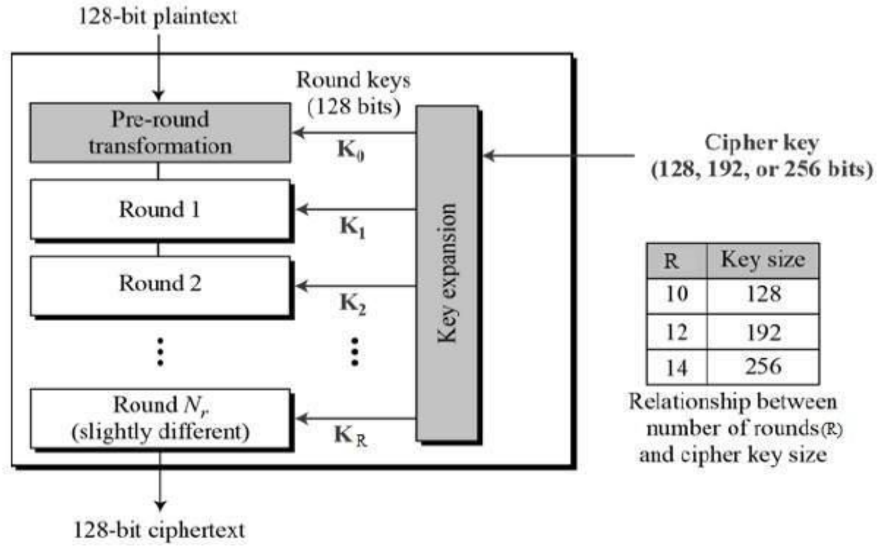
The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order

□ Add round key

□ Mix columns

□ Shift rows

□ Byte substitution



R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

RC-2 Encrypion Algorithm:

In cryptography, RC2 (also known as ARC2) is a symmetrickey block cipher designed by Ron Rivest in 1987. "RC" stands for "Ron's Code" or "Rivest Cipher"; other ciphers designed by Rivest include RC4, RC5, and RC6.

The development of RC2 was sponsored by Lotus, who were seeking a custom cipher that, after evaluation by the NSA, could be exported as part of their Lotus Notes software. The NSA suggested a couple of changes, which Rivest incorporated. After further negotiations, the cipher was approved for export in 1989.

Strongest Data Encryption Algorithms

There are several data encryption algorithms available:

- TripleDES
- Twofish encryption algorithm
- Blowfish encryption algorithm
- Advanced Encryption Standard (AES)
- IDEA encryption algorithm
- MD5 encryption algorithm
- HMAC encryption algorithm
- RSA security

Triple Data Encryption Standard (TripleDES)

This form of data encryption algorithm applies block cipher algorithms thrice to all the data blocks individually.

The magnitude of the key is enlarged to provide extra protection by increasing the encryption ability.

Every individual block constitutes of 64-bit data. In this encryption algorithm, three keys are used where each key constitutes of 56 bits.

A total of three key permutations are provided under this standard:

- Option #1: the three keys are independent
- Option #2: keys 1 and 2 are independent

- Option #3: the three keys are similar

Most importantly, we call #3 triple DES whose key length consists of (3*56 bits = 168 bits) whereas key security consists of (2*56 bits = 112 bits).
The substantially longer key length of this type of encryption algorithms overpowers other encryption techniques.
Nevertheless, after the development of the advanced encryption standard (AES), TripleDES has been rendered old-fashioned.

alsoRead

Post Quantum Crptography

Blowfish Encryption Algorithm

Developed in 1993, the Blowfish encryption algorithm is an alternative for Data Encryption Standard (DES).
Before its creation, encryptions were performed by patents and intellectual properties of firms.
The developer placed the protocol to the public to make it readily available for any interested user.
Compared to DES, it is substantially faster and offers better encryption security.
It is an asymmetric type of encryption protocol: uses a single key for both encryption and decryption.
Like Twofish, it is a block cipher and its block size is 64-bit and the key size lies anywhere between 32 – 448 bits.
It features 18 subkeys, sixteen rounds and has four S-boxes.
Its protection capability has been examined and proved.
Considering blowfish standard is regarded as a Feistel cipher, a single structure is used to encrypt and decrypt data provided that the reverse direction of the round keys is considered.
It is a significantly fast operation because it involves a relatively small number of rounds as well as its clarity of functionality.
Nevertheless, its key-scheduling consumes a lot of time, although it has an upper hand when it comes to protecting brute-force threats.
Also, its 64-bit block length (size) is rather small making it endangered by birthday attacks compared to AES whose block size is 128 bits and above.

Twofish Encryption Algorithm

This form of the encryption algorithm is a symmetric key block cipher which is characterized by 128-bit block size and whose keys' size can run up to 256 bits.

This protocol uses one key for encryption and decryption.

It is a fast and flexible standard for eight-bit and thirty two-bit CPUs, and small smart cards. The protocol works exemplarily in hardware and has numerous functionality commutations between the speed of encryption and the setup time making it distinctive amongst other protocols.

The standard shares some features with its predecessor, blowfish Encryption Algorithm and AES.
At one time, this encryption algorithm was a real contestant for the best encryption standard, but the present AES beat it out.

This algorithm bears several peculiar characteristics that distinguish it from other standards. First, this cryptographic protocol applies substitution-boxes, S-boxes that are pre-computed and key-reliant.

This implies that despite the provision of the S-box, it relies on the cipher key for the decryption of the encrypted data.

The significance of the S-box is to conceal the key connection with the ciphertext.
Secondly, the Twofish encryption standard is accepted as a substantially secure alternative. Encryption protocols whose keys have 128 bits and above are regarded as safe from attacks:

Twofish has a block size of 128 bits.

Twofish protocol comes with several options.

To execute fast encryption, the key setup time can be made longer; this is done when the amount of data (plaintext) to be encrypted is relatively large.

The encryption can be made slower by setting a shorter key setup time when short blocks with constantly alternating keys are to be encrypted.

For some PC users, Twofish is regarded as the best AES protocol due to its peculiar amalgamation of design, resilience, and speed.

5. CONCLUSION

The main goal is to securely store and access data in cloud that is not controlled by the owner of the data. We exploit the technique of elliptic curve cryptography encryption to protect data files in the cloud. Two part of the cloud server improved the performance during storage and accessing of data. The ECC Encryption algorithm used for encryption is another advantage to improve the performance during encryption and decryption process. We assume that this way of storing and accessing data is much secure and have high performance. Our efforts are going on to solve the problem of group sharing of data in the shared data section as in this scheme only member of group can access the data stored over shared data section. One to many, many to one, many to many communication is not possible.

Security is the main challenge, requirement and aspect defined for distributed Cloud System environment. The numbers of security integrated Cloud System frameworks are available to provide distributed data and file storage.

This security system provides the file management, forwarding and storage in encoded form. The authorization, authentication and encoded storage are also provided in distributed Cloud storage. In this present research, a more functional, secure and reliable security system is provided. The capabilities of this provided security system in Cloud Computing are listed hereunder:-

a. The defined improved security system in Cloud Computing is hybrid with private and public access to the Cloud.

b. The private access to the Cloud System is provided using user level authentication. As the user enters to the Cloud System the personalized space is also generated.

c. A user can generate more than one personalized space and load data files only in its personalized Cloud space. The personalized storage space is generated with secure key specification.

d. The public users share the common storage space called global storage space. User can enter to the Cloud System environment without any authentication.

e. For the public user, the encoded file storage is provided using RSA based encryption method by using the same encryption key.

f. · For the private user, the encoded file storage is provided using RSA and DES based hybrid approach. The keys are generated separately and managed in the secure personalized storage space. · User can upload files of different formats and of different size effectively and in secure integrated form.

g. As the user registers to the Cloud System, the user personalized session begins and user can upload and download multiple files in the same session.

h. The session key based encoded communication is provided using integrated SSL tunnel. The tunnel specific encoded communication is performed using block specific encryption and storage.

i. The private user can create the user group and also decide the group members .

j. The secure sharing of the personalized files can be done within the group as well as to other registered users.

k. The file forwarding can be done to other users in encoded form. The key specific sharing can be done.

l. The user can retrieve file back by applying the decryption algorithm and by providing the decryption key available to the user. Based on these features a Hybrid Secure System is provided in this work with group sharing, session based communication and encoded storage. The proposed secure Hybrid Cloud storage system is implemented in Hadoop integrated Cloud environment. To evaluate the system significance and reliability, the implementation specific comparison is provided in this research. The comparative results are concluded as follows:-

- The evaluation of proposed system is done on a sample set of 35 files of seven different formats.
- The evaluation results are provided in terms of process time and encoded file size parameters.
- The evaluation results are represented in tabular form and graphically.
- The first Cloud model for comparison is implemented with RSA based encoding in the hybrid Cloud environment.
- The comparative analysis is done on multiple files and effective observation is taken with encryption and decryption time.
- The comparative results show that the proposed model has improved the efficiency and reduced the time of encryption and decryption

6. FUTURE SCOPE

As the security is the main concern in distributed Cloud environment, in this present research a secure hybrid Cloud System is provided. This proposed security system environment provides the multiple storage and security features. The file level functionality provided in this work includes storage, retrieval, forwarding and key 216 sharing. The security is provided using authentication, authorization, file system storage, secure file communication, session specific communication and group communication. The work can be extended in the future under different aspects.

- The presented system is defined specifically for public and private Cloud access in hybrid and generalized environment, in the future the proposed system can be applied to an organization.
- No level of authorization within an organization and community is provided. In future more dedicated accessibility or authorization can be defined in the hybrid Cloud System environment.
- The presented work is implemented on a generic secure Cloud System without specification of any attack, in the future the work model can be implemented on some attack specific security in a Cloud environment.
- The security model can be extended and applied for more complex Cloud System architectures such as mobile Cloud or green Cloud environment.

- The presented system does not observe the user pattern of the access behaviour and did not define any intelligent security aspect to the environment. In future, some behaviour observation adaptive security constraints can be included to the Cloud environment. The access behaviour and functional behaviour can be applied to improve the security strength in Cloud environment.
- Most of the security considerations are static or the random based. No user information or the server information or file information is integrated while generating the key or the security aspect. In future, the involvement of the data information can be considered to achieve more dynamic and adaptive security method.
- The proposed security system used the tunnel specific and session specific secure transmission to achieve the efficiency but did not use any optimization algorithm to improve the effectiveness or the security. In future some optimization algorithm can be applied to improve the model effectiveness and reliability.

7. REFERENCE

1]VijayaPinjarkar, Neeraj Raja, KrunalJha,AnkeetDalvi, “Single Cloud Security Enhancement using key Sharing Algorithm, ”Recent and Innovation Trends in Computing and Communication, 2016.

[2] V. Vankireddy, N. Sudheer, R. Lakshmi Tulasi, “Enhancing Security and Privacy in Multi Cloud Computing Environment, ”International Journal of Computer Science and Information Technologies, 2015.

[3] Swapnila S Mirajkar, Santoshkumar Biradar, “Enhance Security in Cloud Computing, ”International Journal of Advanced Research in Computer Science and Software Engineering,2014.

[4] Ashalatha R, “A survey on security as a challenge in cloud computing,”International Journal of Advanced Technology & Engineering Research (IJATER) National Conference on Emerging Trends in Technology,2012.

[5]www.google.com

[6] G. L. Prakash, M. Prateek and I. Singh, ’Data Encryption and Decryption Algorithms using Key Rotations for Data Security in Cloud System’, International Journal Of Engineering And Computer Science vol. 3, issue 4, pp. 52155223, April 2014

[7] N. Saravanan, A. Mahendiran, N. V. Subramanian and N. Sairam, ’An Implementation of RSA Algorithm in Google Cloud using Cloud SQL’, Research Journal of Applied Sciences, Engineering and Technology, Oct. 1 2012