Asian Journal of Information Technology 15 (23): 4732-4743, 2016
ISSN: 1682-3915
© Medwell Journals, 2016

Secure Data Transaction and Forwarding in Cloud Environment

[1]K.L. Neela and [2]V. Kavitha
[1]Department of Computer Science and Engineering,
University College of Engineering, Nagercoil, Kanyakumari, Tamil Nadu, Indai
[2]Department of Computer Science and Engineering,
University College of Engineering, Kancheepuram, Tamil nadu, India

Abstract: Now a days, most of the companies are moving to the cloud environment, because of its major advantages such as flexibility, faster implementation and lower cost. Almost every cloud applications have two major challenges such as security and privacy. Today, most of the sensitive data are stored in a third party systems may cause severe security problem. According to third party cloud storage system, data confidentiality depends on key distribution center only. Our secure cloud storage system belongs to the decentralized architecture where storage server offers good scalability. Since many of the security algorithms are required key handling management and regeneration which creates greater overhead. It is proposed to introduce an efficient security algorithm such as a Cyclic Shift Transposition Algorithm (CSTA) for secure cloud storage system. It consists of the combination of shifting and transportation operations. This algorithm not only prevents from known attack but also maintain the integrity of the data. Here the onetime verification system and hash based timestamp technique are used to forward and retrieve the data in a secure manner. It builds the additional security wall over the data. The proposed model provides high data integrity and privacy so that only trusted user can get the data in an efficient manner.

Key words: Secure storage system, OTP, key distribution center, CSTA, key oriented algorithm, third party system

INTRODUCTION

Today, most of the software company (Sales Force.com) operates almost entirely in the cloud. It is difficult to define the cloud computing in a single definition. Actually, Cloud computing provides a pool of computing resources to the users through the internet. It can deploy, allocate or reallocate the resources dynamically and monitor the usage of resources at all times (Grossman, 2009). The organizationuses cloud environment because of its major advantage such as pay-for-what-you-use. It resembles the way electricity, fuel and water are consumed. So sometimes it is referred as utility computing. Cloud computing is composed of three service models and four deployment models and essential characteristics.

Cloud computing consists of three service models such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). IaaS allows users to run any application they please on cloud hardware of their own choice. It provides cloud consumers with a high level of control and responsibility over its configuration and utilization. Iaas builds the virtual infrastructure for the physical resources such as server, storage arrays and networking. The best well known Iaas provider is Amazon web services. Iaas service provider owns the equipment and is responsible for running and maintaining it. The client typically pays on a per-use basis. The consumer does not control and manage the underlying cloud infrastructure.

Platform as a Service (PaaS), allows users to create their own cloud application using supplier-specific tools and language. It limits the developer to provide language and tools. It provides an application runtime environment for the consumer. It provides an efficient approach to operate scale - out application in a cost- effective manner. The best well know paas provider is Google Apps Engine. It delivers operating system and various services without requiring any download or installation, over the internet and Software as a Service (SaaS), allows users to run existing application (Singh and Raj, 2012). In a SaaS environment, users need not to license the software and don't need to invest in hardware. It can be accessed only through the browser. It delivers on-demand application

Corresponding Author: K.L. Neela, Department of Computer Science and Engineering, University College of Engineering, Nagercoil, Kanyakumari, Tamil Nadu, India

that is hosted and managed by the service provider and typically paid for a subscription basis (Bhoyar and Chopde, 2013). Instead of companies installing software on their own servers, software providers host the software and just charge them according to the time they spent using it or a monthly fee.

The cloud deployment model represents a particular type of cloud environment which is distinguished by size and ownership. There are four models one is public cloud, another one is private cloud, third is hybrid cloud and fourth is community cloud. Public cloud is accessed by any users with an internet connection. The capital overhead and operational cost of this model is economical. The provider may provide the service free or in the form of the license policy like pay per use. The example of public cloud is Google.

A private cloud is controlled by a single organization. It permits only the authorized user gives the organization greater and direct control over the data and hybrid cloud is the combination of public and private cloud. An organization may store critical information on private cloud and less secure information are hosted in public cloud. Not only this they can use the hybrid cloud model for processing big data. Hybrid cloud provides scalability, flexibility and security to the data. Community cloud sharing the infrastructure of the same community within the organization. It is a multi tenant platform which allows several companies to work on the same platform. Cloud computing consists of several essential characteristics such as on-demand self service, broad network access, resource pooling, resources are released through on-demand basis and resource usage are monitored and measured based on utilization, i.e., pay for use.

Cloud computing, especially used in small and mid-sized applications and prototypes in order to share the resources such as servers, network equipment etc., resulting in reducing cost (Mell and Grance, 2011). Despite of the tremendous benefits, Security and the confidentiality of the sensitive data are the primary concern during the computation of the cloud. Storing the data in a third party system (centralized system) may cause serious security problems because data confidentiality depends on third party only. Third party systems are semi-trusted one so that unauthorized users can read the encrypted data from cloud storage server. Not only that, there is high communication traffic can occur between the user and storage server when encryption integrated with encoding messages (Rajasekaran, 2013). In our secure cloud storage system provides data confidentiality, secure data transaction and data retrieval. Still now the research problems have to be identified in the challenging area of data storage security and forwarding in cloud computing. In this scheme secure transaction and retrieval can be achieved through CSTA algorithm, one time verification system and hash based time stamp.

The cloud service provider has to ensure three important characteristics in the cloud such as confidentiality, integrity and availability of data. Confidentiality can improve by keeping the data in a safe, i.e., unauthorized users cannot get your data. This is achieved through a cryptographic method. Integrity protects the data from unauthorized user which helps to avoid the accidental alteration to the data. Availability can be achieved through replication of data. A cloud storage system is also known as large scale distributed storage server because it consists of many independent storage servers. Since a huge amount of data is stored in cloud server which creates greater security challenges to the data. Moving data in to the cloud occurs more frequently. So, it needs to secure the data in the cloud computing through cryptographic functions. Cryptography is often associated with scrambled plaintext to cipher text. An early substitution cipher has never offered much confidentiality from the opponent.

The proposed system provides efficient mechanisms for sharing the message between source and destination. CSTA consists of the partition and shifting operation. Many authentication techniques are available in the cloud like static password scheme (e.g., password) for protecting the data from malicious users. But the unauthorized user can reset all passwords just to create confusion (Richa and Satyakshma, 2013). So that it is needed to move to more dynamic password scheme like one time verification system. Although today many attacks such as Phishing, replay attack and man in the middle attack are increasing throughout the world in a cloud environment. So, it needs to move to the strong authentication system. One time verification system reduces the above attack by providing one time password to the client which is valid for that session only. Since OTP valid for one time, the intruders have no chance to reuse the password.

Through the hash based timestamp, a timestamp (Bonnecaze *et al*., 2006) is generated which enable data to be verified without any trusted parties. It ensures data verification and audit ability within the cloud environment. The proposed system not only address the security threats in the cloud data storage system but also focus on some problems which occurs when the data is forwarded from one to another user by the storage server directly under the command of thedata owner.

Literature review: Proxy re-encryption method (Mambo And Okamoto, 1997) has the third party (proxies) to re-encrypt the already encrypted data to create a new encrypted data. But the third party doesn't get to see that the data are decrypted so it never learns anything. Proxy Re-encryption can be applied for email forwarding and secure network file storage but it has some collusion problem (Inbarani *et al*., 2013).

Hierarchical Identity-Based Encryption (Horwitz and Lynn, 2002), each user use his secret key to decrypt any file encrypted using it's paired public key along with all the public keys of the user's ancestors. But it is easy for third parties to learn about file security levels. Threshold Proxy re-encryption (Lin and Tzeng, 2012) supports the encoding operations over encrypted message and forwarding operations over encrypted and encoded message. It follows multiplicative homomorphic property. It provides high access control mechanisms.

Navdeep proposed a method (Kant and Sharma, 2013) which is used to preserve the privacy of the information in order to ensure that this information cannot be misused. The privacy of the information can be ensured using OTP/WTP. The OTP provides a user new password each time and WTP provides a new password for a user weekly. OTP used for one time use and WTP used for frequent use. This proposed model ensures privacy and data security.

Geetanjali and Jainul (2014) proposed a method which encrypts one time password by a public key of the user. This method does not require a third party to control the system. It is designed to improve security, efficiency and to remove dependency of third parties.

Biometric encryption (Omar *et al*., 2014) is proposed to enhance the biometric data confidentiality in cloud computing. Biometric identification includes iris, voice, fingerprint; etc are used for authentication purpose. It provides high security for the document. But it does not solve any of the security issues in cloud computing.

Sonali and Tidke (2015) proposed a method which fully integrates encrypting, encoding and forwarding technique. Here, the storage servers independently perform encoding and re-encryption operation and key servers independently perform partial decryption. This method provides more confidentiality, reliability and availability of the data. But here sometimes key management issues may occur.

Neural data security model (Jegadeeswari *et al*., 2015) consists of a dynamic hashing fragmented component together with feedback neural data security component. This can be achieved by using RSA algorithm and also dynamic hashing is used to store the fragmented sensitive data. It provides high security, less expensive and higher performance over the data.

Secure cloud computing based framework such as smart-frame (Baek *et al*., 2015) is used to build a hierarchical structure for information management and big data analysis. It provides security to the framework based on identity- based encryption, signature and proxy re-encryption. This scheme not only provides scalability but also provides flexibility. This secure framework mainly depends on proxy re-encryption technique in order to provide mobile application in cloud with high security. And also smart frame suggests a centralized service to manage the information storage. Centralized service may cause some serious security issues in a cloud environment.

Cloud trust (Gonzales *et al*., 2015) model provides high level security for Iaas cloud computing system. It provides a secure architecture for virtual images. It consists of CCS security control which is used to estimates the metrics for improving security in the cloud. It provides special safety control to VM images in Iaas provider. But this control does not include all possible insider attacks.

Kanade *et al.* (2015) proposed the method such as partition and encryption method which will help to improve the cloud security using TPA. Data partitioning provides an easy and effective way to store the data in the cloud. In order to ensure data integrity it uses the remote data integrity service. This scheme depends on third party administrator which may cause security issues.

Nepal *et al.* (2015) proposed a trusted storage cloud for scientific workflows called TruXy. TruXy provides security services for storing and sharing the data to another user. It provides security to multiple virtual machines in the cloud. It supports data integrity, availability and secure data sharing services in Truxy.

Xia *et al.* (2015) proposed a privacy-preserving content- based image retrieval scheme which allows the data owner needs to encrypt the image database and then outsource the encrypted image database and index to the cloud. It creates burden to data owner and data users.

Mohamed *et al.* (2013) proposed a scheme which is used to improve the security in a cloud environment. It is composed of three phases. First phase provides strong authentication using OTP and second phase performs encryption in order to improve the confidentiality. The third phase checks the data integrity by using a hashing algorithm. This scheme provides high security and fast recovery of data.

Srinivasulu *et al.* (2014) proposed the scheme which consists of proxy re-encryption, integrated with decentralized erasure code such that a secure storage system is constructed. It integrates encrypting, encoding and forwarding method. It allows TPA to audit the cloud storage without demanding user's time or resources.

Cloud computing adaption framework (Chang and Ramachandran, 2016) has been customized for securing cloud data. It deals with multi- layered security which provides security to the data in the data center. It also ensures reduction of viruses, worms and denial of service attacks. It provides security to the data either at rest or in use. Multilayered security includes access control, intrusion detection system and intrusion prevention system.

Kant and Sharma (2013) proposed a method which provides cryptographic algorithm with samba storage environment and also it provides a convenient way to store and access the files with confidentiality, integrity and authentication properties. It provides efficient mechanisms for data authenticity in cloud architecture.

Murthy (2014) proposed a digital signature based authentication scheme with a decentralized architecture for distributed key management with multiple key distribution centers. The key distribution depends upon the multiple KDC structure. Data security has been provided by using homomorphic encryption technique. A trusted third party is used for key generation. This method also proposed automatic retrieval mechanisms where lost data is recovered by data replication (Sharma and Khiva, 2013).

Problem definitions: With cloud computing all data are stored on cloud server in a secure way. But the questions are that how secure the data are stored in the cloud? Can unauthorized user get our confidential data? But cloud computing companies say that the data is secure but it is not completely sure that. So, security and privacy are always major issues in cloud computing. Actually, the data owner cannot know where the data is stored in a cloud environment. The data is migrated from one server to another. So, it is need to improve security of the confidential data which is stored in cloud server. The sharing of data is possible in a cloud environment. So it is needs to secure the data when it is stored in cloud server.

Cloud Service Provider has to provide security to the data in 3 ways. First, while transmitting the data, second when the data is at rest (storage area) and finally when the data is retrieved. Normally the data are replicated in many servers in order to maintain the data availability in a cloud environment. But if any changes or attack occurs in one replication will affect others also. Not only this, there are lot of attacks such as denial of service attack or man in the middle attack, etc. are damaged the data in a cloud environment. So, always security and integrity are the major challenging factors in a cloud environment. Every cloud service provider has to secure the data in a cloud environment.

In the existing system, centralized approach is used for data storage and data forwarding. Thus, storing data in a third party system may cause serious security problems. Cloud data should be secure and reliable otherwise unauthorized user can easily get the cloud user's confidential information. In cloud all the accounts were accessed with the same password which is very insecure and third party cloud computing doesn't provide proper security to the cloud data (Geetanjali and Jainul, 2014). This is the serious issues in the cloud computing environment because static password can be hacked by hackers.

The proposed scheme mainly concentrates on data confidentiality and data integrity. Data confidentiality is used to protect the confidential data or files from an unauthorized user or systems. The solution for this security problem is that encrypt the data in an efficient manner by using a Cyclic Shift Transposition Algorithm (CSTA). It provides security to the data when it is at rest and also processing the data. Iaas encryption such as CSTA can apply when the data is at rest (storage area) but it is not feasible to encrypt the data at rest in Paas and Saas cloud based application (Ajoudanian and Ahmadi, 2012) .CSTA has performed encryption on a real time before sending the data to the cloud. If an unauthorized user gets the data from cloud storage they can see only encrypted data. Authorized person only can decrypt the data successfully.

Data can be forwarded from one to another in a secure way using one time verification system. Since, OTP is valid for one time only, i.e., for each time user gets the new password. Typically, one-time password is a series of meaningless number or characters. It is difficult to remember the one time password. Here there is no duplication can occur consecutively. OTP authentication generates highly secure one time password, ensuring that only properly authenticated users are authorized access to critical application and data. The username/ password security scheme is considered very insecure for many reasons, including phishing attack, sniffing and other social engineering problems. But this OTP constantly changes at time intervals. So it is impossible for hackers to steal the data. Finally the data integrity can be checked at the receiver side using the hash based timestamp scheme.

The hash based timestamp is used to check the integrity of the data when it is retrieved. Hash algorithm provides an effective way to detect the digital data, whether it is altered or not. The timestamp is another concept that employs a hash algorithm to detect modification. To make meaningful to the document, the application must bind a valid time to the hash. The proposed system improves the security, performance, privacy and reliability in a cloud computing environment.

MATERIALS AND METHODS

Proposed systems: Constructing a secure storage system is a major challenging function when the storage system is distributed. Nowadays hackers can easily access the data storage environment. A third party system is semi trusted one to hold the confidential data, files and records. It may create some serious security issues. So that, we construct a secure cloud data storage system that supports the functions of secure data transaction and forwarding by using CSTA, one time verification system and hash based timestamp. The proposed system provides efficient authentication mechanisms in a cloud environment. Our system is highly decentralized architecture where storage server offers good scalability because the storage server can attach or detach anytime without the control of central authority. It provides efficient and effective data confidentiality, data forwarding and data retrieval. The overall architecture of the proposed method is shown in Fig. 1.

The proposed scheme mainly concentrates on data confidentiality and integrity. Due to the lack of confidentiality, the unauthorized user can easily hack the information. In this study, data owner directly communicate to the cloud server without any third party system. He (Data owner) encrypts the data using CSTA Method and stored in cloud server. When another user (says User B) wants to retrieve the same data, then the data owner checks their credentials from the database. If they are valid then generate one time password using one time verification system. After receiving OTP, they send the result back to the data owner. If it is matched, then it allows the user to decrypt the data using the CSTA decryption method and hash based timestamp method is used to check the integrity of the data. The proposed scheme provides security to the data when it is at rest and also processing the data in the cloud. That is the unauthorized user only view the encrypted form of the message.

Data confidentiality: The initial stage of the work is to store the data in public cloud storage. These can be performed by using Cyclic Shift Transposition Algorithm (CSTA). The proposed method does not depend on any third party system such as key management system. CSTA performs encryption and decryption process based on partition and shifting operation.

Encryption process: The data owner partitions the sentences in to four groups and then perform shifting operation such as row shift, column shift, primary diagonal shift, secondary diagonal shift which finally gives the rearranged text (Selvi and Kavitha, 2012; Gomathi and Manimegalai, 2014). This rearranged text is converted into ASCII format which is the cipher text of the given plain text. Then finally sent the encrypted form of the text to the cloud storage server. The encryption process is shown in Fig. 2.

Owner performs the partition operation by splitting the sentences into 4×4 matrixes. Then perform column shift operation to that matrix in a specific order. The order of the shift changes depends upon the sender and receiver. Subsequently, it performs row shift and diagonal shift operations. Finally the outcome matrix is converted in to ASCII values. If the unauthorized user wants to see the value they get only this encrypted format of data.

Algorithm for the encryption process:

```
Input: Text File
Step1: Each text is split in to an NxN matrix format
Step2: Perform column shift in a certain order specified
Step3: Perform row shift in a certain order specified.
Step4: Perform Diagonal shift in a certain order specified.
Step5: Perform secondary Diagonal shift in a certain order specified.
Step6: Represent the outcome in a linear order
Step 7: Convert the outcome of the ASCII format to get the encrypted text.
Output: Cipher text
```

Decryption process: The receiver requires decrypting the message in order to get the original plain text. This decryption can be performed using the same shifting and the partition operation of the CSTA encryption. The decryption is the reverse process of the encryption and hence it obtains the plain text. The performance of the decryption is shown in Fig. 3.

Algorithm for the decryption process:

```
Input: Cipher text.
Step1: Convert the ASCII values of the text. Split the cipher text into blocks of size NxN.
Step2: Perform Secondary Diagonal shift in an order carried out.
Step3: Perform prime Diagonal shift in an order carried out in a certain order specified.
Step4: Perform Row shift in a certain order specified.
Step5: Perform Column shift in a certain order specified.
Step6: Formulate the outcome in a linear order to get the decrypted text.
Output: Text File
```

CSTA depends on partition and shifting operation. For encryption shifting operation can be performed on column, row, prime diagonal and secondary diagonal. Shift row transposition consists of cyclic shift operation. More specifically, the bytes in a row r (0 = r = 3) are cyclically shifted for r bytes and also column c(0= c = 4) are cyclically shifted for c bytes. Consequently, for 0 = r = 4 and 0= c = N_b=4 the shift row transposition can be formally expressed as follows (Oppliger, 2011):

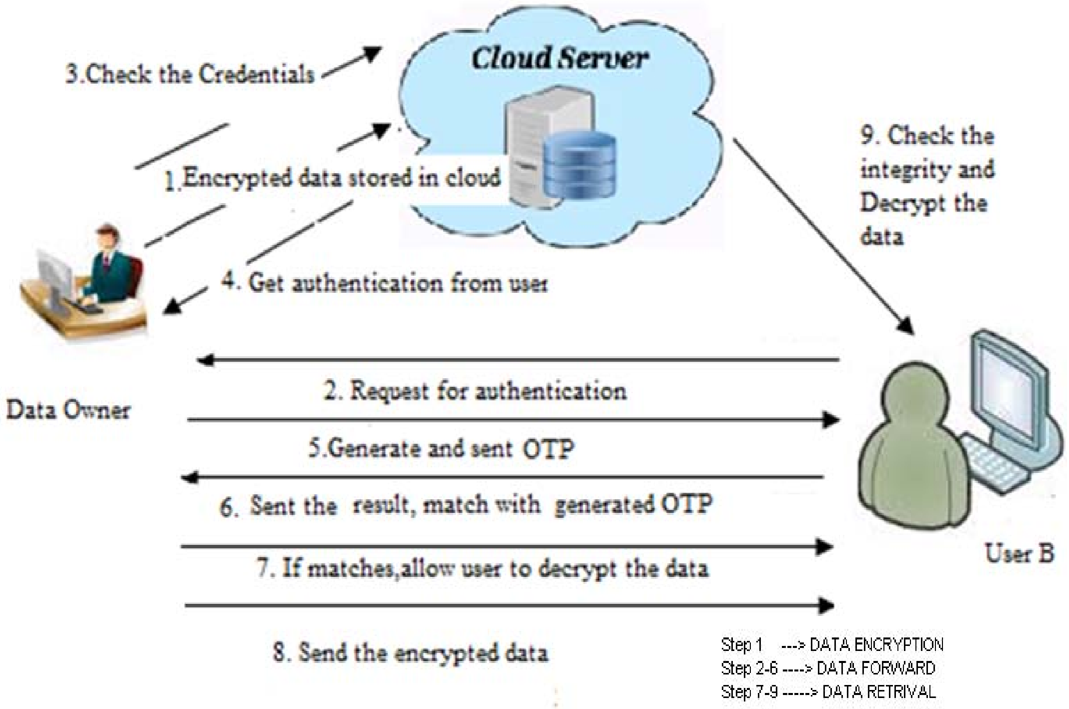
Cloud Server
3.Check the Credentials
1.Encrypted data stored in cloud
4. Get authentication from user
9. Check the integrity and Decrypt the data
Data Owner
2. Request for authentication
5.Generate and sent OTP
6. Sent the result, match with generated OTP
7. If matches,allow user to decrypt the data
8. Send the encrypted data
User B
Step 1 ---> DATA ENCRYPTION
Step 2-6 ----> DATA FORWARD
Step 7-9 -----> DATA RETRIVAL

Fig.1: Architecture diagram

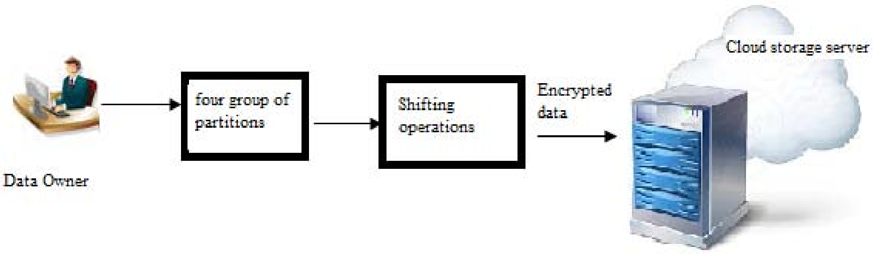

Fig. 2: CSTA encryption process

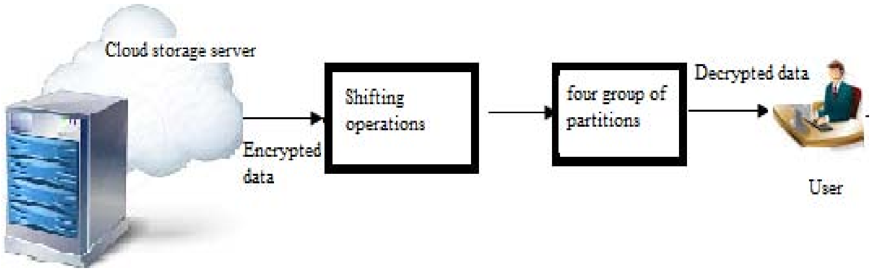

Fig. 3: CSTA decryption process

$$S`_{r,c} = S_{r,c} + \text{shift}\ (r, N_b) \quad (1)$$

Here shift (r, N_b) only depends on the row number r and N_b always equal to 4. The value of N_b depends upon the matrix size. If the first row is cyclically shifted left by 2 bytes (i.e., r = 2) and second row are cyclically shifted left by 1 bytes, i.e., r = 1). Then the Eq. 1 can be taken as:

Shift (0, 4) =2

Shift (1, 4) =1

Note that the element of s` is same as the element of s and that only their ordering changes when the state is transformed. Column shift operation can be formally expressed as follows:

$$S`_{r,c} = S_{r} + \text{shift}\ (_{r}, N_{b})\ \text{mod}\ N_{b,c} \tag{2}$$

Primary diagonal operation can be formally expressed as follows:

$$S`_{r,c} = S_{r\text{-shift}}\ (_{r,}\ N_{b})\ \text{mod}\ N_{b},\ c\text{-}\ \text{shift}\ (_{r}, N_{b})\ \text{mod}\ N_{b} \tag{3}$$

Secondary diagonal operation can be formally expressed as follows:

$$S`_{r,c} = S_{r\text{-shift}}\ (_{r,}\ N_{b})\ \text{mod}\ N_{b},\ c\text{+shift}\ (_{r,}\ N_{b})\ \text{mod}\ N_{b} \tag{4}$$

For decryption the inverted process can be performed i.e., shift row transposition can be expressed as
:

$$S_{r,c} + \text{shift}\ (_{r,}\ N_{b})\ \text{mod}\ N_{b} = s`_{r,c} \tag{5}$$

for $0\leq r\leq 3$ and $0\leq c\leq N_{b} = 4$. The shifting operation such as row, column and diagonal shift can be done depends upon the order of the states. The order of the shift changes depends upon the sender and receiver. This can be expressed in Fig. 4. The order of the shift can vary for each and every user i.e depends upon the sender and receiver. The order of the shift is passed between sender and receiver through secure channel. For row shift, the order to shift the row ($0\leq r\leq 3$) can vary for each row of the matrix similarly for column shift. But in diagonal shift operation can shift either left or right for 1 bytes. Finally, the outcome of the shifting operation can be converted in to ASCII format such as:

```
000110110100011000110000
100000000110111100011001
010001100111000111001000
011 1010000001110010000010
0000000110010100001000000
0011001100001101100000110
1101 00011011110000000000000
0000000
```

for the given text. Some of the encryption pattern can easily work out by cryptanalyst. To make the encryption more complex and unpredictable, it needs to take more number of matrix values for partition and also perform more layer of shifting operation. The cyclic shift transposition algorithm provides efficient authentication mechanisms for the sensitive files. The security has been enhanced in CSTA approach. The most dangerous threats such as denial of service and man in the middle attack cannot be entered in to the cloud storage system by using this cryptographic approach.

Data forward: The second phase of the work is to forward the data to the authorized user. The data owner can check the authorized user without using any third parties. This can be done by using one time verification system; here data owner can forward data to the suitable authorized user. One time verification system can avoid replay attacks. Data forward architecture can be shown in Fig. 5.

In data forwarding phase, user can forward their encrypted data to the receiver. If user B wants to get the information from user A (owner), then he sent the request to the data owner. User A checks their identity from storage server. If they are authenticated, then generate OTP and sent to user B. After receiving the OTP, user B sends the result to the owner, he matches the result with already generated OTP. If matches, then the user is authenticated.

If the user is authenticated, then owner forward the data to user B. It also allows the user B to decrypt the data. In order to ensure privacy, security and integrity of the confidential data OTP authentication services are used in the proposed system. OTP generation can be done by a pseudorandom number with a string of password. One time verification system is more secure than static password because it doesn't have chance to misuse the password. In order to improve the security of static password, we can change the password frequently (Kalaikavitha and Ganaselvi, 2013). By frequently changing the password, the user can easily forget the password, leading to very high administrative cost. In order to avoid this OTP is used. In OTP, for each time a new password is generated for each login session so that only trusted user can decrypt the data. So, it provides high data integrity and privacy. OTP is more reliable than the other and also it ensures safety. The main purpose of one time password is that there is no replay attack can occur and so it provides high data integrity.

Normally, end users often use the different password over the internet since it is difficult to keep in mind and also it is not really strong one. But OTP is a strong authentication protocol because it is not necessary to run the password. Typing an incorrect password would cause invalid until the next time step is reached-OTP is only valid once per time step. OTP is based on the very popular algorithm hashing. OTP schemes avoid the transmission of authentication secrets that are of any value after they have been used. This provides a high level of protection against password sniffing attacks. By continuously altering the password as is done in OTP, we can greatly reduce the risk of many attacks. The

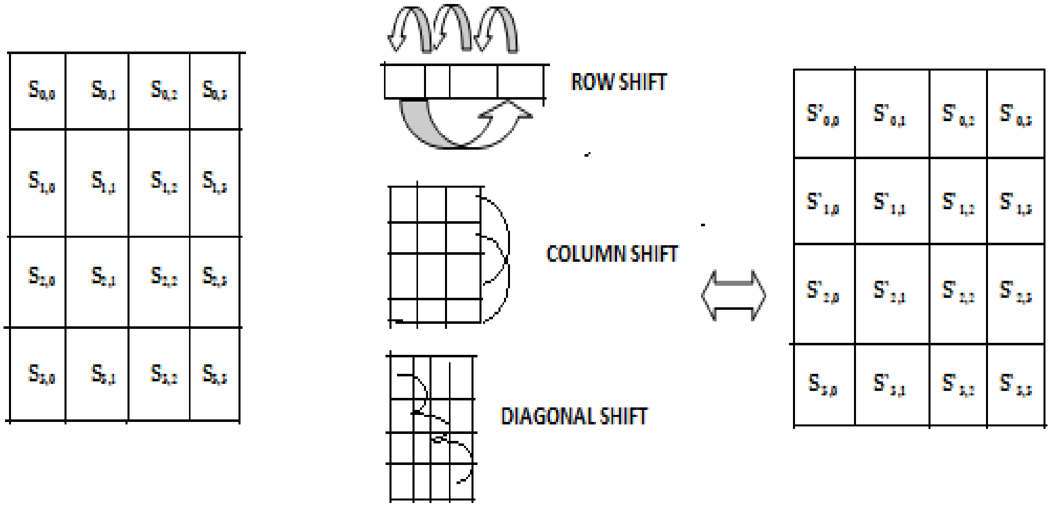
$S_{0,0}$ $S_{0,1}$ $S_{0,2}$ $S_{0,3}$
$S_{1,0}$ $S_{1,1}$ $S_{1,2}$ $S_{1,3}$
$S_{2,0}$ $S_{2,1}$ $S_{2,2}$ $S_{2,3}$
$S_{3,0}$ $S_{3,1}$ $S_{3,2}$ $S_{3,3}$
ROW SHIFT
COLUMN SHIFT
DIAGONAL SHIFT
$S'_{0,0}$ $S'_{0,1}$ $S'_{0,2}$ $S'_{0,3}$
$S'_{1,0}$ $S'_{1,1}$ $S'_{1,2}$ $S'_{1,3}$
$S'_{2,0}$ $S'_{2,1}$ $S'_{2,2}$ $S'_{2,3}$
$S_{3,0}$ $S'_{3,1}$ $S'_{3,2}$ $S'_{3,3}$

Fig. 4: Shifting encryption operation

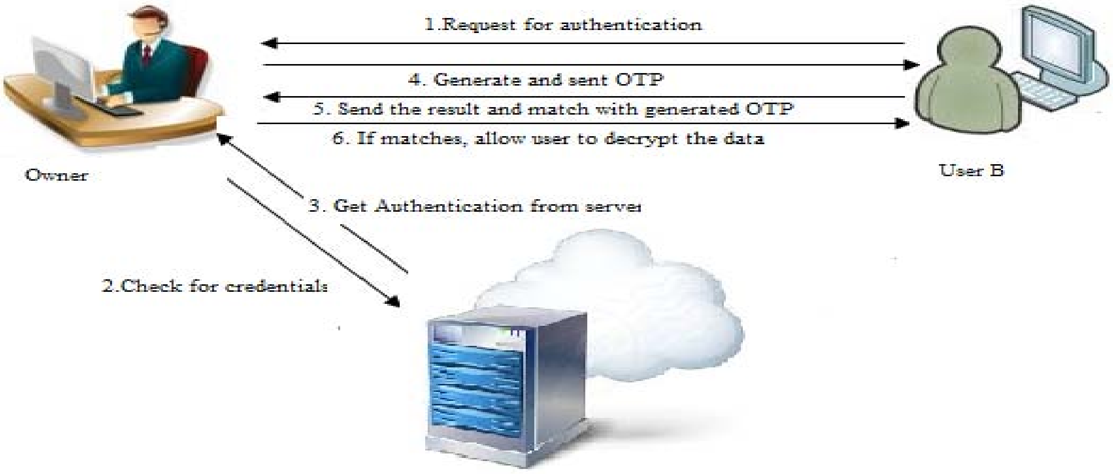
1.Request for authentication
4. Generate and sent OTP
5. Send the result and match with generated OTP
6. If matches, allow user to decrypt the data
Owner
User B
3. Get Authentication from server
2.Check for credentials

Fig. 5: Data forward mechanisms

comparative analysis of the proposed method with the existing method is shown in the Table 1 (Yang *et al*., 2013; Ruj *et al*., 2014; Zhao *et al*., 2011). The access control 1-W-M-R represents that the sharing of encrypted data can be created from the original owner. And the access control M-R-M-W shows that, after owner create one encrypted file on the storage server other user with suitable attributes can also update the encrypted file at a later file without any help from file's original owners. Therefore, user who has suitable access control policy can access the file.

Data retrieval: Sensitive information is protected by data encryption at data owner side. It is needed to ensure data integrity between owner and receiver. After data has been encrypted, a hash key is generated by using the hash based timestamp technique. Then for each encrypted hash code is signed by the owner in order to ensure authentication. This Hash based timestamp is used to check the integrity of the data. It enables data to be verified without any trusted third party.

For the purpose of validation, a timestamp is generated based on hash based timestamp process. The technique for certification based on timestamp is used to certify the document electronically which was created in certain date. Breaching of data privacy has been independently established by the parties. Keeping track of creation and processing time of digital data and proof for no change in data since data and time of time stamping can be established with time stamp.

Table 1: Comparative analysis of existing with proposed method

Scheme	Fine grained access control	Centralized/ decentralized	Access	Privacy preserving authentication	Confidentiality depends
Decryption of ABE Cipher text	Yes	Centralized	1-W-M-R	No Authentication	Decryption time and cipher text size
DAC-MACS	Yes	Decentralized	1-W-M-R	Not privacy preserving	Forward and backward
Fine grained flexible	Yes	Centralized	M-W-M-R	Authentication	Trusted authority
Anonyms authentication	Yes	Decentralized	M-W-M-R	Authentication	Trustee
OTP enabled Authentication and Authorization	Yes	Decentralized	M-W-M-R	Dual mode authentication	Data owner

At signing time the certificate of signature can be validated by timestamp. After being time stamped the document cannot be replaced or modified if a hash based timestamp is provided. The timeliness of critical transaction can be tracked with the help of time information in the timestamp. e.g., the creation, submission or delivery of a document The process of the hash based timestamp is shown as follows:

- Each owner (client) calculate the hash value for the encrypted file say h (E(F))
- The timestamp $\triangle$ t is generated based upon the current time and current date on the generation of the hash value
- Then owner sign the hash to the request form

$$r= (h (E (F)))_c \quad (6)$$

and send the request to the server.

- A copy of the ID, hash code, timestamp and metadata are stored in the data owner side
- When the user (receiver) requests the data, the data owner can verify the integrity of the outsourced data by requesting the server to send the hash code of the document
- Match the retrieved hash code with one already in the data owner side
- If hash code, timestamp and size match, then integrity is verified. Then send the data to the receiver

The original data can be retrieved between data owners and cloud server using hash based timestamp. Timestamp validates the data integrity of the document only for certain date and time. It ensures data verification and audit ability within the cloud environment (Rao *et al*., 2014). Hash function cryptography is used for data authentication for a proof based system designed with hash based timestamp. Data integrity, confidentiality, non-repudiation and ongoing assurance of authenticity can be achieved by hash based timestamp. Building a trusted third party server is very difficult, even though timestamp support third party. Unique third party server relying on time stamping scheme which cannot be trusted because of denial of service attack and server corruption. So the proposed scheme does not depend on any third party for data verification.

RESULTS AND DISCUSSION

Performance evaluation: This study explains the experimentation to examine the cloud security application. The experiment is written in Java Net bean to simulate the data storage, data forward and data retrieval in a cloud environment. The traditional algorithm needs some more amount of time because of key generation mechanisms and also it requires more computation of arithmetic operation where as CSTA does not require any complex computation. So that the performance of the CST algorithm is good and efficient. The CSTA is compared with the traditional encryption technique such as ECC by its processing time which is shown in Fig. 6. The benchmark shows that the CSTA algorithm gives better results than traditional method when it is implemented in net bean environment.

The overall performance of the proposed work scheme such as data confidentiality, data forward and data retrieval provides efficient and effective results. Figure 7 shows the overall processing time of the proposed work. The comparisons depend on the file size and processing time of the algorithm. The X axis represents the file size and the Y axis represents the processing time of the proposed work. The processing time of the proposed work varies for different file sizes which yield better results.

We run the experiment with different file size from 15 -50 kb. The processing time for the different file size is shown in the following tabulate column (Table 2). For each file size the proposed scheme provides better

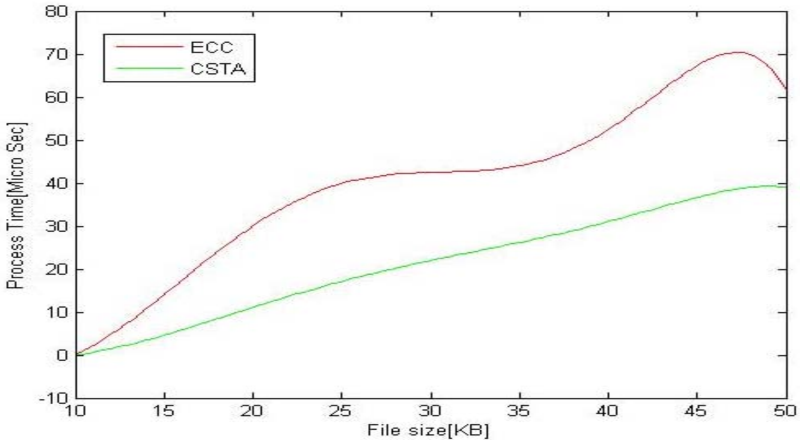

Fig. 6: Comparison between ECC and CSTA

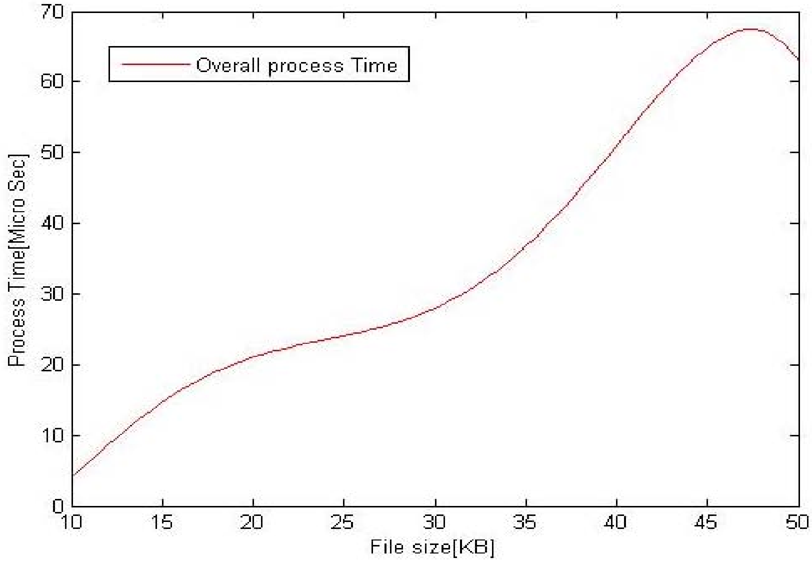

Fig. 7: overall processing time of the proposed work

Table 2: File size vs processing time

File size	Processing time
10	11
25	22
35	39
50	62

performance than the traditional method. The overall performance of the proposed scheme such as data storage, data forward and data retrieval is good and efficient.

CONCLUSION

Data security and Privacy are the major challenges in cloud computing for data storage and data forward in cloud computing. But in our proposed work provides secure data storage, data forward and data retrieval in a cloud environment. In this paper, the first step performs data encryption and decryption which ensures data security. The second step performs one time verification system. It provides strong security because they cannot be guessed or hacked. The third step performs a hash based timestamp in order to check the integrity of the data before retrieving it efficiently. The proposed work does not depend on any third party system. It provides high data integrity and privacy so that only trusted user can get the data in an efficient manner.

ACKNOWLEDGEMENTS

Ms. Neela. K.L. received her B.E. degree in Computer Science and Engineering in 2006 from Oxford Engineering college,Trichy and Master's degree in Computer Science and Engineering in 2008 from J.J. College of Engineering

and Technology, Trichy. She is working as Asst. Professor in University College of Engineering, Nagercoil, Tamil Nadu, India. Her field of interest is Network security and cloud computing.

Dr. V. Kavitha obtained her B.E degree in Computer Science and Engineering in 1996 from Norrul Islam College of Engineering and ME degree in Computer Science and Engineering in 2000 from Mepco Schlenk Engineering College. She received PhD degree in Computer Science and Engineering from Anna University Chennai in the year 2009. Right from 1996, she is in the Department of Computer Science and Engineering under various designations. Presently she is working as Associate Prof in the Department of CSE at University College of Engineering, Kancheepuram. Currently, under her guidance twelve research scholars are pursuing PhD as full time and part time.

Her research interests are wireless networks, mobile computing, network security, wireless sensor networks, image processing and cloud computing. She has published 5 national journal and 30 international journals in areas such as network security, mobile computing, wireless network security and cloud computing.

REFERENCES

Ajoudanian, S. and M.R. Ahmadi, 2012. A novel data security model for cloud computing. Int. J. Eng. Technol., 4: 326-326.

Baek, J., Q.H. Vu, J.K. Liu, X. Huang and Y. Xiang, 2015. A secure cloud computing based framework for big data information management of smart grid. IEEE. Trans. Cloud Comput., 3: 233-244.

Bhoyar, R. and N. Chopde, 2013. Cloud computing: Service models, types, database and issues. Int. J. Adv. Res. Comput. Sci. Software Eng., Vol. 3,

Bonnecaze, A., P. Liardet, A. Gabillon and K. Blibech, 2006. Secure Time-Stamping Schemes: A Distributed Point of View. Ann. Telecommun., 61: 662-681.

Chang, V. and M. Ramachandran, 2016. Towards achieving data security with the cloud computing adoption framework. IEEE. Trans. Serv. Comput., 9: 138-151.

Geetanjali, C. and A. Jainul,. 2014. Modified secure two way authentication system in cloud computing using encrypted one time password. Int. J. Comput. Sci. Inf. Technol., 5: 4077-4080.

Gomathi, S. and D. Manimegalai, 2014. Keyless cryptography in grid computing using cyclic shift transposition algorithm. J. Theor. Appl. Inf. Technol., 63: 112-118.

Gonzales, D., J. Kaplan, E. Saltzman, Z. Winkelman and D. Woods, 2015. Cloud-trust-a security assessment model for Infrastructure as a Service (IaaS) clouds. IEEE. Trans. J. Gonzales, 2015: 1-1.

Grossman, R.L., 2009. The case for cloud computing. IT. Prof., 11: 23-27.

Horwitz, J. and B. Lynn, 2002. Toward Hierarchical Identity-based Encryption. In: Advances in Cryptology, Knudsen, L. (Ed.). Springer, Berlin, Heidelberg, ISBN: 978-3-540-43553-2, pp: 466-481.

Inbarani, W.S., G. Shenbagamoorthy and C.K.C. Paul, 2013. Proxy re-encryption schemes for data storage security in cloud-a survey. Int. J. Eng. Res. Technol., Vol. 2,

Jegadeeswari, S., P. Dinadayalan and N. Gnanambigai, 2015. A neural data security model: Ensure high confidentiality and security in cloud datastorage environment. Proceedings of the 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), August 10-13, 2015, IEEE, Puducherry, India, ISBN: 978-1-4799-8790-0, pp: 400-406.

Kalaikavitha and J. Ganaselvi, 2013. Secure login using encrypted One Time Password (OTP) and mobile based login methodology. Int. J. Eng. Sci., 2: 14-17.

Kanade, M.A., M.R. Mule, M.M. Shuaib and N.M. Nagvekar, 2015. Improving cloud security using data partitioning and encryption technique. Int. J. Eng. Res. Gen. Sci., 3: 1245-1252.

Kant, D.C. and Y. Sharma, 2013. Enhanced security architecture for cloud data security. Int. J. Adv. Res. Comput. Sci. Software Eng., 3: 571-575.

Lin, H.Y. and W.G. Tzeng, 2012. A secure erasure code-based cloud storage system with secure data forwarding. IEEE Transact. Parallel Distributed Syst., 23: 995-1003.

Mambo, M. and E. Okamoto, 1997. Proxy cryptosystems: Delegation of the power to decrypt ciphertexts. IEICE Trans. Fund. Elect. Commun. Comput. Sci., E80-A/1: 54-63.

Mell, P. and T. Grance, 2011. The NIST definition of cloud computing. NIST Special Publication No. 800-145, National Institute of Standard and Technology, U.S. Department of Commerce, September 2011.

Mohamed, E.M., H.S. Abdelkader and E.S. Etriby, 2013. Data security model for cloud computing. J. Commun. Comput., 10: 1047-1062.

Murthy, S., 2014. Cryptographic secure cloud storage model with anonymous authentication and automatic file recovery. J. Soft Comput., 5: 844-849.

Nepal, S., R. Sinnott, C. Friedrich, C. Wise and S. Chen *et al.*, 2015. TruXy: Trusted storage cloud for scientific workflows. IEEE. Trans. Cloud Comput., 2015: 1-14.

Omar, M.N., M. Salleh and M. Bakhtiari, 2014. Biometric encryption to enhance confidentiality in cloud computing. Proceedings of the 2014 International Symposium on Biometrics and Security Technologies (ISBAST), August 26-27, 2014, IEEE, Johor Bahru, Malaysia, ISBN: 978-1-4799-6444-4, pp: 45-50.

Oppliger, R., 2011. Contemporary Cryptography. 2nd Edn., Artech House, Boston, Massachusetts, ISBN: 978-1-60807-145-6, Pages: 559.

Rajasekaran, S., 2013. Authentication based cloud storage and secure data forwarding. Int. J. Comput. Technol., 4: 106-110.

Rao, S., S. Gujrathi, M. Sanghvi and S. Shah, 2014. Analysis on data integrity in cloud environment. J. Comput. Eng., 16: 71-76.

Richa, C. and R. Satyakshma, 2013. One time password for multi-cloud environment. Int. J. Adv. Res. Comput. Sci. Software Eng., 3: 594-597.

Ruj, S., M. Stojmenovic and A. Nayak, 2014. Decentralized access control with anonymous authentication of data stored in clouds. Parallel Distrib. Syst. IEEE. Trans., 25: 384-394.

Selvi, R.K. and V. Kavitha, 2012. Crypto system based authentication using CSTA in grid. Int. J. Comput. Appl., 48: 45-51.

Sharma, S. and N.K. Khiva, 2013. Secure cloud architecture for preserving privacy in cloud computing using OTP/WTP. Global J. Comput. Sci. Technol., Vol. 13,

Singh, N. and G. Raj, 2012. Security on bccp trough AES encryption technique. Spec. Issue Int. J. Eng. Sci. Adv. Technol., 2: 813-819.

Sonali, A.W. and B. Tidke, 2015. Securely data forwarding and maintaining reliability of data in cloud computing. Int. J. Eng. Res. Appl., 5: 72-78.

Srinivasulu, A., C.D. Subbarao and A. Bhudevi, 2014. Dynamic data storage publishing and forwarding in cloud using fusion security algorithms. Comput. Sci. Inf. Technol., 2: 203-210.

Xia, Z., Y. Zhu, X. Sun, Z. Qin and K. Ren, 2015. Towards privacy-preserving content-based image retrieval in cloud computing. IEEE. Trans. Comput. Comput., 2015: 1-11.

Yang, K., X. Jia, K. Ren, B. Zhang and R. Xie, 2013. DAC-MACS: Effective data access control for multiauthority cloud storage systems. IEEE. Trans. Inf. Forensics Secur., 8: 1790-1801.

Zhao, F., T. Nishide and K. Sakurai, 2011. Realizing Fine-Grained and Flexible Access Control to Outsourced Data with Attribute-Based Cryptosystems. In: Information Security Practice and Experience. Feng, B. and J. Weng (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-642-21030-3, pp: 83-97.