

# Protection of User Data in Cloud Computing

Abhishek Sonar  
Department of Computer  
Science  
Vidyalankar Institute of  
Technology  
Mumbai, India

Prajwal Dubey  
Department of IT  
Vidyalankar Institute of  
Technology, Mumbai, India

Vineet Iyer  
Department of Computer  
Science  
SIES Graduate School of  
Technology  
Mumbai, India

## ABSTRACT

There has been an increase in the development of Cloud Computing, thus there is a need for people to know how their personal information be made more secure with the assistance of upcoming technology in the field of cryptography. The problem in the older systems can be defined as although applications are often created with unique features such as data tagging; whose main motive is to prevent unauthorized access to user data, illegal access to data is still possible through some types of application vulnerability (e.g., In March 2009 there was illegal access to Google's documents and spreadsheets). Some cloud providers get their data and various applications corroborated by third-party application security tools, still, there is a need for the establishment of a process that is dedicated to only one organization. In this paper, we have given more emphasis on Symmetric key algorithms e.g., AES (which stands for advanced encryption system), DES (which stands for data encryption system), Blowfish because the information which is stored on the cloud through these algorithms is much more secure and protected as compared to an asymmetric algorithm. In this journal, we have carefully studied security issues, which are faced by many users in the process of storage or transferring data through cloud computing from one place to another. We have presented a detailed and emblematic study of how can various encryption and decryption algorithms make the overall process of communication reliable, smooth, and secure. We have also compared important distinguishing factors between algorithms like key size, scalability, the extent of security, etc.

## Keywords

Algorithms, AES, RSA, 3DES, Blowfish.

## 1. INTRODUCTION

Cloud Computing is the process in which a particular company deploys a certain number of remote servers via the internet to use its wide range of computing services such as storing, managing, and processing data. The word "Cloud" means using the server of a third party system to host, process, and store data. Cloud computing is medium through which a particular user can have access to data infrastructure, computing, and services irrespective of the location of the user with the help of a concept called server virtualization. Cloud computing does the work of relocation, storage of data on virtual servers, and processing of computing resources on the user's side configuration, which is excluded from the control strategy at recipients i.e. 'customers' end.

Cloud computing offers various types of assistance to share assets like systems servers and administrations with numerous individuals utilizing virtualization of assets. Cloud computing can minimize the cost of maintenance and administration of hardware and software resources. These services can be used

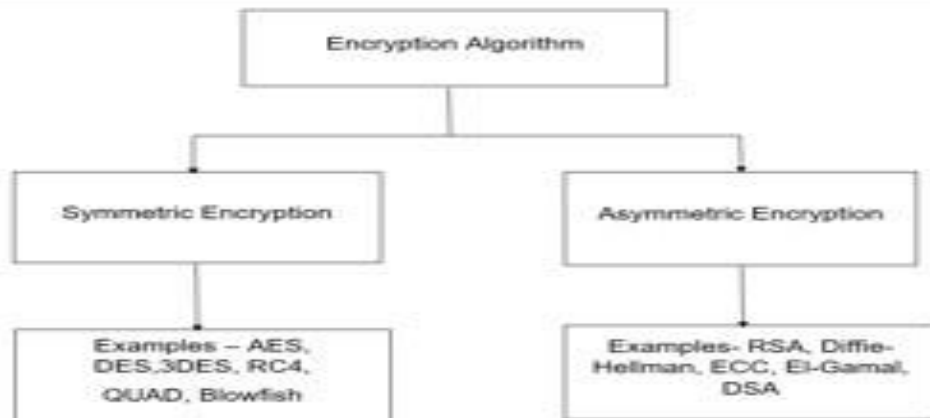
from any location and it does not require users to install any applications on their system for accessing their files. the deployment models public cloud can be accessed by any individual or organization, private cloud service is offered to a specific organization and only members of that organization can access it. This makes it less susceptible to security attacks than public cloud.

At whatever point correspondence over a system is considered there is a need to ensure that there is a type of security for the data being conveyed. The data should be secured when it is going through the system. Cryptography is a field of system security that manages to cover up "genuine" information when it is under transmission between two gatherings. For the most part, the genuine data is changed or covered up into another message and transmitted over the system. This changed message in itself will have neither rhyme nor reason regardless of whether any programmer gets hold of this data. At the point when it arrives at the goal, the recipient will know a technique to de-change the trash message into its unique message, which was sent by the Sender. This technique for changing messages next to the sender and de-changing next to receiver structures the essential model of Cryptography. Changing over genuine data into what resembles trash esteem this procedure is called as Encryption. Separating genuine data once again from this futile content is called as Decryption. The data to be transferred is called as plain text (or message) is taken care of to an Encryption framework. The Encryption framework utilizes a key to change over the plain content to an encrypted structure which appears as though trash esteem. This is likewise called as figure text. A relating key is utilized at the opposite end to decode the figure text back to a unique message. When we state a "key" it implies a bit of string esteem which is taken care of encryption and unscrambling calculations alongside the content for a change. This is like securing your important things a crate and sending it across. At opposite end, receiver will utilize mystery key to unlock the container and extract the message you (Sender) sent. If a programmer were to tap out the message being transmitted in the system any outside source will get the scrambled message. Any outer source won't have the way to decode this message so this figure text won't mean anything to them. They may utilize a few strategies to break this code and get the concealed message out of. This craft of attempting to break figures frames an alternate part of the study called Cryptanalysis. This Encryption and Decryption together guarantee the security of the message being transmitted over the system. This entire encryption and unscrambling technique depends on the reason that both sender and receiver share something of a kind key which isn't known by any outcasts, similar to the programmers. Depending upon how the keys are shared we can characterize cryptography as Symmetric and Asymmetric.

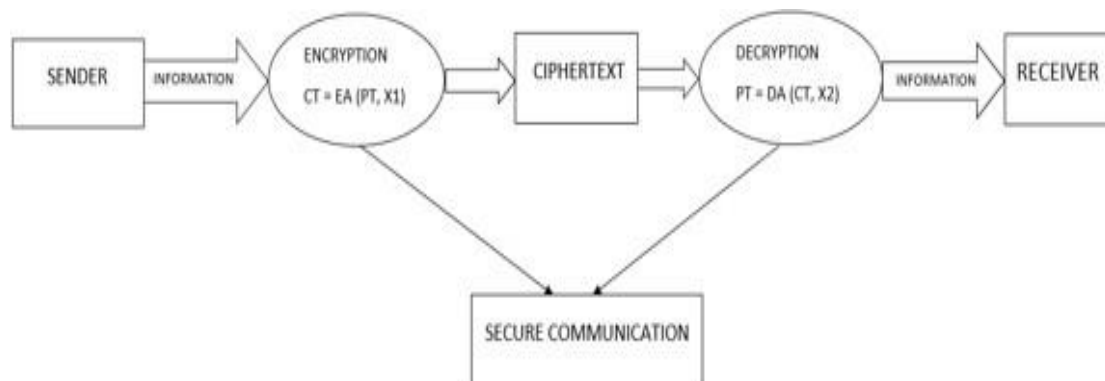
“In simple terms, Cryptography can be seen as a method of storing and disguising confidential data in a cryptic form so that only those for whom it is intended can read it and can communicate information in the presence of an adversary and the security algorithms mitigate security issues by use of cryptography, authentication and distributing keys securely”[1].

In technical language, Cryptography “is the phenomenon of  $CT = EA \{PT, key(x1)\}$  and  $PT = DA \{CT, key(x2)\}$ .

converting user text into cipher text”[9] (which is an unreadable format of plain text) and converting back that cipher text into plain text using certain secure algorithms. Let (PT) be the information i.e. plain text, which is sent by the user, (CT), be the cipher text, (EA) be the encryption algorithm, (DA) by the decryption algorithm and  $x1$  and  $x2$  be the cryptographic keys. The entire process of cryptography can be mathematically represented using cipher text and plain text as-



**Figure 1: Different types of algorithms**



**Figure 2: Cryptography**

Definition of common terminologies used in the process of encryption and decryption:

- 1- Plaintext-Plaintext is the original comprehensible information or data which is” [8] sent by the sender to the receiver. The plain text acts as an input to various encryption algorithms.
- 2- Ciphertext-Ciphertext is the jumbled message (incomprehensible data) containing various characters and digits. It is the output that one gets after plaintext is given as input to the encryption algorithm.
- 3- Encryption Algorithm-Encryption algorithm performs various permutations, operations, and converts “plain text (comprehensible data) into cipher text (incomprehensible

data) with” [9] the help of encryption key.

- 4- Decryption Algorithm-“Decryption Algorithm is the process of converting cipher text (incomprehensible data) into plaintext (comprehensible data) with the help of decryption key” [9].
- 5- “Keys-Keys are considered as input for encryption or decryption and decides the degree of change.
- 6- Sender and Receiver-They are individuals who are participating in communication and exchanging the plaintext” [8].

Advantages of cryptography-

1. Confidentiality- Data transfer confidentially takes

place .no one can access the information between the confidential areas.

2. Authentication -The message authentication code under digital signatures can make data secure for forgeries and spoofing attack
3. Data Integrity -The hash function performs a major role in maintaining the data integrity by creating high standards for accuracy and consistency.

## 2. SECURITY ISSUES AND CHALLENGES OF CLOUD COMPUTING

1. Secure data transfer-In cloud computing, private companies hire various third-party cloud service organizations to maintain, store, and relocate information and data. There is no account of where the actual data of the company goes.
2. Secure Software Interfaces-“Guaranteeing the honesty of the information (move, stockpiling, and recovery) truly implies that it changes just because of approved exchanges. A typical standard to guarantee information trustworthiness doesn't yet exist” [2].
3. Data Separation-“Client might have the option to utilize cloud specialist organizations if security rights are abused, and regardless the cloud specialist co-ops may confront harm to their notoriety. Concerns emerge when it isn't obvious to people why their data is mentioned or how it will be utilized or given to different gatherings” [2].
4. Secure Stored Data-It should be the customer or client who should have control over encryption/decryption keys.
5. User Access Control-“If there should be an occurrence of Payment Card Industry Data Security Standard (PCI DSS) information logs must be given to security troughs and control”[5].

## 3. PROPOSED WORK PLAN

Various encryption and decryption algorithms like AES, DES, 3DES, Blowfish can be used to make communication more secure.

## 4. SECURITY ALGORITHMS USED IN CLOUD COMPUTING

### 4.1 RSA ALGORITHM

RSA is the “most popular and proven asymmetric key cryptography algorithms .so the main crux of the RSA algorithm is on the mathematical fact that it is easy to find and multiply large prime numbers together but it is extremely difficult to factor their products”[3].In asymmetric key cryptography “every user has its public key and private key so one key is public which is available to everyone on the network and one is private”[7] .Let us suppose A has his own “two keys, the public key and private key”[10] and B has his own sets of keys. A takes the user text and encrypts it using his public key. Encryption is performed using this key Cipher text is sent on the internet to B. When this cipher text is received at the B's end what it does is it takes his private key and then grips it to get the plaintext. There are two different keys involved one is the public key of A “for encryption and private key of B for decryption. Similarly, the” [11] other way around if B had to dispatch an information to A. B would take the plaintext. B would take his public.

```

1 import java.io.BufferedReader;
2 import java.io.IOException;
3 import java.io.InputStreamReader;
4 import java.io.PrintWriter;
5 import java.net.Socket;
6
7 public class ServerThread extends Thread {
8     private Server server;
9     private BufferedReader bufferedReader;
10    private PrintWriter printWriter;
11
12    public ServerThread(Socket socket, Server server) throws IOException {
13        this.server = server;
14        this.bufferedReader = new BufferedReader(new InputStreamReader(socket.getInputStream()));
15        this.printWriter = new PrintWriter(socket.getOutputStream(), true);
16    }
17
18    void forwardMessage(String message) { printWriter.println(message); }
19
20    public void run() {
21        JSONObject jsonObject = null;
22        try {
23            while(true) {
24                jsonObject = json.createReader(bufferedReader).readObject();
25                System.out.println("System: " + jsonObject.toString());
26                if (jsonObject.containsKey("n")) {
27                    System.out.println("The: " + jsonObject.getString("n") + " public key (n, e) = (" +
28                        jsonObject.getString("n") + ", " + jsonObject.getString("e") + "). ");
29                    System.out.println("need private key d.");
30                    System.out.println("where d's components i and phi(n) ");
31                    System.out.println("but don't have phi(n). ");
32                    System.out.println("To obtain it, can use formula: phi(n) = (p-1)(q-1)");
33                    System.out.println("where p*q = n & both is are primes");
34                    System.out.println("i.e. need to do prime factorization of n into p & q");
35                }
36                server.forwardMessage(jsonObject.toString(), this);
37            }
38        } catch (Exception e) { server.getServerThreads().remove(this); }
39    }
40 }

```

Figure 3: Implementation of RSA Algorithm.

```

1 package aes;
2
3
4 import java.util.Base64;
5
6 import javax.crypto.Cipher;
7 import javax.crypto.KeyGenerator;
8 import javax.crypto.SecretKey;
9
10 public class aes {
11     static Cipher cipher;
12
13     public static void main(String[] args) throws Exception {
14         KeyGenerator keyGenerator = KeyGenerator.getInstance("AES");
15         keyGenerator.init(128);
16         SecretKey secretKey = keyGenerator.generateKey();
17         cipher = Cipher.getInstance("AES");
18
19         String plaintext = "AES Symmetric Encryption Decryption";
20         System.out.println("Plain Text Before Encryption: " + plaintext);
21
22         String encryptedText = encrypt(plaintext, secretKey);
23         System.out.println("Encrypted Text After Encryption: " + encryptedText);
24
25         String decryptedText = decrypt(encryptedText, secretKey);
26         System.out.println("Decrypted Text After Decryption: " + decryptedText);
27     }
28 }

```

Figure 4: Implementation of AES Algorithm.

### 4.2 Aes Algorithm

“Advanced Encryption Standard (AES) algorithm not only for security but also for great speed. Both hardware and software implementation is faster still. New encryption standards

recommended by NIST to replace DES. Encrypts data blocks of 128 bits in 10, 12 and 14 round depending on the key size. It can be implemented on various platforms especially in small devices. It is carefully tested for many security applications” [9].

### 4.3 Des Algorithm

DES algorithm is mostly used for data protection in the field of computer cryptography. It is of the type symmetric block cipher, which means it is employed on constant length of group of bits. Identical pair of keys are utilized in this process. The size of the DES key starts from 0 bit to 63 bits and it is divided into 2 parts. The first half has 56 bits and the second half has 8 bits. DES consists of 64 binary digits zeros and ones. “The 56-bit keys are generated randomly and used by the DES algorithm. One byte (eight bits) is used for error detection and these eight bits are not used by the algorithm in the processing” [7]. In DES algorithm, the very first step is encryption process. In the encryption process, 64-bit plaintext acts as input to DES block cipher. The output of DES block cipher is 64-bit cipher text. The text, which is in a readable format, is blended with the text, which is in an unreadable format. In decryption, 64-bit text at recipient’s end is transformed back into readable text by the key, which was initially employed for encryption.

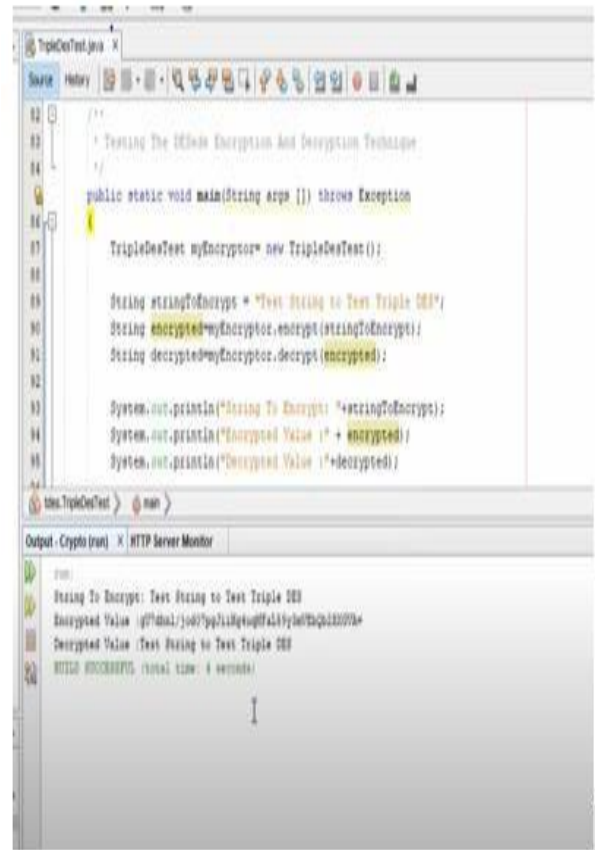


```
65     return stringBuffer.toString();
66 }
67
68 public static void main(String args[]) throws Exception {
69     DESExample myEncryptor = new DESExample("This is key");
70     String stringToEncrypt = "Welcome to the world of Java";
71     String encrypted = myEncryptor.encrypt(stringToEncrypt);
72     String decrypted = myEncryptor.decrypt(encrypted);
73     System.out.println("String To Encrypt: " + stringToEncrypt);
74     System.out.println("Encrypted Value : " + encrypted);
75     System.out.println("Decrypted Value : " + decrypted);
76 }
77 }
```

Output - DESExample (run) | Tasks | HTTP Server Monitor

```
7:00:
String To Encrypt: Welcome to the world of Java
Encrypted Value : 1F200B81e1626e9607349c579b0121Dc/vs/080e+
Decrypted Value : Welcome to the world of Java
BUILD SUCCESSFUL (total time: 0 seconds)
```

Figure 5: Implementation of DES Algorithm.



```
82  /**
83   * Testing The 168bit Encryption And Decryption Technique
84   */
85  public static void main(String args[]) throws Exception
86  {
87      TripleDesTest myEncryptor = new TripleDesTest();
88
89      String stringToEncrypt = "Test String to Test Triple DES";
90      String encrypted = myEncryptor.encrypt(stringToEncrypt);
91      String decrypted = myEncryptor.decrypt(encrypted);
92
93      System.out.println("String To Encrypt: " + stringToEncrypt);
94      System.out.println("Encrypted Value : " + encrypted);
95      System.out.println("Decrypted Value : " + decrypted);
96  }
```

Output - Crypto (run) | HTTP Server Monitor

```
7:00:
String To Encrypt: Test String to Test Triple DES
Encrypted Value : 9ff0a017c0d79a118e4e9ffA4194e07c0d3079+
Decrypted Value : Test String to Test Triple DES
BUILD SUCCESSFUL (total time: 4 seconds)
```

Figure 6: Implementation of 3DES Algorithm

### 4.5 3DES ALGORITHM

In triple DES, a plain text message P is going to use a key K1 to encrypt resulting in cipher text C1

To the output of the first round of encryption a second key K2 is applied. With second key a decryption process is applied on C1. Since it is the wrong, key plaintext is not revealed out. Thus the output will be another round of cipher text C2. On C2 a third key K3 is applied and which is going to encrypt the cipher text C2. This will result in another round of cipher text C3. As a result three different keys applied in two different ways so with key1 and key3 there is a round of encryption and with K2 there is a round of decryption. It is an encryption-decryption- encryption process with three different keys. For Triple DES the DES block cipher with 56-bit keys is still used but since there are three discrete keys the overall length of a key becomes 21 bytes (168 bits).

### 4.6 BLOWFISH ALGORITHM

“Blowfish is a symmetric block cipher that can be effectively used for encryption and safeguarding of data. It takes a variable-length key, from 32 bits to 448 bits, making it ideal for securing data. Blowfish was designed in 1993 by Bruce Schneier as a fast, free alternative to existing encryption algorithms. Blowfish is unpatented and license-free and is available free for all uses. Though it suffers from weak keys problem, no attack is known to be successful against it” [4].



Initial vector size	128 bits	1024 bits	64 bits	64 bits
Security	Secure for both provider and user	Secure only for user	Secure for both provider and user	Secure for both provider and user
Data Encryption Capacity	Large Data	Small Data	Less than AES	Less than AES
Authentication Type	Best Authenticity Provider	Robust Authentic Implementation	Almost same as AES	Less Authentic than AES
Memory Usage	Low RAM needed	Highest Memory Usage	Executes in less than 5 kb	More than AES
Execution Time	Faster	Requires Maximum Time	Less Time than AES to Execute	Equal to AES

## 6. CONCLUSION AND FUTURE SCOPE

The networked world opens up a completely new avenue of services and features that were not possible before to get a better life. Cloud computing is an emerging technology when it comes to providing services such as data storage, transfer, and processing. Many organizations, public and private sectors, enterprises already have their setup established on a cloud server or are moving towards the cloud. This has given rise to many risks, security issues, and challenges associated with cloud computing like data breaching, data transfer, thefts. Third party organizations can get into databases of big companies and extricate personal information that belongs to their clients. Once they can steal the user credential, they can impersonate that user to many different other sources on the web. Encryption algorithm has a huge impact on data protection and security. In this journal, we have reviewed different encryption-decryption techniques like AES, DES, RSA, 3DES, blowfish for “data security and privacy, emphasizing on data storage and use in the cloud, for data protection in the cloud computing environments and to build trust between cloud service providers and users”[6].

## 7. REFERENCES

- [1] Lena Khanna, Anant Jaiswal, “Cloud Computing: Security Issues and Description of Encryption Based Algorithms to Overcome Them”, IJARCSSE 2013.
- [2] Manoj, K. Sai, K. Mrudula, and K. phani Srinivas. "Risk Factors And Security Issues In Various Cloud Storage Operations.", IJITEE 2019.
- [3] Singh, S., Maakar, S.K., and Kumar, S., 2013. A Performance analysis of DES and RSA cryptography. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), 2(3).
- [4] Thakur J, Kumar N. DES, AES, and Blowfish: Symmetric key cryptography algorithms simulation- based performance analysis. International journal of emerging technology and advanced engineering. 2011 Dec 12;1(2):6-12.
- [5] Ertaul, Levent, Sarika Singhal, and Gökay Saldamli. "Security Challenges in Cloud Computing." In Security and Management, pp. 36-42. 2010.
- [6] Rao, Ch Chakradhara, and A. V. Ramana. "Data security in cloud computing." Parameters 2, no. 04 (2016).
- [7] Grover, Nidhi. "A Study of Security Threats and Issues in Cloud Computing." IITM Journal of Management and IT 5, no. 1 (2014): 78-86.
- [8] S. R. Pardeshi, V. J. Pawar, and K. D. Kharat, "Enhancing information security in a cloud computing environment using cryptographic techniques," 2016 International Conference on Communication and Electronics Systems (ICCES), Coimbatore, 2016, pp. 1- 5, doi: 10.1109/CESYS.2016.7890004.
- [9] Mahajan, Prerna, and Abhishek Sachdeva. "A study of encryption algorithms AES, DES, and RSA for security." Global Journal of Computer Science and Technology (2013).
- [10] T. Mantoro, Laurentinus, N. Agani, and M. A. Ayu, "Improving the security guarantees, authenticity and confidentiality in short message service of mobile applications," 2016 4th International Conference on Cyber and IT Service Management, Bandung, 2016, pp. 1-6, doi: 10.1109/CITSM.2016.7577592.
- [11] Swapnil B. Khalekar. , Himesh Kishore. , Aniruddha Desai., 2014, Highly Secured Web Portal : An Illustration of Enhanced Web Security, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 03, Issue 03 (March 2014).