Enhanced Cloud Security using Hybrid Mechanism of RSA, AES and Blowfish Data Encryption with Secure OTP

Gurjeet Singh [1], Dr. Mohita Garg[2]

[1]Research Scholar, Department of Computer Engineering, NWIET, Moga

[2] Associate Professor, Department of Computer Science and Engineering, NWIET, Moga

mrgurjeet93@gmail.com, mohita_cse@northwest.ac.in

ABSTRACT

Cloud Computing has become one of the most talked about technologies in recent times and has got lots of attention from media as well as analysts because it is offering lots of opportunities. Enterprises have been determined to reduce computing costs and for that reason most of them started using it in IT technology then adapted virtualization technology. Cloud computing has revolutionized the way computing and software services are delivered to the clients on demand. It offers users the ability to connect to computing resources and access IT managed services with a previously unknown level of ease. Thus, security concerns among users of the cloud have become a major barrier to the widespread growth of cloud computing. In this research work, we have used the 3 step security mechanism for the keeping the data secure at the cloud. We have implemented the strong authentication mechanism using md5 encrypted OTP and enhanced the security of data using Cloud Broker and RSA, Blowfish and AES. When you log on to your machine and then try to access a resource, say a file server or database, something needs to assure that your username and password are valid. With sensitive data stored in the cloud of the different users, we need a strong authentication mechanism along with OTP. Data breaches because of no/weak authentication. Afterwards we have verified the integrity of data stored at cloud provider using SHA2. Multiple parameters like processing time, processing cost, AES, RSA and blowfish encryption time, OTP generation and encryption time have been calculated and analyzed. We have been able to enhance the security by optimizing the processing time as well as processing cost. After implementing the proposed methodology, it has been summarized that the cloud security can be enhanced by applying the proposed mechanism. The proposed system has reduced the complexity, processing cost which increases the overall efficiency of the system.

KEYWORDS: Cloud Computing, Cloud Security, Security issues, OTP, AES, RSA, Blowfish, and Hashing

INTRODUCTION

Cloud Computing has become one of the most talked about technologies in recent times and has got lots of attention from media as well as analysts because it is offering lots of opportunities. Enterprises have been determined to reduce computing costs and for that reason most of them started using it in IT technology then adapted virtualization technology. For the good of the enterprises it is futuristic to help them in this i.e. Cloud Computing. Cloud Computing has taken the enterprise to new level and allows them to further reduce costs through improved utilization, reduced administration and infrastructure cost and faster deployment cycles. Cloud Computing is a term used to describe both a platform and type of application. As a platform it supplies, configures and reconfigures servers, while the servers can be virtual machine or physical machine. The cloud is a representation for the Internet and is an abstraction for the complex infrastructure it conceals. There are some important points in the definition to be discussed regarding Cloud Computing. Cloud Computing differs from traditional computing paradigms as it is scalable, can be encapsulated as an abstract entity which provides different level of services to the clients, driven by economies of scale and the services are dynamically configurable. Different researchers have stated various benefits of cloud computing due to this reason they have been adopted by enterprises more preferable. Cloud Computing infrastructure allows enterprises to

achieve more efficient use of their IT hardware and software investments. This is achieved by breaking down the physical barrier inherent in isolated systems, automating the management of the group of the systems as a single entity. Cloud Computing can also be termed as virtualized system and a natural evolution for data centers which offer automated systems management. Security controls in cloud computing are similar to those in traditional IT environments. However, because of the cloud service and operational models employed with the implied organizational division of responsibilities and the technologies used to enable cloud services, cloud computing may present different risks to an organization than traditional IT solutions. As part of the transition to cloud computing, it is critical that consumers understand their level of risk tolerance and focus on mitigating the risks that the organization cannot afford to neglect. Often it is not understood that the type of service model being offered by the provider (i.e. IaaS, PaaS or SaaS) has significant impact on the assumed "split of responsibilities" between the consumer and the provider to manage security and associated risks. For IaaS, the provider is supplying (and responsible for securing) basic IT resources such as machines, disks and networks (Buvya et al., 2002). The consumer is responsible for the operating system and the entire software stack necessary to run applications, plus the data placed into the cloud computing environment. As a result, most of the responsibility for securing the applications themselves and the data they use falls onto the consumer. In contrast, for SaaS, the infrastructure, software and data are primarily the responsibility of the provider, since the consumer has little control over any of these features of the service.

RELATED WORK

In order to assess the trend and level of research work done till date, in the area of titled work, an exhaustive literature has been reviewed. A gist of some of the most relevant research work is presented in this chapter under various classified headings. Several books and entities have covered for the last years the concept of cloud computing. It is a hot topic nowadays in the technology and business world; thus there are multiple definitions. The National Institute of Standards and Technology (NIST), provides a well-recognized description for cloud computing (Harold et al., 2009), including its characteristics, service models and deployments models. T. Lindeberg (1998) portrays the different security issues of distributed computing because of its administration conveyance models. In any case, the hidden innovation of cloud without anyone else gives a noteworthy security hazard. Buyya R, Murshed M (2002) talk about the security and protection concerns of cloud computing and some conceivable answers for improve the security. In light of the security arrangements proposed we have concocted a secured structure for distributed computing. In today's worldwide focused business, organizations must improve and take full advantage of its assets to succeed. This obliges empowering its representatives, business accomplices, and clients with the stages and coordinated effort devices that advance development. L.Wang, Gregor Laszewski present a novel technique to hide data in the edges of the image by extending the Least Significant Bit embedding algorithm. This algorithm hides data in the edge pixels and thus ensures better security against attackers. In the Least Significant Bit embedding algorithm (LSB) and Random Least Significant Bit embedding algorithm (RLSB) an attacker can easily detect the presence of hidden image. To overcome these problems a new algorithm is proposed based on least significant bit embedding algorithm (LSB) for hiding secret messages in the edges of the image. The algorithm ELSB hides data in edge pixel. The proposed algorithm is applicable to all kinds of images and can be used in covert communication, hiding secret information like copyrights, trade secrets and chemical formulae.R. Maggiani (2009) listing out the security issues and challenges in cloud environment, the security standards and management tools which are in place and recommended the best solutions which we can rely on. Cloud computing provides scalable and efficient means to manage IT resources in organizations. The flexibility the cloud brings in has some disadvantages over privacy and security. If the providers and consumers follow the security measures discussed above cloud computing will be more secure. As and when the issues around security and privacy are elucidated cloud computing will be accepted widely. Harold C. Lin (2009) proposes an image steganography technique based on the canny edge detection algorithm. It is designed to hide secret data into a digital image within the pixels that make up the boundaries of objects detected in the image. More specifically, bits of the secret data replace the three LSBs of every color channel of the pixels detected by the canny edge detection algorithm as part of the edges in the carrier image.

Kapil Bakshi (2009) discuss the strategy, architecture, and solution details that Cisco brings to the industry and governments. For the purposes of this paper, we will focus on the data center aspects of cloud computing. The intended audience for this paper includes public managers, government executives, IT decision makers, and IT professionals who are evaluating cloud computing strategy and cloud data center solutions. Torry harries (2009) aims to provide a means of understanding the model and exploring options available for complementing your technology and infrastructure needs. . The idea of cloud computing is based on a very fundamental principal of „reusability of IT capabilities'. The difference that cloud computing brings compared to traditional concepts of "grid computing", "distributed computing", "utility computing", or "autonomic computing" is to broaden horizons across organizational boundaries. Resource sharing in a pure plug and play model that dramatically simplifies infrastructure planning is the promise of „cloud computing". The two key advantages of this model are ease-of-use and cost-effectiveness.

GAP ANALYSIS

In order to avail the benefits of cloud, the security of data being transferred between the client and user must be ensured. Security is the key for the Cloud success, security in the cloud is now the main challenge of cloud computing. Until a few years ago, all the business processes of organizations were on their private infrastructure and, though it was possible to outsource services, it was usually non-critical data/applications on private infrastructures. Now with cloud computing, the story has changed. The traditional network perimeter is broken, and organizations feel they have lost control over their data. New attack vectors have appeared, and the benefit of being accessible from anywhere becomes a big threat. After studying the existing papers, it is analyzed that the existing techniques are not capable of protecting data. There are various policies issues and threats in cloud computing technology which include privacy, storage, reliability, security, capacity and more. But most important among these to concern is security and how service provider assures it to maintain. Generally cloud computing has several customers such as ordinary users and enterprises who have different motivations to move to cloud.

Various concerns after analyzing the problems in cloud Computing are: security, integrity, loss of data and third party access.

i. After studying the existing paper [22], it is analyzed that the existing techniques are not capable of protecting data in an efficient way.

ii. For the data integrity, the data can be changed in way before reaching to the server/client. There is no data verification involved.

iii. Unauthorized person can come to know about methodology.

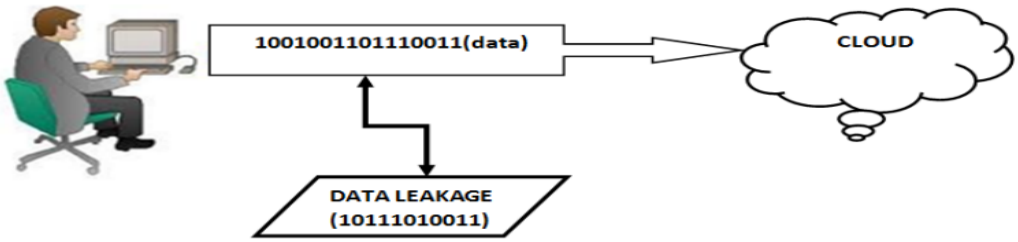

Figure 1. Man in the middle Attack.

PROBLEM FORMULATION

No secure authentication: In the present work, there is no secure authentication procedure defined. When you log on to machine and then try to access a resource, say a file server or database, something needs to

assure that username and password are valid. With sensitive data stored in the cloud of the different users, a strong authentication mechanism along with OTP will be needed. Data breaches because of no/weak authentication.

No Gateway is defined: The user should not be directly connected to the cloud provider as there is high risk of data getting stolen or hacked by the third party intruder. There is a requirement of gateway/broker that acts as an intermediate between the cloud provider and the client.

Weak Encryption Mechanism: In the present work, only one encryption algorithm is chosen i.e. Blowfish for encryption of data at the client's end.

RESEARCH OBJECTIVES

- To implement and study the performance of existing security mechanisms in cloud environment.
- To implement the strong authentication mechanism using encrypted OTP (One-Time Password).
- To enhance the security of data using 3-tier architecture and hybrid combination of RSA and AES and blowfish.
- To develop the proposed algorithm and compare the performance of proposed algorithm with existing algorithm

PROPOSED METHODOLOGY

This thesis aims to provide an understanding of the different attack vectors created by multi-tenancy and virtualization in a public IaaS cloud. The vectors will be explored, focusing on the threats arisen from different tenants coexisting in the same physical host. A critical analysis of the different vectors will be provided along with guidance on how to approach them. This analysis will be performed using previous works from different entities and authors, along with personal knowledge obtained from experience. As part of the aim of this research, a strong foundation will be provided on the terms of cloud computing, multi-tenancy and virtualization. All these areas will be explored giving a strong definition. The different security issues will be also explored in order to provide an introduction to the main focus of the research. The research work is divided into 3 phases:

i. Phase 1: Secure Authentication.

ii. Phase 2: Encryption of File using RSA, AES and Blowfish.

iii. Phase 3: Decryption of file after verification.

STEPS INVOLVED IN PHASE 1

i. Study of existing security mechanisms in cloud computing

ii. Choice of the cloud provider

iii. Registration of Client with the cloud provider

iv. Login of client into the cloud

v. Generation of OTP

vi. Encryption of OTP using MD5

vii. OTP verification

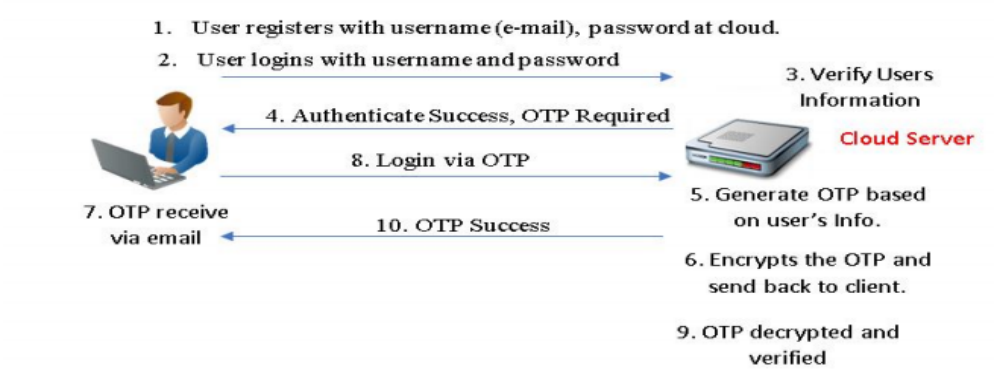

Figure 2. Secure Login using Encrypted OTP Authentication

- Client registers with the cloud provider by providing his/her details like ID, Password, Name etc.
- The cloud stores the credentials of the user in its database and displays the confirmation message.
- After successful registration, the client logins into the system by entering his/her username and password.
- The cloud verifies the ID and password and generates the error message if they are wrong.
- After successful verification, the cloud will generate the OTP based on user's information like name, password, Id etc.

STEPS INVOLVED IN PHASE 2

- Client chooses a file that has to be uploaded to the cloud provider.
- After choosing the file, the client performs the RSA encryption process to prevent the data from Man-In-the-Middle attack.
- The encrypted data is sent to the cloud broker/gateway. The gateway acts as an intermediate between the cloud provider and the user. The clients are connected with the gateway and the gateway is further connected with the cloud provider.
- The gateway receives the encrypted file from the client and further performs the AES encryption on it.
- After encrypting the data, the gateway sends the file to the cloud provider for storage.
- The cloud provider receives the file and performs blowfish encryption and generates the Hash value using SHA2.
- The hash key is send back to the client for future purposes.

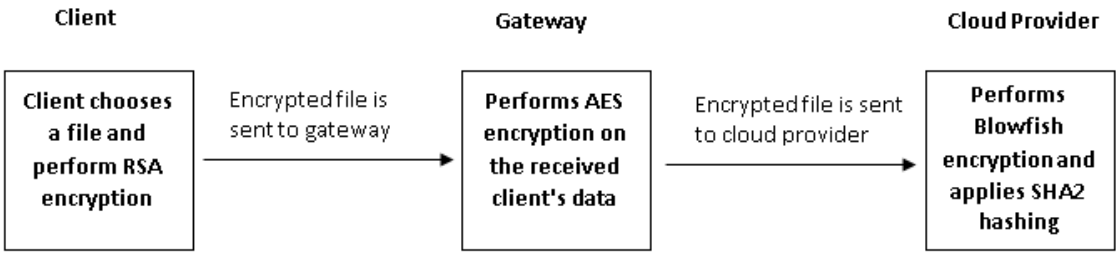

Figure 3. Encryption Process in Cloud Environment

STEPS INVOLVED IN PHASE 3

- Client sends the request to the cloud provider for his/her file via broker.
- The cloud provider generates the new hash value of the requested file and verifies the new key with the previously generated value.
- If the keys are matched, then the cloud provider will apply blowfish decryption. The downloading process will begin or else the modifications have been made to the file and error message is generated.
- In the downloading process, the cloud will send the file to the cloud broker.
- The broker will receive the file and will apply the AES decryption mechanism. The decrypted file is sent to the client.
- The client receives the file and will further apply the RSA decryption.

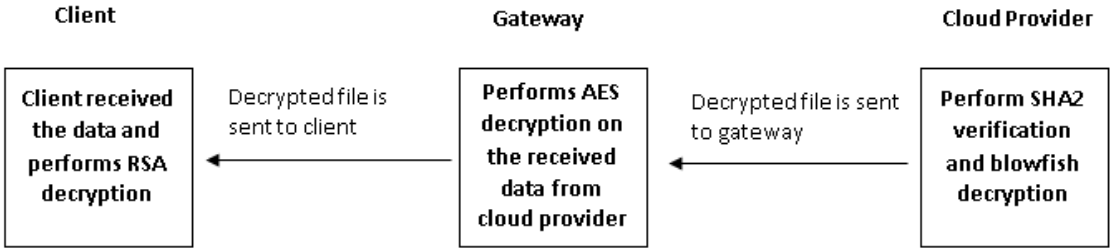

Figure 4. Decryption Process in Cloud Environment

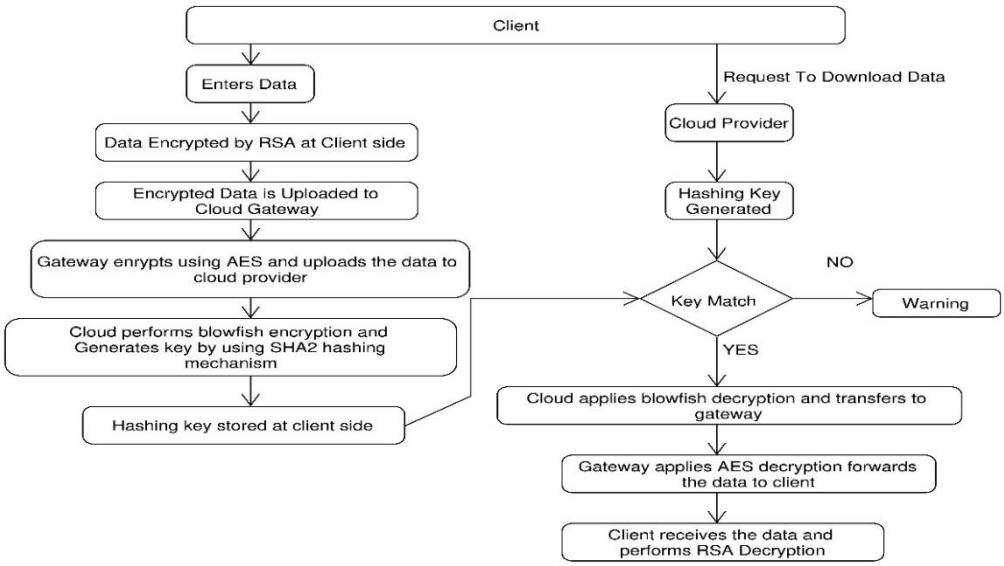

Figure 5 Flow of Work

INTEGRATION CHECK

i. Hash files will be generated in cloud server using SHA-2 algorithms.

ii. Integrity of the data is checked using these hash values.

iii. If all the hash codes are matched then file is downloaded at the gateway or the broker, else file is accessed by someone.

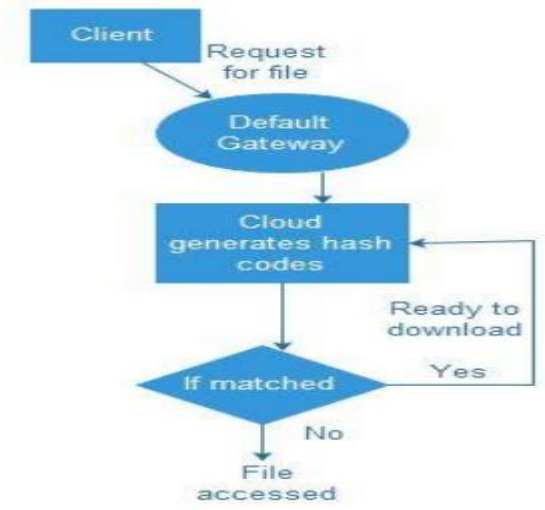

Figure 6 Integration Check

UPLOADING OF FILE AT THE CLOUD SERVER

I. Client will enter the data that has to be sent to the Cloud Provider.

II. The RSA algorithm will be performed at the client side which will encrypt the data before sending the data to the gateway.

III. This encrypted data is then transferred to the gateway.

IV. Gateway will receive the file sent by the client and will transfer it to the cloud provider for storage after applying AES encryption.

V. Cloud provider will receive the data from the gateway and performs blowfish encryption

VI. Cloud provider will apply the SHA2 hashing algorithm on the received file and will send the generated hash key value back to the client.

VII. This model will prevent will the types of attacks like man in the middle attack/ data mining attack.

So, using this approach, two purposes have achieved.

i. If anyone tries to hack the data while transferring from client to the gateway, he/she will get only encoded data.

ii. If anyone tries to perform the mining on the files stored at the cloud provider, no results will be retrieved

During downloading the file from cloud end, the client will follow the following steps:

i. Client will ask the gateway to download his/her stored file.

ii. Gateway will forward the request to the cloud provider and cloud provider will generate the hash value of the stored file using SHA2. This generated value is compared with the client's original key value. If the values have been matched, then the encrypted file is decrypted using blowfish decryption and is sent back to the gateway, else the warnings will be displayed to the user that file has been accessed by someone.

iii. Gateway will receive all the decryption file and will further apply the AES decryption before sending the file to the client.

iv. Client will further perform the RSA decryption to fetch the original data.

ALGORITHMS USED

Security is the key for the Cloud success, security in the cloud is now the main challenge of cloud computing. There are techniques which are used to enhance the cloud computing security i.e. AES and SHA2.

AES

The Advanced Encryption Standard (AES), also known as Rijndael (Keiko et al., 2012) (its original name), is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001. AES became effective as a federal government standard on May 26, 2002 after approval by the Secretary of Commerce. AES is included in the ISO/IEC 18033-3 standard. AES is available in many different encryption packages, and is the first publicly accessible and open vague cipher approved by the National Security Agency (NSA) for top secret information when used in an NSA approved cryptographic module.AES operates on a 4×4 column-major order matrix of bytes, termed the state, although some versions of Rijndael have a larger block size and have additional columns in the state. Most AES calculations are done in a special finite field. The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text. The number of cycles of repetition is as follows:

10 cycles of repetition for 128-bit keys.

12 cycles of repetition for 192-bit keys.

14 cycles of repetition for 256-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform cipher text back into the original plaintext using the same encryption key.

SHA2

SHA-2 (Secure Hash Algorithm 2) is a set of cryptographic hash functions designed by the United States National Security Agency (NSA). Cryptographic hash functions are mathematical operations run on digital data; by comparing the computed "hash" (the output from execution of the algorithm) to a known and expected hash value, a person can determine the data's integrity. For example, computing the hash of a

downloaded file and comparing the result to a previously published hash result can show whether the download has been modified or tampered with. A key aspect of cryptographic hash functions is their collision resistance: nobody should be able to find two different input values that result in the same hash output.SHA-2 includes significant changes from its predecessor, SHA-1.

RSA

RSA is a cryptosystem for public-key encryption, and is widely used for securing sensitive data, particularly when being sent over an insecure network such as the Internet. RSA was first described in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman of the Massachusetts Institute of Technology. Public-key cryptography, also known as asymmetric cryptography, uses two different but mathematically linked keys, one public and one private. The public key can be shared with everyone, whereas the private key must be kept secret. In RSA cryptography, both the public and the private keys can encrypt a message; the opposite key from the one used to encrypt a message is used to decrypt it. This attribute is one reason why RSA has become the most widely used asymmetric algorithm: It provides a method of assuring the confidentiality, integrity, authenticity and non-reputability of electronic communications and data storage. Many protocols like SSH, OpenPGP, S/MIME, and SSL/TLS rely on RSA for encryption and digital signature functions. It is also used in software programs -- browsers are an obvious example, which need to establish a secure connection over an insecure network like the Internet or validate a digital signature. RSA signature verification is one of the most commonly performed operations in IT. RSA derives its security from the difficulty of factoring large integers that are the product of two large prime numbers. Multiplying these two numbers is easy, but determining the original prime numbers from the total -- factoring -- is considered infeasible due to the time it would take even using today's super computers. The public and the private key-generation algorithm is the most complex part of RSA cryptography. Two large prime numbers, p and q, are generated using the Rabin-Miller primality test algorithm. A modulus n is calculated by multiplying p and q. This number is used by both the public and private keys and provides the link between them. Its length, usually expressed in bits, is called the key length. The public key consists of the modulus n, and a public exponent, e, which is normally set at 65537, as it's a prime number that is not too large. The e-figure doesn't have to be a secretly selected prime number as the public key is shared with everyone. The private key consists of the modulus n and the private exponent d, which is calculated using the Extended Euclidean algorithm to find the multiplicative inverse with respect to the totient of n.

GATEWAY (CLOUD DATA DISTRIBUTOR)

Cloud Data Distributor is the entity that receives data (files) from clients, performs various operations on the files. It also participates in data retrieving procedure by receiving chunk requests from clients and forwarding them to Cloud Providers. Clients do not interact with Cloud Providers directly rather via Cloud Data Distributor. This entity deals with Cloud Providers as an agent of clients. To upload data, clients deliver files to the Cloud Data Distributor. Each file is given a privacy level chosen by the client indicating its mining sensitivity. Here mining sensitivity of a file refers to the significance of information that can be leaked through mining the data in the file. The proposed system suggests 4 sensitivity levels of privacy: PL 0, 1, 2, 3. These 4 levels indicate public data (data accessible to everyone including the adversary), low sensitive data (data that do not reveal any private or protected information but can be used to find patterns), moderately sensitive data (protected data that can be used to extract non-trivial financial, legal, health information of a company or an individual), highly sensitive data or private data (data that can be used to extract personal information of an individual or private information of a company, revealing which can prove disastrous) respectively. The higher the privacy levels of a file, the more sensitive the data inside the file. After receiving files from clients, the Cloud Data Distributor partitions each file into chunks with each chunk having the same privacy level of the parent file. The total number of chunks for each file is notified to the client so that any chunk can be asked by the client by mentioning the filename and serial no. Serial no. corresponds to the position of the chunk within the file. Inside the Cloud Data Distributor each chunk is given a unique virtual id and this id is used to identify the chunk within the Cloud Data Distributor and Cloud Providers.

BLOWFISH

Blowfish is an encryption algorithm that can be used as a replacement for the DES or IDEA algorithms. It is a symmetric (that is, a secret or private key) block cipher that uses a variable-length key, from 32 bits to 448 bits, making it useful for both domestic and exportable use. (The U. S. government forbids the exportation of encryption software using keys larger than 40 bits except in special cases.) Blowfish was designed in 1993 by Bruce Schneier as an alternative to existing encryption algorithms. Designed with 32-bit instruction processors in mind, it is significantly faster than DES. Since its origin, it has been analyzed considerably. Blowfish is unpatented, license-free, and available free for all uses domain. Blowfish is included in a large number of cipher suites and encryption products, including SplashID. Blowfish's security has been extensively tested and proven. As a public domain cipher, Blowfish has been subject to a significant amount of cryptanalysis, and full Blowfish encryption has never been broken. Blowfish is also one of the fastest block ciphers in public use, making it ideal for a product like SplashID that functions on a wide variety of processors found in mobile phones as well as in notebook and desktop computers. Schneier designed Blowfish as a general-purpose algorithm, intended as a replacement for the aging DES and free of the problems associated with other algorithms. Notable features of the design include key-dependent S-boxes and a highly complex key schedule. Blowfish has a 64-bit block size and a key length of anywhere from 32 bits to 448 bits. It is a 16-round Feistel cipher and uses large key-dependent S-boxes. It is similar in structure to CAST-128, which uses fixed S-boxes. The algorithm keeps two subkey arrays: the 18-entry P-array and four 256-entry S-boxes. The S-boxes accept 8-bit input and produce 32-bit output. One entry of the P-array is used every round, and after the final round, each half of the data block is XORed with one of the two remaining unused P-entries. The diagram to the right shows Blowfish's F-function. The function splits the 32-bit input into four eight-bit quarters, and uses the quarters as input to the S-boxes. The outputs are added modulo 232 and XORed to produce the final 32-bit output. Since Blowfish is a Feistel network, it can be inverted simply by XORing P17 and P18 to the ciphertext block, then using the P-entries in reverse order.

SIMULATION VIEW OF PROPOSED WORK

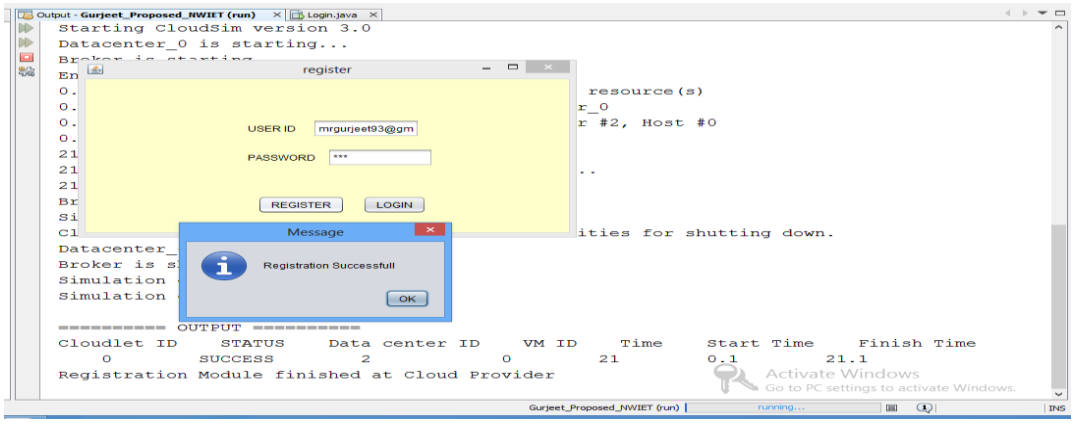

Figure 7. Registration at the cloud provider.

The above figure 7 demonstrates the registration of the user at the cloud provider. The user will enter his/her id and password and will get himself/herself registered at the cloud end. Once the user has been registration, the system will automatically navigation to the login section. On submitting the Login button, the request will go to the cloud provider that will check whether the user's entered data is valid or invalid. After the Login Section is completed, the system will generate the one time password by using the MD5 algorithm and will encrypt the OTP using MD5 encryption algorithm. The encrypted OTP is sent to the user's registered email ID at the cloud provider.

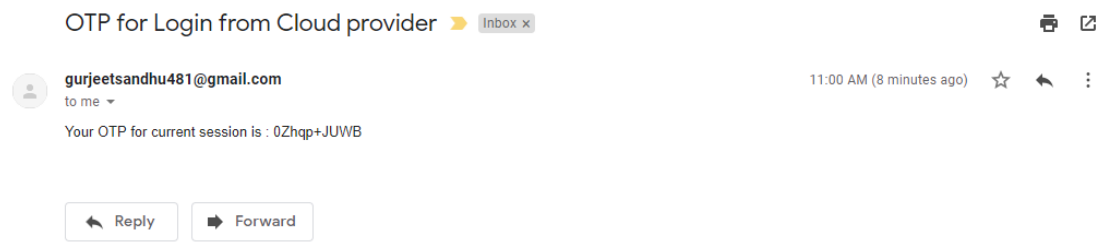

Figure 8. OTP received via e-mail

After OTP verification, the user will enter into the main page where he/she can upload the file to the cloud provider or can download the previously uploaded files from the cloud server. This section will only open when the user's credentials are properly verified by the cloud provider.

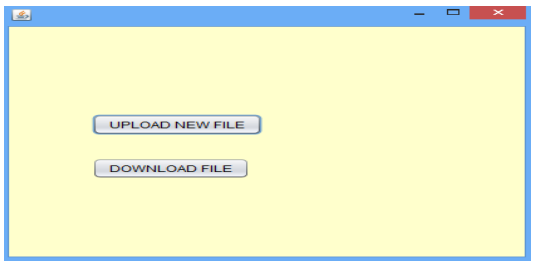

Figure 9. Main Sections after Login Process

User will choose the file from his/her laptop when he/she clicks on the file upload button. This file will be encrypted at the client side using AES. After encryption process is completed, the encrypted data is sent to the gateway.

EXPERIMENTAL RESULTS

After analyzing the loophole of base security of the data between the client and the cloud provider is enhanced by using the AES and verification mechanism at the cloud provider. This experiments work on a machine with the following configuration: Intel Core 2 CPU, 980 MHz, 1.99 GB RAM, Microsoft windows 7. We have the Java version 8 with the Net beans IDE version 8.

Table 1. Readings of the Base work

S.No.	File Type	File Size	Processing Time (milliseconds)	Processing Cost (Rupees)	RSA Encryption Time (Nanosecond)
1	PHP	26bytes	26	79.326	257189433
2	VB	74bytes	74	225.774	246085696
3	HTML	116bytes	116	353.916	1079356
4	MP3	162bytes	162	494.262	1329666
5	CDR	221bytes	221	674.271	1450544
6	JS	328bytes	328	1000.728	912292
7	PSL	383bytes	383	1168.533	554217
8	JAVA	495bytes	495	1510.245	555358

9	FLV	533bytes	533	1626.183	1139225
10	PDF	635bytes	635	1937.385	626630
11	HTML	722bytes	722	2202.822	580446
12	TXT	777bytes	777	2370.627	1094180
13	JSP	828bytes	828	2526.228	428777
14	ASPX	1.00KB	1033	3151.683	542244
15	VLC	1.18KB	1218	3716.118	1102733
16	JAR	1.25KB	1282	3911.382	527989
17	MSI	1.50KB	1543	4707.693	1116988
18	XML	2.34KB	2401	7325.451	814791
19	MP4	3.74KB	3837	11706.687	467549
20	MKV	5.93KB	6080	18550.08	372330
21	ASD	7.49KB	7674	23413.374	400268
22	HIB	10.01KB	10363	31617.513	386584
23	CLASS	15.04KB	15776	48132.576	375751
24	ISD	20.00KB	19000	57969	512024
25	DOCS	23.3KB	23960	73101.96	527419

Table 2. Readings of the proposed work

S.No	File Name	File Size	Processing Time	Processing Cost	OTP Generation and Encryption Time	AES ENCRYPTION TIME	Sha2 Generation Time
1	PHP	26bytes	21.33	65.088	2375	250	2413564
2	VB	74bytes	51.33	156.618	1484	203	2289265
3	HTML	116bytes	87.33	266.454	1719	234	1758999
4	MP3	162bytes	117.33	357.984	1625	62	1362726
5	CDR	221bytes	146	445.446	1704	891	1314831
6	JS	328bytes	212	646.812	1485	141	2686679
7	PSL	383bytes	240.67	734.274	1797	922	1322243
8	JAVA	495bytes	314.67	960.048	2469	187	1530358
9	FLV	533bytes	343.33	1047.51	1656	141	2857733
10	PDF	635bytes	402	1226.502	1375	187	3423920
11	HTML	722bytes	467.43	1425.834	1625	125	2090843

12	TXT	777bytes	496.67	1515.33	1750	125	1860491
13	JSP	828bytes	525.33	1602.792	1547	109	1913517
14	ASPX	1.00KB	656.67	2003.49	1657	94	1389524
15	VLC	1.18KB	774	2361.474	1172	109	1641542
16	JAR	1.25KB	810	2471.31	1531	110	1671762
17	MSI	1.50KB	971.33	2963.538	1438	125	1588516
18	XML	2.34KB	1510.67	4609.044	1469	140	1739614
19	MP4	3.74KB	2401.33	7326.468	1453	172	2004176
20	MKV	5.93KB	3809.33	11622.276	1766	359	1878167
21	ASD	7.49KB	4794.67	14628.528	1641	515	2615977
22	HIB	10.01KB	6473.33	19750.14	1453	671	2967777
23	CLASS	15.04KB	9860.67	30084.894	1781	953	2891373
24	ISD	20.00KB	12838	39168.738	1718	1297	5568360
25	DOCS	23.3KB	14968.67	45669.402	1625	1516	4209625

PERFORMANCE METRICS

After implementing the proposed methodology, we have reached up to a solution that the cloud security can be enhanced by applying the model of RSA, Blowfish, and AES secure authentication with OTP and data verification using SHA2. The data sent/received by the client is of utmost importance and it needs to be handled carefully. We have been able to reduce the processing time, encryption time, processing cost which increases the overall efficiency of the system.

ACCURACY OF THE SYSTEM

Accuracy of the System can is enhance by measuring Processing time and cost as shown in the graphs below, which increases the overall efficiency of the system.

- **Processing Time**

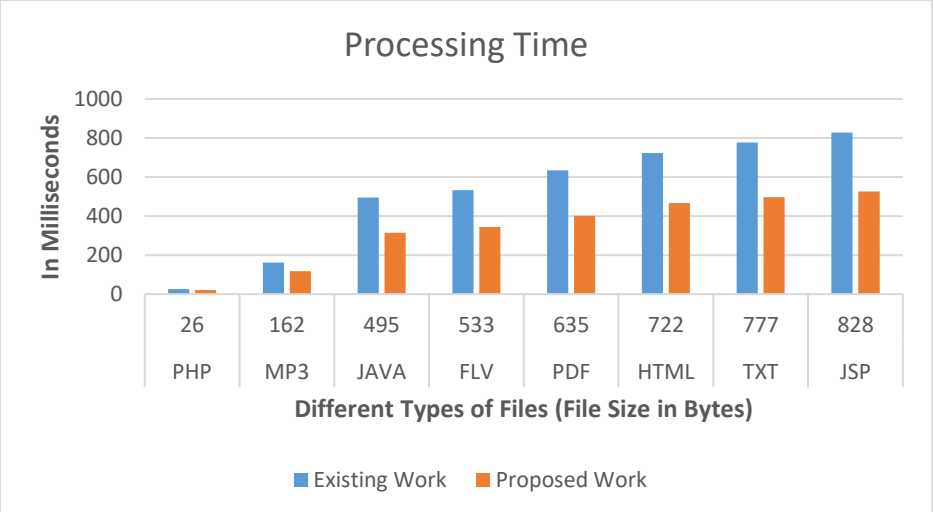

Figure 10. Processing time.

From the above bar chart, it is clear that the processing time has been reduced. The processing time depends upon the size of the file. As the size of the file increases, the processing time will also increase. But we have been able to reduce the processing time of the proposed work as it will finally increase the overall efficiency of the system.

- **COST**

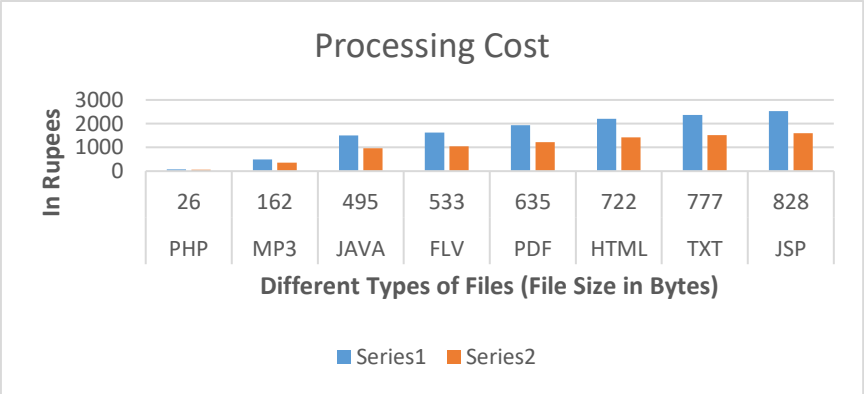

Figure 11. File Size v/s Cost.

From the above bar chart, it is clear that the cost has been reduced. Usually Cloud Computing providers have detailed costing models which are used to bill users on *pay per use basis* .The cost depends upon the size of the file. As the size of the file increases, the Cost will also increase. But we have been able to reduce the Cost of the proposed work as it will finally increase the overall efficiency increase.

OTP GENERATION AND ENCRYPTION USING AES

For the secure authentication, we have generated the one time password via MD5 algorithm and is encrypted using AES and is sent to the client's registered email id. From the below graph, it is clear that by implementing the OTP mechanism, there is no effect on the system. From the number of experiments it is clear that OTP generation and encryption is taking lesser than 1 second

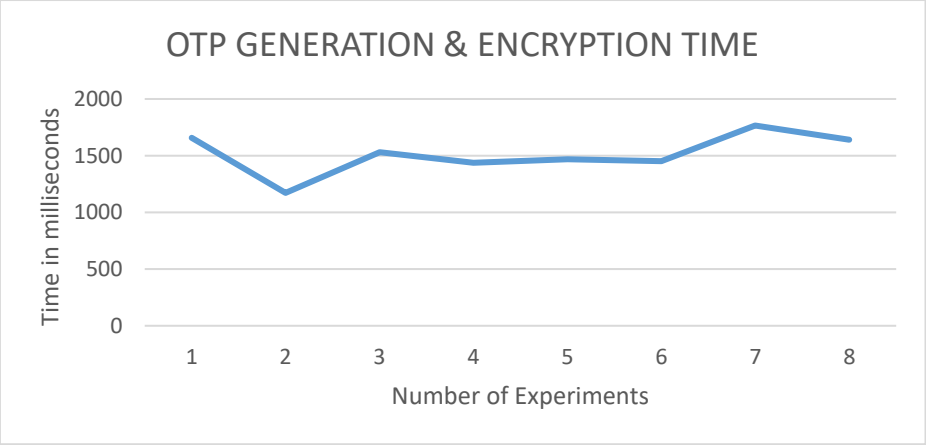

Figure 12. OTP generation and encryption time.

From the below line chart, it is clear that as the size of the file keeps on increasing, the encryption time will keep on increasing. We have taken different types of files of different sizes for testing purposes.

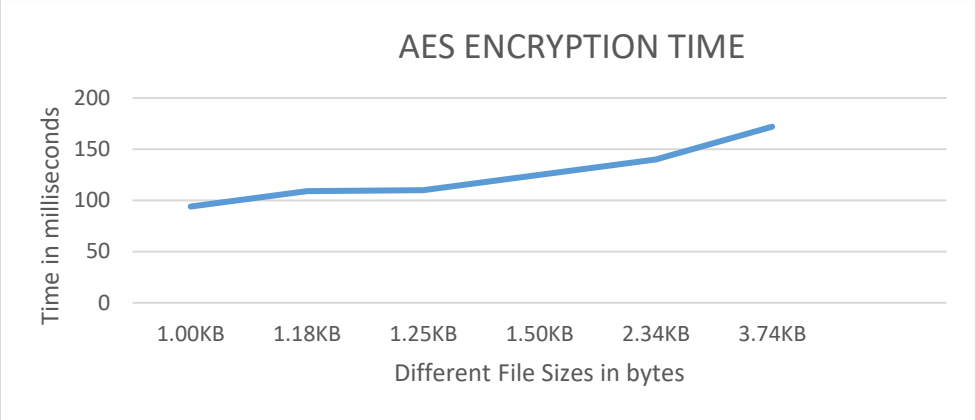

Figure 13. AES Encryption Time.

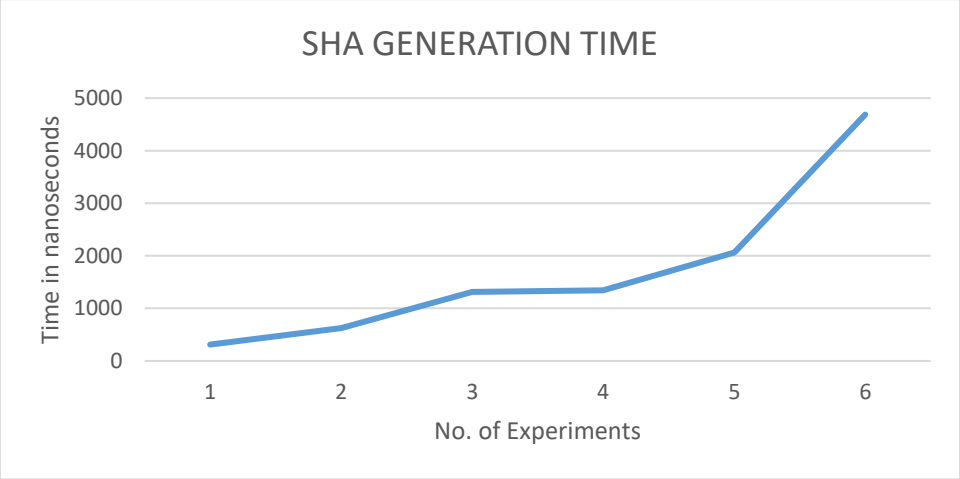

Figure 14. SHA2 Generation Time.

For the secure verification before downloading, we have included the SH2 verification mechanism that will match the newly generated key with the previously stored key. The above figure shows the key generation time using SHA2. From the graph it is clear that the SHA 2 generation is not taking much time and is not having any extra overhead on the entire performance of the system.

CONCLUSION

The primary conclusion of our research is that adoption of user-centric security models and shifting certain parts of communication and computation to the client side allows us to provide the cloud consumers with more visibility and control over their resources. Therefore, using this approach not only the security and privacy concerns of cloud consumers can be addressed more effectively, but also the burden of managing end-users' identities and access control will be reduced from cloud service providers. This study collectively describes cloud computing security challenges in general and describes the mitigation practices that have been proposed to handle the identified challenges. We have successfully implemented the above proposed system and have reached to a solution that by using this proposed mechanism, we can achieve the better security in cloud computing more efficiently.

REFERENCES

1. Mohis M and Devipriya V S, "An improved approach for Enhancing Public Cloud Data Security through Steganographic Technique," IEEE, pp. 1-5, 2016.

2. Kunal V. Raipurkar and Anil V. Deorankar , "Improve Data Security in Cloud Environment by using LDAP and Two Way Encryption Algorithm," IEEE, pp. 1-4, 2016.

3. V.Swath, K.Sudha, R.Aruna, C.Sangeetha and R.Janani, "Providing Advanced Security Mechanism for Scalable Data Sharing In Cloud Storage," IEEE, pp. 1-6, 2016.

4. Shivangi Sengar and Rajesh Kumar Chakrawarti, "Implementation of PDS System with Improved Security and Transparency under Cloud Environment," IEEE, pp. 1-6, 2016.

5. Mohis M and Devipriya V S, "An improved approach for Enhancing Public Cloud Data Security through Steganographic Technique," IEEE, 2016.

6. S. Pandey , A. Dwivedi , J. Pant and M. Lohani , "Security Enforcement using TRBAC in Cloud Computing," IEEE, pp. 1232-1238, 2016.

7. R.K.Shyamasundar, N.V.Narendra Kumar and Muttukrishnan Rajarajan, "Information-Flow Control for Building Security and Privacy Preserving Hybrid Clouds," IEEE, pp. 1410-1417, 2016.

8. P. More and D G Harkut, "Cloud Data Security using Attribute-based Key Aggregate Cryptosystem," *IEEE,* pp. 855-861, 2016.

9. D. Singh and Harsh K Verma, "A New Framework for Cloud Storage Confidentiality to Ensure Information Security," *IEEE,* 2016.

10. Kunal V. Raipurkar and Anil V. Deorankar , "Improve Data Security in Cloud Environment by using LDAP and Two Way Encryption Algorithm," *IEEE,* 2016.

11. S. Sengar and . R. K. Chakrawarti , "Implementation of PDS System with Improved Security and Transparency under Cloud Environment," *IEEE,* 2016.

12. A. Albugmi, M. O. Alassafi , . R. Walters and Gary Wills, "Data Security in Cloud Computing," *IEEE,* pp. 55-59, 2016.

13. A. Singh and M. Malhotra , "Hybrid Two-Tier Framework for Improved Security in Cloud Environment," *IEEE,* pp. 955-960, 2016.

14. Bin Feng, Xinzhu Ma, Cheng Guo, Hui Shi, Zhangjie Fu and Tie Qiu, "An Efficient Protocol with Bidirectional Verification for Storage Security in Cloud Computing," *IEEE,* pp. 1-13, 2016.

15. Mrinal Kanti Sarkar and S. Kumar, "A Framework to Ensure Data Storage Security in Cloud Computing," *IEEE,* 2016.

16. N.Thillaiarasu and ChenthurPandian.S, "Enforcing Security and Privacy over Multi – Cloud Framework Using Assessment Techniques," *IEEE,* 2016.

17. R. R. Gupta, G. Mishra, S. Katara, A. Agarwal, M. K. Sarkar, R. Das and S. Kumar, "Data Storage Security in Cloud Computing Using Container Clustering," *IEEE,* 2016.

18. S.Petcy Carolin and M.Somasundaram, "Data Loss Protection And Data Security Using Agents For Cloud Environment," *IEEE,* pp. 1-5, 2016.

19. T. Mavroeidakos, A. Michalas and Dimitrios D. Vergados , "Security Architecture based on Defense in Depth for Cloud Computing Environment," *IEEE,* 2016.

20. Deepak H. Sharma, C A. Dhote and Manish M. Potey, "Intelligent Transparent Encryption-Decryption as Security-as-a-Service from Clouds," *IEEE,* pp. 359-362, 2016.

21. Deepak H. Sharma, C A. Dhote and Manish M. Potey, "Implementing Intrusion Management as Security-as-a-Service from Cloud," *IEEE,* pp. 363-366, 2016.

22. K. V. Raipurkar and A. V. Deorankar, "Improve Data Security in Cloud Environment by using LDAP and Two Way Encryption Algorithm," Symposium on Colossal Data Analysis and Networking (CDAN), IEEE, 2016.