INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 3, Issue 6)
Available online at www.ijariit.com

A Novel Technique of Data Security in Cloud Computing based on Blowfish with MD5 method

Harpreet Kaur
Student
Yadavindra College of Engineering, Talwandi Sabo, Punjab
happygill.kaur22@gmail.com

Abstract: In cloud computing, users transfer their burden of installing the software, data maintenance, infrastructure, storage space etc. on the cloud service provider. These providers offer to their clients the possibility to store, retrieve and share data with other users in a transparent way [1]. In this paper, a method is implemented for ensuring the data security of the files being uploaded to the cloud by different clients. This security is achieved through a technique of encryption using blowfish with the MD5 method. The results of the proposed method have shown that the size of the encrypted file is decreased as compared to that of existing techniques. So, the storage space of the cloud is used in a much efficient manner when the proposed technique is implemented. Moreover, the replay attack is also defended more efficiently through the proposed method thus increasing the security level of the cloud. With the use of MD5 method along with blowfish, the data is encrypted in lesser time hence making it difficult for the attacker to hack the data, of the client, which is uploaded to the cloud.

Keywords: Cloud, Blowfish, MD5, Security.

I. INTRODUCTION

Cloud computing, also known as on-demand computing, is a kind of internet-based computing that provides shared processing resources and data to computers and other devices on demand[1]. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in third party data centers [2].

Cloud computing refers to both the applications delivered as services over the internet and the hardware and systems software in the data centers that provide those services. The services themselves have long been referred to as Software as a Service (SaaS)[3].Some vendors use terms such as IaaS (Infrastructure as a Service) and PaaS (Platform as a Service) to describe their products, but we eschew these because accepted definitions for them still vary widely. The line between "low-level" infrastructure and a higher-level "platform" is not crisp. We believe the two are more alike than different, and we consider them together [3].

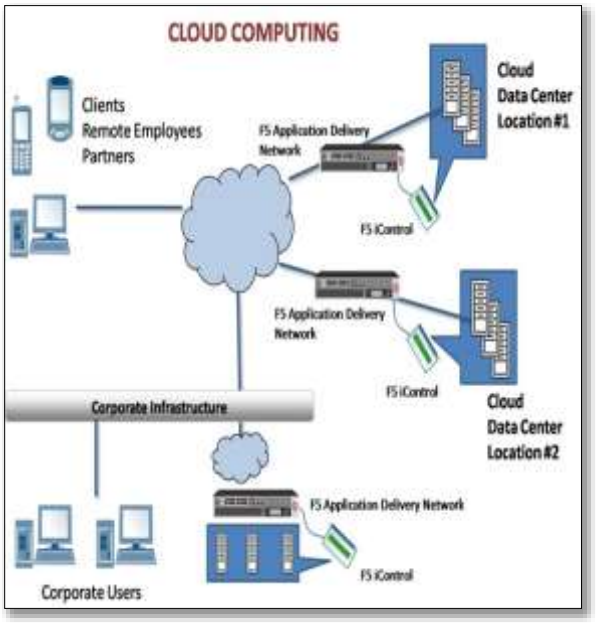


Figure 1 Basic Framework for Cloud Computing [4]

A. Characteristics of Cloud Computing

Cloud computing exhibits the following key characteristics: [5]

- High Reliability
- Versatility
- Extremely inexpensive
- On demand service
- Ultra large scale
- Increased productivity
- Security

B. Applications of Cloud Computing

By using the services of cloud service provider user's transfer their burden of installing the software, data maintenance, infrastructure, storage space etc. on the cloud service provider. These providers offer to their clients the possibility to store, retrieve and share data with other users in a transparent way.

Clouds shift the responsibility to install and maintain hardware and basic computational services away from the customer (e.g., a laboratory or consortium) to the cloud vendor [6]. Higher levels of the application stack and administration of sharing remain intact, and remain the customer's responsibility.

The following features, especially the first three, are commonly associated with clouds:

Resource Outsourcing: Instead of a consumer providing their own hardware, the cloud vendor assumes responsibility for hardware acquisition and maintenance [7].

Utility Computing: The consumer requests additional resources as needed, and similarly releases these resources when they are not needed. Different clouds offer different sorts of resources, e.g., processing, storage, management software, or application services.

Large Numbers of Machines: Clouds are typically constructed using large numbers of inexpensive machines. As a result, the cloud vendor can more easily add capacity and can more rapidly replace machines that fail, compared with having machines in multiple laboratories. Generally speaking, these machines are as homogeneous as possible both in terms of configuration and location [8].

Automated Resource Management: This feature encompasses a variety of configuration tasks typically handled by a system administrator. For example, many clouds offer the option of automated backup and archival. The cloud may move data or computation to improve responsiveness. Some clouds monitor their offerings for malicious activity.

Virtualization: Hardware resources in clouds are usually virtual as they are shared by multiple users to improve efficiency. That is several lightly-utilized logical resources can be supported by the same physical resource.

C. Data Security and Integrity in Cloud Computing

Cloud computing poses privacy concerns because the service provider can access the data that is in the cloud at any time. It could accidentally or deliberately alter or even delete the information. In a cloud provider platform being shared by different users, there may be a possibility that information belonging to different customers resides on same data server [2]. Therefore information leakage may arise by mistake when information for one customer is given to another.

There is a number of security issues associated with cloud computing but these issues fall into two broad categories [9]:

- Security issues faced by cloud providers(organizations providing software, platform or infrastructure as a service via the cloud)
- Security issues faced by their customers (companies or organizations who host applications or store data on the cloud) [10].

The service provider must ensure that their infrastructure is secure and that their clients ' data and applications are protected while the user must take measures to fortify their application and use strong passwords and authentication measures.

Some of the key features in providing data security and integrity in cloud computing are as follows: [11]

- Identity management
- Physical security
- Personnel security
- Availability
- Application security
- Privacy

D. Cloud Computing in Cryptography

Cloud computing is an evolving paradigm, shifting the computing and storage capabilities to external service providers. Especially due to this loss of direct control on outsourced data, users are reluctant for adopting cloud services [12]. To construct a secure cloud computing system, security at infrastructure, service platforms and application software levels have to be studied for secure cloud computing system. Information encryption is one of the effective means to achieve cloud computing information security. Users can encrypt data that is stored or processed in the cloud to prevent unauthorized access [13]. Traditionally, information encryption focuses on specified stages and operations, such as data encryption for cloud computing, a system level design has to be implemented.

II. LITEARATURE REVIEW

Wang et al. (2010) proposed a scheme to help enterprises to efficiently share confidential data on cloud servers. This goal was achieved by first combining the hierarchical identity-based encryption (HIBE) system and the cipher text-policy attribute-based encryption (CP-ABE) system, and then making a performance-expressivity tradeoff, finally applying proxy re-encryption and lazy re-encryption to the scheme.

Wang et al. (2011) had studied the problem of ensuring the integrity of data storage in cloud computing the proposed method allowed a third part auditor(TPA) on behalf of the cloud client, in order to verify the integrity of dynamic data stored on the cloud. In particular, to achieve efficient data dynamics, the proposed method improved the existing proof of storage methods by manipulating the classic Merkle Hash Tree construction for block tag authentication. In proposed design PKC based homomorphic authenticator (BLS signature) was used to equip the verification protocol with public auditability.

Nesrine et al. (2013) had proposed ID based cryptography in which the data is firstly encrypted and then stored on the public cloud server. This concept had also offered access control so that only authorized users can use the data. With the help of this approach unauthorized user even not get the data without client's permission. It provided secrecy for encrypted data which was stored in public servers and it had offered controlled data access and sharing among users, so that unauthorized users or untrusted servers cannot access or search over data without client's authorization.

Tirthani et al. (2014) had explained about cloud security issues and then proposed a security model for cloud in which Diffie Hellman Key Exchange and Elliptical Curve Cryptography algorithms were used. In this research paper, they had contemplated a design for cloud architecture which ensured secured movement of data at client and server end. They had used the non-breakability of Elliptic curve cryptography for data encryption and Diffie Hellman Key Exchange mechanism for connection establishment. The proposed encryption mechanism used the combination of linear and elliptical cryptography methods

Khan et al. (2014) analyzed the different cryptographic algorithms used for ensuring the security of the data in the cloud. Two types of encryption algorithms were described, one is a symmetric algorithm and asymmetric algorithms. Cloud storage contains a large set of databases and for such a large database asymmetric-key algorithm's performance was slower when compared to symmetric-key algorithms.

Bollavarapu et al. (2014) had explained the algorithms used for data storage security in the cloud and desktops and to overcome these problems encryption and decryption techniques like RSA and RC4 had been discussed here in more details. The server and the email management software was installed in the cloud and managed by service providers.

Ora et al. (2015) described a solution to maintain data security and data integrity. This scheme contained a combination of RSA Partial homomorphic and MD5 hashing algorithm.In this solution, data was encrypted by RSA Partial before uploading it to a cloud server. After uploading its hash value was calculated by MD5 hashing scheme. All these approaches undergo the following step Encryption/Decryption, Data uploading on a cloud, Hashing, and Verification. This scheme contained a combination of RSA Partial homomorphic and MD5 hashing algorithm.

Abbas et al. (2016) aimed to provide an effective, flexible and secure method to improve data security in cloud computing. The experimental results showed that the key generation complexity was decreased and there was no need to issue a certificate because the use of Modified Identity-Based Cryptography (MIBC) and Elliptic Curve Integrated Encryption Scheme (ECIES) had provided data confidentiality and data integrity. The main idea of this proposal was to combine the security of MIBC and ECIES with Trusted Cloud (TC).The use of MIBC had significantly decreased the key generation complexity and abolished the need to issue the certificate.

III. METHODOLOGY USED

A. Flow of Proposed Method

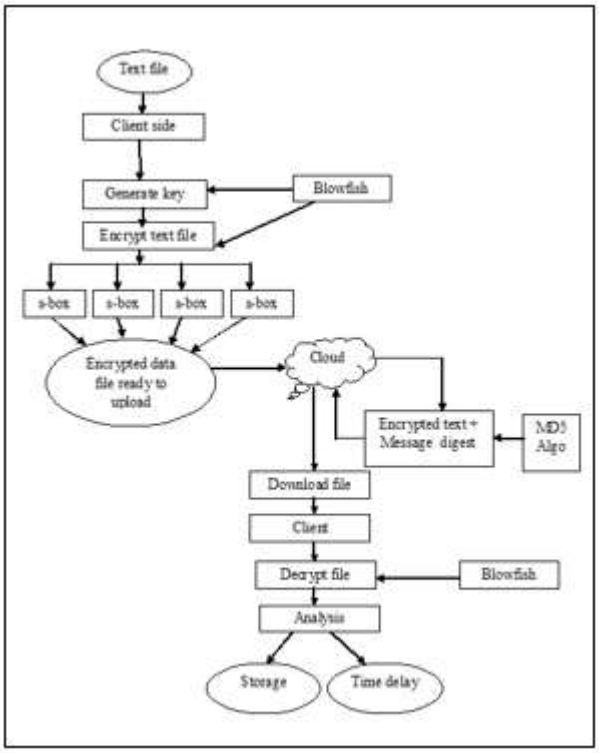


Figure 2: Flowchart of Proposed Method

In this proposed work blowfish and MD5 method is used for encryption of data and uploading it to cloud securely. The various steps of the proposed method are described as below:

1) Initially, at the client side, the text file is encrypted using blowfish algorithm and the key is generated.
2) Then, the encrypted file is uploaded to the cloud server.
3) After successfully uploading the file, data owner gets the general details like encryption time, input file size, and encrypted file size.
4) The hash value of the uploaded file is calculated using MD5 hashing algorithm for maintaining the data integrity of the file.
5) The encrypted text along with the attached message digest is stored in the cloud.
6) Now the client can download the file from the cloud by decrypting it using the same key.
7) The decryption time can be calculated.
8) The total time can be obtained by adding both encryption and decryption time.

File name	Input file size (bytes)	Encrypted file size (bytes)	Encryption time(ms)	Decryption time(ms)	Total time (encryption+decryption)
new1.txt	519815	948894	200	140	340
new2.txt	581632	1063774	160	100	260
new3.txt	378754	687316	110	102	212
new4.txt	1315331	2405820	388	304	692

9) Finally, the execution time and the storage capacity of encrypted files can be used as time delay and storage parameters respectively for comparing the proposed technique with the existing Diffie Hellman and AES.
10) When the replay attack is simulated from outside, it is offended well by the blowfish and MD5 technique because this technique takes lesser time is encryption and decryption of files as compared to the existing technique.

IV. RESULTS

A. Storage and Time Delay Parameters Using Diffie Hellman AES Technique

Table 1: Storage and time delay parameters using existing diffie hellman AES technique:

Table 1 shows the input file size, encrypted file size, encryption time and decryption time of different files when encrypted for securing data on the cloud using Diffie Hellman AES technique.

B. Storage and Time Delay Parameters by Using Blowfish with MD5 Method

Table 2: Storage and Time Delay Parameters Using Blowfish with MD5 Method

File name	Input file size (bytes)	Encrypted file size (bytes)	Encryption time(ms)	Decryption time(ms)	Total time (encryption + decryption)
new1.txt	519815	943899	50	148	198
new2.txt	581632	1062705	40	119	159
new3.txt	378754	686359	30	89	119
new4.txt	1315331	2378298	139	310	449

Table 2 shows the input file size, encrypted file size, encryption time and decryption time of different files when encrypted for securing data on the cloud using Blowfish and MD5 method.

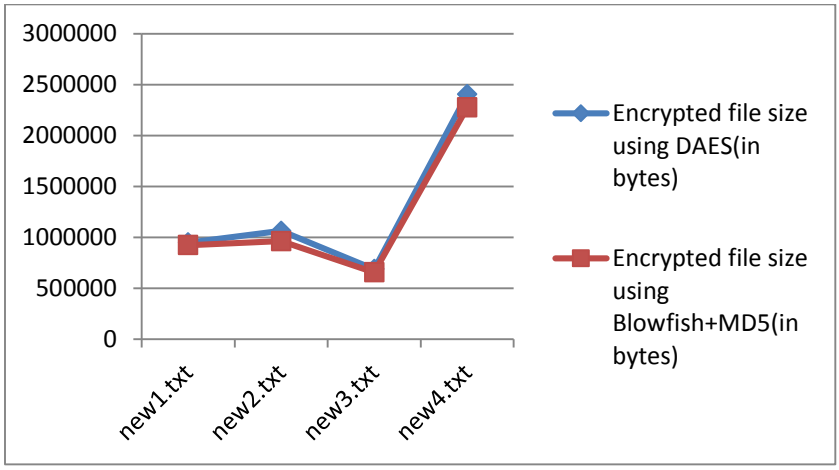
C. Comparison of DHAES Technique and Blowfish with MD5 Technique

Table 3: Comparison of DHAES and Blowfish with MD5 on the Basis of Storage Parameter

File name	Encrypted file size using DHAES(in bytes)	Encrypted file size using Blowfish+MD5(in bytes)
new1.txt	948894	923899
new2.txt	1063774	962705
new3.txt	687316	656359
new4.txt	2405820	2278298

Table 3 Shows The Encrypted File Size Of The Text Files Using Existing Dhaes Technique And Proposed Blowfish With Md5 Technique.

Graph 1: Comparison of DHAES and Blowfish with MD5 on the Basis of Storage Space



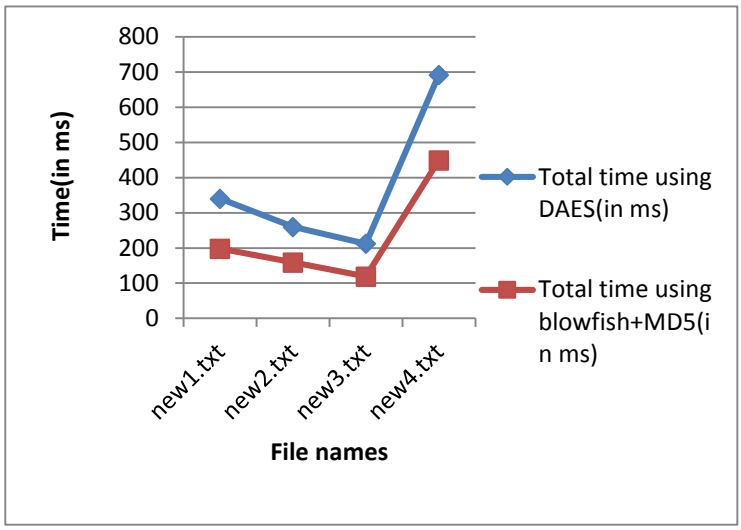
Graph 1 represents the comparison of the size of the encrypted files when DHAES and Blowfish with the MD5 method are implemented on the same files.

Table 4: Comparison of DHAES and Blowfish+MD5 on the Basis of Time Delay

File name	Total time using DHAES(in ms)	Total time using blowfish+MD5(in ms)
new1.txt	340	198
new2.txt	260	159
new3.txt	212	119
new4.txt	692	449

Table 4 shows the values of the time delay parameter when the existing DHAES technique and proposed blowfish with the MD5 method are implemented on text files.

Graph 2: Comparison of DHAES and Blowfish+MD5 on the Basis of Time Delay



Graph 2 represents the comparison of total time taken by both the techniques when implemented on 4 different files.

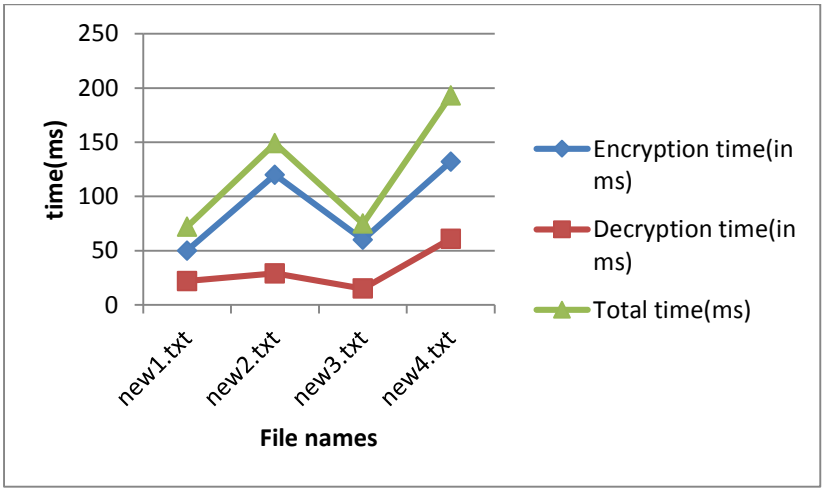
D. Time Delay Parameter Using Blowfish Algorithm When Replay Attack is Simulated

Table 5: Time Delay Parameter Using Blowfish Algorithm When Replay Attack is simulated

File name	Encryption time(in ms)	Decryption time(in ms)	Total time(ms)
new1.txt	50	22	72
new2.txt	120	29	149
new3.txt	60	15	75
new4.txt	132	61	193

Table 5 gives us the encryption time, decryption time and the total time of 4 different text files using blowfish with MD5 technique when replay attack is simulated.

Graph 3: Time Delay Parameter Using Blowfish Algorithm When Replay Attack is Simulated



Graph 3 represents the encryption time, decryption time and the total time of 4 different text files using blowfish with MD5 technique when replay attack is simulated.

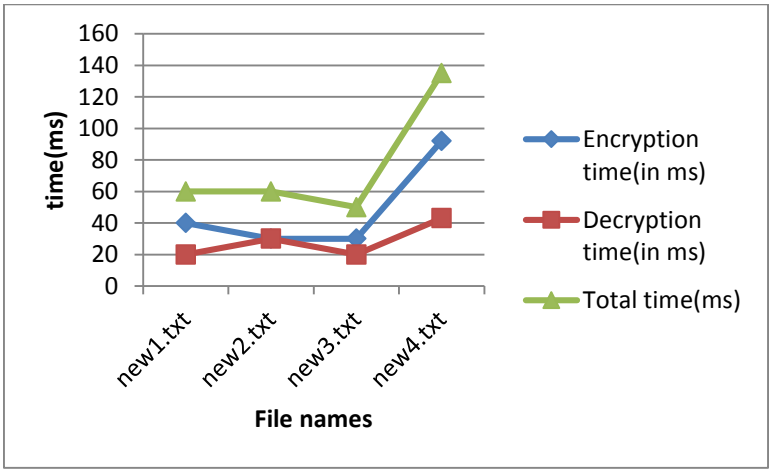
E. Time Delay Parameter Using Blowfish + Md5 Method When Replay Attack is Simulated

Table 6: Time Delay Parameter Using Blowfish with Md5 Method When Attack Replay is simulated

File name	Encryption time(in ms)	Decryption time(in ms)	Total time(ms)
new1.txt	40	20	60
new2.txt	30	30	60
new3.txt	30	20	50
new4.txt	92	43	135

Table 6 shows the encryption time, decryption time and total time taken by the newly implemented technique i.e. blowfish and MD5 method when a replay attack is simulated.

Graph 4: Time Delay Parameters Using Blowfish with Md5 Method When Attack Replay is simulated



Graph 4 represents the encryption time, decryption time and total time taken by the newly implemented technique i.e. blowfish and MD5 method when a replay attack is simulated.

F. Comparison of Blowfish and Blowfish with Md5 Method on The Basis of Time Delay Parameter

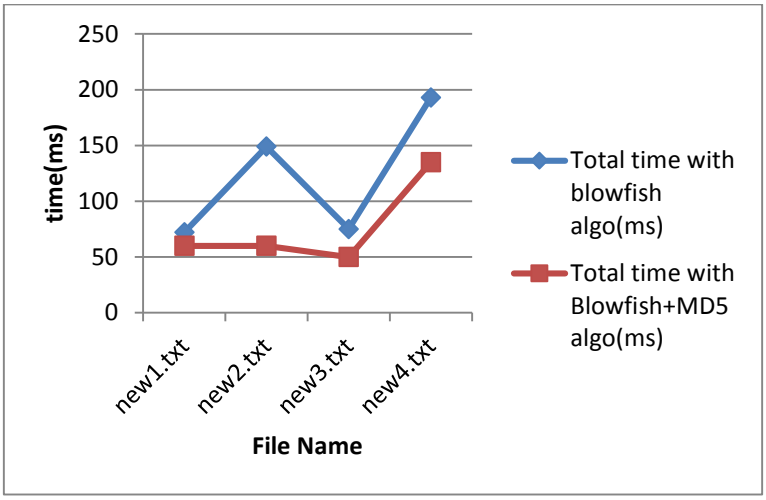
Table 7: Comparison of Blowfish and Blowfish with Md5 Method On The Basis Of Time Delay Parameter

File name	Total time with blowfish algo(in ms)	Total time with Blowfish+MD5 algo(in ms)
new1.txt	72	60
new2.txt	149	60
new3.txt	75	50
new4.txt	193	135

Table 7 depicts the total time obtained after implementing the existing blowfish and proposed blowfish with MD5 techniques on the same files when a replay attack is simulated.

As it is very clear on the table that the total time of encryption and decryption time is less when the proposed technique i.e. Blowfish +MD5 is implemented.

Graph 5: Comparison of Blowfish and Blowfish with Md5 Method on The Basis of Time Delay Parameter



Graph 5 depicts the comparison of total time obtained after implementing the existing blowfish and proposed blowfish with MD5 techniques on the same files when a replay attack is simulated.

On comparing the total time of encryption and decryption used in both the techniques it is clear that the new technique i.e. blowfish and MD5 takes lesser time than the existing technique. It means the replay attack is defended well through the new technique of Blowfish with MD5 method because it will be very difficult for an attacker to crack the code in such a small piece of time so the proposed technique provides more security to the data at the cloud.

V. CONCLUSION AND FUTURE SCOPE

In this research, a method is implemented for ensuring the data security of the files being uploaded to the cloud by different clients. This security is achieved through a technique of encryption using blowfish and MD5 method. The experimental results of the proposed method show that size of the encrypted file is decreased by approximately 7% as compared to that of existing technique, resulting in the lesser storage space consumption at the cloud. Hence, the storage space of the cloud is used in a much efficient manner when the proposed technique is implemented. Also, the time taken by the proposed method of MD5 with blowfish, in encryption and decryption of the text file, is approximately 42% lesser than that of the existing Diffie Hellman with AES technique.

Moreover, the replay attack is also defended more efficiently through the proposed method thus increasing the security level of the cloud. With the use of MD5 method along with blowfish, the total time for encryption and decryption of text files is reduced by approximately 33% as compared to the existing technique, hence making it difficult for the attacker to hack the data of the client, which is uploaded on the cloud.

So, the experimental results of our research work show that the proposed technique of blowfish with MD5 outperforms the existing technique in terms of storage space and time delay.

In future, we can do work on various other methods of encryption which may enhance the security of the system. The hybrid techniques of encryption can be developed in order to improve the encryption and decryption time. Research can also be done on the image, audio and video data. Security enhancements can be done on actual cloud data like Amazon.in.

REFERNCES

1. Ora, P., & Pal, P. R. (2015, September). Data security and integrity in cloud computing based on RSA partial homomorphic and MD5 cryptography. In *Computer, Communication, and Control (IC4), 2015 International Conference on* (pp. 1-6). IEEE.
2. Chen, D., & Zhao, H. (2012, March). Data security and privacy protection issues in cloud computing. In *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on* (Vol. 1, pp. 647-651). IEEE.
3. Khan, M. S. S., & Deshmukh, M. S. S. (2014). Security in cloud computing using cryptographic algorithms. IJCA.
4. Kamara, S., & Lauter, K. E. (2010, January). Cryptographic Cloud Storage. In *Financial Cryptography Workshops* (Vol. 6054, pp. 136-149).
5. Zargari, S., & Benford, D. (2012, September). Cloud forensics: concepts, issues, and challenges. In Emerging Intelligent Data and Web Technologies (EIDWT), 2012 Third International Conference on (pp. 236-243). IEEE.
6. Auxilia, M., & Raja, K. (2014, December). Dynamic Access Control Model for Cloud Computing. In Advanced Computing (ICoAC), 2014 Sixth International Conference on (pp. 47-56). IEEE.
7. Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R., & Molina, J. (2009, November). Controlling data in the cloud: outsourcing computation without outsourcing control. In Proceedings of the 2009 ACM workshop on Cloud computing security (pp. 85-90). ACM.

8. Luo, X., Yang, L., Ma, L., Chu, S., & Dai, H. (2011, November). Virtualization security risks and solutions for Cloud Computing via divide-conquer strategy. In Multimedia Information Networking and Security (MINES), 2011 Third International Conference on (pp. 637-641). IEEE.

9. Abbas, S. A., & Maryoosh, A. A. B. Data Security for Cloud Computing based on Elliptic Curve Integrated Encryption Scheme (ECIES) and Modified Identity based Cryptography (MIBC).

10. Rewagad, P., & Pawar, Y. (2013, April). Use of digital signature with diffie hellman key exchange and AES encryption algorithm to enhance data security in cloud computing. In *Communication Systems and Network Technologies (CSNT), 2013 International Conference on* (pp. 437-439). IEEE.

11. Kaur, Randeep, and Supriya Kinger. "Analysis of security algorithms in cloud computing." *International Journal of Application or Innovation in Engineering and Management* 3, no. 3 (2014): 171-6.

12. Kaaniche, N., Boudguiga, A., & Laurent, M. (2013, June). ID based cryptography for secure cloud data storage. In *CLOUD 2013: IEEE 6th International Conference on Cloud Computing* (pp. 375-382). IEEE.

13. Tirthani, N., & Ganesan, R. (2014). Data Security in Cloud Architecture Based on Diffie Hellman and Elliptical Curve Cryptography. *IACR Cryptology ePrint Archive*, *2014*, 49.

