A Hybrid Method to Reduce Congestion and Provide Data Security in Cloud Computing

[1]Md Asif Mushtaque, [2]Harsh Dhiman, [3]Shahnawaz Hussain, [4]Pinki
M.Tech (CS&E): School of Computing Science and Engineering
Galgotias University
U.P, India
[1]asifmushtaque000@gmail.com, [2] hharshdhiman@gmail.com , [3]shahnawazhussaincse@gmail.com
[4]pinks.angel90@gmail.com

Abstract—Cloud Computing is used everywhere because it provides on demand access of resources and reduces cost, it offers dynamic allocation of resources for guaranteed and reliable services. Users store their data on a single virtual server, when user wants to access any data that data might be changed or modified by unauthorized people for malicious purpose because user's do not have direct control of data So security is a big challenge for cloud computing, to enhance the reliability of services it is necessary to increase the security level in the cloud where the user should free from integrity, authentication, correctness or confidentiality. In this paper, I proposed a new method for congestion control and data security in cloud computing. Many authors have given their ideas on data security in cloud but no one gives the full control to the user. This approach control congestion by reducing the size of data. In the existing method cloud service provider uses compression technique to reduce the size of data after that encrypt data that increases the size of ciphertext in comparison to compressed data. A new encryption technique is used to encrypt the compressed data that does not increase the size of ciphertext. There are two main advantages of this method (i) it does not increase the size of ciphertext comparison to existing method and (ii) reduce the congestion between server and user by fast transmission. If we reduce the size of data then data would be transferred between user and server in less amount of time so in this case this method controls the congestion between user and server.

Keywords- Cloud Computing, AHSP Algorithm, congestion control, Cloud Storage, Cloud Security, Data Privacy, Data Security, Integrity, Confidentiality, Reliability.

I. INTRODUCTION

Cloud computing is a type of computing which provides the facility to use resources available on cloud system, in other word we can say that it is a model where resources are retrieved through network, it allows user to use technology enabled services through the internet [1, 2]. Cloud computing is an internet based service where the user can easily use storage, services without knowing how it is actually working internally. Cloud computing is a collection of virtual machines in which user only uses the services provided by the virtual machines they don't have a control on virtual machines. In cloud computing several organizations store their data on a single virtual server sometimes multiple operating systems are executed on a single virtual server, in this case there is chances of threat from other machine. So there is a need of high level security especially in public cloud system.

There are some main characteristics of cloud computing:

- Location Independence: it means location of device is not necessary for the user where it is located, the user only uses the services through internet. They don't need to know what kind of device is used by user or cloud; they only know how to use it [2].
- Multitenancy: it means a single piece of resource is used by multiple users. A single user is known as the tenant. So cloud provides a facility to use a single instance of resource across a large pool of users.
- Reliability: uses multiple redundant (copied) sites which make it well suitable for business and disaster recovery.
- Measured service: it means cloud automatically measures about services, resources used by users and providing transparency from users.
- Scalability: modification of services quickly according to user's requirement without any problem in existing services.
- Security: due to centralization of data security is the main characteristics of data. It provides better security but need to increase the security level.

- On demand self service: in which user can use the services according to their need without interference of the service provider.

II. RELATED WORK

In [1], the authors have proposed a model and where they discussed on the multi level sign agreement from a service provider for data security but there would be some problem. If a service provider sign on the agreement and data is accessed by hacker then service provider would be responsible, according to [1] data can be protected only from service provider not from outside hackers. So, this model is not very effective for user and service provider. In [3], the authors used a HMAC scheme to encrypt data and used two times encryption at the time uploading a file and distribution of file. Uses of two times encryption means it will take double time which increases the time complexity. Many authors have given their ideas some of them uses existing method and some authors have proposed their new ideas.

III. PROPOSED MODEL

In this section I discussed on proposed model how they would be beneficial for users and service provider.

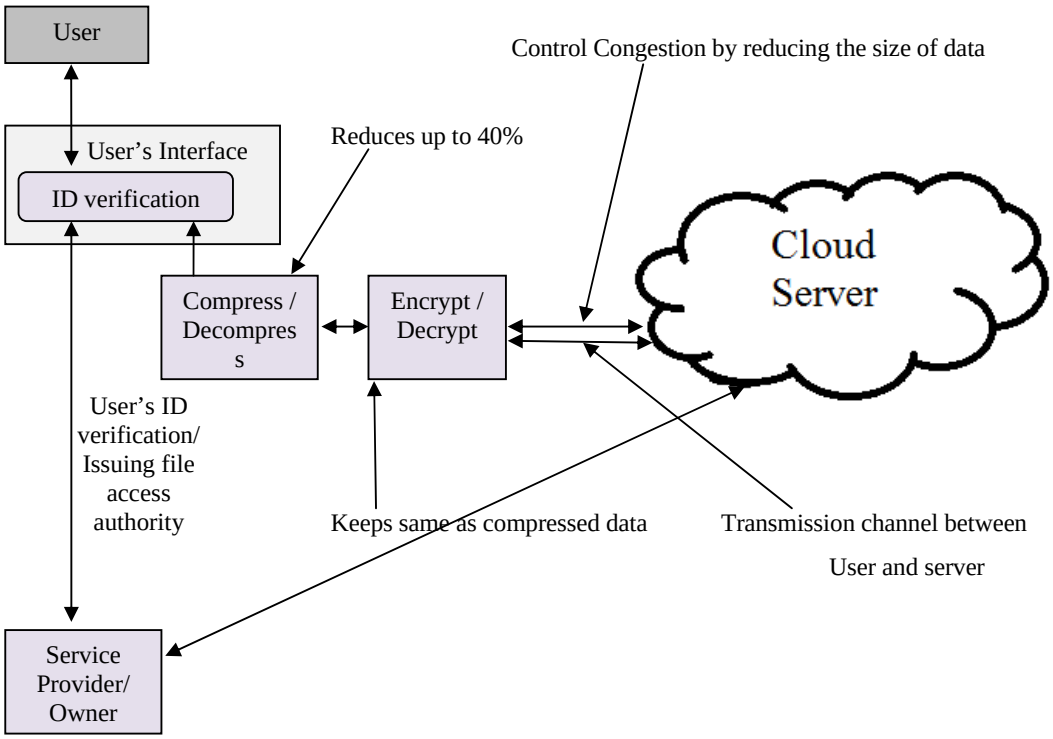

Figure1. Structure of Proposed Model

Figure1 shows the complete structure of the proposed model, in this model when the user is allowed to upload/download file to or from cloud server, if the user wants to upload their data then that data will be compressed first then encryption is performed. In this model I am compressing data by using existing method Arithmetic coding and to encrypt I proposed own encryption algorithm (AHSP Algorithm) when we compress data then it'll reduce size up to 20-25%. The advantages of using compression technique is that after reducing the size of data it'll take 20-25% less storage space of cloud server, this technique saves the space of server another advantage is if we reduce the size then we can transfer data within less time in comparison to the original file because channel has the limited bandwidth. For example: suppose the channel capacity is to transfer data 1MBPS, the size of the original file is 10MB then this file would be transferred in 10 seconds. After using this model, the size of the original file is 10MB, size of the compressed file is 7MB, and now this file would be transferred in 7 seconds. When file would be transferred within 7 second then it reduces the congestion on channel or between cloud server and user because cloud server supports multitenancy feature where a single resource is used by multiple users. In existing method all encryption techniques require same of maximum storage space for encrypted data in comparison to the original data. This model keeps more effect on larger file where it can reduce maximum size. If we use both approach then security becomes high then service provider can provide reliable service with high security.

➢ PROPOSED ALGORITHM: AHSP ALGORITHM

This is an encryption algorithm performed on the user side, when users are allowed to access the services of cloud the encryption algorithm will performed by clicking on upload or download button. The secret key should be entered by the user. In the existing encryption algorithm we use a reverse process of encryption to decrypt the file, the key feature of AHSP Algorithm is that there is no need to implement/design a decryption algorithm it decrypts the data by the same process. This algorithm performs a variable number of rounds depend on the key length and in each round this algorithm generates a random key. It is impossible to predict the random key in each round so this algorithm provides a high security level. There are no any limitations of key length but the key size should be square of any number.

Encryption Algorithm:

Step 1: read file example.txt

Step 2: read two different keys of equal length (key1 and key2)

Step 3: find l = length of key1

Step 4: key1= {key1+ rand(l)} x key2

Step 5: initialize R=0 where R is the number of rounds

Step 6: rand_key_generation (key1)

- Reverse kry1 and store as RevOfKey1
- Convert each character into its ASCII value
- Sum= $\sum$ASCII * P**.** Where P is the position of character in key1.
- Calculate mod (sum, l)
- Arrange key1and RevOfKey1 into matrix form
- Add mod value with each character in key1 matrix.
- N1w key1 $\bigoplus$ RevOfKey1
- Obtain NewKey

Step 7: Now Perform Exclusive-Or operation with example.txt and NewKey.

- Ciphertext= example.txt $\bigoplus$ NewKey

Step 8: now initialize key1= NewKey and R= R+1.

Step 9: Repeat step 6, 7 & 8 until R<= Length of the key

Step 10: Stop.

Decryption is the same process of encryption.

Example: Suppose key1= aw23e45t6 and key2= fd34eft56 then key1= key1 + rand(l) x key2, where rand() is the random function.

Now, Key1= aszxwdefy

L=9

RevOfKey1= yfdewxzsa

a=97, s=115, z=122, x=120, w=119, d=100, e=101, f=102, y=121.

Sum= 97*1+115*2+122*3+120*4+119*5+100*6+101*7+102*8+121*9= 4980

mod= (4980, 9)= 3

a	s	z
x	w	d
e	f	y

97	115	122
120	119	100
101	102	121

Key1 _matrix

y	f	e
d	w	x
z	s	a

122	102	101
100	119	120
122	115	97

RevOfKey1_matrix

100	118	125
123	122	103
104	105	124

After adding mod value to key1_matrix

100	118	125
123	122	103
104	105	124

$\oplus$

122	102	101
100	119	120
122	115	97

Xoring between key1matrix after add mod value and RevOfKey1_matrix

30	16	24
31	13	31
18	26	29

ASCII value of key1= 97, 115, 122, 120, 119, 100, 101, 102, 121.

ASCII of newkey = 30, 16, 24, 31, 13, 31, 18, 26, 29.

The character for the ASCII value of newkey is not available on keyboard. These are the reserved notification symbol for system so it is impossible to crack the new key. This process is performed in each round of this algorithm so key1 and newkey would be change in each round.

The first key is should be entered from user, it may be password of user or service provider can add extra button for a secret key but the key service provider should be unaware from secret key and providers have to cooperate their user to secure data.

IV. RESULT

# Rounds		**Key1**	**Newkey**	**Plaintext**	**ciphertext**
Initial Stage	**Key1=** aw23e45t6, Key2= fd34eft56				
R1		qazwsxedc	Po)=ji7>1	algorithm	Uj8”-!3dt
R2		Po)=ji7>1	Yvu&@t$.;	Uj8”-!3dt	Ki0m^+c%?
R3		Yvu&@t$.;	H64=<[o*l	Ki0m^+c%?	Fw#}jr|p!
R4		H64=<[o*l	v-({p\k@^	Fw#}jr|p!	G4d*?-A]. **(Final Ciphertext for key length 9 bytes)**

Table1. Result of AHSP Algorithm for key Length 9bytes (72 bits)

In Table1 we can see that how the key is changing and it is unexpected key, sometimes it may be happen new key would be invisible symbol that’s why I am showing all these characters because character of each ASCII is not available on the keyboard but in the above example we can see that how invisible character means the system notification symbol is used as a new key. This algorithm is not only for key length of 9 bytes it supports

variable key length but length should be square of any integer. If we use the maximum length of the key then it is more secure because for maximum key length there would be the maximum number of rounds which keeps user's data secure from unauthorized user. Like existing encryption algorithm AHSP Algorithm doesn't require

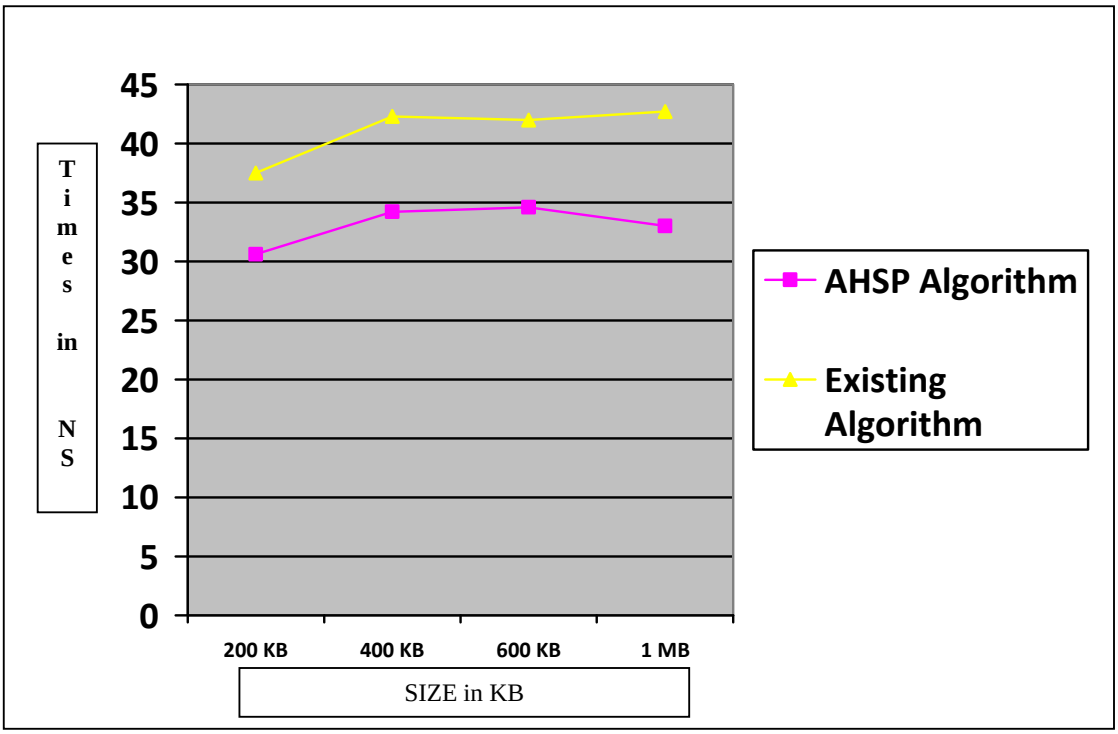

Fig. 2: Transmission Time of Existing Algorithm and AHSP Algorithm

V. CONCLUSION / FUTURE WORK

Security of stored data is a very big challenge for cloud service provider, on cloud computing users only access their data and use the service provided by the service provider, users don't know anything about its security and don't have any control on their data. So, in this paper we proposed our new algorithm to give control to users for their data security. From the above example and Table1 we analyzed that this algorithm provides better security and also gives full control to users. This paper also presents the hybrid model for cloud in this model two different techniques are used compression and encryption. For compression I used the existing method and to encrypt I used own encryption algorithm. We know that cloud server contains very huge amount of data and multiple user accesses a cloud server at the same time so this hybrid model reduce the size of data that saves the storage space of cloud server and increase throughput of cloud computing. Finally after all experiment we found that AHSP Algorithm provides better security which is controlled by the users itself.

VI. ACKNOWLEDGMENT

I want to say thank to all fellow students, proof readers, respected faculties and my supportive friend. Especially to my family and that person who guides me through the way, he continually and persuasively conveys a spirit of adventure in regard to my thesis/project work Md. Tauqir Azam Kausar. Thank You So Much.

REFERENCES

[1] Ashutosh Satapathy, J. Chandrakant Badajena and Chinmayee Rout," A Secure Model and Algorithms for Cloud Computing Based on Multicloud Service Providers", International Journal of Engineering Research & Technology (IJERT), Vol. 2 Issue 12, December – 2013.

[2] http://en.wikipedia.org/wiki/Cloud_computing accessed on 25th April 2014.

[3] G. Rahul Reddy and N. J. Subashini, "Secure Storage Services and Erasure Code Implementation in Cloud Servers", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 1, January – 2014.

[4] Kamyab Khajehei, "Secure Communication in Cloud by Using ECC Algorithm", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 1, January – 2014.

[5] Pradnyesh Bhisikar and Prof. Amit Sahu, "Security in Data Storage and Transmission in Cloud Computing", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 3, March 2013.

[6] Abhishek Patel and Mayank Kumar, "A Proposed Model for Data Security of Cloud Storage Using Trusted Platform Module", ", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 4, April 2013.

[7] Kangchan Lee, "Security Threats in Cloud Computing Environments", International Journal of Security and Its Applications, Vol. 6, No. 4, October, 2012.

[8] Rohit Maheshwari and Sunil Pathak, "A Proposed Secure Framework for Safe Data Transmission in Private Cloud", International Journal of Recent Technology and Engineering (IJRTE), ISSN: 2277-3878, Volume-1, Issue-1, April 2012.

[9] Anthony T. Velte, "Cloud Computing A Practical Approach 1st Edition", Tata Mcgraw Hill, ISBN-13-9780070683518, 2009.
[10] Md Asif Mushtaque, H, Dhiman, S. Hussain and S. Maheshwari, "Evaluation of DES, TDES, AES, Blowfish and Twofish Encryption Algorithm: Based on Space Complexity", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 4, April – 2014.
[11] Mansoor Ebrahim, Shujaat Khan and Umer Bin Khalid, "Symmetric Algorithm Survey: A Comparative Analysis", International Journal of Computer Applications (0975 – 8887) Volume 61– No.20, January 2013.
[12] Md Asif Mushtaque and Mr. Khushal Singh, "Feasibility Evaluation of Symmetric Key Encryption Techniques for Wireless Channel and Disk Storage", International Journal for Research in Applied Science and Engineering Technology (IJRASET), Vol. 2 Issue V, May 2014.