Cloud Security based on ECC- Diffie-hellman Protocol and Storage Optimization using compression technique

Shefali Gupta, Mr. Deepak Kumar Xaxa
M. Tech Student (CSE)
Department of Computer Science, Mats University, Aarang, C.G. India

***Abstract* – Cloud Computing is an Internet-centric way of computing. Internet is representing either the medium or the platform through which cloud computing services are delivered and made accessible. Cloud computing has become argot in the area of high performance distributed computing as it provides on demand access to share resources over internet. In cloud computing massive data storage is one of the great challenging tasks in terms of the reliable storage of sensitive and quality of storage service as well as security issues is also arises. This paper proposes privacy preserving authenticated access control scheme with optimal allocation method along with data compression technique. Which is safer then the opposite system the planned system may concentrate to the storage capability via exploitation Huffman compression methodology and applicable planning used for resource allocation in cloud. This system generates the error rate which is 0.00882. This is the minimum error rate ever proposed in various previous papers; this less error rate leads to high security of the encrypted data.**

***IndexTerms* –Cloud Security, ECC, Diffie-hellman, Compression Technique in cloud.**

I. INTRODUCTION

Research in Cloud computing, conjointly called on-demand computing is receiving lot of attention from both environment like academic as well as industrial that could be a quite Internet-based computing which has shared process resources and knowledge to computers and alternative devices on demand. Existing methodology resolve the problem arises in the previous papers. There are many researches has been done in cloud computing in which some are worked on cloud security where as some worked on scheduling. There is another problem specify for cloud is storage enhancement. In proposed methodology scheduling perform for resource allocation which is probabilistic based dynamic allocation algorithm. Much of the data stored in clouds in highly sensitive. Thus Security and privacy are very important issues in cloud environment. In one hand, the user ought to attest itself before initiating any dealing, and on different hand, it should be ensured that cloud doesn't tamper with the info that's outsourced. ECC-DH (Elliptical curve cryptography with Diffie-Hellman) used for resolving this issues only authenticated user can be access data within the cloud.

Clouds offer a really sizable amount of resources, together with platforms for computation, knowledge centers, storages, Networks, firewalls and code in style of services. At an equivalent time it additionally provides the ways that of managing these resources such users of cloud will access them while not facing any quite performance connected issues. In lossless compression a variation known as adaptive Huffman committal to writing involves calculative the possibilities dynamically supported recent actual frequencies within the sequence of supply symbols, and dynamic the committal to writing tree structure to match the updated chance estimates. it's used seldom in apply, since the price of change the tree makes it slower than optimized adaptive arithmetic committal to writing, that's a lot of versatile and contains a higher compression.

The Proposed system addressed the problem of service request scheduling in cloud computing system. We introduce a cloud environment which is very secure in the manner of access control via technique of elliptical curve cryptography and for key exchanging this system is uses diffie-hellman algorithm to enhance the security level in cloud [11]. To enhance the storage capacity of cloud system has an compression technique which is Huffman Encoding technique. In cloud computing allocation tasks is also an issue there are static and dynamic allocation methods among these two systems is uses Probabilistic based dynamic allocation algorithm for allocating resource to the users. This paper address the subsequent problems associated with the cloud computing:

- Security in cloud: Failure to confirm applicable security protection once mistreatment cloud services might ultimately end in higher prices and potential loss of business, therefore eliminating any of the potential edges of cloud computing.

- Storage in Cloud Environment: A lot of that data exists on a cloud, therefore ought to shrewdness to make sure data warehousing is increased and does not transform a frail affiliation in cloud stage.

- Load Balancing in Cloud Computing: Load equalization is one in every of major issue within the cloud surroundings which will be achieved via applicable planning formula(Scheduling or allocation method).

II. PROPOSED SYSTEM

Proposed system contain Key choice and key exchange policy by victimization diffie-hellman and ECC(Elliptical curve cryptography) so security problems in cloud may be resolve. For storage and load reconciliation Huffman and probabilistic aware dynamic allocation algorithmic program is employed so cloud optimization is achieved.

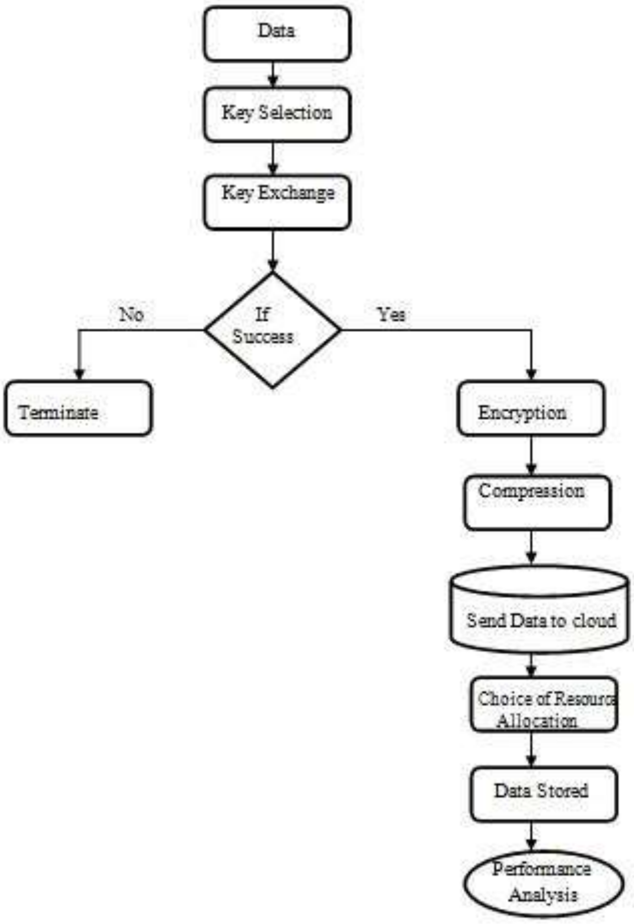

Fig.1. Methodology of the proposed system

A. ECC-DH (Elliptical Curve Cryptography with Diffie-hellman algorithm)

Security in cloud can be achieved via Elliptical Curve Cryptography with Diffie-hellman algorithm.

Diffie- Hellman Algorithm: In Cloud computing domain, there square measure set of vital policies, that embrace problems with privacy, anonymity, security, liability and dependableness. Diffie-Hellman key exchange protocol is first public key cryptography scheme. It was proposed by Witfield Diffie and Martin Hellman in 1976 [2]. The Diffie-Hellman key agreement protocol was the first practical method for establishing a shared secret over an unsecured communication channel. It uses two keys -- one secret and other private key. This scheme is based on the difficulty of computing logarithmic functions for prime exponents. This is known as Discrete Logarithm Problem (DLP)

Steps in the algorithm

- Alice and Bob agree on a prime number p and a base g.
- Alice chooses a secret number a, and sends Bob (g^a mod p).
- Bob chooses a secret number b, and sends Alice (g^b mod p).
- Alice computes ((g^b mod p) ^{a} mod p).
- Bob computes ((g^a mod p) ^{b} mod p).
- Both Alice and Bob can use this number as their key. Notice that p and g need not be protected.

Elliptical curve cryptography: Elliptical Curve Cryptography is an approach in cloud computing that's supported public key cryptography to supply on demand computing Security to the knowledge. Different researches have focused on the fact that user generally has to access large volumes of data from the cloud in a secured manner. ECC generates keys through the properties of the elliptic curve equation rather than the normal technique of generation because the product of terribly massive prime numbers [2]. The technology is employed in conjunction with most public key coding ways, like RSA, and Diffie-Hellman. In ECC majority of public-key crypto (RSA, D-H) use either integer or polynomial arithmetic with very large polynomials. Elliptic Curve Discrete Logarithm Problem. Elliptic Curve Cryptography is defined with help of following parameters as:

$$P = (q, FR, a, b, c, G, n, h)\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

- q: the prime umber or 2m that defines curve's form.
- FR: field representation.
- a, b: the curve coefficients.
- G: the base point (Gx, Gy).
- n: the order of G. It must be big prime number.
- h: cofactor co-efficient [22].

Elliptic Curves (EC) over finite fields are used to implement public-key protocols. The Elliptic curve is defined on either prime field GF (p) or binary field GF (2n). Since arithmetic in latter field is much faster, we work in GF (2n). An elliptic curve E is defined by the simplified projective coordinates as follow:

$$Y^2Z + XYZ = X^3 + aX^2Z + bX^3\ldots\ldots\ldots\ldots.. (2)$$

This public key cryptography scheme is defined over two fields: prime Galois Field, GF (p), or over binary extension Galois Field, GF (2m). In GF (p), the equation of Elliptic Curve is:

$$Y^2 \bmod p = x^3 + ax + b \bmod p\ldots\ldots\ldots\ldots\ldots (3)$$

Where:

$$4a3 + 27b2 \bmod p \neq 0\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. (4)$$

with elements of GF (p) as integers between 0 and p-1. In GF (2m), the equation of Elliptic Curve is given by:

$$y2 + xy = x^2 + ax^2 + b\ldots\ldots\ldots\ldots\ldots\ldots\ldots(5)$$

where: b ≠ 0. Over GF (2m), rules for point addition and point doubling can be implemented [22].Elliptic Curves on R, Elliptic curves, known and studied since centuries, used by Andrew Wiles in his proof of Fermat's last theorem are algebraic curves or Weierstra curves.

$$y^2 = x^3 + ax + b$$

B. Compression Technique For cloud storage

The greatest advantage of cloud storage is it enables users at any time access data. In cloud system, storage management system automatically analyses user's requirements and locate and transform data, which greatly facilitate the users. Compression can be used to reduce the size of files and speeding up the transmission time over networks [12]. However, not all compression techniques have the same features and capabilities to improve the performance of transmission over networks.

Huffman Algorithm: A Huffman code is a particular type of optimal prefix code that is commonly used for lossless data compression. The Huffman algorithm is a so-called "greedy" approach to solving this problem in the sense that at each step, the algorithm chooses the best available option. It turns out that this is sufficient for finding the best. Adaptive Huffman algorithms develop the tree while calculating the frequencies and there will be two trees in both the processes. In this approach, a tree is generated with the flag symbol in the beginning and is updated as the next symbol is read. In Huffman Coding the characters in a data file are converted to binary code. In Huffman coding, the most common characters in the file have the shortest binary codes, and the characters which are least common have the longest binary code [13].

The algorithm to generate Huffman code is:

- Parse the input and count the occurrence of each symbol.
- Determine the probability of occurrence of each symbol using the symbol count.
- Sort the symbols according to their probability of occurrence, with the most probable first.
- Then generate leaf nodes for each symbol and add them to a queue.
- Take two least frequent characters and then logically group them together to obtain their combined frequency that leads to the construction of a binary tree structure.

- Repeat step 5 until all elements are reached and there remains only one parent for all nodes which is known as root.
- Then label the edges from each parent to its left child with the digit 0 and the edge to right child with 1.
- Tracing down the tree yields to "Huffman codes" in which shortest codes are assigned to the character with greater frequency.

C. Resource Allocation algorithm used for Load Balancing in Cloud Environment

The service providers have a huge number of users, they have to deal with massive data, which are more difficult to schedule [1]. The requests from users must be scheduled efficiently, so scheduler needs to calculate a proper sequence to response those requests.

Probabilistic workload-aware dynamic resource allocation

In our work, probabilistic models are used in the decision making process, to describe, drive and analyze cloud resource elasticity The pseudo-code of the resource allocation algorithm is presented in Algorithm 1.

Algorithm 1: The Proposed resource allocation algorithm

Input: R,C,S,T

Output: QoS estimation

```
1.  $S_{tvm} \leftarrow 0.2$;
2.  $S_{tpm} \leftarrow 0$;
3.  $P_{st}$ = // Physical Machine Switching time;
4.  for r in R do
5.      $\alpha_r' = []$;
6.  for t in [1,T] do
7.      $\omega_t^* =$ Algorithm 2(R,C,S,T, $\alpha'$);
8.      $\omega_t' \leftarrow 0$;
9.      for r in R do
10.         $\omega_t' \leftarrow \omega_t' + \omega_{r,t}'$;
11.         $\alpha_r'$.append($\alpha_{r,t}'$);
12.     $\Delta = S_{tvm} . \omega_t'$ ;
13.     If $\Delta > 0$ then
14.         $S_{tvm} \leftarrow \max(S_{tvm}(1.0 - s') P_{st})$;
15.     else
16.         $S_{tvm} \leftarrow (1.0 + s) P_{st}$;
17.         $Q \leftarrow Q - \Delta / \omega_t' T$;
18. Return Q;
```

Algorithm 1 main loop iterates over each time t (Lines 6–17). Initially, it obtains an estimation of resource demand using Algorithm 2, described below (Line 7). Afterwards, it computes the actual resource demands (Lines 8 and 10) and updates a by appending a r,t to each list a (Line 10). Once the difference Δ between the estimated and the actual resources demand has been computed (Line 12), the algorithm adjusts the value of according to the progression of this error. Namely, if the error in t is greater than 0, S_{tvm} becomes the maximum between (1.0 – s')S_{tvm}, s'>0 and S_{tpm} (Line 14). Both s and s are constants that contain the value 0.5ns.Conversely, if it is smaller than 0, S_{tvm} is multiplied by (1.0 + s) and Q is incremented by -Δ/(w't T) (Lines 16 and 17). Finally, after the end of the loop, Q is returned. Here S_{tvm} defines the switching time of virtual machine S_{tpm} define the switching time of physical machine. P_{st} define the switching time in physical machine.

Algorithm 2: Estimation of resource demand

Input: R C S ,time- slot t,α'

Output: Number ω_t of resources to be allocated

```
$\omega_t^* \leftarrow 0$;
for r in R do
    $C = S_{r.t}$ ;
    $\mu_r.\sigma_r^2 \leftarrow$ MLE ($\alpha_r'$);
    $\alpha_r$ = draw ($N(\mu_r,\sigma_r^2)$) );
    $\omega_t^* \leftarrow \omega_t^* + \omega_c \alpha_r$;
Return $\omega_t^*$ ;
```

Algorithm 2 estimates the resource demand which has to be allocated to the used. Here w_t defines the resource demand by the user to be allocated.

III. EXPERIMENTAL RESULTS & ANALYS

The graph shows the overall performance of the system in comparison with the previous work done. The experimental results shows the workflow of the proposed methodology

Entropy Value – This value defines the security of the encrypted data .In this context, entropy is the expected average of the information contained in each message. 'Messages' can be modeled by any flow of information.

Entropy Value = -∑PI* log(PI) where PI is the data

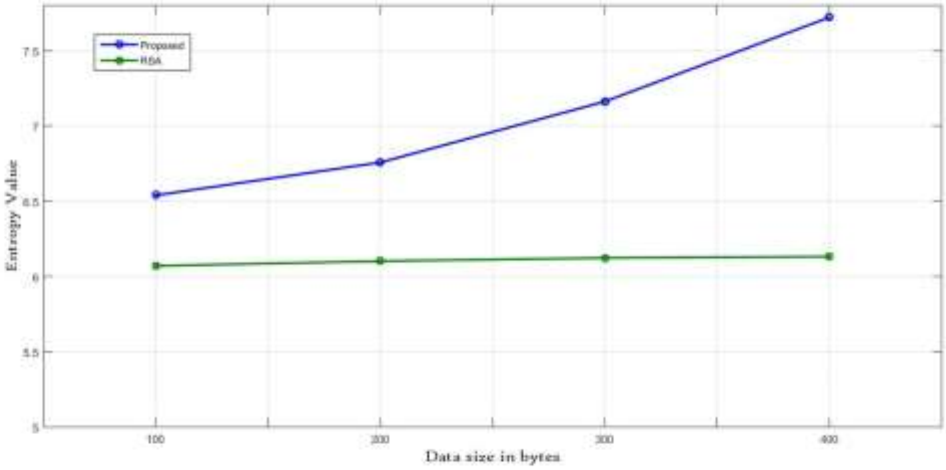

Fig 2 Graph of Entropy value

This graph defines the relation between the data size and entropy value in which proposed value is more than the RSA. The graph concludes that RSA is lagging behind, in comparison with the proposed system which result in High Secured data.

Error Rate- Error rate of a channel. The frequency with that errors or noise square measure introduced into the channel. Error rate is also measured in terms of incorrect bits received per bits transmitted.

Error Rate = ∑ [Original – Received]^2/ Length of the original message

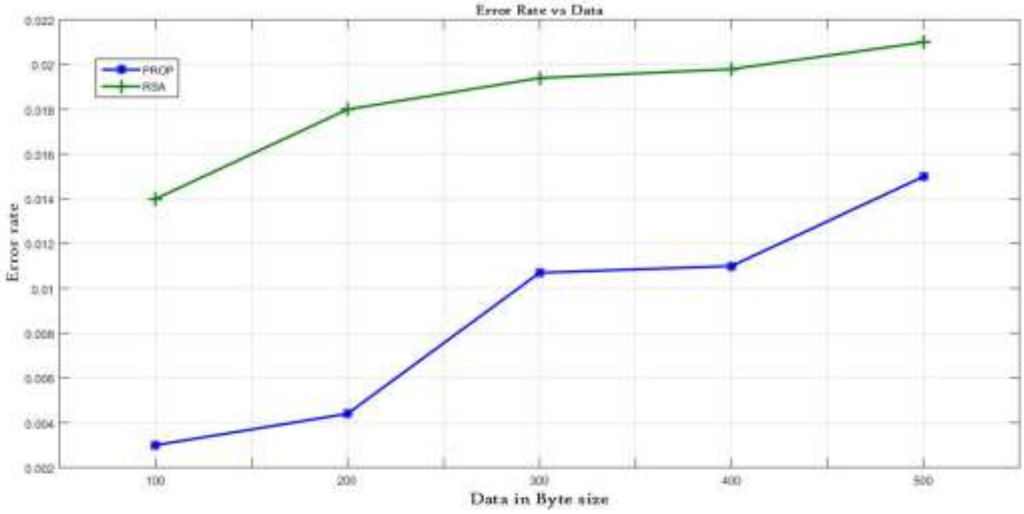

Fig 3 Graph of Error rate

This graph defines the relation between data and error rate, by analyzing this graph the conclusion is that the proposed system has minimum error rate as compared with the previous system results.

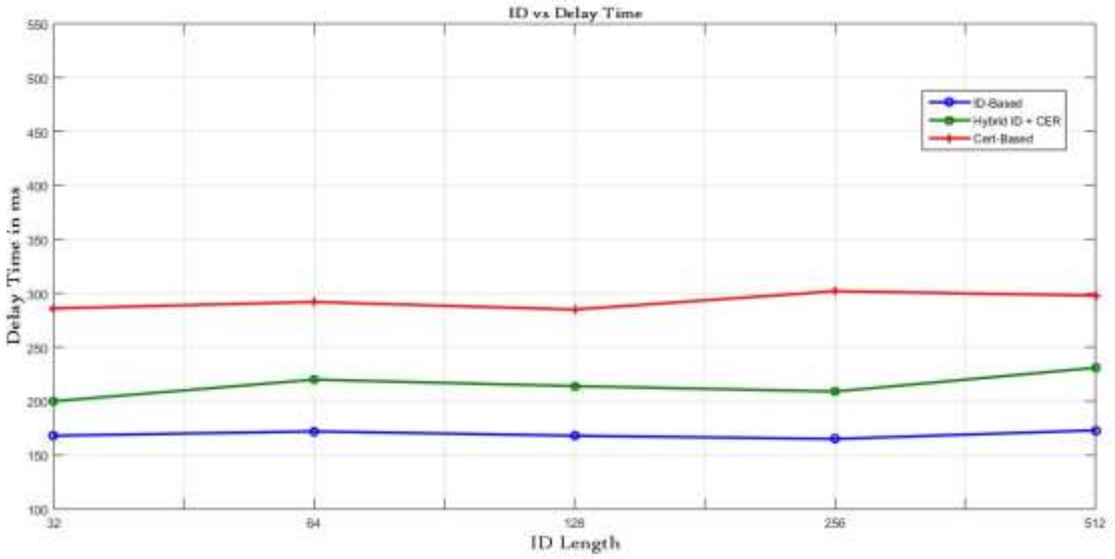

Fig 4 Graph of Time Delay

This graph describes about the comparison of delay time of the transmission between the Certificate based , hybrid (Id +Certificate) and ID based. Where the ID based transmission have the minimum transmission delay time.

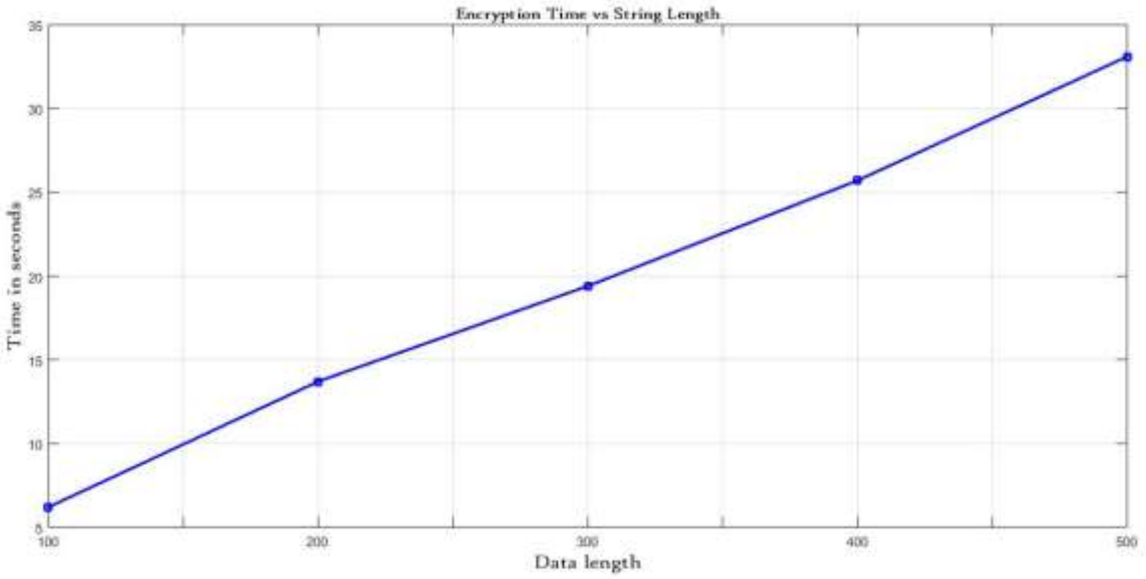

Fig 5 Graph of Encryption time vs string length

The graph below describe the encryption time according to the length of the message.

III. CONCLUSION

Successful implementation of the proposed system which is “Cloud Security framework based on ECC and compression technique and Diffie-hellman protocol”. The proposed method concludes that the error rate is minimum then the previous paper. This method consists of the Security in cloud which is handle by two keys which is schedule by ECC-DH that gives the entropy value which is greater than the other, that means this system is more secure than the other system. As per study, the previous papers didn’t focus on Storage of cloud, which can be reduced via Huffman compression technique. This methodology is related to the security issues in cloud has been successful implement with average error rate with two types of methods for storage and resource allocation in cloud computing. In the proposed system scheduling is done with various key size factors. The Average Error rate is 2.94% and the Average value of Entropy value that is defining the Security of encrypted data is 93.81%.

REFERENCES

[1] Fernando Koch, Marcos D. Assuncao, Carlos Cardoha, Macro A.S. Netto “Optimizing resource costs of Cloud computing for educatin”, 2015 Elsevier.Ltd

[2] Sushmita Ruj, Member, IEEE, Milos Stojmenovic, Member, IEEE, and Amiya Nayak, Senior Member, IEEE, "Decentralized Access Control with Anonymous Authentication of Data Stored in Clouds", IEEE Transaction and distributed system vol 25 feb 2014.
[3] Mohammad Iftekhar Husaina,n, Steven Y. Kob, Steve Uurtamob, Atri Rudrab, Ramalingam Sridharb "Bidirectional data verification for cloud storage", 2014 Elsevier.Ltd
[4] Nikos Tziritas , Samee Ullah Khana, Cheng-Zhong Xua, Thanasis Loukopoulos Spyros Lalis " On minimizing the resource consumption of cloud applications using process migrations" 2013 Elsevier.Ltd**(3)**
[5] Nidhi Bansala, Amitab Mauryaa, Tarun Kumara, Manzeet Singha, Shruti Bansalb "Cost performance of QoS Driven task scheduling in cloud computing" 2015 Elsevier.Ltd.
[6] Dong Yuan∗, Yun Yang, Xiao Liu, Jinjun Chen, "On-demand minimum cost benchmarking for intermediate dataset storage in scientific cloud workflow systems", 2011 Elsevier.Ltd.
[7] Maurizio Giacobbe, Antonio Celesti, Maria Fazio∗, Massimo Villari,Q1 Antonio Puliafito "Towards energy management in Cloud federation: A survey in the perspective of future sustainable and cost-saving strategies" 2015 Elsevier.Ltd.
[8] Lifei Weia, Haojin Zhu , Zhenfu Cao , Xiaolei Dong , Weiwei Jia , Yunlu Chen, Athanasios V. Vasilakos, "Security and privacy for storage and computation in cloud computing" 2013 Elsevier.Ltd.
[9] Mouna Jouinia , Latifa Ben Arfa Rabaia, "Comparative Study of Information Security Risk Assessment Models for Cloud Computing systems" 2016 Elsevier.Ltd.
[10] Rizwana Shaikha, Dr. M. Sasikumarb, "Data Classification for achieving Security in cloud computing" 2015 Elsevier.Ltd.
[11] Rabi Prasad Padhy, Manas Ranjan Patra, Suresh Chandra Satapathy, "Cloud Computing: Security Issues and Research Challenges" (IJCSITS) Vol. 1, No. 2, December 2011.
[12] K.Govinda , Yuvaraj Kumar, "Storage Optimization in Cloud Environment using Compression Algorithm" , IJETTCS Volume 1, Issue 1, May-June 2012.
[13] Aarti, "Performance Analysis of Huffman Coding Algorithm", Volume 3, Issue 5, May 2013 www.ijarcsse.com.
[14] Dalvir Kaur ,Kamaljeet Kaur, "Data Compression on Columnar-Database Using Hybrid Approach (Huffman and Lempel-Ziv Welch Algorithm) Dalvir", Volume 3, Issue 5, May 2013www.ijarcsse.com.
[15] Zhongyuan Lee1, Ying Wang1, Wen Zhou;"A dynamic priority scheduling algorithm on service request scheduling in cloud computing", 978-1-61284- -8/ll/$26.00 ©2011 IEEE.
[16] Xiaocheng Liu, Albert Y. Zomaya, Fellow IEEE, Chen Wang, Bing Bing Zhou, Junliang Chen, Ting Yang, : Priority-Based Consolidation of Parallel Workloads in the Cloud. IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 9, September 2013**.**
[17]Feng Liu, Haitao Wu, Xiaochun Lu, Xiyang Liu, Lei Fan, Genetic algorithm based optimization model for reliable data storage in cloud environment, Adv. Sci. Technol. Lett. 50 (2014) 74e79**.**
[18]Shachee Parikh and Richa Sinha, "Double Level Priority based Optimization Algorithm for Task Scheduling in Cloud Computing" International Journal of Computer Applications Vol. 62, No. 20, 2013
[19]Muhammad Baqer Mollah, Kazi Reazul Islam, Sikder Sunbeam Islam, "Next generation of computing through cloud computing technology", 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), 2012.
[20] Xiaocheng Liu, Albert Y. Zomaya, Fellow IEEE, Chen Wang, Bing Bing Zhou, Junliang Chen, Ting Yang, : Priority-Based Consolidation of Parallel Workloads in the Cloud. IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 9, September 2013**.**