Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 6, June 2015, pg.361 – 369

RESEARCH ARTICLE

An Efficient and Secure Remote Data Back-up Technique for Cloud Computing

Somesh P. Badhel[1], Prof. Vikrant Chole[2]

[1]Department of Computer Science and Engineering
[1]G. H. Raisoni Academy of Engineering and Technology, Nagpur, India
[2]Department of Computer Science and Engineering
[2]G. H. Raisoni Academy of Engineering and Technology, Nagpur, India

Abstract: Cloud computing provides various kinds of services to its users. Storage-as-a-service is one of the services provided by cloud infrastructure in which large amount of electronic data is stored in cloud. Data backup and disaster recovery in the event of network or cloud service provider failure is an important issue in Cloud computing paradigm. In situations like Flood, Fire, earthquakes or any hardware malfunction or any accidental deletion our data may no longer remain available. In this paper we propose a solution for Data recovery in cloud computing platform by allowing the process of migration from one cloud service provider to another in much efficient way.

Keywords: Seed Block, Random number generator, Remote Server, Main Server, J-bit encoding

I. INTRODUCTION

Cloud Computing is a computing technology that is based on Internet. It involves sharing of resources in spite of having own local storage and devices to handle different services. Cloud Computing has become a gigantic technology that surpasses all the other older computing technologies (Grid Computing). Cloud Computing provides various advantages as compared to previous computing technologies. Cloud Computing provides Supercomputing and high performance computing power to its clients at a very low cost.

Cloud computing involves networks of a large groups of servers that are typically running a low-cost consumer PC technology along with specialized connections that spread data-processing storage across them. This shared Cloud infrastructure contains large pools of systems which are linked together. These virtualized pools are used to maximize the power of cloud computing and data is stored in the form of virtualized pools.

As Cloud Computing involves sharing of computing resources there are large number of users that share the same storage and other computing resources. Therefore there is a strong need for a mechanism to prevent other users to access your confidential and useful data either intentionally or accidentally also if it happens that some other user on the cloud access your data and makes some modification or any deletion then it must be recoverable to its original state in an efficient way. Also the stored data is at danger due to any natural calamities such as any Flood, Fire etc. Natural disasters for example flood can make recovering data an impossible task. Flood water contains contaminated water which may contain dust, sand dregs and other materials that may affect the platters and sectors of the hard drive. The parts of the hard drive may seize up just as an engine that will start slugging in it. Also many businesses that relied on electronic data suffer total or temporary data loss due to hardware damage and failure. Data integrity is another issue while recovering lost data.

This paper is organized as follows: Section II focuses on the existing methods and techniques for the backup purpose in the cloud computing domain that are successful to some extent. In Section III, we discuss about the remote data backup server and proposed efficient technique. Section IV describes the detailed implementation of the proposed backup technique and Section V shows the results and experimentation analysis of the proposed method. Finally, in Section VI conclusions are given.

II. RELATED WORK

The following sections explain the survey of various papers regarding this concern. Different methods that have been proposed for having data backup for Cloud Computing are given bellow.

1. Seed Block Algorithm

In [1], Ms. Kruti Sharma has proposed a Seed Block Algorithm Architecture (SBA) and suggested a remote backup server. The remote Backup server is a replica of original cloud server which is physically situated at a remote location. This method is based on the concept of Exclusive-OR (XOR) operation of digital computing. The whole mechanism consists of three main parts 1.The Main Cloud Server 2.Clients of the Cloud and 3.The Remote Server. The SBA uses a random number and a unique client id associated with each client.

Whenever a new Client is get registered with the cloud its unique client id is get XOR with a random number. The result of this XOR operation is called as a Seed Block which will be used only for that particular client. Whenever a client stores any Data on to the Cloud it is saved in Cloud and at the same time it is XORed with its Seed Block and the resultant Data' is stored in the remote server. If any accidental data loss occurs in the main Cloud then in such cases the original data is recovered by XORing the Data' with the Seed Block of that particular client to obtain Data'' i.e. the original Data file.

This technique is fully capable of recovering the data files accurately in any data loss situation also at the same time it maintains data integrity. The diss-advantage of this technique is that it is inefficient because the data files on the remote server uses the same space as in the main Cloud so in this way there is wastage of storage space. The storage space in the remote Server can be reduced by applying the compression techniques to achieve high efficiency.

2. Parity Cloud Service

In [2], Chi-won Song, Sungmin Park, Dong-wook Kim, Sooyong Kang, have proposed a novel data recovery service framework for cloud infrastructure, the Parity Cloud Service (PCS) provides a privacy-protected personal data recovery service. In this proposed framework user data is not required to be uploaded on to the server for data recovery. All the necessary server-side resources that provide the recovery services are within a reasonable bound. The advantages of Parity Cloud Service are that it provides a reliable data recovery at a low cost but the disadvantage is that its implementation complexity is higher.

3. Backup for cloud and Disaster Recovery for Consumers and SMBs

In [3], Vijaykumar Javaraiah introduced a mechanism for online data backup technique for cloud along with disaster recovery. In this approach the cost of having the backup for Cloud platform has been reduced and also it protects data from disaster at the same time the process of migration from one cloud service provider to another becomes easier and much simpler. In this approach the consumers' are not dependent on the service provider and it also eliminates the associated data recovery cost. A simple hardware box is used that achieves all these at little cost.

4. High Distribution and Rake Technology

In [4], Yoichiro Ueno, Noriharu Miyaho, Shuichi Suzuki,Muzai Gakuendai, Inzai-shi, Chiba,Kazuo Ichihara, proposed the innovative file back-up concept HS-DRT, that makes use of an effective ultra-widely distributed data transfer mechanism and a high-speed encryption technology. This system consists of two sequences one is Backup sequence and other is Recovery sequence. The data to be backed-up is received In Backup sequence. The recovery sequence is used when there is a disaster or any data loss occurs the Supervisory Server (one of the components of the HSDRT) starts the recovery sequence. There are some limitations in this approach and due to which, this model cannot be declared as a perfect technique for Cloud back-up and recovery. Although this model can be used for movable clients such as laptops Smart phones etc. the data recovery cost is comparatively increased and also there is increased redundancy.

5. **Efficient Routing Grounded on Taxonomy**

In [5], Giuseppe Pirr´o, Paolo Trunfio, Domenico Talia, Paolo Missier and Carole Goble proposed Efficient Routing Grounded on Taxonomy (ERGOT) which is fully based on the semantic analysis and does not focuses on time and implementation complexity. This system is based on the Semantics that provide support for Service Discovery in cloud computing. This model is built upon 3 components one A DHT (Distributed Hash Table) protocol second A SON (Semantic Overlay Network), and third A measure of semantic similarity among service description We makes a focus on this technique because it is not a simple back-up technique rather it provides retrieval of data in an efficient way that is totally based on the semantic similarity between service descriptions and service requests. ERGOT proposes a semantic-driven query answering in DHT-based systems by building a SON over a DHT but it does not go well with semantic similarity search models. The drawback of this model is an increased time complexity and implementation complexity.

6. **Shared Backup Router Resources**

In [6], Eleni Palkopoulouy, Dominic A. Schupke, Thomas Bauscherty, proposed one technique that mainly focuses on the significant reduction of cost and router failure scenario i.e. (SBBR). It involves logical connectivity of IP that will be remain unchanged even after a router failure. The most important factor of this model is that it provides the network management system via multi-layer signaling. Additionally this model shows how service imposed maximum outage requirements that have a direct effect on the setting of the SBRR architecture (e.g. imposing a minimum number of network-wide shared router resources locations).The problem with model is that it is unable to include optimization concept with cost reduction.

7. **Rent out the Rented Resources**

In [7], Sheheryar Malik, Fabrice Huet, proposed the lowest cost point of view a model “Rent out the Rented Resources”. This technique focuses on reducing the cloud service’s monetary cost. It proposed a model for cross cloud federation which consists of three phases that are 1) Discovery, 2) Matchmaking and 3) Authentication. This model is simply based on the concept of cloud vendors that rent the resources from different venture(s) and after virtualization, rents it to the clients as cloud services.

8. **Cold and Hot Back-up Strategy**

In [8], Lili Sun, Jianwei An, Yang Yang, Ming Zeng, suggested a technique in which there is a gradual increase in cost with the increase in data i.e. The Cold and Hot back-up strategy that performs backup and recovery on trigger basis of failure detection. In CBSRS (i.e. Cold Backup Service Replacement Strategy) recovery process, it is triggered when a service failure is detected and it will not be triggered when there is no failure i.e. when the service is available. The HBSRS (i.e. Hot Backup Service Replacement Strategy), is a transcendental recovery strategy for service composition that is used for dynamic network. During the implementation of process, the backup services remains in the activated state and the first returned results of services will be used to ensure the successful implementation of service composition.

The advantages and disadvantages of all the above discussed techniques are described in the Table-I.

Sr.no.	Approach	Advantage	Disadvantage
1	SBA[1]	Simple to implement	inefficient
2	Parity Cloud Service[2]	Reliable Privacy Low cost	High complexity
3	LINUX BOX[3]	Simple Low cost	High bandwidth, Complete server backup at a time
4	HSDRT[4]	Used for movable clients	Costly, Increased redundancy
5	ERGOT[]	Exact match retrieval, privacy	Increased complexity
6	Cold/Hot Backup Strategy[8]	Triggered only when failure detected	Cost increases as data increases

Table-I. Comparison between various techniques of Back-up and recovery

III. PROPOSED METHOD

The Backup Server is a remote data repository situated at a remote location (i.e. far away from the main server) which is the copy of the main cloud for the backup purpose. As this Backup server is situated at remote location and having the complete state of the main cloud, then this remote location server is termed as Remote Data Backup Server. The main cloud is termed as the central repository and remote backup cloud is termed as remote repository. In case of any natural calamity such as flood, fire, earthquack, any human errors or any hardware mulfunction if the central repository is lost then the cloud data can be recovered from remote backup server. The remote backup server also facilitates the user to collect information from any remote location even if network connectivity is not available to the main cloud or if data not found on main cloud.

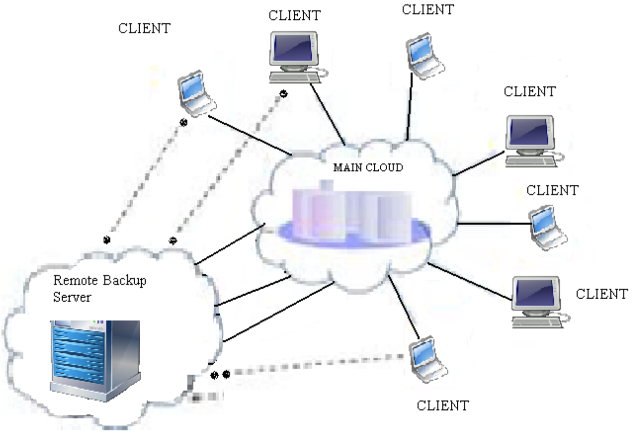

Fig.1 Remote data Backup Server and its Architecture

As shown in Fig-1 clients are allowed to access the files from remote repository if the data is not found on central repository (i.e. indirectly).

The Remote backup services should cover the following issues:

1) Data Integrity

Data integrity refers to maintaining and assuring the accuracy and consistency of data over its entire life-cycle, and is a critical aspect to the design, implementation and usage of any system which stores, processes, or retrieves data.

2) Data security

User data protection is an important concern on the remote server and either intentionally or unintentionally, it should not be accessed by third party or any other users/clients.

3) Data Confidentiality

Client's data files must be kept confidential such that if there are number of users simultaneously accessing the cloud, then data files that are personal to only a particular client must be kept hidden from other clients on the cloud during file access.

4) Trustworthiness

The remote cloud must possess the Trustworthiness characteristic. Because the user/client stores their private data; therefore the cloud and remote backup cloud must play a trustworthy role.

5) Cost efficiency

The cost of process of data backup and recovery should be efficient so that maximum no. of company/clients can take advantage of back-up and recovery service. There are many techniques that have focused on these issues. In forthcoming section, we will be discussing a technique of back-up and recovery in cloud computing domain that will cover the foresaid issues.

IV. DESIGN OF THE PROPOSED METHOD

In this method we basically use the concept of Exclusive OR (XOR) operation of the digital computing world. For ex: - Suppose we have two data files: A and B. When we XOR A and B it produces X i.e. $X = A\oplus B$. Now if suppose A data file gets destroyed due to any reason and we want our A data file back then we can get our A data file back with the help of B and X data file i.e. $A = X\oplus\mathbf{B}$. Here, first we set a random number in the cloud and every client will be given a unique client id. Second, whenever the client is being registered in the main cloud; then client id of that client and a random number is EXORed ( ) with each other to generate seed block for that particular client. The generated seed block corresponding to each client is stored at remote server. Whenever client stores a file in the cloud first time it is stored in main cloud server and at the same time the main file of the client is EXORed with the Seed Block of that particular client. And the resultant EXORed file is compressed and then the resultant compressed file is stored at the remote server in the form of file' (pronounced as File dash). As we are compressing the backup file on the backup server the memory requirement of the backup server becomes very less than that of the main cloud. This reduces the cost of backup recovery system and makes the process very efficient. If the main cloud is crashed / damaged or file has been deleted mistakenly, then to get the original file back, the compressed file on the remote backup server is decompressed and then the decompressed file is EXORed with the seed block of the corresponding client to produce the original file. The resulted file i.e. original file will be sent back to the requested client. The architecture representation of the Seed Block Algorithm is shown in the Fig.2.

Proposed Algorithm

```
Algorithm 1:
Initialization: Main Cloud: $M_c$; Remote Server: $R_s$;
Clients of Main Cloud: $C_i$; Files: $a_i$ and $a'_{i;}$
Compressed File: $a_{c;}$
Seed block: $S_i$; Random Number: $r$;
Client's ID: $Client\_Id_i$
Input: $a_1$ created by $C_1$; is generated at $M_c$;
Output: Recovered file $a_1$ after deletion at $M_c$
Given: Authenticated clients could allow uploading,
downloading and do modification on its own files only.
Step 1: Generate a random number.
            Int $r = rand\ ( );$
Step 2: Create a seed Block $S_i$ for each $C_i$ and Store
$S_i$ at $R_s$.
$S_i = r \oplus Client\_Id_{i;}$ (Repeat step 2 for all clients)

Step 3: If $C_i$ creates/modifies $a_1$ and stores at
$M_c$, then $a'_1$ create as
            $a'_1 = a_1 \oplus S_i$
Step 4: Compress $a'_1$ as:
            $a_c = compress(a'_1);$
    1. Read $a'_1$ per byte.
    2. Determine read byte as nonzero or zero byte.
    3. Write nonzero byte into data I and write bit '1' into temporary byte data, or only write bit '0' into
       temporary byte data for zero input byte.
    4. Repeat step 1-3 until temporary byte data filled with 8 bits of data.
    5. If temporary byte data filled with 8 bit then write the byte value of temporary byte data into data II.
    6. Clear temporary byte data.
    7. Repeat step 1-6 until end of file is reach.
    8. Write combined output data $a_c$ as
        a) Write original input length.
        b) Write data I.
        c) Write data II.
Step 5: Store $a_c$ at $R_s$
Step 6: If server crashes and $a_1$ deleted from $M_c$
Then, we do reverse process to retrieve the original as:
            $a'_1 = deCompress(a_c);$

    1. Read original input length.
    2. If was compressed separately, decompress data I and data II (optional).
    3. Read data II per bit.
    4. Determine whether read bit is '0' or '1'.
    5. Write to output, if read bit is '1' then read and write data I to output, if read bit is '0' then write zero byte to
       output.
    6. Repeat step 2-5 until original input length is reach.
Step 7: $a_1 = a'_1 \oplus S_{i;}$
        Return $a_1$ to $C_i$.
Step 8: END.
```

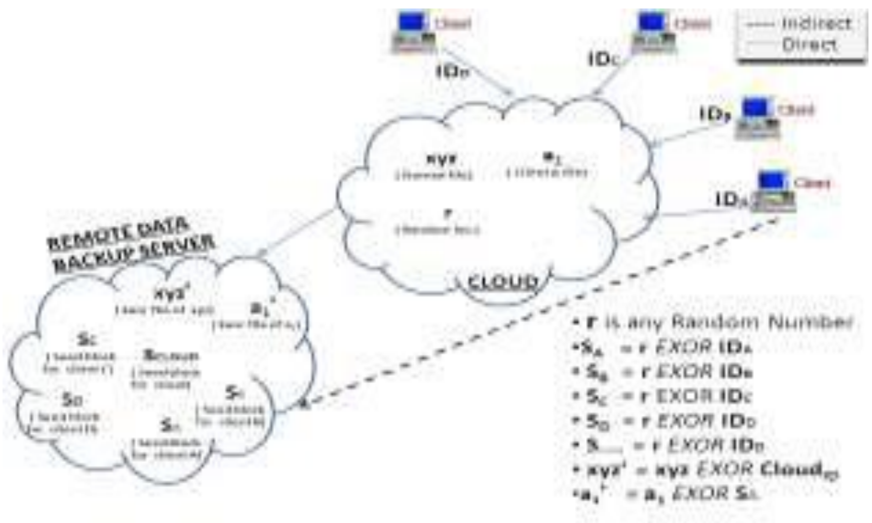

Fig. 2 Seed Block Algorithm and its Architecture

V. Experimentation And Result Analysis

In this section, we discuss the experimentation and result analysis of the proposed algorithm. For experimentation we focused on different minimal system requirement for main cloud's server and remote server as depicted in Table-II. From Table-II, we see that memory requirement is 12GB and 8GB for the main cloud's server and remote server respectively; this can be extended as per the necessity. From Table-II, it is observed that memory requirement is less in remote server as compare to the main cloud's server because files will be stored on remote server after compression this will lead to reduced memory requirement for the backup server.

Table-II System Environment

	Main Cloud's Server	Remote Cloud Server
CPU	Core2 Quad Q6600 2.4GHz	Core2 Quad Q6600 2.4GHz
MEMORY	12 GB (DDR2-800)	8 GB (DDR2-800)
OS	Any Windows/Linux Platform	Any Windows/Linux Platform
HDD	SATA 500 GB or more (7200 rpm)	SATA250 GB or more (7200 rpm)

During experimentation, we found that size of original data file stored at main cloud is large as compared to the size of Back-up file stored at Remote Server as depicted in Table- III. Even though the size of backup file is less than the size of file on main cloud still after recovery process the resultant file will be exactly same as the original one. In order to make this fact plausible, we perform this experiment for different types of files. Results tabulated in Table-III. This experiment shows that proposed technique is very much robust in maintaining the size of recovery file same as that the original data file. From this we conclude that proposed Backup technique recovers the data file without any data loss.

Table-III: Performance analysis for different types of files

Type	Size of Original file on main cloud	Size of Backup file on remote server	Size of recovered file
.txt	21 KB	5 KB	21 KB
.doc	1483 KB	874 KB	1483 KB
.pdf	1122 KB	764 KB	1122 KB
.jpg	25 KB	21 KB	25 KB
.xl	23 KB	3 KB	23 KB

Processing Time means time taken by the process when client uploads a file at main cloud and that includes the assembling of data such as the random number from main cloud, seed block of the corresponding client from the remote server for EXORing operation; after assembling, performing the EXORed operation of the contents of the uploaded file with the seed block and finally stored the EXORed file onto the remote server. Performance of this experiment is tabulated in Table-IV. We also observed that as data size increases, the processing time increases. On other hand, we also found that performance which is megabyte per sec (MB/sec) being constant at some level even if the data size increases as shown in Table-IV.

Table-IV Effect of data size on processing time

Practical Data Size	Processing Time on Main Cloud Time(in sec) (Approx.)	Processing Time on Remote Cloud Time(in sec) (Approx.)	Performance (MB/sec)
1KB	6.76	2	150
64KB	12.8	3	160
2MB	3600	5	164
32 MB	8400	8	250
1GB	16200	15	280
4GB	32100	35	280
6GB	52000	45	280

The Fig-3 shows the experimentation result of proposed algorithm. As fig-3 (a) shows the original file which is uploaded by the client on main cloud. Fig-3 (b) shows the EXORed file which is stored on the remote server. This file contains the secured EXORed content of original file and seed block content of the corresponding client. Fig-3 (c) shows the recovered file; which indirectly sent to client in the absence of network connectivity and in case of the file deletion or if the cloud gets destroyed due to any reason.

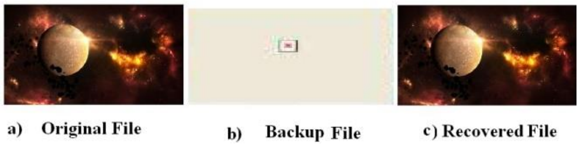
a) Original File b) Backup File c) Recovered File

Fig.3 Sample output image of SBA Algorithm

VI. CONCLUSION

In this paper, we presented detail design of proposed Backup recovery technique for cloud computing. Proposed technique is robust in helping the users to collect information from any remote location in the absence of network connectivity and also to recover the files in case of the file deletion or if the cloud gets destroyed due to any reason also it reduces the memory requirement of backup cloud to a lower value as compared to main cloud, this achieves higher efficiency as compared to existing backup technologies for cloud computing. Experimentation and result analysis shows that proposed technique also focuses on the security concept for the back-up files stored at remote server, without using any of the existing encryption techniques. The time related issues are also being solved by proposed techniques such that it will take minimum time for the recovery process.

REFERENCES

[1] Ms. Kruti Sharma, Prof. Kavita R Singh, “Seed Block Algorithm: A Remote Smart Data Back-up Technique for Cloud Computing” Intrnational Conference on Communication Systems and Network Technologies IEEE 2013.

[2] Chi-won Song, Sungmin Park, Dong-wook Kim, Sooyong Kang, “Parity Cloud Service: A Privacy-Protected Personal Data Recovery Service,” International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-11 2011.

[3] Vijaykumar Javaraiah, Brocade Advanced Networks and Telecommunication systems (ANTS), 2011,"Backupforcloud and Disaster Recovery for Consumers and SMBs," IEEE 5th International Conference, 2011.

[4] Yoichiro Ueno, Noriharu Miyaho, Shuichi Suzuki,Muzai Gakuendai, Inzai-shi, Chiba,Kazuo Ichihara, "Performance Evaluation of a Disaster Recovery System and Practical Network System Applications," Fifth International Conference on Systems and Networks Communications, pp 256-259 2010.

[5] Giuseppe Pirr´o, Paolo Trunfio , Domenico Talia, Paolo Missier and Carole Goble, "ERGOT: A Semantic-based System for Service Discovery in Distributed Infrastructures," 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing 2010.

[6] Eleni Palkopoulouy, Dominic A. Schupke, Thomas Bauscherty, "Recovery Time Analysis for the Shared Backup Router Resources (SBRR) Architecture", EEE ICC 2011.

[7] Sheheryar Malik, Fabrice Huet, December "Virtual Cloud: Rent Out the Rented Resources," 6th International Conference on Internet Technology and Secure Transactions,11-14 ,Abu Dhabi, United Arab Emirates 2011.

[8] Lili Sun, Jianwei An, Yang Yang, Ming Zeng, "Recovery Strategies for Service Composition in Dynamic Network," International Conference on Cloud and Service Computing 2011.

[9] Y.Ueno, N.Miyaho, and S.Suzuki, , 2009, "Disaster Recovery Mechanism using Widely Distributed Networking and Secure Metadata Handling Technology", Proceedings of the 4th edition of the UPGRADE-CN workshop, pp. 45-48.

[10] Xi Zhou, Junshuai Shi, Yingxiao Xu, Yinsheng Li and Weiwei Sun, "A backup restoration algorithm of service composition in MANETs," Communication Technology ICCT 11th IEEE International Conference, pp. 588-591 2008.

[11] M. Armbrust et al, "Above the clouds: A berkeley view of cloud computing," http://www.eecs.berkeley.edu/Pubs/TechRpts/2009//EEC
S-2009-28.pdf.

[12] F.BKashani, C.Chen,C.Shahabi.WSPDS, "Web Services Peer to Peer Discovery Service ," ICOMP 2004.

[13] P.Demeester et al., 1999, "Resilience in Multilayer Networks," IEEE Communications Magazine, Vol. 37, No. 8, p.70-76.

[14] S. Zhang, X. Chen, and X. Huo, "Cloud Computing Research and Development Trend," IEEE Second International Conference on Future Networks, pp. 93-97 2010.

[15] T. M. Coughlin and S. L. Linfoot, "A Novel Taxonomy for Consumer Metadata," IEEE ICCE Conference 2010.

[16] K. Keahey, M. Tsugawa, A. Matsunaga, J. Fortes, "Sky Computing", IEEE Journal of Internet Computing, vol. 13, pp. 43-51 2009.

[17] M. D. Assuncao, A.Costanzo and R. Buyya, "Evaluating the Cost- Benefit of Using Cloud Computing to Extend the Capacity of Clusters," Proceedings of the 18th International Symposium on High Performance Distributed Computing (HPDC 2009), Germany 2009.

[18] Wayne A. Jansen, "Cloud Hooks: Security and Privacy Issues in Cloud Computing, 44th Hawaii International Conference on System Sciences.Hawaii 2011.

[19] Jinpeng et al, "Managing Security of Virtual Machine Images in a Cloud Environment", CCSW, Chicago, USA 2009.

[20] Ms..Kruti Sharma,Prof K.R.Singh, "Online data Backup And Disaster Recovery techniques in cloud computing:A review", IJEIT, Vol.2, Issue 5 2012.

[21] I Made Agus Dwi Suarjaya, "A New Algorithm for Data Compression Optimization", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No.8, 2012.