ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

AN IMPROVED SECURITY OF CLOUD STORED BIG DATA BY USING THE IDENTITY BASED HYBRID ENCRYPTION SCHEMES FOR SMART GRIDS

B.Tamilselvi, PG Scholar, Department of CSE, RVS College of Engineering and Technology, Coimbatore.
Arulprakash. P, [2]Assistant Professor, Department of CSE, R.V.S. College of Engineering and Technology, Coimbatore.
P.Vidhya, PG Scholar, Department of CSE, RVS College of Engineering and Technology, Coimbatore.

E-mail: [1]tamil.balajothi@gmail.com, [2]arulprakash247@gmail.com, [3]vidhya.12it@gmail.com

Abstract---Cloud computing provides a way of mean to handle the large volume of data in the efficient manner. As the number of data increased in number on the smart grid, the data handling becomes the biggest issue. In the existing work, it is overcome by concentrating on the identity based encryption which is used to protect the data from the cloud server and as well as limit the data access in the secured manner for the corresponding users. However this work may lack from security in case of hacker who knows the identity information of the users and might corrupt the data's is that are stored in the cloud. This problem is overcome in the proposed methodology by introducing the identity based hybrid encryption. This approach would encapsulate data contents before encryption using identity information which will leads to a secured environment. The experimental results conducted were proves that the proposed methodology provides better result than the existing approach in terms of improved security level.

Keywords--- Hadoop Distributed File System,Big data, cloud computing, secure, information management, smart grid.

I. INTRODUCTION

1.1 INTRODUCTION TO BIG DATA

Big data is data that exceeds the processing capacity of conventional database systems. The data is too big, moves too fast, or doesn't fit the strictures of your database architectures. To gain value from this data, you must choose an alternative way to process it.

The hot IT buzzword of 2012, big data has become viable as cost-effective approaches have emerged to tame the volume, velocity and variability of massive data. Within this data lie valuable patterns and information, previously hidden because of the amount of work required to extract them. To leading corporations, such as Walmart or Google, this power has been in reach for some time, but at fantastic cost. Today's commodity hardware, cloud architectures and open source software bring big data processing into the reach of the less well-resourced. Big data processing is eminently feasible for even the small garage start-ups, who can cheaply rent server time in the cloud.

The value of big data to an organization falls into two categories: analytical use, and enabling new products. Big data analytics can reveal insights hidden previously by data too costly to process, such as peer influence among customers, revealed by analyzing shoppers' transactions, social and geographical data. Being able to process every item of data in reasonable time removes the troublesome need for sampling and promotes an investigative approach to data, in contrast to the somewhat static nature of running predetermined reports.

The past decade's successful web start ups are prime examples of big data used as an enabler of new products and services. For example, by combining a large number of signals from a user's actions and those of their friends, Facebook has been able to craft a highly personalized user experience and create a new kind of advertising business. It's no coincidence that the lion's shares of ideas and tools underpinning big data have emerged from Google, Yahoo, Amazon and Facebook.

The emergence of big data into the enterprise brings with it a necessary counterpart: agility. Successfully exploiting the value in big data requires experimentation and exploration. Whether creating new products or looking for ways to gain competitive advantage, the job calls for curiosity and an entrepreneurial outlook.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

1.2 HADOOP IN BIG DATA

Hadoop has become a central platform to store big data through its Hadoop Distributed File System (HDFS) as well as to run analytics on this stored big data using its MapReduce component.Many of us would have certainly heard about Big Data, Hadoop and analytics. The industry is now focused primarily on them and Gartner identifies strategic big data and actionable analytics as being among the Top 10 strategic technology trends of 2013.

According to the Gartner website: 'Big Data is moving from a focus on individual projects to an influence on enterprises strategic information architecture. Dealing with data volume, variety, velocity and complexity is forcing changes to many traditional approaches. This realisation is leading organisations to abandon the concept of a single enterprise data warehouse containing all information needed for decisions. Instead, they are moving towards multiple systems, including content management, data warehouses, data marts and specialised file systems tied together with data services and metadata, which will become the logical enterprise data warehouse.

There are various systems available for big data processing and analytics, alternatives to Hadoop such as HPCC or the newly launched Red Shift by Amazon. However, the success of Hadoop can be gauged by the number of Hadoop distributions available from different technological companies such as IBM Info Sphere Big Insights, Microsoft HD Insight Service on Azure, ClouderaHadoop, Yahoo's distribution of Hadoop, and many more. There are basically four reasons behind its success:

- It's an open source project.
- It can be used in numerous domains.
- It has a lot of scope for improvement with respect to fault tolerance, availability and file systems.

One can write Hadoop jobs in SQL like Hive, Pig, Jaql, etc, instead of using the complex MapReduce.This enables companies to modify the Hadoop core or any of its distributions to adapt to the company's own requirements and the project's requirements. Here the focus is mainly on the basics of Hadoop. However, in forthcoming series, the primarily focus on fault tolerance and the availability features of Hadoop.

Formally, Hadoop is an open source, large scale, batch data processing, distributed computing framework for big data storage and analytics. It facilitates scalability and takes care of detecting and handling failures. Hadoop ensures high availability of data by creating multiple copies of the data in different nodes throughout the cluster. By default, the replication factor is set to 3. In Hadoop, the code is moved to the location of the data instead of moving the data towards the code. In the rest of this article, "whenever I mention Hadoop, I refer to the Hadoop Core package available from http://hadoop.apache.org". There are five major components of Hadoop:

- MapReduce (a job tracker and task tracker)
- NameNode and Secondary NameNode
- DataNode (that runs on a slave)
- Job Tracker (runs on a master)
- Task Tracker (runs on a slave)

1.2.1 MapReduce

The MapReduce framework has been introduced by Google. According to a definition in a Google paper on MapReduce, MapReduce is "A simple and powerful interface that enables the automatic parallelisation and distribution of large-scale computations, combined with an implementation of this interface that achieves high performance on large clusters of commodity PCs.

It has basically two components: Map and Reduce. The MapReduce component is used for data analytics programming. It completely hides the details of the system from the user.

1.2.2 HDFS

Hadoop has its own implementation of distributed file systems called Hadoop Distributed File System. It provides a set of commands just like the UNIX file and directory manipulation. One can also mount HDFS as fuse-dfs and use all the UNIX commands. The data block is generally 128 MB; hence, a 300 MB file will be split into 2 x 128 MB and 1 x 44 MB. All these split blocks will be

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

copied 'N' times over clusters. N is the replication factor and is generally set to 3.

1.2.3 Name Node

NameNode contains information regarding the block's location as well as the information of the entire directory structure and files. It is a single point of failure in the cluster, i.e., if NameNode goes down, the whole file system goes down. Hadoop therefore also contains a secondary NameNode which contains an edit log, which in case of the failure of NameNode, can be used to replay all the actions of the file system and thus restore the state of the file system. A secondary NameNode regularly creates checkpoint images in the form of the edit log of NameNode.

1.2.4 DataNode

DataNode runs on all the slave machines and actually stores all the data of the cluster. DataNode periodically reports to NameNode with the list of blocks stored. Job Tracker and Task Tracker Job Tracker runs on the master node and Task Tracker runs on slave nodes. Each Task Tracker has multiple task-instances running, and every Task Tracker reports to Job Tracker in the form of a heart beat at regular a interval, which also carries details of the current job it is executing and is idle if it has finished executing. Job Tracker schedules jobs and takes care of failed ones by re-executing them on some other nodes. Job Tracker is currently a single point of failure in the Hadoop Cluster.

II. LITERATURE REVIEW

1."PRIVACY PRESERVING ACCESS CONTROL WITH AUTHENTICATION FOR SECURING DATA IN CLOUDS"-2012- SUSHMITA RUJ, MILOS STOJMENOVIC, AMIYA NAYAK - IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)

Research in cloud computing is receiving a lot of attention from both academic and industrial worlds. In cloud computing, users can outsource their computation and storage to servers (also called clouds) using Internet. This frees users from the hassles of maintaining resources on-site. Clouds can provide several types of services like applications (e.g., Google Apps, Microsoft online), infrastructures (e.g., Amazon's EC2, Eucalyptus, Nimbus), and platforms to help developers write applications (e.g., Amazon's S3, Windows Azure).In this work, we propose a new privacy preserving authenticated access control scheme for securing data in clouds. In the proposed scheme, the cloud verifies the authenticity of the user without knowing the user's identity before storing information. Our scheme also has the added feature of access control in which only valid users are able to decrypt the stored information.Access policies can be in any of the following formats: 1) Boolean functions of attributes, 2) Linear Secret Sharing Scheme (LSSS) matrix, or 3) Monotone span programs. Any access structure can be converted into a Boolean function. Our scheme not only provides fine-grained access control but also authenticates users who store information in the cloud. The cloud however does not know the identity of the user who stores information, but only verifies the user's credentials. Key distribution is done in a decentralized way. One limitation is that the cloud knows the access policy for each record stored in the cloud.

2. "TOWARD SECURE AND DEPENDABLE STORAGE SERVICES IN CLOUD COMPUTING" -2012- CONG WANG,QIAN WANG,KUI REN,NING CAO, AND WENJING LOU -IEEE Transactions on Services Computing, vol.5

Several trends are opening up the era of cloud computing, which is an Internet-based development and use of computer technology. The ever cheaper and more powerful processors, together with the Software as a Service (SaaS) computing architecture, are transforming data centers into pools of computing service on a huge scale. The increasing network bandwidth and reliable yet flexible network connections make it even possible that users can now subscribe high quality services from data and software that reside solely on remote data centers. Though the benefits are clear, such a service is also relinquishing users' physical possession of their outsourced data, which inevitably poses new security risks toward the correctness of the data in cloud. In order to address this new problem and further achieve a secure and dependable cloud storage service, we propose in this paper a flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data. The proposed design allows users to audit the cloud storage with very lightweight communication and computation cost. The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast data error

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

localization, i.e., the identification of misbehaving server. Error localization is a key prerequisite for eliminating errors in storage systems. It is also of critical importance to identify potential threats from external attacks. However, many previous schemes do not explicitly consider the problem of data error localization, thus only providing binary results for the storage verification. Our scheme outperforms those by integrating the correctness verification and error localization (misbehaving server identification) in our challenge-response protocol: the response values from servers for each challenge not only determine the correctness of the distributed storage, but also contain information to locate potential data error(s).

3. "CRYPTOGRAPHIC CLOUD STORAGE - SENY KAMARA, KRISTIN LAUTER"-2008-FC'10 Proceedings of the 14th international conference on Financial cryptograpy and data security

Advances in networking technology and an increase in the need for computing resources have prompted many organizations to outsource their storage and computing needs. This new economic and computing model is commonly referred to as cloud computing and includes various types of services such as: infrastructure as a service (IaaS), where a customer makes use of a service provider's computing, storage or networking infrastructure; platform as a service (PaaS), where a customer leverages the provider's resources to run custom applications; and finally software as a service (SaaS), where customers use software that is run on the providers infrastructure.

Cloud infrastructures can be roughly categorized as either private or public. In a private cloud, the infrastructure is managed and owned by the customer and located on-premise (i.e., in the customers region of control). In particular, this means that access to customer data is under its control and is only granted to parties it trusts. In a public cloud the infrastructure is owned and managed by a cloud service provider and is located o_-premise (i.e., in the service provider's region of control). This means that customer data is outside its control and could potentially be granted to untrusted parties. It describes, at a high level, several architectures that combine recent and non-standard cryptographic primitives in order to achieve our goal.

To use the service, MegaCorp deploys dedicated machines within its network. Depending on the particular scenario, these dedicated machines will run various core components. Since these components make use of a master secret key, it is important that they be adequately protected and, in particular, that the master key be kept secret from the cloud storage provider and PartnerCorp. If this is too costly in terms of resources or expertise, management of the dedicated machines (or specific components) can alternatively be outsourced to a trusted entity.

A. EXISTING SYSTEM

The main idea of existing system, security solution for the Smart-Frame is to allow all the involved entities, i.e., top and regional cloud computing centers and end-users to be represented by their identities which can be used as encryption keys or signature verification keys. The entities in the lower level can use the identities of higher-level entities to encrypt their data for secure communication with the entities in the higher level. For example, the regional centers use the top cloud's entity to encrypt their messages. By employing an identity-based re-encryption scheme, the information storages, which are components of regional clouds, can re-encrypt the received confidential data from the end-user devices so that services requested by the end-users decrypt and process the confidential data without compromising the information storages' private keys. One of the obvious benefits we can gain from applying identity-based cryptography to the Smart-Frame is that through using identities rather than digital certificates which depend on traditional public key infrastructure (PKI), we can save significant amount of resources for computation and communications and resolve scalability issues. The saving gained from the elimination of digital certificate in the big data environment is especially momentous.

B. PROPOSED SYSTEM

In this proposed system, the concept of encryption key encapsulation mechanism is extended to the identity based setting. This approach would encapsulate data contents before encryption using identity information which will leads to a secured environment. We show that an identity-based encryption scheme can be constructed by combining an identity-based encryption key encapsulation mechanism with a data encapsulation mechanism.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

A DEM is a symmetric encryption scheme which consists of the following two algorithms.

· Enc : is a deterministic encryption algorithm which takes as input 1k, a key K and a message m ∈ {0, 1}∗, and outputs a cipher text c ∈ {0, 1}∗, where K ∈ KDEM is a key in the given key space, and m is a bit string of arbitrary length. We denote this as c ←Enc(K,m).

· Dec : is a deterministic decryption algorithm which takes as input a key K and a cipher text c, and outputs the message m ∈ {0, 1}∗or a symbol ⊥to indicate that the cipher text is invalid.

For the purposes of this work, it is only required that a DEM is secure with respect to indistinguishability against passive attackers (IND-PA). Formally, this security notion is captured by the following game played between a PPT adversary A and a challenger C.

·Initial: A runs on input 1k and submits two equal length messages, m0 and m1.

·Challenge: C chooses a random key K ∈ KDEM as well as a random bit $\lambda \in \{0, 1\}$, and sends c∗← Enc (K, m λ) to A as a challenge cipher text.

·Guess: The adversary A produces a bit λ^* and wins the game if $\lambda^* = \lambda$.

This approach leads to a secured framework for securely handling the large volume of data's that are present in the network environment. This approach provides a flexible environment for the users who can store their data's into the cloud environment with assured data integrity.

III. ARCHITECTURE DIAGRAM

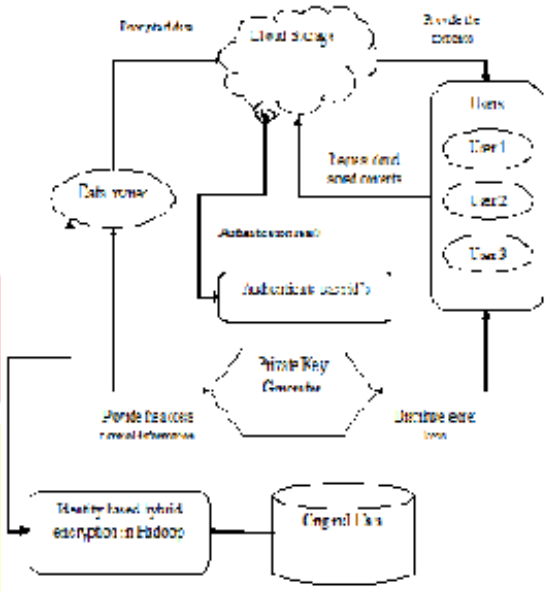

Figure 3.1 Architecture Diagram

IV. MODULE DESCRIPTION

- Network Setup: In this module, the Hadoop setup is done. The location data's will be gathered from the various environments with their description. The Hadoop environment to process those big data will be setup in this module. After creation of the environment of the Hadoop which consists of the data node and name node, the task will be partitioned and distributed to the different data nodes using the map reduce technology. The name node in the Hadoop environment is responsible for the collection of data from the multiple sites and partitioning them for distributing to the different data nodes. The data node is responsible for the processing and resulting an expected output
- Key Generation:
  - Setup: Given a security parameter λ, the PKGgenerates a secret master key mk and a set ofparameters params. The PKG distributes paramsto all the clouds and end-users.
  - ExtractTCKey: Upon receiving a top cloud'sidentity TC, the PKG generates a private key KTCassociated with TC by running the privatekey extraction algorithm Extract providing TCas input. We denote this process by KTC←ExtractRCKey (params,mk, TC).

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

- ExtractISKey: Upon receiving an identity ofthe information storage in the regional cloud,denoted by IS, the PKG generates a private keyK_{IS} associated with IS by running the privatekey extraction algorithm Extract providing ISas input. We denote this process by K_{IS}←ExtractISKey (params,mk, IS).
- ExtractServiceKey:Upon receiving an identity ofthe service A in the regional cloud, denoted bySerA, the PKG generates a private key K_{SerA} associatedwith SerA by running the private keyextraction algorithm Extract providing SerAas input. We denote this process by K_{SerA}←ExtractServiceKey (params,mk,SerA).
- ExtractEUKey:Upon receiving an end-user'sidentity EU, the PKG generates a private key K_{EU}associated with EU by running the private keyextraction algorithm Extract providing EU asinput. We denote this process by K_{EU}←ExtractEUKey (params,mk, EU).
- Encrypt2TC:Each information storage in the regional cloud can encrypt a message M into a ciphertext C_{TC} by running the IBE encryption algorithm Encrypt with params and the top cloud's identity TC. We denote this process by C_{TC}← Encrypt2TC (params, TC, M).
- DecryptTC: The top cloud can decrypt a received ciphertext C to M by running the IBE decryption algorithm Decrypt with the private key K_{TC} associated with the top cloud's identity TC. We denote this process by M ←DecryptTC (params, K_{TC}, C_{TC}).

- Proxy Re-Encryption By Information Storage:
  - RKGen: Providing its own private key KIS, its identity IS and the server A's identity SerA as input, the information storage in the regional cloud generates a re-encryption key RKIS →SerA. We denote this process by RKIS →SerA←RKGen (KIS, IS, SerA).
  - Reencrypt: The information storage in the regional cloud re-encrypts the ciphertext CIS using the re-encryption key RKIS →SerA and obtains a ciphertextCSerA. We denote this process by CSerA←Reencrypt (RKIS →SerA, CIS).
  - DecryptService: The service A decrypts CSerA using its private key KSerA. We denote this by M ←DecryptService (KSerA, CSerA).
- Signature Generation :

  Signature Generation by End-Users
  - SignEU:Each end-user can generate a signature σ for a message M using the private key KEU associated with its identity EU. We denote this process by σ ←SignEU (params, KEU, M).
  - VerifyEU: Any party can verify a signature σ for some message M using params and the identityof the end-user, EU. We denote this processby d ←VerifyEU (params, EU,σ, M), where d iseither "accept" or "reject".

  Signature Generation by Entities in Regional Cloud:
  - SignIS:Each information storage in the regionalcloud can generate a signature s for a messageM using the private key KIS associated with itsidentity IS. We denote this process by σ←SignIS (params, KIS, M). Each service in theregional cloud (denoted by SerA as a representative)can also generate a signature in thesame way.
  - VerifyIS: Any party can verify a signature σ for some message M using params and the information storage's identity IS. We denote this process by d ←VerifyIS (params, IS, σ , M), where d is either "accept" or "reject". The signatures generated by a service in the regional cloud (denoted by SerA as a representative) can be verified in the same way.

  Signature Generation by Top Cloud:
  - SignTC: The top cloud can generate a signature σ for a message M using the private key KTC associated with its identity TC. We denote this process by s ←SignTC (params, KTC, M).
  - VerifyTC: Any party can verify a signature σ for some message M using params and the identity of the top cloud, TC. We denote this process by d ←VerifyTC (params, TC, σ, M), where d is either "accept" or "reject".

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

V. IMPLEMENTATION

The below diagrams showsencryption and decryption results.The user encrypt the contents using their private key generated by private key generater. The top cloud stores the content of user and their personal information. User is authenticated by their signature in top cloud. After verifying the certificate the contents of user is send to regional cloud and it gives permission to user to decrypt the contents stored in cloud.

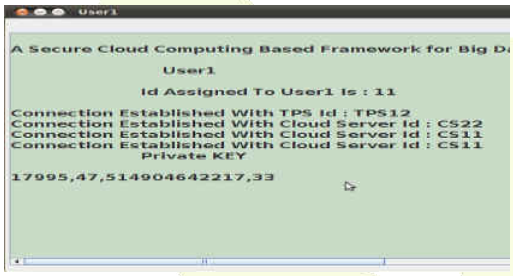

Figure 5.1Private key Generation

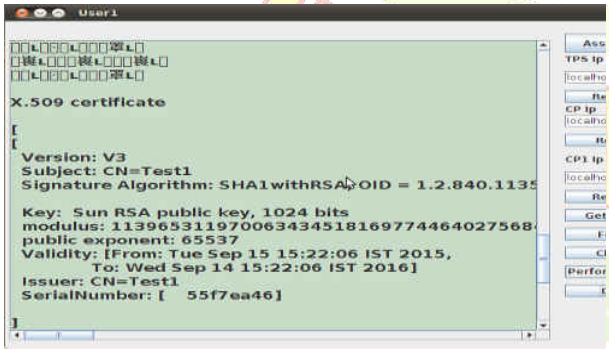

Figure 5.2Decrpting Contents

VI. CONCLUSION

In this work, Smart-Frame, a general framework for big data information management in smart grids based on cloud computing technology has been introduced. Our basic idea is to set up cloud computing centers at three hierarchical levels to manage information: top, regional, and end-user levels. While each regional cloud center is in charge of processing and managing regional data, the top cloud level provides a global view of the framework. Additionally, in order to support security for the framework, we have presented a solution based on identity-based cryptography and identity-based proxy re-encryption. As a result, our proposed framework achieves not only scalability and flexibility but also security features

VII. FUTURE WORK

In future, proxy encryption can be introduced to provide the more security for the users for ensuring the trustable environment. Data freshness can be retained by introducing some efficient approaches which can update the modified contents in the centralized server periodically Fault tolerance aware mechanism can be introduced between the cloud server, thus the data corruption while transferring down layer to top layer can be avoided successfully

REFERENCE

[1]. Joonsang Baek, Quang Hieu Vu, Joseph K. Liu, Xinyi Huang, and Yang Xiang, Senior Member, "A Secure Cloud Computing Based Framework for Big Data Information Management of Smart Grid"IEEE Transactions On Cloud Computing, Vol. 3, No. 2, April/June 2015.

[2]. Chang F, Dean J, Ghemawat S, Hsieh WC, Wallach DA, Burrws M, Chandra T, Fikes A, Gruber RE (2006) Big table: a distributed storage system for structured data. In: 7th UENIX symposium on operating systems design and implementation, pp 205–218.

[3]. Cong Wang, Qian Wang, KuiRen, Ning Cao, and Wenjing Lou, "Toward Secure and Dependable Storage Services in Cloud Computing", IEEE Transactions On Services Computing, Vol. 5, No. 2, April-June 2012.

[4]. D. Richard Kuhn, Edward J. Coyne, Timothy R. Weil, "Adding Attributes to Role-Based Access Control", IEEE Computer, vol. 43, no. 6 (June, 2010) , pp. 79-81.

[5]. Dean J, Ghemawat Dean S (2008) MapReduce: simplified data processing on large clusters. Commun ACM 51:107–113.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

[6]. Hongwei Li, Yuanshun Dai, Bo Yang, "Identity-Based Cryptography for Cloud Security", IACR Cryptology ePrint Archive, 2011.

[7]. Liu H, Orban D (2011) Cloud MapReduce: a MapReduce implementation on top of a cloud operating system. In: IEEE/ACM international symposium on cluster, cloud and grid computing, pp 464–474.

[8]. MateiZaharia, Andy Konwinski, Anthony D. Joseph, Randy Katz, Ion Stoica, "Improving MapReduce Performance in Heterogeneous Environments", 8th USENIX Symposium on Operating Systems Design and Implementation.

[9]. SenyKamara, Kristin Lauter, "Cryptographic Cloud Storage", Proceedings of Financial Cryptography: Workshop on Real-Life Cryptographic Protocols and Standardization 2010.

[10]. SushmitaRuj, Milos Stojmenovic, AmiyaNayak, "Privacy Preserving Access Control with Authentication for Securing Data in Clouds", 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing.