A HYBRID CLOUD APPROACH WITH NOVEL BASED KEY GENERATION ALGORITHM FOR SECURE DATA DEDUPLICATION

Dr. A. Kousalya **Arunjith.R**

[1]Associate Professor, Department of Computer Science and Engineering,
United Institute of Technology, Tamil Nadu, India
kousivetri@gmail.com
[2]M.E Student, Department of Computer science and Engineering,
United Institute of Technology, Tamil Nadu, India
arunjith.babloo@gmail.com

ABSTRACT:

With the main focus of research in Data deduplication technique gearing towards efficient data compression, the resulting protocols tend to be vulnerable to various security attacks. Over the years, emphasis has also been placed on improving the security of thesenetworks. Different solutions have been proposed for different types of attacks. However, these solutions often compromise efficiency. To protect the privacy of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing and also we design and implement a new system which could protect the security for predictable message. The main idea of our technique is Hybrid encryption key generation algorithm. The hash functions are used to define the tag generation and convergent key generation is done from the data content itself to check for deduplication. In Convergent Encryption technique, to support the duplicate check in the database, the key is derived from the file with the help of cryptographic hash function. To avoid the deterministic key generation, the encryption key for

file on our system will be generated with the aid of private key cloud server with privilege

Index terms-Cloud computing, Convergent Algorithm, Hybrid Encryption, De-duplication, RC5 Encryption, Virtual platform

1. INTRODUCTION

CLOUD computing has been envisioned as the next generation information technology (IT) architecture for enterprises, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk . From users' perspective, including both individuals and IT enterprises,

storing data remotely to the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with location independence, and avoidance of capital expenditure on hardware, software, and personnel maintenances, etc. While cloud computing makes these advantages more appealing than ever, it also brings new and challenging security threats toward user's outsourced data. Since cloud service providers (CSP) are separate administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons. First of all, although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity. Examples of outages and security breaches of noteworthy cloud services appear from time to time. Second, there do exist various motivations for CSP to behave unfaithfully toward the cloud users regarding their outsourced data status. For examples, CSP might reclaim storage for monetary reasons by discarding data that have not been or are rarely accessed, or even hide data loss incidents to maintain a reputation.

2. RELATED WORKS

P. Anderson and L. Zhang [6] have discussed about a typical community of laptop users, share a considerable amount of data in common. As conventional backup solutions are not well suited in storing large quantities of personal and corporate data on laptops or home computers. These often have poor or intermittent connectivity, which are vulnerable to theft or hardware failure of the system. Thus this paper describes an algorithm which takes advantage of the data which is common between users to increase the speed of data backups, and effectively reduce the storage requirements. This algorithm supports client-end per-user encryption which is necessary for a confidential personal data.

The authors of paper [2], [3] & [8] have introduced different methods and techniques in consideration with security measures as well as the approaches which can be used for improving the scalability and effectiveness. The authors Sven Bugiel, Stefan Nurnberger, Ahmad-Reza Sadeghi and Thomas Schneider of paper [8], have proposed an architecture and protocols that accumulate slow secure computations over time and provide the possibility to query them in parallel on demand by leveraging the benefits of cloud computing. Whereas the authors M. Bellare, C. Namprempre, and G. Neven of the paper [4] provides signature schemes which are defined by either

explicitly or implicitly from existing literature and also security proofs.

M. Bellare and A. Palacio of the paper[2] provides a proof for Guillou-Quisquater(GQ)based on the assumed security of RSA under one more inversion, inwhich an extension of the usual one-wayness assumption that was introduced in this paper. It also provides such a proof for the Schnorr identification schemes based on a corresponding discrete-log related assumption.

MihirBellare, Sriram Keelveedhi and Thomas Ristenpart [5] presented about a new cryptographic primitive, Message-Locked Encryption (MLE), where the key forthe encryption and decryption are derived from the message. MLE providesa way to achieve secure deduplication (space-efficient secure outsourced storage), which is a goal currently targeted by numerous cloud-storage providers. Hence providing definitions both for privacy and for a form of integrity that we call tag consistency and also provide ROM security analyses of a natural family of MLE schemes that includes deployed schemes.

To overcome problem of providing secure outsourced storage that both supports deduplication and resists brute-force attacksbecause in cloud storage service providers such as Dropbox, Mozyetc. Perform deduplication to save space by storing only one copy of each file uploaded to the database. This makes clients to conventionally encrypt their files though the savings are lost. Hence the authors M. Bellare, S. Keelveedhi, and T. Ristenpart of the paper [4] designed a system named DupLESS, which combines a CE-type baseMLE scheme with the prominant ability to obtain the message-derived keys with the help of a key server (KS) shared amongst a certain group of clients.It enables the clients to store encrypted data with an existing service which have the service to perform deduplication on their behalf, and still achieves strong confidentiality guarantees.

Efficient and scalable deduplication techniques are required to serve the need of removing duplicated data in bigdata processing platforms such as Hadoop. The technique proposed by the authors Dongzhan Zhang, ChengfaLiao, Wengjin Yang and Ran Tao in this paper[1] is a new file aggregation scheme based on MapReduce is proposed and the recent SHA-3 standard Keccak is used for hash computation. Moreover, the overall Deduplication procedure is implemented based on MapReduce and HBase so as to providespeed up the deduplication procedure and scalability to cater to the challenges brought by the Bigdata Era. Their approach was to secure data deduplication scheme with efficient PoW process for dynamic ownership management.

On storing a unique copy of the duplicate data, cloud providers finely reduce their storage as well as data transfer costs.

The advantages of data deduplication unfortunately comes with a high cost in terms of new security and privacy challenges. To make this cost effective the authors Pasquale Puzio, RefikMolva ,MelekOnen and Sergio Loureiro of the paper [7] have proposed ClouDedup, a secure and efficient storage service which assures block-level deduplication and data confidentiality at the same time. Although based on convergent encryption, they have also included another new component i.e. to implement the key management for each block together withactual deduplication operation. Hence this approach has not impacted the overall storage and computational costs.

Thoughthe data deduplication has a lot of pros in security and privacy, the major concern is theuser's sensitive data which are susceptible to both attacks insider and outsider. A convergent encryption method enforces/ensures data confidentiality while making deduplication feasible. In traditional deduplication systems based on convergent encryption did provide confidentiality but doesn't support the duplicate check operation. The authors Sharma Bharat and Mandre B.R of the paper[9] has introduced the idea of authorized data deduplication to protect data security by involving different privileges of users in the duplicate check. In this, the users with different roles and privileges are involved in duplicate check besides the data itself. Here the files are encrypted with differential privilege keys. The user can verify presence of file after deduplication in cloud with the help of a third party auditor by auditing the data. Also the auditor audits and verifies the uploaded file on time.

3. PROPOSED METHODOLOGY

We design and implement a new system which could protect the security for predictable message. The main idea of our technique is that the Novel Encryption Key Generation Algorithm, for simplicity, the hash functions is used to define the tag generation functions and convergent keys. In traditional convergent encryption, to support duplicate check, the key is derived from the file by using some cryptographic hash function. To avoid the deterministic key generation, the encryption key for file in our system will be generated with the aid of the private key cloud server with privilege**.** In dynamic environments, a new encrypted shared key has to be generated for every join/leave event and forwarded to the Key Distribution Centre (KDC) of the requester. A Hybrid Encryption Algorithm (HEA) for generating a secured (encrypted) shared key is proposed for the dynamic environments.

4. SYSTEM MODEL

The Objective of the project is eliminating duplicate copies of repeating data. Data deduplication is one of the important data compression techniques used for

eliminating duplicate copies of repeating data, and it has been widely used in cloud storage to reduce the amount of storage space and to save bandwidth. In order to protect the confidentiality of sensitive data while supporting deduplication, the convergent data encryption technique has been proposed to encrypt the user data before outsourcing. In traditional deduplication system data files are consuming so much of space in the cloud database because of deduplicated files. Also security of sensitive data was compromised.

4.1 SYSTEM DIAGRAM & DESCRIPTION

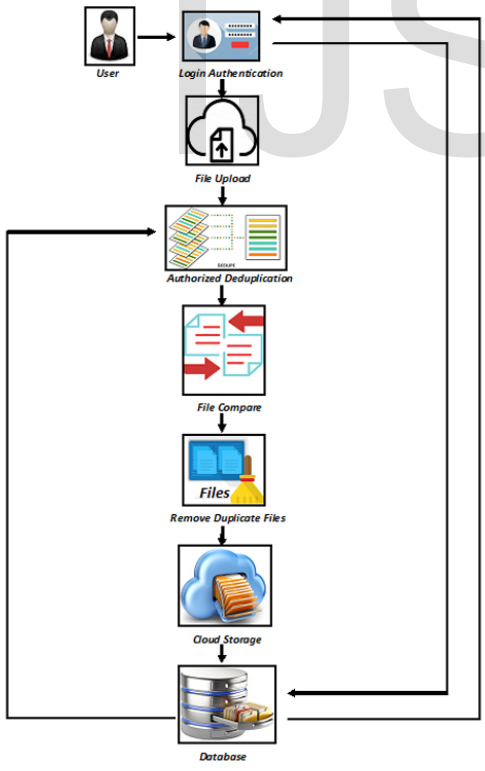

Fig: 4.1 Architecture Flow Diagram

The process of deduplication of files in the cloud database is represented in the figure.4.1 Here the user can log into the profile and upload the file to the database. Before uploading to the database there occurs a deduplication check where the file is checked in the database with the help of convergent key. File comparison is done with the files which are present in the database. Name based and content based comparison is done in the system so that dedupliation can be minimized. Hence the duplicate files can be removed and we can store the new data to the database with the help of Cloud storage.

4.2 PROPOSED DESCRIPTION

a. Convergent Encryption:

Convergent cryptography provides data confidentiality in deduplication. A user (or data owner) generates a convergent key from every original data copy and then encrypts the information copy with the generated convergent key. Simultaneously a tag is generated for the data copied. This tag is used to identify the duplication of data copied. Based on the data content tags are generated, hence the tag remains if two data are same. The tag alone is sent to cloud server to verify the duplication with the existing tags

present. The tag and convergent key is generated automatically .The convergent keys cannot be generated using the tag and involve the data concealment. The tag generated for the data copy and the encrypted data copy will be stored in the cloud. The same tag is used to retrieve data from cloud. This encryption scheme has four functions:

- KeyGen(M)-> K is the key generation rule that maps a datacopy M to a convergent key K
- Encrypt(K,M) -> C is that the radically symmetrical cryptography rule that takes each the convergent key K and also the data copy M as inputs then outputs a cipher text C
- Decrypt(K,C) -> M is that the cryptography rule that take seach the cipher text C and also the convergent key K as inputs then outputs the initial data copy M
- TagGen(M) ->T(M) is that the tag generation rule that maps the initial data copy M and outputs a tag T(M). Tendency to permit TagGen to come up with a tag from the corresponding cipher text by mistreatment T(M)=TagGen(C),wherever C=Encrypt(K,M).

b. RC5 Encryption:

RC5 is a fast, symmetric block cipher. A distinct data block size, usually consisting of 64 bits, is transformed into another distinct-size block. Key size, block size and the number of rounds are convertible and variable in RC5 ciphers. The Algorithm uses key encryption and decryption as well as key expansion. This algorithm also has a variable-length secret key, providing flexibility in its security level. One of the main significant feature of the design of RC5 is its simplicity. The encryption is based on only three operations: (i) addition, (ii) exclusive-or, and (iii) rotation.

RC5 is a parameterized algorithm and designated as RC5-w/r/b. The parameters are as follows:

- w - is the word size, expressed in bits. The standard value is 32 bits. The allowable values are 16, 32, and 64. RC5 encrypts two-word blocks. They are plaintext and ciphertext blocks, which are each 2w bits long.
- r - is the number of rounds and the allowable values are 0, 1_255.
- b – These are the no. of bytes in the secret key K
- K - The b-byte secret key. Allowable values of b are 0, 1_255.

RC5 uses an "expanded key table," S, derived from the user's supplied secret key K.

The size t of table S depends on the number r of rounds: S has t=2(r+1) words.

Encryption

We declare two w-bit registers X& Y as input blocks. Also we assume that key-expansion has already been performed, i.e. the array S[0…t-1] has been computed. Here is the encryption algorithm derived in Pseudo-code:

X = X + S[0];

Y = Y + S[1];

Fori=1 to *r* do

X = ((X⊕Y) <<< Y) + S [2*i];

Y = ((Y⊕X) <<< X) + S [2*i+1];

The registers A and B have the output. We observe the exceptional simplicity of this 5-line algorithm. We can also note that each RC5 round updates both the registers X and Y, whereas a "round" in DES updates only half of its registers. An RC5 "half-round" (one of the assignment statements updating A or B of the loop above) is thus more analogous to a DES round.

Decryption

The decryption routine is derived from the encryption routine easily to perform the operation.

It has a vice versa operation that of encryption.

For i=*r* down to 1 **do**

Y = ((Y - S[2 * i + 1]) >>> X)⊕ X;

X = ((X - S[2 * i]>>> Y) ⊕ Y;

Y = Y – S[1];

X = X – S[0];

c. Hybrid Encryption:

In this hybrid method [Fig.4.2] we are about to combine convergent key algorithm as well as RC5 algorithms in order to provide efficient security measures. Here the length of the key is controlled. The hybrid key operation is performed in the following way:

Step 1: Select the file to be uploaded to the cloud server.

Step 2: The convergent key is generated from the data copy by user to perform encryption operation.

Step 3: A tag is generated from the data copy simultaneously which is used to identify the duplicate data copy.

Step 4: The tag generated for the data individual copy and the encrypted data copy will be stored in the cloud.

Step 5: The tag is again encrypted by the RC5 key to provide additional security to the deduplication system for number of rounds.

Step 6: Same key can be used to decrypt the tag.

Step 7: Apart from this, encrypted shared key is generated for every join/leave event and forwarded to the key distribution centre (KDC) of the requester.

Step 8: This can be then verified by different privileged user and data copy.

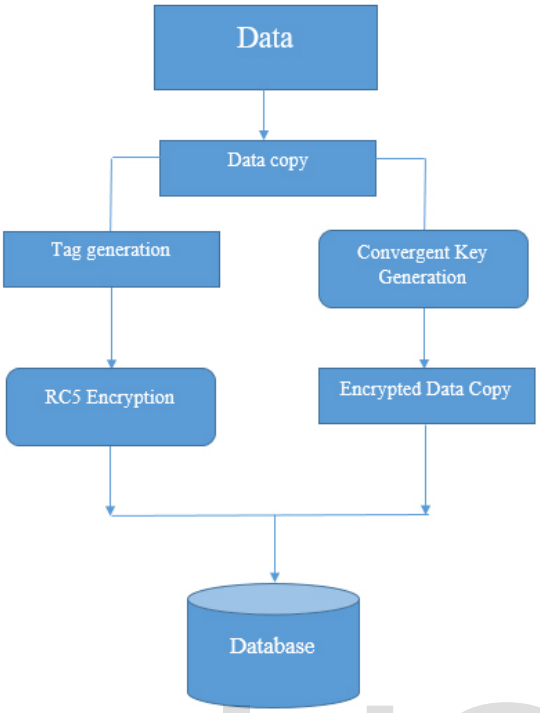

Fig: 4.2 Hybrid Flow Diagram

5. IMPLEMENTATION AND RESULT ANALYSIS

The proposed technology performance and the final results are mentioned in this section. The simulation of the experiment is done using CloudSim. This tools helps to provide the fundamental requirements to describe virtual machine, data centres and computing resources. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team's progress throughout the development activity. Also proposed system effectively secures the data and helps to reduce the duplication of files.

5.1 Implementation Screenshots

Snapshot is nothing but every moment of the application while running. It gives the clear elaborated of application. It will be useful for the new user to understand for the future steps.

5.1.1: Table Name: TOKEN DETAILS

This table 5.1.1 is used for storing the details of token.

Column Name	Data Type	Allow Null
Id	nvarchar(50)	☑
time	nvarchar(50)	☑
file_token	nvarchar(50)	☑
status	nvarchar(50)	☑
		☐

Table 5.1.1: Token Details

5.1.2 Enter the convergent key:

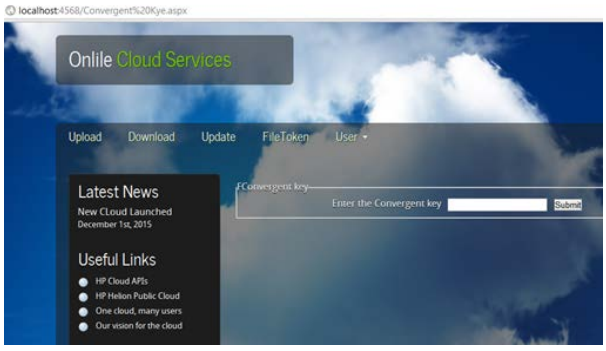

Fig 5.1.2: Enter the convergent key

The above fig 5.1.2 shows the user want to upload a file cloud, user needs to enter convergent key.

5.1.3 File uploads:

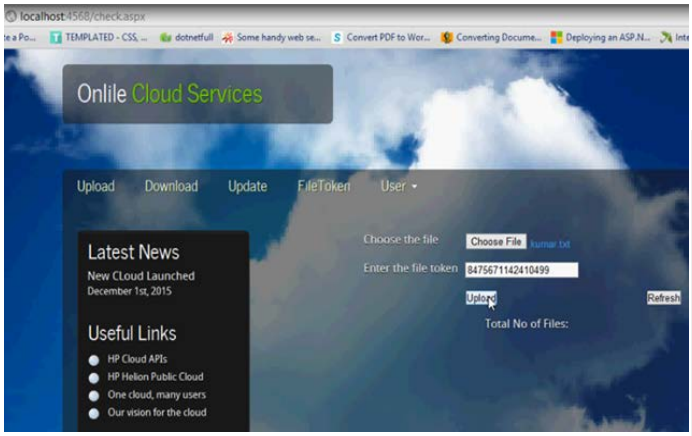

Fig 5.1.3: File Uploads

The above fig 5.1.3 shows the design of uploading file.

5.1.4 Duplicate check:

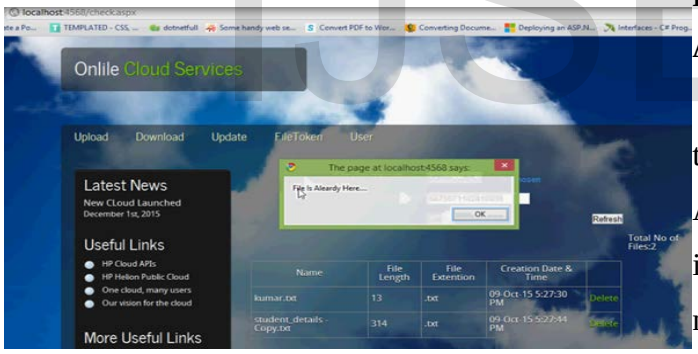

Fig 5.1.4: Duplicate Check

The above fig 5.1.4 shows, the user try to upload same file again, Duplicate check scheme will raise the message.

5.2 Security Algorithm Performance Analysis

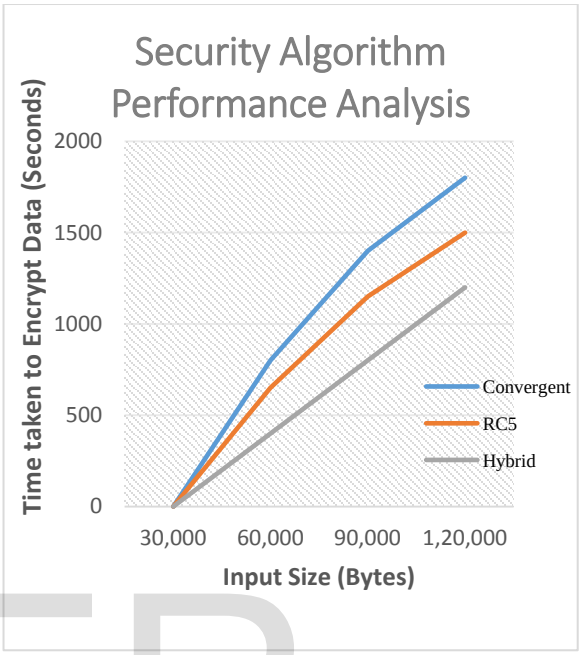

Fig 5.2: Security Algorithm Performance Analysis

In this section we have to represent the sensible result of the work done. According to the analysis report in Fig 5.2, it is clear that the Hybrid encryption method is more efficient in providing the security. Also the other algorithms are less economical comparing to Hybrid encryption method. The time required to complete the encryption is comparatively less for the Hybrid method. Thus on the win we can utilize the Hybrid method, since it provides a better performance.

5.3 Deduplication Ratio

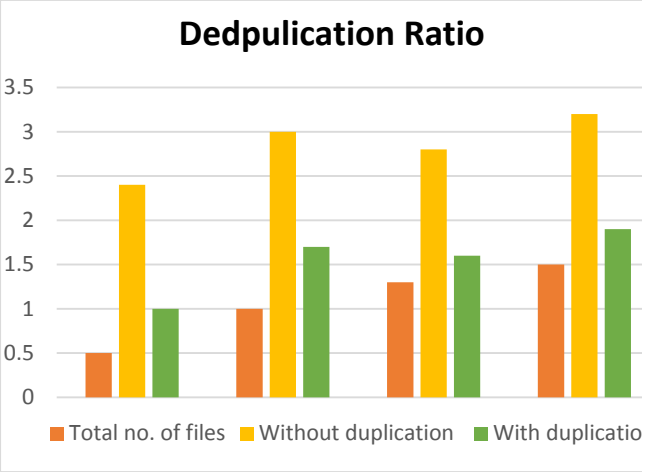

Fig 5.3: Deduplication Ratio

In the above comparison fig 5.3, we can clearly state that application for cloud storage that states Deduplication provides the better performance. Whereas the application that does not utilize Deduplication needs huge amount of files.

6. CONCLUSION& FUTURE WORK

We proposed a privacy-preserving deduplication system for data storage security in cloud computing. We used, the Hybrid based encryption technique, in order to ensure and secure that the external user/hacker would not learn any knowledge/detail about the data content stored on the cloud server during the sharing process, which not only reduces the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage/loss. In future work, we focus to construct an effective deduplication system in order to obtain a more efficient deduplication check scheme.

REFERENCES

[1] Dongzhan Zhang, ChengfaLiao, Wengjin Yang, Ran Tao, "Data Deduplication Based on Hadoop", pp. 147-152, 2017.

[2] M. Bellare and A. Palacio, "Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks," in Proc. 22nd Annu. Int. Cryptol. Conf. Adv. Cryptol., 2002, pp. 162–177.

[3] M. Bellare, C. Namprempre, and G. Neven, "Security proofs for identity-based identification and signature schemes", J. Cryptol., vol. 22, no. 1, pp. 1–61, 2009.

[4]M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraidedencryptionfordeduplicated storage", in Proc. 22nd USENIXConf. Sec. Symp., 2013, pp. 179–194.

[5]M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-lockedencryption and secure deduplication," in Proc. 32nd AnnualInt.Conf. Theory Appl. Cryptographic Techn., 2013, pp. 296–312.

[6] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-

duplication," in Proc. 24th Int. Conf. Large Installation Syst. Admin., 2010, pp. 29–40.

[7]Pasquale PuzioSeclud, RefikMolva, Melek O nen, Sergio Loureiro, "ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage", Issued: 2013.

[8] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider, "Twin clouds: An architecture for secure cloud computing", in Proc. Workshop Cryptography Security Clouds, 2011, pp. 32–44.

[9]Sharma Bharat, Mandre B.R. "A Secured and Authorized Data Deduplication in Hybrid Cloud with Public Auditing",Volume 120 – No.16, June 2015.