International Journal of Computer Science and Engineering Communications
Volume.4, Issue.4 (2016): Page.1445-1452
www.scientistlink.com/ijcsec

Secured Online Control Frame work for multi cloud Storage System using Authorized Deduplication Method

I.Nandhini
PG Scholar,
[1]Department of Computer Engineering, Knowledge Institute of Technology,
Salem, TamilNadu, India
nanthinibtechit@gmail.com

Abstract: *Data deduplication is the specialized and resent compression technology in the cloud computing. In hybrid cloud server storing of repeated data courses many storage management challenges. The data deduplication overcome this challenges by reduce the copy of data and save the space in the cloud service provider (CSP). It also saves the bandwidth of the storage cloud. This deduplication method provides the confidentiality to protect the owner data which has been stored in the cloud service by using the various encryption techniques. In this paper we propose new convergent encryption technique and authentication process to store and retrieval the data in cloud services. It is first official attempt to address the issue of secure authorized data/content deduplication. Data/content and the differential privileges of clients are considered while data duplicate check thus differing from traditional deduplication approaches. Our work is to propose new data/content deduplication construction supporting system which authorizes data duplicate check in public multi cloud architecture. However security analysis shows that proposed system is secure in terms of definitions specified in the proposed security model. Model of proposed authorized data duplicate check is implemented and executed using this prototype. The result outcome shows that proposed authorized data duplicate check scheme incurs minimal overhead in public multi cloud architecture as compared to normal/earlier operations.*

Keywords: *Deduplication, Hybrid cloud, cloud service provider, Authorized dat duplicate check.*

1. INTRODUCTION

Data deduplication is a recent terminology based on the utility and consumption of cloud computing resources[1]. It reduce the burden of maintaining big data, more and more enterprises and organizations have chosen to outsource data storage to cloud storage providers. This makes data management a critical challenge for the cloud storage providers[9]. To achieve optimal usage of storage resources, many cloud storage providers perform deduplication[2], which exploits data redundancy and avoids storing duplicated data from multiple users.

Deduplication Strategy. According to the architecture and the granularity of data processing, deduplication strategies can be mainly classified into the following types. In terms of deduplication granularity, there are two main deduplication strategies. One is File-level deduplication: the data redundancy is exploited on the file level and thus only a single copy of each file is stored on the server. Another one is Block-level deduplication: each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained deduplication. It is worth noting that for block level deduplication, the block size can be either fixed or variable in practice, and each method has its advantages and disadvantages . In this work, we focus on the block-level deduplication with fixed block size. From the perspective of deduplication architecture, there are also two strategies. (1) Target-based deduplication: users are unaware of any deduplication that might occur to their outsourced files. They just upload the files to the data storage server which then performs deduplication upon receiving the data. (2) Source based deduplication: unlike target-based deduplication, before uploading the data, the user first sends an identifier/tag of the data (e.g., a hash value of the data and thus much shorter) to the server for redundancy checking and thus duplicated data would not be sent over the network[6].

Large File Deduplication. In this paper, we focus on large file deduplication. Normally block-level deduplication can provide more space savings than file-level deduplication does in large file storage. Taking as an example, Alice and Bob want to store the same large file M in a server. Suppose the server performs file-level deduplication, which means only one copy of M will be saved. Later, Bob downloads M, appends several new pages to it, and uploads the modified file (denoted byM0) to the server. SinceM0 is different from M, the server needs to store the whole file M0. However, if block-level deduplication is

used, the server only needs to store the appended pages (denoted by ˈM), reducing the space cost from O(jMj + jM0j) to O(jMj + jˈMj). This approach can bring a significant space saving since jˈMj, jˈM0j. One drawback of the more fine-grained block-level deduplication is that it requires more processing resources. Fortunately, file-level deduplication and block level deduplication are not incompatible with each other.

In this paper, we present a technique that can achieve both of them (i.e., dual-level deduplication). Another aspect that should be taken into consideration is the bandwidth savings from large file deduplication. It has been reported that the cost of transferring data is almost the same as the space cost of storing the same amount of data for two months in the Amazon S3 server. Since uploading large files would consume extensive bandwidth, source-based deduplication seems to be a better choice for large file outsourcing. Unlike target-based deduplication which requires users to upload their files regardless of the potential data redundancy among those files, source-based deduplication could save the bandwidth significantly by eliminating the retransmission of duplicated data. Therefore, in contrast to target-based deduplication which saves only space, source-based deduplication can in addition save network bandwidth, which makes it more attractive in large file deduplication. However, source-based deduplication also has a drawback. A dishonest user who has learnt a piece of information about a file may claim that he/she owns the file. Such a problem has been identified by Halevi et al. in [14]. To overcome such an attack, they proposed a new notion called Proof of Ownership (PoW) where the user proves to the server that he/she indeed owns the entire file. It is clear that PoW should be implemented along with source-based deduplication. In the rest of the paper, we consider PoW as a default component in source-based deduplication[3]. From the above analysis, we can see that it is desirable to have Dual-Level Source-Based (DLSB) Deduplication for large files. Such a mechanism can achieve the best savings on space, computation, and bandwidth. In a DLSB Deduplication system, the user firstly sends a file identifier to the server for file redundancy checking. If the file to-be-stored is duplicated in the server, the user should convince the server that he/she indeed owns the file by performing a PoW protocol. Otherwise, the user uploads the identifiers/tag of all the file blocks to the server for block-level deduplication checking[8]. Finally, the user uploads data blocks which are not stored in the server.

Data Privacy. In the discussions above, we haven't considered data privacy issues. In reality, end users may not entirely trust the cloud storage servers. In order to protect data privacy, files may be encrypted first before being uploaded to the server. This brings a new challenge for deduplication since different users will use different keys to perform encryption, which would make two identical files completely different after being encrypted. Although searchable encryption can support equality testing of encrypted data, cloud storage providers still cannot perform any deduplication. The main reason is that, if a user does not store his encrypted file on the server due to deduplication, e.g., another user Alice has stored the same file in the server, then Bob could not retrieve the original filelater since he cannot decrypt Alice's file[6].

2. PRELIMINARIES

The advanced data deduplication system supporting authorized data duplicate check. In this new deduplication system, multi hybrid cloud architecture is introduced to solve the problem. The private keys for privileges access will not be issued to share users/clients directly, which will be kept and managed by the private cloud server system instead[4]. By this way, the users/clients cannot share these private keys of privileges access in this proposed construction, which means that it can put off the privilege access key sharing among users/clients in the above straightforward procedure. To get a file token, the user/client needs to send a request to the private cloud server system. To perform the data duplicate check for some data file, the user/client needs to get the data file token from the private cloud system. The private cloud server system will also check the user's/client's identity before issuing the equivalent file token to the user/client. The authorized data duplicate check for this file can be performed by the user/client with the public cloud system before uploading this data file[15]. Based on the results outcome of data duplicate check, the user/client either uploads this data file or runs PoW (Proof of Ownership).

2.1. S-CSP: S-CSP is an entity that provides a storage service in public cloud system. The S-CSP system provides the data outsourcing resource service and stores data on behalf of the users/client. To reduce the storage maintenance cost, the S-CSP system eliminates the storage of repeated data via data deduplication and keeps only unique data/file[5][13].

2.2. Authorized duplicate check: Authorized user/client is able to use his/her unique private keys to generate query for certain data file and the privileges access he/she owned with the help of private cloud system, while the public cloud system performs data duplicate check directly and tells the user/client if there is any data duplicate[11].

2.3. Data users: A user/client is an entity that wants to outsource data storage to the S-CSP system and access the data/file later. In a storage system supporting data deduplication, the user/client only uploads unique data/file but does not upload any duplicate data/file to save the upload memory, which may be owned by the same user/client or different users/client. In the authorized data deduplication system, each user/client is issued a set of privileges access in the setup of the secure system. Each data file is protected with the convergent encryption key and privilege access keys to realize the authorized data deduplication with differential privileges access[10].

2.4. Convergent encryption: Convergent encryption scheme provides data confidentiality in data deduplication. A user/data owner derives a convergent key from each original data copy and securely encrypts the data copy with the convergent key[15]. Furthermore, the user/client also derives a tag for the data copy, besides that the tag will be used to detect data duplicates. Here, assume that the generated tag correctness property holds, that is if two data copies are similar, then their tags are similar. To detect data duplicates, the user/client first sends the tag to the public cloud server to check if the equal copy has been already stored. That is both the convergent key and the generated tag are independently derived and the tag cannot be used to figure out the convergent key and compromise data privacy[7]. Both the encrypted cipher data copy and its corresponding generated tag will be stored on the public cloud side.

2.5. Private cloud: Compared with the traditional data deduplication architecture in cloud computing system, this is a new parameter introduced for facilitating user's/client secure usage of cloud service system. Particularly, since the computing resources at data owner side are restricted and the public cloud system is not fully trusted in practice, private cloud system is able to provide data owner with an execution surroundings and infrastructure working as an interface between user/client and the public cloud system[12]. The private keys for the privileges access are managed by the private cloud system, who answers the file token requests from the users/clients. The crossing point offered by the private cloud system allows user/client to submit files and queries to be stored and computed correspondingly. Through the proposed architecture reduce the overhead and execute the data deduplication efficiently.

2.6. PoW (Proof of Ownership): In cloud storage, the server may be untrusted in terms of security and reliability since it may have the incentive to reduce the cost by shifting users' data to slower and cheaper storage devices. It may also intend to hide data loss/damage due to accidents or attacks in order to save the reputation. Therefore, it is also important for the users to do regular checking on the availability of their data. In order to allow the end user and the data storage server to perform secure and efficient data storage checking, a new cryptographic primitive called Proof of ownership.

3. CHALLENGES IN PRIVILEGE ACCESS

The rapid increases of digital data today's cloud service system providers offer both highly available memory and massively parallel computing services at relatively minimum costs. As cloud computing system becomes prevalent, a maximizing amount of data is being stored in the cloud system and shared by users/clients with specified privileges access, which define the access rights of the stored data/file. One critical issue of cloud storage system services is the administration of the ever-maximizing volume of data. To make data administration scalable in cloud computing system, data deduplication has been a well-known method and has attracted more and more attention lately. Deduplication is a specialized data compression method for eliminating duplicate copies of repeating data in memory. As an alternative of keeping multi data copies with the same content/information, data deduplication eliminates redundant data by keeping only one copy and referring other redundant data to that copy.

Even though deduplication brings a lot of advantages, security and privacy concerns arise as client's sensitive data are susceptible to both inside and outside malicious attacks. Traditional encryption scheme, while providing data privacy, is incompatible with deduplication. Particularly, traditional encryption schemes require different clients to encrypt their data with their own generated keys. Thus, equal data copies of different client's will lead to different cipher texts, making data deduplication impossible. However, existing data deduplication systems cannot support differential authorization data duplicate check, which is important in many real time applications. In previous deduplication systems, the private cloud system is involved as a proxy system to allow data users to securely perform data duplicate check with differential privileges access. Existing prototype overhead is high in file upload operations.

3.1 Differential privileges based Deduplication Approach

It is an authorized data deduplication system each client is issued a set of privileges access during system initialization. Each file/data uploaded to the cloud system is also bounded by a set of privileges to denote which kind of clients is allowed to perform the data duplicate check and access the data's/files. Before submitting user data duplicate check request for a file, the client needs to take that file and its own privileges access as inputs. The client is able to find a data duplicate for the file if and only if there is a copy of this file/data and a matched privilege access stored in cloud system. For e.g., in a company/organization, many different privileges access will be assigned to employees/end user's. In order to save computation cost and efficient administration, the data will be moved to the memory/storage server provider (i.e. S-CSP) in the public cloud with specified privileges access and the data deduplication scheme will be applied to store only one copy of the user same file in public cloud. Because of secure privacy thought, some files will be encrypted and permitted the duplicate check by employees/end user's with respective privileges to realize the access control.

The user access right to a file is defined based on a set of privileges access. The exact definition of a privilege varies across more applications.

For e.g., user may define a role-based privilege access according to job positions (e.g., CEO, Project Lead, and Engineer), or user may define a time-based privilege access that specifies a valid time period (e.g., 2015-12-12 to 2016-01-31) within which a file/data can be accessed. A user/client, say Alice, may be assigned two privileges access "CEO" and "access right valid on 2015-12-12", so that alice can access any file/data whose access role is "CEO" and accessible time period starts from 2015-12-12. Each privilege access is represented in the figure of a short message called token. Each file/data is associated with some file/data tokens, which represent the tag with specified privileges access. A user/client computes and sends data duplicate check tokens to the public cloud system for authorized data duplicate check.

4. PROPOSED SYSTEM

This proposed work focuses on to reduce overhead by extern the hybrid cloud architecture for data deduplication in terms of multi cloud integration system. Thus multi public clouds are integrating each other and apply deduplication in each and every public cloud under receiving duplication check user request. Through the following architecture reduce the overhead and execute the deduplication efficiently. This Proposed system is designed to solve the differential privilege access problem in secure data deduplication. In this system, the S-CSP system is honest but interested and will honestly perform the data duplicate check upon receiving the duplicate request from users/client. In this system a higher level secure privacy is defined and achieved. This authorized data duplicate check method incurs minimal overhead compared.

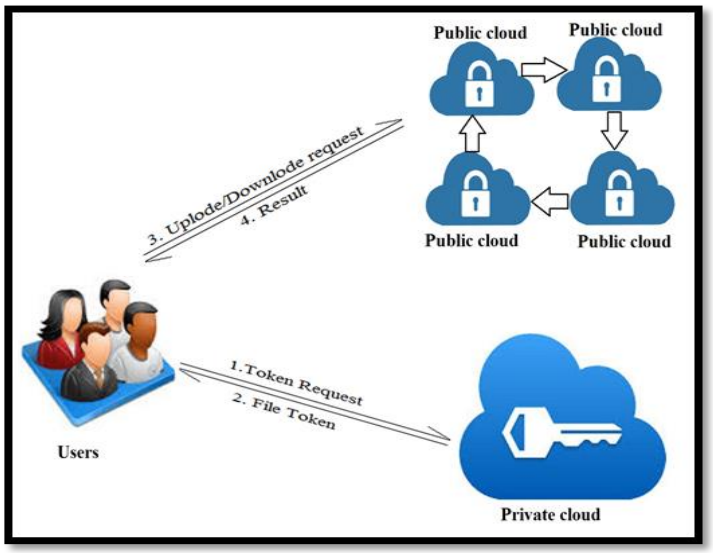

Figure 1. Architecture for Authorized deduplication in multi cloud

4.1 Convergent encryption:

It provides data confidentiality in deduplication. A data owner derives a convergent key from each original data and encrypts the data with the convergent key. Additionally, the user also derives a tag for the data; such that the tag are want to detect duplicates. A convergent encryption scheme can be defined with four primitive functions:
On input 1^{λ}, the algorithm generates a prime p, the descriptions of two groups G,G_T of order p, a generator g of G and a bilinear map e : $G \times G \rightarrow G_T$. Choose an integer s $\in$ N and three hash function $H_1 : \{0,1\} \rightarrow Z_p$, H_2: $\{Z_p\}^s \rightarrow G$, $H_3 : G \rightarrow \{Z_p\}^s$. Pick s elements randomly $u_1, u_2,\ldots, u_s \rightarrow_R G$. The system parameters are P =< p, g, G, G_T, e, H_1, H_2, H_3, s, u_1, $u_2,\ldots, u_s$ >.
KeyGen(M**).** Given a data file M = M[1] … M[n] where for all $1 \le i \le n$, M[i] $\in$ $\{Z\}^s$, compute the master key k_{mas} and each block key k_i as follows,
Enc(k_i,M[i]**).** Given a block message M[i], and the corresponding block key k_i, output the block ciphertext as C[i] = $H_3(k_i)$ + M[i].
Dec(ki;C[i]**).** Given a block ciphertext C[i], and the corresponding block key k_i, compute M[i] = $H_3(k_i)$ _ C[i]. If $k_i = H_2$ (M[i]), output M[i]; otherwise output $\perp$.
TagGen(M**).** Given the fileM = M[1] … M[n], output thefile tag T0 and each block tag T_i as follows,
F TagGen(M; i**):** take M and the block index i, generate the master key kmas, the corresponding block key ki and block ciphertext C[i], split C[i] into s sectors: $\{C[i][j]\}_1 \le j \le s$, and output, $T_i = (k_i \prod_{j=1}^{s} u_j^{c[i][j]})k_{mas}$. In our scheme, some auxiliary data $aux_i = e(k_i, T_0)$ is also generated and attached to the block tag T_i during block tag generation for some discussions on T_i and aux_i.

5. SYSTEM ANALYSIS
The tag of user file content will be determined by the file F and the privilege to perform the authorized deduplication. This tag is difference with the traditional notation of tag; we call it file token instead. For support authorized access, a secret key k_p will be bounded with the data privilege p to generate a file token of the user file. Let ω_{fp}=TagGen(F,k_p) denote as the token of file F that is only allowed to access by owner with privilege p. The token of a file F could only be computed by the owner with privilege p. As a result, if a file has been uploaded by an owner with a duplicate file token $\omega_{fp'}$, then a duplicate check sent from owner will be successful if and only if the owner also has the file F and privilege ρ. This token generation function could be implemented as H(F,k_p), where H(.) denotes a cryptographic hash function. In the paper we present a straight forward method with the technique of token generation TagGen(F,k_p) for the deduplication system. The main idea of this basic construction is to issue corresponding privilege keys to different privileges of each user will compute the file tokens process and perform the duplicate check based on the privilege keys and files. If suppose that there are N users in the system and the privileges in the universe is defined as P={$\rho_1,\ldots..,\rho_n$}. For each privilege ρ in P, a private key k_p willbe selected. For a user U with a set of privileges P_U, the owner will be assigned the set of keys $\{k_p\}_{pi€Pu}$.

5.1. File uploading.

The data owner U with privilege set can be consider as P_U that wants to submit and share a data file F with other users have the privilege set P_F= {p_j} The user computes and sends S-CSP the file token ω_{fp}=TagGen(F,k_p)for all p p€P_F. If a duplicate is found by the Storage Cloud Service Provider, the owner proceeds PoWs of this file with the S-CSP. If the proof is passed, the owner will be assigned a pointer, which allows the owner to access the file. Otherwise, if no data copy is found in the cloud storage, the user perform the encryption process for the file C_FEnc(k_p,F) with the convergent key k_F= $KeyGen_{CE}$(F) and uploads (C_F ,{$\omega'_{F,p}$}) to the cloud server. The convergent key k_F is stored by the user locally. The challenger picks randomly $F \rightarrow \{0,1\}$. If F = 0, then runs the source M as, (M_0;Z) M(λ).Otherwise, if F = 1, chooses M_1 uniformly at random from$\{0,1\}^{Mo}$. Set M = M_F. Suppose n(λ) is the block numbers. For each i = 1,…,n(λ), the challenger computes k_i**P-KeyGen**(M[i]) and then computes the ciphertext as,C[i] **Enc**(k_i ,M[i]). The challenger also computes the file tag and block tags as follows, T0 **F-TagGen**(M); Ti **PTagGen** (M[i]). Set T = T_0; $T_1,\ldots,T_n$(λ)g. Finally, the challenger gives auxiliary information Z, tags T, and the ciphertext C to the adversary.

5.2. File retrieving.

The owner wants to download a file F. He first sends a request and the file token to the Storage Cloud Service Provider. Upon receiving the request and file token, it will check whether the user is eligible to download F. If the request failed, the S-CSP sends response to the user to indicate the download failure. Otherwise, the S-CSP send the corresponding content of the user i.e. ciphertext CF to the owner. Upon receiving the encrypted file from the Cloud Service, the owner uses the key k_F stored locally to recover the original file F. Several security problems have been occurring in the construction of authorized deduplication. In this each user will be issued private keys $\{k_p\}_{piPu.}$ for their corresponding privileges of their data, denoted by P_U in our above construction. These private keys $\{k_p\}_{pi€Pu.}$ can be used to generate file token for duplicate check which is

applied by the user. However, during file uploading, the owner gets file tokens for sharing data with some other owners with privileges PF, which is only chosen from P_U. This deduplication system cannot protect the privilege private key which has been sharing among the owners. The same private key will be issued by the user for the same privilege in the construction. As a result, the owner may produce new privilege set P^* by collude and generate privilege private keys. This new privilege set does not belong to any of the colluded user. The deduplication system cannot provide the security for the sensitive date of the user but the traditional convergent encryption system can protect the semantic security of unpredictable files.

6. Performance Analysis

Conduct test evaluation on our prototype. Our evaluation process focuses on comparing the security, bandwidth and storage of the proposed system. The authorized deduplication save the storage space and bandwidth by eliminating data copies and it also provide authentication to protect the sensitive data in the cloud.

TABLE 1 Computation Time of Tag Generation

Algorithms	Number of sector per block			
	64	128	256	512
Key generation	0.421	0.963	1.641	3.6985
Encryption	0.437	0.878	1.069	3.876
Decryption	0.345	0.845	1.908	3.564

In the computation time of both tag generation and block key retrieval increases with the number of sectors per block. This is due to the fact that for each sector, we need to compute one exponentiation operation. Moreover, the computation cost of block key retrieval is slightly higher than that of tag generation due to the additional group inversion operation. Specifically, when s = 128 (block size is 4 KB), the computation time is 0.878 seconds for tag.

Improved Security of the data

We improve our PoW protocol for stronger security based on the PoS protocol construction. Note that in order to allow PoS, the server should maintain the leaves of the Merkle-Hash-Tree. Equipped with these additional data, we show that our proof of ownership protocol can be improved to obtain stronger security by requiring the prover (user) to returning data blocks instead of aggregated block tags. Notice that here we can allow the adversary to have all the tags of a file. In a proof of ownership protocol, upon receiving a challenge query Q = f(i,v_i)g (i Є [1; n],v_i Є Zp), we require the user/prover to return λj =P(I,j)ЄQ viC[i][j]; 1 ≤ j ≤ s. After receiving the proof, the server computers π(I,j)ЄQ $e^{(ki; T0)}$ and T = π(I,j)ЄQ Ti ^{vi} , and then checks e(T,g) =Q(j.i) Є Q e(ki,T0) + e(Q_{sj}=u_{ij}; T0). The security of the various size data can compare with and without deduplication process the result shows in the fig2 from this result the deduplication process is give more security for the data in the public clod storage.

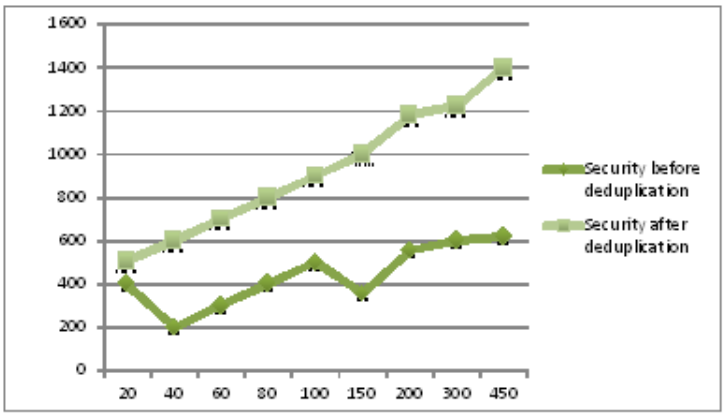

Figure 3. comparison of security

Improved bandwidth of the data

In the above fig 3 the bandwidth of the different storage data are compared. The different size of data can store in the cloud. The deduplication process reduces the copy of data and save the bandwidth of the store data. The fig 3 describes the comparison of bandwidth of deduplication to other methods.

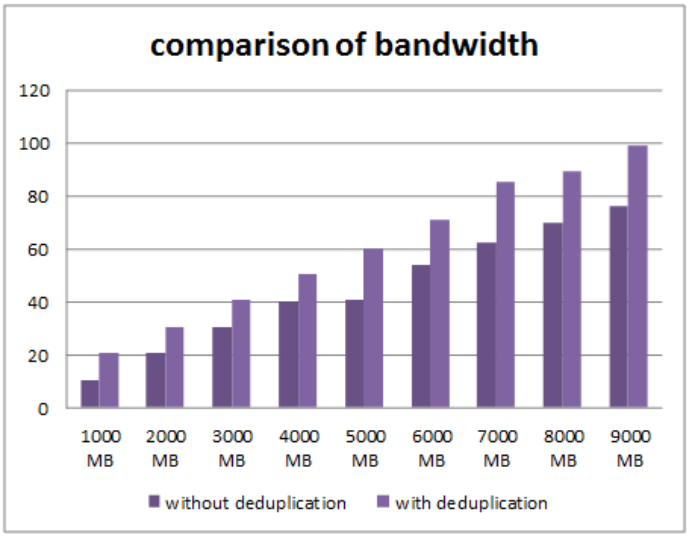

Figure 3. Comparison of bandwidth

7. CONCLUSIONS

In this paper, the notion of authorized data deduplication was proposed to protect the data level security by including differential privileges of users/clients in the data duplicate check. Also presented several new data/content deduplication methods supporting authorized data duplicate check in hybrid cloud architecture, in which the data duplicate-check tokens of files are generated by the private cloud server system with private keys. Security analysis demonstrates that these approaches are secure in terms of insider and outsider intruders specified in the proposed security architecture. As a proof of concept, implemented a prototype of proposed authorized data duplicate check method and executed on prototype. Results showed that authorized data duplicate check method incurs low overhead compared to convergent encryption scheme and network transfer.

8. FUTURE SCOPE

To protect the privacy of sensitive data/content whereas supporting data deduplication, the convergent encryption scheme has been proposed to encrypt the user data/content before outsourcing. Future work focuses on to reduce overhead by extern the hybrid cloud architecture for data deduplication in terms of multi cloud integration system. Thus multi public clouds are integrating each other and apply deduplication in each and every public cloud under receiving duplication check user request.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my Knowledge Institute of Technology, Guide and Staff members for their continuous support of for this paper and finally my friends for their coordination in this work.

REFERENCES

[1] Chuanyi LIU, Dapeng JU, Yu GU, Youhui ZHANG, Dongsheng WANG, David H.C. Du2.Semantic Data De-duplication for Archival Storage System. In IEEE Transactions, August 2008.

[2] Neal Leavitt, Hybrid Clouds Move to the Fore front. Published by the IEEE Computer Society, May 2013.

[3] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data deduplication scheme for cloud storage," Tech. Rep. IBM Research, Zurich, ZUR 1308-022, 2013.

[4] S. Bugiel, S. Nurnberger, A. Sadeghi, and T Schneider Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011

[5] J. Li,X. Chen, M. Li, J. Li, P. Lee, and W. Lou. Secure Stand. deduplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems,Volume 25,Issue 6,2013.

[6] Rongmao Chen, Yi Mu. BL-MLE: Block-Level Message- Locked Encryption for Secure Large File Deduplication. IEEE Transactions on Information Forensics and Security, August 2015.

[7] M. Bellare, S. Keelveedhi, and T. Ristenpart.Message-locked encryption and secure deduplication. In EUROCRYPT, pages 296–312, 2013.

[8] P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted de-duplication. In Proc. of USENIX LISA, 2010.

[9] Pooja S Dodamani, Pradeep Nazareth. A Survey on Hybrid Cloud with De-Duplication, International Journal on Recent and Innovation Trends in Computing and Communication, vol. 2, Issue 12, December 2014.

[10] Survey on Secure Authorized De-duplication in Hybrid Cloud, International Journal on Recent and Innovationm Trends in Computing and Communication, Volume: 2 Issue: 11 3574 – 3577.

[11] Bhushan Choudhary, Amit Dravid A Study on Authorized Deduplication Techniques in Cloud Computing, (IJARCET) Volume 3, Issue 12, April 2014.

[12] Aparna Ajit Patil, Dhanashree Kulkarni. A Survey on:Secure Data Deduplication on Hybrid Cloud Storage Architecture,International Journal of Computer Applications Volume110–No. January2015.

[13] Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P.C. Lee, and Wenjing Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication", IEEE, Volume: 26, 2015.

[14] libcurl, (1997). [Online]. Available: http://curl.haxx.se/libcurl/

[15] C. Ng and P. Lee, "Revdedup: A reverse deduplication storage system optimized for reads to latest backups," in Proc. 4th Asia-Pacific Workshop Syst., http://doi.acm.org/10.1145/2500727. 2500731, Apr. 2013.

About Author

Ms.I.Nandhini is Completed B.Tech Information Technology in Vivekanandha institute of engineering and technology for women. Then she pursuing M.E, (Computer Science Engineering) in Knowledge Institute of Technology, Salem, India 2015.She has present a paper in 1 International conference, 3 National conference and 2 Journal.