A Secured Deduplication of Encrypted Data Over an Attribute-Based Cloud Storage

Olu Olu Osaronwolu[#1], Matthias Daniel[*2], V. I. E Anireh[#3]

[#1]*Msc Student, Computer Science department, Rivers State University, Nigeria.*
[#2#3]*Lecturers, Computer Science department, Rivers State University, Nigeria.*

Abstract

Cloud Storage offer users an easy way of storing and sharing data, but due to privacy concerns of users, data encryption was introduced. Among available forms of data encryption, Attribute based encryption (ABE) has been largely adopted because it provides a pliable secured approach for users to save and distribute data in cloud by enforcing an access-control policy over data, hence only users who satisfy the access-control policy will be able to decrypt and access the encrypted data. However, traditional ABE cloud storages do not support secured data deduplication (a process of eliminating copies of identical data) which is vital in data management to ensure judicious utilization of storage space and network band-width. This work proposes a scheme to integrate secured data deduplication on an Attribute-based Encryption Cloud Storage by resorting to a dual cloud setting, where one (private) cloud executes data deduplication, while the other (public) cloud provides storage service. Python programming language, jQuery, JavaScript, CSS, HTML. Heroku server was used to test the system during development. Experimental results show that secure deduplication of encrypted data can be achieve on an Attribute-based cloud storage through a dual cloud approach where the private cloud is trusted.

Keywords - *Deduplication, ABE, Cloud computing.*

I. INTRODUCTION

Cloud computing [13] can be simply put as a recent category of network-based computing built on the platform of internet. In cloud computing a set of instructions is executed over a single or multiple server rather than a traditional computing device i.e. a Personal Computer (PC) or smart mobile device (smartphones, tablets or ultrabooks). This makes Cloud Computing a dispersed architecture where server resources are centralized on a scalable platform to make computing power and services available whenever it is requested [1].

Among the services provided by the cloud, cloud storage and data sharing has grown to be very popular as cloud infrastructures provides the facilities for users to efficiently save, manage and distribute files.

Cloud storage provides a means for data providers to remotely manage, maintain and back up data. These data providers are also referred to as users. Hence users who have stored their data online can easily access their data from any location via internet connection.

Cloud storage also goes further to create a platform where users can confidently transfer files unto the cloud without breaking the file confidentiality/privacy, exposing sensitive information to unauthorized users, but would like certain users defined by a specified credential to have access rights to the transferred file.

The confidentiality, privacy and security concerns of data providers over their files in the cloud leads to the introduction of data encryption. Therefore, through the technique of Attribute-Based Encryption (ABE), users can save files in encrypted forms, governed by an access policy so that only the users specified by the policy i.e. those whose attributes (or credentials) fulfilled the security standard enforced over the file, will be able to decrypt and access the encrypted data, in ABE encryption the private key belonging to a user is linked to a defined set of attributes, so when a message is encrypted under a given access policy i.e. a set of attributes, only a user whose attributes match the defined access control policy can decrypt the ciphertext with his/her private key. [2]

With ABE the challenges of data confidentiality and security are resolved, data becomes protected in a format that allows only authorized users possessing the defined security criteria, thus meeting the access policy standard to access the data.

But with the introduction of an Attribute-Based Encryption to data new issues arises in the area of effective management of the cloud data for maximum utilization of storage space and network bandwidth because in a cloud with plethora of users there is a strong tendency that multiple users would possibly encrypt and upload the same file to the storage and this will lead to having so many duplicate copies of the same data, having such duplicate files will hence improperly consume storage space and bandwidth therefore there is need to avoid/eliminate such duplicates. [3]

Data Deduplication which is the process of detecting and eliminating redundant data in a data set and an important tool in cloud storage management becomes almost impossible to carry out in an Attribute-Based Encryption storage. Some existing cloud system which still carry out this data deduplication management process do so through

following steps which either break the security and privacy measures enforced over the data or make them very vulnerable to attack by adversaries.

The method of enforcing data deduplication on an Attribute-based encryption cloud storage introduced within this paper eliminates duplicates without breaking the security and privacy that data providers very much desire over their data. A standard Attribute-Based Storage does not support data deduplication but with the method proposed within this work a separate cloud is used to deduplicate data before storing on an open-public cloud.

An Attribute-Based Cloud Storage provides the benefit for data providers to use an access policy to share their files saved in encrypted form on the cloud with certain user possessing predefined attributes (or credentials) in order to avoid releasing sensitive information to unwanted parties. In this method of encryption, the private key of a user is linked to a set of attributes, when a message is encrypted under the defined access policy i.e. over an attribute set, only a user whose private key satisfies the access policy associated with the encrypted data will be able to decrypt the data.

While Attribute-Based Storage provides a desired security and privacy level it cannot securely perform data deduplication; a vital process required in data management scheme of a cloud storage to eliminate copies of duplicated data, hence save storage volume and network bandwidth.

To illustrate this problem of data deduplication over an Attribute-Based cloud storage, let's consider three data providers- Jude, Julie and Judith, who wants to upload the same copy of a large software S to the cloud. Jude uploads the software S to the cloud and encrypts S under an access-control policy A over a specified attribute set, so that only users whose credentials match the defined attribute set or access policy will have the authorized ability to decipher the ciphertext. Later Julie uploads the exact file, although this time with a different access policy over another set of attributes and finally Judith also uploads this same file but with a totally different access policy over a different set of attributes. Since these files are all encrypted the cloud system will not be able to detect that the underlying plaintext of Jude, Julie and Judith's files are all the same, so it will store S three times thus wasting storage volume and network bandwidth.

Since Attribute-Based Storage and Data Deduplication are two separate techniques that are highly employed in cloud storage and an Attribute-Based Storage does not support Data Deduplication. It will be needful to develop a type of cloud storage that supports both of these technologies.).

II. LITERATURE REVIEW

The concept of public key cryptography also referred to as asymmetric key cryptography was introduced in [12]. Identity Based Encryption (IBE) [4] was later developed as an improvement. They asserted that their method for resisting collusion attacks was efficient on Attribute Based Encryption. But the computational cost of their scheme in areas such as ciphertext size and private-key size exponentially increased with attribute number. In their paper "Id-based encryption for complex hierarchies with applications to forward security and broadcast encryption" they presented a Forward-Secure Hierarchical Identity-Based Encryption (fsHIBE) scheme and showed that it can additionally design a forward-secure public-key broadcast encryption scheme, where the privacy of pre-transmission is secured in broadcast encryption platform. They further categorized fs-HIBE into a collusion- resistant multiple hierarchical ID-based encryption scheme, and proposed its usability in secure interaction with entities possessing several roles in role-based access control. The bilinear Diffie-Hellman [12] assumption in random oracle model formed the basis for the security of this scheme. The forward-Secure HIBE Scheme protocol was built on the development of [5] and also on the Forward-Secure Public-Key Encryption (fs-PKE) scheme developed by [6]. In the protocol the requirements of dynamic joins, forward security, joining time-obliviousness, and independent key updates are all met.

Building upon the work of [4], another form of Identity-Based Encryption (IBE) scheme was proposed and called "Fuzzy Identity-Based Encryption" [2]. In this scheme an identity was seen as a descriptive attribute set. The scheme allowed a private key for identity, k, to decrypt a message encrypted with an identity, k1, if and only if the identities k and k1 are closely related when measured by the "set overlap" distance metric.

[7] developed a more robust type of Attribute-Based Encryption Cryptosystem and their accomplishment was showcased. Their construction was term Key-Policy Attribute-Based Encryption Scheme and they acknowledged that it was similar to a Secret-Sharing Scheme but the big difference between both systems was that while the secret-sharing scheme gave room for two different users to collude, their construction prohibited such cooperation. To illustrate this; let's say the access structure "A AND B" governs the key of User 1, while the access structure of "B AND C" governs the key of User 2, it will be impossible to view a ciphertext having just the attribute C through the means of colluding. Such security was attained by employing and standardizing the scheme of [2] unto Fuzzy Identity Based Encryption to handle the technicality.

Two large universe Attribute-Based Encryption approaches were proposed by [9]. In this kind of construction attributes can be formed from any string and at system set up, these attributes are not required

to be listed. At foremost, Prime order bilinear groups was used to build their original Ciphertext Policy Attribute Based Encryption (CP-ABE), whereas their later construction had remarkable advancements over the large universe KP-ABE construction by [10].

[11] developed a data deduplication strategy that provided and space efficiency and data security in a single-server storage and distributed storage systems

III. RESEARCH METHODOLOGY

Prototyping is adopted in this research work in order to implement and evaluate the functionality of the proposed system and to also gather feedbacks from which the system can be improved upon later on. Prototyping is analysed as a unique research tool because it provides a proof of concept. [8] in an extensive bibliography provides evidences of the growing use of prototyping as an application system design and development methodology.

Prototyping is often referred to as evolutionary design or incremental development because of how it cuts across these two models to offer a means to implement an idea, evaluate/test the idea and further improve upon the idea, thus is serves as an early working version of a concept. Prototyping involves phases that are similar to the waterfall design model with a major difference being that each phase is fully completed before you can move unto the next, since it is harder to change something that was not discovered on the concept stage.

A. SYSTEM DESIGN

The proposed ABE cloud storage with secure deduplication of Encrypted data with four entities involved: data providers/outsourcers, Clients/End-users, attribute authority, and Cloud (Public and private)

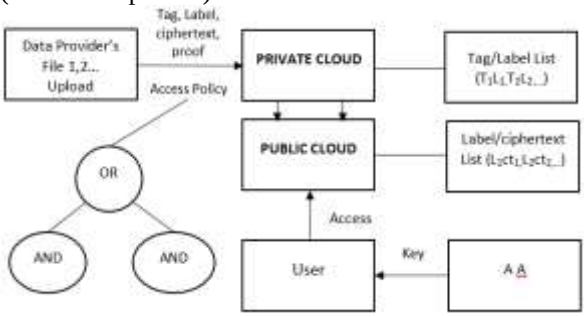


Fig. 1: Architecture of the Proposed System

A data provider wishes to transfer their data to the cloud and share it among clients having specific credentials. The Attribute Authority generates and gives each user a decryption key associated with their specified credentials or attribute set. Cloud is made up of the public cloud which manages data storage and the private cloud which executes security and deduplication computation. When a file storage request is created, the Data Provider is obliged to produces a Label (L) and Tag (T) related to the file, then encrypt the file under an access structure that governs the desire attribute set. Also, each data provider produces a proof pf to validate the association of the tag, label and ciphertext/encrypted data ct, although the proof is not saved but only used at the phase of tag checking for all new generated data storage request. When the storage request is received, validation of the proof is first executed by the private cloud, which then moves forward to compares the new tag against existing tags in the system for any match. Where no match for the new tag exist, the private cloud then adds it and the label to a tag-label list, and forwards new encrypted data and label, (ct, L) to the public cloud for storing. But if there is a match of the new tag with those in the system, the private cloud runs the following steps of instruction.

- Where access policy governing ct is contained in ct', given that ct is an existing ciphertext and ct' is a tag from a new request. The private cloud proceeds to eliminate the new request for storage; else, if the access policy in ct' is a subset of that in ct, the private cloud instructs the public cloud to replace the saved tuple (L', ct') with the new tuple (L, ct) provided L = L'.
- Where the access policies governing ct and ct' are not mutually contained, then the private cloud executes a ciphertext regeneration scheme to produce a new ciphertext having identical underlying plaintext file and related with an access structure which becomes the union of the two access structures, then it forwards the original label and the resulting ciphertext to the public cloud storage.

On the part of the client/end-user phase, only those who have a valid ABE private key produced by the attribute authority, and meet the set of attributes specified by the Access structure, would then have the rightful privilege to download, decrypted and access an item. Each client-user can the use the label to evaluate the genuineness of the information/item, and where there is no discrepancy accept the decrypted information/item as valid. For the adversarial model of the cloud storage system, the assumption that the private cloud is "curious-but-honest" is made. That is to say that it will try to get encrypted data but will be trusted to oblige with protocols, while the public cloud is distrusted, because it has very high tendency to tamper with the label and encrypted data tuple transferred from the private cloud, however such maliciousness will eventually detected either in the private cloud or by the user. An additional dissimilarity between the private and public cloud is a private is trusted never to collude with users, but the public cloud storage can collude with users. This assumption is based on real-world practice where private cloud are more trusted than public cloud storages. It is also assumed that data users could attempt to access data outside their rights of authorization. Additionally, together with attempts of gaining information from encrypted data

store on the cloud, 'duplicate faking attacks' may also be carried out by malicious users.

This section presents a brief description of the procedures mention in the architecture of the proposed system, the following are essentials components of the algorithms:

Note: the following symbols and the terms they represent

"1λ" represents set up
"pars" represents public parameters
"msk" represents master private key
"**A**" is an Attribute Set A
"skA" is an Attribute-based Private key
"A" is an Access Structure
"skT" is a Trapdoor key (and is never revealed to anyone.)
"CT" is a tuple comprising of Tag (T), Label (L), ciphertext (ct), Message (M).

a) System set:

In this phase the system prepares necessary file, modules and parameters for execution. The major input of this phase is the security parameter λ. The output is the public parameter pars and the master private key (msk). The Attribute Authority runs this set up. It is represented by the equation (1λ) → (pars, msk)

b) Key Generation:

This phase takes as input the output of the previous phase together with a defined attribute set and produces an Attribute based Private key for the defined set as output. The Attribute Authority also executes this phase. It is represented by the equation (pars, msk, **A**) → $sk_{\mathbf{A}}$

c) Encrypt File (Generate Tag and Label/Upload):

In this phase the system takes as input: the public parameter, message and access structure governing a defined attribute set and gives as output, a trapdoor key and a collection of other metadata called a tuple. This phase is executed by the user (data outsourcer and securely uploaded to the private cloud). It is represented by the equation CT = (T, L, ct, pf)

d) Validity-Test (pars, CT) → 1/0:

In this phase, the system takes as input: public parameter and the Tuple (CT) the Taking the public parameter pars and a tuple CT as the input, then compares it with existing records. If a match exists and the proof is valid "1" is return. If not, "0" is returned. The private cloud executes this phase.

e) System Evaluation (Equality testing):

The system takes as input: public parameter, and two tuples for evaluation and deduplication where required. This part of the system is executed on the private cloud. The output is in binary for either positive or negative return.

f) Decrypt:

It is important to include this part of the system. The public parameter, label, encrypted data and Attribute based private key relating to a defined attribute set, are taken as input and the output is the legal information which is only possible when the security standard is satisfied.

Security Notion

Semantic Security is a cryptographic measure that is enforced in order to ensure that only trivia information about the plaintext can be possibly extracted from a ciphertext in any attack or intrusion. Semantic Security in cloud storage further guarantees data confidentiality which is the protection of data so that only users with defined authorization can have access to its information while unauthorized users will be denied the rights to access the protected data.

The privacy provided by an encryption system is captured by indistinguishability under chosen plaintext attacks (IND-CPA) or indistinguishability under chosen ciphertext attacks (IND-CCA) but introducing secured deduplication into an encryption system voids those privacies so let's say a malicious user therefore a trusted private cloud is introduced in this system to handle the complex mathematical computation need to enable both secure encrypted data deduplication and also maintain data confidentiality and integrity. An ABE cloud storage system with secure deduplication Π is IND-CPA secure if the advantage function referring to the security game $\mathrm{GAME}^{\mathrm{IND}}_{\Pi,\mathbf{A}}$

$\mathbf{Adv}^{IND}_{\Pi,A}(\lambda) \stackrel{\text{def}}{=} \Pr[b'=b]$ is negligible in the security parameter λ for any probabilistic polynomial-time (PPT) adversary algorithm A.

While for PRV-CDA an attribute-based storage system with secure deduplication Π is PRV-CPA secure if the advantage function referring to the security game $\mathrm{GAME}^{\mathrm{PRV}}_{\Pi,\mathbf{A}}$

$\mathbf{Adv}^{PRV-CDA}_{\Pi,A}(\lambda) \stackrel{\text{def}}{=} \Pr[b'=b]$ is negligible in the security parameter λ for any probabilistic polynomial-time (PPT) adversary algorithm A

The proposed system employs PRV-CDA in the private cloud while the public cloud is duly secure under IND-CPA. This is because attacks are mostly launched at the public cloud since the private cloud does not collude with users.

IV. RESULT

The deduplication system was demonstrated with a textual document, every document is unique with respect to its contents, encrypted on the bases of data attributes. The plain text and cipher text that undergone deduplication is shown in Table 4.1.

The data were encrypted using 128-bits encryption. However, increase in attribute led to increase in the duration of deduplication check. Data operation time was measured in milliseconds as shown in Table 4.2.

Table 4.1: Expected Output Key Generation

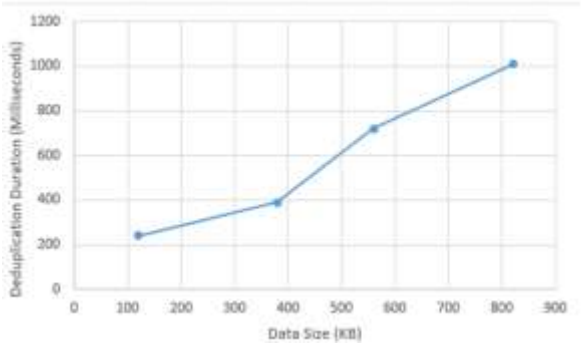
Number of Files	Contents	Encrypted Keys	Deduplication	Duplicate
1	Electronic mail (more commonly called e-mail) is the process of exchanging electronic messages between computers over a network-usually the internet. It is the most widely used internet application.	pbkdf2_sha256$150000$h-4^44m!6=hk&6g(2np57)i80_p4ws!o_8&ajvo7le9nao2r$/VzylR5rE4KNXU95T+NOg1kMsExIptH/EcOo0cFa9as=	CHECKED	NOT FOUND
2	A set of rules have to be specified according to which emails are categorized as spam or ham. A set of such rules should be created either by the use of the filter.	pbkdf2_sha256$150000$h-4^44m!6=hk&6g(2np57)-i80_p4ws!o_8&ajvo7le9nao2r$1FL4LxpxL2yPDB9yTmtJ4yfLhRwT/y1toLVCw4+zOv0=	CHECKED	NOT FOUND
3	A good spam filtering service should also be able to scan the attachments in an e-mail for viruses and malicious software. Historically, this has been one of the most often deployed methods for spreading viruses to computers.	pbkdf2_sha256$150000$h-4^44m!6=hk&6g(2np57)-i80_p4ws!o_8&ajvo7le9nao2r$kh/Gt8HriN5lFeZOU0pkoV78lrXKyxgPNZcVDlWJ5iE=	CHECKED	NOT FOUND

Table 4.2: Duration on Deduplication Check

No. Records	Data Size (KB)	Encryption Bits	Deduplication Duration (Milliseconds)	Number of Attributes
1	120	128	240	5
2	380	128	390	10
3	560	128	720	15
4	820	128	1010	20

The graphical representation of size of data against deduplication duration is depicted in Figure 4.1.

Figure 4.1



Comparison Analysis

Data storage appeared in encrypted form thereby storage space was minimized, hence, proposed and existing systems supported data privacy. The size of data was 560KB for proposed and existing systems respectively and the deduplication durations were measured in Milliseconds as depicted in Table 4.3.

Table 4.3: Proposed system versus Existing System

Metric	Balasundaram (2018)	Proposed System
Deduplication	Yes	Yes
Number of Attribute	15	15
Data Size (KB)	560	560
Time (Milliseconds)	1480	720
Data Privacy	Encrypted	Encrypted

CONCLUSION

Data Integrity, Security and Privacy has been the concern of so many data providers and cloud users. Whereas effective data management has remained the concern of cloud storage services providers. The proposed system achieves our goal by merging the technique of deduplication with the unique feature of Attribute-Based Encryption (ABE) storage. This construction provides the architecture for a Secure Deduplication of Encrypted Data over an ABE Cloud Storage which permits a company to securely save data on a public cloud, but still uphold the sensitive information pertaining to the organization's structure in a private cloud. In ABE, data are associated with attributes for each of which a public key component is defined. The encrypted associates the set of attributes to the message by encrypting it with the corresponding public key components. From the proposed ABE scheme, a hybrid cloud storage setting which comprises only a private and public cloud, has been constructed and endowed with the capability of carrying out a Secure Deduplication of Encrypted Data over an ABE Cloud Storage. Here the private cloud performs computations and hence, security and deduplication whereas the public cloud only saves files. Data Integrity, Security and Privacy has been the concern of so many data providers and cloud users. Whereas effective data management has remained the concern of cloud storage services providers. The proposed system achieves our goal by merging the technique of deduplication with the unique feature of Attribute-Based Encryption (ABE) storage. This construction provides the architecture for a Secure Deduplication of Encrypted Data over an ABE Cloud Storage which permits a company to securely save data on a public cloud, but still uphold the sensitive information pertaining to the organization's structure in a private cloud. In ABE, data are associated with attributes for each of which a public key component is defined. The encrypted associates the set of attributes to the message by encrypting it with the corresponding public key components. From the proposed ABE scheme, a hybrid cloud storage setting which comprises only a private and public cloud, has been constructed and endowed with the capability of carrying out a Secure Deduplication of Encrypted Data over an ABE Cloud Storage. Here the private cloud performs computations and hence, security and deduplication whereas the public cloud only saves files.

RECOMMENDATIONS

Several significant achievements have been made in this work to integrate secure data deduplication on an attribute-based encrypted cloud storage, but there are other features that can improve the efficiency of this system, such as searchable-ciphertext capabilities. Such capability will make deduplication possible on an already existing ABE cloud storage and not only during incoming file requests. Furthermore, while we have saved storage space by removing duplicate data and ensured data security by storing data in an encrypted format, still, we can extend this work to include image processing files and video deduplication.

ACKNOWLEDGMENT

I sincerely appreciate Dr Daniel Matthias, for his guidance, patience and intellectual knowledge which guided me in the realization of this project. I equally thamk Dr V. I. E Anireh, for his advice and co-supervision. Many thanks also to Dr E.O Bennett and Dr N.D Nwiabu my lecturers through the program.

To my beloved parents Rev. Friday Osaronwolu and Rev. Mrs. Salome A. Osaronwolu whose support are unprecedented, there are not enough words to express my gratitude. To Dr. C. S. N. Chujor a worthy role model and inspiring uncle, Princess Ochenya Ngei,

Mr. and Mrs. Oforu Peter, Mr. and Mrs. Oviemo, I am watered by your love and built by your support. Obele Saloka, Chigonum Ikegwuru, Lady Belle, Morris Joshua, you are all special and part of my motivation. Finally, my dearest sibling who I cannot botherless - Obarijimah and Salome, indeed I am thankfully lucky to share this life with you two.

REFERENCES

[1] Rabi Prasad Padhy, M. R. (July-Dec 2014). "*Cloud Computing: Security Issues and Research Challenges*". IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), Vol. 1, No. 2..

[2] Sahai, A., & Waters, B. (2005). "*Fuzzy identity-based encryption. Advances in Cryptology - EUROCRYPT 2005"*, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, Proceedings--Amos Fiat, M. N. (1993). Broadcast Encryption. *CRYPTO '93* (pp. 480-491). California: Springer-Verlag Berlin, Heidelberg.

[3] B. Zhu, K. L. (2008). "*Avoiding the disk bottleneck*". 6th USENIX Conference on File and Storage Technologies (pp. pp. 269-282). San Jose: USENIX

[4] Yao, D., & Nelly Fazio, Y. D. (2011). "*Id-based encryption for complex hierarchies with applications to forward security and broadcast encryption*". ACM Conference on Computer and Communications Security-CCS 2004.

[5] Craig Gentry, A. S. (2002). "*Hierarchical ID-Based Cryptography*." *ASIACRYPT '02* (pp. 548-566). Springer-Verlag Berlin, Heidelberg.

[6] Ran Canetti, S. H. (2003). *"A forward-secure public-key encryption scheme*". *EUROCRYPT'03* (pp. 255-271). Warsaw, Poland: Springer-Verlag Berlin, Heldelberg.

[7] Goyal, V., & O. Pandey, A. S. (2006). "*Attribute-based encryption for fine grained access control of encrypted data*". ACM Conference on computer and communication security (pp. 89-98). Alexandria: Springer.

[8] *Jenkins, A., & Fellers, J. (1986). "*An Annotated Bibliography on prototyping*." Institute for research on the management of informations, working paper., (p. W613). Indiana.

[9] Waters, Y. R. (2013). "*Practical constructions and new proof methods for large universe attribute-based encryption*". ACM SIGSAC Conference on Computer and Communications Security, CCS'13 (pp. 463-474). Germany: ACM.

[10] Lewko, A. B., & B. Waters. (2011). "*Unbounded HIBE and attribute-based. Advances in Cryptology - EUROCRYPT 2011"* 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn (pp. 547-567). Springer.

[11] M. W. Storer, K. M. (2008). "*Secure data deduplication. ACM Workshop On Storage Security And Survivability," StorageSS* (pp. 1-10). Alexandria: ACM.

[12] Diffie, W., & Hellman, M. (1976). "*New directions in cryptography*". IEEE Transactions on information Theory, 644-654.

[13] Srivastava, A. (Dec 2014). "*A Detailed Literature Review on Cloud Computing.*" Asian Journal of Technology & Management Research [ISSN: 2249 -0892], Vol. 04 - Issue: 02.

[14] Miguel Castro, B. L. (1999). "*Practical Byzantine Fault Tolerance*". Third Symposium on Operating Systems Design and Implementation. New Orleans.

[15] Bellare, M., & Keelveedhi, S. (2015). "*Interactive message-locked encryption and secure deduplication*". *Public-Key Cryptography - PKC 2015 - 18th IACR International Conference on Practice and Theory in Public-Key Cryptography* (pp. 516-538). Gaithersburg, MD, USA: Springer.