International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 9 Issue 7 ǁ 2021 ǁ PP. 74-85

Enhanced Secure Data De-Duplication Interfaced with Private Cloud as Infrastructure for Public Cloud Storage

Yigermal Semahegn Amsalu[1], Professor Seelam Sowjaniya[2]
[1]*Department of Computer and Information Technology*
Defence University, College of Engineering
Debrezeyit,Oromia,Ethiopia
[2]*Department of Computer and Information Technology*
Defence University, College of Engineering
Debrezeyit,Oromia,Ethiopia

Abstract - *Cloud computing is an Internet-based technology, that provides variety of services over Internet such as storage of data, software and hardware and also it turned in premise practice of traditional computing technology to a different approach called off-promises utilization of computing infrastructure such as storage, computer and bandwidth in pay as you go basis. This emerging technology is being adopted by varies companies. To ensure security of enterprises and users' data which is stored in cloud is in an encrypted format for which we cannot apply de-duplication technique. In this thesis work we use common storage infrastructure for enterprise-wise public data to optimize the storage size. Finally, as a proof of concept, we implemented a prototype of our proposed duplicate-check scheme and conducted test bed experiments using our prototype. For hosting purpose of this research, we use Jelastic cloud Platform that is provided by next generation java hosting which can run and scale any java application. So, to develop the application first we must have an account of Jelastic cloud platform.*

Keywords *Cloud, De-Duplication, Security*

Date of Submission: 16-07-2021 Date of acceptance: 01-08-2021

I. INTRODUCTION

In the current days large cloud computing storage service providers like Microsoft Sky drive, Amazon and Google drive storage attracts millions of users. In addition to this data redundancy was once an acceptable operational part of the backup process, the rapid growth of digital content in the data centre has pushed organizations to rethink how they approach this issue and to look for ways to optimize storage capacity utilization across the enterprise. The flud backup system [10] and goggle drive [13] etc can save on storage costs by removing duplication. This technique used to improve storage utilization and network data transfers to reduce the number of bytes that must be sent. In most organizations almost all activities are run by computers and network infrastructures within a few minutes this makes technology becomes too wonderful. So, each institution, companies, universities, colleges and other governmental and non-governmental sectors needs ICT infrastructures for their operations. Hence these entities use data storage servers in which storage spaces are duplicate data with lack of security. To efficiently use these storage servers, we need to use de-duplication process that means unique chunks of data, or byte patterns, are identified and stored with enhanced security. Generally, the global age of cloud computing becomes too famous and millions of the end users uses this many services remotely and in day today user's data also increased. It leads to the problem of ever-increasing data managements. The other problem to this stored data is access privileges which defines the access right of the stored data in cloud storages. The technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent.

Instead of keeping multiple data copies with the same content, de-duplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy. De-duplication can take place at either the file level also called as Single Instance Storage (SIS) which will remove the duplicate copy of same file [22, 23] or the block level. For file level de-duplication, it eliminates duplicate copies of the same file. De-duplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files but it needs high processing time compared to SIS.Based on the data divided/broken de-duplication classified into block level and file level but each of it has its disadvantages and advantages.

Disadvantages of block level de-duplication

- Block de-duplication requires more processing power than the file level de-duplication
- The number of identifiers that need to be processed increases greatly.
- Its index for tracking the individual chunks of each block gets also much larger

Advantages of file level de-duplication

- It requires less processing power since files' hash numbers are relatively easy to generate.
- So, the file-level de-duplication can be performed most easily
- It does not require many CPU and memory resources to implement.
- Tags (indexes) of the file are smaller. This is for short duplicate searching computation in which duplication check is conducted.
- The while reassembly check is conducted in block level de-duplication because when we come to file level de-duplication reassembly is very less since only unique files are stored.

II. MOTIVATION OF THE STUDY

The cloud computing paradigm is the next generation architecture for the business of information technology which presents to its users some huge benefits in terms of computational costs, storage costs, and bandwidth and transmission. Costs typically the cloud technology transfers all the data, databases and software's over the internet for the purpose of achieving huge cost savings for the CSP. Due to this explosive growth of digital data, there is a clear demand from CSP for more cost-effective use of their storage and network bandwidth for the data transfer purpose. Also, the use of Internet and other digital services have given rise to a digital data explosion, including those in cloud storages

III. RELATED WORKS

In this sub chapter we refer different research papers from IEEE and ACM publications and we were discussed or tried to review different literature that are written by other researcher and have an idea related with this study as we analysis these papers we are coming to our work by using gaps of the literature as a building block while literatures are used for our initial idea.

[15] Address Data de-duplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Promising as it is, an arising challenge is to perform secure de-duplication in cloud storage. Although convergent encryption technique has been extensively adopted for secure de-duplication, a critical issue of making convergent encryption practical is too efficiently and reliably manages a huge number of convergent keys. They first introduce a baseline approach in which each user holds an independent master key for encrypting the convergent keys and outsourcing them to the cloud. However, such a baseline key management scheme generates an enormous number of keys with the increasing number of users and requires users to dedicatedly protect the master keys.

[3] The authors propose de-duplication for VM which is different from the other de-duplication models' researchers considers in virtualization environment actually it is useful for efficient use of storage space but their system protects de-duplication replacing from the old data not from the new entry.

[20] The authors use the technique of map reduce special feature of map reduce function to automatically partition the computing job according to the security level of the data. So, the computation of data is on private cloud the computation of non-sensitive data is done on the public cloud. The authors use hybrid cloud architecture they only consider data security they don't consider data duplication in the public cloud.

[35] Cloud storage systems are becoming increasingly popular. A promising technology that keeps their cost down is de-duplication, which stores only a single copy of repeating data. The outers used the technique of client-side De-duplication attempts to identify de-duplication opportunities already at the client and save the Bandwidth of uploading copies of existing files to the server. In this work we identify the drawback is that exploit client-side de-duplication, allowing an attacker to gain access to arbitrary-size files of other users based on very small hash signatures of these files. More specifically, an attacker who knows the hash signature of a file can convince the storage server that it owns that file.

[23] Cloud storage service providers such as Drop box, Mozy, and others perform de-duplication to save space by only storing one copy of each file uploaded. The authors for DupLESS Server-Aided Encryption for De-duplicated Storage attempts to solve space saving and security and they address cross user de-duplication clients encrypt under message-based keys obtained from a key-server via an oblivious PRF protocol. The drawback of this work doesn't consider clients privilege and it only supports simple storage

interface. At the same time the author doesn't consider channel privacy leakage in cross user de-duplication which reveals information of the file.

[33] The authors of ClouDedup: Secure De-duplication with Encrypted Data for Cloud Storage this paper uses the technique of convergent encryption to achieve confidentiality while de-duplication feasible in block level de-duplication here we identify two draw backs effects one is efficient key management for the ever-increasing file and processing time for each block of data while it searches duplicate file takes long time

[27] The authors' of this paper suggests de-duplication as a potential application of their incremental deterministic public-key encryption scheme. But this will only work with a single client. It won't allow de-duplication across clients, since they would all have to share the secret key.

[43] This paper uses fault Tolerant digital signature Scheme to improve the speed of data de-duplication and data integrity for the outsourced data. The data de-duplication strategy used in proposed scheme is the fixed-sized blocks and block-level de-duplication the main drawback of this paper is the worst case in that cloud storage server will regard all blocks as a new blocks and store all of these blocks, resulting in storing duplicate blocks.

[28] The authors of this paper mainly focus on to optimize the private cloud storage backup in order to provide high throughput to the users of the organization by increasing the de-duplication efficiency. The main limitation of the paper is the concern is given for de-duplication not for the security as well.

III. METHODOLOGY

Secure Hash Algorithm (SHA) was developed by NIST along with NSA [31] in 1993; SHA was published as a FIPS. The SHA is called secure because it is designed to be computationally infeasible to find two different messages which produce the same message digest. Any change to a message in transit will result in a different message digest, and the signature will fail to verify. Secure Hash Algorithm (SHA) is necessary to ensure the security of the Digital Signature Algorithm (DSA).It takes a message of any length $<2^{64}$ bits as input and produces a 160-bit message digest as output. The message digest is then input to the DSA, which computes the signature for the message. Signing the message digest rather than the message often improves the efficiency of the process, because the message digest is usually much smaller than the message.

Advanced Encryption Standard

The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology [29].AES is based on the Rijndael cipher developed by two Belgian cryptographers, Joan Daemenand Vincent Rijmen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes

Hashed Message Authentication Code

HMAC treats the hash function as a black box. This hash two benefits. First, an existing implementation of a hash function can be used as a module in implementing HMAC. The bulk of the HMAC code is pre-packaged and ready to use without modification. Second, to replace a given hash function in an HMAC implementation, we must simply remove the existing hash function module and drop in the new module. This could be done if a faster hash function were desired. More important, if the security of the embedded hash function were compromised, the security of HMAC could be retained simply by replacing the embedded hash function with a more secure one (replacing with SHA-1)

Proof of Ownership Protocol

Proof of ownership is the protocol in which it enables to protect uploading the same content of data in the storage while it proves the ownership to the storage server to avoid duplicate copies. Here this is run in between the user and the storage server (prove, verify). When we say prove the user should have to pass his/her identity to the server. A short value of token $\emptyset(M)$ is derived by the verifier from the data copy M to prove the owner ship of the M. Hence the token $\emptyset$ is the tag of the M and privileges of the user. Generally POW is the protocol where the client proves the server that it has the original file M. Proofs of ownership (PoW) for de-duplication systems, such that a client can efficiently prove to the storage server that he/she owns a file without uploading the file itself [26] Proposed an alternate PoW plan by selecting the projection of a record onto some randomly chosen bit-positions as the record verification

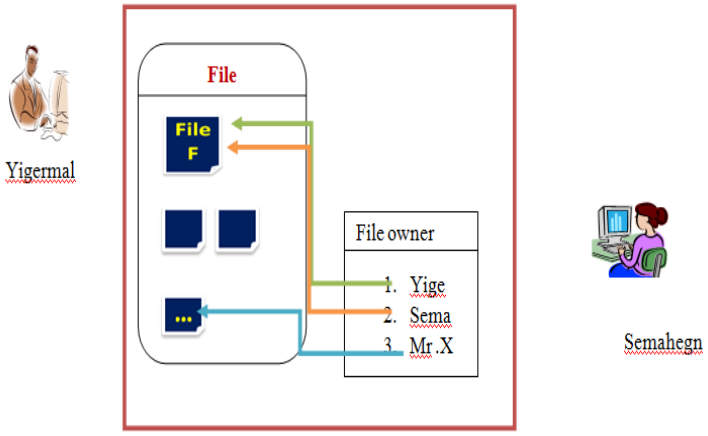


Figure 3-5 Proof of Ownership Protocol

In the above figure both Yigermal and Semahegn have the File F. if they want to upload to the storage server two of them need not to upload. Semahegn only shares the privileges of yigermal rather than to upload the same content of the file. This means that while yigermal is the owner of the file semahegn linked to the file or takes pointer of the file F.

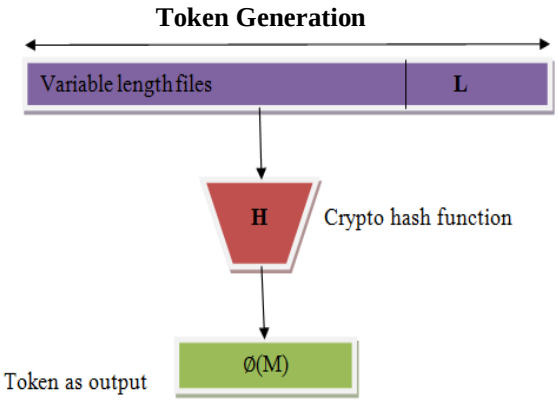


Figure 3-6 Token Generations

IV. SELECTION OF CLOUD PLATFORM AND SERVICE PROVIDER

This chapter discussed all about the platforms and the service providers that are selected in this study and in this case this we look over more 6 service provider and from this the research is most discussed about jelastic public cloud platform because in this research we use this platform as PaaS environment

Analysis and Comparison of Cloud Application Development and Hosting Platforms

In this research we select platform as a service provider by comparing the different service such as software as a service, dedicated IT, hosting provider and infrastructure as a service by the following figure we can consider the difference and we can compare them and we can select one of the best services that is suitable for this research.

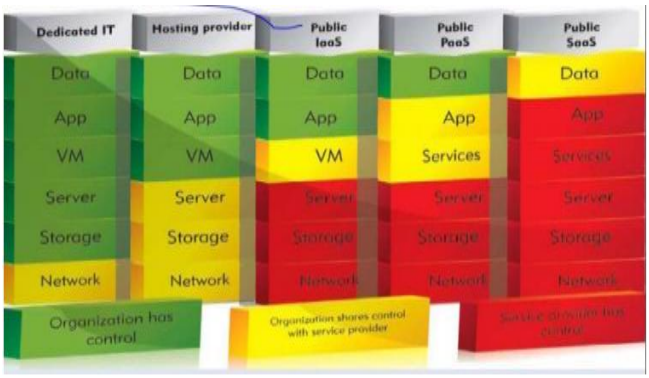


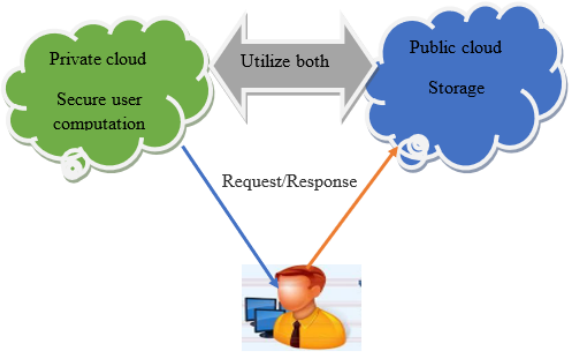
Figure 1-1 Comparison among cloud services
Jelastic Public Cloud Platform

Jelastic is a cloud services provider that combines PaaS (Platform as a Service) and CaaS (Container as a Service) in a single package for hosting providers, telecommunication companies, enterprises and developers. Jelastic provides support of Java, PHP, Ruby, Node.js, Python, .NET, Go environments and custom docker containers.The deployment can be easily performed using GIT/SVN with automatic updates, archives (zip, war, ear) right from the dev panel or via integrated plugins like Maven, Eclipse, NetBeans, IntelliJ IDEA.

V. THE PROPOSED DEDUPLICATION ARCHITECTURE

This chapter describes the architecture of the proposed de-duplication system.

At a high level, our setting of interest is an enterprise network, consisting of a group of a listed clients (for example, employees of a company) who will use the S-CSP and store data with de-duplication technique. In this setting, de-duplication can be frequently used in these settings for data backup and disaster recovery applications while greatly reducing storage space. In our thesis we have to follow the hybrid cloud data model for our prototype development and system architecture.



Now a day's cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. These service providers provide services by hiding implementation and platform at the same time so by using virtualization it is possible to provide unlimited services. To solve storage problem as well as privacy we try to solve using de-duplication with enhanced data security. Although data de-duplication brings a lot of benefits, security and privacy concerns arise as users' sensitive data are susceptible to both inside and outside attacks. For saving resources consumption in network bandwidth and storage capacities, many cloud services, namely Drop box apply client-side de-duplication. Our assumption is by taking large enterprises which uses common storage service. Example college systems which consist of many Departments like Administration, Examination Control, information Technology, Electronics etc and students too. All the repeated data of these departments are stored in cloud thus occupying a huge storage space.

File Uploading Algorithm

1. User should register first
2. User should Login to enter into the system.

3. User selects a file **F**
4. Compute tag of **F(ØF=TagGen(F))** request token of **F** to private server and private server module resent to user token of **F({(Ø'F =TagGen(ØF, k_p))**

Where **ØF**=File token,**k_p**= privilege key

5. User requests token of **F**to public cloud for de-duplication and public cloud verify token weather it exist or not and sent the result to the user finally public cloud module verifies.
6. If user gets not duplicate it proceeds to upload else server runs **POW**.
7. While uploading the user performs :

a) User gets key from private cloud **K_F** ,**K_F** =key of the file

b) **CF**=Encrypt(**F**, **k_F**)

c) While receiving **CF** , the private cloud server module generates token of the file **(ØF)** and privilege keys of the file owner **(PK)** for final uploading to the public cloud. Hence, **Ø**Fis uploaded to the public while **k_F and P_K** is stored in the database.

8. End.

VI. IMPLEMENTATION

Cloud User Model

A user is an entity that wants to outsource data to the S-CSP and access the data later.Therefore a user first registers to the system and gets his/her username and password. After a user is authenticated by the system user logged into the user page and selects a file to generate hash value of the file. This hash value of the file is used for computation of the file token.

Figure 6-4 Hash Computation Value of the file

As we have seen from fig 6-3 a file is first computed using secure hash algorithm. This file tag is send to the private cloud module. In private cloud module there is hash based message authentication code (HMAC) to compute the file token (**ØF).**

Private Cloud Model

It is honest and credible server which computes secure operations of privilege keys and file tokens are computed by the private cloud module. For computation of file token user computes hash of the file and sends to the private cloud. This file token is used to prevent duplication in public cloud storage.

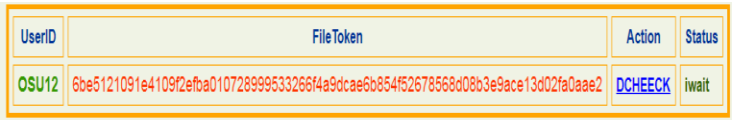


Figure 6-5 Token of the file

The user clicks DCHEEK link. If the file is not yet stored in the public cloud the file is uploaded to cloud but if the file is already stored in the cloud the server runs proof of ownership to the stored file.

Figure 6-6Proof of Ownership

The second user shares the privilege keys of the first user to have the ownership of the file. Hence if there are any subsequent users of this file the content of the file is not yet uploaded. The subsequent user's shares reference of the file attribute from the first user.

Figure 6-7 File attributes shared by Subsequent users

In the above figure 6-5 the first user file id is 2 and the owner of the file is OSU12 which is shared by the subsequent user. The subsequent user file id is 211 and file name is de.pdf. Finally these subsequent user uploads file attributes rather than the file content.

Public Cloud Model

In this module, we develop Cloud Service Provider module as storage of encrypted metadata files in a public cloud module. This is an entity that provides data storage service in public cloud. The S-CSP provides the data stores encrypted data on behalf of the users. To reduce the storage cost, the S-CSP eliminates redundant data via De-duplication and keeps only unique data.The following fig shows the proto type of the public cloud model in which it stores only unique files.

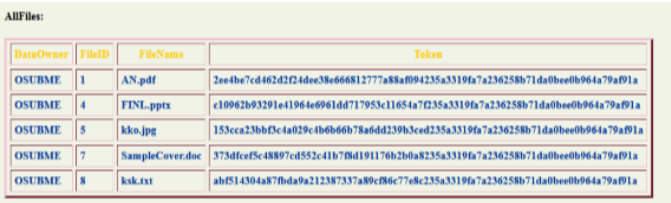


Figure 6-8 Public Cloud Model

VII. RESULTS AND DISCCUSIONS

The data used for this experiment was collected from the data center of the Oromia State University. The university comprises of the following departments each department have their own data in the data center which uses samba file server:

- Information Technology
- Law
- Accounting and Public Finance
- Economics
- Business and Information System
- Human Resource and Leadership

- Management
- English

Communication Medium
The server and client machine are connected via category 6 UTP cable. The reported data rate of the local area network is 90 Mbps.

Experiment Parameters
From here we see the following parameters:-
1. The download parameter
2. The upload parameter
3. Time parameter
4. Network usage and server load

The download parameter was deliberately ignored as it cannot be optimized. Any user who later wants get his file must download it from the server at a rate of the network speed. There is no other way out. Uploading however can be optimized
The department of IT with respect to the remaining department was analyzed and the following results were found.
The total size of these files is 16 GB out of which only 12.5GBis unique to the department. The remaining 3.5 GB was duplicate to the departments. As a result3.5 GB of disk space was wasted.

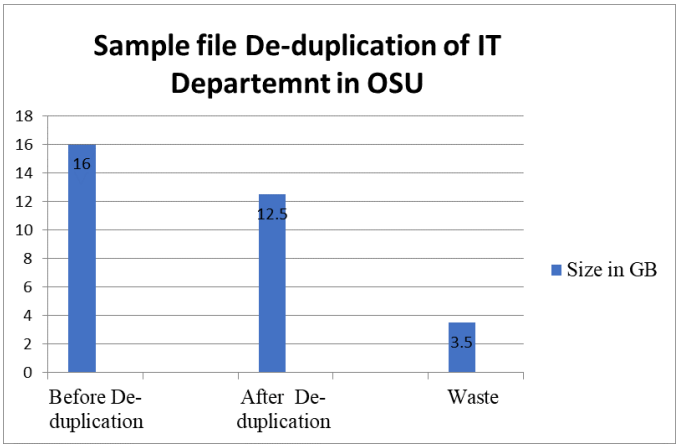


Figure 7-2 file duplication in IT Department

As it can be seen from figure 7-2 12.5 GB of which is 21.88 % (3.5*100/16) of the optimum size has been wasted. Similarly, analysing files of all departments at the same time has produced the following results.

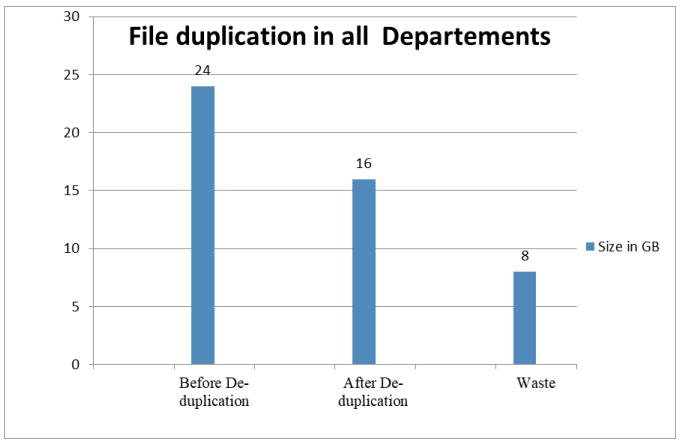


Figure 7-3 File duplication across all departments

Overall, 33.3% of duplication was found across all departments. And the results show that applying de-duplication could have saved 8 GB of disk space. No more explanation is required here to convince the advantages of de-duplication with regard to storage efficiency. If we look at the case large scale servers for example cloud storage servers, the size of disk space consumed and the logistics required to run the service is enormous. Imagine the benefits of applying de-duplication in a world where 75% of its digital data are duplicate copies [16]. The next advantage of file de-duplication is reducing the network traffic, reduced transmission time and reducing the server load.

File Size(KB)	Elapsed Time in Milliseconds		Ratio First Upload/Next Upload
	First Upload	Next Upload	
2MB	65.5	11.5	5.7
3.5MB	136.8	13.0	10.5
8MB	220.0	13.0	16.9
10MB	258.9	22	11.8

7.3.1 Network usage and Server load Parameter

As it can be seen from the table above, elapsed time of the first upload increases as the file increases. The time needed to complete next uploads shows very little variation as the size of the file increases.

The difference in elapsed time between the first upload and subsequent uploads can be clearly shown in the following graph.

As it can be seen from Table 7-2, it is easy to conclude that network traffic can be minimized if de-duplication is applied. For example, let us see the case of 10 Mb file, it has taken 258.9 milliseconds to complete the upload. However since the next upload request needs only to provide proof, it has taken only 11.8 milliseconds. Hence it can be concluded that applying de-duplication improves storage efficiency, minimizes server load and increases network band width.

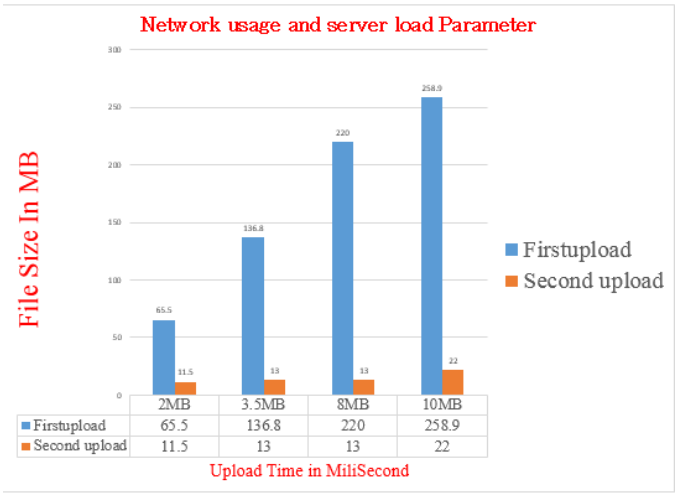


Figure 7-4 Elapsed time of first upload vs subsequent uploads

VIII. CONCLUSION, RECOMMENDATION AND FUTURE WORK

In this paper we studied that security and efficient use of storage space. De-duplication technique is very important for cloud storage service providers to manage the ever increasing user's data and data storage servers. Duplication of the files is identified by file tokens. This is secure for external and internal attacks. Hence authorized user uses file tokens for duplication check with files stored in public cloud module. We use private cloud module for secure data operation while the public cloud module uses as S-CSP to store encrypted data storage and used for to protect redundant file which have different file name but have the same contents. Generally as the demand of data stored increases in the cloud data de-duplication is one of the techniques used to improve storage efficiency.

Recommendations

Now a day's most organizations, institutions, colleges and universities have their own data at data centers. Example in real scenario we had tried to observe the Oromia State Universities data center. The university has department data in the file server (samba file server). Each instructor uploads many repeated

data in this file server. If we apply this work in real scenario we will have different advantages for ICT administrators, for users and for the university.

For ICT administrators they will get good knowledge in security and efficient use of data servers. For users it lets secure de-duplication operation and technology transfer and it creates creation awareness in resource management. For the university it saves extra coast encourage having additional file servers and increases the confidence in security. Finally small change in file content would result in a completely different hash value. Hence, identifying duplicates using a hash function is not applicable. Further research is required that addresses this issue.

Future works

The proposed system can be extended further, as a future work for successful implementation and deployment of this work in real time scenario like large industries data center & institutions with additional proto type functionalities

A. Preparing Your Paper

1) Paper Size: Prepare your paper in full-size format on US letter size paper (8.5 by 11 inches).

2) Type Sizes and Typefaces: Follow the font type sizes specified in Table I. The font type sizes are given in points, same as in the MS Word font size points. Times New Roman is the preferred font.

3) Paper Margins: Paper margins on the US letter size paper are set as follows: top = 0.75 inches, bottom = 1 inch, side = 0.625 inches. Each column measures 3.5 inches wide, with a 0.25-inch gap between the two columns.

4) Paper Styles: Left- and right-justify the columns. On the last page of your paper, adjust the lengths of the columns so that they are equal. Use automatic hyphenation and check spelling and grammar. Use high resolution (300dpi or above) figures, plots, drawings and photos for best printing result.

TABLE I
TYPE SIZE FOR PAPERS

Type size (pts.)	Appearance		
	Regular	Bold	Italic
6	Table superscripts		
8	Section titles[a], references, tables, table names[a], table captions, figure captions, footnotes, text subscripts, and superscripts		
9		Abstract, Index Terms	
10	Authors' affiliations, main text, equations, first letter in section titles[a]		Subheading
11	Authors' names		
22	Paper title		

[a]Uppercase

B. Preparing Your PDF Paper for IEEE Xplore©

Detailed instructions on how to prepare PDF files of your papers for IEEE *Xplore*© can be found at

http://www.ieee.org/pubs/confpubcenter

PDF job setting files for Acrobat versions 4, 5 and 6 can be found for downloading from the above webpage as well. The instructions for preparing PDF papers for IEEE *Xplore*© must be strictly followed.

II. HELPFUL HINTS

A. Figures and Tables

Try to position figures and tables at the tops and bottoms of columns and avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be centered below the figures; table captions should be centered above. Avoid placing figures and tables before their first mention in the text. Use the abbreviation "Fig. #," even at the beginning of a sentence.

Figure axis labels are often a source of confusion. Use words rather than symbols. For example, as shown in Fig. 1, write "Magnetization," or "Magnetization (M)" not just "M." Put units in parentheses. Do not label axes only with units. In the example, write "Magnetization (A/m)" or "Magnetization (A□m^{-1})." Do not label axes with a ratio of quantities and units. For example, write "Temperature (K)," not "Temperature/K."

Multipliers can be very confusing. Write "Magnetization (kA/m)" or "Magnetization (10^3 A/m)." Figure labels should be legible, at 8-point type.

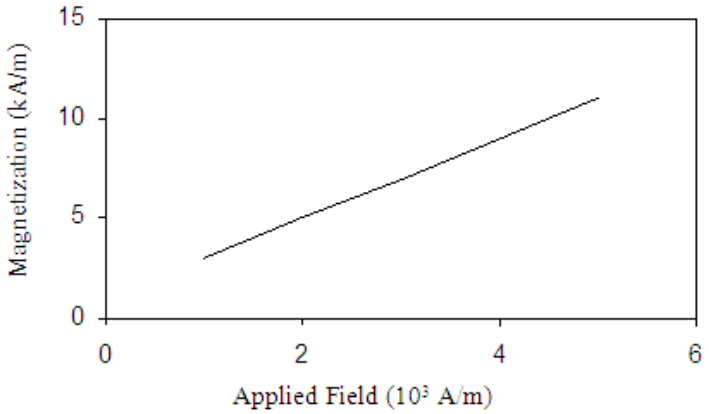


Fig. 1 Magnetization as a function of applied field.
Note how the caption is centered in the column.

B. References

Number citations consecutively in square brackets [1]. Punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]. Use "Ref. [3]" or "Reference [3]" at the beginning of a sentence: "Reference [3] was the first …"

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the reference list. Use letters for table footnotes (see Table I). *IEEE Transactions* no longer use a journal prefix before the volume number. For example, use "*IEEE Trans. Magn.*, vol. 25," not "vol. MAG-25."

Give all authors' names; use "et al." if there are six authors or more [4]. Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. In a paper title, capitalize the first word and all other words except for conjunctions, prepositions less than seven letters, and prepositional phrases.

For papers published in translated journals, first give the English citation, then the original foreign-language one [6].

C. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even if they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title unless they are unavoidable.

D. Equations

Number equations consecutively with equation numbers in parentheses flush with the right margin, as in (1). To make your equations more compact, you may use the solidus (/) and the exp function, etc. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use an en dash (–) rather than a hyphen for a minus sign. Use parentheses to avoid ambiguities in denominators. Punctuate equations with commas or periods when they are part of a sentence, as in

$$\frac{e^{ix}}{2} = \frac{\cos\ x + i \sin\ x}{2} \Rightarrow \exp(\ ix\ ) / 2 = (\cos\ x + i \sin\ x) / 2\ . \quad (1)$$

Symbols in your equation should be defined before the equation appears or immediately following. Cite equations using "(1)," not Eq. (1)" or "equation (1)," except at the beginning of a sentence: "Equation (1) is …"

E. Other Recommendations

The Roman numerals used to number the section headings are optional. Do not number ACKNOWLEDGEMENT and REFERENCES and begin Subheadings with letters. Use two spaces after periods (full stops). Hyphenate complex modifiers: "zero-field-cooled magnetization." Avoid dangling participles, such as, "Using (1), the

potential was calculated." Write instead, "The potential was calculated using (1)," or "Using (1), we calculated the potential."
Use a zero before decimal points: "0.25," not ".25." Use "cm^3," not "cc." Do not mix complete spellings and abbreviations of units: "Wb/m^2" or "webers per square meter," not "webers/m^2." Spell units when they appear in text: "…a few henries," not "…a few H." If your native language is not English, try to get a native English-speaking colleague to proofread your paper. Do not add page numbers.

III. UNITS

Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as "3.5-inch disk drive."
Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.

IV. SOME COMMON MISTAKES

The word "data" is plural, not singular. In American English, periods and commas are within quotation marks, like "this period." A parenthetical statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.) A graph within a graph is an "inset," not an "insert." The word alternatively is preferred to the word "alternately" (unless you mean something that alternates). Do not use the word "essentially" to mean "approximately" or "effectively." Be aware of the different meanings of the homophones "affect" and "effect," "complement" and "compliment," "discreet" and "discrete," "principal" and "principle." Do not confuse "imply" and "infer." The prefix "non" is not a word; it should be joined to the word it modifies, usually without a hyphen. There is no period after the "et" in the Latin abbreviation "et al." The abbreviation "i.e." means "that is," and the abbreviation "e.g." means "for example." An excellent style manual for science writers is [7].

ACKNOWLEDGMENT

The preferred spelling of the word "acknowledgment" in America is without an "e" after the "g." Try to avoid the stilted expression, "One of us (R. B. G.) thanks …" Instead, try "R.B.G. thanks …" Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

[1]. M. King, B. Zhu, and S. Tang, "Optimal path planning," *Mobile Robots*, vol. 8, no. 2, pp. 520-531, March 2001.
[2]. H. Simpson, *Dumb Robots*, 3rd ed., Springfield: UOS Press, 2004, pp.6-9.
[3]. M. King and B. Zhu, "Gaming strategies," in Path Planning to the West, vol. II, S. Tang and M. King, Eds. Xian: Jiaoda Press, 1998, pp. 158-176.
[4]. B. Simpson, et al, "Title of paper goes here if known," unpublished.
[5]. J.-G. Lu, "Title of paper with only the first word capitalized," *J. Name Stand. Abbrev.*, in press.
[6]. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Translated J. Magn. Japan*, vol. 2, pp. 740-741, August 1987 [*Digest 9th Annual Conf. Magnetics Japan*, p. 301, 1982].
[7]. M. Young, *The Technical Writer's Handbook*, Mill Valley, CA: University Science, 1989.