Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

Design and Implementation of Overlay File System for Cloud-Assisted Mobile Applications

[1]Vijayaganth V, [2]Bhavadharini S, [3]Deepti Sharma, [4]Dharini K S

[1]Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore-641407, India.
[2]Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore-641407, India
[3]Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore-641407, India
[4]Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore-641407, India

Abstract:
The mobile apps that require more resources to perform computation tasks will be offloaded to the cloud. So this leads to the situation where the task on both the cloud and mobile apps at the same time. Here lies a main challenge which has to confirm the process has to share and also access the loaded file on both mobile and cloud. The cloud in exceptional manner which should be in an efficient manner and also stable. Distrib-uted file system which exist and network file system does not convince these requirements. At present the used system for system that is made simple does not support file exist for offloading process or does not offload process while existing the particular file. This issue is man-aged by the paper with associate degreed which implements and appli-cation-extend filling system which refers to Overlay File System. In order to boost powers, original copies of establishment is maintained and buffered by OFS and placed on each of the cloud and mobile device. The consistency is ensured by OFS and assured that whoever gets to read will gain the modern knowledge. Unified read of the information which is independent of location and accessible at original storage is created by OFS in order to ensure the transparency of the location. Mobile systems special option on final level filling system may cause challenges which must be overcome. OFS can support resource demanding tasks and pre-vent data loss. The testing is done by the paper for OFS image on robot OS with a real applications of mobile. OFS will support accessibility of file with proper user authentication.

KEYWORDS: operating system; cloud computing; file organisation; mo-bile application.

I. INTRODUCTION

Cloud computing is the technology which helps in solving the problem with compute and storage capacity. It provides us the ability to share different types of resources such as compute, storage, networks, mi-gration over internet [2]. The cloud services are available globally in different regions which makes it easier to use from any remote loca-tion Cloud computing has variety of computing resources available which are more flexible and cost-efficient. The security of data is also added advantage. The cloud computing has made reforms in the usage and management of resources and technology. However, the ad-vantages of affordable, negligible management and bigger flexibility go with exaggerated security issues. The cloud provides elasticity where one can scale up or down the resources when need and can overcome the problem of network traffic [3], [4].Hence we

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

can come to a fact that security and good performance are the two most im-portant aspects for large-scale systems. So, in this paper we approach the issue of performance and security through secure data replication problem. We start with detaching and reproducing the user data in the cloud for better security and performance that fragments the files pro-vided by the user into pieces and replicates them. Only a partial amount of data is stored in the nodes which prevent the attacker from getting access to some meaning full information. The node separation is done by FS -Algorithm. In-order to improve data retrieval the selec-tion of nodes are done in a manner that they are not adjacent. This method requires more memory since the data is not placed in a se-quential order. To improve the retrieval time of data the fragments are replicated which leads to highest read/write heads [10]. The fragments are also partially stored in the nodes of the virtual servers.

II. LITERATURE SURVEY

1. Security issues analysis for cloud computing

Cloud Computing may be a versatile, efficient, and well-tried delivery platform for providing business or client IT services over the web. Cloud Computing presents an additional level of risk as a result of es-sential services area unit often outsourced to a 3rd party, that makes it more durable to keep up information security and privacy problems, upcoming separations, and service allocations, and demonstrate com-pliance ability to adapt to fluctuations according to the needs, enhance the production work, and provides essentials for value decrease in considerations and economical way of proceeding.

2. Improved vigorous credential generation scheme for securi-ty of user identification in mobile-cloud computing

The best way to proceed the resource restriction of mobile devices, mobile users could make use of the cloud process and storage ser-vices. The way in which the user has to proceed in securing the stor-age matters the availability of resources. This is the main advantage for the user to store files in a secured manner. The enhancement paves a way for success of this storage over the clouds. User is ignorant concerning adversary's spiteful activities. Important improvement in work is time and energy consumption.

III. EXISTING MODEL

When it comes to offloading resource demanding computational tasks the file systems that are existing now are not much effective. Applications that run on hosts in wireless networks must cope with constraints on access to data that are generally not present in wired network [15]. Network file system like NFS, solely support remote accessing of file from the platforms where their shopper code is properly established. However, putting in place the patron code typically desires root privilege that the users won't be having. This also wishes for the credentials from the user to access the information processing system that the user won't be willing to unleash to the cloud. Consistency is one of the main problem in the ex-isting file systems. The data moved to the cloud should be kept secured. Unau-thorized access to the data by attackers should be prevented. So, a week entity can move the whole cloud to risk. So, in such scenario, the security provided to the

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

cloud should effectively increase the effort of the attacker to retrieve a mean-ingful amount of data after an intrusion.

As associate degree example, to make sure of proper working of the application in both medium (cloud and mobile with fair consistency. Moreover, the existing FS cannot guarantee such consistency. At times the system can crash as a result of inconsistency.

IV. PROPOSED MODEL

To address these issues, we have a tendency to propose Associate in file system called Overlay file system (OFS).By default, OFS ensures that every task whether or not in mobile or cloud scan the most recent knowledge in the file First, the proper working of computational tasks in mobile device is ensured by robust con-sistency and therefore the cloud. Second, files that are accessing the tasks will be free to move across different devices. Among them are coordinating shared data across multiple devices and servers, offloading code from devices to the cloud, and integrating heterogeneous components with vastly different software stacks and hardware resources [1] It could be as result of file state and operations are within the applications user house, and so will be replicated and along with the tasks can be moved to different locations of the virtual servers.

The work of Application development and management of the system is modified in the application layer. In contrast to standard file sys-tems, that area unit a part of software package, Overlay File System operates at the appliance level [5]. For instance, with Overlay File Sys-tem, to set up the system the root privilege is not needed. First of all, we should study the filesystem and understand the requirements to support offloading some tasks on to the cloud. Secondly, the tendency to style and implement OFS as an answer to fulfil these needs. To im-prove compatibility and scale back deployment efforts, we have a tendency to separate the working from certain task in the systems [6]. Whenever cloud assisted mobile app is launched it is notified to the offloading service or once Associate in Nursing offloaded task is on the brink of be migrated back to the mobile device.

Once the notification is received, the service that is offloading resources asks the Overlay file system middleware to enhance the state of the system and there-fore the connected applications threads to update application-specific states. The service in the cloud is given information concerning the migration with infor-mation.

The thread that is liable for the offloaded object is contacted by the offloading task, specified the injected code is re- established in the thread. The file sessions are opened, the file pointers are moved to a new location. Then, the OFS middle-ware concerning the offloaded object, specified I/O in the offloaded thread is served by the Overlay file system middleware. Such kind of interactions provokes multiple overhead that is enclosed within the result of performance.

OFS provides excellent consistency and make sure that each one who reads is provided with the newest information. The write-invalidate and write-update poli-cies are combined in-order to scale back the network traffic and to cut back the delay in execution once the newest information is not accessible remotely.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

V. SYSTEM ARCHITECTURE

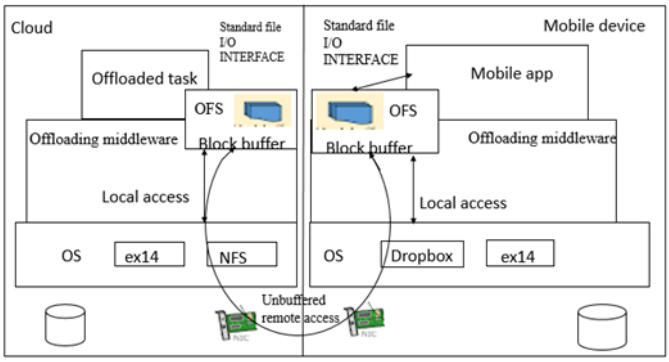

Fig 1. Architecture overview

OFS can act as a part of the system that can manage the assigned pro-cess. Overlay File System will be the file accessing platform by remote access that will issue the tasks to applications and the virtual server will provide the information available as original information. OFS will not create issue in terms of accessibility as there is no root privi-lege. In OFS, the file stored is actually retrieved from the storage and it is split into fragments and stored in multiple cloud servers. The frag-ments are stored in the virtual servers and encrypted to provide se-cured cloud storage. OFS is providing easy and simple access to the files and also secured storage is possible by implementing storage of fragments in multiple cloud servers. File need to be stored in OFS is split into multiple fragments and these fragments are stored partially into the virtual servers and while retrieving the available clouds will provide the complete information. OFS file system is managing the file storage in cloud servers and also improving the security of storage by storing the fragments in different order and not in adjacent order. The file need to be stored in multiple cloud storage is completely se-cured and it is more effective in performance. Thus, if the requested data is absent in local memory, a page-fault handler will retrieve the page either from the local memory of another cluster or from disk [11]. The virtual servers are completely secured as fragments in nodes are stored in different locations and in partial manner.

The accessibility of the file stored in the cloud is possible only if the user is authenticated and buffer is maintained by OFS and it can done with local access. The device can make use of OFS to prevent data loss as the fragments of the original file is duplicated and stored in vir-tual storage. Based on the usage, the cloud servers will provide the file in sequential order as original file is stored. The sharing of the file is also managed by OFS and it is done by sharing the key between the user and the receiver. The node that is being attacked will not be the reason for retrieval of complete information because the nodes of servers will contain only half of the information and also not in se-quential order. OFS will carry out the entire process of storing the file and also the offloading tasks. It is evident that the OFS will perform the task of storing file in servers in different nodes of servers and provide easy access to the file and original document is retrieved with authentication of the user. OFS is the file system that manages buffer for easy retrieval and effective performance.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

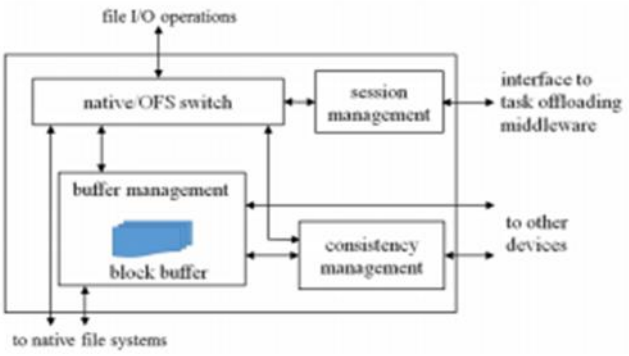

Fig 2. Overlay Filesystem

OFS will play a major role in progressing the process. The file storage in the most secured way is being made possible here. The OFS will send the fragments of the file to the nodes and they are stored in dif-ferent locations. OFS will communicate with other system components and do transfer the data based on requirements. OFS is actually in the constructed in a manner to perform easy file access and data sharing to the authenticated user. The functionality of overlay file system is designed in such a way to provide secure cloud storage for files. And it is majorly used to share files to verified user. The code is generated and it is shared between sender and receiver. The main goal of OFS is to enhance the performance, secured sharing and storage.

The objective of OFS is to share file to the verified user. The file to be accessed is made fragments and they are stored in multiple cloud servers. These virtual servers are monitored for accessibility. The available servers are updated in the hash tables. The user requesting for the file must provide credentials and this request is forwarded to the log server and the requested user is verified. The request is accept-ed when the user is authenticated and the cloud servers available will provide the file in original format. Above figure illustrates the main components of OFS.

The OFS will provide the platform for the user to implement secured and safe file storage without any loss of data while storing. Another main role of OFS switch is to maintain and manage the way of storing the file into fragments within the clouds and to denote the user while retrieving. Management of the block buffer is handled by buffer man-agement. For better knowledge, mapping table is maintained for each file and is saved in the file separately and status of the storage is main-tained in mapping table. LRU-like algorithm is used to expel blocks in order to maintain the storage of the data in the proper order, which is selected by the user while storing and also the count of the fragments. To enhance the speed and to make the process of storing in simple manner, block buffer is created by us in virtual address space. Next, session management component handles the sessions of file and also improve the consistency while retrieving the file in sequential order.

When a process is being proceeded, notification is made in adjacent side component also, it copies the contents of the file in order like cur-rent occurrence in every file and also its sub-parts in the cloud storage. Running programs require consistency guarantee which is provided by consistency management .For this purpose, access to the shared files and blocks cached in block buffer is monitored by consistency man-agement.

VI. MODULE DESCRIPTION

6.1 User Authentication

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

User must be authenticated which is meant to be the method of identi-fication you're attempting to assure a user that they are saying about them. If the user wants to access the file, a user has to offer the cre-dentials that will be forwarded to the log server for identification. The user credentials are authenticated and the way it gets proceeded is Login Credentials. Various constructor's area unit in usage and the in-stance uses the Login data provided. The primary data for verification is that the name, and also the second information could be a request handler used for sending the concerned user request data to the Log server for authentication of the user with provided details. Request Handler contains an important technique that provides the login cre-dentials to the Log server. The instance uses an absolutely easy han-dler that saves the username considered with watchword in an instance variable, in order that it are always transferred on throughout the invo-cation of the handle technique from the Log server. It is actually at-tainable to form call backs that will be provided with the login creden-tials, and that data is verified by the log sever.

6.2 Fragmentation

In this Module, the file need to be uploaded in cloud is divided into different fragments. The file that has to be stored is then completely divided into fragments and are stored in nodes of the multiple cloud servers. And the fragments are partially stored in the servers. The in-formation cannot be retrieved because the fragments are not located adjacent. The fragments are placed in random order to prevent from attack. The fragments are duplicated in order to prevent the loss of in-formation. At first, we tend to focus on the improved privacy and se-curity. As declared higher, the fragments need to prevented from loss of content and also to improve privacy.

6.3 Data Replication & encryption.

The data is replicated to improve secured data storage in the virtual servers. It also manages file sharing by generation of key. The data in file is replicated and are partially stored in the virtual server. The data fragmented is stored in the nodes. These nodes containing the individ-ual fragments are stores in multiple cloud servers. Every reproduction inside is actually unambiguously established, and of conditions that has got to be simple in a desired way where the duplication of data in accordance to the shopper understand the duplication process to pre-vent any issue with loss of data of required file. The fragments are replicated in order to prevent the cause where the loss is data happens. With the scope of future this actions are required.

The fragments are included in nodes of virtual servers. These servers must contain the nodes at different locations and not in adjacent posi-tions. These are located at different locations to prevent the unauthor-ised access of data which is at high risk. User uploaded information is completely encrypted for secure information storage in cloud. En-crypted information area unit hold on in numerous virtual server with fragments. During this module, we tend to area unit shuffling the cloned fragments by the Fs-OFS algorithmic program. By victimization this algorithmic program, cloned fragments area unit shuffled like (1-4, 2-1, 3-2, 4-3).Once the user requested for the data, it'll retrieve the mandatory fragments within the ordered order. Once all the fragments area unit collected, can manufacture a complete data to the user.

6.4 Cloud Server Analysis

The process must have the correct occurrence list as a service requires the service supplier which appreciates importance of the security in storing files properly with the client's applications which deployed in virtual machine instances in the way that can get server standing in the order with fragments order. To the current aim, we look forward to outline because the basic module which

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

applies a loss of data and data compromise that leads to the failure at the roughness of a server's in-stance. It's observed that the impact of the fact will be considered in these servers are often obtained by applying server computing mecha-nisms straight to the cloud storage than applying itself, that's how the errors gets predicted for example, analysis of server of banking ser-vice are often included by replication of the complete VM instance during which application tier gets deployed on various occurrences, and crashes of server are often detected in the proper way as that of the failures in accordance with the recognitions like the data compro-mise. Wherever the first and backup parts area unit run in VM in-stances of the banking service's considerations.

The planning stage begins once shopper appeals the service supplier to supply server analysis support to its applications. During this phase, the service supplier should 1st analyse the client's needs, contest them with accessible, and kind a whole server analysis resolution victimiza-tion acceptable. We tend to note that every that every a singular set of server properties which will be characterised victimization it's practi-cal, operational, and structural attributes. The availability of cloud is updated in hash table, so that the available cloud will provide original file in sequential order.

6.5 Data retrieval and Decryption

Data Retrieval is the way of providing the information retrieval mod-ule that describes the retrieval data from completely non-identical vir-tual server solely demonstrate user. Information area unit encrypted in numerous virtual server including fragments. The growth of data has been accompanied by a growth in the energy usage and carbon footprint of IT along with increased costs [16]. Information area unit gets retrieved from totally non-identical virtual server and info gets mixed and gets converted into decrypted format. Decrypted infor-mation area unit transfer to original information for user extraction. The major role is justified in here is to realize the fact of retrieval by removing faults and compromise of data of the system throughout failures. At current aim, this element supports ft.−units that understand the fact of retrieval in order where associate errors often resumes back into a traditional operation mode.

VII. RESULT

File is divided into fragments and fragmented knowledge gets repli-cated over the cloud nodes Every of the nodes stores solely a of a se-lected file containing partial fragment gets stored wholly which en-sures that even during a triple-crown attack, no meaning data gets un-concealed by the assaulter. The results received after performing ex-periments showed the support for real apps and the data storage in the desired location must be revealed to identify consistency of storage with concerned nodes of the clouds. After all the experiments, the lifetime of battery is in a state of extension. Lastly, we've gained knowledge about OFS that it is best suitable for read-intensive apps, for some written procedure and for systems where procedure offload-ing gets implemented.

VIII. CONCLUSION

The data transferred to a public cloud should be secured enough. OFS has 2 limitations. First, because of the appliance level precedence, OFS isn't responsive to exact location of each fragment over the nodes. Thus, though a file is being stored by the user itself, OFS still needs to get the concerned locations of nodes over the server from the local device before the info gets accessed by the tasks it must be re-vealed to the administrator to maintain secured storage over the virtual servers. This increases the overhead of accessing files in a same cloud storage. And also OFS wholly supports the way of storing data of files, during which the application can retrieve it by managing the

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 9467 - 9475
Received 05 March 2021; Accepted 01 April 2021.

im-portant progress. It doesn't always give away the contribution in se-cured storage which maintains tasks from more than one application. Our future work on OFS will be completely to manage these in con-cerned manner.

REFERENCES

[1] Zhang, A. Szekeres, D. Van Aken, I. Ackerman, S. D. Gribble, A. Krishnamurthy, and H. M. Levy, "Customizable and extensible de-ployment for mobile/cloud applications," in OSDI '14, 2014.

[2] C. Borcea, X. Ding, N. Gehani, R. Curtmola, M. A. Khan, and H. Debnath, "Avatar: Mobile distributed computing in the cloud," in Mo-bileCloud '15, 2015.

[3] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in EuroSys 2011.

[4] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of com-putation offloading for mobile systems," Mob. Netw. Appl., vol. 18, no. 1, pp. 129–140, Feb. 2013.

[5] R. Ramachandran, D. J. Pearce, and I. Welch, "AspectJ for multi-level security," in Proceedings of the Fifth AOSD Workshop on As-pects, Components, and Patterns for Infrastructure Software, 2006, pp. 13– 17.

[6] "Phonelab: A smartphone platform testbed," https://www.phone-lab.org/, [Online; accessed 02-Feb-2016].

[7] "Bionic sources (official repository)," https://android.googlesource.com/ platform/bionic/, [Online; accessed 5-Mar-2016].

[8] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in Proceedings of the IEEE Infocom 2012, March 2012, pp. 945–953.

[9] R. Tobbicke, "Distributed file systems: Focus on andrew file sys-tem/dis- ¨ tributed file service (afs/dfs)," in Mass Storage Systems, 1994.'Towards Distributed Storage and Data Management Sys-tems.'First International Symposium. Proceedings., Thirteenth IEEE Symposium on. IEEE, 1994, pp. 23–26.

[10] J. Protic, M. Tomasevic, and V. Milutinovic, "Distributed shared memory: concepts and systems," Parallel Distributed Technology: Sys-tems Applications, IEEE, vol. 4, no. 2, pp. 63–71, Summer 1996.

[11] J. B. Carter, "Distributed shared memory: concepts and systems," J. Parallel Distrib. Comput., vol. 29, no. 2, pp. 219–227, Sep. 1995. [Online]. Available: http://dx.doi.org/10.1006/jpdc.1995.1119

[12] Y. Go, N. Agrawal, A. Aranya and C. Ungureanu, "Reliable con-sistent and efficient data sync for mobile apps", Proc. 13th USENIX Conf. File

[13] D. Perkins et al., "Simba: Tunable end-to-end data consistency for mobile apps", Proc. 10th Eur. Conf. Comput. Syst., pp. 7:1-7:16, 2015.

[14] B. Atkin and K. P. Birman, "MFS: An adaptive distributed file system for mobile hosts", 2003.

[15] E. B. Nightingale and J. Flinn, "Energy-efficiency and storage flexibility in the blue file system", Proc. 6th Conf. Symp. Opearting Syst. Des. Implementation, pp. 363-378, 2004.

[16] E. B. Nightingale and J. Flinn, "Energy-efficiency and storage flexibility in the blue file system", Proc. 6th Conf. Symp. Opearting Syst. Des. Implementation, pp. 363-378, 2004.

[17] Dr.J.GeorgeChellinChandranJencyAnand, V.Vijayaganth, “Data security as a service in cloud”, International Journal of Advanced Re-search in Computer Science Engineering and Information Technology, vol. 2, Issue 3.

[18] V. Vijayaganth, P. Purusothaman, M. Krishnamoorthi, A Com-prehensive Survey on Security Challenges and Techniques in Big Da-ta, International Journal of Psychosocial Rehabilitation, 2020,Volume 24,Issue 06, Pages 6502-6508, 2020

[19] Vijayaganth V, Data Mining Techniques for Social Network Analysis, 2018, Cognitive Social Mining Applications in Data Analyt-ics and Forensics, Volume 1, Pages 25-40.

[20] Naveenkumar M, Srithar S, Vijayaganth V, Ramesh kalyan G, Identifying the Credentials of Agricultural Seeds in Modern Era, Inter-national Journal of Advanced Science and Technology, Vol. 29, No. 7s, page. 4458 – 4468, 2020

[21] N. Nandhini and R. Bhavani, Feature Extraction for Diseased Leaf Image Classification using Machine Learning, 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1-4, 2020

[22] M Naveenkumar and Vishnu Kumar Kaliappan 2019 J. Phys.: Conf. Ser. 1362 012063