IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 10, October 2021

DOI: 10.17148/IJARCCE.2021.101021

MyCloud: A Self-Hosted Hierarchical Cloud Service

Karan Gopal Gaur[1], Mayank Tiwari[2], Pratik Gupta[3]

[1,2]Computer Science Engineer, Department of Computer Science & Engineering,

Swami Keshvanand Institute of Technology, Management & Gramothan,

Ram Nagariya Rd, Shivam Nagar, Jagatpura, Jaipur-302017, (R.J), India.

[3]Acropolis Institute of Technology & Research, Mangliya Sadak, Indore-453771, (M.P), India.

Abstract: MyCloud is a personal online hosting for clients for personal and business use. It enables users to store and access data over the cloud from their trusted source (Personal disks or other trusted platforms). The platform provides the company to reserve the right to admission for their employees only. Further, it facilitates sharing of data based on hierarchies in the profession. While there are many such platforms available, no Cloud services exist which can work with as few resources as MyCloud capable of processing with RaspberryPi for a small scale business to working for a full-fledged organisation.

Keywords: Cloud Storage, Hierarchical Cloud Storages Model, Normalized DB Schema for Cloud Storage, Caching for Cloud Storage, Scalable cloud.

I. INTRODUCTION

MyCloud's vision is to provide users with software for Infrastructure As A Service (IAAS) for a Cloud Storage Solution for Business and personal use. Requirements for MyCloud is fueled by the recent ongoing distrust on many Cloud Platforms for data breach and the use of data for various research without the end user's consent. MyCloud provides an altogether different solution where all essential data is stored on a client's premises or platforms trusted by clients to meet data privacy demands. Moreover, MyCloud provides hierarchical Cloud Storage Solutions with various profiles and roles. Admin users have permission to create new roles and restrict or grant access to numerous files in the organisation as per the roles. In addition, MyCloud will keep track of all the recent changes made to any organisation files by the users and their reciprocal action to promote security and liabilities. Admin users are also entitled to manage permissions for various privileges given to each Role. The privileges include allowing users with specific roles to edit files directly on the cloud or limiting and managing the storage limits. These features for private businesses make MyCloud entrusting security and a viable alternative to many cloud platforms.

II. SOFTWARE USED

A. VSCode Editorial: VSCode is a globally renowned code editor for various frameworks. VSCode has vast support and extensions to work with git, javascript, express, databases, etc. - which assists in the development and version control.

B. RaspberryPI: To showcase the efficiency, we have used a Raspberry PI 3 B+ as our backend server. MyCloud is optimised to work with ARM Cortex-A53 1.4GHz processor with 1GB RAM and 300 Mpbs of ethernet speed.

C. MongoDB: MongoDB is a popular choice amongst unstructured databases with fast query response and high flexibility. Being a NoSQL Database, Mongo is proven and well suited for Cloud Storage with merits such as change friendly and self-sharding allowing horizontal scaling. Thus MongoDB is perfect for storing hierarchal data used for MyCloud files.

D. Redis: Redis is an in-memory data structure used as a cache and to store values in the form of key-value pairs. Redis uses the physical memory of the system for storage. Further, Redis has O(1) search complexity for queries as the keys act as a hash to fetch the index.

E. NodeJS: NodeJS is a backend platform for running Javascript outside of the browser via a V8 engine. NodeJS is best suited for the backend involving high IO operations. NodeJS has a native implementation for Non-Blocking IO Calls. Non-Blocking IO Calls helps in facilitating multiple and concurrent requests performing operations on files storage via IO busses. This improves the throughput of the application.

IJARCCE

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 10, October 2021

DOI: 10.17148/IJARCCE.2021.101021

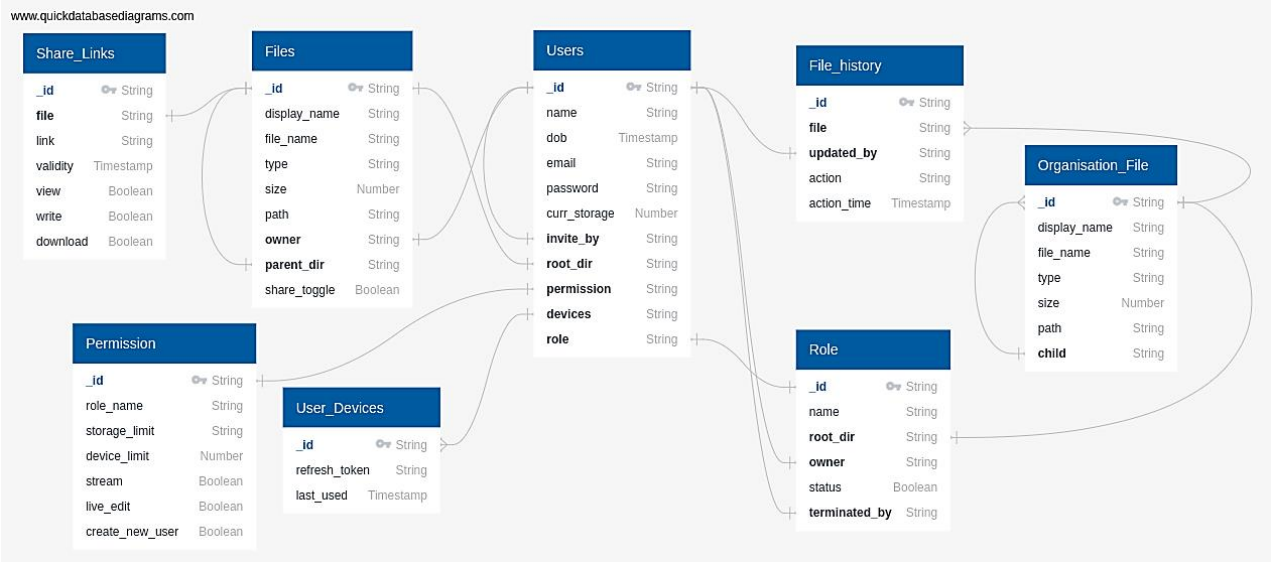
F. ReactJS: ReactJS is a Javascript library well suited for frontend and provides feasibility to design and reuse UI Components. ReactJS is developed and maintained by Facebook.

G. Nginx: Nginx is the foremost layer of security for servers via reverse proxy for the infrastructure and servers. Additionally, Nginx also works as HTTP Cache and Load Balancer.

H. External Storage: MyCloud allows clients to choose their trustworthy source for data storage. Clients can choose on-premises file storage similar to a 500 GB hard disk or AWS S3 Buckets serving as a scalable storage resource.

III. DATABASE

Database designing is crucial for a data-driven project similar to cloud storage. It is essential to reduce the number of IO Calls to physical storage. Due to limited IO busses available, IO Calls often time bottlenecks the response time for each query. Thus, for Cloud Storage, priority is given to Database Designing to reduce redundancies and overhead computations. Additionally, for managing hierarchical file data and roles for business storage services, a flexible NoSql Database is required.



While Cassandra DB is an alternative to store all files within the database and save IO calls in the process, MongoDB is better suited. Since, as the cloud storage increases, relatively query response time also increases, resulting in poor performance of the database. Thus, storing files in physical disks is preferred, reducing database size and improving

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

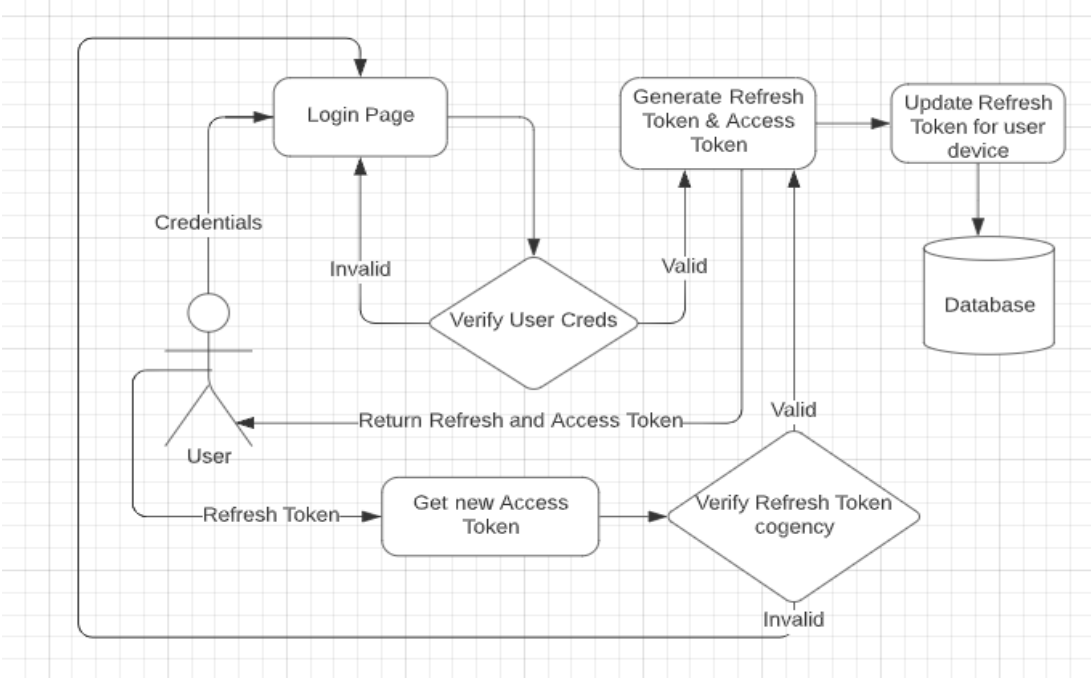
International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 10, October 2021

DOI: 10.17148/IJARCCE.2021.101021

database backups and uptime in the process. Also, MongoDB provides flexibility towards design changes and supports auto-sharding (making database horizontally scalable) as data grows.
Since the file hierarchies are subjective to users, information related to user files must be stored within the DB. Such methodology helps decrease file IO operations (All changes for renaming files, moving a file from one directory to another, etcetera managed at DB level) and also grants atomicity for file storage.

IV. SECURITY IMPLEMENTATIONS



For security, we have implemented OAuth 2.0 with rotating refresh tokens. Each client has the option to update token expiry durations. Rotating refresh tokens are crucial to protect users from the theft of user passwords or tokens. Once a user logs in from a new device, a RefteshToken (RToken), AccessToken (AToken) and a unique DeviceID for each device are returned from the server. AToken allows access to all services of MyCloud for the specific user until expired. Once the AToken expires, a new token is generated in exchange for the RToken. The new RToken and AToken are returned to the user, extending access to MyCloud services. If the token is breached and a request for new AToken is received with unknown or used RToken, all user devices are flagged and revoked. The user has to provide a username and password to regain access to MyCloud services.
AccessTokens are mandatory for users to access MyCloud services. ATokens carry the user information regarding roles, permissions and profile information. ATokens thus act more than a source of authentication and authorisation as they also reduce the number of database calls.

V. CACHING

The heavy dependency of Cloud Storage services over the database degrades the performance of the infrastructure. All API requests require interaction from the database ranging from login to folder information. Often users request redundant queries leading to overhead processing and resource consumption. Thus caching is introduced to reduce redundant query processing. All Least-Recently-Used (LRU) queries from the database are stored as key-value pairs in Redis for personal files (MyCloud). For OurCloud, caching of data is based on Most-Frequently-Used (MFU). Caching leads to a reduced redundant database as information regarding frequently used files is stored within the cache.

A drawback of caching is maintaining consistency between the database and the cache. Caching uses the volatile physical memory of the system which are susceptible to data loss under power outages. Further, many concurrent changes in files and storage may lead to inconsistencies between the cache and the database. Such an enigmatising user experience may be pivotal for business. A Write-Thorough Cache implementation is enforced to avoid data inconsistency. A Write-Through Cache works by updating the database and simultaneously upserting the value in the cache in real-time.

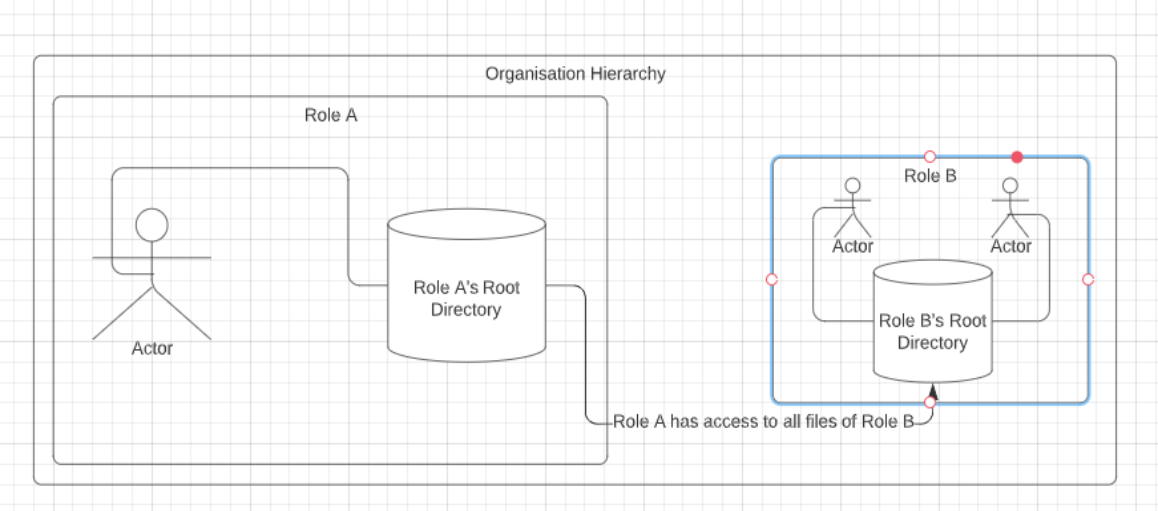
IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 10, October 2021

DOI: 10.17148/IJARCCE.2021.101021

VI. ADMIN USER OPERATIONS



A. Manage Roles and permissions: Admin user privileges are reserved for Database Administrators responsible for managing requirements for business. An admin user is an ordinary user with additional privileges to manage the roles and permissions of an organisation. Admin users reserve the right to create, update or delete any role in the organisation.

If a new role is introduced, a root directory folder is specifically created to manage, share and access all data available for all users with the given role under the business cloud. To manage hierarchies in the organisations, the admin user must add the root directory of role A (Lower in the hierarchy) to the CHILD list of role B (Presumably higher in the hierarchy) root directory of the parent role. Thus all the child folders of A are now accessible to role B forming a hierarchy tree.

All new users are allotted basic permissions and limits. Admin users can downgrade or allocate add-on permission to each user. However, additional privileges account for high load and server computations.

B. Manage user data: Since MyCloud is a private Cloud Storage service, new users can only be registered by invitation. Admin users have permission to invite new users to the cloud and assign roles. Similarly, at the end of a user's tenure, an admin can suspend the user's account or permanently delete their account along with all the data.

VII. CLIENT USER OPERATIONS

A. Manage User Files: MyCloud is first and foremost a cloud service providing users online storage for their files. Each user can upload and manage files from anywhere via the internet and browser. All files are associated with the UUID mapped with the name given by the user. All file information regarding size, owner and path is stored within the database. Folder hierarchies do not exist on physical storage but are virtually depicted and maintained by the database. Each user has a limited storage limit available as set by the admin user.

B. MyCloud Services: Users could also access their files on the fly. Currently, MyCloud supports editing text files, watching video files of specific encoding, and playing songs over the browser without the need of downloading. However, only admin users can provide access to over aforementioned cloud services.

C. File Share Services: When the data is over the cloud, users often need to share it with their peers. Downloading and sharing the files is an inept inconvenience. MyCloud provides a timesaver option where users can directly provide a link to access respective files to their friends and families. Users can manage security access via such links (download file, read-only, read and write) which have a pre-set expiry. For security concerns, a share-link option is not extended to OurCloud (Organisation Cloud files).

D. OurCloud services: OurCloud is the business branch of MyCloud used to share files within the organisation. Unlike MyCloud, users don't need additional permissions to manage files (editing text files, watching video files of specific encoding, and playing audio files). Further, OurCloud has no enforced storage limitations. OurCloud also has a feature to monitor changes to a file and keeps track of the type of task performed in an organisational file (History

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 10, October 2021

DOI: 10.17148/IJARCCE.2021.101021

include - Who accessed file, who modified file contents, who downloaded file). However, sharing of files is restricted for security purposes for OurCloud files.

VIII. PERFORMANCE IMPROVEMENTS

A. While applications such as Droppy and OwnCloud provide solutions for IAAS for Cloud Storage, they lack scalable and robust solutions. MyCloud has additional benefits of caching, a portal for organisation files and also file and server monitoring.

B. With security concerns, MyCloud provides options of physical storage at the client's trusted source. Further, MyCloud provides hierarchical file storage for organisations.

C. MyCloud's system design emphasised working with minimum resources (Raspberry Pi for the individual cloud). However, scalable to work with small businesses.

D. The database is optimised to reduce the number of IO calls which bottlenecks the platform's performance.

IX. FUTURE SCOPE

The future of this project is to switch from Monolithic to Microservice Architecture to improve scalability and load resistance on the platform. Microservices Architecture is a form of system designing which helps in load management and increasing uptime of servers by compartmentalisation of services which enables all services to work independently and prevent any downtime by a Domino Effect. The services communicate with each other using KAFKA. KAFKA acts as a messenger and queues all messages for a service during server downtime to avoid any loss of communication.

X. CONCLUSIONS

This paper helps share the best practices for designing and building a Cloud Storage Platform for individual businesses. The heart of this project is the database designing for hierarchical file storage. This enables small-time businesses to store crucial data on secure premises of their trust at minimal costs. Although the server performance takes a hit when working with frugal resources like RaspberryPI, the cost for service outmatches most other solutions.

XI. REFERENCES

[1] Silverwind - Design and Implementation of Self-Hosted Cloud Storage - Droppy https://github.com/silverwind/droppy

[2] Ashish Chakraborty, “System Design Analysis for Google Drive” https://towardsdatascience.com/system-design-analysis-of-google-drive-ca3408f22ed3

[3] Enrico Bocchi, Idolio Drago and Marco Mellia - “Personal Cloud Storage Benchmarks and Comparison”, IEEE Xplore. ISSN: 2168-7161

[4] Liu Hao and Dezhi Han - “The study and design on secure Cloud-Storage System”, IEEE Xplore. ISBN: 978-1-4244-8165-1

