International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 06 | June 2019 www.irjet.net p-ISSN: 2395-0072

Improving Data Storage Security and Performance In Cloud Environment

Snehal A. Ghogare[1], Prof. Varshapriya J. N[2]

[1]M.Tech Student, Dept of Computer Engineering and IT, VJTI College, Mumbai, Maharashtra, India
[2] Associate Professor, Dept of Computer Engineering and IT, VJTI College, Mumbai, Maharashtra, India

---

Abstract - *A Most of the existing cloud storage platforms do not provide a complete implementation of all security components as well as the performance of system degrades under high concurrent pressure and large-scale environment. Compared to boot from local disks, booting from volumes often lead to failure, those are largely due to the low performance of Cinder. Database, threading model, deployment architecture and other factors will affect the performance of cinder largely. Under certain conditions all of these can become a performance bottleneck. We will provide a layer of security to storage systems by extending the OpenStack cloud computing stack to support heterogeneous architectures and accelerators, distributing the storage to each cloud in order to help everyone to be more convenient to use cinder. Our middleware split a file according to the computing capability of node, being uploaded into segments, encrypting them and then upload each segment to each backend cloud storage. When user want to download a file, system again recreate the original uploaded file and allow downloading it.*

***Key Words*: Cloud Storage, Data security, Openstack Cinder, Compute, Storage, Authentication.**

1. INTRODUCTION

Cloud Service Provider(CSP) is responsible for maintaining and monitoring the out sourced data. Cloud is a public environment where there are many possibilities to attack the data. Data outsourcing brings security issues in the cloud while move to storage. Once the data is outsourced to the cloud, CSP is only responsible for maintaining, monitoring and controlling the data. Now a days many organizations and enterprises have started to outsource their data to cloud. Cloud is a public environment where there are lots of possibilities to attack the user data. Security is the highest concern in the cloud environment. Outsourced data to the cloud are kept by third party CSP. In this situation, data may be attacked from inside as well as outside the cloud. Data security is ensured by security parameters such as confidentiality, integrity and availability. This paper presents a middleware-oriented framework that integrates different IaaS Storage Clouds. Based on the registered users access token, the middleware does the authentication with the IaaS cloud frameworks. The middleware relies on a service level manager which decides how to split a file being uploaded, its encryption and decryption followed by merger for download. The evaluation of the framework shows high improvement in the security as we are distributing the storage as per the computing capability of node and also encrypting the data.

In the proposed system, the issue of progressing files to a different user through storage servers is completely under the command of the data owner. The data owner uploads the file which is in turn encrypted and stored in the server. The distributed storage system consists of distributed storage servers and key servers. These key servers are extremely secured by security techniques and performance of system will be improved as the data is distributed according to the computing capability of node so that retrieval time will be minimium.

In the existing cloud storage system, digital data is stored in logical pools, the physical storage contains multiple servers or locations, and the hosting company manages physical environment. Storage capacity is given on lease to people and organizations for the storage of different type of data. Cloud storage services are accessible via web services (API) or by applications that utilize the API. Amazon Web Services introduced their cloud storage service. AWS S3 is a one of the first cloud storage supplier.

Other cloud storage services are popular services like Google Drive, Google Cloud Platform, Smugmug, Dropbox, Box and OneDrive. Cloud storage is based on highly virtualized infrastructure. It is a multitenant system with multiple storage devices inside. The storage cloud is built from low cost components for ensuring reliability in the software, and building advanced functionality on top of this foundation.

2. RELATED WORK

Some open source platforms such as Swift (OpenStack cloud), Walrus (Eucalyptus cloud), and Cumulus (Nimbus cloud) are used to analyze the security blocks for the cloud storage platforms. Table 1 explains the security components summary of the instantiations of every platform.

Authentication and authorization components provide a adequate security level. Users can make use of different access credential mechanisms (e.g. passwords, certificates) to access the data. In Amazon S3 platform, it provide an Access Control List (ACL) mechanism where owners can assign policies to specific customers, groups and data containers(data bucket). Note, however, that there is still room for improvement in these components. OpenStack provides a manageable interface to cover area of the

authentication and authorization mechanisms with out need to modify the whole platform. In some caes, ACLs are deficient to fully catch the enterprise environment's complexity. Approaches such as Role-base Access Control (RBAC) might be integrated.[4][6]

Table -1: Summary of Security components of existing cloud storage platforms.

	Logging/ Auditing	Device Authenti-cation	Data Protection	Credential Storage	Extended Services
Eucalyptus	Logs Health System	Cloud:WS Storage:None	None	Stored in folder	No support
Openstack	Logs Health System	Cloud:Keystone Storage:None	None	None	Authentication/ Authorization
Nimbus	Logs	Cloud:SSH Storage:None	None	Stored in folder	"Plugin" support
Amazon S3	Logs Mgmt system		Server side Client side		No support

Finally, there are various components that are not hold up in most platforms. For instance, existing device authentication/authorization and secure communications components are designed to protect the communications between cloud entities, but they are not used to protect the communications inside the storage subsystem. There is no direct support for storing the credentials in secure and tamper-resistant containers. As for data protection, no platform provides mechanisms that implement data-at-rest encryption. With the exception case of Nimbus platform and (partially) OpenStack platform, it is impossible to implement specific plugins that provide additional extended services such as proof of storage services.[4][7]

3. PROPOSED METHOD

In the proposed system, authentication is to be checked in an openstack inorder to make only an authenticated user to access the instances created in the cloud. Following are the proposed steps for block encryption

1. Nova API call for volume mount

2. Intercept call and mount volume as a block device to virtualization host.

3. Get Key via a SSL/ TLS connection.

4.Loop and dm-crypt block device

5. Update VM Config for block devices

6. Create VM and start it.

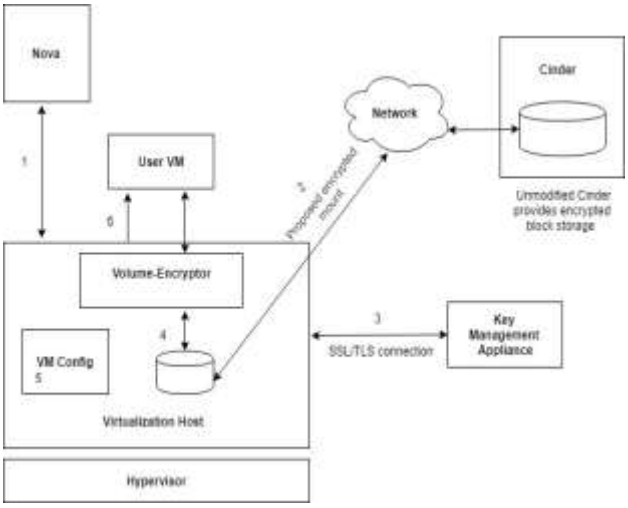

Fig -1: Proposed System architecture

4. EXPERIMENT, EVALUATION AND DISCUSSION

4.1 Openstack block storage:

IRJET Instances use an empemeral volume by default. This kind of volume does not save the changes made on it and reverts to its original state when the current user give away control. One of the methods for storing data permanently in openstack cloud is the cinder block storage service. Openstack block storage service consists of four services-

- Cinder api
- Cinder scheduler
- Cinder volume
- Cinder backup

Cinder api
API service provides an HTTP endpoint for API requests.Two versions of API are supported and required for the cloud. So Cinder provides six endpoints. The cinder-api verifies the identity requirements for an incoming request and after that routes them to the cinder-volume for action through the message broker.

Cinder scheduler
Scheduler service reads requests from the message queue and selects the optimal storage provider node to create or manage the volume.

Cinder volume
The service works with a storage back end through the drivers. The cinder volume gets requests from the scheduler and responds to read and write requests sent to block storage service to maintain state.

Cinder Backup
We can use several back ends at the same time. For each back end you need one or more dedicated cinder volume

service cinder backup. The backup service works with the backup backend through the driver architecture.

4.2 Factors that lead to poor performance:

We analyse the deployment architecture and find the factors that lead to poor performance.

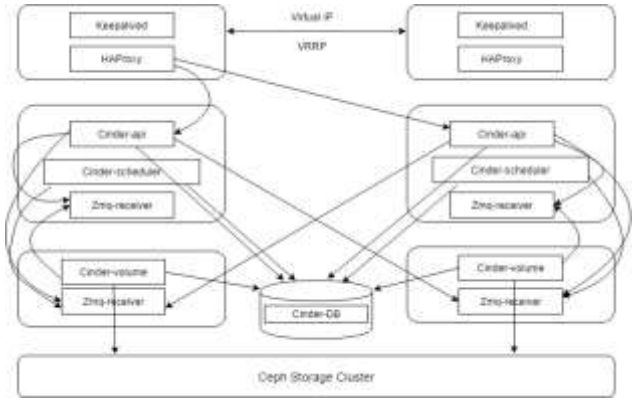

Fig -2: Deployment architecture of cinder block storage

Following are the factors that lead to poor performance-
1. HA Proxy report error

2. The number of cinder api workers

- Cinder api takes longer time to process each request under high concurrency pressure.
- Database connection driver is implemented beyond eventlet's monkeypatching, blocking database calls with block eventlet thread

3. Cinder volume has more serious performance issues

- very heavy workload
- create volume
- initialize connection
- Cannot consume message timly from MQ
- Takes a longer time to process each request

4. Storage driver

If the ICSI target information is not stored in a separate file on creation, even a node reboot, your existing volumes on that node will be restored automatically.

5. RBD Rados call may block cinder-volume

- rbd and rados liberty are not patched by eventlet
- long running tasks block eventlet loop
- cinder volume becomes a zombie process

5. CONCLUSION

This paper shows how to implement an encrypted cloud storage system on the basis of OpenStack Cinder, and how to increase the security and performance by analyzing the deployment architecture of cinder. The compatibility of this system allows it to be deployed in existing storage systems which uses Cinder as storage technology and provide cinder APIs and Keystone APIs without modifying those existing systems.

REFERENCES

[1] R. Nivedhaa and J. Jean Justus, "A Secure Erasure Cloud Storage system using Advanced Encryption Standard algorithm and Proxy Re-encryption", IEEE Conference on Communication and Signal Processing (ICCSP) April, 2018.

[2] B. Fathima Mary, George Amalarethinam, "Data Security Enhancement in Public Cloud Storage using Data Obfuscation and Steganography", IEEE transaction on data storage 2017.

[3] Bin Feng, Cheng Guo, "An Efficient Protocol with Bidirectional Verification for Storage Security in Cloud Computing", IEEE transaction on cloud computing May, 2016.

[4] Roman, R., Felipe, M., Gene, P. and Zhou, "Complying with Security Requirements in Cloud Storage Systems", Journal of Computers, 11(3), pp.201-206, 2016.

[5] N.Saranya, S.Nivedha, "Implementing Authentication in an Openstack Environment", International Conference on Computer Communication and Informatics (ICCCI - 2016), Jan. 07 – 09, 2016.

[6] Pragya Jain, Aparna Datt, S.C. Gupta, "Cloud Service Orchestration based Architecture of OpenStack Nova and Swift ", Intl. Conference on Advances in Computing, Communications and Informatics (ICACCI), Sept. 21-24, 2016.

[7] Peidong Sha , Zhixiang Zhu," The modification of RSA algorithm to adapt fully homomorphic encryption algorithm in cloud computing" ,IEEE 2016.

[8] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," IEEE Transactions on Dependable and Secure Computing, vol. 13, no. 3, pp. 312–325, 2016.

[9] Sandeep Nehe, Prof. M.B. Vaidya, "Data Security using Data slicing over storage clouds", International Conference on Information Processing (ICIP) Vishwakarma Institute of Technology, Dec 16-19, 2015.

[10] L S Girish, Dr. H. S. Guruprasad, "Building Private Cloud using OpenStack" , International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), volume 5, Issue 3, May-June-2014.

[11] Bin S. Suganya, P. Damodharan, "Enhancing Security for Storage Services in Cloud Computing", International Conference on Current Trends in Engineering and Technology, ICCTET, 2013.

[12] S. Suganya, P. Damodharan, "Enhancing Security for Storage Services in Cloud Computing", International Conference on Current Trends in Engineering and Technology, ICCTET, 2013.

[13] C. Cachin, I. Keidar, and A. Shraer, "Trusting the cloud," ACM STGACT News, vol. 40, no. 2, p. 81, 2009.