OpenStack Storage Vs. Amazon Storage

Course: Computing Clusters, Grids and Clouds
Professor: Andrey Shevel
Student: Thi Yen Nhi VO

1. Background

Amazon Web Services (AWS) [1] is considered as the foremost vendor, accounting for around half of the cloud computing platform sales across the industry. Today, AWS is the $7 billion business and it's expected to be around a $50 billion business by 2020. Services provided by Amazon are diverse, ranging from compute, storage, networking, database, analytics, application services, deployment, management, mobile, developer tools to tools for the Internet of things. The most central and best-known of these services are Amazon Elastic Compute Cloud, so called "EC2", Amazon Simple Storage Service (S3) and Amazon Elastic Block Store (EBS).

OpenStack [2], in the other hands, is also a leading technology, supported by almost every major IT vendor. According to some estimates, what was initially worth $850m in 2014 is now growing continuously and is expected to exceed $2.4bn by 2017. Services provided by OpenStack are also diverse, including 9 projects known as the most central parts of OpenStack: Compute, Object Storage, Block Storage, Network Management, Authentication, Image Service, Billing Service, Cloud Template and Dashboard.

They are the two leading vendors in the booming cloud market. Each of them has their own merits and was designed for different purposes, however, they do share some features in common. Storage feature is one of them.

This document will try to compare between AWS storage services and the OpenStack storage component to discover strengths and weaknesses of each products and how to decide which platform to use.

2. Storage at glance

Object storage and Block storage are referred as the most common storage's types in the cloud computing market. In order to fulfill the user's demands, both OpenStack and AWS are investing heavily in these two storage features. Each type has its pros and cons, therefore when to use which depends mainly on users' specific needs, the size of the environment and also the budget.

2.1. Object storage

In Object storage, data is organized in form of units of storage, so called objects. Each object contains data, metadata, and the unique identifier. Data can be anything such as files, images or media. Metadata with unlimited size can house anything from security classification of the files resided within the object to the application associated with the information. Whoever has store a picture on Facebook or a song on Spotify has used object storage without knowing about this.

Object storage is ideal for storing large amount of data because when the data grows, object-based storage architectures can be scaled out and managed simply by adding nodes. They are also designed for high availability by duplicating objects and store them across the distributed system. By doing so, if one or more node fail, the data can still be made available.

However, object storage are not suitable for objects that require frequent access to read, write, update the object's content because object storage requiring the entire object to be accessed, updated then re-written which will cause a huge overload.

2.2. Block storage

In Block storage, data are split into even sized blocks of data, each block has its own address. Each raw block data can be controlled as an individual item. This architecture is usually deployed in Storage Area Network (SAN).

Block storage is designed for flexibility and versatility. Operating system can connect to the raw storage volumes and use them as individual hard drives. They are most likely to be encountered in the majority of enterprise workloads due to the wide variety of uses in application and file storage such as database storage and virtual machine file system. This type of storage also supports individual formatting of file systems like NFS, NTFS or SMB (Windows) or VMFS (VMware) which are required by the applications.

3. OpenStack Storage

In OpenStack [3] project, there are two types of storage: ephemeral and persistent. If users deploy only the OpenStack Compute Service (nova), that users do not have access to any form of persistent storage by default. The disks associated with VMs are called "ephemeral," meaning that (from the user's point of view) this type of storage will disappear when a virtual machine is terminated. Persistent storage in the other hand is always available, regardless of the state of a running instance. There are two main types of persistent storage, including object storage and block storage. Table 1 will explains their main characteristics of each storage type.

Table 1. OpenStack Storage

	Ephemeral storage	Block storage	Object storage
Used to…	Run operating system and scratch space	Add additional persistent storage to a VM	Store data including VM images
Accessed through…	A file system	A block device that can be partitioned, formatted and mounted	The REST API
Accessible from…	Within a VM	Within a VM	Anywhere
Managed by…	OpenStack Compute (Nova)	OpenStack Block Storage (cinder)	OpenStack Object Storage (swift)
Persists until…	VM is terminated	Deleted by user	Deleted by user
Sizing determined by…	Administrator configuration of size settings, known as flavors	User specification in initial request	Amount of available physical storage
Example of typical usage…	10 GB first disk, 30 GB second disk	1 TB disk	10s of TBs of dataset storage

Swift and Cinder are the most central projects of OpenStack and are well received by the cloud platform community. When it comes to storage, swift is perfectly fit for storing large or growing data while cinder will be used as virtual drives/volumes to virtual servers whenever they are needed.

3.1. Cinder (OpenStack block storage)

Cinder [4] provides persistent block storage for the compute engine. The main idea of Cinder is to create an abstract layer between users and the physical block storage devices to manipulate multiple types of backend storages by the same interfaces. With this layer, users do not need to about the detail of the physical layer.

Cinder also supported various storage platform such as Ceph, CloudByte, Coraid, EMC (ScaleIO, VMAX and VNX), GlusterFS, Hitachi Data Systems, IBM Storage (Storwize family, SAN Volume Controller, XIV Storage System, and GPFS), Linux LIO, NetApp, Nexenta, Scality, SolidFire, HP (StoreVirtual and 3PAR StoreServ families) and Pure Storag.

Cider architecture is illustrated as figure 1

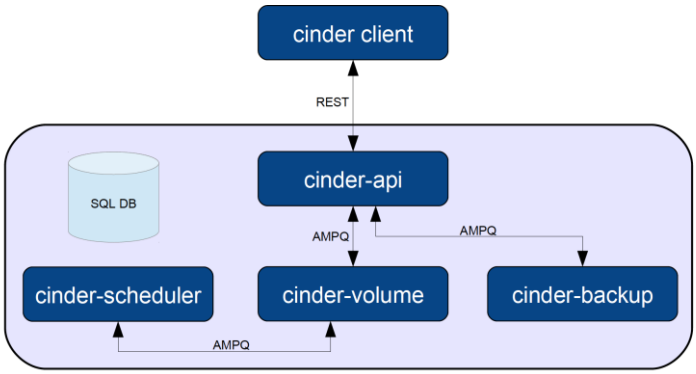

Figure 1. Cinder architecture

Cinder-api

- A gateway for authenticating and routing requests throughout the Block Storage service
- Incoming request will be sent to the cinder-scheduler component for dispatching to the appropriate volumes

Cinder-scheduler

- Sending request to the appropriate cinder-volume component via the AMPQ
- Can be configured to deliver request using round-robin

Cinder-volume

- Manages multiple storage back-ends
- Interacts directly with both hardware and software providing block storage services

Cinder-backup

- Provides backup services

3.2. Swift (OpenStack object storage)

Swift [5] is a robust highly scalable and fault tolerant object storage platform. Anything stored in OpenStack Swift as an object and each object can be accessed by a URI. Swift also supports automatic replication of content from failed disks to other active nodes. The simple version of Swift architecture is presented in figure 2.

In the architecture, the proxy nodes are where all the processes are handled and processed. Proxy is also the place where swift client can interact with the system. Storage nodes are where the object data are hosted.

Note that Swift uses the idea of eventual consistency which means data is not synchronously replicated across the storage clusters. This may lead to inconsistency of data in certain scenarios especially if server is down or the system is overloaded. To solve this problem, they used proxy server to ensure the I/O request is routed to the server where the newest version of object data is kept.

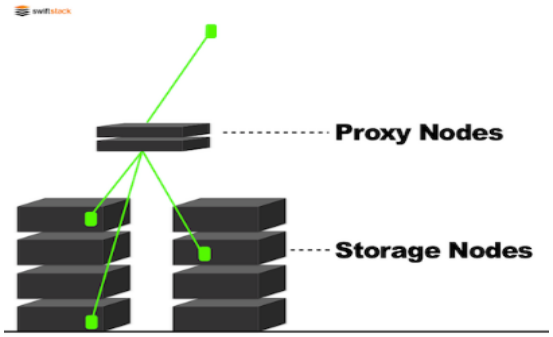

Figure 2. Swift architecture

4. Amazon Storage

Amazon storage services [1] provides a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web. Their goals is to maximize simplicity and robustness but also ensure that their storage services are fast, highly scalable, available and durable. In addition, storage services provided by AWS can be used alone or together with other AWS services and can be granted programmatic access using the AWS SDKs tools.

Comparing to OpenStack storage services, they have Amazon Simple Storage Service (S3) for object storage and Elastic Block Storage (EBS) for block storage. AWS S3 is known as highly scalable object storage and suitable for fast web object storage, while AWS is designed for high-performance block storage and very useful in system that needs very fast “instance” disk.

In Amazon storage services, each user when they buy a service has their own container, so called “bucket”. Each bucket has their own policy and data is stored inside the bucket which means you will get to decide who can access which.

5. Comparisons

Amazon storage services and OpenStack storage services share a lot in commons. Technically, they can offer users the similar services. However, they each have their own pros and cons.

Amazon storage services are commercial products, therefore they would take care of all of the complicated technology and offer users the simplest way to use their services. OpenStack in the other hand requires a lot of knowledge to configure and manage the system and a huge budget for infrastructure to provide the similar services. However, in OpenStack, it’s possible for users to customize the system in the way it suits best for their purpose while we cannot customize the services provided by Amazon.

6. Conclusions

Ultimately, if the company is a start up with a few IT staff, amazon would be the better choice because they come with flexible customer support options. In contrast, if there are available infrastructure and users want to get the most out of it, then OpenStack seems to be the better option. However, before building an in-house solution with OpenStack, users should consider the costs for both building and maintaining the system.

7. References

[1] "Amazon Web Services," [Online]. Available: http://docs.aws.amazon.com/. [Accessed 7 6 2016].

[2] "OpenStack market expected to reach $2.4bn by 2017," [Online]. Available: https://451research.com/report-short?entityId=87456&referrer=marketing. [Accessed 7 6 2016].

[3] "Welcome to OpenStack Documentation," [Online]. Available: http://docs.openstack.org/. [Accessed 7 6 2016].

[4] "Destiny - The Cloud," [Online]. Available: http://cloudn1n3.blogspot.ru/2014/11/openstack-series-part-6-cinder-block.html. [Accessed 7 6 2016].

[5] "Destiny - The Cloud," [Online]. Available: http://cloudn1n3.blogspot.ru/2014/11/openstack-series-part-7-swift-object.html. [Accessed 7 6 2016].