

PROFESSIONAL TRAINING REPORT
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award Of Bachelor of
Technology Degree in Information Technology

By

S Bhuvaneshwaran

REG NO : 40120017

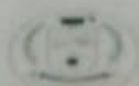


DEPARTMENT OF INFORMATION TECHNOLOGY SCHOOL OF
COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

April-2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

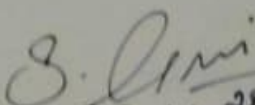
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

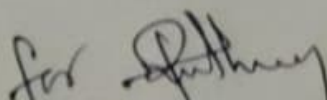
DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **DANISH HARRISON** (Reg. No: **40120024**) who carried out the project entitled "**IMPLEMENTATION OF OCTAVIA LOAD BALANCER IN OPENSTACK OF DEVSTACK**" under my supervision from January 2023 to April 2023.

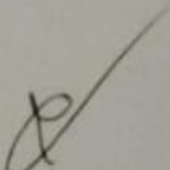

Internal Guide ^{28/4/23}

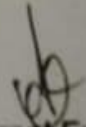
DR. S.GOWRI M.E., Ph.D.,


Head of the Department

Dr. R. SUBHASHINI M.E., Ph.D.,

Submitted for Viva voce Examination held on


Internal Examiner


External Examiner

DECLARATION

I **BHUVANESHWARAN.S (Reg. No. 40120017)** hereby declare that the Project Report entitled "**Implementation of Octavia Loadbalancer in OPENSTACK of DEVSTACK**" is done by me under the guidance of **Dr. J. Jabez** and is submitted in partial fulfilment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE: 04/2023



PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them. I convey my thanks to **Dr. T Sasikala, M.E., Ph.D.**, Dean, School of Computing **Dr. R Subhashini, M.E., Ph.D.**, Head of the Department of Information Technology for providing me with necessary support and details at the right time during the progressive reviews. I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. J Jabez, M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement that paved way for the successful completion of my project work. I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

CERTIFICATE OF COMPLETION

		इलेक्ट्रॉनिक्स एवं सूचना प्रौद्योगिकी मंत्रालय MINISTRY OF ELECTRONICS AND INFORMATION TECHNOLOGY	Ref.No.: 48 /FSPCC/BC001/04/2023  रा.इ.सू.प्रौ.सं NIELIT
COURSE COMPLETION LETTER			
It is certified that			
Bhuvaneshwaran S			
Sathyabama Institute of Science and Technology			
has successfully undergone the Bridge Course Training Program on			
Cloud Computing Technology			
Under "FutureSkills PRIME" Programme			
commenced from 11 Mar, 2023			
 Shri. V. Sridharan Chief Investigator Cloud Computing Technology Lead Resource Centre	L R PRAKASH Digitally signed by L R PRAKASH Date: 2023.04.26 03:49:56 +05'30' Cdr.(Retd.) L.R. Prakash Senior Director Centre Head C-DAC, Chennai		

ABSTRACT

The implementation of Octavia load balancer in Devstack of OpenStack provides a powerful tool for balancing traffic in cloud environments. This abstract provides an overview of the process involved in setting up Octavia in Devstack. The implementation involves the installation and configuration of Devstack, followed by the addition of Octavia as a service. Once Octavia is installed, load balancing pools and listeners can be created, and members can be added to pools to balance traffic. The abstract highlights the benefits of using Octavia load balancer in Devstack, including improved performance and availability of services, automatic failover, and scalability. Overall, the implementation of Octavia load balancer in Devstack is a valuable addition to any OpenStack cloud infrastructure.

Cloud computing has revolutionized the way businesses and individuals use and store data. This abstract provides an overview of what cloud computing is and how it works. In essence, cloud computing refers to the practice of using remote servers to store, manage, and process data rather than using a local server or personal computer. Cloud computing offers a number of benefits, including cost savings, scalability, flexibility, and increased collaboration. Cloud services are provided by a variety of companies, including Amazon Web Services, Microsoft Azure, and Google Cloud Platform, among others. This abstract also discusses the various types of cloud computing, including public, private, and hybrid clouds. Public clouds are accessible to anyone on the internet, while private clouds are dedicated to a specific organization or group. Hybrid clouds combine elements of both public and private clouds. The abstract concludes by highlighting the importance of cloud computing in today's digital landscape and its potential to transform the way we work .

TABLE OF CONTENT

CHAPTER NO:	TITLE	PAGE NO:
1	INTRODUCTION	1
	1.1 Specification	3
	1.1.1 Hardware Configuration	3
	1.1.2 Software Sonfiguration	3
2	CLOUD COMPUTING AND VM VIRTUAL BOX	4
	2.1 Cloud Computing	4
	2.2 Multicloud and Hybrid Multicloud	4
	2.3 Cloud Computing Architecture	5
	2.4 Types of cCloud	5
	2.5 Advantages of Cloud Computing	8
	2.6 Disadvantages of Cloud Computing	9
	2.7 Virtual Box	10
	2.8 Pros and Cons of Virtual Box	11
3	UBUNTU AND ITS FEATURES	13
	3.1 Ubuntu	13
	3.2 Installization Ubuntu in VM Box	13
	3.3 Features of Ubuntu	14
	3.4 Work for Ubuntu	15
	3.5 Pros and Cons of Ubuntu	16
4	DEVSTACK AND ITS FEATURES	18
	4.1 Devstack	18

	4.2 Advantages of Devstack	18
	4.3 Disadvantages of Devstack	19
	4.4 Limitation of Dev Stack	20
	4.5 Work For Devstack	20
	4.6 System Requirement	21
	4.6.1 Windows	21
	4.6.2 Ubuntu	22
5	PROJECT IMPLEMENTATION	23
	5.1 Creating Openstack	23
	5.2 Devstack Installization	24
	5.3 Installization Required Package	24
	5.4 Download Devstack	24
	5.5 Nano Local Conf	25
	5.6 Run Dev	26
6	OPENSTACK AND ITS FEATURES	27
	6.1 OpenStack	27
	6.2 Work for OpenStack	28
	6.3 Advantages of OopenStack	30
	6.4 Disadvantages of OpenStack	31
	6.5 installization of OpenStack	32
7	CONCLUSION AND REFERENCE	34
	7.1 Conclusion	34
	7.2 Future Scope	34
	7.3 Reference	35

LIST OF FIGURES

FIGURE NO	NAME OF FIGURE	PAGE NO
2.1	Cloud Computing	4
2.2	Infrastructure as a service	6
2.3	Platform as a service	7
2.4	Software as a service	8
2.5	VM Virtual Box	11
2.6	Oracle VM virtual box manager	11
3.1	UBUNTU Home Screen	15
3.2	Opening Terminal in UBUNTU	17
5.1	ADD user	24
5.2	DEVSTACK installation	25
5.3	Installing required package	25
5.4	Download package	25
5.5	Nona Local.conf	26
5.6	Run the Devstack	27
5.7	Result of Devstack	27
5.8	Login page of openstack	28
6.1	Openstack	30
6.2	Openstack dashboard	31

CHAPTER 1

INTRODUCTION

Octavia is an open-source load balancer service that provides high availability to OpenStack cloud infrastructure. Devstack is a set of scripts used to deploy an OpenStack environment on a single machine. The combination of Octavia load balancer and Devstack provides a powerful tool for balancing traffic in cloud environments. In this paper, we will discuss the implementation of Octavia load balancer in Devstack of OpenStack. We will provide an overview of the installation and configuration process of Devstack and Octavia, as well as the creation of load balancing pools, listeners, and members. We will also explore the benefits of using Octavia load balancer in Devstack, such as increased performance, availability, and scalability of services, as well as automatic failover. Overall, the implementation of Octavia load balancer in Devstack is a valuable addition to any OpenStack cloud infrastructure, and this paper will provide a comprehensive guide to its setup and usage.

Octavia is an open-source load balancer service that is designed to provide high availability to OpenStack cloud infrastructure. It is a scalable and feature-rich load balancer that can handle a large amount of traffic and distribute it evenly across multiple servers or instances. It can also perform SSL termination, health checks, and automatic failover, ensuring the smooth and uninterrupted functioning of the applications or services. Octavia load balancer is an essential component of OpenStack cloud infrastructure as it helps improve the performance, reliability, and scalability of applications and services. It provides a flexible and powerful tool for balancing traffic, ensuring high availability, and enhancing the user experience. As an open-source project, Octavia is free to use and can be customized to fit the needs of any organization or project.

To implement Octavia Loadbalancer in OpenStack Devstack, you first need to ensure that you have the Devstack environment set up with OpenStack installed. Once you have a working Devstack environment.

Steps to Implement Octavia Load balancer:

Add Octavia service to Devstack:

You can add Octavia service to Devstack by updating the local.conf file. You need to add the following lines to your local.conf file:

enable_plugin octavia <https://opendev.org/openstack/octavia/stable/rocky>

Run Devstack:

After adding the Octavia service to Devstack, you need to run Devstack to install the Octavia service. You can run Devstack by executing the following command: Arduino
./stack.sh

This command will install all the OpenStack services along with Octavia.

Verify Octavia installation:

Once the installation is complete, you need to verify that Octavia is installed and running correctly. You can verify the installation by executing the following command: openstack load balancer list

This command should return a list of load balancers, indicating that Octavia is installed and running correctly.

Create a load balancer:

To create a load balancer, you need to execute the following command:

Css :

```
openstack loadbalancer create --name my-lb --vip-subnet-id <subnet-id>
```

This command will create a load balancer with the name "my-lb" and assign it to the specified subnet.

Add a listener:

Once the load balancer is created, you need to add a listener to it. A listener is a component that listens for incoming traffic and forwards it to the appropriate backend pool. To add a listener, you need to execute the following command:

Css :

```
openstack loadbalancer listener create --name my-listener --protocol HTTP --  
protocol-port 80 --loadbalancer my-lb --default-pool my-pool
```

This command will create a listener named "my-listener" that listens for HTTP traffic on port 80 and forwards it to the "my-pool" backend pool.

Add a pool:

Finally, you need to add a pool to the load balancer. A pool is a group of backend servers that are responsible for handling the traffic forwarded by the listener. To add a pool, you need to execute the following command:

Css :

```
openstack loadbalancer pool create --name my-pool --lb-algorithm ROUND_ROBIN -  
-protocol HTTP.
```

This command will create a pool named "my-pool" that uses the ROUND_ROBIN load balancing algorithm to distribute HTTP traffic to the backend servers.

With these steps, you can implement Octavia Loadbalancer in OpenStack Devstack and start using it to distribute traffic to your applications.

1.1 Specification

1.1.1 Hardware Configuration

1. Processor-I5
2. Speed-3GHz
3. RAM - 8GB(min)
4. Hard Disk -500GB
5. KeyBoard -Standard Windows Keyboard
6. Mouse - Two or Three Button Mouse
7. Monitor- LCD
8. Ethernet connection:(LAN)

1.1.2 Software Configuration

1. Windows os
2. VM virtual box
3. Ubuntu os

CHAPTER 2

CLOUD COMPUTING AND VM VIRTUAL BOX

2.1 CLOUD COMPUTING

In the context of technology, "cloud" refers to a network of remote servers that are hosted on the internet and used to store, manage, and process data. These servers are typically operated by third-party service providers, who offer their resources to businesses and individuals who need them.

Cloud computing allows users to access and use computing resources on-demand, without needing to own or maintain their own hardware or software. This means that businesses and individuals can scale their computing resources up or down as needed, without having to invest in expensive infrastructure.

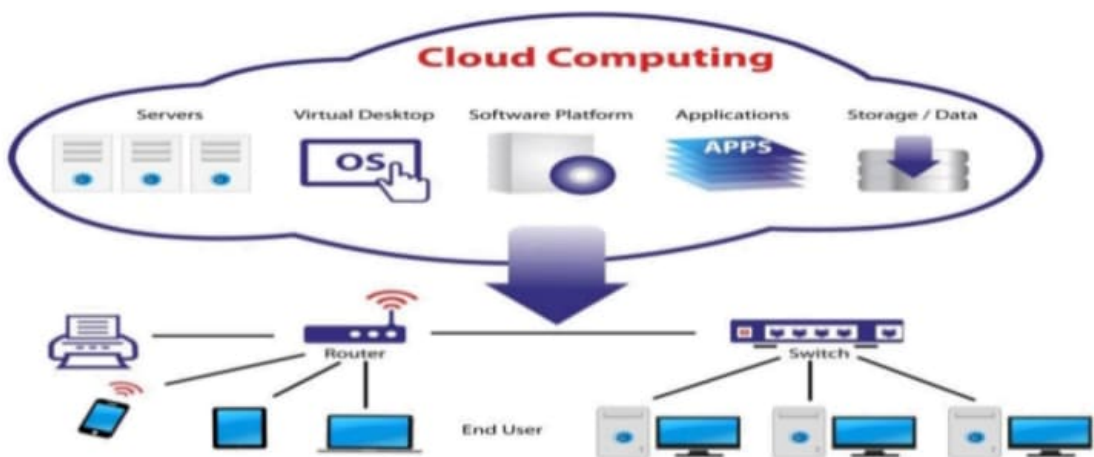


Fig 2.1 Cloud Computing

2.2 MULTICLOUD AND HYBRID MULTICLOUD

Multicloud is the use of two or more clouds from two or more different cloud providers. Having a multicloud environment can be as simple using email SaaS from one vendor and image editing SaaS from another. But when enterprises talk about multicloud, they're typically talking about using multiple cloud services including SaaS, PaaS, and IaaS services from two or more of the leading public cloud providers.

Hybrid multicloud is the use of two or more public clouds together with a private cloud environment.

Organizations choose multicloud to avoid vendor lock-in, to have more services to choose from, and to access to more innovation. But the more clouds you use each with its own set of management tools, data transmission rates, and security protocols

the more difficult it can be to manage your environment. Multicloud management platforms provide visibility across multiple provider clouds through a central dashboard, where development teams can see their projects and deployments, operations teams can keep an eye on clusters and nodes, and the cyber security staff can monitor for threats.

2.3 CLOUD COMPUTING ARCHITECTURE

Cloud computing architecture refers to the components and sub-components required for cloud computing.

These components typically refer to:

1. Front end(fat client, thin client)
2. Back-end platforms(servers, storage)
3. Cloud-based delivery and a network(Internet, Intranet, Intercloud).

Hosting a cloud: There are three layers in cloud computing. Companies use these layers based on the service they provide.

1. Infrastructure
2. Platform
3. Application

2.4 TYPES OF CLOUD

There are several different types of cloud services, including:

Infrastructure as a Service (IaaS) - which provides access to computing resources such as virtual machines, storage, and networking.

Platform as a Service (PaaS) - which provides a platform for developers to build and deploy their applications, without needing to manage the underlying infrastructure.

Software as a Service (SaaS) - which provides access to software applications over the internet, without needing to install or maintain the software locally.

2.4.1. INFRASTRUCTURE AS A SERVICE (IAAS) :

IaaS stands for Infrastructure as a Service, which is a type of cloud computing service that provides users with access to virtualized computing resources over the internet.

With IaaS, users can access and manage virtualized computing resources such as servers, storage, and networking components, without needing to own or maintain their own physical hardware. Instead, the IaaS provider manages the underlying

infrastructure, while users have control over the operating systems, applications, and storage they deploy on the virtual machines.

IaaS allows users to quickly and easily scale up or down their computing resources as needed, without having to invest in expensive hardware. This makes it a flexible and cost-effective option for businesses and individuals who need access to computing resources for a variety of purposes, such as hosting websites or running complex applications.

Some popular examples of IaaS providers include Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform.

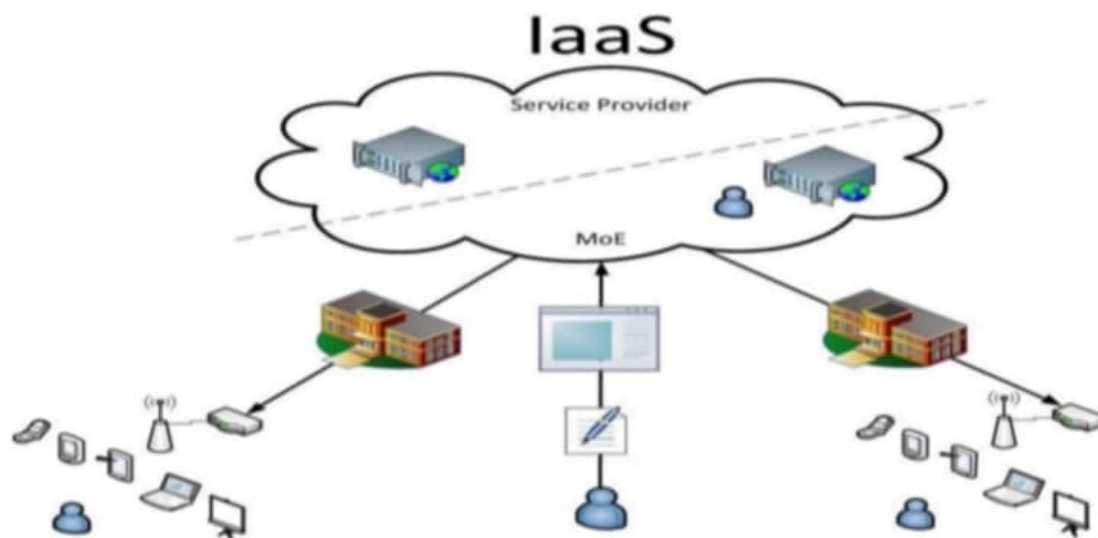


Fig 2.4.1 Infrastructure As a Service

2.4.2. PLATFORM AS A SERVICE (PAAS) :

PaaS stands for Platform as a Service, which is a type of cloud computing service that provides users with a platform for developing, deploying, and managing their applications over the internet.

With PaaS, users can access a pre-configured platform that includes all the necessary infrastructure, middleware, and tools needed to develop and deploy applications, without needing to manage the underlying hardware or software.

PaaS providers typically offer a range of services, such as application hosting, development tools, database management, and integration with other cloud services. Users can access these services through a web-based interface, API, or command line interface, depending on the provider.

PaaS allows users to focus on developing and deploying their applications, without having to worry about the underlying infrastructure. This makes it a popular choice

for developers who want to quickly and easily create and deploy applications, without needing to invest in expensive hardware or software.



Fig 2.4.2 Platform As a Service

2.4.3. SOFTWARE AS A SERVICE (SAAS) :

SaaS stands for Software as a Service, which is a type of cloud computing service that provides users with access to software applications over the internet.

With SaaS, users can access and use software applications hosted by a third-party provider, without needing to install or maintain the software locally. Users typically access the software through a web browser or mobile app, and pay a subscription fee to the provider for access.

SaaS providers manage the underlying infrastructure, including servers, storage, and networking, as well as software updates and maintenance. This means that users can focus on using the software to accomplish their goals, without having to worry about the underlying technology.

SaaS is a popular choice for businesses and individuals who need access to software applications for a variety of purposes, such as productivity tools, customer relationship management (CRM) software, and accounting software. It allows users to quickly and easily access the software they need, without having to invest in expensive hardware or software licenses.



Fig 2.4.3 Software As a Service

2.5 ADVANTAGES OF CLOUD COMPUTING

Cloud computing offers several advantages, some of which include:

Scalability: Cloud computing allows for easy scaling up or down of computing resources as per the needs of the organization or user. This flexibility helps to optimize resources and reduce costs.

Cost savings: Cloud computing can reduce IT costs by eliminating the need for expensive hardware and software investments, maintenance, and upgrades. It offers a pay-as-you-go model where users only pay for the resources they use, which helps to minimize waste.

Accessibility: Cloud computing services are available over the internet, making it easy for users to access their applications and data from anywhere, anytime, and from any device.

Disaster recovery: Cloud computing offers an easy and affordable way to back up data, applications, and systems. In the event of a disaster or data loss, users can quickly restore their data from the cloud.

Collaboration: Cloud computing enables collaboration among team members, regardless of their location. It offers a centralized location for sharing files and documents, which improves team productivity and efficiency.

Security: Cloud computing providers offer robust security measures to protect user data, applications, and systems from cyber threats. They also ensure compliance with industry standards and regulations. Virtualization is achieved through the use of

specialized software, called a hypervisor.

Innovation: Cloud computing enables organizations to quickly adopt new technologies and innovations without the need for large investments in hardware and software. This helps to foster innovation and competitive advantage.

2.6 DISADVANTAGE OF CLOUD COMPUTING

Cloud computing also has some disadvantages, some of which include:

Dependency on Internet Connectivity: Cloud computing requires an internet connection to access the services and data stored in the cloud. In the absence of a stable internet connection, the user may experience downtime or loss of access to their data.

Security Risks: Storing data in the cloud creates security risks, such as data breaches and cyber attacks. Despite the robust security measures put in place by cloud providers, these risks cannot be entirely eliminated.

Vendor Lock-in: Moving data and applications from one cloud provider to another can be challenging and costly. This can create a dependency on the cloud provider and limit the user's flexibility in terms of choosing other providers.

Limited Control: Cloud computing services are managed and operated by the cloud provider, and users have limited control over the infrastructure and management of their data and applications.

Cost: While cloud computing offers cost savings, it can also be expensive, especially when users require more resources or services than initially anticipated. Users may also incur additional costs for data storage, bandwidth, and support services.

Compatibility: Some legacy applications or systems may not be compatible with cloud computing, requiring modifications or upgrades to make them compatible.

Performance Issues: The performance of cloud computing services is dependent on the internet connection and the quality of service provided by the cloud provider. Users may experience performance issues if the provider experiences downtime, maintenance, or high traffic.

2.7 VIRTUAL BOX

VirtualBox is a free and open-source virtualization software that enables users to run multiple operating systems on a single physical computer. It was developed by Oracle and is available for Windows, Mac OS X, Linux, and Solaris.

VirtualBox works by creating a virtual machine (VM) on the host computer, which acts as a separate computer system with its own hardware resources, including CPU, RAM, storage, and network interfaces. Users can install and run any operating system, such as Windows, Linux, or macOS, on the virtual machine, just as they would on a physical computer.

VirtualBox provides a range of features, including support for 3D graphics, USB device integration, and snapshots, which allow users to save the state of the virtual machine and restore it later. It also supports different networking modes, allowing users to connect virtual machines to the internet, local networks, or isolated networks.

VirtualBox is a popular tool for developers, system administrators, and IT professionals, as it enables them to test and run multiple operating systems and applications on a single physical computer. It is also used by individuals who want to run legacy applications or experiment with different operating systems without affecting their primary system.



Fig:2.7 VM Virtual Box



Fig:2.7.1 Oracle VM Virtual Box Manager

2.8 PROS AND CONS OF VM VIRTUAL BOX

Pros:

1. **Cost-effective:** VirtualBox is free and open source, which makes it an attractive option for individuals and small businesses that want to use virtualization technology without incurring additional expenses.
2. **Easy to use:** VirtualBox has a user-friendly interface and is relatively easy to set up and configure. It also supports a wide range of operating systems and virtual devices, making it a versatile option for many users.
3. **Resource-efficient:** VirtualBox uses minimal system resources, which means it can run on almost any hardware configuration. This makes it a good option for users who want to run virtual machines on older or less powerful hardware.
4. **Flexible:** VirtualBox allows users to create and configure virtual machines with a high degree of flexibility, including customizing hardware settings and network configurations.
5. **Snapshot and cloning:** VirtualBox allows users to take snapshots of their virtual machines, which can be useful for testing and troubleshooting purposes. Users can also clone virtual machines to create multiple copies with the same configuration.

Cons:

1. **Performance:** VirtualBox may not perform as well as other virtualization

solutions, such as VMware or Hyper-V, especially when running resource-intensive applications.

2. **Limited features:** VirtualBox has fewer advanced features compared to other virtualization solutions, such as support for nested virtualization, which can limit its functionality for some users.
3. **Security:** VirtualBox may pose security risks if not properly configured. For example, if the virtual machines are not isolated from the host system, malicious code or viruses may be able to spread between the two systems.
4. **Support:** While VirtualBox has an active user community, official support from Oracle is limited, which may make it difficult to get help with more complex issues.
5. **Updates:** VirtualBox updates are not always timely or frequent, which can result in compatibility issues with newer operating systems or hardware.

CHAPTER 3

UBUNTU AND ITS FEATURES

3.1 UBUNTU

Ubuntu is a free and open-source Linux-based operating system that is widely used for desktop, server, and cloud computing applications. It is developed by Canonical and is based on the Debian architecture. Ubuntu is known for its user-friendly interface and ease of use, making it popular among users who are new to Linux or open-source software. It is also highly customizable and supports a range of software and development tools, including Python, Java, and Ruby.

Ubuntu provides a wide range of applications and tools, including LibreOffice, Firefox, and the GNOME desktop environment. It also supports a wide range of hardware, making it suitable for use on a variety of devices, from desktops and laptops to servers and IoT devices. Ubuntu releases new versions every six months, with Long Term Support (LTS) versions released every two years. LTS versions are supported for five years, while regular versions are supported for nine months. Ubuntu has a strong community of users and developers who contribute to its development and support. It is widely used in enterprise environments, as well as by individuals and organizations who value open-source software and the benefits it provides.

Ubuntu has a software center that makes it easy to find and install applications. The software center has a wide range of applications, including popular ones like LibreOffice, Firefox, and VLC media player. Ubuntu has built-in security features, including the AppArmor security framework, which helps protect against malware and other security threats. Ubuntu has a large and active community of users and developers who provide support and contribute to its development. This community provides a wealth of resources, including forums, documentation, and tutorials. Ubuntu is highly customizable, allowing users to personalize their experience and tailor it to their needs. This includes the ability to customize the desktop environment, themes, and icons.

3.2 INSTALLIZATION UBUNTU IN VM BOX

1. To install Ubuntu in VirtualBox, please follow these steps:
2. Download the Ubuntu ISO file from the official website:
<https://ubuntu.com/download>

3. Open VirtualBox and click on the "New" button to create a new virtual machine.
4. Give your virtual machine a name, select "Linux" as the type, and "Ubuntu" as the version.
5. Allocate the amount of RAM you want to assign to the virtual machine.
6. Create a virtual hard disk for the virtual machine. You can select the default settings or customize them based on your requirements.
7. After creating the virtual machine, select it and click on the "Settings" button.
8. In the "Settings" window, select "Storage" and click on the "Empty" CD/DVD icon.
9. Click on the "Add" button and select the Ubuntu ISO file you downloaded earlier.
10. Click on the "Start" button to start the virtual machine.
11. Ubuntu will start booting from the ISO file. Follow the on-screen instructions to complete the installation process.
12. Once the installation is complete, you can use Ubuntu in VirtualBox.

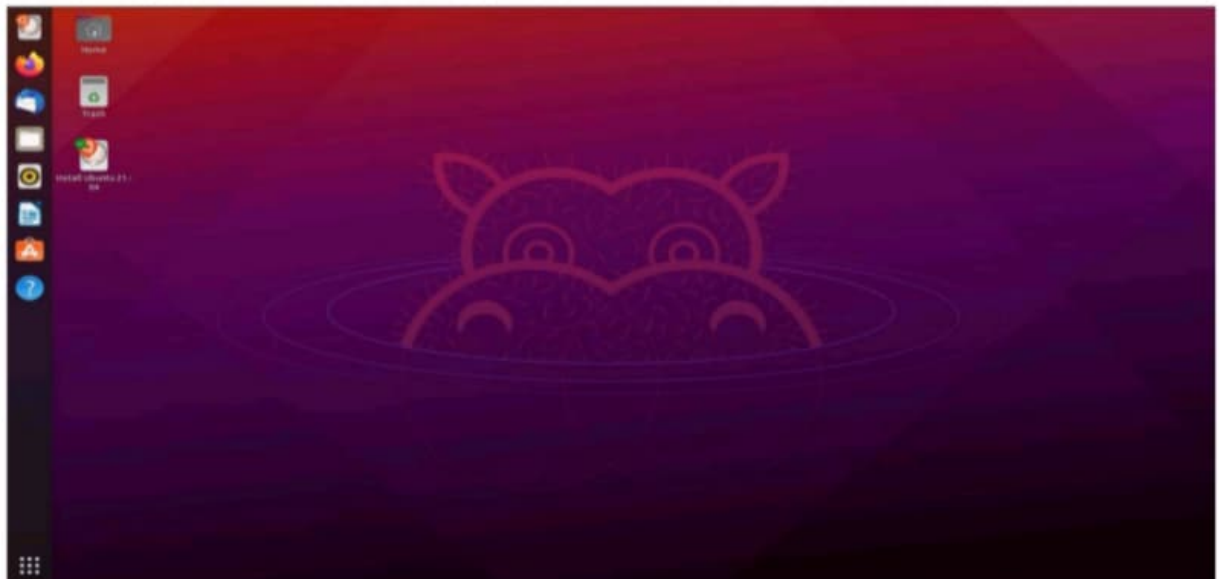


Fig 3.2 Installization Ubuntu in VM Box

3.3 FEATURES OF UBUNTU

Ubuntu has a wide range of features that make it a popular choice for personal and professional use. Here are some of its key features:

Open-source: Ubuntu is an open-source operating system, which means that the source code is available to the public, and anyone can modify and distribute it.

User-friendly interface: Ubuntu has a simple and intuitive user interface, which makes it easy to use even for beginners.

Security: Ubuntu has a strong focus on security, and it includes built-in security features such as AppArmor and a firewall.

Stability: Ubuntu is known for its stability and reliability, and it includes features such as error reporting and automatic updates.

Package management: Ubuntu has a powerful package management system that allows users to easily install, update, and remove software packages from a centralized repository.

Multilingual support: Ubuntu supports a wide range of languages, making it accessible to users from different regions of the world.

Customizability: Ubuntu can be customized to suit individual needs, with a range of desktop environments, themes, and software packages available.

Compatibility: Ubuntu is compatible with a wide range of hardware and software, making it a versatile operating system.

Community support: Ubuntu has a large and active community of users and developers who provide support, documentation, and software updates.

Overall, Ubuntu is a feature-rich and versatile operating system that is suitable for a wide range of applications and use cases.

3.4 WORK FOR UBUNTU

If you're interested in working for Ubuntu, there are a few ways you can get involved:

Join the Ubuntu community: The Ubuntu community is a large and active group of users and developers who contribute to the development of the operating system. By joining the community, you can participate in discussions, contribute code, and help with testing and bug reporting.

Apply for a job at Canonical: Canonical is the company that develops and maintains Ubuntu. They often have job openings for developers, engineers, and other positions. You can visit their website to see current job openings and apply online.

Contribute to Ubuntu projects: Ubuntu has a wide range of projects that you can contribute to, such as documentation, translations, and software development. By contributing to these projects, you can help improve the operating system and gain experience working with Ubuntu.

Start your own Ubuntu-related project: If you have an idea for an Ubuntu-related

project, you can start your own project and contribute to the Ubuntu ecosystem. This could be anything from a new software application to a hardware project that runs on Ubuntu.

Overall, there are many ways to get involved with Ubuntu and contribute to its development. Whether you're a developer, designer, or enthusiast, there are opportunities to help improve Ubuntu and make it even better.

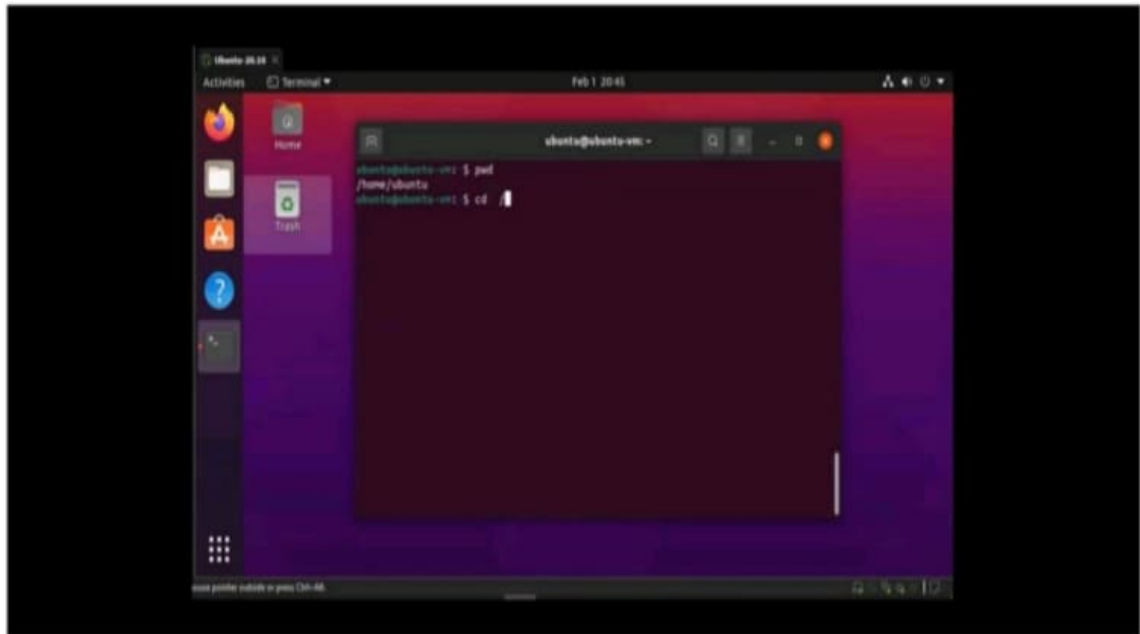


Fig:3.4 Opening Terminal in Ubuntu

3.5 Pros and Cons

Pros:

1. **Free and open-source:** Ubuntu is free to download and use, and its open-source nature means that the code is transparent and can be modified and shared.
2. **User-friendly:** Ubuntu has a user-friendly interface and is relatively easy to set up and use, making it a good option for both new and experienced users.
3. **Wide range of software:** Ubuntu has access to a wide range of open-source software that can be installed with a few clicks. This makes it a versatile option for many users, from web developers to gamers.
4. **Stable:** Ubuntu is known for its stability, with regular updates and security patches that help to keep the system running smoothly.

5. **Customizable:** Ubuntu can be customized to suit individual user preferences, from the desktop environment to the software installed.

Cons:

1. **Compatibility issues:** While Ubuntu has a wide range of software available, not all software is compatible with Linux. This can be a problem for users who rely on specific programs or applications that are only available for Windows or macOS.
2. **Limited support:** While Ubuntu has an active community of users who can offer help and advice, official support from Canonical, the company behind Ubuntu, is limited and may not be sufficient for more complex issues.
3. **Learning curve:** Ubuntu may require a bit of a learning curve for users who are not familiar with Linux or command-line interfaces. This can be a barrier for some users who are used to more user-friendly operating systems.
4. **Hardware compatibility:** Ubuntu may not be compatible with all hardware configurations, especially older or less common hardware. This can make it difficult to use Ubuntu on some systems.
5. **Updates:** While Ubuntu's regular updates are a pro for stability and security, they can also cause compatibility issues with some hardware or software configurations. Users may need to check for updates and troubleshoot any issues that arise after an update.

CHAPTER 4

DEVSTACK AND ITS FEATURES

4.1 DEVSTACK

DevStack is a tool for deploying OpenStack, an open-source cloud computing platform, on a single machine for development and testing purposes. It provides an easy and efficient way to set up a complete OpenStack environment on a local machine or a virtual machine. DevStack is maintained by the OpenStack community and is designed to be used by developers, testers, and users who want to experiment with OpenStack without the need for a full-scale deployment. It can be used to test new features, develop applications that integrate with OpenStack, and learn about the platform.

DevStack installs and configures all the necessary components of OpenStack, including Keystone (identity service), Glance (image service), Nova (compute service), Neutron (networking service), and Horizon (dashboard). It also supports a range of optional components, including Swift (object storage), Cinder (block storage), and Heat (orchestration). DevStack is designed to be easy to install and configure, with a simple script that automates the process. It is also highly customizable, allowing users to configure various aspects of the OpenStack environment, such as the networking topology, the number of compute nodes, and the storage backend.

Overall, DevStack is a powerful tool for deploying and testing OpenStack, providing a simple and efficient way to set up a complete OpenStack environment on a single machine. It is widely used by developers and testers in the OpenStack community and is a valuable tool for anyone interested in learning about or working with OpenStack.

4.2 ADVANTAGES OF DEVSTACK

DevStack is an open-source project designed to provide developers with a complete, pre-configured development environment for OpenStack. Here are some advantages of using DevStack:

Easy to Install: DevStack is easy to install, and it can be installed on a single machine or on multiple machines.

Pre-Configured Environment: DevStack provides a pre-configured development environment for OpenStack, which saves developers time and effort.

Flexibility: DevStack is highly configurable, which allows developers to customize their development environment according to their needs.

Supports Multiple Services: DevStack supports multiple OpenStack services, including Nova, Swift, Glance, and Keystone, among others.

Open-Source: DevStack is an open-source project, which means that developers can modify and contribute to the project as needed.

Testing and Debugging: DevStack is an excellent tool for testing and debugging OpenStack services, as it provides a complete environment for developers to test their code.

Community Support: DevStack has a large community of developers who provide support and contribute to the project, which helps to ensure that the project is up-to-date and reliable.

4.3 DISADVANTAGES OF DEVSTACK

Here are some potential disadvantages of using DevStack:

Resource Intensive: DevStack requires a significant amount of resources to run, including disk space, memory, and CPU power. This may make it difficult to run on low-end hardware.

Complex Configuration: While DevStack is highly configurable, this also means that the configuration process can be complex and time-consuming, particularly for developers who are new to the project.

Limited Scalability: DevStack is designed for use in a development environment and may not be suitable for large-scale production deployments.

Limited Security: DevStack is designed to be easy to set up and use, which means that it may not include all of the security features necessary for production environments.

Dependency Conflicts: DevStack may have dependency conflicts with other software installed on the same machine, which can cause issues during the installation process.

Limited Platform Support: DevStack is primarily designed to run on Linux systems, which may limit its use for developers working on other platforms.

It's worth noting that some of these disadvantages may not be significant issues for all developers, and that many of them can be addressed with careful planning and configuration.

4.4 LIMITATIONS OF DEVSTACK

Here are some limitations of using DevStack:

Limited Customization: While DevStack is highly configurable, it may not support all of the customization options that some developers require.

Compatibility Issues: DevStack may not be compatible with all versions of OpenStack, which can limit its usefulness for developers who need to work with older or newer versions of the platform.

Lack of Documentation: While DevStack has extensive documentation, some parts of the project may not be well-documented, which can make it difficult for developers to troubleshoot issues or customize the environment.

Limited Community Support: While DevStack has a large community of developers, some issues may not be well-supported due to the limited resources of the community.

Performance Issues: DevStack may not be optimized for performance, which can result in slower performance compared to other OpenStack development environments.

Resource Constraints: DevStack may require a significant amount of resources, including memory, disk space, and CPU power, which can be a limitation for developers working on machines with limited resources.

Limited Compatibility with Other Tools: DevStack may not be compatible with other tools and platforms that some developers require for their workflows, which can limit its usefulness for those developers.

4.5 WORK FOR DEVSTACK

Devstack is a powerful tool that allows you to quickly set up an OpenStack environment for development and testing purposes. As a developer or operator, you can use Devstack to test your applications and configurations in a local environment before deploying them to a production environment.

If you are working with Devstack, some of the tasks you may need to perform include:

Setting up Devstack:

You will need to download and install Devstack on your local machine. You can follow the instructions on the Devstack website to get started.

Configuring Devstack:

Once you have Devstack installed, you will need to configure it to work with your OpenStack environment. This may involve setting up networking, storage, and other services.

Starting Devstack:

You can start Devstack by running the "stack.sh" script. This script will install and configure all the necessary services and components for your OpenStack environment.

Testing applications:

Once you have Devstack up and running, you can start testing your applications in a local environment. This can help you identify and fix issues before deploying your applications to a production environment.

Debugging issues:

If you encounter issues while working with Devstack, you will need to debug them to determine the root cause. This may involve reviewing logs, configuration files, and other system information.

Upgrading Devstack:

As new versions of OpenStack are released, you may need to upgrade your Devstack environment to take advantage of new features and improvements. You can follow the Devstack upgrade guide to upgrade your environment.

Overall, working with Devstack involves setting up and configuring an OpenStack environment, testing applications, and debugging issues. With Devstack, you can quickly iterate on your applications and configurations, reducing the time and effort required to deploy your applications to a production environment.

4.6 SYSTEM REQUIREMENTS

4.6.1 WINDOWS

1. Microsoft®Windows®7/8/1064-bit
2. 4GB RAMminimum,8GBRAMrecommended
3. GBofavailablediskspaceminimum,
4. GBRecommended500MBforIDE+1.5GBforAndroidSDKandemulatorsyste
mimage

5. 1280x800 minimum screen resolution.

4.6.2 UBUNTU

1. 2Ghz dual core processor
2. 8Gib RAM (system memory)
3. 25 GB of hard drive space (or USB stick, memory card or external drive but see Live CD for an alternative approach)
4. VGA capable of 1024x768 screen resolution
5. Either a CD/DVD drive or a USB port for the installer media
6. Internet access is helpful

CHAPTER 5

PROJECT IMPLEMENTATION

5.1 CREATING A STACK USER

1. Add user.
2. We need to add a user to install DevStack.
3. adduser stack.

```
root@Openstack:~# adduser stack
Adding user 'stack' ...
Adding new group 'stack' (1001) ...
Adding new user 'stack' (1001) with group 'stack' ...
Creating home directory '/home/stack' ...
Copying files from '/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for stack
Enter the new value, or press ENTER for the default
    Full Name []: Openstack
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
root@Openstack:~#
```

FIG:5.1 Add User

5.2 DEVSTACK INSTALLATION

- Since this user will be making many changes to your system, it will need to have sudo privileges:
- `echo "stack ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers`

```
root@Openstack:~# echo "stack ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers
root@Openstack:~#
```

Fig:5.2 Devstack Installation

5.3 INSTALLING REQUIRED PACKAGE

1. sudo
2. Git

```
root@Openstack:~# apt-get install git sudo
Reading package lists... Done
Building dependency tree
Reading state information... Done
sudo is already the newest version (1.8.21p2-3ubuntu1.2).
sudo set to manually installed.
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs
  git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 251 not upgraded.
Need to get 4,741 kB of archives.
After this operation, 34.0 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Fig:5.3 Installing Required Package

5.4 DOWNLOAD DEVSTACK

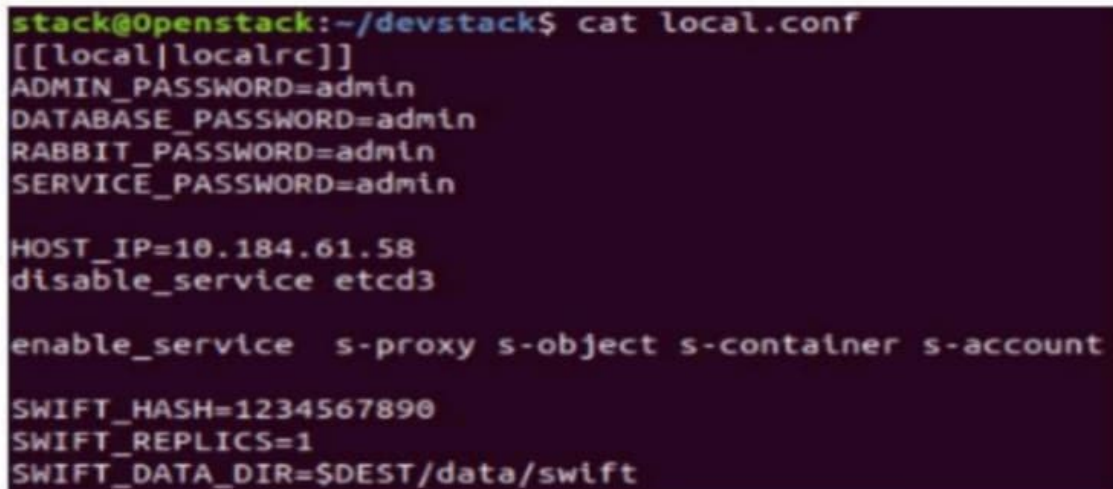
1. `git clone https://opendev.org/openstack/devstack -b stable/ussuri`
2. `cd devstack`

```
cloud@openstack:~$ git clone https://opendev.org/openstack/devstack -b stable/ussuri
Cloning into 'devstack'...
remote: Enumerating objects: 46244, done.
remote: Counting objects: 100% (46244/46244), done.
remote: Compressing objects: 100% (21131/21131), done.
remote: Total 46244 (delta 32691), reused 37526 (delta 24408)
Receiving objects: 100% (46244/46244), 9.51 MiB | 1.41 MiB/s, done.
Resolving deltas: 100% (32691/32691), done.
cloud@openstack:~$
```

Fig:5.4 Downloading DevStack

5.5.NANO LOCAL.CONF

1. In the context of Ubuntu terminal, the "nanolocal.conf" command is likely referring to using the nano text editor.
2. It is used to edit the "local.conf" file in the openstack configuration directory.
3. This file typically contains configuration settings for various openstack services.
4. Make the necessary changes to the file using the nano editor.
5. Once you have finished making changes, save the file by pressing ctrl + o and then exit the editor by pressing ctrl + x.
6. Set the administrative password. This password is used for the admin and demo accounts set up as OpenStack users.
7. Set the MySQL administrative password. The default here is a random hex string which is inconvenient if you need to look at the database directly for anything.
8. Set the RabbitMQ password. Set the service password. This is used by the OpenStack services (Nova, Glance, etc) to authenticate with Keystone.

A terminal window with a dark purple background. The prompt is 'stack@Openstack:~/devstack\$'. The command 'cat local.conf' has been executed, displaying the contents of the file. The configuration includes administrative passwords, host IP, service enablement, and Swift settings.

```
stack@Openstack:~/devstack$ cat local.conf
[[local|localrc]]
ADMIN_PASSWORD=admin
DATABASE_PASSWORD=admin
RABBIT_PASSWORD=admin
SERVICE_PASSWORD=admin

HOST_IP=10.184.61.58
disable_service etcd3

enable_service s-proxy s-object s-container s-account

SWIFT_HASH=1234567890
SWIFT_REPLICS=1
SWIFT_DATA_DIR=$DEST/data/swift
```

Fig:5.5 Nano Local.Conf

5.6 RUN DEVSTACK

- ./stack.sh

```
++ declare -r -g _DEVSTACK_FUNCTIONS=1
++++ dirname /home/stack/devstack/functions
+++ cd /home/stack/devstack
+++ pwd
++ FUNC_DIR=/home/stack/devstack
++ source /home/stack/devstack/functions-common
++++ set +o
++++ grep xtrace
+++ _XTRACE_FUNCTIONS_COMMON='set -o xtrace'
+++ set +o xtrace
++ source /home/stack/devstack/inc/ini-config
++++ set +o
++++ grep xtrace
+++ INC_CONF_TRACE='set -o xtrace'
+++ set +o xtrace
++ source /home/stack/devstack/inc/meta-config
++++ set +o
++++ grep xtrace
+++ _XTRACE_INC_META='set -o xtrace'
+++ set +o xtrace
```

Fig:5.6 Run the Devstack

```
=====
DevStack Component Timing
=====
Total runtime                4321

run_process                  146
test_with_retry              12
apt-get-update               62
pip_install                  734
restart_apache_server        28
wait_for_service             82
git_timed                    240
apt-get                      29
=====
```

Fig:5.7 Result of DevStack

CHAPTER 6

OPENSTACK AND ITS FEATURES

6.1 OPENSTACK

OpenStack is an open-source cloud computing platform that allows users to manage and control resources, such as compute, storage, and networking, through a web-based dashboard. It was first launched in 2010 as a joint project between Rackspace Hosting and NASA. OpenStack is designed to be modular and scalable, and can be used to build public or private clouds. It consists of several core services, including Nova for compute, Cinder for block storage, Neutron for networking, Glance for image management, and Keystone for authentication and authorization. OpenStack is often used by organizations that require flexibility and control over their cloud infrastructure, as it allows them to customize and manage their resources to meet specific requirements. Additionally, being open-source, it provides a high degree of transparency and encourages collaboration among developers.



Fig:6.1 OpenStack

Private cloud: OpenStack can be used to build a private cloud infrastructure for organizations to manage their own resources, applications and data.

Public cloud: OpenStack can be used by service providers to build and offer public cloud services to customers.

Hybrid cloud: OpenStack can be used to build a hybrid cloud infrastructure that combines private and public cloud resources, allowing organizations to scale up or down as needed.

DevOps: OpenStack can be used to provide developers and IT teams with on-demand access to infrastructure resources, enabling faster application development and testing.

Big Data: OpenStack can be used to provide scalable and flexible infrastructure for big data processing, storage and analysis.

Network function virtualization (NFV): OpenStack can be used to build an NFV infrastructure that allows for virtualization of network functions, such as firewalls and load balancers.

High-performance computing (HPC): OpenStack can be used to provide high-performance computing infrastructure, which can be used for scientific research and simulations.

Overall, OpenStack is a versatile and flexible platform that can be used for a wide range of cloud computing use cases.

6.2 WORK FOR OPENSTACK

Here are some of the main components and work of OpenStack:

Compute (Nova): This component provides the virtualization layer and manages the creation, scheduling, and termination of virtual machines (VMs) on the cloud infrastructure.

Storage (Cinder and Swift): These components provide different types of storage services. Cinder provides block storage, which is used to store persistent data for VMs, while Swift provides object storage, which is used for storing large volumes of unstructured data.

Networking (Neutron): This component provides network connectivity for the VMs and enables the creation of virtual networks, routers, and firewalls.

Identity (Keystone): This component provides authentication and authorization services and manages user and project information for the OpenStack environment.

Dashboard (Horizon): This component provides a web-based graphical user interface (GUI) for managing OpenStack resources.

Orchestration (Heat): This component provides a framework for automating the deployment and scaling of OpenStack resources.

Telemetry (Ceilometer): This component provides monitoring and metering services to track resource usage and performance.

OpenStack works by providing a set of APIs that enable users to interact with the various OpenStack components and manage their cloud resources. Users can deploy OpenStack on their own hardware or use a public cloud service that runs on OpenStack. OpenStack is highly flexible and can be customized to meet the specific needs of different organizations, making it a popular choice for building and managing cloud infrastructure.

The image shows the OpenStack login interface. At the top, there is the OpenStack logo, which consists of a red square with a white 'O' inside, followed by the word 'openstack' in a lowercase, sans-serif font. Below the logo, the text 'Log in' is displayed. Underneath, there are two input fields: the first is labeled 'User Name' and the second is labeled 'Password'. The 'Password' field has a small eye icon to its right, indicating a toggle for password visibility. At the bottom right of the form, there is a blue button with the text 'Sign in' in white.

Fig:6.2 Login Page of OpenStack

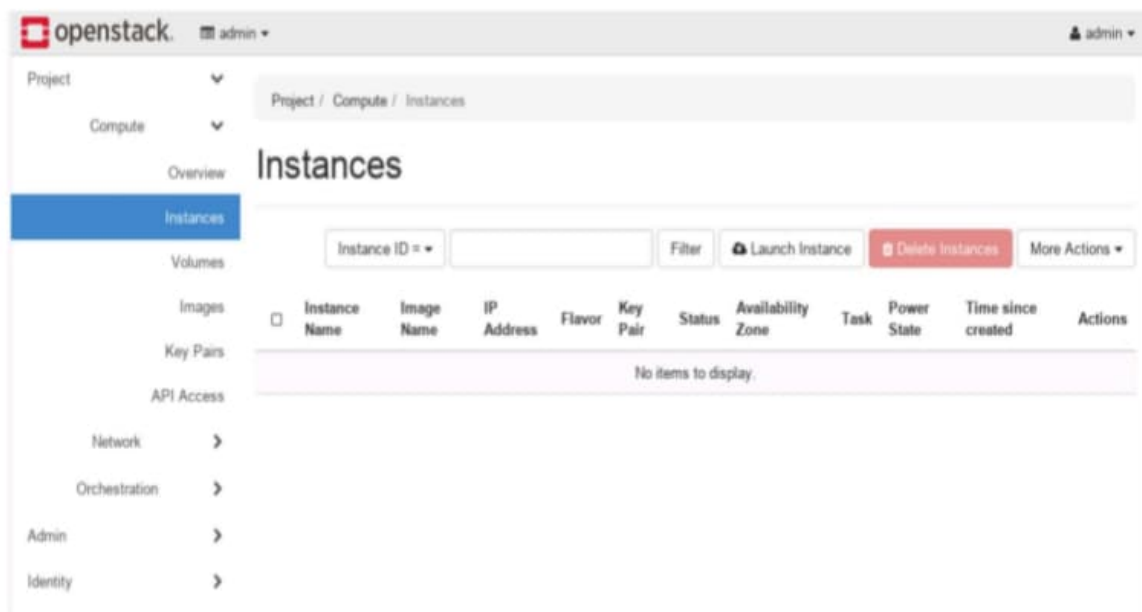


Fig 6.2.1 OpenStack Dashboard

6.3 ADVANTAGES OF OPENSTACK

OpenStack is an open-source cloud computing platform that provides a range of benefits to organizations looking to deploy and manage private or public clouds. Some of the key benefits of OpenStack include:

Flexibility: OpenStack is a highly customizable cloud computing platform that enables organizations to configure and deploy their cloud environment to meet their specific needs.

Scalability: OpenStack enables organizations to scale their cloud computing resources up or down, as needed, to meet changing demand.

Cost-Effective: OpenStack is an open-source platform, which means it is free to use, modify and distribute. This makes it a cost-effective solution for organizations that want to deploy private or public clouds.

Open Standards: OpenStack adheres to open standards, which makes it easier for organizations to integrate with other systems and services.

High Availability: OpenStack provides high availability and fault tolerance features that ensure the cloud infrastructure remains available even if some components fail.

Multi-Tenancy: OpenStack provides multi-tenancy features that enable multiple users or groups to share the same cloud infrastructure while maintaining isolation and security.

API Compatibility: OpenStack provides a standardized API that makes it easy to integrate with other systems and services.

Community Support: OpenStack has a large and active community of developers and users who collaborate to improve and enhance the platform. This provides a wealth of resources, support, and knowledge for organizations using OpenStack.

Vendor-Neutral: OpenStack is vendor-neutral, which means it can be deployed on any hardware or cloud infrastructure, providing greater choice and flexibility for organizations.

Overall, OpenStack provides a range of benefits that enable organizations to deploy and manage private or public clouds that are flexible, scalable, cost-effective, and easy to integrate with other systems and services.

6.4 DISADVANTAGES OF OPENSTACK

While OpenStack provides a range of benefits, there are also some potential disadvantages to consider. Some of the disadvantages of OpenStack include:

Complexity: OpenStack is a highly customizable cloud computing platform, which can make it complex to deploy and manage. Organizations need to have skilled personnel with expertise in OpenStack to manage and maintain the cloud infrastructure.

Resource Requirements: OpenStack requires significant hardware resources to run, which can be costly to acquire and maintain. Organizations need to ensure they have the necessary hardware infrastructure in place to support OpenStack.

Integration Challenges: While OpenStack provides a standardized API for integration with other systems and services, integration can still be challenging, especially with legacy systems or proprietary software.

Security Risks: OpenStack provides a range of security features, but organizations need to ensure they properly configure and manage the security features to avoid potential security risks.

Limited Vendor Support: OpenStack is an open-source platform, which means that vendor support can be limited compared to proprietary cloud computing platforms. Organizations need to ensure they have the necessary support in place to manage and maintain the OpenStack infrastructure.

Learning Curve: OpenStack is a complex platform with a steep learning curve, which can make it challenging for organizations to deploy and manage.

Organizations need to ensure they have the necessary resources and training in place to properly manage the OpenStack infrastructure.

Overall, while OpenStack provides a range of benefits, organizations need to carefully consider the potential challenges and disadvantages before deploying OpenStack. Organizations need to ensure they have the necessary resources, skills, and support in place to properly manage and maintain the OpenStack infrastructure.

6.5 INSTALLIZATION OF OPENSTACK

Installing OpenStack is a complex process that requires a good understanding of the underlying technology and the steps involved. Here are the basic steps to install OpenStack:

Choose a distribution: OpenStack is available in several distributions, including Canonical's Ubuntu, Red Hat's RHEL, and Mirantis' OpenStack. Choose a distribution that fits your requirements.

Prepare the host system: Ensure that the host system meets the minimum hardware and software requirements, such as CPU, RAM, disk space, and operating system.

Install and configure the database: OpenStack requires a database to store configuration data, metadata, and other information. The recommended database is MariaDB or MySQL.

Install and configure message broker: OpenStack uses message brokers to communicate between its components. RabbitMQ is the most commonly used message broker in OpenStack.

Install and configure Identity service (Keystone): Keystone is the identity service in OpenStack that manages authentication and authorization. It should be installed and configured before other OpenStack services.

Install and configure other OpenStack services: After Keystone is set up, you can install and configure other OpenStack services such as Compute (Nova), Networking (Neutron), Block Storage (Cinder), Object Storage (Swift), and Image Service (Glance).

Configure the dashboard: OpenStack has a web-based dashboard (Horizon) that allows users to manage their cloud resources. You can configure Horizon to customize the look and feel of the dashboard.

Verify the installation: After the installation is complete, verify that OpenStack is

working correctly by creating instances, networks, and storage.

These are only the basic steps involved in installing OpenStack. Depending on your use case and requirements, you may need to perform additional configurations and customizations. It's also recommended to follow the official OpenStack documentation for your chosen distribution for detailed installation instructions.

In summary, the installation of OpenStack requires a good understanding of the underlying theory and architecture of the platform. This includes knowledge of the deployment models, networking, storage, and security services provided by OpenStack. By understanding these concepts, users can successfully install and configure OpenStack for their specific cloud infrastructure needs.

To further expand on the summary, the installation of OpenStack involves deploying a combination of components, including Keystone, Nova, Glance, Cinder, Neutron, Horizon, and Swift. These components work together to provide a comprehensive cloud infrastructure that allows users to manage and provision compute, storage, and network resources.

CHAPTER 7

CONCLUSION AND REFERENCES

7.1 CONCLUSION

OpenStack is a popular open-source software platform used to build and manage private and public cloud infrastructures. The project was launched in 2010 and has grown to become one of the most widely used cloud computing platforms in the world. In conclusion, OpenStack offers many benefits to organizations looking to build their own private or public cloud infrastructure, including flexibility, scalability, and cost-effectiveness. With a large and active community of developers and users, OpenStack is constantly evolving to meet the changing needs of cloud computing. However, implementing and managing OpenStack can be complex and requires a high level of expertise. Organizations looking to adopt OpenStack should carefully consider their specific needs, resources, and capabilities before making a decision. It is also important to have a clear understanding of the potential challenges and risks associated with OpenStack, such as security issues and interoperability with other cloud platforms.

Overall, OpenStack is a powerful and versatile platform for building and managing cloud infrastructures, but it requires careful planning and expertise to implement successfully.

7.2 FUTURE SCOPE

As we know, cloud computing is getting wide in the last decade. Most of the biggest giants investing heavily to leverage the growing trend, such as Google, AWS, Microsoft, or Rackspace.

OpenStack is Open Source freely available with Apache 2.0 license and is backed by thousands of big technology companies. Over the years, it is kept on improving its popularity. Hundreds of enterprises are using it to build their own private cloud, and some are delivering support services. So, career-wise, in current business situations, I would suggest it is one of the best innovations to look for. The future calls for such a stable environment, and as a free architecture with some excellent long-term features, OpenStack will continue its legacy.

If you're looking for something that allows you to make better use of your current hardware, doesn't lock you into costly support contracts and has a wide global engineering community (over 100,000 members across 187 countries and increasing year on year by around 33 percent), then OpenStack could be the solution for you.

One thing is to remain there for secure cloud computing, with OpenStack getting a promising future for sure in at least the next ten years. OpenStack is a bundle of numerous cutting edge items that offer the best experience for cloud computing. So learning it and gaining experience in a few projects would not only make you a seasoned professional in the cloud but also help you develop skills in problem-solving.

REFERENCES

1. Pettey, C. and L. Goasduff. Gartner Executive Programs Worldwide Survey of More Than 2,000 CIOs Identifies Cloud Computing as Top Technology Priority for CIOs in 2011.
2. 2011[cited2012 04/06/2012]. <http://www.gartner.com/it/page.jsp?id=1526414>.
3. P., M. and T. Grance., The NIST Definition of Cloud Computing. 2011.
4. Sabahi, F. Cloud computing security threats and responses. in Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on. 2011.
5. Canonical. Ubuntu Cloud: Technologies for future-thinking companies. 2012.
6. <http://www.canonical.com/about-canonical/resources/white-papers/ubuntu-cloud-technologies-future-thinking-companies>.
7. <https://cloudsecurityalliance.org/research/security-guidance/>.
Eucalyptus Web Pages (OpenSource). <http://open.eucalyptus.com>.
Nimbus Project Web Page. <http://www.nimbusproject.org>
OpenNebula Web Page. <http://www.opennebula.org>.