



**UNIVERSIDAD LATINA
DE COSTA RICA**

POWERED BY **Arizona State University**

Universidad Latina de Costa Rica

Adm. de Sistemas Operativos y Redes

Proyecto

Tema del Proyecto:

Openstack

Estudiantes:

Mauricio Taylor Fonseca

Sebastian Vargas Delgado

Profesor:

Carlos Andres Mendez Rodriguez

23 de agosto del 2024

San Pedro, Montes de Oca

Tabla de Contenido

Introducción	3
Objetivos	3
Objetivo General	3
Marco Teórico	4
OpenStack	4
Mejor Prácticas	7
Requerimientos de OpenStack Detallados	7
Requerimientos Técnicos Detallados	7
Requerimientos Específicos por Servicio	8
Instalación de OpenStack	9
Gestión de OpenStack	11
Lanzamiento de una Instancia del OpenStack	21
Consideraciones al lanzar una Instancia	24
Conclusiones	26
Bibliografía	27

Introducción

Con el surgimiento de la computación en la nube, la gestión eficiente de recursos y la automatización de servicios han adquirido una importancia sin precedentes. Por lo que OpenStack es una gran opción puesto que es una plataforma de código abierto para la construcción de infraestructuras en la nube, ofreciendo un conjunto de herramientas robustas para la administración de grandes volúmenes de datos y recursos computacionales.

Asimismo se utiliza un sistema operativo de Ubuntu Desktop, puesto que es un entorno ampliamente utilizado por su versatilidad y compatibilidad con diversas tecnologías de código abierto. Además de que es mejor utilizar SO con un entorno Desktop o con interfaz pues si queremos realizar revisiones via web esta nos lo proporcionará.

Objetivos

Objetivo General

- Implementar y configurar una infraestructura de nube privada utilizando OpenStack sobre Ubuntu Desktop, con el fin de proporcionar una plataforma de gestión de recursos computacionales flexible, escalable y segura.

Marco Teórico

OpenStack

OpenStack, es una plataforma de nube abierta, que ofrece una gran flexibilidad en cuanto a los sistemas operativos que se pueden utilizar en sus distintos componentes. Sin embargo, hay ciertos aspectos clave que se deben considerar para garantizar un funcionamiento óptimo y seguro.

Primeramente, los sistemas operativos más comunes al utilizar openstack.

CentOS/RHEL: Son las distribuciones más utilizadas en entornos OpenStack debido a su estabilidad, soporte a largo plazo y amplia comunidad. Puesto que esta distribución es conocida por su estabilidad y madurez, lo que las hace ideales para entornos de producción. Además de ofrecer soporte extendido, lo que garantiza que los sistemas estén actualizados y protegidos durante un período prolongado; y una gran comunidad de usuarios y desarrolladores asegura una amplia documentación y soporte.

Ubuntu: Esta es una opción popular, especialmente para entornos más ágiles y con una mayor frecuencia de actualizaciones. Porque ofrece actualizaciones frecuentes y un ciclo de lanzamiento más rápido. Además de que permite una alta personalización y cuenta con una gran variedad de paquetes disponibles.

Debian: Similar a Ubuntu, Debian ofrece un alto grado de personalización y estabilidad lo que lo convierte en una opción popular para muchos usuarios..

Fedora: Ideal para entornos de desarrollo y pruebas, al ser una distribución más reciente con las últimas tecnologías, puesto que es una distribución orientada a la innovación .

Asimismo tenemos los requerimientos generales que requiere Openstack en el sistema operativo donde se desea instalar.

Kernel: Un kernel estable y actualizado es esencial para garantizar la seguridad y el rendimiento de la plataforma.

Paquetes: Se necesita una serie de paquetes para instalar y configurar los servicios de OpenStack. Estos incluyen herramientas como la virtualización, las redes, el almacenamiento y la gestión.

Configuración: La configuración del sistema operativo debe ajustarse para la optimización del rendimiento de los servicios de OpenStack. Esto es la configuración de redes, el almacenamiento, la seguridad y los servicios del sistema.

Seguridad: Además, es fundamental mantener el sistema operativo actualizado con los últimos parches de seguridad para proteger la plataforma de ataques.

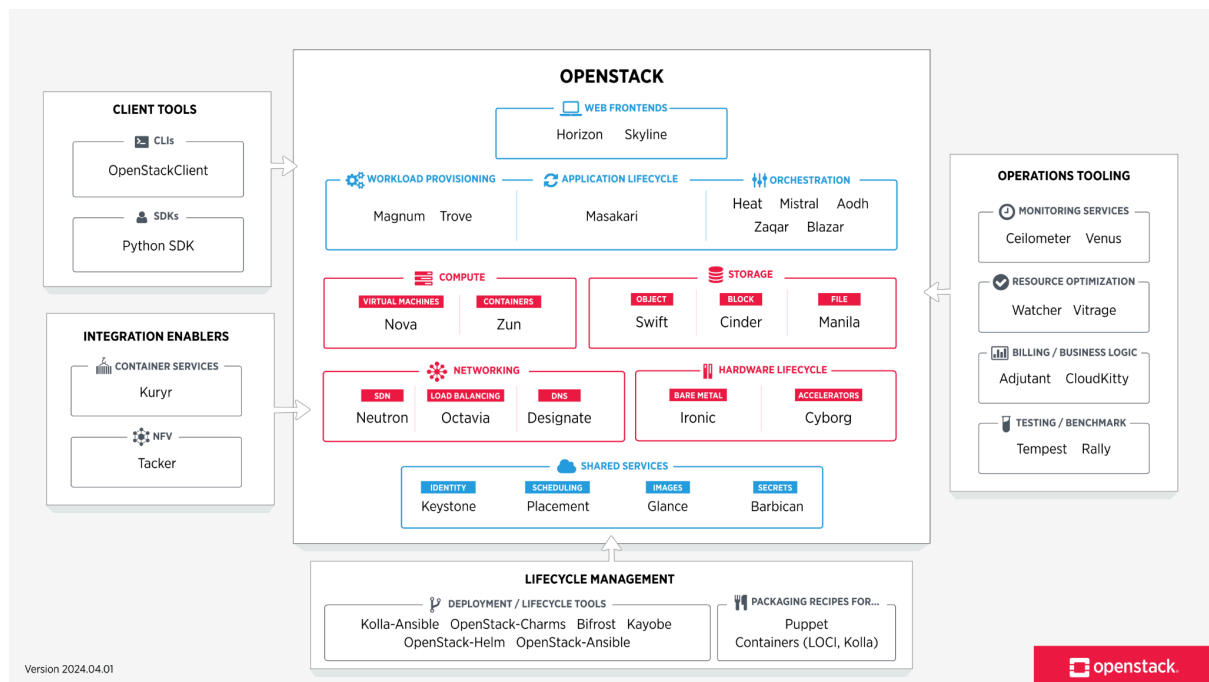
También tenemos los requerimientos que necesita cada servicio de openstack para su correcto funcionamiento.

Controlador de Computación (Nova): Este requiere de un sistema operativo estable y con soporte para KVM o QEMU.

Servicio de Bloque de Almacenamiento (Cinder): Cinder necesita un sistema operativo con soporte para iSCSI, NFS o CIFS, además de los drivers necesarios para los dispositivos de almacenamiento.

Servicio de Objetos (Swift): Requiere un sistema operativo estable y con soporte para los protocolos de almacenamiento distribuido.

Red (Neutrón): Éste necesita un sistema operativo con soporte para redes virtuales y enrutamiento.



Además tenemos consideraciones que tenemos que tomar en cuenta al momento de utilizar openstack; como son:

Versiones: Es importante utilizar versiones compatibles de los sistemas operativos y los servicios de OpenStack.

Hardware: Los requisitos de hardware varían según la carga de trabajo y el tamaño de la nube.

Virtualización: La mayoría de las implementaciones de OpenStack utilizan KVM como hipervisor, por lo que el sistema operativo debe ser compatible con él.

Contenedores: Aunque OpenStack tradicionalmente se ha utilizado para gestionar máquinas virtuales, cada vez es más común su uso con contenedores (Docker, Kubernetes).

Nubes híbridas: Si se planea integrar OpenStack con otras nubes, es necesario considerar la compatibilidad entre los sistemas operativos y las APIs.

Mejor Prácticas

- **Automatización:** Automatizar la instalación y configuración de OpenStack reduce el riesgo de errores humanos y agiliza el proceso.
- **Actualizaciones:** Mantener los sistemas operativos y los servicios de OpenStack actualizados con los últimos parches de seguridad.
- **Monitoreo:** Implementar herramientas de monitoreo para supervisar el rendimiento y la salud de la plataforma.
- **Documentación:** Mantener una documentación detallada de la configuración y los procesos.

Requerimientos de OpenStack Detallados

Requerimientos Técnicos Detallados

- **Kernel:**
 - Virtualización: El kernel debe soportar tecnologías de virtualización como KVM o QEMU para ejecutar múltiples máquinas virtuales.
 - Networking: Debe tener soporte para redes virtuales y enrutamiento, lo que es fundamental para la comunicación entre los servicios de OpenStack.

- Almacenamiento: Debe soportar diferentes tipos de almacenamiento, como iSCSI, NFS y CIFS.
- **Paquetes:**
 - Herramientas de virtualización: QEMU, KVM, libvirt.
 - Gestión de paquetes: apt, yum, dnf.
 - Bases de datos: MySQL, PostgreSQL.
 - Servicios de mensajería: RabbitMQ.
 - Herramientas de configuración: Ansible, Puppet, Chef.
- **Configuración:**
 - Redes: Configuración de interfaces de red, enrutamiento, DNS y DHCP.
 - Almacenamiento: Configuración de volúmenes, particiones y sistemas de archivos.
 - Seguridad: Configuración de firewalls, usuarios y grupos, y políticas de seguridad.
 - Servicios del sistema: Configuración de servicios como SSH, NTP y cron.
- **Seguridad:**
 - Parches de seguridad: Aplicar regularmente parches de seguridad para proteger el sistema operativo y las aplicaciones.
 - Hardening: Configurar el sistema operativo para minimizar la superficie de ataque.
 - Controles de acceso: Implementar controles de acceso basados en roles para limitar el acceso a los recursos.

Requerimientos Específicos por Servicio

- **Nova:**

- Virtualización: Soporte para KVM o QEMU para crear y gestionar máquinas virtuales.
- Networking: Integración con Neutron para proporcionar redes virtuales a las instancias.
- Almacenamiento: Interacción con Cinder para gestionar el almacenamiento de las instancias.

- **Cinder:**

- Almacenamiento: Soporte para diferentes tipos de almacenamiento (iSCSI, NFS, CIFS).
- Backend: Integración con backends de almacenamiento como iSCSI, Ceph o NetApp.

- **Swift:**

- Almacenamiento distribuido: Soporte para protocolos de almacenamiento distribuido como Ring.
- Escalabilidad: Capacidad para escalar horizontalmente para almacenar grandes cantidades de datos.

Instalación de OpenStack

Primeramente, instalamos una versión simplificada y liviana de OpenStack para facilitar su instalación y así poder realizar sus pruebas y revisar sus funcionalidades. Con el siguiente comando:

```
sebas@sebas-VirtualBox:~$ sudo snap install microstack --beta
```

Con el siguiente comando podemos revisar la versión del microstack.

```
sebas@sebas-VirtualBox:~$ snap list microstack
Nombre      Versión  Rev  Seguimiento  Editor      Notas
microstack  ussuri   245  latest/beta  canonical✓  -
sebas@sebas-VirtualBox:~$
```

Seguidamente procedemos a inicializar los servicios y recursos con el siguiente comando.

```
sebas@sebas-VirtualBox:~$ sudo microstack init --auto --control
2024-08-22 16:28:56,228 - microstack_init - INFO - Configuring clustering ...
2024-08-22 16:28:56,560 - microstack_init - INFO - Setting up as a control node.
2024-08-22 16:29:01,573 - microstack_init - INFO - Generating TLS Certificate and Key
```

Asimismo podemos ver la disponibilidad de máquinas que microstack nos ofrece para poder lanzar una instancia

```
sebas@sebas-VirtualBox:~$ microstack.openstack flavor list
+-----+-----+-----+-----+-----+-----+-----+
| ID | Name      | RAM | Disk | Ephemeral | VCPUs | Is Public |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | m1.tiny   | 512 | 1    | 0          | 1     | True      |
| 2 | m1.small  | 2048 | 20   | 0          | 1     | True      |
| 3 | m1.medium | 4096 | 20   | 0          | 2     | True      |
| 4 | m1.large  | 8192 | 20   | 0          | 4     | True      |
| 5 | m1.xlarge | 16384 | 20   | 0          | 8     | True      |
+-----+-----+-----+-----+-----+-----+-----+
```

Aquí seguidamente iniciamos una instancia, podemos ver que nos dispone de dos maneras de conexión, por ssh y vía web.

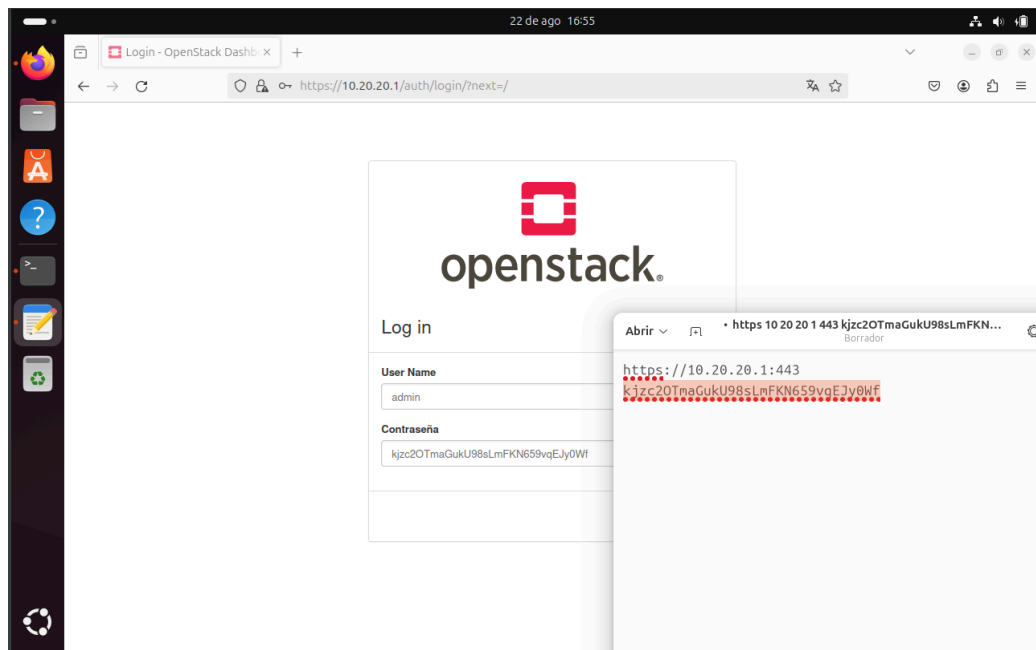
```
sebas@sebas-VirtualBox:~$ microstack launch cirros -n virtu
Creating local "microstack" ssh key at /home/sebas/snap/microstack/common/.ssh/id_microstack
Launching server ...
Allocating floating ip ...
Server virtu launched! (status is BUILD)

Access it with `ssh -i /home/sebas/snap/microstack/common/.ssh/id_microstack cirros@10.20.20.52`
You can also visit the OpenStack dashboard at https://10.20.20.1:443
sebas@sebas-VirtualBox:~$
```

Antes de realizar la conexión necesitamos obtener su clave de seguridad por lo que escribimos el siguiente comando.

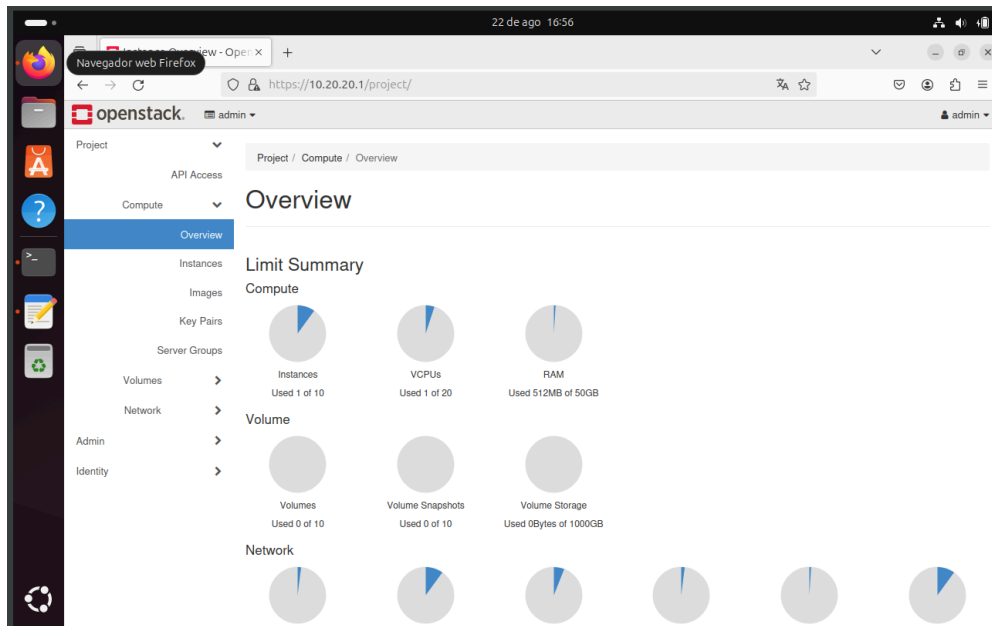
```
sebas@sebas-VirtualBox:~$ sudo snap get microstack config.credentials.keystone-password
kjzc2OTmaGukU98sLmFKN659vqEJy0Wf
sebas@sebas-VirtualBox:~$
```

Ahora si, una vez tenemos la clave de seguridad y su respectivo *Username* que es *Admin*, podemos acceder al dashboard de openstack gestionar todas la instancias, redes, routers, entre otros; en este caso accedemos de manera vía web para su mejor entendimiento.

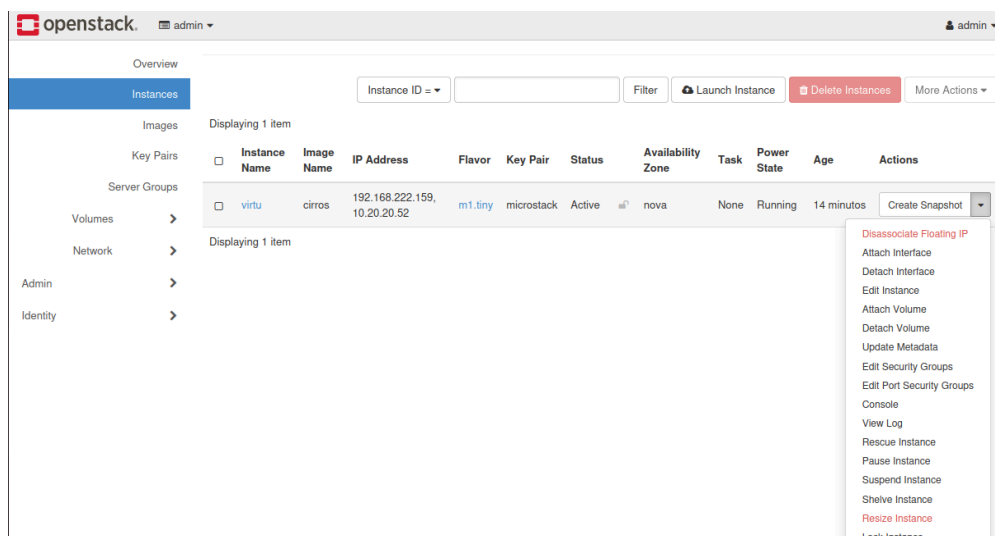


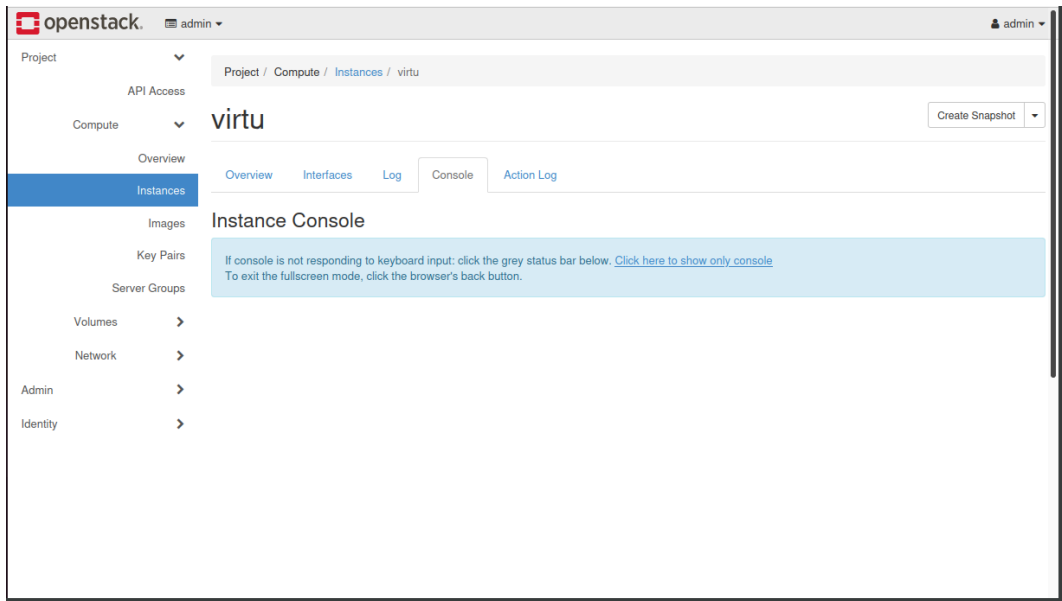
Gestión de OpenStack

Primeramente una vez accedemos a openstack podemos ver las opciones que este nos ofrece, además de poder gestionar los recursos que se están utilizando.

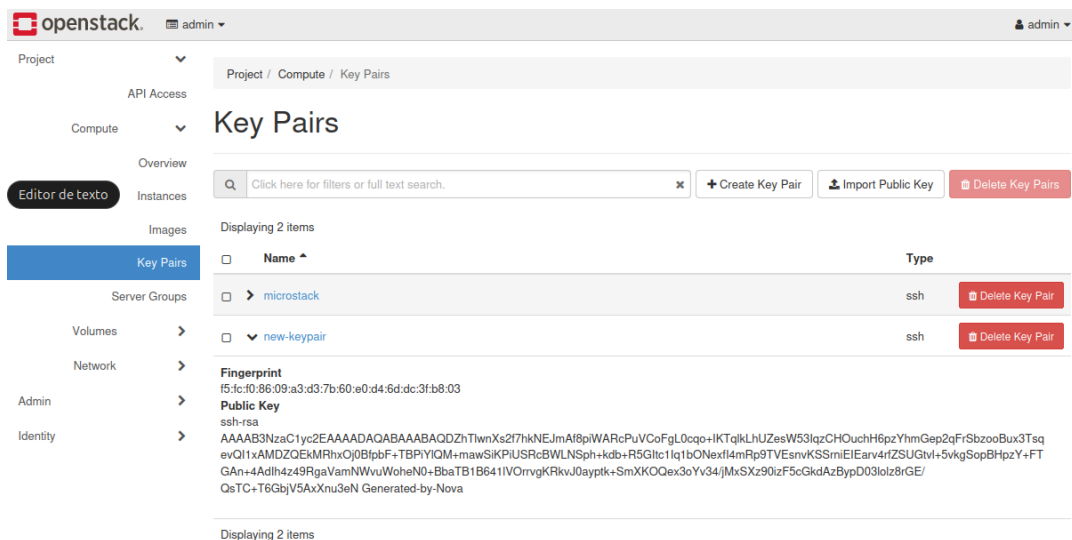


Seguidamente tenemos la opción de instancias, donde podemos gestionar todas las instancias creadas y poder modificarlas a las necesidades, pero más adelante veremos eso.





Ahora en la misma opción de *Compute* podemos ver la opción de *Key Pairs* que es una opción de crear un clave de seguridad por medio de ssh y esto poder utilizarlo en una instancia.



Seguidamente tenemos de crear un volumen o un bloque de almacenamiento

Create Image

Image Details

Metadata

Image Details

Specify an image to upload to the Image Service.

Image Name

Image Description

Image Source

File

Browse....

Format

ISO - Optical Disk Image

Image Requirements

Kernel

Choose an image

Ramdisk

Choose an image

Architecture

Minimum Disk (GB)

0

Minimum RAM (MB)

0

Image Sharing

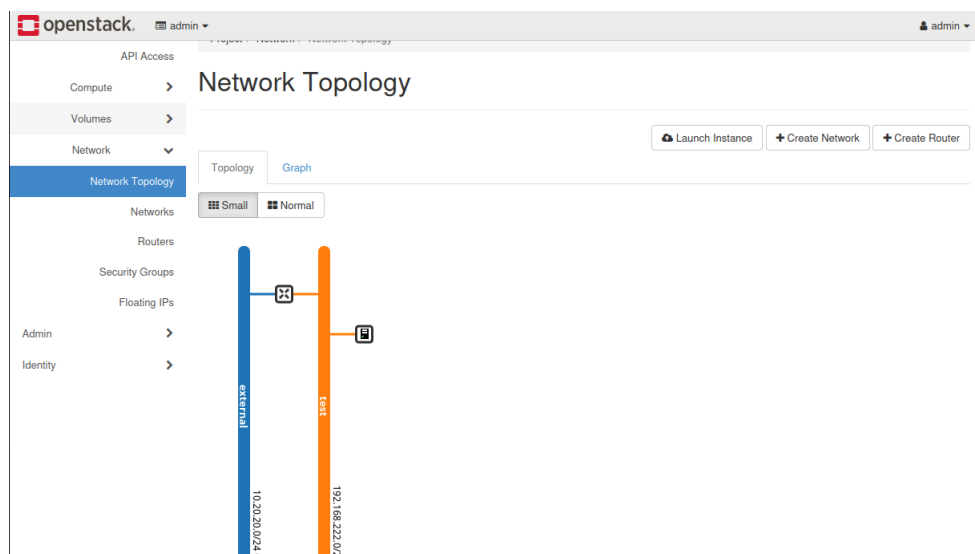
Visibility

Private Shared Community Public

Protected

Yes No

En la opción de *Network*, tenemos la opción de *Network Topology* donde podremos observar la topología de nuestra nube.



Ahora en la opción de *Network* tenemos la opción de crear y gestionar una red. Además de poder configurar su respectiva subnet donde podremos configurar un address, así como la versión de la IP ya sea IPv4 o IPv6.

Create Network

Subnet Name: new-subnetwork

Network Address: 10.254.0.0/28

IP Version: IPv4

Gateway IP: 10.254.0.1

☐ Disable Gateway

Creates a subnet associated with the network. You need to enter a valid "Network Address" and "Gateway IP". If you did not enter the "Gateway IP", the first value of a network will be assigned by default. If you do not want gateway please check the "Disable Gateway" checkbox. Advanced configuration is available by clicking on the "Subnet Details" tab.

Cancel Back Next

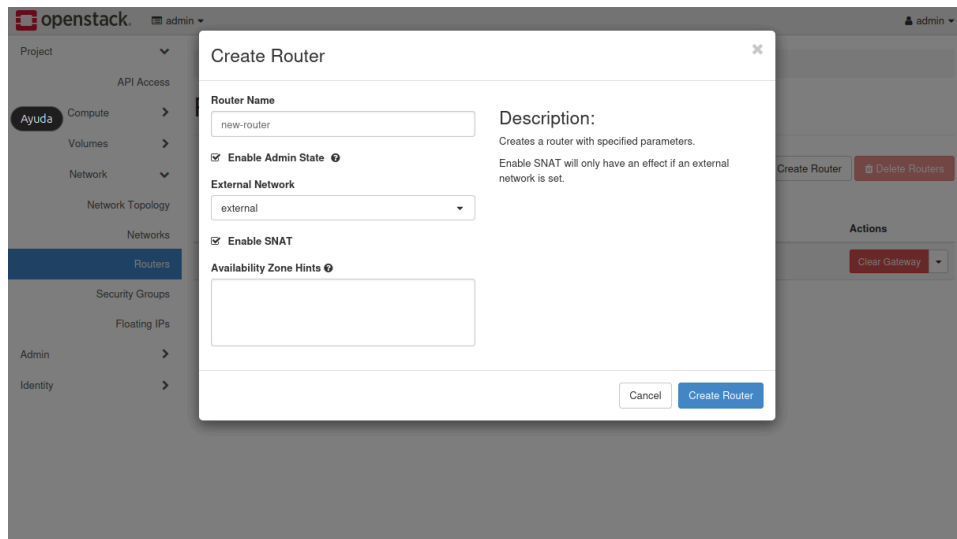
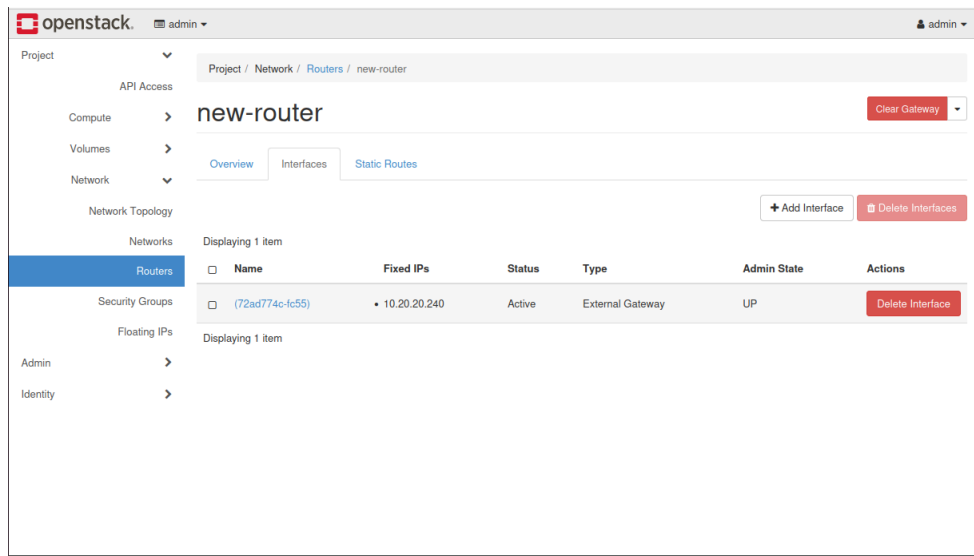
Networks

Displaying 3 items

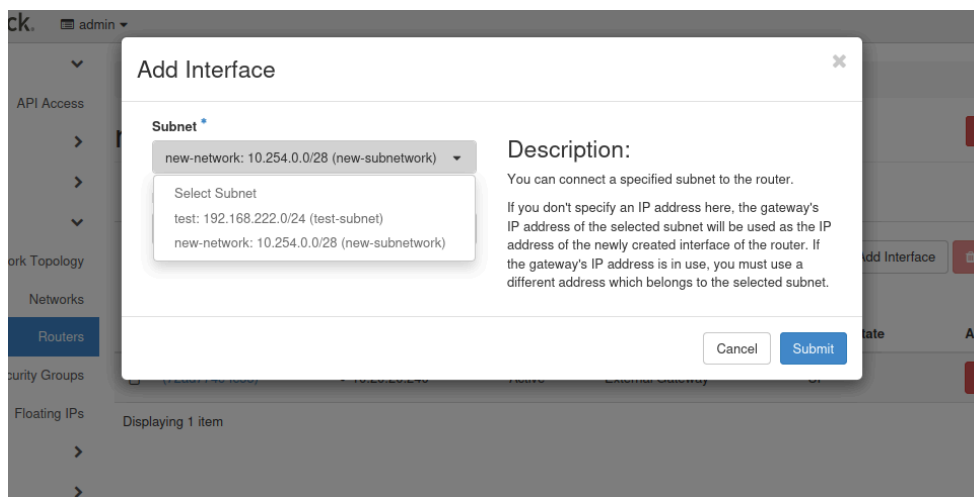
<input type="checkbox"/>	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
<input type="checkbox"/>	test	test-subnet 192.168.222.0/24	no	no	Active	UP	-	Edit Network
<input type="checkbox"/>	external	external-subnet 10.20.20.0/24	no	Si	Active	UP	-	Edit Network
<input type="checkbox"/>	new-network	new-subnetwork 10.254.0.0/28	no	no	Active	UP	-	Edit Network

Displaying 3 items

Asimismo, podemos crear y gestionar los *Router* que tengamos



En dicho *Router* podremos seleccionar la subnet que va a tener.



Ahora bien, en la opción de *Network* tenemos la opción de *Security Groups* donde podremos crear y gestionar ciertas de reglas que queramos definir.

The screenshot shows the OpenStack dashboard interface. The top navigation bar includes the OpenStack logo, a user menu for 'admin', and a breadcrumb trail: 'Project / Network / Security Groups'. The left sidebar contains a tree view with categories like Project, API Access, Compute, Volumes, Network (selected), Network Topology, Networks, Routers, Security Groups (highlighted), Floating IPs, Admin, and Identity. The main content area is titled 'Security Groups' and features a search filter, '+ Create Security Group', and '- Delete Security Groups' buttons. A table displays one item:

<input type="checkbox"/>	Name	Security Group ID	Description	Actions
<input type="checkbox"/>	default	b9395562-5a49-4fe3-8785-dd6b1272e89d	Default security group	<button>Manage Rules</button>

Below the table, it indicates 'Displaying 1 item'.

Como podemos ver en los *Security Groups* como mencionamos tiene la opción de gestionar ciertas reglas.

The screenshot shows the 'Manage Security Group Rules' page in the OpenStack dashboard. The breadcrumb trail is 'Project / Network / Security Groups / Manage Security Group Rules'. The left sidebar is similar to the previous screenshot, with 'Security Groups' highlighted. The main content area is titled 'Manage Security Group Rules: New-SecurityGroup (074ea4f0-e10f-437d-bcce-2f514c2032d1)' and includes '+ Add Rule' and '- Delete Rules' buttons. A table displays two items:

<input type="checkbox"/>	Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
<input type="checkbox"/>	Egress	IPv4	Any	Any	0.0.0.0/0	-	-	<button>Delete Rule</button>
<input type="checkbox"/>	Egress	IPv6	Any	Any	:::/0	-	-	<button>Delete Rule</button>

Below the table, it indicates 'Displaying 2 items'.

Aquí podemos ver las opciones de reglas que tienen los *Security Groups* para poder realizar conexiones, entre otros.

Add Rule

Rule *

- Custom TCP Rule
- Custom UDP Rule
- Custom ICMP Rule
- Other Protocol
- All ICMP
- All TCP
- All UDP
- DNS
- HTTP
- HTTPS
- IMAP
- IMAPS
- LDAP
- MS SQL
- MYSQL
- POP3
- POP3S
- RDP
- SMTP
- SMTPS
- SSH

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Cancel Add

Add Rule

Rule *

Custom TCP Rule

Description ?

Direction

Ingress

Open Port *

Port

Port ?

Remote *

CIDR

CIDR ?

0.0.0.0/0

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Cancel Add

Add Rule



Rule *

SSH



Description ?

Remote * ?

CIDR



CIDR * ?

0.0.0.0/0

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Cancel

Add

Lanzamiento de una Instancia del OpenStack

Como mencionamos antes en la opción de *Compute* tenemos la opción de *Instance*, donde podremos gestionar y crear dichas instancias, por lo que ahora crearemos una instancia, primeramente escribiéndole un nombre.

Launch Instance

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Instance Name *

new-instance

Description

Availability Zone

nova

Count *

1

Total Instances (10 Max)

20%

1 Current Usage
1 Added
8 Remaining

Cancel < Back Next > Launch Instance

Seguidamente podremos escoger una imagen que basarnos para crear una instancia pero no es obligatoria.

Launch Instance

Instance source is the template used to create an instance. You can use an image, a snapshot of an instance (image snapshot), a volume or a volume snapshot (if enabled). You can also choose to use persistent storage by creating a new volume.

Select Boot Source

Image

Volume Size (GB) *

1

Create New Volume

Yes No

Delete Volume on Instance Delete

Yes No

Allocated

Displaying 0 items

Name	Updated	Size	Type	Visibility
Select an item from Available items below				

Displaying 0 items

Available 1

Select one

Q Click here for filters or full text search.

Displaying 1 item

Name	Updated	Size	Type	Visibility
> cirros	8/22/24 10:38 PM	12.13 MB	QCOW2	Public

Ahora sí, tenemos que elegir que tipo de máquina virtual vamos a utilizar por lo que podremos ver las opciones disponibles que tendremos a disposición.

Launch Instance

Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

Allocated

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
Select an item from Available items below						

Available

Select one

Click here for filters or full text search.

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
m1.tiny	1	512 MB	1 GB	1 GB	0 GB	Yes
m1.small	1	2 GB	20 GB	20 GB	0 GB	Yes
m1.medium	2	4 GB	20 GB	20 GB	0 GB	Yes
m1.large	4	8 GB	20 GB	20 GB	0 GB	Yes
m1.xlarge	8	16 GB	20 GB	20 GB	0 GB	Yes

Ahora asignamos que *Network* va a pertenecer la instancia creada.

Launch Instance

Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Networks provide the communication channels for instances in the cloud.

Allocated

Select networks from those listed below.

Network	Subnets Associated	Shared	Admin State	Status
Select an item from Available items below				

Available

Select at least one network

Click here for filters or full text search.

Network	Subnets Associated	Shared	Admin State	Status
test	test-subnet	No	Up	Active
external	external-subnet	No	Up	Active
new-network	new-subnetwork	No	Up	Active

Podremos asignarle un *Security Group*.

Launch Instance

Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Select the security groups to launch the instance in.

▼ Allocated 1

Displaying 1 item

Name	Description
> default	Default security group

Displaying 1 item

▼ Available 1

Select one or more

Q

Click here for filters or full text search.

×

Displaying 1 item

Name	Description
> New-SecurityGroup	

Displaying 1 item

Así como podremos asignarle una *Key Pair*.

Launch Instance

Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

+ Create Key Pair

Import Key Pair

Allocated

Displaying 0 items

Name	Type
Select a key pair from the available key pairs below.	

Displaying 0 items

▼ Available 2

Select one

Q

Click here for filters or full text search.

×

Displaying 2 items

Name	Type
> microstack	ssh
> new-keypair	ssh

Displaying 2 items

Consideraciones al lanzar una Instancia

Al momento de elegir la imagen estas deben de ser imágenes de sistemas operativos que estén optimizadas para el entorno de OpenStack y que incluyan los paquetes necesarios. Además de que estas imágenes usadas estén actualizadas con los últimos parches de seguridad y mejoras de rendimiento.

Asimismo al definir el Flavor apropiado se debe seleccionar un flavor que corresponda a los requisitos de la aplicación que correrá en la instancia. Aunque si las opciones predefinidas no satisfacen tus necesidades, se debe considera crear flavors personalizados que alineen mejor con los requisitos específicos de tu aplicación.

Al momento de configurar la red y la seguridad; primeramente las redes y subredes con IPs que mejor se adapten a tus necesidades. Asegurándote de que la instancia esté conectada a la red correcta y que las reglas de cortafuegos de los *Security Groups* estén bien configuradas, es decir que los *Security Groups* restringan el tráfico solamente necesario, así minimizando puertos y protocolos abiertos así reduciendo posibles vulnerabilidades.

Así como si la instancia necesita ser accesible desde fuera de la red privada, podemos asignar una IP Flotante. Además de poder asignarle una clave SSH al iniciar la instancia para permitir un acceso seguro y sin contraseña.

Conclusiones

Este proyecto ha funcionado como una guía práctica para quienes deseen copiar y adaptar la implementación de OpenStack en sus entornos, proporcionando una base sólida para el desarrollo de infraestructuras en la nube.

Además, se ha conseguido implementar y configurar una infraestructura de nube privada utilizando OpenStack en Ubuntu Desktop, estableciendo una plataforma robusta para la gestión de recursos computacionales. Esta plataforma destaca por su capacidad de ofrecer flexibilidad, escalabilidad y seguridad dentro de un entorno de nube privada.

Bibliografía

- OpenStack Foundation. (s.f.). *OpenStack: Open source software for creating private and public clouds.* OpenStack. Recuperado de <https://www.openstack.org/>
- Canonical Ltd. (s.f.). *OpenStack on Ubuntu.* Ubuntu. Recuperado de <https://ubuntu.com/openstack>