

UNIVERSIDAD POLITECNICA DE CHIAPAS

ING. EN DESARROLLO DE SOFTWARE

ALUMNO:

Luis Daniel Cruz Gómez - 201255

Alejandro Mauricio Ocampo López - 201227

Jesús Eduardo Jiménez Guillen - 201258

PROFESOR:

Alonso Macias

MATERIA:

Sistemas Operativos

CUATRIMESTRE:

7

GRUPO:

В



Descripción de la arquitectura general OpenStack

La arquitectura de OpenStack está constituida por una gran cantidad de proyectos open source, los cuales se utilizan para establecer el undercloud y el overcloud de OpenStack que emplean los administradores de sistemas y los usuarios de la nube respectivamente. Los underclouds contienen los elementos clave que necesitan los administradores de sistemas para configurar y gestionar los entornos de OpenStack de los usuarios finales, conocidos como overclouds. Hay seis servicios básicos estables que gestionan la informática, las conexiones en red, el almacenamiento, la identidad y las imágenes, y más de doce servicios opcionales que varían según la consolidación del desarrollo. Los seis servicios principales constituyen la infraestructura que permite al resto de los proyectos gestionar los paneles, la coordinación, el aprovisionamiento de equipos sin sistema operativo, la mensajería, los contenedores y la gobernabilidad.

• ¿Cómo funciona OpenStack?

➡ Básicamente, OpenStack es una serie de comandos conocidos como scripts. Esos scripts están agrupados en paquetes llamados "proyectos", los cuales transmiten tareas que generan entornos de nube.

Descripción detallada de cada uno de los componentes de la arquitectura OpenStack

Una implementación de OpenStack contiene una serie de componentes que proporcionan API para acceder a los recursos de la infraestructura. Esta página enumera los diversos servicios que se pueden implementar para proporcionar dichos recursos a los usuarios finales de la nube.

Web Frontend

Horizon

- Horizon es la implementación canónica del Dashboard de OpenStack, que proporciona una interfaz de usuario basada en la web para los servicios de OpenStack, incluidos Nova, Swift, Keystone, etc.
- OpenStack Horizon es una interfaz gráfica basada en web a la que los administradores y usuarios de la nube pueden acceder para administrar los servicios de cómputo, almacenamiento y redes de OpenStack.
- OpenStack Horizon ofrece tres versiones de paneles de gestión: un panel de usuario, un panel del sistema y un panel de configuración. Los administradores de la nube pueden personalizar los elementos visuales de la interfaz de Horizon.

> Api Proxies

❖ El módulo ec2api es parte de OpenStack, un esfuerzo del equipo de infraestructura de OpenStack para proporcionar pruebas de integración continuas y revisión de código para OpenStack y proyectos comunitarios de OpenStack que no forman parte del software principal. El módulo en sí mismo se utiliza para configurar y administrar de manera flexible el servicio API de EC2 para OpenStack.

Soporte de EC2 API para OpenStack. Este proyecto proporciona un servicio de API de EC2 independiente que persigue dos objetivos:

- Implemente la API de VPC que ahora está ausente en la API EC2 de nova
- Cree un servicio independiente para la compatibilidad con la API de EC2 que se adapte no solo a la API de VPC, sino también al resto de la API de EC2 actualmente presente en nova

Workload Provisioning

4 Magnum

Magnum es un servicio de API de OpenStack mejorado por el equipo de contenedores de OpenStack que hace que los motores de orquestación de contenedores como Apache Mesos, Kubernetes y Docker Swarm estén disponibles como recursos de clase inicial dentro de OpenStack. Magnum aplica calor para orquestar una imagen del sistema operativo que incluye Kubernetes y Docker y ejecuta esa imagen en particular en una máquina virtual o bare metal dentro de la configuración del clúster.

4 Sahara

- ❖ El proyecto Sahara tiene como objetivo proporcionar a los usuarios un medio simple para aprovisionar marcos de procesamiento de datos (como Hadoop, Spark y Storm) en OpenStack. Esto se logra especificando parámetros de configuración como la versión del marco, la topología del clúster, los detalles del hardware del nodo y más.
- Sahara se puede definir como un componente para el aprovisionamiento rápido y sencillo de clústeres de Hadoop. Muchos usuarios definirán varios parámetros, como el número de versión de Hadoop, la información de tipo de nodo (configuración de RAM y CPU, especificación de espacio en disco), tipo de topología de clúster y más.

♣ Trove

Trove es una base de datos como servicio para OpenStack. Está diseñado para ejecutarse completamente en OpenStack, con el objetivo de permitir a los usuarios utilizar rápida y fácilmente las funciones de una base de datos relacional sin la carga de manejar tareas administrativas complejas. Los usuarios de la nube y los administradores de bases de datos pueden aprovisionar y administrar múltiples instancias de bases de datos según sea necesario.

Masakari

Masakari proporciona un servicio de alta disponibilidad de instancias para nubes OpenStack mediante la recuperación automática de instancias fallidas. Actualmente, Masakari puede recuperar máquinas virtuales (VM) basadas en KVM de eventos de falla como el proceso de VM inactivo, el proceso de aprovisionamiento inactivo y la falla del host de cómputo nova. Masakari también proporciona un servicio de API para gestionar y controlar el mecanismo de rescate automatizado.

Murano

Murano permite a los desarrolladores de aplicaciones y administradores de la nube publicar varias aplicaciones listas para la nube en un catálogo navegable. Los usuarios de la nube, incluidos los inexpertos, pueden usar el catálogo para crear entornos de aplicaciones confiables con solo presionar un botón. Murano usa OpenStack Heat para organizar los recursos de infraestructura para la aplicación.

📥 Solum

Hacer que los servicios en la nube sean más fáciles de consumir e integrar con su proceso de desarrollo de aplicaciones al automatizar el proceso de fuente a imagen y simplificar la implementación centrada en la aplicación.

Freezer

Freezer es una plataforma distribuida de copia de seguridad, restauración y recuperación ante desastres como servicio. Está diseñado para múltiples sistemas operativos (Linux, Windows, OSX...), enfocado en proporcionar eficiencia y flexibilidad para copias de seguridad basadas en bloques, copias de seguridad incrementales basadas en archivos, acciones puntuales, sincronización de trabajos (es decir, sincronización de copias de seguridad en múltiples nodos) y muchas otras características. Está destinado a ser útil para todos los entornos, incluidas las grandes nubes efímeras.

Orchestration

🚣 Heat

Heat organiza los recursos de infraestructura para una aplicación en la nube basándose en plantillas en forma de archivos de texto que se pueden tratar como código. Heat proporciona una API REST nativa de OpenStack y una API de consulta compatible con CloudFormation. Heat también proporciona un servicio de ajuste de escala automático que se integra con los servicios de telemetría de OpenStack, por lo que puede incluir un grupo de ajuste de escala como recurso en una plantilla.

Mistral

Mistral es un servicio de flujo de trabajo. La mayoría de los procesos comerciales consisten en varios pasos distintos interconectados que deben ejecutarse en un orden particular en un entorno distribuido. Uno puede describir dicho proceso como un conjunto de tareas y relaciones de tareas (a través del lenguaje basado en YAML) y cargar dicha descripción en Mistral para que se encargue de la gestión del estado, el orden de ejecución correcto, el paralelismo, la sincronización y la alta disponibilidad.

4 Aodh

- ❖ El objetivo de Aodh es habilitar la capacidad de desencadenar acciones basadas en reglas definidas contra muestras o datos de eventos recopilados por Ceilometer.
- El proyecto de servicio de alarmas (aodh) proporciona un servicio que habilita la capacidad de desencadenar acciones basadas en reglas definidas contra datos de métricas o eventos recopilados por Ceilometer o Gnocchi.

\rm 🚣 Senlin

- Senlin es un servicio de agrupación en clústeres para nubes OpenStack. Crea y opera grupos de objetos homogéneos expuestos por otros servicios de OpenStack. El objetivo es facilitar la orquestación de colecciones de objetos similares.
- Senlin es un servicio para crear y administrar un clúster de múltiples recursos en la nube. Senlinproporciona una API REST nativa de OpenStack y está prevista una API de consulta compatible con AWS.

🖊 Zaqar

❖ Zaqar es un servicio de mensajería en la nube de múltiples inquilinos para desarrolladores web y móviles. El servicio presenta una API RESTFUL completa, que los desarrolladores pueden usar para enviar mensajes entre varios componentes de sus SaaS y aplicaciones móviles. Detrás de esta API se encuentra un motor de mensajería eficiente diseñado teniendo en cuenta la escalabilidad y la seguridad. Otros componentes de OpenStack pueden integrarse con Zaqar para mostrar eventos a los usuarios finales y comunicarse con los agentes invitados que se ejecutan en la capa "sobre la nube". Los operadores de la nube pueden aprovechar Zaqar para proporcionar equivalentes de SQS y SNS a sus clientes.

📥 Blazar

❖ Blazar es un servicio de reserva de recursos para OpenStack. Blazar permite a los usuarios reservar un tipo/cantidad específica de recursos durante un período de tiempo específico y arrienda estos recursos a los usuarios en función de sus reservas.

> Compute

Nova (Virtual Machines)

- Nova es el proyecto OpenStack que proporciona una forma de aprovisionar instancias informáticas (también conocidas como servidores virtuales). Nova admite la creación de máquinas virtuales, servidores baremetal (mediante el uso de ironic) y tiene soporte limitado para contenedores del sistema. Nova se ejecuta como un conjunto de demonios sobre los servidores Linux existentes para brindar ese servicio.
- Requiere los siguientes servicios adicionales de OpenStack para la función básica:

Keystone:

 proporciona identidad y autenticación para todos los servicios de OpenStack.

• Glance:

Esto proporciona el repositorio de imágenes de cómputo.
 Todas las instancias informáticas se inician a partir de imágenes de vistazo.

• Neutron:

 este es responsable de aprovisionar las redes virtuales o físicas a las que se conectan las instancias informáticas en el arrangue.

♣ Zun (Containers)

- Zun proporciona una API de OpenStack para lanzar y administrar contenedores respaldados por diferentes tecnologías de contenedores. A diferencia de Magnum, Zun es para usuarios que desean tratar los contenedores como un recurso administrado por OpenStack. Se supone que los contenedores administrados por Zun se integran bien con otros recursos de OpenStack, como la red Neutron y el volumen Cinder. Los usuarios reciben API simplificadas para administrar contenedores sin la necesidad de explorar las complejidades de las diferentes tecnologías de contenedores.
- Zun es un servicio de contenedor de OpenStack. Su objetivo es proporcionar un servicio de API para ejecutar contenedores de aplicaciones sin necesidad de administrar servidores o clústeres.

Storage

Swift (Object)

- Swift es un almacén de objetos/blobs altamente disponible, distribuido y eventualmente consistente. Las organizaciones pueden usar Swift para almacenar muchos datos de manera eficiente, segura y económica. Está diseñado para escalar y optimizado para durabilidad, disponibilidad y concurrencia en todo el conjunto de datos. Swift es ideal para almacenar datos no estructurados que pueden crecer sin límites.
- Swift es un blob/objeto-almacén eventualmente consistente y distribuido. El proyecto de almacenamiento de objetos de OpenStack se llama Swift y proporciona software para el almacenamiento en la nube para que podamos recuperar y almacenar una gran cantidad de datos junto con una API general. Se crea para la escala y se actualiza para la concurrencia, la disponibilidad y la durabilidad en todo el conjunto de datos. El almacenamiento de objetos es ideal para almacenar datos no estructurados que podrían crecer sin limitaciones.

Cinder (Block)

Cinder es un servicio de almacenamiento de bloques para OpenStack. Virtualiza la gestión de dispositivos de almacenamiento en bloque y proporciona a los usuarios finales una API de autoservicio para solicitar y consumir esos recursos sin necesidad de saber dónde se implementa realmente su almacenamiento o en qué tipo de dispositivo. Esto se hace mediante el uso de una implementación de referencia (LVM) o controladores de complemento para otro almacenamiento.

Manila (File)

OpenStack Manila (sistema de archivos compartidos) facilita una API abierta para administrar recursos compartidos dentro de la estructura independiente del proveedor. Primitivos estándar, como la capacidad de denegar/dar, eliminar y crear acceso a cualquier recurso compartido. Se puede aplicar en una variedad de entornos de red diferentes o independientes. Los dispositivos de almacenamiento técnico a través de los sistemas de datos Hitachi, INFINIDAT, Quobyte, Oracle, IBM, HP, NetApp y EMC pueden ser compatibles y las tecnologías de sistemas de archivos, así como Ceph y Red Hat GlusterFS.

Networking

Neutron (SDN)

- OpenStack Neutron es un proyecto de redes SDN centrado en ofrecer redes como servicio (NaaS) en entornos informáticos virtuales.
- ❖ Neutron se puede definir como un proyecto de OpenStack. Brinda la facilidad de " conectividad de red como servicio " entre varios dispositivos de interfaz (como vNIC) que son manejados por otros tipos de servicios OpenStack (como Nova). Opera la API de red de OpenStack.
- Maneja cada faceta de red para VNI (infraestructura de red virtual) y varios factores de capa de autorización de PNI (infraestructura de red física) en una plataforma OpenStack. La red OpenStack permite que los proyectos creen topologías avanzadas de la red virtual. Puede incluir algunos de los servicios como VPN (red privada virtual) y un firewall.

Octavia (Load balancing)

Octavia es una solución de equilibrio de carga a escala de operador de código abierto diseñada para funcionar con OpenStack. Octavia nació del proyecto Neutron LBaaS y, a partir del lanzamiento de Liberty de OpenStack, Octavia se ha convertido en la implementación de referencia para Neutron LBaaS versión 2. Octavia logra su entrega de servicios de balanceo de carga administrando una flota de máquinas virtuales, contenedores o servidores bare metal —conocidos colectivamente como ánforas— que gira bajo demanda. Esta característica de escalado horizontal bajo demanda diferencia a Octavia de otras soluciones de balanceo de carga, lo que hace que Octavia sea verdaderamente adecuado "para la nube".

Designate (DNS)

- ❖ Designate es un servicio DNSaaS de múltiples inquilinos para OpenStack. Proporciona una API REST con autenticación Keystone integrada. Se puede configurar para generar registros automáticamente en función de las acciones de Nova y Neutron. Designate admite una variedad de servidores DNS, incluidos Bind9 y PowerDNS 4.
- ❖ Designate se puede definir como una API REST multiinquilino para administrar DNS. Facilita DNS como el Servicio. Este componente es compatible con varias tecnologías de back-end como BIND y PowerDNS. No ofrece el servicio DND ya que su objetivo es interactuar mediante un servidor DNS (existente) para administrar zonas DNS según el arrendatario.

Hardware Lifecycle

Ironic (Bare metal)

- ❖ Ironic es un proyecto de OpenStack que aprovisiona máquinas bare metal (a diferencia de las virtuales). Se puede usar de forma independiente o como parte de OpenStack Cloud, y se integra con los servicios OpenStack Identity (keystone), Compute (nova), Network (neutron), Image (glance) y Object (swift)
- ❖ El servicio Bare Metal administra el hardware a través de protocolos de administración remota comunes (por ejemplo, PXE e IPMI) y específicos del proveedor. Proporciona al operador de la nube una interfaz unificada para una flota heterogénea de servidores, al mismo tiempo que proporciona al servicio de cómputo una interfaz que permite que los servidores físicos se administren como si fueran máquinas virtuales

Cyborg (Accelerators)

- Cyborg proporciona un marco de gestión de propósito general para aceleradores (incluidos GPU, FPGA, dispositivos basados en ASIC, etc.)
- Como usuario final de Cyborg, usará Cyborg para crear y administrar aceleradores con herramientas o la API directamente.

Shared Services

Keystone (Identity)

Keystone es un servicio de OpenStack que proporciona autenticación de clientes de API, detección de servicios y autorización multiinquilino distribuida mediante la implementación de la API de identidad de OpenStack. Es compatible con LDAP, OAuth, OpenID Connect, SAML y SQL.

Placement (Scheduling)

- El servicio API de Placement se introdujo en la versión 14.0.0 Newton dentro del repositorio nova y se extrajo al repositorio de Placement en la versión 19.0.0 Stein
- ❖ Este es un modelo de pila y datos de REST API que se utiliza para rastrear los inventarios y usos del proveedor de recursos, junto con diferentes clases de recursos. Por ejemplo, un proveedor de recursos puede ser un nodo de cómputo, un grupo de almacenamiento compartido o un grupo de asignación de IP. El servicio de Placement rastrea el inventario y el uso de cada proveedor. Por ejemplo, una instancia creada en un nodo de cómputo puede ser un consumidor de recursos como RAM y CPU de un

proveedor de recursos de nodo de cómputo, disco de un proveedor de recursos de grupo de almacenamiento compartido externo y direcciones IP de un proveedor de recursos de grupo de IP externo.

Glance (Images)

- Los servicios de imágenes de Glance incluyen la detección, el registro y la recuperación de imágenes de máquinas virtuales. Glance tiene una API RESTFUL que permite consultar metadatos de imágenes de máquinas virtuales, así como recuperar la imagen real. Las imágenes de VM disponibles a través de Glance se pueden almacenar en una variedad de ubicaciones, desde sistemas de archivos simples hasta sistemas de almacenamiento de objetos como el proyecto OpenStack Swift.
- El proyecto de servicio de imágenes (vistazo) proporciona un servicio en el que los usuarios pueden cargar y descubrir activos de datos que están destinados a ser utilizados con otros servicios. Esto actualmente incluye imágenes y definiciones de metadatos.

Barbican (Secrets)

Barbican es el servicio OpenStack Key Manager. Proporciona almacenamiento, aprovisionamiento y gestión seguros de datos secretos, como contraseñas, claves de cifrado, certificados X.509 y datos binarios sin procesar.

Operations Tooling

Esos servicios ofrecen API dirigidas principalmente a administradores e implementadores de la nube, para ayudar con las operaciones en la nube.

Monitoring Services

4 Ceilometer

El objetivo de Ceilometer es recopilar, normalizar y transformar de manera eficiente los datos producidos por los servicios de OpenStack. Los datos que recopila están destinados a usarse para crear diferentes vistas y ayudar a resolver varios casos de uso de telemetría. Aodh y Gnocchi

Monasca

- Monasca es una solución de monitoreo como servicio de código abierto, tolerante a fallas, altamente escalable, de alto rendimiento y multiinquilino que se integra con OpenStack. Utiliza una API REST para el procesamiento y consulta de métricas de alta velocidad y tiene un motor de transmisión de alarmas y un motor de notificaciones.
- Monasca utiliza API REST para métricas de alta velocidad, procesamiento de registros y consultas. Integra un motor de transmisión de alarmas, un motor de notificaciones y un motor de agregación.

Venus

Venus es un proyecto OpenStack que tiene como objetivo proporcionar una solución integral para la recopilación de registros, limpieza, indexación, análisis, alarmas, visualización, generación de informes y otras necesidades, lo que implica ayudar al operador o al mantenedor a resolver rápidamente problemas de recuperación, comprender la salud operativa. de la plataforma, y mejorar el nivel de gestión de la plataforma. Que pueden incluir registros de OpenStack, registros del sistema operativo, registros de servicios de plataforma en la nube y registros relacionados con aplicaciones virtualizadas

> Resource Optimization

Watcher

Watcher proporciona un servicio de optimización de recursos flexible y escalable para nubes basadas en OpenStack de múltiples inquilinos. Watcher proporciona un servicio de optimización de recursos flexible y escalable para nubes basadas en OpenStack de múltiples inquilinos. Watcher proporciona un ciclo de optimización completo, que incluye todo, desde un receptor de métricas, un procesador de optimización y un aplicador de planes de acción.

👃 Vitrage

Vitrage es el servicio de OpenStack para el Análisis de Causa Raíz. Se utiliza para organizar, analizar y visualizar alarmas y eventos de OpenStack, obtener información sobre la causa raíz de los problemas y deducir su existencia antes de que se detecten directamente.

> Billing / Bussines Logic

📥 Adjutant

- Un marco de flujo de trabajo básico creado con Django y Django-Rest-Framework para ayudar a automatizar las tareas de administración dentro de un clúster de OpenStack.
- ❖ El objetivo de Adjutant es proporcionar un lugar y acciones estándar para completar la funcionalidad que falta en Keystone y permitir la fácil adición de lógica comercial a tareas más complejas y conexiones con sistemas externos.

Cloudkitty

- CloudKitty es un proyecto de calificación como servicio diseñado para traducir métricas en precios.
- CloudKitty admite múltiples recopiladores, múltiples políticas de calificación y múltiples salidas.
- CloudKitty es altamente modular, lo que facilita agregar nuevas funciones.

> Testing / Benchmark

Tempest

Tempest es un conjunto de pruebas de integración que se ejecutarán en un clúster OpenStack en vivo. Tempest tiene baterías de pruebas para la validación de la API de OpenStack, escenarios y otras pruebas específicas útiles para validar una implementación de OpenStack. Tempest debería poder ejecutarse en cualquier nube OpenStack, ya sea un solo nodo, una nube LXC de 20 nodos o una nube KVM de 1000 nodos. Tempest proporciona capacidad de complemento, lo que significa que cualquier complemento de Tempest se puede conectar y ejecutar su prueba junto con las pruebas de Tempest.

Patrole

❖ Patrole es una herramienta de validación de seguridad para verificar que el control de acceso basado en roles esté correctamente configurado y aplicado en una nube de OpenStack. Tiene una batería de pruebas dedicadas a validar la corrección e integridad de la implementación de RBAC de la nube.

♣ Rally

Rally es una herramienta de evaluación comparativa y análisis de rendimiento para OpenStack que se puede utilizar para automatizar la medición y la creación de perfiles centrada en cómo los nuevos cambios de código afectan el rendimiento de OpenStack, detectan problemas de escalamiento y rendimiento e investigan cómo las diferentes arquitecturas de implementación y hardware afectan el rendimiento de OpenStack. Se puede utilizar como una herramienta básica para un sistema OpenStack CI/CD que mejoraría continuamente su SLA, rendimiento y estabilidad.

Client Tools

Herramientas y bibliotecas del lado del cliente para interactuar con las API de OpenStack.

> CLIS

OpenStackClient

- ❖ Interfaz de línea de comandos única para servicios OpenStack con un conjunto de comandos y formato uniformes.
- OpenStackClient (también conocido como OSC) es un cliente de línea de comandos para OpenStack que reúne el conjunto de comandos para Compute, Identity, Image, Object Storage y Block Storage API en un solo shell con una estructura de comando uniforme.

> SDKs

Python SDK

SDK de Python compatible con múltiples nubes para la API REST de OpenStack que expone tanto el conjunto completo de API de bajo nivel como la lógica empresarial seleccionada de nivel superior.

Integration Enablers

≠ El software de esta sección facilita la integración de componentes OpenStack en pilas de infraestructura abierta adyacentes.

Container Services

4 Kuryr

♣ La idea detrás de Kuryr es poder aprovechar la abstracción y todo el arduo trabajo que se realizó en Neutron y sus complementos y servicios y usarlo para proporcionar redes de nivel de producción para casos de uso de contenedores. En lugar de que cada complemento o solución independiente de Neutron intente encontrar y cerrar las brechas, podemos concentrar los esfuerzos y enfocarnos en un solo lugar: Kuryr

> NFV

4 Tacker

❖ Tacker proporciona un administrador de VNF genérico (VNFM) y un orquestador de NFV (NFVO) para implementar y operar servicios de red y funciones de red virtual (VNF) en una plataforma de infraestructura de NFV como OpenStack. Se basa en ETSI MANO Architectural Framework y proporciona una pila funcional para orquestar servicios de red de extremo a extremo mediante VNF.

> Lifecycle Management

Herramientas y recetas de paquetes para ayudar a instalar y mantener el ciclo de vida de las implementaciones de OpenStack.

> Deployment / Lifecycle Tools

♣ Kolla-Ansible

❖ Kolla-Ansible implementa un plano de control OpenStack en contenedores utilizando contenedores Kolla, orquestados a través de Ansible. El proyecto tiene como objetivo la simplicidad y la confiabilidad, al tiempo que proporciona un modelo de configuración flexible e intuitivo.

OpenStack-Charms

❖ La Guía OpenStack Charm es la principal fuente de documentación para el proyecto OpenStack Charms. El proyecto supervisa la colección de Charmed Operators, también llamados simplemente "charms", que se utilizan para implementar y administrar nubes OpenStack utilizando MAAS y Juju.

TripleO

TripleO es un proyecto destinado a instalar, actualizar y operar nubes OpenStack utilizando las propias instalaciones de nube de OpenStack como base, basándose en Nova, Ironic, Neutron y Heat para automatizar la gestión de la nube a escala del centro de datos.

Bifrost

❖ Bifrost (pronunciado bye-frost) es un conjunto de libros de jugadas de Ansible que automatiza la tarea de implementar una imagen base en un conjunto de hardware conocido usando ironic. Proporciona una utilidad modular para la implementación única del sistema operativo con la menor cantidad de requisitos operativos que sea razonablemente posible.

4 Kayobe

Kayobe despliega un plano de control OpenStack en contenedores a metal desnudo. Bifrost se utiliza para descubrir y aprovisionar los servidores en la nube. Kolla se utiliza para crear imágenes de contenedores para los servicios de OpenStack. Kolla Ansible se utiliza para implementar un plano de control OpenStack en contenedores.

OpenStack-Helm

Colección de gráficos de Helm que implementan OpenStack y servicios relacionados en Kubernetes de manera simple, resistente y flexible.

OpenStack-Ansible

OpenStack-Ansible proporciona playbooks y roles de Ansible para la implementación y configuración de un entorno OpenStack.

OpenStack-Chef

Los libros de cocina de Chef para OpenStack automatizan la creación, el funcionamiento y el consumo de las implementaciones en la nube

Packaging Recipes for Popular Frameworks

♣ Loci

El objetivo de LOCI es proporcionar imágenes y herramientas compatibles con OCI, livianas y compatibles con CI/CD para los servicios de OpenStack. Puede crear imágenes LOCI en un entorno con espacio de aire sin modificaciones (suponiendo que tenga un espejo git, package y pypi).

Puppet-OpenStack

Los módulos Puppet OpenStack brindan automatización de TI escalable y confiable a las implementaciones en la nube de OpenStack.

Ventajas y desventajas frente a otras arquitecturas

Ventajas de OpenStack

- Autonomía para el usuario
- Escalabilidad
- Código abierto

Desventajas de OpenStack

- Desarrollo rápido de nuevas funciones
- Poco soporte técnico oficial
- Sólo funciona en sistemas Linux

Guía de instalación

INSTALACION Y CONFIGURACION OPENSTACK

La instalación de OpenStack se realiza sobre el sistema operativo Ubuntu 14.04 LTS Desktop.

Se comprueba si los scripts de configuración de OpenStack están disponibles dentro de una carpeta raíz del sistema operativo.

El acceso a la carpeta principal del OpenStack, se realiza de la siguiente manera:

root@ubuntumaster:/home/ubuntumaster# cd openstackgeek/icehouse root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse#

Los scripts de configuración serán ejecutados en el siguiente orden:

El paquete de instalación que permite definir la conectividad de red y direccionamiento en la nube, se ejecuta con el comando ./openstack_networking.sh.

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehous # ./openstack_networking.sh
Reading package lists... Done

Building dependency tree
Reading state information... Done

libencode-locale-perl libexporter-lite-perl libfakeroot libfdt1
libfile-listing-perl libfont-afm-perl libhtml-form-perl libhtml-format-perl
libhtml-parser-perl libhtml-tagset-perl libhtml-tree-perl
libhtml-parser-perl libhttp-daemon-perl libhttp-date-perl
libhttp-message-perl libhttp-negottate-perl libho-html-perl
libbio-stringy-perl libjson-perl libjson-xs-perl libhup-mediatypes-perl
liblup-protocol-https-perl libnet-http-perl libnetcf1
libparse-debcontrol-perl librados2 librbd1 libsd11.2debian libseccomp2
libsigsegv2 libspice-server1 libstdc++-4.8-dev libtie-ixhash-perl
libusbredirparser1 libvirt0 libwww-perl libwww-robotrules-perl libxen-4.4
libxenstore3.0 libxml2-utils msr-tools python-cheetah python-libvirt
python-vm-builder python3-magic qemu-keymaps gemu-system-common
qemu-system-x86 qemu-utils seabios sharutils wdiff

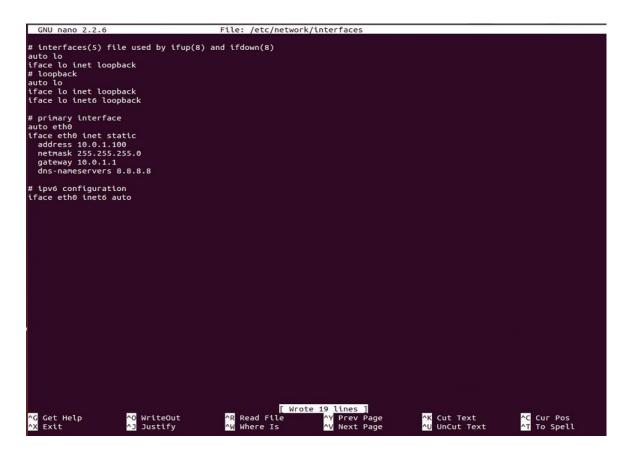
Suggested packages:
default-mta mail-transport-agent augeas-doc debtags bsd-mailx mailx
cvs-buildpackage devscripts-el gnuplot libterm-size-perl libyaml-syck-perl
mutt svn-buildpackage wšm debian-keyring equivs libsoap-lite-perl
mini-dinstall python-bzrlib g++-multilib g++-4.8-multilib gcc-4.8-doc
libstdc++6-4.8-dbg gawk-doc augeas-tools libdata-dump-perl
libcrypt-ssleay-perl libstdc++-4.8-doc radvd lvm2 libauthen-ntlm-perl
python-markdown python-pygments python-memcache samba vde2 sgabios
```

Con este script se instalan los siguientes servicios:

- ✓ Vlan gemu-kvm
- ✓ Ntp

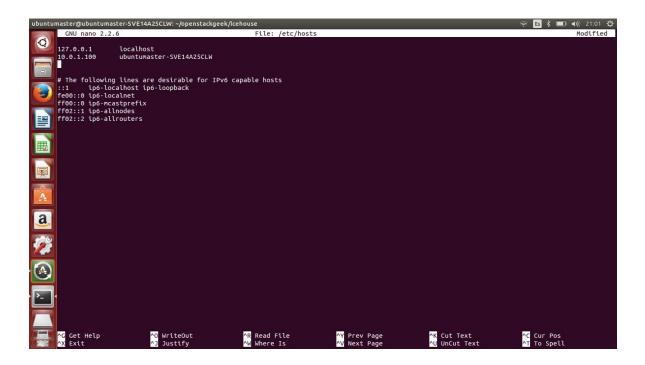
Una vez terminada la ejecución del script, es necesario configurar la interfaz Ethernet con una dirección IPv4 estática no enrutable y una dirección IPv6 auto configurable. Para ello, se edita manualmente el archivo ubicado en la ruta /etc/network/interfaces.

root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse# nano /etc/network/interfaces



Luego, editar el archivo ubicado en la ruta /etc/hosts para incluir una entrada al controlador.

root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse# nano /etc/hosts



Reiniciar el equipo para guardar la configuración realizada.

Después de editar la configuración de red, es necesario verificar si el servidor soporta la virtualización del procesador con el script ./openstack_serve_test.sh.



Cuando el equipo no soporta la virtualización (KVM), se debe habilitar desde la BIOS. Luego, es necesario actualizar el repositorio del sistema operativo.

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse# ./openstack_system
update.sh
ign http://security.ubuntu.com trusty-security InRelease
Get:1 http://security.ubuntu.com trusty-security Release.gpg [933 B]
Ign http://extras.ubuntu.com trusty InRelease
Get:2 http://security.ubuntu.com trusty-security Release [63,5 kB]
Get:3 http://extras.ubuntu.com trusty Release.gpg [72 B]
Hit http://extras.ubuntu.com trusty Release
Ign http://co.archive.ubuntu.com trusty InRelease
Get:4 http://security.ubuntu.com trusty-security/main Sources [79,6 kB]
Hit http://extras.ubuntu.com trusty/main Sources
Ign http://co.archive.ubuntu.com trusty-updates InRelease
Get:5 http://security.ubuntu.com trusty-security/restricted Sources [2.061 B]
Hit http://extras.ubuntu.com trusty/main amd64 Packages
Get:6 http://security.ubuntu.com trusty-security/universe Sources [21,9 kB]
Get:7 http://security.ubuntu.com trusty-security/multiverse Sources [1.922 B]
Ign http://co.archive.ubuntu.com trusty-backports InRelease
Hit http://extras.ubuntu.com trusty/main i386 Packages
Get:8 http://security.ubuntu.com trusty-security/main amd64 Packages [264 kB]
Hit http://co.archive.ubuntu.com trusty Release.gpg
100% [8 Packages bzip2 0 B] [Waiting for headers] [Waiting for headers] [Waitin
```

DEFINICIÓN Y ASIGNACIÓN DE LAS VARIABLES DE ENTORNO

Se inicia ejecutando el script a continuación:

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse# ./openstack_setup.sh
```

Ingresar el nombre de la interfaz de red de área local del equipo. En este caso la Ethernet configurada anteriormente Eth0.

Se pregunta si el equipo se va a configurar como controlador, la respuesta debe ser afirmativa, en caso contrario se estará configurando un nodo por lo cual pedirá la url generada desde el servidor controlador.

Indicar una contraseña que se utilizará para que los servicios de OpenStack puedan comunicarse con MySQL.

Enter a password to be used for the OpenStack services to talk to MySQL:

Definir la dirección de correo electrónico para cuentas de servicio

```
Enter the email address for service accounts vycardozo@poligran.edu.co
```

Digitar nombre que se utilizara como región por defecto

```
Enter a short name to use for your default reg<sup>o</sup>on: Bogota
```

Se pregunta si el equipo controlador es multi-nodo, la respuesta debe ser afirmativa, para la inclusión de los nodos.

```
Is this a multi node inst<mark>ill? y</mark>
```

El script de configuración emite una URL que será utilizada para la configuración de los nodos.

```
The following URL will be used for configuring the other rigs in this cluster. Copy it.
https://sgsprunge.appspot.com/eSTYP269J7uOlqaP
```

SPLUNK

Con el fin de controlar y supervisar los eventos generados por la ejecución de lospaquetes en el transcurso de la instalación de OpenStack, obteniendo como resultado un log de actividades.

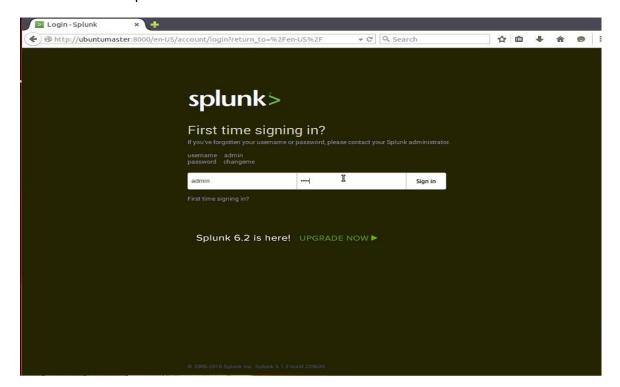
Se ejecuta el script ./openstack_splunk.sh

Se puede acceder a través de la siguiente URL (asumiendo que utiliza la dirección IP o nombre correcto del controlador): http://ubuntumaster:8000

```
writing new private key to 'privKeySecure.pem'
-----
Signature ok
subject=/CN=ubuntumaster/0=SplunkUser
Getting CA Private Key
writing RSA key
Done

If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com
The Splunk web interface is at http://ubuntumaster:8000
```

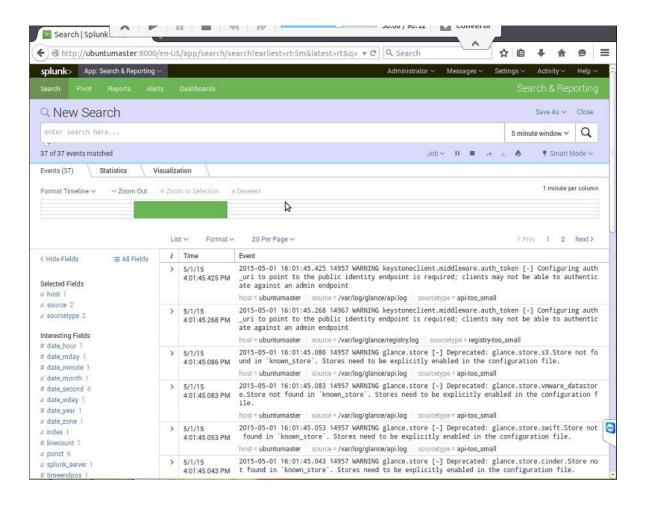
Inicio de sesión Splunk



Es necesario cambiar la contraseña de inicio de sesión.



Mediante la interfaz de Splunk se puede observar la secuencia de actividades (Descripción y hora).



INSTALACIÓN Y CONFIGURACIÓN DE BASE DE DATOS

El siguiente comando instala MySQL y RabbitMQ.

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse# ./openstack_mysql.sh
Reading package lists... Done
Bullding dependency tree
Reading state information... Done
The following extra packages will be installed:
    erlang-asn1 erlang-base erlang-corba erlang-crypto erlang-diameter
    erlang-asn1 erlang-base erlang-erl-docgen erlang-eunit erlang-ic
    erlang-inets erlang-mnesia erlang-nox erlang-odbc erlang-os-mon
    erlang-parsetools erlang-percept erlang-public-key erlang-runtime-tools
    erlang-snmp erlang-ssh erlang-ssl erlang-syntax-tools erlang-tools
    erlang-webtool erlang-xmerl libodbc1 libsctp1 lksctp-tools
Suggested packages:
    erlang erlang-manpages erlang-doc xsltproc fop erlang-ic-java
    erlang-observer libmyodbc odbc-postgresql tdsodbc unixodbc-bin
    libeache-memcached-perl libmemcached
The following NEW packages will be installed:
    erlang-asn1 erlang-base erlang-corba erlang-crypto erlang-diameter
    erlang-asn1 erlang-base erlang-erl-docgen erlang-eunit erlang-ic
    erlang-edoc erlang-eldap erlang-erl-docgen erlang-eunit erlang-ic
    erlang-parsetools erlang-percept erlang-public-key erlang-runtime-tools
    erlang-snmp erlang-ssh erlang-ssl erlang-syntax-tools erlang-tools
    erlang-memcache rabbitmq-server
0 upgraded, 32 newly installed, 0 to remove and 201 not upgraded.
```

Durante la instalación, es solicitada la contraseña de MySQL que se definió anteriormente, con el fin de establecer una contraseña al usuario root.

```
Configuring mysql-server-5.5

While not mandatory, it is highly recommended that you set a password for the MySQL administrative "root" user.

If this field is left blank, the password will not be changed.

New password for the MySQL "root" user:

***

<0k>
```

Confirmar la contraseña



Para finalizar la instalación satisfactoriamente, seguir los pasos a continuación:

Ingresar la contraseña correspondiente al usuario root de MySQL

```
In order to log into MySQL to secure it, we'll need the current password for the root user. If you've just installed MySQL, and you haven't set the root password yet, the password will be blank, so you should just press enter here.

Enter current password for root (enter for none):
```

El script pregunta si desea cambiar la contraseña al usuario root, la respuesta debe ser negativa.

```
You already have a root password set, so you can safely answer 'n'.
Change the root password? [Y/n]
```

De forma predeterminada, la instalación de MySQL tiene un usuario anónimo, lo que permite que cualquier persona ingrese en el motor de base de datos sin necesidad de tener una cuenta de usuario creada. Se debe eliminar esta opción antes de pasar a un entorno de producción.

```
Remove anonymous users? [Y/n] y
... Success!
```

Normalmente, conectarse con el usuario root únicamente debe permitirse desde 'localhost'. Esto asegura que no sea adivinada la contraseña administradora de la red.

Por ello, se deshabilita el inicio de sesión de forma remota para el usuario root.

```
Disallow root login remotely? [Y/n] y
... Success!
```

Por defecto MySQL viene con un 'test' de base de datos que cualquiera puede acceder. Debe ser eliminado.

```
Remove test database and access to it? [Y/n] y

- Removing privileges on test database...
... Success!
```

Recargar tablas de privilegios

```
Reload privilege tables now? [Y/n] y
... Success!
```

Confirmar la creación de la base de datos y usuarios, se utiliza la misma contraseña que se definió en la configuración de MySQL.

KEYSTONE

Es utilizado por OpenStack para proporcionar autenticación y autorización de alto nivel en todos los servicios instalados. Se inicia la instalación de Keystone escribiendo el siguiente comando:

```
The following NEW packages will be installed:
keystone libjs-jquery libjs-sphinxdoc libjs-underscore librabbitmq1
libyaml-0-2 python-ampg python-amyjson python-babel python-babel-localedata
python-decorator python-das python-dogpile.cache python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-sqlalchemy-python-sqlalchemy-python-sqlalchemy-python-sqlalchemy-python-sqlalchemy-python-sqlalchemy-python-sqlalchemy-python-sqlalchemy-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-python-pyth
```

Para verificar la lista de usuarios, utilizar las siguientes líneas de comando.

GLANCE

Es el servicio encargado de proporcionar inscripción y entrega de discos e imágenes de aplicaciones y/o sistemas operativos. Las imágenes almacenadas se utilizan como plantillas.

```
Reading package lists... Done
Bullding dependency tree
Reading state information... Done
The following extra packages will be installed:
    python-json-patch python-json-pointer python-jsonpatch python-warlock
The following NEW packages will be installed:
    python-glanceclient python-json-patch python-json-pointer python-jsonpatch
    python-warlock
0 upgraded, 5 newly installed, 0 to remove and 201 not upgraded.
Need to get 55,9 kB of archives.
After this operation, 477 kB of additional disk space will be used.
Get:1 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-json-pointer all 1.0-2build1 [5.158 B]
Get:2 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-json-patch all 1.3-4 [9.088 B]
Get:3 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-json-patch all 1.3-4 [2.312 B]
Get:4 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-warlock all 1.1.0-ubuntu2 [5.278 B]
Get:5 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-warlock all 1.1.0-ubuntu2 [5.278 B]
Get:6 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-yalocclient all 1:0.12.0-ubuntu2 [7.278 B]
Get:7 http://co.archive.ubuntu.com/ubuntu/ trusty/main python-yalocclient all 1:0.12.0-ubuntu2 [7.278 B]
Get:9 http://co.arch
```

Una vez la instalación finalice, consultar las imágenes disponibles en el sistema.

CINDER

Se utiliza para proporcionar los archivos de volumen adicionales para instancias en ejecución.

```
# root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse# ./openstack_cinder.sh

Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
    libdewmapper-event1.02.1 watershed
Suggested packages:
    thin-provisioning-tools
The following NEW packages will be installed:
    libdewmapper-event1.02.1 lvm2 watershed
0 upgraded, 3 newly installed, 0 to remove and 201 not upgraded.
Need to get 492 kB of archives.
After this operation, 1.427 kB of additional disk space will be used.
Get:1 http://co.archive.ubuntu.com/ubuntu/ trusty/main libdevmapper-event1.02.1 amd64 2:1.02.77-6ubuntu2 [10,8 kB]
Get:2 http://co.archive.ubuntu.com/ubuntu/ trusty/main watershed amd64 7 [11,4 kB]
Get:3 http://co.archive.ubuntu.com/ubuntu/ trusty/main lvm2 amd64 2.02.98-6ubuntu2 [470 kB]
Fetched 492 kB in 1s (279 kB/s)
```

Luego que la instalación de Cinder está completa, determinar la necesidad de espacio y ejecutar el script de creación de volumen de bucle de retorno (Se debe tener en cuenta que un archivo de bucle de retorno tiene por lo menos 1 GB de tamaño).

Para consultar los tipos de almacenamiento instalados, ejecutar la siguientes instrucción

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehous:# cinder type-list
| ID | Name |
| 32fb3f44-5d49-4ca7-aa10-0b38bd22a8c2 | Storage |
```

NOVA (Controlador)

Es un controlador de estructura Cloud Computing, es parte principal de sistema de laaS. Ofrece múltiples servicios para el control de redes, imágenes, arrancar y detener instancias.

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse: ./openstack_nova.sh
```

Cuando la instalación se haya completado, se pueden consultar los servicios que se ejecutan de la siguiente manera:

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehous<mark>e</mark># nova-manage service list
Binary
                                                                                                           Updated_At
                                                                 Zone
                    Host
nova-consoleauth ubuntumaster
                                                                  internal
                                                                                      enabled
                                                                                                           2015-05-01 21:21:00
                                                                                                          2015-05-01 21:21:01
2015-05-01 21:21:00
2015-05-01 21:21:01
2015-05-01 21:21:01
nova-conductor
                    ubuntumaster
                                                                  internal
                                                                                      enabled
nova-cert
nova-scheduler
                    ubuntumaster
                                                                  internal
                                                                                      enabled
                                                                                      enabled
                                                                  internal
                    ubuntumaster
                                                                                      enabled
nova-compute
                    ubuntumaster
                                                                 nova
                                                                  internal
                                                                                      enabled
                                                                                                           2015-05-01 21:21:01
                    ubuntumaster
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse#
```

NOVA (Plataforma computo)

```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse / ./openstack_nova_compute.sh
```

Ingresar el nombre del recurso para la NIC de la plataforma

Una vez que la plataforma de cómputo se ha configurado, verificar de nuevo los servicios de Nova.

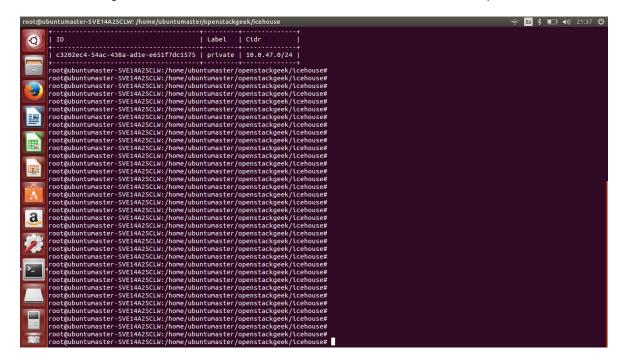
```
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehou<mark>:</mark>e# nova-manage service list
Binary Host
nova-consoleauth ubuntumaster
                                                                                                             Jpdated At
                                                                  Zone
                                                                                       enabled
                                                                                                            2015-05-01 21:23:21
2015-05-01 21:23:21
2015-05-01 21:23:20
2015-05-01 21:23:21
                                                                   internal
                                                                                                     :-)
nova-conductor
                     ubuntumaster
                                                                                       enabled
                                                                   internal
nova-cert
nova-scheduler
                     ubuntumaster
                                                                   internal
                                                                                       enabled
                     ubuntumaster
                                                                  internal
                                                                                       enabled
                                                                                                             2015-05-01 21:23:21
                                                                                       enabled
nova-compute
                     ubuntumaster
                                                                  nova
                                                                                                             2015-05-01 21:23:21
nova-network
                     ubuntumaster
                                                                   internal
                                                                                       enabled
root@ubuntumaster:/home/ubuntumaster/openstackgeek/icehouse#
```

A continuación se crea un rango de red privada IPv4, el cual será utilizado para indicar las lps`s a cada instancia establecida a partir de las imágenes almacenadas en Glace.

Bloquea la red 10.0.47.0/24, es decir la primera ip asignable es la 10.0.47.1/24.

root@ubuntucontrol-SVE14A25CLW:/home/ubuntucontrol/openstackgeek/icehouse# nova-manage network create private --fixed_range_v4=10.0.47.0/24 --nu m_networks=1 --bridge=br100 --bridge_interface=wlan0 --network_size=255

Mediante la siguiente línea de comando, se observa la lista de redes disponibles.



HORIZON

Es la aplicación canónica de Dashboard de OpenStack, que proporciona una interfaz de usuario basada en web.

```
rootBubuntumaster:/home/ubuntumaster/openstackgeek/icehouse#./openstack_horizon.sh
Reading state information... Done
Bullding dependency tree
Reading state information... Done
mencached is already the newest version.
The following extra packages will be installed:
apache2-bin apache2-data libapri libaprutil1 libaprutil1-dbd-sqlite3
libaprutil1-idap openstack-dashboard-ubuntu-theme python-appconf
python-clometerclient python-cloudfiles python-compressor python-django
python-django-horizon python-heatclient python-openstack-auth
python-troveclient
Suggested packages:
apache2-doc apache2-suexec-pristine apache2-suexec-custom apache2-utils
python-psycopg2 python-psycopg python-flup python-sqlite
geoip-database-contrib python-django-doc ipython bpython libgdali
python-lesscpy
The following NEW packages will be installed:
apache2 apache2-bin apache2-data libapache2-mod-wsgl libapri libaprutil1
libaprutil1-ddd-sqlitea libaprutil1-idap openstack-dashboard
openstack-dashboard-ubuntu-theme python-appconf python-django-horizon
python-heatclient python-compressor python-django python-django-horizon
python-heatclient python-openstack-auth python-troveclient

0 upgraded, 19 newly installed, 0 to remove and 201 not upgraded.

Need to get 5.769 kB of archives.

After this operation, 53,3 MB of additional disk space will be used.

Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main libaprutil1 amd64 1.5.3-1 [76,4 kB]

Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main python-appconf all 0.5-2build1 [7.696 B]

Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main python-appconf all 0.5-2build1 [7.696 B]

Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main python-propressor all 1.5.3-1 [8,6 kB]

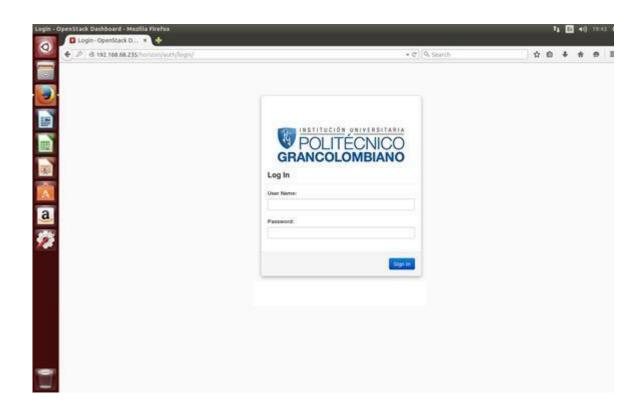
Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main python-appconf all 0.5-2build1 [7.696 B]

Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main python-oppressor all 1.6.1-2ubuntu0.8 [5.9 kB]

Get: http://co.archive.ubuntu.com/ubuntu/ trusty/main python-propressor all 1.6.3-1 [8.64 B]

Get: http://co.archive.ubuntu.com/ubuntu/
```

Al terminar la instalación de horizon, podemos acceder mediante la dirección xxxxxx a la interfaz de OpenStack.



CONCLUSION

- Independientemente de la opción que elija, la computación en la nube siempre estará ahí Hay buenas razones para apoyarlo. En términos de costo y flexibilidad, el servicio
- ♣ En cierto sentido, también es una solución más sostenible para el ambiente porque evita el uso de hardware adicional que se puede implementar baja eficiencia. En cualquier caso, el factor más importante es, sin duda, la seguridad de los datos. La confianza tiene Cuando deciden ceder sus datos o los de sus clientes servicio externo. Entonces, en este sentido, vale la pena el esfuerzo de crear su propia Utilice la nube OpenStack, que es un proyecto completo y conocido.

Bibliografía

- OpenStack. (s/f-a). Openstack.org. Recuperado el 21 de noviembre de 2022, de https://wiki.openstack.org/wiki/Main_Page
- ➤ OpenStack Architecture. (s/f). Www.javatpoint.com. Recuperado el 21 de noviembre de 2022, de https://www.javatpoint.com/openstack-architecture
- OpenStack, C. L. A. (s/f-b). Software. OpenStack. Recuperado el 21 de noviembre de 2022, de https://www.openstack.org/software/project-navigator/openstack-components
- (S/f). Edu.co. Recuperado el 21 de noviembre de 2022, de https://alejandria.poligran.edu.co/bitstream/handle/10823/682/Manual%20de%20Instala cion%20y%20Configuracion....Computing OpenStack.pdf?sequence=3&isAllowed=y