



Guía de referencias técnicas – Servicios de infraestructura en la nube con OpenStack

1.0 – Versión final

25 de noviembre de 2014



Se permite el uso comercial de la obra y de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Licencia Creative Commons Atribución CompartirIgual 3.0 Costa Rica.

<http://creativecommons.org/licenses/by-sa/3.0/cr/legalcode>

Índice

| | |
|--|----------|
| 1 Control del documento..... | 4 |
| 1.1 Aprobación del documento..... | 4 |
| 1.2 Historial de cambios..... | 4 |
| 1.3 Documentación relacionada..... | 4 |
| 2 Introducción..... | 6 |
| 2.1 Objetivo..... | 6 |
| 2.2 Audiencia..... | 6 |
| 2.3 Terminología y acrónimos..... | 6 |
| 3 Descripción de la solución..... | 8 |
| 4 Diseño técnico de la solución..... | 9 |
| 4.1 Enfoque del diseño..... | 9 |
| 4.2 Descripción de los componentes funcionales..... | 9 |
| 4.3 Resumen de las relaciones entre los componentes funcionales..... | 12 |
| 4.3.1 Nodo controlador de OpenStack..... | 13 |
| 4.3.1.1 Descripción..... | 14 |
| 4.3.1.2 Especificaciones de la tecnología..... | 14 |
| 4.3.1.3 Recursos requeridos..... | 15 |
| 4.3.2 Nodo de virtualización de OpenStack..... | 15 |
| 4.3.2.1 Descripción..... | 16 |
| 4.3.2.2 Especificaciones de la tecnología..... | 17 |
| 4.3.2.3 Recursos requeridos..... | 17 |
| 4.4 Roles y responsabilidades..... | 18 |
| 4.4.1 Definición de roles..... | 18 |
| 4.5 Descripción del proceso..... | 20 |
| 4.5.1 Descripción general a alto nivel..... | 20 |
| 4.5.2 Niveles del proceso..... | 21 |
| 4.6 Restricciones en la aplicación de los componentes funcionales asociados con la solución..... | 22 |
| 4.7 Consideraciones del diseño relacionadas con el desempeño..... | 23 |
| 4.8 Niveles y métodos de tolerancia a fallos..... | 23 |
| 4.8.1 Redundancia a nivel de conectividad..... | 24 |
| 4.8.1 Redundancia a nivel de almacenamiento..... | 25 |

4.10 Seguridad.....26

5 Contactos.....28

1 Control del documento

1.1 Aprobación del documento

A continuación, se mencionan las personas que revisaron este documento, verificaron su información y aprobaron el contenido con base en los requerimientos de los servicios definidos durante su creación:

| Nombre | Rol | Fecha de aprobación |
|-------------------------------|-----------------|---------------------|
| M. Sc. Herson Esquivel Vargas | Profesor asesor | 25 nov 2014 |
| M. Sc. Jaime Gutiérrez Álfaro | Lector interno | 25 nov 2014 |
| M. Sc. Jorge Ortega Pérez | Lector externo | 25 nov 2014 |
| | | |
| | | |
| | | |
| | | |

1.2 Historial de cambios

La siguiente bitácora de cambios contiene un registro de las modificaciones que se han hecho en este documento:

| Publicado / Revisado | Versión # | Autor | Sección / Naturaleza del cambio |
|----------------------|-----------|---|---|
| 25 Nov 2014 | 1.0 | Ing. Jennifer Vargas | Versión final de la guía de referencias técnicas. |
| 16 Feb 2015 | 1.1 | Bach. Cristina Vargas Ramírez Ing. Jennifer Vargas | Revisión Filológica y correcciones. |

1.3 Documentación relacionada

Los siguientes documentos son necesarios para utilizar apropiadamente esta guía. Asegúrese de tener acceso a dichos documentos al consultarla.

| Nombre del documento | Pequeña descripción |
|---|---|
| Guía de operaciones – Servicios de infraestructura en la nube con OpenStack | Esta guía posee todos los detalles de mantenimiento, soporte y operatividad de la solución. |

| Nombre del documento | Pequeña descripción |
|--|---|
| Guía de implementación – Servicios de infraestructura en la nube con OpenStack | Esta guía posee todos los detalles de configuración, instalación e implementación de la solución. |

2 Introducción

Esta sección describe el propósito, la audiencia y la terminología de este documento.

2.1 Objetivo

Brindar información técnica sobre la solución para dar servicios de infraestructura en la nube utilizando OpenStack. El propósito es proporcionar detalles sobre la solución, entre los cuales se incluye la arquitectura y diseño de la solución, las tecnologías utilizadas a nivel de software, el equipo recomendado y las limitaciones técnicas.

2.2 Audiencia

Esta guía ha sido creada para que la utilicen, principalmente, pequeñas y medianas empresas, o bien personas de Costa Rica que estén interesadas en implementar una solución que proporcione servicios de infraestructura en la nube, ya sea para uso personal, uso interno de su empresa o para proveer servicios de hospedaje a empresas externas.

Además, se enfoca en el uso por parte de las PYMES costarricenses, pero no se limita a ellas y puede ser consultada por otro tipo de audiencia. Asimismo, estas guías se pueden distribuir y modificar libremente, si se respetan los términos de su licencia (CC BY-SA 3.0 CR).

2.3 Terminología y acrónimos

En esta sección se define la terminología y los acrónimos comúnmente utilizados en el documento.

| Término/Acrónimo | Definición |
|-------------------|---|
| Nodo | Equipo virtual o físico en el cual se instala un sistema operativo con sus respectivos servicios. |
| Aprovisionamiento | Significa proveer, acondicionar y habilitar un servicio para que el usuario final se pueda beneficiar de él. Es decir, que satisfaga sus requerimientos con un servicio bajo demanda con la calidad acordada. |

| Término/Acrónimo | Definición |
|-----------------------|--|
| Escalabilidad | Indica la habilidad de un sistema para reaccionar y adaptarse a las necesidades continuas del negocio sin perder calidad. Se debe manejar un crecimiento continuo de trabajo de manera fluida, o bien estar preparado para crecer sin perder la calidad de los servicios ofrecidos. |
| Orquestación | Describe el arreglo, automatización, coordinación y gestión de sistemas de computadores complejos, así como servicios. Trata de alinear las solicitudes del negocio con las aplicaciones, los datos y la infraestructura. |
| Horizon | Es un servicio que proporciona un portal web, el cual interactúa con los distintos servicios de <i>OpenStack</i> , por ejemplo: creación de instancias virtuales, asignación de direcciones IP y configuración de controles de acceso. |
| Nova/Compute | Se encarga de gestionar el ciclo de vida de las instancias virtuales en un ambiente OpenStack, así como del aprovisionamiento y deprovisionamiento de máquinas virtuales bajo demanda. |
| Neutrón | Se trata de un componente de OpenStack que se encarga de proporcionar el servicio de red para los otros servicios o componentes de OpenStack. Posee un API para que los usuarios puedan definir sus redes, así como cualquier dispositivo conectado a ellas. Soporta dispositivos de red de una gran cantidad de vendedores y tecnologías. |
| Swift | Objeto de almacenamiento de OpenStack. Es altamente tolerante a fallos, posee una arquitectura escalable y redundante que replica datos a través de todos sus nodos. |
| Cinder | Proporciona bloques de almacenamiento persistentes a instancias de máquinas virtuales en tiempo de ejecución. |
| Keystone | Este componente es el encargado de proveer servicios de autenticación y autorización a los demás componentes de OpenStack. |
| Glance | Se encarga de almacenar y obtener imágenes de las máquinas virtuales. OpenStack usa este componente durante la fase de aprovisionamiento. |
| Ceilometer/Telemetría | Se encarga de la monitorización y medición de la nube de OpenStack para propósitos financieros, de control de calidad, escalabilidad y estadísticos. |
| Heat | Es el servicio de orquestación de OpenStack. |
| Trove | Este es un nuevo componente de Openstack liberado en la versión Juno . Provee "Base de datos como un servicio" tanto para las bases de datos relacionales como no relacionales. |

3 Descripción de la solución

El presente documento se basa en las propiedades relevantes de la solución para proveer servicios de infraestructura en la nube utilizando OpenStack. Dicha solución fue diseñada con versiones de software específicas y toma en cuenta los procesos y roles de quienes podrían consultar las guías.

Además, en el documento se recomiendan las propiedades del equipo por utilizar, pero no se especifican modelos o marcas. El equipo específico deberá ser analizado por la empresa o persona interesada en implementar esta solución, de acuerdo con su capacidad financiera. Las especificaciones de los equipos deben ser consideradas requerimientos mínimos, pues cualquier variación de la capacidad de los recursos del equipo podría invalidar o poner en riesgo el rendimiento de la solución.

4 Diseño técnico de la solución

Esta sección describe el hardware, el software y las consideraciones de diseño que componen esta solución, según los componentes necesarios para implementar el servicio.

4.1 Enfoque del diseño

Para implementar esta solución IaaS, se utiliza OpenStack en su versión **Juno** y algunos de sus servicios como Glance, Keystone, Horizon, Nova, Compute. Además, se usa el servidor de base de datos MariaDB y el servidor de mensajes RabbitMQ, para el manejo de la comunicación entre componentes. El sistema operativo que se implementará se basa en Linux, debido a los requerimientos de la propia aplicación.

Para su implementación se necesitan dos servidores como mínimo. Es preferible que ambos sean redundantes en todos sus componentes para garantizar la estabilidad y disponibilidad de los sistemas.

Los equipos recomendados son servidores pequeños que cumplen con los requerimientos básicos para este tipo de implementación. Puesto que no se especifican las marcas de los equipos, sino las características necesarias, queda en manos de la empresa o persona que desea utilizar esta solución seleccionar su equipo de conveniencia, al que se le pueden ajustar las características sin disminuir los recursos recomendados. Además, es importante contar con una persona que posea conocimientos en Administración Linux y uso básico mediante línea de comandos (terminal) para una implementación exitosa.

4.2 Descripción de los componentes funcionales

Esta sección describe de manera técnica cada uno de los componentes funcionales de la solución. En ella no se utilizarán todos los componentes lógicos con los que cuenta OpenStack, pero sí los principales. Seguidamente, se muestra la interacción que posee OpenStack con algunos de sus componentes lógicos:

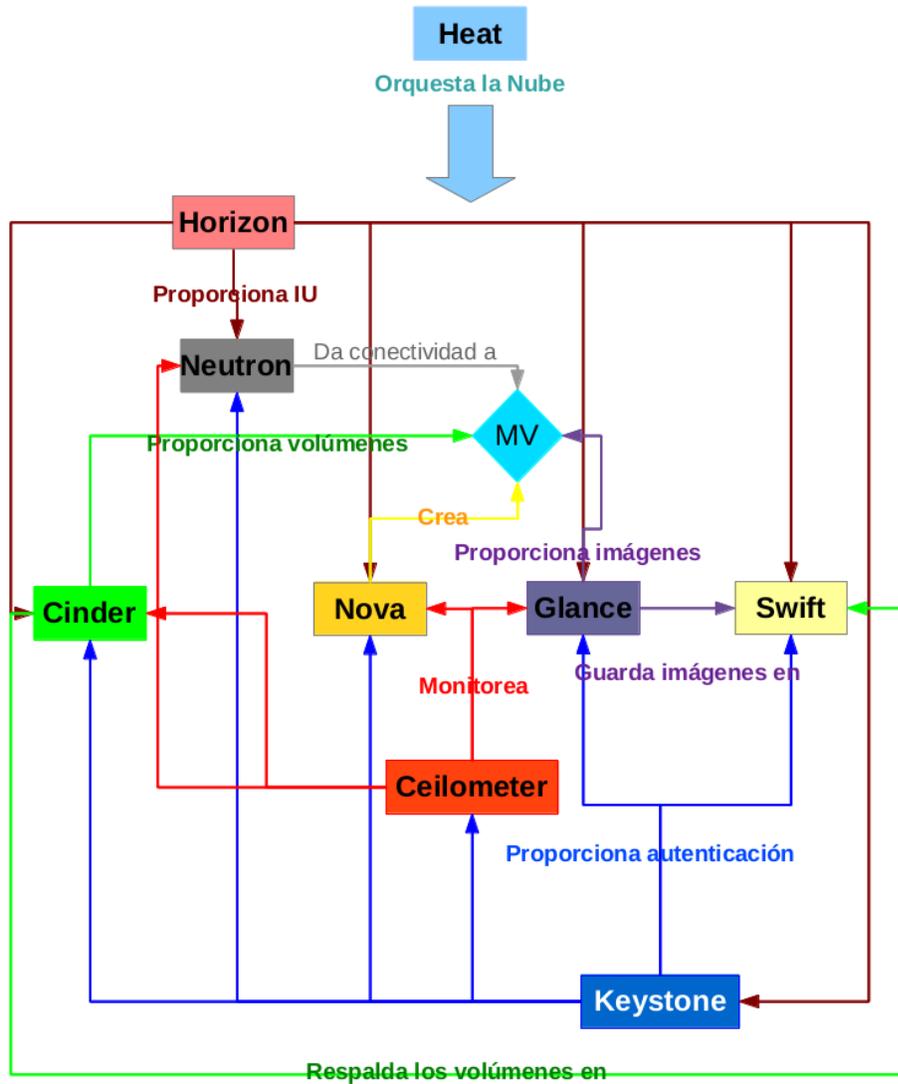


Figura #1. Componentes de OpenStack

El siguiente diagrama muestra los componentes que serán utilizados en esta solución:

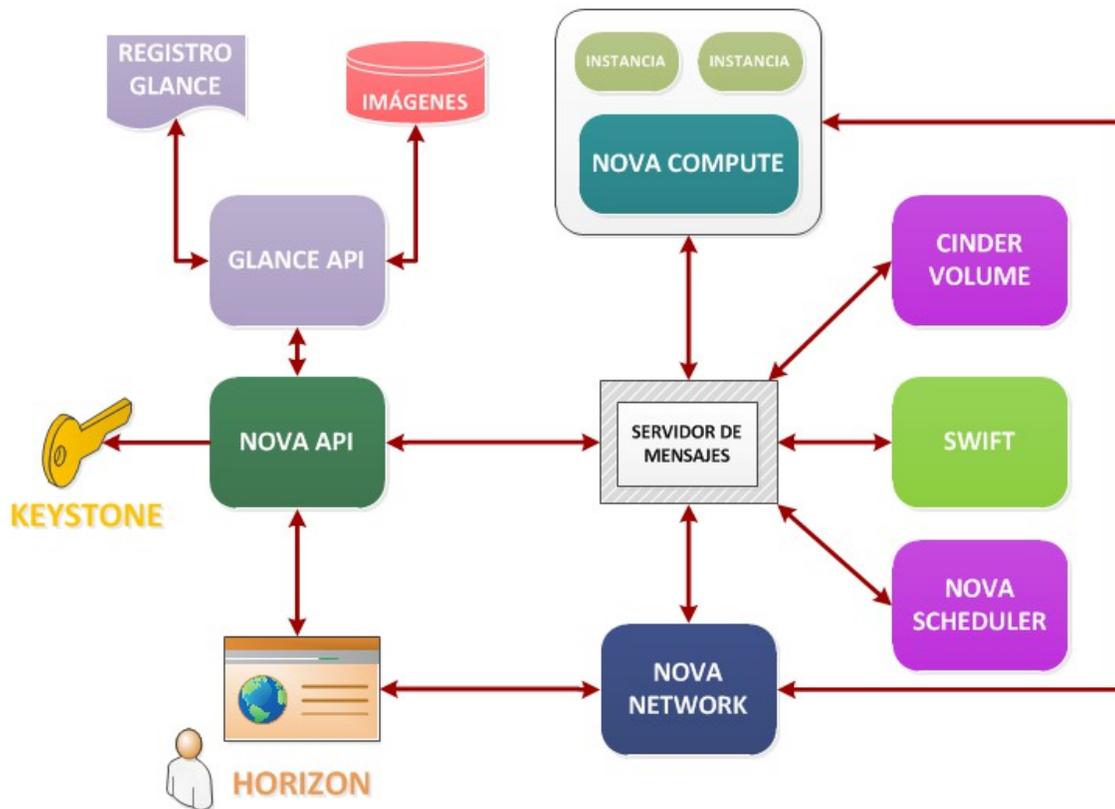


Figura #2. Interacción de servicios de OpenStack

Dichos componentes interactúan de la siguiente manera:

- Un usuario interactúa con la API de Nova (directamente, a través del API o, indirectamente, a través de Horizon) para ejecutar/crear una instancia virtual.
- El servicio Nova-API le pedirá que se autentique previamente con Keystone.
- Una vez autenticado le mostrará las imágenes disponibles en Glance.
- Cuando seleccione una imagen y las características para la instancia (Flavor), la petición se enviará a Nova-scheduler.
- El servicio Nova-scheduler determinará en qué nodo debe ejecutarse la instancia virtual en caso de existir más de un nodo de virtualización.

- El servicio Nova-Compute del nodo seleccionado se encargará de ejecutar la instancia sobre el hipervisor que disponga.

4.3 Resumen de las relaciones entre los componentes funcionales

A continuación, se presentan los componentes funcionales que se utilizarán en la solución de servicios de infraestructura en la nube. Se contará con dos nodos físicos con conectividad a una red pública con acceso a Internet y una red privada para labores de administración.

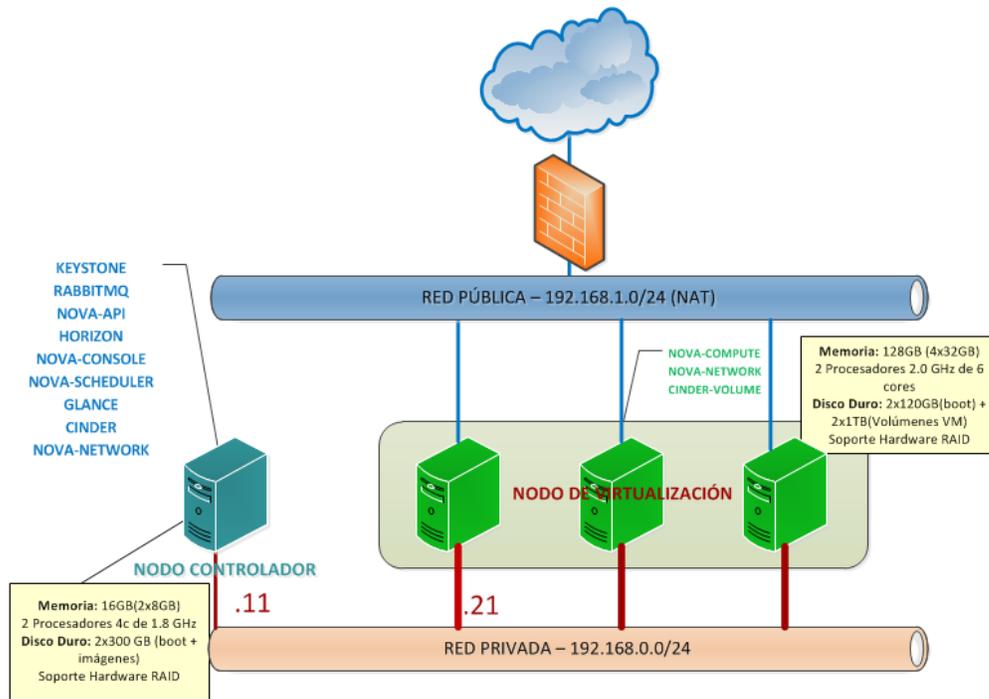


Figura #3. Componentes funcionales

En esta arquitectura se cuenta con dos nodos:

- En el **nodo controlador** se ejecutan los servicios de identidad (Keystone), imágenes (Glance), partes del componente Nova-Network exclusivas para la administración de redes, servicio de base de datos (MariaDB), de mensajería (RabbitMQ), el servicio Cinder, así como el Dashboard (Horizon). Este nodo cuenta con una conexión a la red interna con acceso a Internet por medio de NAT.
- En el **nodo de virtualización** se ejecuta el servicio de hipervisor, en este caso, KVM, Cinder Volume -antes conocido como Nova-Volume- y Nova Network. Dicho nodo se

encarga de la operación de instancias de máquinas virtuales. En este caso, es posible ejecutar más de un nodo de virtualización. Cuantos más nodos, mayor será la cantidad de máquinas virtuales que se puedan crear.

El nodo de virtualización cuenta con conexión a una red privada, que se utiliza para comunicaciones internas y labores administrativas. Además, requiere acceso a Internet por medio de NAT. El o los nodos de virtualización también se encuentran conectados a una red pública con acceso a Internet, que se utilizará para asignar direcciones IP a las instancias de máquinas virtuales.

El diagrama siguiente muestra las características del equipo involucrado en la implementación de la solución de servicio de infraestructura en la nube utilizando OpenStack:

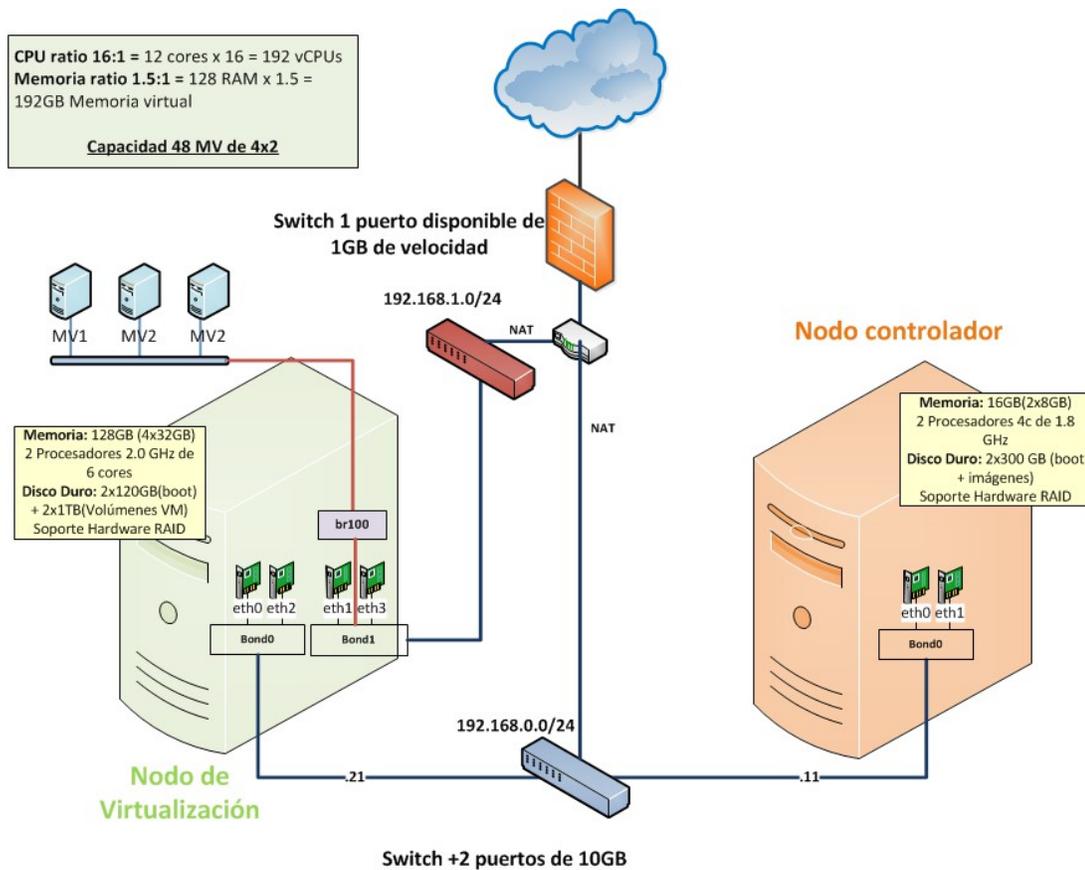


Figura #4. Características del equipo por utilizar y su interacción

4.3.1 Nodo controlador de OpenStack

4.3.1.1 Descripción

Como se mencionó en la sección anterior, este nodo hospeda los servicios más importantes de OpenStack. Sin los servicios que se encuentran instalados en este equipo, OpenStack no podría funcionar. Por eso, es importante garantizar su disponibilidad por medio de la redundancia de sus componentes. Es decir, se debe contar con discos duros redundantes, al menos dos fuentes de poder, dos procesadores, dos DIMMs de memoria, entre otros. En el diagrama que se encuentra a continuación se puede apreciar el diseño de este equipo.

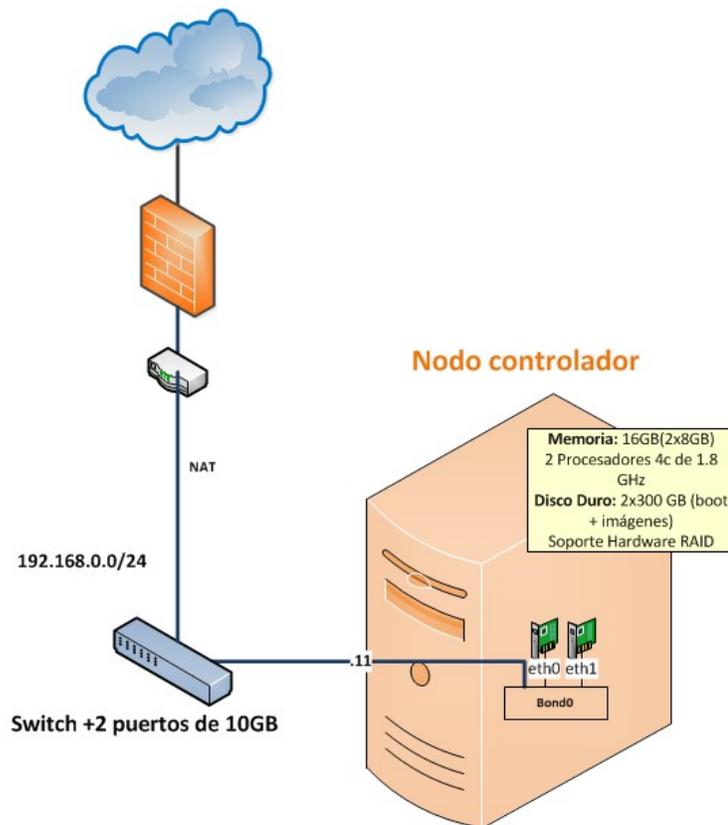


Figura #5. Nodo controlador

4.3.1.2 Especificaciones de la tecnología

Este proyecto no se ha diseñado para un solo cliente, pues es claro que las necesidades pueden variar de cliente en cliente o de empresa en empresa. La marca o el equipo que se utilizará tampoco se menciona en la lista de recursos requeridos, por lo tanto, la empresa o persona interesada en esta implementación deberá decidir qué marcas y modelos de equipos utilizará, de acuerdo con sus posibilidades financieras.

Se recomienda un servidor básico de tipo *rack*, que posea un controlador que soporte RAID por hardware para la redundancia de sus discos duros. El disco duro en el que se va a instalar el sistema operativo debe ser configurado en hardware RAID antes de la instalación del sistema operativo. El sistema operativo se debe instalar sobre el volumen lógico y redundante luego de la creación del RAID por hardware.

Es preferible que las tarjetas para el tráfico interno sean de 10 GbE de velocidad para garantizar un desempeño óptimo durante la creación de las máquinas virtuales, ya que viajan imágenes a través de esa red durante la instalación/creación de esas instancias. Además, se recomienda que ambas tarjetas estén configuradas en **Bonding** de modo activo-pasivo para garantizar la redundancia a nivel de conectividad local. Sin embargo, el uso de tarjetas de 10 GB no es un requerimiento obligatorio. En caso de no contar con la infraestructura necesaria para la utilización de tarjetas de 10 GbE para el manejo del tráfico administrativo, se pueden utilizar tarjetas de 1 GbE, configuradas en Bonding de agregación para obtener un ancho de banda de 2 GB de velocidad.

4.3.1.3 Recursos requeridos

Esta sección describe las características del equipo recomendado para el nodo controlador. Incluye los tipos de hardware y software requeridos por el nodo controlador. La tabla incluye la cantidad de componentes necesarios, una descripción de sus características y un costo estimado. Se excluyen marcas y números de parte de vendedores. Los precios estimados se muestran en dólares americanos.

| Cantidad | Descripción | Precio estimado \$ |
|---|---|--------------------|
| 1 | Servidor standalone/rack con 2 procesadores de 1.8GHZ de 4 cores y 2 tarjetas de red integradas | 3532 |
| 2 | Procesador de 4cores de 1.8GHz | 758 |
| 2 | Dimm de 8GB de memoria RAM | 498 |
| 2 | Disco duro SATA VE 3.5in intemo de 300GB | 1198 |
| 1 | Ethernet 10Gb 2P 530T Adptr | 749 |
| 1 | ISO Ubuntu server 14.04 64 bits | 0 |
| 1 | Servidor MariaDB 15.1 Distrib 5.5.40-MariaDB | 0 |
| 1 | rabbitmq_server-3.2.4 | 0 |
| 1 | OpenStack Juno | 0 |
| Total costo estimado en dólares americanos | | 6735 |

Tabla #1. Lista de recursos requeridos por el nodo controlador

4.3.2 Nodo de virtualización de OpenStack

4.3.2.1 Descripción

El nodo de virtualización es aquel que alberga las instancias de máquinas virtuales creadas en OpenStack. Este último es escalable de manera horizontal, por lo que se pueden agregar tantos equipos como sea necesario para extender las capacidades de la solución.

En la arquitectura de esta solución se incluye, solamente, un equipo para hospedar las máquinas virtuales, de modo que es importante garantizar su disponibilidad mediante el uso de componentes redundantes (discos duros redundantes, al menos dos fuentes de poder, dos procesadores, dos DIMMs de memoria, entre otros). En el diagrama que se muestra a continuación se puede apreciar el diseño de este equipo.

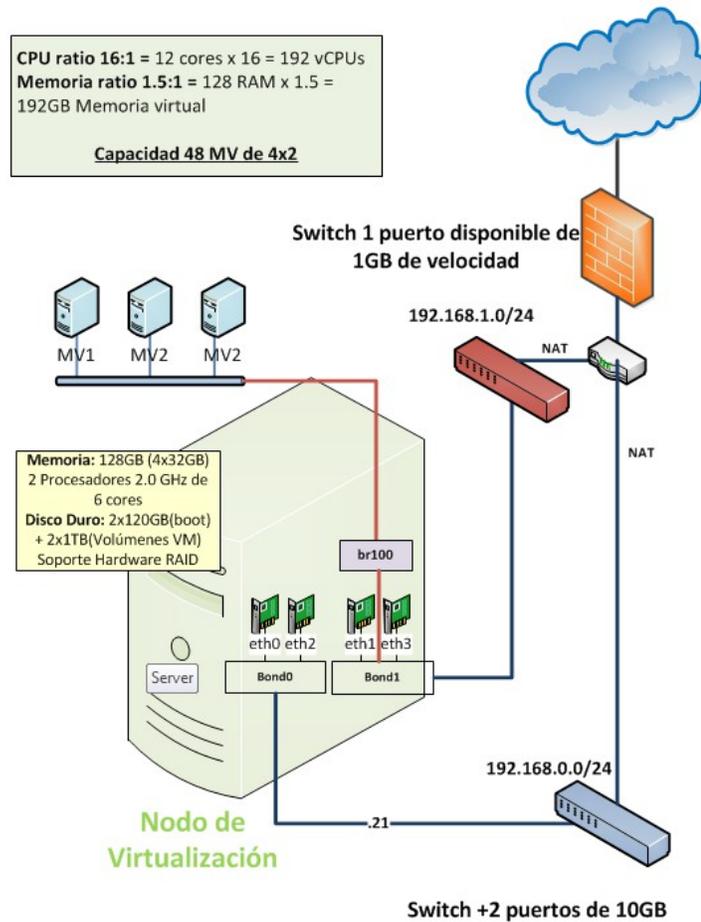


Figura #6. Nodo de virtualización

4.3.2.2 Especificaciones de la tecnología

Como se ha establecido anteriormente, la marca o el modelo por utilizar en los equipos no se mencionarán en la lista de recursos requeridos. La empresa o persona interesada en esta implementación deberá decidir qué marcas y modelos de equipos utilizará, de acuerdo con sus posibilidades financieras y sus necesidades específicas. Sin embargo, se brindan los detalles del equipo recomendado, sus características y, aproximadamente, su costo en dólares americanos al momento de crear esta guía.

Se recomienda un servidor básico de tipo *rack*, que posea un controlador que soporte RAID por hardware para la redundancia de sus discos duros. El disco duro en el que se va a instalar el sistema operativo debe ser configurado en hardware RAID antes de su instalación. Además, el último se debe instalar sobre el volumen lógico y redundante luego de la creación del RAID por hardware. El disco en el que van a almacenarse los volúmenes de las máquinas virtuales también debe ser configurado en RAID N+1, por medio de hardware para garantizar la redundancia de los datos almacenados en ese dispositivo.

En el nodo de virtualización se recomienda que las tarjetas para el tráfico interno sean de 10 GbE de velocidad, con el fin de garantizar un desempeño óptimo durante la creación de las máquinas virtuales. De igual manera, es preferible que las tarjetas, tanto para tráfico interno como externo, estén configuradas en *bonding*, con el fin de garantizar la redundancia a nivel de conectividad. Ambos "bondings" deben utilizar el modo agregación para tener un ancho de banda del doble de la velocidad de las tarjetas utilizadas.

4.3.2.3 Recursos requeridos

Esta sección describe las características del equipo recomendado para el nodo de virtualización. Incluye cualquier tipo de hardware y software requerido por este nodo. La tabla que se muestra incluye la cantidad de componentes necesarios, así como una descripción de sus características y el costo estimado en dólares americanos. Se excluyen marcas y números de parte de vendedores.

| Cantidad | Descripción | Precio estimado \$ |
|---|---|--------------------|
| 1 | Servidor Standalone/Rack con 2 procesadores de 2.0GHZ de 6 cores y 2 tarjetas de red integradas | 3581 |
| 2 | Procesador de 6cores de 2.0GHz | 1030 |
| 4 | Dimm de 32 GB de memoria RAM | 3996 |
| 2 | Disco duro SATA 7.2k 3.5in interno de 1TB | 538 |
| 2 | Disco duro SATA VE 3.5in interno de 120GB | 798 |
| 1 | Ethernet 10Gb 2P 530T Adptr | 749 |
| 1 | ISO Ubuntu server 14.04 de 64 bits | 0 |
| 1 | OpenStack Juno | 0 |
| Total costo estimado en dólares americanos | | 10692 |

Tabla #2. Lista de recursos requeridos por el nodo de virtualización

De la lista de recursos requeridos se excluyen los switches y el firewall mostrados en los diagramas de arquitectura y diseño. La idea de este proyecto es que sea implementado sobre una infraestructura existente, la cual debe contar con ese tipo de elementos. En caso de carecer de switches y firewall, a continuación se proporcionan los detalles de los equipos que se podrían adquirir y su costo estimado en dólares americanos:

| Cantidad | Descripción | Precio estimado \$ |
|--|---|--------------------|
| 1 | Switch 28 puertos 1GB y 2 puertos 10GB ethernet | 1800 a 3200 |
| 1 | switch 4 puertos de 10GbE | 4300 |
| 1 | firewall + IPS appliance | 2972 |
| Costo total aproximado en dólares | | 4772 |

Tabla #3. Lista de recursos requeridos del equipo extra

4.4 Roles y responsabilidades

En este apartado se identifican las responsabilidades asignadas a cada rol involucrado en la implementación y operación de esta solución. Los detalles de las tareas de cada uno de los roles se pueden encontrar en las Guías de implementación y en la Guía de operaciones.

4.4.1 Definición de roles

La siguiente tabla describe los roles necesarios para la entrega de los servicios definidos. Nótese que las funciones detalladas de cada uno de ellos se pueden encontrar en las guías de operaciones e implementación.

| Nombre del rol | Definición corta del rol | Habilidades especiales/capacitación |
|----------------------------|--|--|
| Administrador Linux | Se encarga de la instalación del sistema operativo y la configuración de cada uno de los nodos que se utilizarán , así como de su mantenimiento. Entre sus tareas se encuentran: encargarse de actualizaciones de seguridad del sistema operativo, instalar parches y trabajar de manera muy cercana con el Administrador de OpenStack cuando se deba realizar cualquier tipo de actualizaciones, ya que podría afectar el funcionamiento de alguno de los componentes de OpenStack. | Conocimientos en GNU/Linux, basados en Debian, específicamente, Ubuntu. |
| Instalador de OpenStack | Se encarga de la instalación y configuración del software de OpenStack en cada uno de los nodos, así como de asegurarse de que todos los componentes funcionen de manera adecuada. | Conocimientos en OpenStack y sus componentes. Conocimientos en virtualización mediante KVM. |
| Administrador de OpenStack | Se encarga de administrar la solución una vez implementada, así como de las actualizaciones de software de OpenStack. | Conocimientos a nivel de usuario avanzado de los distintos componentes de OpenStack (Horizon, Glance, Keystone, Nova, KVM, entre otros). |

4.5 Descripción del proceso

4.5.1 Descripción general a alto nivel

En el siguiente diagrama se muestra, paso a paso, el proceso de implementación de la solución. Los procesos en púrpura corresponden a las tareas que se deben ejecutar en el nodo controlador. Por su parte, los procesos en color verde se refieren a las tareas ejecutadas en el nodo de virtualización y los procesos en amarillo, a aquellas tareas que deben ejecutarse a modo de preparación.

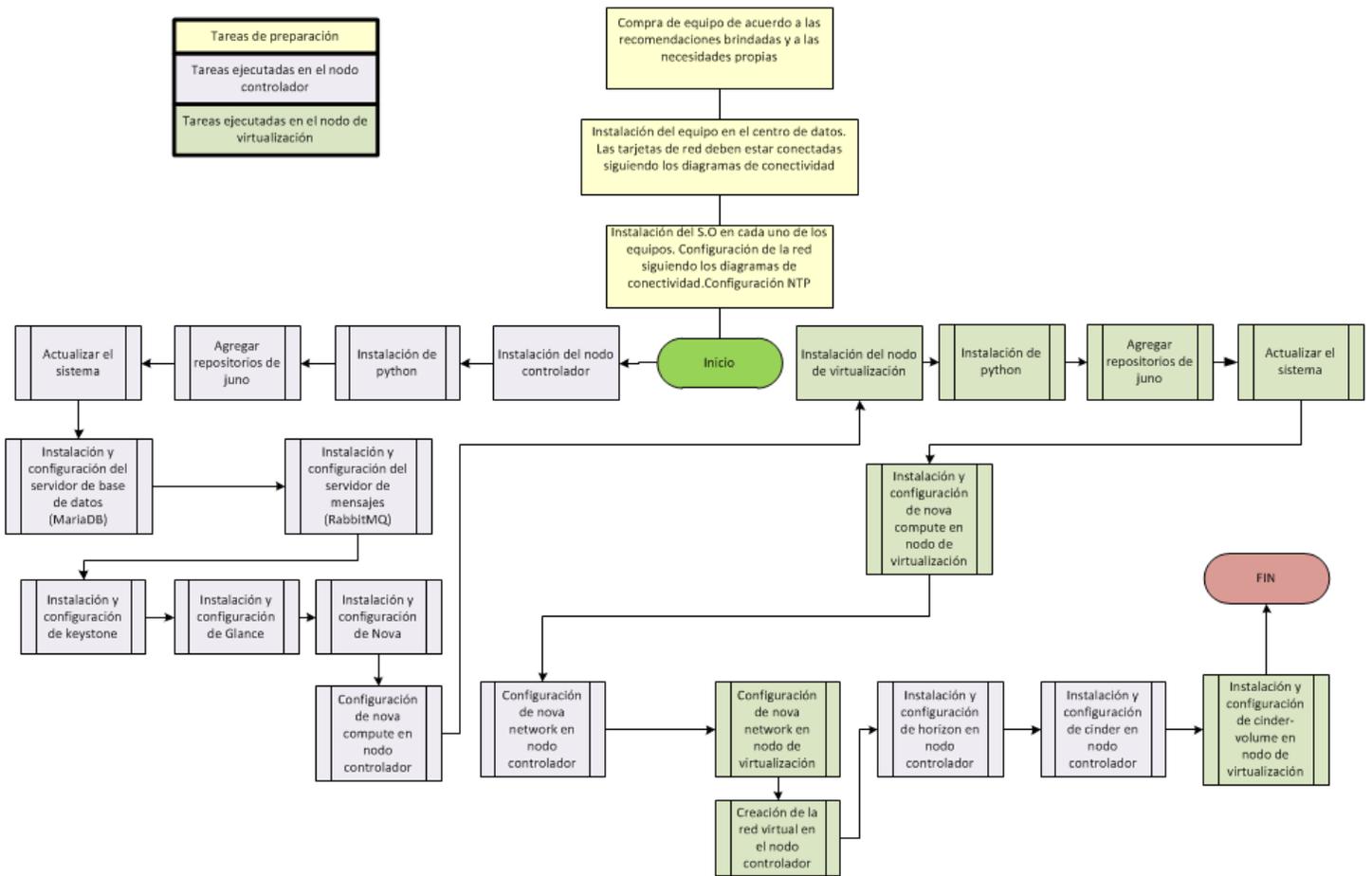


Figura #7. Flujo de implementación de la solución de servicios de infraestructura en la nube

4.5.2 Niveles del proceso

El siguiente diagrama muestra la relación que existe entre los procesos y cada uno de los componentes/servicios que se incluyen en estas guías y en la solución de IaaS basada en OpenStack.

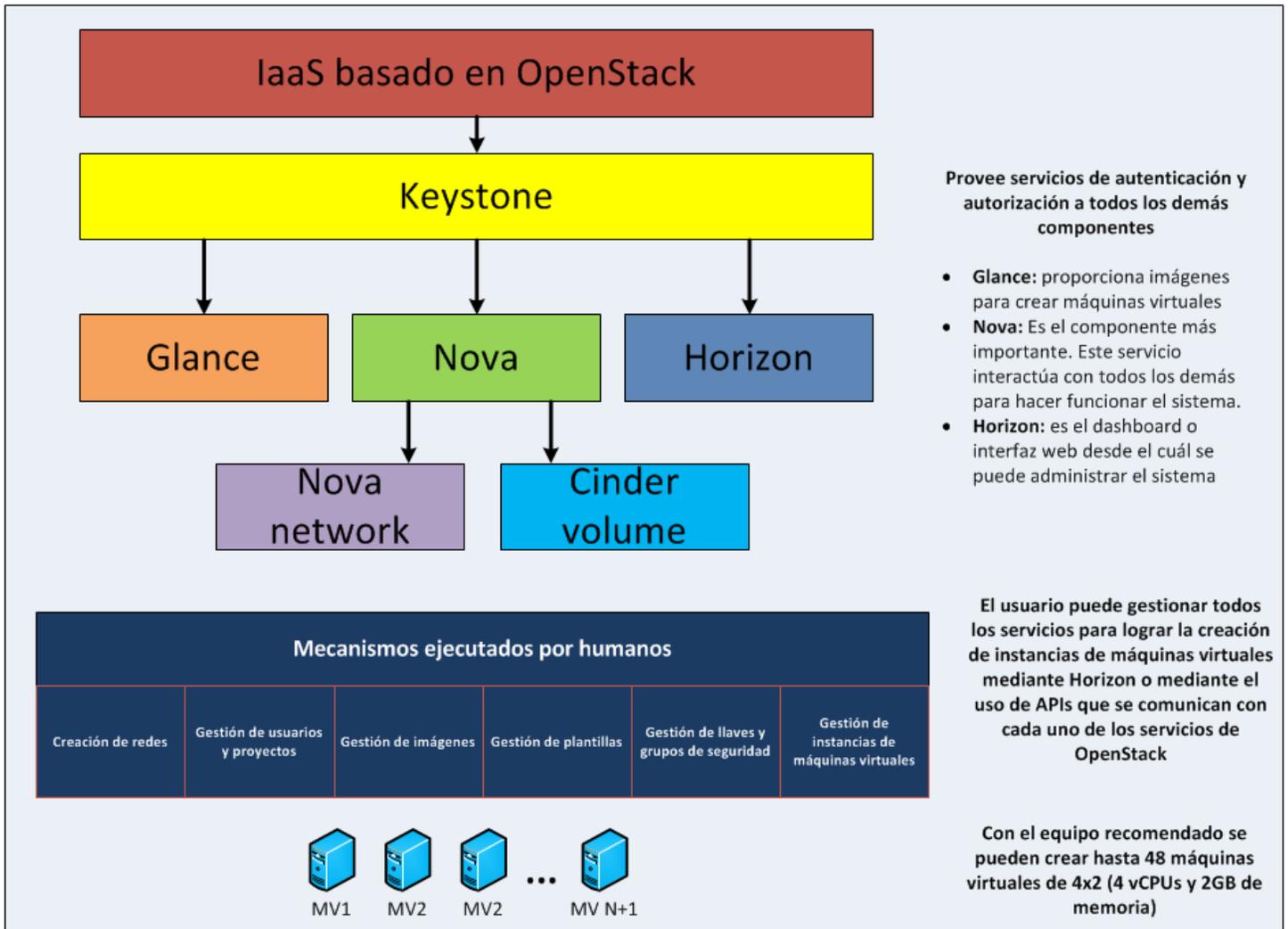


Figura #8. Diagrama de procesos de OpenStack

4.6 Restricciones en la aplicación de los componentes funcionales asociados con la solución

En seguida, se explican los límites de la solución, así como la flexibilidad o excepciones aceptables que presenta.

Como se detalló en la sección anterior, el nodo de virtualización cuenta con 128 de memoria RAM y un total de 12 cores de procesamiento.

En esta solución se utilizan los siguientes ratios que corresponden a los valores estándar de OpenStack:

- Ratio de asignación de CPU: 16:1
- Ratio de asignación de RAM: 1.5:1

El ratio de asignación de CPU de **16:1** significa que el "scheduler" de OpenStack asigna hasta 16 cores virtuales por cada core físico. En este caso, el nodo de virtualización tiene 12 cores, por lo que el "scheduler" de OpenStack posee 192 cores virtuales disponibles para ser utilizados por las máquinas virtuales que se vayan a crear. Si se asignan cuatro cores virtuales a cada instancia, se tendrá una capacidad de 48 instancias virtuales en el nodo físico de 12 cores.

La fórmula para el número de instancias que se puede crear en un nodo de virtualización es la siguiente:

$(OR * PC) / VC$, donde:

- OR es el número de cores virtuales por core físico.
- PC es el número de cores físicos.
- VC es el número de cores virtuales por instancia.

De manera similar, el ratio de asignación de memoria RAM de **1.5:1** significa que el "scheduler" de OpenStack asigna instancias a un nodo físico, siempre que el total de la RAM asociada a esa instancia sea menos de 1.5 veces el tamaño de la RAM disponible en el nodo físico.

En este caso, el nodo de virtualización posee 128 GB de RAM, por lo que el "scheduler" asigna instancias a ese nodo hasta que la suma de la RAM asociada con las instancias llegue a 192 GB.

De este modo, si creamos máquinas virtuales de 2 GB de memoria, podríamos hospedar hasta 48 de esas instancias en el nodo físico.

En resumen, esta solución soporta hasta 48 máquinas virtuales de cuatro CPU virtuales y 2 GB de memoria. Queda en manos del usuario decidir qué tamaño de máquinas virtuales son de su conveniencia, según las necesidades de las aplicaciones en su negocio.

4.7 Consideraciones del diseño relacionadas con el desempeño

OpenStack posee una arquitectura altamente escalable., esto significa que en caso de que la cantidad de recursos en esta solución llegue o se acerque a sus límites, bastará con agregar un equipo extra, que posea las mismas características que el anterior (o similares), a la infraestructura.

Como se mencionó anteriormente, se recomienda usar tarjetas de 10 GbE de velocidad para el manejo del tráfico de administración sobre el cual se van a transportar con frecuencia las imágenes con las que se construirán las instancias de máquinas virtuales. La utilización de tarjetas de 1 GbE para el tráfico interno permite el adecuado funcionamiento de todos los servicios de OpenStack, pero podría afectar los tiempos de respuesta a la hora de crear nuevas instancias.

4.8 Niveles y métodos de tolerancia a fallos

Esta sección describe la estrategia técnica para tolerar fallos de los componentes de esta solución, así como el nivel de redundancia del equipo utilizado.

OpenStack cuenta con una arquitectura redundante por sí misma, de manera que no necesita software de terceros para proporcionar redundancia. Tampoco requiere de un hardware costoso con redundancia en todos los niveles físicos de sus componentes, si se cuenta con más de un equipo que distribuya la carga en cada uno de los nodos (controlador y de virtualización). Es decir, OpenStack está hecho para funcionar en hardware modesto y estándar, gracias a que es escalable de manera horizontal. De esta manera, al agregar más equipo aumenta la capacidad de la solución.

Asimismo, OpenStack proporciona redundancia en caso de algún fallo en uno de los equipos y, puesto que más de uno forma parte de cada nodo controlador o de virtualización, se puede

remover, de manera temporal o permanente, cualquiera de ellos y, a la vez, garantizar que el componente siga funcionando con el nodo o los nodos restantes.

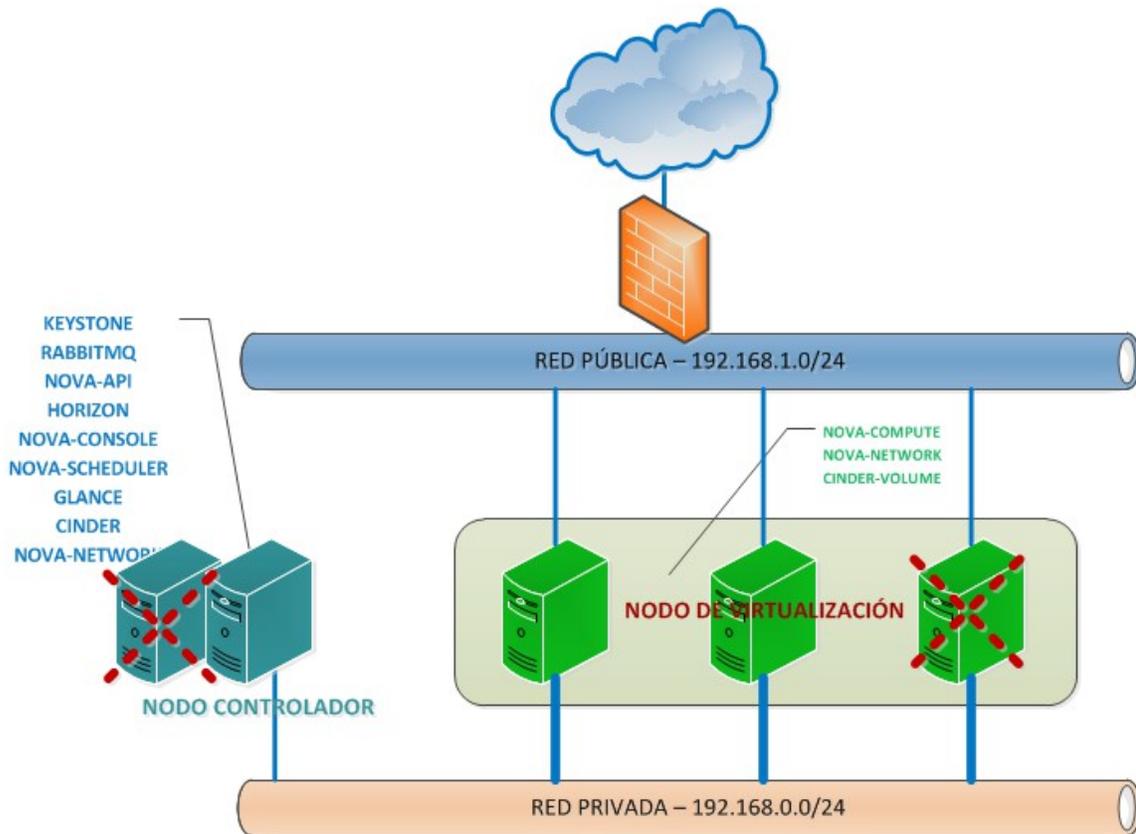


Figura #9. Redundancia de OpenStack

Sin embargo, agregar más equipos a la solución, sin tener claras las necesidades de un cliente específico, aumenta al doble o más su costo. Por eso, en este caso, el diseño dispone de un solo equipo para cada nodo, lo que hace necesario ocuparse de la redundancia de los componentes de cada uno de los servidores utilizados (nodo controlador y nodo de virtualización).

4.8.1 Redundancia a nivel de conectividad

Se recomienda que la infraestructura de red sobre la cual se va a instalar esta solución cuente con componentes redundantes. Es decir, por cada switch o enrutador en la infraestructura, debería existir otro de respaldo configurado en cluster que esté disponible en caso de fallos.

La arquitectura y el diseño de esta solución contemplan la redundancia en sus conexiones al utilizar al menos dos tarjetas para cada red, configuradas en bonding, como se muestra en el diagrama de conectividad, a continuación. Se espera que cualquier otro dispositivo de red utilizado como parte de la infraestructura también sea redundante o posea algún mecanismo de recuperación.

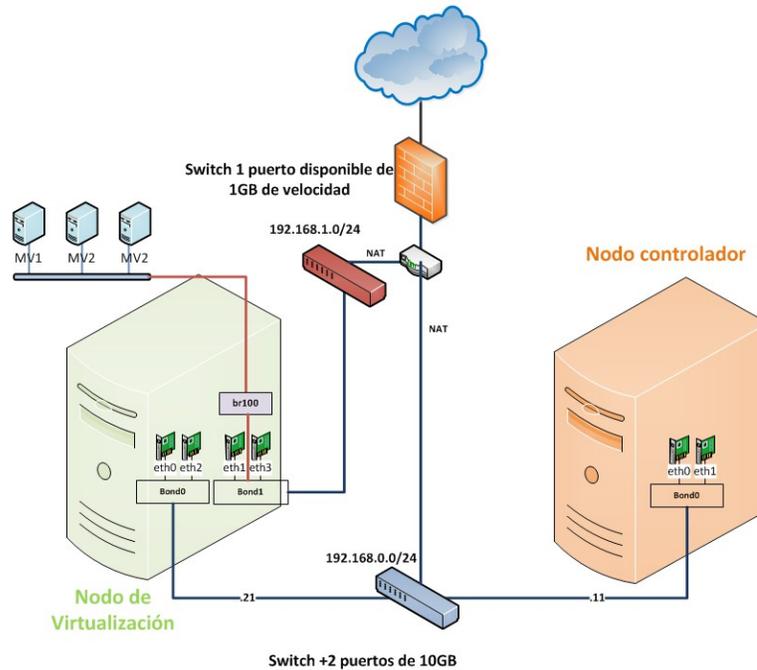


Figura #10. Diagrama de conectividad de red

4.8.1 Redundancia a nivel de almacenamiento

En el diseño de este proyecto no se utilizan dispositivos de almacenamiento externos, por lo que es importante que los dispositivos de almacenamiento internos de cada uno de los servidores cuenten con algún nivel de redundancia.

Como parte de la solución, los servidores utilizados deben soportar RAID por hardware y todos sus dispositivos de almacenamiento deben estar configurados en RAID por hardware de tipo N+1 (modo de tolerancia a fallos).

Seguidamente, se muestra el diagrama de la configuración de los dispositivos de almacenamiento en cada uno de los servidores. En él podrá apreciar que por a cada disco duro interno, hay otro de respaldo en espejo.

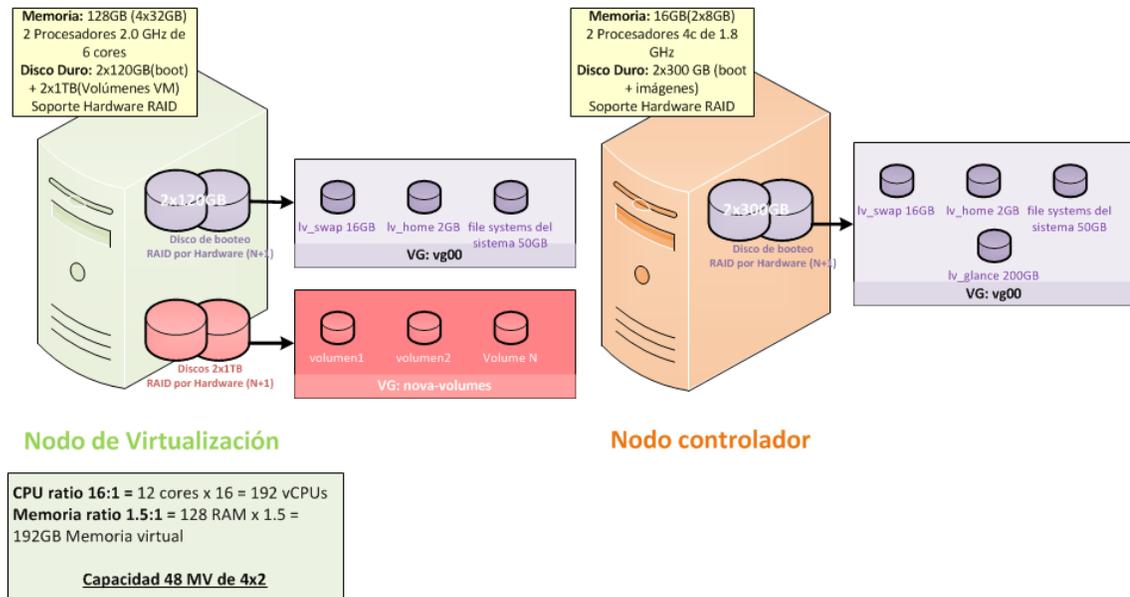


Figura #11. Diagrama de diseño de almacenamiento

4.10 Seguridad

Este apartado describe los componentes de seguridad utilizados en la solución. La puesta en práctica de los mecanismos de seguridad se encuentra detallada en la Guía de implementación.

Los dos nodos que forman parte de la solución poseen configuraciones estándar de seguridad a nivel de sistema operativo, como se observa seguidamente:

- El cceso remoto para root mediante SSH se encuentra restringido.
- Las cuentas de usuarios (locales) están configuradas para expirar de acuerdo con las políticas estándares.
- De acuerdo con la política implementada, la configuración de las contraseñas de usuarios admite longitudes de al menos ocho caracteres, entre los cuales se incluyen especiales y numéricos.

Los servicios de OpenStack soportan varios métodos de seguridad como contraseñas, políticas y cifrado. De manera adicional, los servicios soportados incluyen servidores de bases de datos y de mensajes que soportan al menos seguridad por medio de contraseñas.

Tanto detrás de una red pública como de una privada, debe contemplarse la incorporación de un *firewall* y un IDS que ayude a mitigar cualquier riesgo.

5 Contactos

En seguida, se brinda información de las personas que puede contactar en caso de consultas, cambios técnicos, sugerencias de diseño, etc.

| Nombre | Rol | Dirección de correo-e |
|-----------------|--|------------------------------|
| Jennifer Vargas | Creadora de la guía de referencias técnicas. | jvargas@softwarelibre.cr.org |
| | | |



Guía de implementación– Servicios de infraestructura en la nube con *OpenStack*

1.0 – Versión Final

25 de Noviembre de 2014



Se permite el uso comercial de la obra y de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Licencia Creative Commons Atribución CompartirIgual 3.0 Costa Rica.

<http://creativecommons.org/licenses/by-sa/3.0/cr/legalcode>

Índice

| | |
|--|-----------|
| 1 Control del documento..... | 3 |
| 1.1 Aprobación del documento..... | 3 |
| 1.2 Historial de cambios..... | 3 |
| 1.3 Documentación relacionada..... | 3 |
| 2 Introducción..... | 5 |
| 2.1 Objetivo..... | 5 |
| 2.2 Audiencia..... | 5 |
| 2.3 Terminología y acrónimos..... | 5 |
| 3 Descripción de la solución..... | 7 |
| 4 Prerrequisitos de la solución..... | 8 |
| 4.1 Dependencias de infraestructura de la solución..... | 8 |
| 4.1.1 Dependencias físicas..... | 8 |
| 4.1.2 Dependencias de software..... | 9 |
| 5 Preparación..... | 10 |
| 5.1 Roles y responsabilidades en la fase de preparación..... | 10 |
| 5.2 Tareas de preparación..... | 10 |
| 5.2.1 Configuración de la red..... | 15 |
| 5.2.2 Configuración NTP..... | 17 |
| 6 Consideraciones de implementación..... | 20 |
| 6.1 Consideraciones de tamaño..... | 20 |
| 6.2 Consideraciones adicionales..... | 21 |
| 6.2.1 Tipo de CPU a utilizar en los equipos..... | 21 |
| 6.2.2 Asignación de recursos (CPU y memoria)..... | 21 |
| 6.3 Estándares de configuración..... | 22 |
| 7 Implementación..... | 25 |
| 7.1. Resumen..... | 25 |
| 7.2 Roles de implementación..... | 26 |
| 7.3 Diagrama de procesos..... | 26 |
| 7.4 Actividades de implementación..... | 27 |
| 7.5 Instalación de prerrequisitos..... | 29 |
| 7.6 Instalación del nodo controlador..... | 31 |
| 7.6.1 Instalación del servidor de bases de datos..... | 31 |
| 7.6.2 Instalación del servidor de mensajes..... | 34 |
| 7.6.3 Instalación del servicio de Identidad de OpenStack (Keystone)..... | 35 |
| 7.6.3.1 Configuración de prerrequisitos de Keystone..... | 35 |
| 7.6.3.2 Instalación de los paquetes de Keystone..... | 36 |
| 7.6.3.3 Configuración de los usuarios de Keystone..... | 38 |
| 7.6.3.4 Creación de los scripts (OpenRC) de ambiente de OpenStack..... | 42 |
| 7.6.4 Instalación del servicio de Imágenes de OpenStack (Glance)..... | 44 |
| 7.6.4.1 Configuración de prerrequisitos de Glance..... | 44 |

| | |
|---|-----------|
| 7.6.4.2 Instalación y configuración de Glance..... | 46 |
| 7.6.5 Instalación del servicio Compute de OpenStack (Nova)..... | 51 |
| 7.6.5.1 Configuración de prerequisites de Nova..... | 51 |
| 7.6.5.2 Instalación y configuración de Nova-compute..... | 52 |
| 7.6.6 Configuración del servicio Nova-network en el nodo controlador..... | 55 |
| 7.6.7 Instalación de la Interfaz web de OpenStack (Horizon)..... | 56 |
| 7.6.8 Instalación del controlador de volúmenes (Cinder)..... | 58 |
| 7.6.8.1 Configuración de prerequisites de Cinder..... | 59 |
| 7.6.8.2 Instalación y configuración de Cinder..... | 60 |
| 7.7 Instalación del nodo de virtualización..... | 63 |
| 7.7.1 Instalar y configurar el nodo del servicio Compute (Nova Compute)..... | 63 |
| 7.7.2 Instalación y configuración del servicio Nova-network en el nodo de virtualización. | 67 |
| 7.7.3 Creación de la red virtual en Nova-network..... | 68 |
| 7.7.4 Instalación y configuración de Cinder-volume en el nodo de virtualización..... | 69 |
| 7.8 Solución de problemas..... | 73 |
| 7.9 Pruebas..... | 74 |
| 7.10 Mecanismo de respaldo..... | 75 |
| 8 Contactos..... | 77 |

1 Control del documento

1.1 Aprobación del documento

A continuación, se mencionan las personas que revisaron este documento, verificaron su información y aprobaron el contenido con base en los requerimientos de los servicios definidos durante su creación:

| Nombre | Rol | Fecha de aprobación |
|-------------------------------|-----------------|---------------------|
| M. Sc. Herson Esquivel Vargas | Profesor asesor | 25 nov 2014 |
| M. Sc. Jaime Gutiérrez Álfaro | Lector interno | 25 nov 2014 |
| M. Sc. Jorge Ortega Pérez | Lector externo | 25 nov 2014 |
| | | |
| | | |
| | | |
| | | |

1.2 Historial de cambios

La siguiente bitácora de cambios contiene un registro de las modificaciones que se han hecho en este documento:

| Publicado / Revisado | Versión # | Autor (opcional) | Sección / Naturaleza del cambio |
|----------------------|-----------|---|---|
| 25 nov 2014 | 1.0 | Ing. Jennifer Vargas | Versión final de la guía de implementación. |
| 16 feb 2015 | 1.1 | Bach. Cristina Vargas Ramírez Ing. Jennifer Vargas | Revisión filológica y correcciones. |
| | | | |
| | | | |

1.3 Documentación relacionada

Los siguientes documentos son necesarios para utilizar apropiadamente esta guía. Asegúrese de tener acceso a dichos documentos al utilizarla.

| Nombre del documento | Pequeña descripción |
|--|--|
| Guía de referencias técnicas – Servicios de infraestructura en la nube con OpenStack | Esta guía posee detalles sobre la arquitectura y el diseño de esta solución, incluyendo todos los detalles de los paquetes de software que se utilizarán, así como las características del equipo sugerido. |
| Guía de operaciones – Servicios de infraestructura en la nube con OpenStack | Esta guía posee todos los detalles de mantenimiento, soporte y operatividad de la solución. |
| Documentación oficial de OpenStack | http://docs.openstack.org/ En el repositorio de documentación de <i>OpenStack</i> se puede encontrar toda la documentación del proyecto como guías de usuario, de operatividad, documentación sobre cómo aprender a utilizar todos los API de OpenStack, guías de instalación en las distintas distribuciones de Linux, entre otros. |
| Capacitación de OpenStack | http://docs.openstack.org/training-guides/content/ OpenStack ha puesto a disposición del público guías con ejercicios y capacitaciones que puede aprovechar cualquier persona para aprender acerca del proyecto. |

2 Introducción

El presente apartado describe el propósito, la audiencia y la terminología de este documento.

2.1 Objetivo

El propósito de esta guía radica en proporcionar los detalles de instalación y configuración necesarios, así como las instrucciones que permitan la implementación exitosa de la solución para dar servicios de infraestructura en la nube utilizando OpenStack.

2.2 Audiencia

Esta guía ha sido creada con el fin de apoyar, principalmente, a pequeñas y medianas empresas de Costa Rica, que estén interesadas en la implementación de una solución para proporcionar servicios de infraestructura en la nube, ya sea para uso interno o para dar servicios de hospedaje a empresas externas.

La guía se enfoca en el uso por parte de las PYMES costarricenses, pero no está limitada a ellas y puede ser accedida por otro tipo de audiencia. Estas guías se pueden distribuir y modificar libremente, siempre que se respeten los términos de su licencia (CC BY-SA 3.0 CR).

2.3 Terminología y acrónimos

Seguidamente, se brindan definiciones de la terminología y los acrónimos comúnmente utilizados en este documento.

| Término/Acrónimo | Definición |
|-------------------|--|
| Nodo | Equipo virtual o físico en el cual se instala un sistema operativo con sus respectivos servicios. |
| Aprovisionamiento | Significa proveer, acondicionar y habilitar un servicio para que el usuario final se pueda beneficiar de él. Es decir, que satisfaga sus requerimientos con un servicio bajo demanda con la calidad acordada. |
| Escalabilidad | Indica la habilidad de un sistema para reaccionar y adaptarse a las necesidades continuas del negocio sin perder calidad. Se debe manejar un crecimiento continuo de trabajo de manera fluida o bien estar preparado para expandirse sin perder la calidad de los servicios ofrecidos. |

| Término/Acrónimo | Definición |
|-----------------------|--|
| Orquestación | Describe un arreglo, automatización, coordinación y gestión de sistemas de computadores complejos, así como servicios. Trata de alinear las solicitudes del negocio con las aplicaciones, los datos y la infraestructura. |
| Horizon | Es un servicio que proporciona un portal web, el cual interactúa con los distintos servicios de OpenStack, por ejemplo: creación de instancias virtuales, asignación de direcciones IP y configuración de controles de acceso. |
| Nova/Compute | Se encarga de gestionar todo el ciclo de vida de las instancias virtuales en un ambiente OpenStack, así como del aprovisionamiento y deprovisionamiento de máquinas virtuales bajo demanda. |
| Neutrón | Se trata de un componente de OpenStack que se encarga de proporcionar el servicio de red para los otros servicios o componentes de OpenStack. Posee un API para que los usuarios puedan definir sus redes, así como cualquier dispositivo conectado a ellas. Soporta dispositivos de red de una gran cantidad de vendedores y tecnologías. |
| Swift | Objeto de almacenamiento de OpenStack. Es altamente tolerante a fallos, posee una arquitectura escalable y redundante que replica datos a través de todos sus nodos. |
| Cinder | Proporciona bloques de almacenamiento persistentes a instancias de máquinas virtuales en tiempo de ejecución. |
| Keystone | Este componente es el encargado de proveer servicios de autenticación y autorización a los demás componentes de OpenStack. |
| Glance | Se encarga de almacenar y obtener imágenes de las máquinas virtuales. OpenStack usa este componente durante la fase de aprovisionamiento. |
| Ceilometer/Telemetría | Se encarga de la monitorización y medición de la nube de OpenStack para propósitos financieros, de control de calidad, escalabilidad y propósitos estadísticos. |
| Heat | Es el servicio de orquestación de OpenStack. |
| Trove | Este es un nuevo componente de Openstack, liberado en la versión Juno . Provee "Base de datos como un servicio" tanto para las bases de datos relacionales como las no relacionales. |

3 Descripción de la solución

En seguida, se describe la solución, de qué manera funciona y sus componentes principales.

Este documento se basa en las propiedades relevantes de la solución para proveer servicios de infraestructura utilizando OpenStack. La solución se diseñó con versiones de software específicas y toma en cuenta los procesos y roles de quienes podrían hacer uso de las guías. Además, se proporcionan los detalles de instalación y configuración de todos los componentes de software requeridos.

4 Prerrequisitos de la solución

En esta sección se incluyen las acciones, prerrequisitos, versiones de software a utilizar y cualquier otra dependencia requerida antes de implementar esta solución.

4.1 Dependencias de infraestructura de la solución

Esta sección menciona los prerrequisitos y correquisitos para la implementación de la solución.

4.1.1 Dependencias físicas

A continuación, se describen algunos activos físicos que son necesarios para la infraestructura en la cual se va a implementar la solución.

- Debe existir una infraestructura de red con enrutadores que permitan el acceso a Internet, tanto a los servidores que serán utilizados para la instalación de componentes de *OpenStack* como a las instancias de máquinas virtuales que serán creadas.
- Es necesario contar con un switch que disponga de dos puertos de 10 GbE para poder conectar las tarjetas de 10 GbE en la red interna. Si no es posible conseguir puertos de 10 GbE y los servidores solo cuentan con puertos de 1 GbE, aún se puede implementar la solución. Sin embargo, se debe tener claro que se carecería de las características recomendadas para alcanzar un desempeño óptimo.
- Todos los servidores deben ser capaces de resolver nombres internos y externos, por lo que es necesario contar con acceso a un servidor DNS en la infraestructura o utilizar una resolución local de nombres en su defecto (`/etc/hosts`).
- Es altamente recomendada la existencia de un firewall con IPS a la entrada de su infraestructura de red.
- Los servidores incluidos en esta solución son de tipo rack, por lo que se necesitan 2 U de espacio por cada servidor en uno o más racks ya existentes, así como sistemas adecuados de ventilación y eléctricos en el centro de datos.

4.1.2 Dependencias de software

OpenStack tiene los siguientes requerimientos mínimos para cada uno de los nodos que vamos a utilizar:

- **Nodo controlador:** necesita al menos un procesador, 2 GB de memoria y 5 GB de almacenamiento.
- **Nodo de virtualización:** necesita al menos un procesador, 2 GB de memoria y 10 GB de almacenamiento.

Además, está soportado en una gran variedad de distribuciones de GNU/Linux, entre las que se pueden mencionar:

- Ubuntu
- Red Hat
- Fedora
- CentOS
- Suse

Es importante que el sistema operativo sobre el cual se va a instalar OpenStack sea de 64 bits y que el equipo soporte aceleración por hardware para un mejor desempeño. Igualmente, el equipo a utilizar en el nodo de virtualización debe soportar virtualización.

OpenStack Posee licencia Apache 2.0 y soporta varios hipervisores, tales como VMware ESX y ESXi, MS Hyper-V, Xen, KVM/QEMU y Virtual Box.

5 Preparación

5.1 Roles y responsabilidades en la fase de preparación

En esta sección se identifican los roles necesarios para realizar las tareas de preparación para implementar la solución. Puede consultar “Roles y Responsabilidades”, apartado que se encuentra en la “Guía de referencias técnicas”, para conocer las definiciones de algunos de los siguientes roles.

Administrador de Linux

Esta persona debe encargarse de la instalación y configuración del sistema operativo en todos los servidores que serán utilizados para implementar esta solución, como se define seguidamente.

Instalador de OpenStack

Esta persona debe tener conocimientos en Linux y debe saber trabajar en línea de comandos (terminal). Debe seguir todas las instrucciones que se encuentran en la sección de implementación de este documento.

Administrador de OpenStack

Esta persona será el usuario del sistema de OpenStack una vez instalado. Es importante que realice todas las tareas explicadas en la Guía de operaciones para asegurar el funcionamiento correcto de la solución y certificar adecuadamente su implementación.

5.2 Tareas de preparación

Esta sección describe las tareas necesarias para prepararse para implementar esta solución.

En el presente documento no se incluyen los detalles de configuración del sistema operativo ni de las interfaces de red. Se da por sentado que se cuenta con el conocimiento necesario para realizar operaciones de administración avanzadas a nivel de sistema operativo.

Antes de proceder con la instalación de OpenStack es necesario realizar las siguientes tareas:

- Instalar Ubuntu server 14.04 LTS de 64 bits en todos los nodos.

- Configurar los discos duros internos en modo RAID + 1 (configuración de RAID por hardware) y el almacenamiento de la siguiente manera:
 - Nodo controlador:
 - Este nodo posee dos discos duros internos de 300 GB. Ambos discos duros deben ser configurados en RAID N+1 (modo de redundancia de fallos). Para obtener los detalles sobre cómo realizar la configuración de RAID por hardware, lea los manuales de usuario del fabricante.
 - Se debe utilizar LVM como manejador de las estructuras y configurarse de la siguiente manera:
 - Crear un grupo de volúmenes llamado vg00.
 - Crear un volumen lógico en el cual se instalará el sistema operativo con un tamaño de 50 GB.
 - Crear un volumen lógico para la swap (lv_swap) con un tamaño de 16 GB.
 - Crear un volumen lógico para el sistema de ficheros /home (lv_home) con un tamaño de 2 GB.
 - Deberá dejar 200 GB de espacio disponible en este grupo de volúmenes que será utilizado por Glance, como se especifica más adelante en la fase de implementación.
 - Nodo de virtualización:
 - Este nodo posee 2 discos duros internos, cada uno de 120 GB y 2 de 1 TB. Los discos de 120 GB deben ser configurados en RAID N+1 (RAID por hardware) para formar un solo volumen con capacidad de redundancia en caso de fallos. Los discos de 1 TB también deben ser configurados en modo RAID N+1 (RAID por hardware). Para obtener los detalles sobre cómo realizar la configuración de RAID por hardware, lea los manuales de usuario del fabricante de los servidores adquiridos.

- Se debe utilizar LVM como manejador de las estructuras y configurarse de la siguiente manera:
 - Con el disco redundante de 120 GB deberá crear un grupo de volúmenes llamado **vg00** y dentro de él:
 - Crear un volumen lógico donde será instalado el sistema operativo con un tamaño de 50 GB.
 - Crear un volumen lógico con un tamaño de 16 GB para la swap (*lv_swap*).
 - Crear un volumen lógico para el sistema de ficheros /home (*lv_home*) con un tamaño de 2 GB.
 - Con el disco redundante de 1 TB, deberá crear un grupo de volúmenes con el nombre **nova-volumes**. Este volumen será utilizado por **Cinder** de *OpenStack* para crear volúmenes que se presentarán a las instancias de máquinas virtuales.

En el diagrama que se observa en seguida puede encontrar los detalles de la configuración del almacenamiento que se ha explicado en los puntos anteriores:

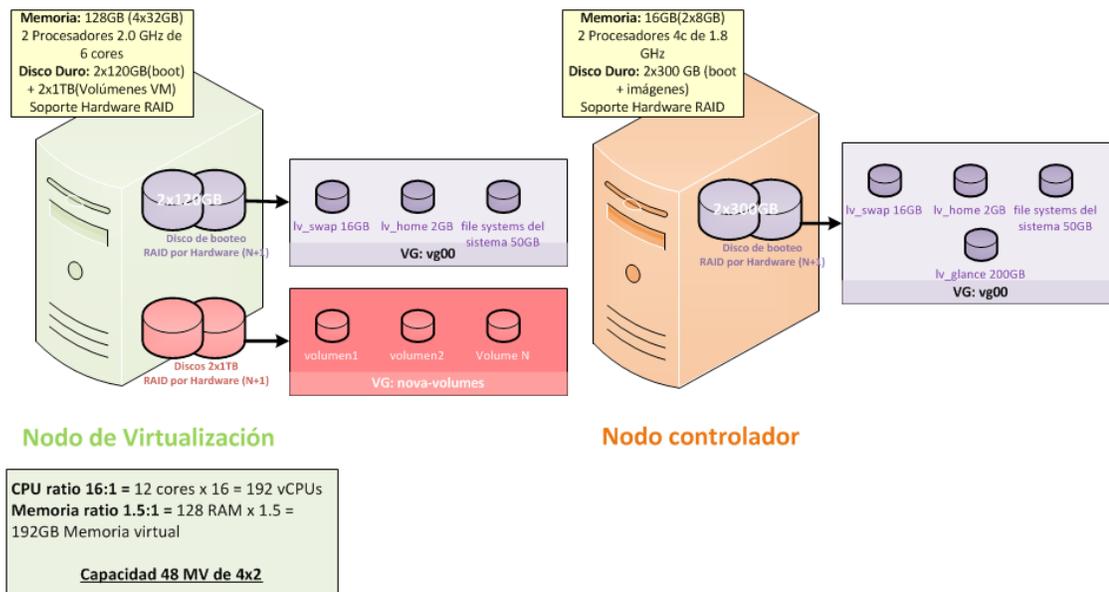


Figura #1. Diseño de almacenamiento

- Configurar las tarjetas de red:
 - **Nodo controlador:**
 - eth0 y eth1 deben estar configurados en bonding haciendo balanceo de carga (bond0). Estas tarjetas deben ser, preferiblemente, de 10 GbE.
 - Para obtener detalles sobre la configuración de tarjetas que utilizan bonding, acuda a los centros de documentación de Ubuntu.
 - Asignar a bond0 la IP correspondiente en la red privada para labores de administración. En el caso de este diseño la IP a utilizar es 192.168.0.11 con máscara 255.255.255.0.
 - **Nodo de virtualización:**
 - eth0 y eth2 deben estar configurados en bonding haciendo balanceo de carga (bond0). Estas tarjetas deben ser, preferiblemente, de 10 GbE.
 - eth1 y eth3 deben estar configurados en bonding haciendo balanceo de carga (bond1).
 - Para obtener detalles sobre la configuración de tarjetas utilizando bonding, acuda a los centros de documentación de Ubuntu.
 - Asignar a bond0 la IP correspondiente en la red privada para labores de administración. En el caso de este diseño la IP a utilizar es 192.168.0.21 con máscara 255.255.255.0.
 - Bond1 no tendrá una IP asignada, sino una configuración especial sin dirección IP.
 - El diagrama que se muestra en la figura #2 refleja la configuración necesaria a nivel de conectividad en ambos nodos:

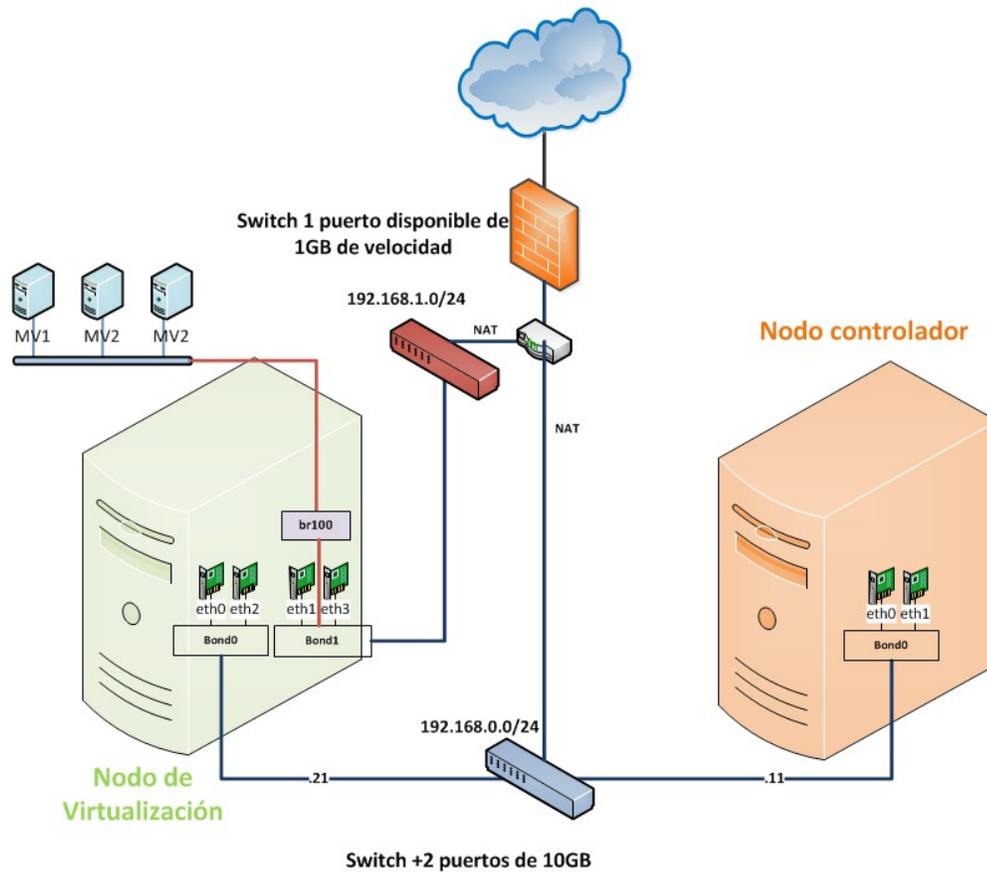


Figura #2. Diagrama de conectividad

- Configurar la resolución de nombres de manera local (/etc/hosts), cada uno de los nodos debe resolver los nombres de los otros nodos, así como las direcciones IP.
- Ambos nodos necesitan acceso a Internet durante el proceso de instalación de OpenStack, por lo que se debe configurar NAT para proveer acceso a Internet desde la red privada.
- Es necesario contar con acceso a root o a un usuario con privilegios de root para llevar a cabo la instalación.
- Todos los nodos deben tener NTP configurado y su hora y fecha deben ser la misma. En caso contrario, se deberá sincronizar el tiempo con algún servidor NTP de manera manual y utilizar el comando **ntpdate <ipdelServidorNTP>**.

5.2.1 Configuración de la red

A continuación, se presentan los detalles de la configuración de red de los equipos. Para más claridad observe el diagrama de la figura #2.

Nodo controlador:

a) Configurar la primera interfaz como interfaz de administración (red privada):

- Dirección IP: 192.168.0.11
- Máscara de red: 255.255.255.0
- Gateway: 192.168.0.1

b) Configurar la segunda interfaz como interfaz con acceso al exterior (NAT):

- Dirección IP: 192.168.1.11
- Máscara de red: 255.255.255.0
- Gateway: 192.168.1.1

c) Agregar los nombres de todos los nodos en /etc/hosts:

```
# vi /etc/hosts
192.168.0.11 controller
192.168.0.21 compute
```

Nodo de virtualización:

a) Configurar la primera interfaz como interfaz de administración (red privada):

- Dirección IP: 192.168.0.21
- Máscara de red: 255.255.255.0
- Gateway: 192.168.0.1

Nota: para nodos extras de virtualización, asignar IPs de 21 en adelante (21 a 30, por ejemplo).

b) Configurar la segunda interfaz pública sin dirección IP.

```
# vi /etc/network/interfaces

# The external network interface
auto bond1
iface bond1 inet manual
    up ip link set dev $IFACE up
    down ip link set dev $IFACE down
```

Nota: reemplace bond1 con el nombre de la interfaz de red según corresponda.

c) Agregar los nombres de todos los nodos en /etc/hosts:

```
# vi /etc/hosts

192.168.0.11 controller
192.168.0.21 compute
```

Probar la configuración de la red:

- Se recomienda probar que la conexión de red funciona correctamente. Para ello, pruebe hacer ping desde cada uno de los nodos a las direcciones IP de los otros nodos y a una dirección de Internet.

```

root@controller:~# ping openstack.org
PING openstack.org (192.237.193.83) 56(84) bytes of data.
64 bytes from 192.237.193.83: icmp_seq=1 ttl=50 time=97.4 ms
^C
--- openstack.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 97.409/97.409/97.409/0.000 ms
root@controller:~# ping compute
PING compute (192.168.0.21) 56(84) bytes of data.
64 bytes from compute (192.168.0.21): icmp_seq=1 ttl=64 time=0.385 ms
^C
--- compute ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.385/0.385/0.385/0.000 ms
root@controller:~# █

```

5.2.2 Configuración NTP

Nodo controlador

Por defecto, el nodo controlador sincroniza su tiempo con un pool de servidores públicos. Sin embargo, de manera opcional, se puede modificar la configuración para incluir algún servidor NTP local con el que ya cuenta el ambiente.

- a) Instale el servicio NTP:

```
# apt-get install ntp
```

- b) Modifique el archivo de configuración de NTP para incluir el nombre o dirección IP de su servidor NTP local:

```
# vi /etc/ntp.conf
<<Agregar la siguiente línea al final del archivo de configuración>>
server IP_Servidor_NTP iburst
```

Nota: reemplace IP_Servidor_NTP por la dirección IP o nombre del servidor NTP local.

- c) Busque las líneas restrict y elimine las opciones nopeer y noquery:

```
# vi /etc/ntp.conf
```

Antes:

```
restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery
```

Después:

```
restrict -4 default kod notrap nomodify
restrict -6 default kod notrap nomodify
```

d) Reinicie el servicio NTP para cargar la nueva configuración:

```
# service ntp restart
```

Nodo de virtualización

El resto de nodos de OpenStack apuntarán al servidor controlador para la sincronización del tiempo. Puede utilizar tanto el nombre como la dirección IP.

a) Instale el servicio NTP:

```
# apt-get install ntp
```

b) Modifique el archivo de configuración de NTP para incluir el nombre o dirección IP de su servidor NTP local:

```
# vi /etc/ntp.conf
<<Agregar la siguiente línea al final del archivo de configuración>>
server 192.168.0.11 iburst
```

c) Reinicie el servicio NTP para cargar la nueva configuración:

```
# service ntp restart
```

Utilizando el comando `ntpq -p`, compruebe que la configuración NTP funciona en todos los nodos:

```
# ntpq -p
```

```
root@compute:~# ntpq -p
      remote           refid      st t when poll reach   delay   offset  jitter
=====
+ns2.fratec.net  200.59.16.50      2 u   44   64   17   12.435  44.398  3.931
-c7.criticalcolo 128.227.205.3     2 u   43   64   17   89.428   1.820  2.035
+chato.crisp.cr  193.79.237.14    2 u   39   64   17   10.983  55.183  1.909
*ntp3.fratec.net .GPS.             1 u   37   64   17   10.689  38.698  1.864
controller       91.189.89.199    3 u    2   64    1    0.212   3.268  3.437
juniperberry.ca  193.79.237.14    2 u   36   64   17  159.725  33.523  0.642
root@compute:~#
```

6 Consideraciones de implementación

Para esta solución se utilizará Ubuntu server 14.04 LTS en cada uno de los equipos. Además, se usarán dos servidores en total.

6.1 Consideraciones de tamaño

Esta sección describe los detalles de tamaño o capacidad de equipo que se deben considerar para la implementación exitosa de esta solución.

OpenStack tiene los siguientes requerimientos mínimos para cada uno de los nodos que vamos a utilizar:

- **Nodo Controlador:** necesita al menos un procesador, 2 GB de memoria y 5 GB de almacenamiento.
- **Nodo de virtualización:** necesita al menos un procesador, 2 GB de memoria y 10 GB de almacenamiento.

Es importante utilizar componentes redundantes en equipos donde se necesite garantizar estabilidad y disponibilidad. A continuación, se describen las características del equipo recomendado en esta implementación:

Nodo controlador

En el nodo controlador se instalan los componentes más importantes de OpenStack. En seguida, se mencionan las características técnicas del equipo recomendado:

- 2 procesadores de 1.8 GHZ y 4 cores cada uno. Se utilizan dos procesadores para garantizar redundancia, ya que solamente se cuenta con un servidor en esta pequeña solución.
- 2 DIMM de memoria RAM de 4 u 8 GB.
- 2 discos duros internos de 300 GB de tamaño. Se utilizan 2 discos para garantizar redundancia de datos. En ellos se instalará el sistema operativo.

Nodo de virtualización

En este nodo se crean las instancias de máquinas virtuales, por eso es importante contar con gran cantidad de memoria y de procesadores/cores físicos que permitan la asignación de recursos a buena cantidad de máquinas virtuales. Los tamaños de estos dos recursos pueden ser ajustados de acuerdo con sus necesidades específicas.

A continuación, se especifican las características del equipo recomendado. Con esa cantidad de memoria y de CPU se pueden soportar hasta 48 máquinas virtuales de 4 CPU virtuales y 2 GB de memoria virtual, utilizando el ratio de 16:1 para los CPU y 1.5:1 para la memoria.

- 2 procesadores de 2 GHZ y 6 cores cada uno. Se utilizan dos procesadores para garantizar redundancia, ya que solamente se cuenta con un servidor en esta pequeña solución.
- 4 DIMMs de memoria RAM de 32 GB, o bien 8 de 16GB.
- 2 discos duros internos de 120 GB de tamaño. Se utilizan 2 discos para garantizar redundancia de datos. Este será el dispositivo en el que se instalará el sistema operativo.
- 2 discos duros internos de 1 TB cada uno. Se utilizan 2 discos para garantizar redundancia de datos. En estos dispositivos se crearán los volúmenes de datos que serán presentados a las instancias de máquinas virtuales para la creación de sistemas de ficheros requeridos por las aplicaciones o usuarios utilizando **Cinder**.

6.2 Consideraciones adicionales

Esta sección cubre otras consideraciones que pueden ser importantes para implementar exitosamente esta solución.

6.2.1 Tipo de CPU a utilizar en los equipos

El CPU del equipo a utilizar debe ser de 64 bits y debe soportar virtualización de instancias (VT-x de procesadores Intel o AMD-v para chips AMD).

6.2.2 Asignación de recursos (CPU y memoria)

OpenStack permite la sobre utilización de los CPU y memoria en los nodos de virtualización. Esta capacidad permite incrementar el número de instancias virtuales que corren a la vez en un nodo de virtualización; sin embargo, reduce su desempeño.

Se utilizan los siguientes ratios:

- Ratio de asignación de CPU: 16:1
- Ratio de asignación de RAM: 1.5:1

El ratio de asignación de CPU de 16:1 significa que al "scheduler" de OpenStack le corresponde asignar hasta 16 cores virtuales por cada core físico. Si un nodo físico tiene 12 cores, el "scheduler" de OpenStack ve 192 cores virtuales disponibles para ser utilizados por las máquinas virtuales. Si se asignan 4 cores virtuales a cada instancia, se tendrá una capacidad de 48 instancias virtuales en el nodo físico de 12 cores.

La fórmula para el número de instancias que se pueden crear en un nodo de virtualización es la siguiente:

$(OR * PC) / VC$, donde:

- OR es el número de cores virtuales por core físico.
- PC es el número de cores físicos.
- VC es el número de cores virtuales por instancia.

De manera similar, el ratio de asignación de RAM de 1.5:1 significa que el "scheduler" de OpenStack asigna instancias a un nodo físico, siempre que el total de la RAM asociada a esa instancia sea menos que 1,5 veces el tamaño de la RAM disponible en el nodo físico. Por ejemplo, si un nodo físico tiene 48 GB de RAM, el "scheduler" asigna instancias a ese nodo hasta que la suma de la RAM asociada con las instancias llegue a 72 GB. Por eso, si creamos máquinas virtuales de 8 GB de memoria, podríamos hospedar hasta 9 de esas instancias en ese nodo físico.

6.3 Estándares de configuración

En este documento encontrará una serie de instrucciones necesarias para la implementación de un IaaS basado en OpenStack.

Como parte de las instrucciones encontrará una serie de comandos Unix y OpenStack que deberán ser ejecutados en cada uno de los equipos donde se instalará OpenStack. Por eso, resulta importante aclarar lo siguiente:

- **\$ <comando>**

Los comandos que inicien con el símbolo \$ se pueden ejecutar utilizando cualquier usuario sin privilegios de root o directamente, como el usuario root en su defecto.

- **# <comando>**

Los comandos que inicien con el símbolo # se deben ejecutar, solamente, con el usuario root.

Además, es importante mencionar los siguientes detalles:

- El servidor en el cual se instalará el nodo controlador de OpenStack se llamará **controller**.
- El servidor en el cual se instalará el nodo de virtualización de OpenStack tendrá el siguiente nombre: **compute**.
- La red utilizada para comunicaciones internas será 192.168.0.0/28
- La red utilizada para comunicaciones externas y con acceso a Internet será 192.168.1.0/28.
- La dirección IP asignada al nodo controlador es 192.168.0.11. En caso de haber más nodos controladores, pueden recibir la IP .11 en adelante.
- La dirección IP asignada al nodo de virtualización es 192.168.0.21. Cualquier otro nodo de virtualización puede recibir IP de la .21 en adelante.
- La comunicación con el exterior se dará por medio de NATEO en el enrutador de la red donde se está implementando la solución. Dicha configuración no es parte de este proyecto, por lo que no encontrará detalles sobre cómo configurar switches ni enrutadores en este documento.
- El tenant y usuario con permisos de administración en OpenStack se llamarán **admin**.

- El tenant y usuario para pruebas y ejemplos documentados en esta solución será **demo**. Otro nombre de su preferencia puede ser utilizado durante la fase de instalación.

7 Implementación

7.1. Resumen

Esta sección describe los roles, procesos, herramientas, configuración de equipo y configuración de software necesarios para la implementación de todos los servicios que forman parte de esta solución.

Seguidamente, se muestra un diagrama de la arquitectura de esta solución. En ella se puede apreciar la interacción del equipo y los componentes.

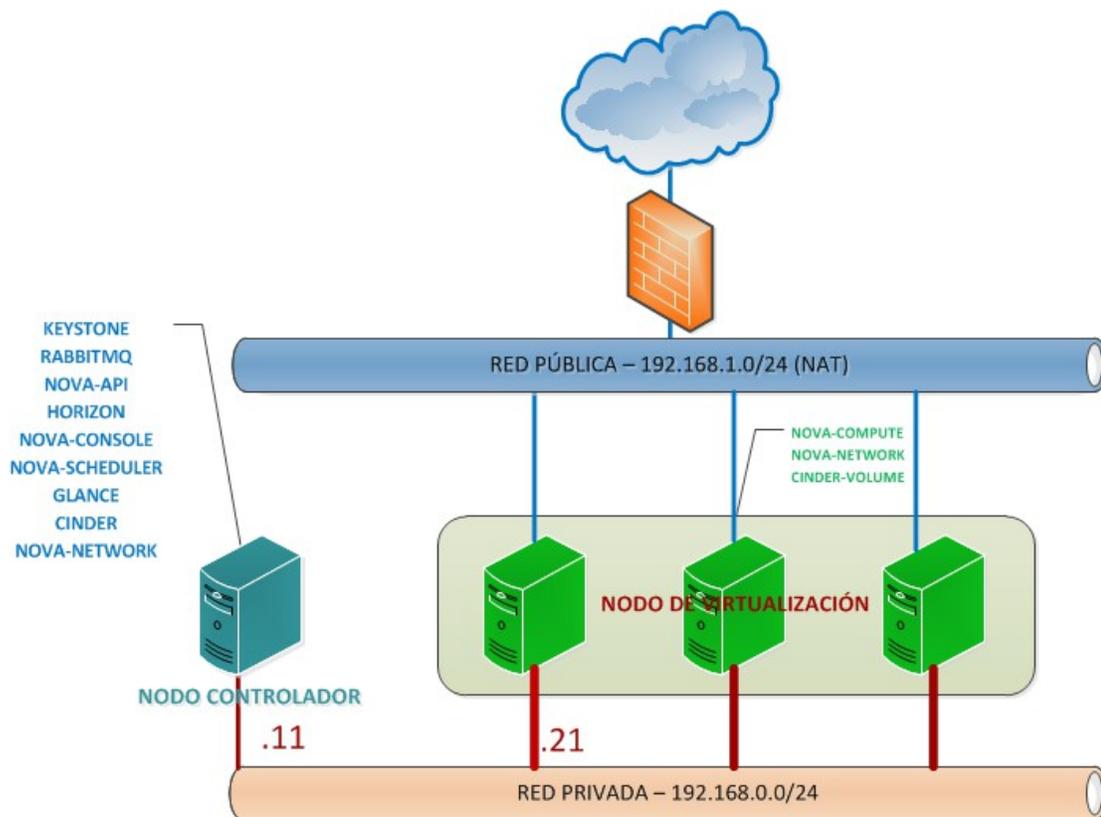


Figura #3. Arquitectura de la solución

7.2 Roles de implementación

La persona a cargo de esta implementación debe tener conocimientos básicos de administración de servidores GNU/Linux. Esta persona se hará cargo de la instalación de todos los componentes/servicios de OpenStack en cada uno de los nodos: nodo controlador y nodo de virtualización.

7.3 Diagrama de procesos

El siguiente diagrama muestra la manera como se relacionan los distintos servicios que forman parte de esta implementación.

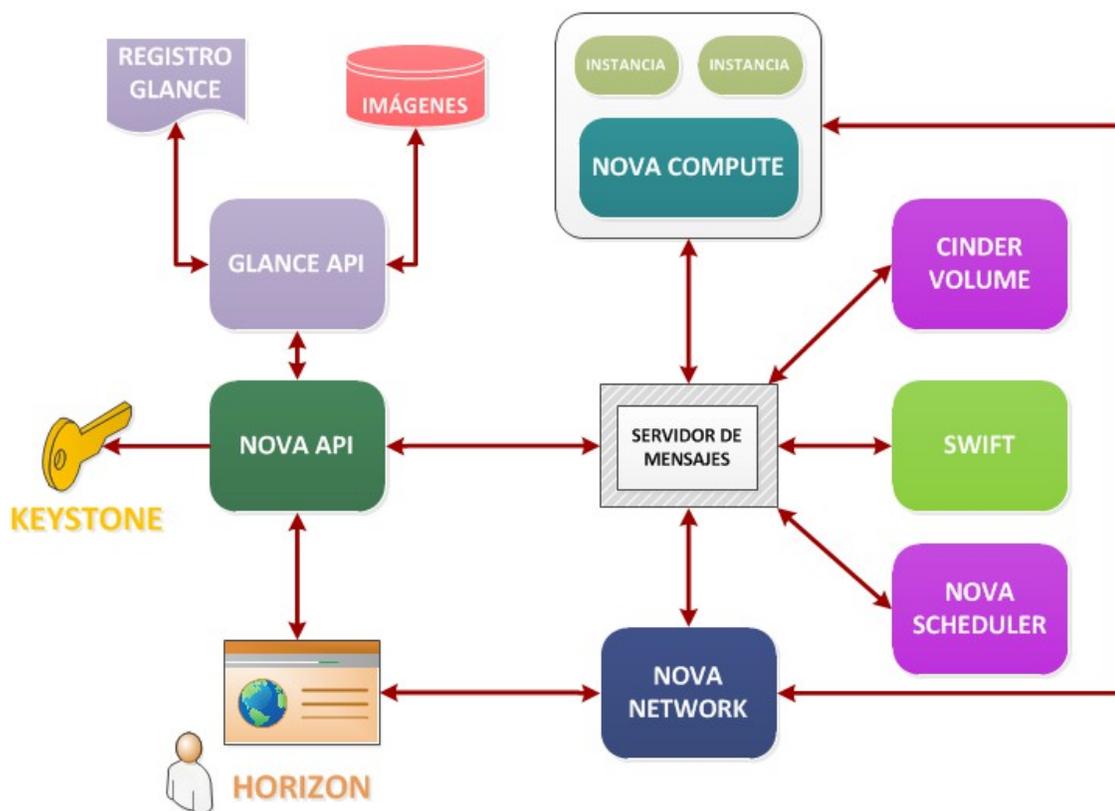


Figura #4. Diagrama de componentes de OpenStack

- Un usuario interactúa con la API de nova (directamente, a través del API o, indirectamente, a través de Horizon) para ejecutar/crear una instancia virtual.
- El servicio nova-API le pedirá que se autentique previamente con Keystone.

- Una vez autenticado le mostrará las imágenes disponibles en Glance.
- Cuando seleccione una imagen y las características para la instancia (Flavor), se enviará a nova-scheduler la petición.
- El servicio Nova-scheduler determinará en qué nodo se debe ejecutar la instancia virtual en caso de existir más de un nodo de virtualización.
- El servicio Nova-compute del nodo seleccionado se encargará de ejecutar la instancia sobre el hipervisor que disponga.
- El servicio Nova-network realizará las configuraciones necesarias en la red de la nueva instancia virtual.
- El servicio Cinder-volume se encargará de gestionar los volúmenes asociados a una instancia virtual.

7.4 Actividades de implementación

Esta sección describe las actividades necesarias para implementar esta solución, ya sea en un ambiente de pruebas o de producción.

La solución se desarrolla sobre dos nodos: nodo controlador y nodo de virtualización. Ambos se instalan utilizando Ubuntu server 14.04 LTS sobre equipos de 64 bits.

En el primer equipo se realiza la instalación del nodo controlador de OpenStack, que posee servicios de control como el de autenticación, el de imágenes con almacenamiento local, el de bases de datos para el registro de todos los detalles de la configuración, así como el Dashboard.

En el segundo equipo se instala el hipervisor y el servicio **compute** que se hará cargo de la creación de máquinas virtuales, así como el servicio de red de **nova**. En este segundo equipo se hospedarán las máquinas virtuales utilizando almacenamiento local.

El nodo controlador tendrá los siguientes servicios de OpenStack instalados:

- **Servicios de apoyo:**
 - ➔ Base de datos MySQL
 - ➔ Servidor RabbitMQ

- **Servicios de OpenStack:**
 - ➔ Servicio de identidad Keystone.
 - ➔ Servicio de imágenes Glance.
 - ➔ Servicio Nova Compute.
 - ➔ Servicio Nova Network.
 - ➔ Horizon, que es el Portal web para la gestión de OpenStack.
 - ➔ Servicio Cinder (anteriormente conocido como Nova-Volume) para la gestión de bloques de almacenamiento para las máquinas virtuales.

Los siguientes servicios de OpenStack se instalarán en el nodo de virtualización:

- Nova Compute
- Hipervisor Nova
- KVM
- Nova network, que es el servicio básico de redes de OpenStack.
- Cinder-volume (anteriormente conocido como nova-volume) que va a proporcionar volúmenes a las máquinas virtuales.

En el diagrama que se presenta a continuación puede apreciar la secuencia de tareas que será realizada durante la implementación de esta solución. Las tareas en amarillo son de preparación, en púrpura se encuentran las tareas a ser ejecutadas en el nodo controlador y, en verde, las del nodo de virtualización.

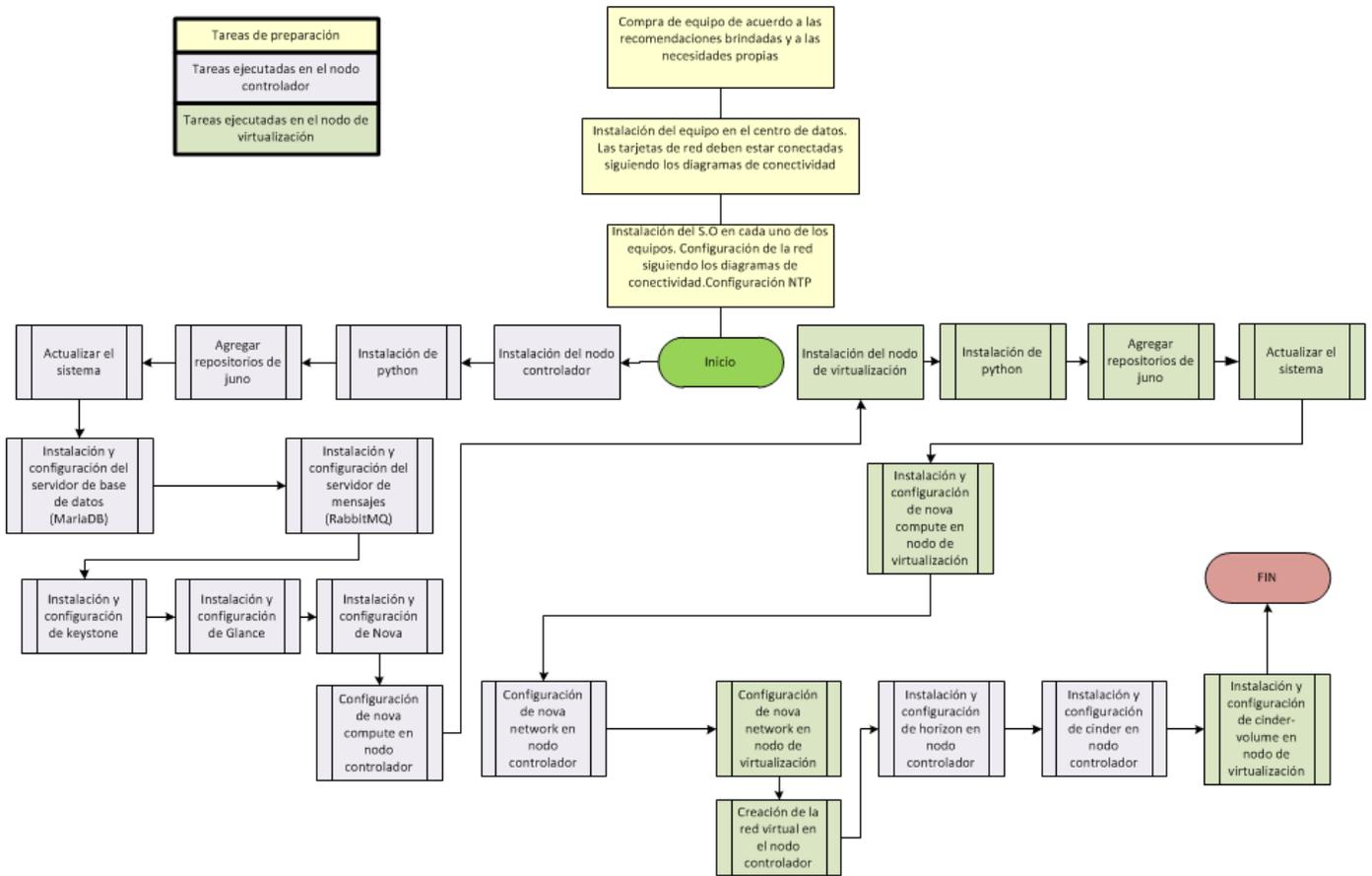


Figura #5. Flujo de implementación.

7.5 Instalación de prerrequisitos

El software de OpenStack está disponible en los repositorios de Ubuntu, sistema operativo que estamos utilizando. Los pasos siguientes los debe ejecutar en todos los nodos de OpenStack (controlador y de virtualización). Además, los comandos que se muestran a continuación se deben instalar desde root.

- a) Instale el paquete python-software-properties desde los repositorios de Ubuntu.

```
# apt-get install python-software-properties
```

```

root@controller:~# apt-get install python-software-properties
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  python-software-properties
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 19.6 kB of archives.
After this operation, 138 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu/ trusty-updates/universe python-software-properties all 0.92.37.2 [19.6 kB]
Fetched 19.6 kB in 0s (63.3 kB/s)
Selecting previously unselected package python-software-properties.
(Reading database ... 152037 files and directories currently installed.)
Preparing to unpack ../python-software-properties_0.92.37.2_all.deb ...
Unpacking python-software-properties (0.92.37.2) ...
Setting up python-software-properties (0.92.37.2) ...
root@controller:~# █

```

b) Agregue el repositorio de OpenStack Cloud-archive de Ubuntu.

```
# add-apt-repository cloud-archive:juno
```

```

root@controller:~# add-apt-repository cloud-archive:juno
Ubuntu Cloud Archive for OpenStack Juno
More info: https://wiki.ubuntu.com/ServerTeam/CloudArchive
Press [ENTER] to continue or ctrl-c to cancel adding it

Reading package lists...
Building dependency tree...
Reading state information...
The following NEW packages will be installed:
  ubuntu-cloud-keyring
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 5,086 B of archives.
After this operation, 34.8 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu/ trusty/universe ubuntu-cloud-keyring all 2012.08.14 [5,086 B]
Fetched 5,086 B in 0s (15.6 kB/s)
Selecting previously unselected package ubuntu-cloud-keyring.
(Reading database ... 152053 files and directories currently installed.)
Preparing to unpack ../ubuntu-cloud-keyring_2012.08.14_all.deb ...
Unpacking ubuntu-cloud-keyring (2012.08.14) ...
Setting up ubuntu-cloud-keyring (2012.08.14) ...
Importing ubuntu-cloud.archive.canonical.com keyring
OK
Processing ubuntu-cloud.archive.canonical.com removal keyring
OK
root@controller:~# █

```

c) Actualice todos los paquetes y la distribución en los tres nodos y reinicie.

```
# apt-get update && apt-get dist-upgrade
```

```
# shutdown -ry 0
```

7.6 Instalación del nodo controlador

El nodo controlador contará con los siguientes servicios de OpenStack instalados:

Servicios de apoyo:

- ➔ Base de datos MySQL
- ➔ Servidor RabbitMQ

Servicios de OpenStack:

- Servicio de identidad Keystone
- Servicio Nova Compute
- Horizon, que es el Portal web para la gestión de OpenStack.
- Servicio de imágenes Glance
- Servicio de bloques de almacenamiento Cinder

Todos los comandos que se muestran a continuación se deben instalar utilizando el acceso root.

7.6.1 Instalación del servidor de bases de datos

a) Instale los siguientes paquetes:

```
# apt-get install mariadb-server python-mysqldb
```

- i. Durante el proceso de instalación, el sistema le solicitará que le asigne una contraseña al usuario root de la base de datos. Escoja una contraseña fuerte y segura. Puede utilizar el siguiente comando para generar contraseñas aleatorias:

```
$ openssl rand -hex 10
```

b) Una vez instalada la base de datos, modifique el archivo de configuración `/etc/mysql/my.cnf` de la siguiente manera:

- i. Busque el bloque [mysqld] y asigne la IP privada del nodo controlador a bind-address.

```
# vi /etc/mysql/my.cnf
```

```
[mysqld]
#
# * Basic Settings
#
user                = mysql
pid-file            = /var/run/mysqld/mysqld.pid
socket              = /var/run/mysqld/mysqld.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
tmpdir              = /tmp
lc-messages-dir    = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address       = 127.0.0.1
bind-address        = 192.168.0.11
```

- ii. Además, en el mismo archivo, al final del bloque [mysqld], agregue los siguientes parámetros:

```
[mysqld]
...
default-storage-engine = innodb
innodb_file_per_table
collation-server = utf8_general_ci
init-connect = 'SET NAMES utf8'
character-set-server = utf8
```

```
default-storage-engine = innodb
innodb_file_per_table
collation-server = utf8_general_ci
init-connect = 'SET NAMES utf8'
character-set-server = utf8
```

- c) Reinicie el servicio de base de datos.

```
# service mysql restart
```

```

root@controller:~# service mysql restart
* Stopping MariaDB database server mysqld      [ OK ]
* Starting MariaDB database server mysqld      [ OK ]
* Checking for corrupt, not cleanly closed and upgrade needing tables.
root@controller:~# █

```

d) Ejecute el siguiente script para asegurar la base de datos.

```
# mysql_secure_installation
```

- i. Ejecute ENTER para iniciar el proceso de tuneo de seguridad.
- ii. Se le solicitará ingresar la contraseña de root de MariaDB que configuró durante la instalación. Digítela y presione la tecla ENTER.

```

Enter current password for root (enter for none):
OK, successfully used password, moving on...

```

- iii. A continuación, responda que no quiere cambiar la contraseña de root, pues ya lo ha hecho durante el proceso de instalación de la base de datos. Digite la letra "n" y presione la tecla ENTER para continuar.

```

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

```

- iv. Remueva el acceso a los usuarios anónimos, puesto que conformaría un agujero de seguridad. Digite "Y" y presione ENTER para continuar.

```

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

```

- v. Deshabilite el acceso remoto a root desde la base de datos. Digite "Y" y presione ENTER para continuar.

```

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

```

- vi. Elimine la base de datos "test" en caso de que exista. Digite "Y" y presione ENTER para continuar.

```
Remove test database and access to it? [Y/n] Y
- Dropping test database...
ERROR 1008 (HY000) at line 1: Can't drop database 'test'; database doesn't exist
... Failed! Not critical, keep moving...
- Removing privileges on test database...
... Success!
```

- vii. Para finalizar, recargue la tabla de privilegios para aplicar todos los cambios. Digite "Y" y presione ENTER para finalizar.

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

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
```

7.6.2 Instalación del servidor de mensajes

OpenStack utiliza un servicio de mensajería para coordinar las operaciones y el estado de la información entre los distintos servicios, el cual se ejecuta de manera típica en el nodo controlador. Además, OpenStack soporta varios servicios de mensajería como RabbitMQ, Qpid y ZeroMQ. En este caso vamos a utilizar RabbitMQ.

- a) Instale el servicio RabbitMQ.

```
# apt-get install rabbitmq-server
```

```
Setting up rabbitmq-server (3.2.4-1) ...
Adding group `rabbitmq' (GID 129) ...
Done.
Adding system user `rabbitmq' (UID 121) ...
Adding new user `rabbitmq' (UID 121) with group `rabbitmq' ...
Not creating home directory `/var/lib/rabbitmq'.
* Starting message broker rabbitmq-server [ OK ]
Processing triggers for libc-bin (2.19-0ubuntu6.3) ...
Processing triggers for ureadahead (0.100.0-16) ...
root@controller:~#
```

- b) Configuración de RabbitMQ

- i. Cambie la contraseña del usuario **guest** que crea RabbitMQ durante el proceso de instalación.

```
# rabbitmqctl change_password guest <nuevo_password>
```

Nota: Reemplace <nuevo_password> por una contraseña segura. Puede utilizar el siguiente comando para generar contraseñas aleatorias:

```
$ openssl rand -hex 10
```

```
root@controller:~# openssl rand -hex 10
5326e4ce4cb39266alae
root@controller:~# rabbitmqctl change_password guest 5326e4ce4cb39266alae
Changing password for user "guest" ...
...done.
```

7.6.3 Instalación del servicio de Identidad de OpenStack (Keystone)

El servicio de identidad de OpenStack se encarga de realizar un traqueo de los usuarios y sus permisos. Además, proporciona un catálogo con todos los servicios disponibles y sus API.

7.6.3.1 Configuración de prerequisites de Keystone

- a) Cree una base de datos:
 - i. Ingrese a la base de datos con el siguiente comando:

```
# mysql -u root -p<contraseñaDeRoot>
```

Nota: reemplace <contraseñaDeRoot> con la contraseña de root de la base de datos.

- ii. Cree la base de datos llamada **Keystone**:

```
CREATE DATABASE keystone;
```

- iii. Asigne los siguientes permisos a la base de datos **keystone**:

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'localhost' \
  IDENTIFIED BY '<KEYSTONE_DBPASS>';
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'% ' \
  IDENTIFIED BY '<KEYSTONE_DBPASS>';
```

Nota: reemplace <KEYSTONE_DBPASS> con una contraseña segura. Puede utilizar el siguiente comando para generar contraseñas aleatorias:

```
# openssl rand -hex 10
```

- iv. Salga de la base de datos.

```
exit
```

7.6.3.2 Instalación de los paquetes de Keystone

- a) Instale los paquetes Keystone y python-keystoneclient.

```
# apt-get install keystone python-keystoneclient
```

- b) Modifique el archivo de configuración de Keystone `/etc/keystone/keystone.conf` de la siguiente manera:

```
# vi /etc/keystone/keystone.conf
```

- i. Genere un token de administración para Keystone.

```
# openssl rand -hex 10
```

```
root@controller:~# openssl rand -hex 10
080d080584cc2a94b25f
root@controller:~# █
```

- ii. Busque la sección `[DEFAULT]` y coloque el token de administración generado en el paso anterior.

```
[DEFAULT]
#
# Options defined in keystone
#
# A "shared secret" that can be used to bootstrap Keystone.
# This "token" does not represent a user, and carries no
# explicit authorization. To disable in production (highly
# recommended), remove AdminTokenAuthMiddleware from your
# paste application pipelines (for example, in keystone-
# paste.ini). (string value)
admin_token=080d080584cc2a94b25f█
```

- iii. Busque la sección `[database]` y configure la conexión a la base de datos.

```
[database]
```

```
...
```

```
connection = mysql://keystone:<KEYSTONE_DBPASS>@controller/ keystone
```

Nota: Reemplace `<KEYSTONE_DBPASS>` con la contraseña asignada a la base de datos Keystone durante su creación.

- iv. Busque la sección `[token]` y configure el token UUID y el driver SQL.

```
[token]
```

```
...
```

```
provider = keystone.token.providers.uuid.Provider
driver = keystone.token.persistence.backends.sql.Token
```

```
#provider=<None>
provider=keystone.token.providers.uuid.Provider
# Token persistence backend driver. (string value)
#driver=keystone.token.persistence.backends.sql.Token
driver=keystone.token.persistence.backends.sql.Token
```

- v. Devuélvase a la sección [DEFAULT] y habilite la opción **verbose** poniéndola en **true**:

```
[DEFAULT]
...
verbose = True
```

- vi. Guarde los cambios hechos al archivo de configuración.

- c) Llene la base de datos del servicio de identidad ejecutando el siguiente comando:

```
# su -s /bin/sh -c "Keystone-manage db_sync" Keystone
```

Nota: Al finalizar de manera exitosa, verá varios registros, los cuales finalizarán en **done**, como los que se muestran en seguida:

```
2014-11-13 21:09:25.990 10582 INFO migrate.versioning.api [-] 52 -> 53...
2014-11-13 21:09:27.571 10582 INFO migrate.versioning.api [-] done
2014-11-13 21:09:27.571 10582 INFO migrate.versioning.api [-] 53 -> 54...
2014-11-13 21:09:27.735 10582 INFO migrate.versioning.api [-] done
2014-11-13 21:09:27.736 10582 INFO migrate.versioning.api [-] 54 -> 55...
2014-11-13 21:09:27.991 10582 INFO migrate.versioning.api [-] done
2014-11-13 21:09:28.031 10582 INFO migrate.versioning.api [-] 0 -> 1...
2014-11-13 21:09:28.328 10582 INFO migrate.versioning.api [-] done
2014-11-13 21:09:28.329 10582 INFO migrate.versioning.api [-] 1 -> 2...
2014-11-13 21:09:28.749 10582 INFO migrate.versioning.api [-] done
```

- d) Reinicie el servicio Keystone para aplicar los cambios realizados.

```
# service keystone restart
```

```
root@controller:~# service keystone restart
keystone stop/waiting
keystone start/running, process 10626
root@controller:~#
```

- e) Elimine el archivo de base de datos SQLite, que no es necesario en esta instalación.

```
# rm -f /var/lib/keystone/keystone.db
```

- f) Debe agregar en el crontab del usuario Keystone un script que limpie los tokens expirados en la base de datos para evitar que se haga muy grande y genere problemas de desempeño. Ejecute el siguiente comando como root:

```
# (crontab -l -u keystone 2>&1 | grep -q token_flush) || \
echo '@hourly /usr/bin/keystone-manage token_flush >/var/log/keystone/keystone-
tokenflush.log 2>&1' \
>> /var/spool/cron/crontabs/keystone
```

Nota: la entrada en el crontab de Keystone quedará de la siguiente manera:

```
root@controller:~# crontab -l -u keystone
@hourly /usr/bin/keystone-manage token_flush >/var/log/keystone/keystone-tokenfl
ush.log 2>&1
root@controller:~#
```

7.6.3.3 Configuración de los usuarios de Keystone

Después de instalar Keystone, es necesario crear proyectos (tenants), usuarios y roles en su ambiente. Siga las instrucciones que se especifican a continuación. Tome en cuenta que debe utilizar el token administrativo que fue generado durante el proceso de configuración de Keystone.

- a) Configure el token administrativo en una variable de entorno de manera temporal como se muestra seguidamente. Cabe recordar que utilizaremos el mismo token administrativo generado anteriormente, que se encuentra en el archivo de configuración de Keystone.

```
$ export OS_SERVICE_TOKEN=$(cat /etc/keystone/keystone.conf | grep -i
admin_token|cut -d = -f2)
```

- b) También, configure el "endpoint" del servicio.

```
$ export OS_SERVICE_ENDPOINT=http://controller:35357/v2.0
```

Nota: reemplace **controller** con el nombre o dirección IP del nodo controlador. En nuestro caso, su nombre es controller.

- c) Proceda a crear los tenants, usuarios y roles requeridos por OpenStack, de la siguiente manera:
- i. Cree el tenant administrativo (**admin**), el tenant **demo** y el tenant llamado **service**, que será utilizado por OpenStack para interactuar con otros servicios de

OpenStack. Se creará un usuario para cada uno de los servicios que serán instalados. Utilice los siguientes comandos:

```
# keystone tenant-create --name admin --description "Tenant Administrativo"

# keystone tenant-create --name demo --description "Tenant Demo"

# keystone tenant-create --name service --description "Tenant Service"
```

```
root@controller:~# keystone tenant-create --name admin --description "Tenant Administrativo"
+-----+
| Property | Value |
+-----+
| description | Tenant Administrativo |
| enabled | True |
| id | 5a0d3c1c49cb4e67aada421c3b26539f |
| name | admin |
+-----+
root@controller:~# █
```

ii. Ahora cree los usuarios **admin** y **demo** en Keystone.

```
# keystone user-create --name admin --pass <ContraseñaAdmin> --email <DirecciónE-mail>

# keystone user-create --name demo --pass <ContraseñaDemo> --email <DirecciónE-mail>
```

Nota: reemplace **<ContraseñaAdmin>** ,**<ContraseñaDemo>** y **<DirecciónE-mail>** por una contraseña segura y una dirección de correo que quiera utilizar para estos efectos. Recuerde que puede generar contraseñas aleatorias utilizando el siguiente comando:

```
# openssl rand -hex 10
```

```

root@controller:~# keystone user-create --name admin --pass 123456 --email jenni
.jenni.tec@gmail.com
+-----+-----+
| Property |          Value          |
+-----+-----+
| email    | jenni.tec@gmail.com    |
| enabled  | True                   |
| id       | e8a6038dd3214ebc977aafa0359a65d9 |
| name     | admin                  |
| username | admin                  |
+-----+-----+
root@controller:~# █

```

- iii. Cree los roles administrativos con nombre **admin** y **_member_**.

```

# keystone role-create --name admin
# keystone role-create --name _member_

```

```

root@controller:~# keystone role-create --name admin
+-----+-----+
| Property |          Value          |
+-----+-----+
| id       | 5fb58e43800e4580a1b15d57ff3b19cb |
| name     | admin                  |
+-----+-----+
root@controller:~# █

```

```

root@controller:~# keystone role-create --name _member_
+-----+-----+
| Property |          Value          |
+-----+-----+
| id       | 5545e0a7481b44a390931a2981571e7f |
| name     | _member_              |
+-----+-----+
root@controller:~# █

```

- iv. Agregue el tenant administrativo y el usuario **admin** al rol **admin** creado en el paso anterior. Además, agregue el usuario **demo** y el tenant **demo** al rol **_member_**:

```

# keystone user-role-add --tenant admin --user admin --role admin
# keystone user-role-add --tenant demo --user demo --role _member_

```

Nota: este comando no producirá ninguna salida.

- v. En caso de verse en la necesidad de crear más tenants y usuarios o roles, puede seguir el procedimiento descrito en el punto **c**.
- d) Cree el "API endpoint" y la entidad de servicio. La entidad de servicio es utilizada para manejar un catálogo de servicios en OpenStack. Los demás servicios utilizan este catálogo para ubicar otros servicios en el ambiente. La entidad de servicio también maneja un catálogo de "API endpoints" que es asociado con cada uno de los servicios en

su ambiente OpenStack. Los servicios de OpenStack utilizan este catálogo para saber cómo comunicarse con otro servicio en el ambiente.

- i. En este punto es necesario que las variables de entorno que fueron configuradas en el paso **a** sigan funcionando. En caso de haber cerrado su sesión, configúrelas nuevamente.
- ii. Cree la entidad de servicio de Keystone utilizando el siguiente comando:

```
# keystone service-create --name keystone --type identity --description "OpenStack Identity"
```

```
root@controller:~# keystone service-create --name keystone --type identity --description "OpenStack Identity"
+-----+-----+
| Property | Value |
+-----+-----+
| description | OpenStack Identity |
| enabled | True |
| id | 3a1f73b48de44b2391b09aa978c76f65 |
| name | keystone |
| type | identity |
+-----+-----+
root@controller:~#
```

- iii. Cree el "API endpoint" para el servicio Keystone. Utilice el comando que se indica más adelante. El ID del servicio que utilizará corresponde con del servicio creado en el punto anterior.

```
# keystone endpoint-create \
--service-id $(keystone service-list | awk '/ identity / {print $2}') \
--publicurl http://controller:5000/v2.0 \
--internalurl http://controller:5000/v2.0 \
--adminurl http://controller:35357/v2.0 \
--region regionOne
```

Nota: reemplace **controller** con el nombre o dirección IP del nodo controlador. En nuestro caso, su nombre es controller.

```

root@controller:~# keystone endpoint-create \
> --service-id $(keystone service-list | awk '/ identity / {print $2}') \
> --publicurl http://controller:5000/v2.0 \
> --internalurl http://controller:5000/v2.0 \
> --adminurl http://controller:35357/v2.0 \
> --region regionOne
+-----+
| Property | Value |
+-----+
| adminurl | http://controller:35357/v2.0 |
| id       | 7498531ce5fb40899da6099118a80665 |
| internalurl | http://controller:5000/v2.0 |
| publicurl | http://controller:5000/v2.0 |
| region   | regionOne |
| service_id | 3a1f73b48de44b2391b09aa978c76f65 |
+-----+
root@controller:~#

```

- iv. A cada servicio que sea instalado en OpenStack le debe agregar este tipo de información a la entidad de Keystone.

7.6.3.4 Creación de los scripts (OpenRC) de ambiente de OpenStack

Estos scripts se utilizan para facilitar la interacción con el servicio de identidad de OpenStack. Se deben customizar con los detalles de los tenants y usuarios creados anteriormente para evitar tener que configurar todo el tiempo las variables de entorno o colocar los datos de autenticación para utilizar el servicio de identidad de manera manual.

- a) Cree dos scripts con los datos de los tenants admin y demo. Ambos scripts se deben crear bajo el directorio home del usuario Keystone (**/var/lib/Keystone/**).
 - i. Cree el script para cargar las variables de ambiente del tenant **admin**.

```
# vi /var/lib/keystone/admin-openrc.sh
```

Agregue las siguientes líneas en el script:

```

export OS_TENANT_NAME=admin
export OS_USERNAME=admin
export OS_PASSWORD=<ContraseñaAdmin>
export OS_AUTH_URL=http://controller:35357/v2.0

```

- ii. Cree el script para cargar las variables de ambiente del tenant **demo**.

```
# vi /var/lib/keystone/demo-openrc.sh
```

Agregue las siguientes líneas en el script:

```
export OS_TENANT_NAME=demo
export OS_USERNAME=demo
export OS_PASSWORD=<ContraseñaDemo>
export OS_AUTH_URL=http://controller:5000/v2.0
```

Nota: reemplace **<ContraseñaAdmin>** y **<ContraseñaDemo>** con la contraseña asignada al usuario admin creado en la sección anterior. También, reemplace **controller** con el nombre o dirección IP del nodo controlador, que en nuestro caso es controller.

- b) Para ejecutar el cliente de Keystone como determinado tenant y usuario, solo necesitará cargar las variables de entorno utilizando los scripts como se muestra a continuación.
 - i. Cargar las variables del tenant admin:

```
# source /var/lib/keystone/admin-openrc.sh
```

- ii. Cargar las variables del tenant demo:

```
# source /var/lib/keystone/demo-openrc.sh
```

- c) Puede verificar que las variables de entorno fueron cargadas en su sesión al utilizarlos siguientes comandos (deberá ver los valores correspondientes a cada tenant):

```
# echo $OS_TENANT_NAME
# echo $OS_USERNAME
# echo $OS_PASSWORD
# echo $OS_AUTH_URL
# keystone tenant-list
```

```

root@controller:~# source /var/lib/keystone/demo-openrc.sh
root@controller:~# echo $OS_USERNAME
demo
root@controller:~# echo $OS_TENANT_NAME
demo
root@controller:~# echo $OS_AUTH_URL
http://controller:35357/v2.0
root@controller:~# source /var/lib/keystone/admin-openrc.sh
root@controller:~# echo $OS_USERNAME
admin
root@controller:~# echo $OS_TENANT_NAME
admin
root@controller:~# echo $OS_AUTH_URL
http://controller:35357/v2.0
root@controller:~# █

```

7.6.4 Instalación del servicio de Imágenes de OpenStack (Glance)

El servicio de imágenes de OpenStack permite a los usuarios descubrir, agregar y obtener imágenes de las máquinas virtuales. Posee un API llamado REST que brinda la posibilidad de consultar los meta datos de las imágenes de las máquinas virtuales y obtener una imagen a partir de ellos. Las imágenes pueden ser almacenadas en file systems o bloques de almacenamiento y ser puestas a disposición del usuario a través del servicio Glance.

7.6.4.1 Configuración de prerequisites de Glance

- a) Cree el file system en el que almacenará las imágenes de Glance. El tamaño depende de la cantidad de imágenes que se planea almacenar. Tome en cuenta que durante el proceso de instalación de cada uno de los nodos se configuró LVM como parte del proceso de preparación. En LVM, debe haber suficiente espacio para crear este volumen. Vamos a utilizar 200 GB de almacenamiento.
 - i. Usando LVM, cree el volumen lv_glance en el VG creado durante el proceso de instalación. En este caso, el nombre del VG es vg00.
 - Cree el volumen lv_glance con un tamaño de 200 GB (puede ser creado con un tamaño menor o mayor, de acuerdo con la disponibilidad de espacio en su nodo controlador).

```
# lvcreate -L 204800 -n lv_glance /dev/vg00
```

Nota: reemplace /dev/vg00 por el nombre del VG de LVM creado en su nodo controlador.

- Cree el file system usando ext4.

```
mkfs.ext4 /dev/vg00/lv_glance
```

- Cree el punto de montaje.

```
# mkdir -p /var/lib/glance/images
```

- Agregue el file system en /etc/fstab.

```
# vi /etc/fstab
<<Agregue la siguiente línea al final del archivo>>
/dev/mapper/vg00-lv_glance /var/lib/glance/images ext4 defaults 0 2
```

- Monte el file system.

```
# mount /var/lib/glance/images
```

- Utilizando df -k, compruebe que el file system está montado.

```
# df -k /var/lib/glance/images
```

b) Cree la base de datos y los credenciales en el servicio de identidad (Keystone).

- i. Ingrese a la base de datos utilizando el usuario root.

```
# mysql -u root -p<ContraseñaBD>
```

Nota: reemplace **<ContraseñaBD>** con la contraseña de root de la base de datos.

- ii. Una vez conectado a la base de datos, cree la base de datos **Glance** y asígnele los permisos correspondientes utilizando los siguientes comandos:

```
CREATE DATABASE glance;
GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'localhost' \
  IDENTIFIED BY '<ContraseñaGlanceDB>';
GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'%' \
```

```
IDENTIFIED BY '<ContraseñaGlanceDB>';
```

Nota: Reemplace **<ContraseñaGlanceDB>** con la una contraseña segura.

```
MariaDB [(none)]> CREATE DATABASE glance;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> █
```

iii. Salga de la base de datos escribiendo **exit**.

c) Cargue las variables de entorno del tenant y usuario admin en Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

d) Utilizando Keystone, proceda con la creación del usuario Glance, asocie dicho usuario al tenant service y al role admin, cree el servicio glance y los "endpoint" de este servicio utilizando los comandos que se muestran a continuación:

```
# keystone user-create --name glance --pass <ContraseñaGlance>
# keystone user-role-add --user glance --tenant service --role admin
# keystone service-create --name glance --type image \
  --description "Servicio de Imagenes de OpenStack"

# keystone endpoint-create \
  --service-id $(keystone service-list | awk '/ image / {print $2}') \
  --publicurl http://controller:9292 \
  --internalurl http://controller:9292 \
  --adminurl http://controller:9292 \
  --region regionOne
```

Nota: Reemplace **<ContraseñaGlance>** con la una contraseña segura y **controller**, con el nombre o dirección IP del nodo controlador.

7.6.4.2 Instalación y configuración de Glance

a) Instale los siguientes paquetes:

```
# apt-get install glance python-glanceclient
```

b) Modifique el archivo de configuración de glance /etc/glance/glance-api.conf siguiendo las siguientes instrucciones:

```
# vi /etc/glance/glance-api.conf
```

- i. Busque la sección [database] y configure el acceso a la base de datos de la siguiente manera:

```
[database]
...
connection = mysql://glance:<ContraseñaGlanceDB>@controller/glance
```

Nota: reemplace **<ContraseñaGlanceDB>** con la contraseña de la base de datos de Glance.

- ii. En las secciones [keystone_authtoken] y [paste_deploy], configure el acceso al servicio de identidad:

```
[keystone_authtoken]
...
auth_uri = http://<controller>:5000/v2.0
identity_uri = http://<controller>:35357
admin_tenant_name = service
admin_user = glance
admin_password = <ContraseñaGlance>

[paste_deploy]
...
flavor = keystone
```

Nota: Reemplace **<ContraseñaGlance>** con la contraseña que le asignó al usuario Glance en Keystone y reemplace **controller** por el nombre o dirección IP del nodo controlador. En caso de que haya alguna variable `auth_host`, `auth_port`, `auth_protocol`, coméntela. La opción `identity_uri` reemplaza esas otras variables de configuración.

- iii. En la sección [glance_store], configure el file system o directorio en el que almacenará las imágenes (/var/lib/glance/images), el cual fue creado en la fase de preparación de este servicio.

```
[glance_store]
...
default_store = file
filesystem_store_datadir = /var/lib/glance/images/
```

- iv. Habilite la opción **verbose** en la sección [DEFAULT]. Verbose es útil para pruebas. En caso de utilizarla en producción no es necesario que esté activa.

```
[DEFAULT]
...
verbose = True
```

- v. Una vez hechas las modificaciones en ese archivo, cambie el archivo de configuración /etc/glance/glance-registry.conf de la misma manera. Siga los pasos de "i" a "iv" antes descritos, pero **excluya el paso "iii"**.

```
# vi /etc/glance/glance-registry.conf
```

- c) Llene la base de datos Glance ejecutando el siguiente comando:

```
# su -s /bin/sh -c "glance-manage db_sync" glance
```

- Nota:** verá una serie de registros desplegarse en la terminal, los cuales finalizan en **done**, como se muestra en seguida.

```
2014-11-14 12:47:35.883 23497 INFO glance.db.sqlalchemy.migrate_repo.schema [-]
creating table metadef_namespaces
2014-11-14 12:47:36.168 23497 INFO glance.db.sqlalchemy.migrate_repo.schema [-]
creating table metadef_objects
2014-11-14 12:47:36.465 23497 INFO glance.db.sqlalchemy.migrate_repo.schema [-]
creating table metadef_properties
2014-11-14 12:47:36.761 23497 INFO glance.db.sqlalchemy.migrate_repo.schema [-]
creating table metadef_resource_types
2014-11-14 12:47:37.058 23497 INFO glance.db.sqlalchemy.migrate_repo.schema [-]
creating table metadef_namespace_resource_types
2014-11-14 12:47:37.488 23497 INFO migrate.versioning.api [-] done
2014-11-14 12:47:37.488 23497 INFO migrate.versioning.api [-] 35 -> 36...
2014-11-14 12:47:37.975 23497 INFO migrate.versioning.api [-] done
root@centos11:~#
```

- d) Reinicie el servicio de imágenes.

```
# service glance-registry restart
```

```
# service glance-api restart
```

```
root@controller:~# service glance-registry restart
glance-registry stop/waiting
glance-registry start/running, process 23544
root@controller:~# service glance-api restart
glance-api stop/waiting
glance-api start/running, process 23569
root@controller:~# █
```

e) De la forma como se explica seguidamente, verifique que la instalación se ha llevado a cabo de manera adecuada.

- i. Descargue una imagen para probar Glance. En este caso, se utilizará una de CirrOS. En el siguiente enlace podrá encontrar más imágenes Cloud para utilizar: http://docs.openstack.org/image-guide/content/ch_obtaining_images.html

```
# mkdir /tmp/images
# cd /tmp/images
# wget http://cdn.download.cirros-cloud.net/0.3.3/cirros-0.3.3-x86_64-disk.img
```

- ii. Cargue las variables de entorno del tenant y usuario admin de Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

- iii. Suba la imagen descargada a Glance utilizando el siguiente comando:

```
# glance image-create --name "cirros-0.3.3-x86_64" --file cirros-0.3.3-x86_64-disk.img \
--disk-format qcow2 --container-format bare --is-public True --progress
```

```

root@controller:/tmp/images# glance image-create --name "cirros-0.3.3-x86_64" --
file cirros-0.3.3-x86_64-disk.img \
> --disk-format qcow2 --container-format bare --is-public True --progress
[=====>] 100%
+-----+
| Property | Value |
+-----+
| checksum | 133eae9fb1c98f45894a4e60d8736619 |
| container_format | bare |
| created_at | 2014-11-14T19:49:25 |
| deleted | False |
| deleted_at | None |
| disk_format | qcow2 |
| id | 57b9849d-2ec1-4f0e-9094-5f1949a84869 |
| is_public | True |
| min_disk | 0 |
| min_ram | 0 |
| name | cirros-0.3.3-x86_64 |
| owner | 5a0d3c1c49cb4e67aada421c3b26539f |
| protected | False |
| size | 13200896 |
| status | active |
| updated_at | 2014-11-14T19:49:25 |
| virtual_size | None |
+-----+
root@controller:/tmp/images#

```

iv. Para confirmar que la imagen ha sido subida a Glance ejecute el siguiente comando:

```
# glance image-list
```

```

root@controller:/tmp/images# glance image-list
+-----+
| ID | Name | Disk Format | Co
+-----+
| 57b9849d-2ec1-4f0e-9094-5f1949a84869 | cirros-0.3.3-x86_64 | qcow2 | ba
+-----+
| 13200896 | active |
+-----+
root@controller:/tmp/images#

```

v. Elimine el directorio temporal creado para esta prueba.

```
# rm -r /tmp/images
```

7.6.5 Instalación del servicio Compute de OpenStack (Nova)

Este servicio hospeda y gestiona sistemas de la nube, es el componente más importante de OpenStack.

Nova interactúa con el servicio de identidad de OpenStack, al igual que con el servicio de imágenes y con el dashboard (Horizon). El acceso a las imágenes está limitado por proyectos (tenants) y usuarios. Las cuotas se delimitan por proyecto.

7.6.5.1 Configuración de prerequisites de Nova

- a) Antes de iniciar el proceso de instalación, se debe crear la base de datos del servicio Compute y agregarlo al catálogo de servicios de Keystone.
 - i. Ingrese a la base de datos utilizando el usuario root.

```
# mysql -u root -p<ContraseñaBD>
```

Nota: reemplace <ContraseñaBD> con la contraseña de root de la base de datos.

- ii. Una vez conectado a la base de datos, cree la base de datos **Nova** y asígnele los permisos correspondientes utilizando los siguientes comandos:

```
CREATE DATABASE nova;
GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'localhost' \
  IDENTIFIED BY '<ContraseñaNovaDB>';
GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'%' \
  IDENTIFIED BY '<ContraseñaNovaDB>';
```

Nota: Reemplace <ContraseñaNovaDB> con la una contraseña segura.

- iii. Salga de la base de datos escribiendo **exit**.
 - b) Cargue las variables de entorno del tenant y usuario admin en Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Utilizando Keystone, cree el usuario Nova. Asocie dicho usuario al tenant service y al role admin. Cree el servicio nova y el "endpoint" de este servicio usando los comandos que se encuentran a continuación:

```
# keystone user-create --name nova --pass <ContraseñaNova>
```

```
# keystone user-role-add --user nova --tenant service --role admin
# keystone service-create --name nova --type compute \
--description "Servicio Compute de OpenStack"
# keystone endpoint-create \
--service-id $( keystone service-list | awk '/ compute / {print $2}') \
--publicurl http://controller:8774/v2/%\tenant_id)s \
--internalurl http://controller:8774/v2/%\tenant_id)s \
--adminurl http://controller:8774/v2/%\tenant_id)s \
--region regionOne
```

Nota: Reemplace **<ContraseñaNova>** con la una contraseña segura y **controller**, con el nombre o dirección IP del nodo controlador.

7.6.5.2 Instalación y configuración de Nova-compute

a) Instale los siguientes paquetes:

```
# apt-get install nova-api nova-cert nova-conductor nova-consoleauth \
nova-novncproxy nova-scheduler python-novaclient
```

b) Modifique el archivo de configuración de nova `/etc/nova/nova.conf`. Siga estas instrucciones:

```
# vi /etc/nova/nova.conf
```

i. Busque la sección `[database]` y configure el acceso a la base de datos se la siguiente manera:

```
[database]
...
connection = mysql://nova:<ContraseñaNovaDB>@controller/nova
```

Nota: Reemplace **<ContraseñaNovaDB>** con la contraseña que le asignó a la base de datos de nova.

ii. En la sección `[DEFAULT]`, configure el acceso a RabbitMQ.

```
[DEFAULT]
```

```
...
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = <ContraseñaRabbitMQ>
```

Nota: Reemplace **<ContraseñaRabbitMQ>** con la contraseña del usuario **guest** que le había asignado durante la fase de configuración de RabbitMQ en la sección Configuración de RabbitMQ (Sección 8.1.5.2 punto b).

- iii. En las secciones [DEFAULT] y [keystone_authtoken], configure el acceso a Keystone.

```
[DEFAULT]
...
auth_strategy = keystone

[keystone_authtoken]
...
auth_uri = http://<controller>:5000/v2.0
identity_uri = http://<controller>:35357
admin_tenant_name = service
admin_user = nova
admin_password = <ContraseñaNova>
```

Nota: Reemplace **<ContraseñaNova>** con la contraseña que le asignó al usuario Nova en Keystone y, **<controller>**, por el nombre o dirección IP del nodo controlador. En caso de que haya alguna variable `auth_host`, `auth_port`, `auth_protocol`, coméntela. La opción `identity_uri` es el reemplazo de esas otras variables de configuración.

- iv. En la sección [DEFAULT] configure la variable ***my_ip***, ***vnserver_listen*** y ***vnserver_proxyclient_address*** para que utilicen la IP privada del nodo controlador.

```
[DEFAULT]
...
```

```
my_ip = 192.168.0.11
vncserver_listen = 192.168.0.11
vncserver_proxyclient_address = 192.168.0.11
```

Nota: reemplace **192.168.0.11** con la dirección IP correspondiente que se asignó en la subnet privada del nodo controlador.

- v. En la sección [glance] configure la ubicación de Glance.

```
[glance]
...
host = controller
```

Nota: reemplace **controller** por el nombre o dirección IP del nodo controlador.

- vi. Habilite la opción verbose en la sección [DEFAULT]. Verbose es útil para pruebas y, en caso de utilizarla en producción, no es necesario que esté activa.

```
[DEFAULT]
...
verbose = True
```

- c) Llene la base de datos Nova ejecutando el siguiente comando:

```
# su -s /bin/sh -c "nova-manage db sync" nova
```

Nota: verá una serie de registros desplegarse en la terminal, que finalizará en **done**, como se muestra seguidamente.

```
2014-11-14 15:10:47.179 26565 INFO 248_add_expire_reservations_index [-] Skippe
adding reservations_deleted_expire_idx because an equivalent index already exi
ts.
2014-11-14 15:10:47.192 26565 INFO migrate.versioning.api [-] done
2014-11-14 15:10:47.193 26565 INFO migrate.versioning.api [-] 248 -> 249...
2014-11-14 15:10:47.291 26565 INFO migrate.versioning.api [-] done
2014-11-14 15:10:47.292 26565 INFO migrate.versioning.api [-] 249 -> 250...
2014-11-14 15:10:47.415 26565 INFO migrate.versioning.api [-] done
2014-11-14 15:10:47.416 26565 INFO migrate.versioning.api [-] 250 -> 251...
2014-11-14 15:10:48.000 26565 INFO migrate.versioning.api [-] done
2014-11-14 15:10:48.000 26565 INFO migrate.versioning.api [-] 251 -> 252...
2014-11-14 15:10:49.351 26565 INFO migrate.versioning.api [-] done
2014-11-14 15:10:49.351 26565 INFO migrate.versioning.api [-] 252 -> 253...
2014-11-14 15:10:49.845 26565 INFO migrate.versioning.api [-] done
2014-11-14 15:10:49.845 26565 INFO migrate.versioning.api [-] 253 -> 254...
2014-11-14 15:10:50.339 26565 INFO migrate.versioning.api [-] done
root@controller:/tmp/images#
```

d) Reinicie el los servicios de Nova:

```
# service nova-api restart
# service nova-cert restart
# service nova-consoleauth restart
# service nova-scheduler restart
# service nova-conductor restart
# service nova-novncproxy restart
```

```
root@controller:/tmp/images# service nova-api restart
nova-api stop/waiting
nova-api start/running, process 26608
root@controller:/tmp/images# service nova-cert restart
nova-cert stop/waiting
nova-cert start/running, process 26657
root@controller:/tmp/images# service nova-consoleauth restart
nova-consoleauth stop/waiting
nova-consoleauth start/running, process 26679
root@controller:/tmp/images# service nova-scheduler restart
nova-scheduler stop/waiting
nova-scheduler start/running, process 26700
root@controller:/tmp/images# service nova-conductor restart
nova-conductor stop/waiting
nova-conductor start/running, process 26723
root@controller:/tmp/images# service nova-novncproxy restart
nova-novncproxy stop/waiting
nova-novncproxy start/running, process 26766
root@controller:/tmp/images#
```

e) Ahora proceda a instalar Nova compute en el nodo de virtualización. Para ello, siga las instrucciones de la sección **"8.1.6.1 Instalar y configurar el nodo del servicio Compute (Nova Compute)"**. Al final de esa sección, al seguir las instrucciones del punto **d.**, puede verificar que Nova ha sido instalado con éxito.

7.6.6 Configuración del servicio Nova-network en el nodo controlador

a) Modifique el archivo de configuración de Nova (/etc/nova/nova.conf).

```
# vi /etc/nova/nova.conf
```

i. En la sección [DEFAULT], configure los API del grupo de redes y seguridad.

```
[DEFAULT]
```

```
...
network_api_class = nova.network.api.API
security_group_api = nova
```

b) Reinicie los servicios de Nova.

```
# service nova-api restart
# service nova-scheduler restart
# service nova-conductor restart
```

```
root@controller:/tmp/images# service nova-api restart
nova-api stop/waiting
nova-api start/running, process 27669
root@controller:/tmp/images# service nova-scheduler restart
nova-scheduler stop/waiting
nova-scheduler start/running, process 27718
root@controller:/tmp/images# service nova-conductor restart
nova-conductor stop/waiting
nova-conductor start/running, process 27740
root@controller:/tmp/images#
```

c) Ahora configure Compute Network en el nodo de virtualización. Las instrucciones se pueden encontrar en la sección **"7.7.2 Instalación y configuración del servicio Nova-network en el nodo de virtualización"**.

7.6.7 Instalación de la Interfaz web de OpenStack (Horizon)

Horizon es una interfaz web que permite a los administradores y usuarios de la nube administrar los servicios y recursos de OpenStack. Como parte de esta solución, se instala en el nodo controlador.

Además, Horizon permite interacciones por medio de la web con el servicio Compute de OpenStack, mediante el uso de los API y se ejecuta sobre el servicio web Apache.

a) Instale los siguientes paquetes:

```
# apt-get install openstack-dashboard apache2 libapache2-mod-wsgi memcached python-memcache
```

b) Modifique el archivo de configuración de Horizon de la siguiente manera:

```
# vi /etc/openstack-dashboard/local_settings.py
```

- i. Configure el Dashboard para que utilice los servicios de OpenStack del nodo controlador

```
OPENSTACK_HOST = "controller"
```

Nota: reemplace **controller** por el nombre o dirección IP del nodo controlador.

- ii. Habilite el Dashboard para todos los sistemas.

```
ALLOWED_HOSTS = ['*']
```

- iii. Configure la sesión "**memcached**" del servicio de almacenamiento. Comente cualquier otra sesión de la configuración de almacenamiento.

```
CACHES = {
    'default': {
        'BACKEND': 'django.core.cache.backends.memcached.MemcachedCache',
        'LOCATION': '127.0.0.1:11211',
    }
}
```

- iv. Configure la zona horaria (este paso es opcional).

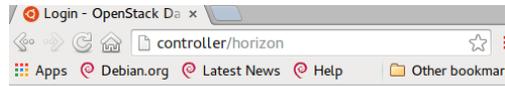
```
TIME_ZONE = "America/Costa_Rica"
```

- c) Reinicie el servicio web y el servicio de sesión de almacenamiento.

```
# service apache2 restart
# service memcached restart
```

- d) Ingresando al Dashboard, verifique que la instalación y configuración han sido realizadas de manera exitosa. Para autenticarse, utilice el usuario **admin** o **demo** y su contraseña respectiva.

```
http://controller/horizon
```



ubuntu[®] OpenStack Dashboard

Log In

User Name

Password

[Sign In](#)

ubuntu[®] OpenStack Dashboard
admin Sign Out

- Project >
- Admin >
 - System >
 - Overview
 - Hypervisors
 - Host Aggregates
 - Instances
 - Flavors
 - Images
 - Defaults
 - System Information
- Identity >

Overview

Usage Summary

Select a period of time to query its usage:

From: To: [Submit](#) The date should be in YYYY-mm-dd format.

Active Instances: 0 Active RAM: 0 bytes This Period's VCPU-Hours: 0 This Period's GB-Hours: 0

Usage

[Download CSV Summary](#)

| Project Name | VCPU | Disk | RAM | VCPU Hours | Disk GB Hours |
|----------------------|------|------|-----|------------|---------------|
| No items to display. | | | | | |
| Displaying 0 items | | | | | |

7.6.8 Instalación del controlador de volúmenes (Cinder)

Cinder es el servicio de bloques de almacenamiento de OpenStack, que se encarga de agregar almacenamiento persistente a una instancia de máquina virtual. Asimismo, proporciona infraestructura para la gestión de volúmenes e interactúa con OpenStack Compute para proporcionar volúmenes a las instancias. También, habilita la creación de respaldos de

volúmenes. El servicio se agrega en el nodo controlador y los volúmenes residen en el nodo de virtualización.

Para instalar y configurar este servicio, ejecute estas instrucciones en el nodo controlador:

7.6.8.1 Configuración de prerequisites de Cinder

- a) Antes de iniciar el proceso de instalación, debe crear la base de datos del servicio Cinder y agregarla al catálogo de servicios de Keystone.
 - i. Ingrese a la base de datos utilizando el usuario root.

```
# mysql -u root -p<ContraseñaBD>
```

Nota: reemplace **<ContraseñaBD>** con la contraseña de root de la base de datos.

- ii. Una vez conectado a la base de datos, cree la base de datos **Cinder** y asígnele los permisos utilizando los siguientes comandos:

```
CREATE DATABASE cinder;
GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'localhost' \
  IDENTIFIED BY '<ContraseñaCinderDB>';
GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'%' \
  IDENTIFIED BY '<ContraseñaCinderDB>';
```

Nota: Reemplace **<ContraseñaCinderDB>** con la una contraseña segura.

- iii. Salga de la base de datos escribiendo **exit**.

- b) Cargue las variables de entorno del tenant y usuario admin en Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Utilizando Keystone, cree el usuario Cinder. Asocie dicho usuario al tenant service y al role admin, cree el servicio Cinder y el "endpoint" de este servicio utilizando los comandos descritos a continuación:

```
# keystone user-create --name cinder --pass <ContraseñaCinder>
# keystone user-role-add --user cinder --tenant service --role admin
# keystone service-create --name cinder --type volume \
  --description "Servicio de objetos de almacenamiento de OpenStack"
# keystone service-create --name cinderv2 --type volumev2 \
```

```

--description "Servicio de objetos de almacenamiento de OpenStack"
# keystone endpoint-create \
--service-id $(keystone service-list | awk '/ volume / {print $2}') \
--publicurl http://controller:8776/v1/%(tenant_id)s \
--internalurl http://controller:8776/v1/%(tenant_id)s \
--adminurl http://controller:8776/v1/%(tenant_id)s \
--region regionOne
# keystone endpoint-create \
--service-id $( keystone service-list | awk '/ volumev2 / {print $2}') \
--publicurl http://controller:8776/v2/%(tenant_id)s \
--internalurl http://controller:8776/v2/%(tenant_id)s \
--adminurl http://controller:8776/v2/%(tenant_id)s \
--region regionOne

```

Nota: Reemplace **<ContraseñaCinder>** con la una contraseña segura y, **controller**, con el nombre o dirección IP del nodo controlador.

7.6.8.2 Instalación y configuración de Cinder

a) Instale los siguientes paquetes:

```
# apt-get install cinder-api cinder-scheduler python-cinderclient
```

b) Modifique el archivo de configuración de Cinder `/etc/cinder/cinder.conf`. Siga estas instrucciones:

```
# vi /etc/cinder/cinder.conf
```

i. En la sección `[DEFAULT]` configure la opción `volume_group` de la siguiente manera:

```
volume_group = nova-volumes
```

ii. Busque la sección `[database]` y configure el acceso a la base de datos como se muestra seguidamente (agregue la sección en caso de no existir):

```
[database]
...
```

```
connection = mysql://cinder:<ContraseñaCinderDB>@controller/cinder
```

Nota: reemplace <ContraseñaCinderDB> con la contraseña correspondiente.

- iii. En la sección [DEFAULT] configure el acceso a RabbitMQ. Configure la estrategia de autenticación y configure my_ip con la IP privada del nodo controlador:

```
[DEFAULT]
...
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = <ContraseñaRabbitMQ>

auth_strategy = keystone

my_ip = 192.168.0.11
```

Nota: reemplace <ContraseñaRabbitMQ> por la contraseña de RabbitMQ. Reemplace 192.168.0.11 con la IP de la red privada del nodo controlador.

- iv. En la sección [keystone_authtoken] configure el acceso a Keystone :

```
[keystone_authtoken]
...
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = cinder
admin_password = <ContraseñaCinder>
```

Nota: Reemplace <ContraseñaCinder> con la contraseña correspondiente. Además, reemplace **controller** con el nombre o dirección IP del nodo controlador. En caso de que haya alguna variable auth_host, auth_port, auth_protocol, coméntela. Recuerde que la opción identity_uri es el reemplazo de esas otras variables de configuración.

- v. Habilite la opción Verbose en la sección [DEFAULT]. Verbose es útil para pruebas. En caso de utilizarla en producción no es necesario que esté activa.

```
[DEFAULT]
...
verbose = True
```

- c) Complete la base de datos Cinder ejecutando el siguiente comando:

```
# su -s /bin/sh -c "cinder-manage db sync" cinder
```

Nota: verá una serie de registros desplegarse en la terminal, los cuales finalizan en **done**, como se muestra en la imagen.

```
2014-11-22 22:13:44.494 9292 INFO migrate.versioning.api [-] 17 -> 18...
2014-11-22 22:13:45.105 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:45.105 9292 INFO migrate.versioning.api [-] 18 -> 19...
2014-11-22 22:13:45.606 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:45.606 9292 INFO migrate.versioning.api [-] 19 -> 20...
2014-11-22 22:13:45.804 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:45.804 9292 INFO migrate.versioning.api [-] 20 -> 21...
2014-11-22 22:13:45.867 9292 INFO 021_add_default_quota_class [-] Added default quota class data into the
DB.
2014-11-22 22:13:45.886 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:45.886 9292 INFO migrate.versioning.api [-] 21 -> 22...
2014-11-22 22:13:46.075 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:46.076 9292 INFO migrate.versioning.api [-] 22 -> 23...
2014-11-22 22:13:46.215 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:46.216 9292 INFO migrate.versioning.api [-] 23 -> 24...
2014-11-22 22:13:46.789 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:46.789 9292 INFO migrate.versioning.api [-] 24 -> 25...
2014-11-22 22:13:48.074 9292 INFO migrate.versioning.api [-] done
2014-11-22 22:13:48.074 9292 INFO migrate.versioning.api [-] 25 -> 26...
2014-11-22 22:13:48.097 9292 INFO 026_add_consistencygroup_quota_class [-] Added default consistencygroup
s quota class data into the DB.
2014-11-22 22:13:48.115 9292 INFO migrate.versioning.api [-] done
root@controller:~#
```

- d) Reinicie los servicios del bloque de almacenamiento para aplicar la configuración realizada.

```
# service cinder-scheduler restart
# service cinder-api restart
```

- e) El proceso de instalación de Cinder en el nodo controlador se ha completado. Configure el almacenamiento en el nodo de virtualización. Idealmente, esta configuración se podría realizar en un nodo separado; sin embargo, para efectos de esta implementación se llevará a cabo en el nodo de virtualización, puesto que posee 1TB de espacio asignado para esta tarea. Seguidamente, configure Cinder Volume en el nodo de virtualización. Las instrucciones se pueden encontrar en la sección **"7.7.4 Instalación y configuración de Cinder-volume en el nodo de virtualización"**.

7.7 Instalación del nodo de virtualización

Los siguientes servicios de OpenStack se instalarán en el nodo de virtualización:

- Nova Compute
- Hipervisor Nova
- KVM
- Nova Network(servicio básico de redes de OpenStack)
- Cinder-volume

7.7.1 Instalar y configurar el nodo del servicio Compute (Nova Compute)

En esta sección podrá encontrar las instrucciones para instalar nodos del servicio Compute, con los cuales se puede escalar su ambiente de manera horizontal al agregar más de estos nodos. La configuración utiliza QEMU con la extensión KVM.

- a) Instale los siguientes paquetes:

```
# apt-get install nova-compute sysfsutils
```

- b) Modifique el archivo de configuración de Nova (/etc/nova/nova.conf) con los siguientes detalles:

```
# vi /etc/nova/nova.conf
```

- i. En la sección [DEFAULT] configure el acceso a RabbitMQ.

```
[DEFAULT]
...
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = <ContraseñaRabbitMQ>
```

Nota: reemplace **<ContraseñaRabbitMQ>** con la contraseña del usuario **guest**, asignada durante la fase de configuración de RabbitMQ en la sección Configuración de RabbitMQ (Sección **7.6.2**, punto **b**).

- ii. En las secciones [DEFAULT] y [keystone _authtoken] configure el acceso a Keystone.

```
[DEFAULT]
...
auth_strategy = keystone

[keystone_authtoken]
...
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = nova
admin_password = <ContraseñaNova>
```

Nota: Reemplace <ContraseñaNova> con la contraseña que le asignó al usuario Nova en Keystone y, **controller**, por el nombre o dirección IP del nodo controlador. En caso de que haya alguna variable auth_host, auth_port, auth_protocol, coméntela. Recuerde que la opción identity_uri es el reemplazo de esas otras variables de configuración.

- iii. En la sección [DEFAULT] configure las siguientes variables:

```
[DEFAULT]
...
my_ip = 192.168.0.21
vnc_enabled = True
vncserver_listen = 0.0.0.0
vncserver_proxyclient_address = 192.168.0.21
novncproxy_base_url = http://controller:6080/vnc_auto.html
```

Nota: reemplace **192.168.0.21** con la dirección IP correspondiente asignada en la subnet privada del nodo de virtualización. También, reemplace **controller** con el nombre o dirección IP del nodo controlador.

- iv. En la sección [glance] configure la ubicación de Glance.

```
[glance]
...
host = controller
```

Nota: reemplace **controller** por el nombre o dirección IP del nodo controlador.

- v. Habilite la opción **verbose** en la sección [DEFAULT]. Verbose es útil para pruebas y, en caso de utilizarla en producción no es necesario que esté activa.

```
[DEFAULT]
...
verbose = True
```

- vi. Determine si el equipo en el que está instalado el nodo de virtualización soporta aceleración por hardware para las máquinas virtuales.

```
# egrep -c '(vmx|svm)' /proc/cpuinfo
```

Recuerde:

Si el comando retorna un valor mayor que 1, su equipo soporta aceleración por hardware por lo que puede utilizar KVM.

Si el comando retorna un valor igual a 0, su equipo no soporta aceleración por hardware y debe utilizar QEMU en lugar de KVM.

```
root@compute:~# egrep -c '(vmx|svm)' /proc/cpuinfo
8
root@compute:~# █
```

- vii. Si su equipo soporta aceleración por hardware continúe en el paso (c), si no, siga las instrucciones que se detallan a continuación:

- Modifique el archivo de configuración de Nova compute (/etc/nova/nova-compute.conf) en la sección [libvirt].

```
[libvirt]
...
virt_type = qemu
```

c) Reinicie el servicio Nova compute para aplicar los cambios.

```
# service nova-compute restart
```

d) Para verificar si la instalación y configuración se realizaron de manera exitosa siga estos pasos:

i. Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

ii. Cargue las variables de entorno del tenant y usuario admin de Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

iii. Obtenga la lista de servicios nova disponibles en el sistema. La salida debe indicar cuatro componentes instalados en el nodo controlador y uno, en el nodo de virtualización, como se puede ver en la imagen.

```
# nova service-list
```

```
root@controller:/tmp/images# nova service-list
```

| Id | Binary | Host | Zone | Status | State | Updated_at |
|----|------------------|------------|----------|---------|-------|----------------------------|
| 1 | nova-cert | controller | internal | enabled | up | 2014-11-14T22:04:21.000000 |
| 2 | nova-consoleauth | controller | internal | enabled | up | 2014-11-14T22:04:16.000000 |
| 3 | nova-scheduler | controller | internal | enabled | up | 2014-11-14T22:04:21.000000 |
| 4 | nova-conductor | controller | internal | enabled | up | 2014-11-14T22:04:17.000000 |
| 6 | nova-compute | compute | nova | enabled | up | 2014-11-14T22:04:15.000000 |

```
root@controller:/tmp/images#
```

iv. Obtenga la lista de imágenes que se encuentra en el catálogo de imágenes de Glance, con el fin de verificar que la conectividad con Keystone y Glance funcionan adecuadamente.

```
# nova image-list
```

```

root@controller:/tmp/images# nova image-list
+-----+-----+-----+-----+
| ID                | Name                | Status | Server |
+-----+-----+-----+-----+
| 57b9849d-2ec1-4f0e-9094-5f1949a84869 | cirros-0.3.3-x86_64 | ACTIVE |        |
+-----+-----+-----+-----+
root@controller:/tmp/images#

```

7.7.2 Instalación y configuración del servicio Nova-network en el nodo de virtualización

Esta sección explica el proceso de instalación de un sistema bastante simple que provee direcciones IP a las instancias virtuales utilizando DHCP. Mediante varios nodos de virtualización se puede sacar provecho de la funcionalidad de MultiHost que proporciona redundancia, pues reparte las funciones de red a través de todos los nodos de virtualización existentes.

- a) Instale los paquetes de Nova-network.

```
# apt-get install nova-network nova-api-metadata
```

- b) Configure Nova-network. Para ello, modifique el archivo de configuración `/etc/nova/nova.conf` con los detalles que verá en seguida:

```
# vi /etc/nova/nova.conf
```

- i. En la sección `[DEFAULT]` configure los siguientes parámetros de red:

```

[DEFAULT]
...
network_api_class = nova.network.api.API
security_group_api = nova
firewall_driver = nova.virt.libvirt.firewall.IptablesFirewallDriver
network_manager = nova.network.manager.FlatDHCPManager
network_size = 254
allow_same_net_traffic = False
multi_host = True
send_arp_for_ha = True

```

```
share_dhcp_address = True
force_dhcp_release = True
flat_network_bridge = br100
flat_interface = <NombreInterfazRed>
public_interface = <NombreInterfazRed>
```

Note: Reemplace **<NombreInterfazRed>** con el nombre de la interfaz de red conectada a la red externa.

c) Reinicie los servicios de Nova-network.

```
# service nova-network restart
# service nova-api-metadata restart
```

```
root@compute:~# service nova-network restart
nova-network stop/waiting
nova-network start/running, process 5391
root@compute:~# service nova-api-metadata restart
nova-api-metadata stop/waiting
nova-api-metadata start/running, process 5419
root@compute:~# █
```

7.7.3 Creación de la red virtual en Nova-network

Para empezar a crear instancias virtuales se requiere infraestructura de la red virtual a la que se van a conectar esas instancias. Típicamente, esta red provee acceso a Internet a las máquinas virtuales mediante el uso de una IP y una serie de reglas de seguridad. El tenant llamado **admin** es el dueño de esta red, pues brinda acceso externo a la red de muchos otros tenants.

Además, esta red comparte la misma subnet asociada con la red física a la que se conecta la tarjeta de red en el nodo de virtualización. Es necesario especificar el rango de esa red que será usado por las instancias virtuales para evitar interferencia con otros dispositivos de la red externa.

Para crear la red virtual, siga las instrucciones que se detallan a continuación.. Tome en cuenta que los comandos se deben ejecutar en el nodo controlador.

a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace **<usuario>** y **<controller>** por los valores correspondientes.

b) Cargue las variables de ambiente del tenant y usuario **admin** de Keystone:

```
# source /var/lib/keystone/admin-openrc.sh
```

c) Cree la red utilizando el comando que verá a continuación:

```
# nova network-create virt-net --bridge br100 --multi-host T \
--fixed-range-v4 192.168.1.0/24
```

Nota: reemplace **192.168.1.0/24** con la subnet asociada a la red externa de la red física. Este comando no produce ninguna salida. También, puede modificar el nombre de la red (virt-net) por el nombre que guste utilizar para identificar esta red.

d) Verifique que la red ha sido creada de manera exitosa.

```
# nova net-list
```

```
root@controller:~# nova net-list
+-----+-----+
| ID                | Label  | CIDR          |
+-----+-----+
| 229c935d-fc75-48bd-8b31-b0db64eb0e46 | demo-net | 192.168.10.0/24 |
+-----+-----+
root@controller:~#
```

7.7.4 Instalación y configuración de Cinder-volume en el nodo de virtualización

a) Inicie sesión en el nodo de virtualización y hágase root.

```
# ssh <usuario>@compute
```

```
# sudo su -
```

Nota: reemplace **<usuario>** por el usuario correspondiente y, **compute**, con la dirección IP o nombre del nodo de virtualización.

b) Instale el siguiente paquete:

```
# apt-get install cinder-volume
```

- c) Modifique el archivo ***/etc/cinder/cinder.conf*** siguiendo las instrucciones que verá a continuación.

```
# vi /etc/cinder/cinder.conf
```

- i. En la sección [DEFAULT] configure la opción `volume_group` de la siguiente manera:

```
volume_group = nova-volumes
```

- ii. En la sección [database] configure el acceso a la base de datos (en caso de no existir la sección créela).

```
[database]
...
connection = mysql://cinder:<ContraseñaCinderDB>@controller/cinder
```

Nota: reemplace **<ContraseñaCinderDB>** con la contraseña de la base de datos Cinder creada en el nodo controlador.

- iii. En la sección [DEFAULT] configure el acceso a RabbitMQ.

```
[DEFAULT]
...
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = <ContraseñaRabbitMQ>
```

Nota: Reemplace **<ContraseñaRabbitMQ>** con la contraseña correspondiente y, **controller**, por el nombre o dirección IP del nodo controlador.

- iv. En las secciones [DEFAULT] y [keystone_authtoken], configure el acceso a Keystone.

```
[DEFAULT]
```

```

...
auth_strategy = keystone

[keystone_auth_token]
...
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = cinder
admin_password = <ContraseñaCinder>

```

Nota: reemplace **<ContraseñaCinder>** con la contraseña correspondiente y, **controller**, por la dirección IP o nombre del nodo controlador. En caso de encontrar alguna de las siguientes opciones, coméntelas: auth_host, auth_port, and auth_protocol.

- v. En la sección [DEFAULT] configure la ubicación del servicio Glance:

```

[DEFAULT]
...
glance_host = controller

```

Nota: reemplace **controller** por el nombre o dirección IP del nodo controlador.

- vi. Habilite la opción verbose en la sección [DEFAULT], puesto que es útil durante la fase de instalación; sin embargo, se debe deshabilitar en nodos de producción.

```

[DEFAULT]
...
verbose = True

```

- d) Para finalizar la instalación, reinicie los servicios de Cinder-volume para aplicar la configuración realizada.

```

# service tgt restart
# service cinder-volume restart

```

e) Para confirmar que Cinder ha sido instalado de la forma adecuada, siga las instrucciones que se muestran a continuación:

- i. Inicie sesión en el nodo controlador y hágase root.
- ii. Cargue las variables de ambiente del usuario y proyecto admin.

```
# source /var/lib/keystone/admin-openrc.sh
```

- iii. Utilice el siguiente comando para verificar el estado de los servicios de Cinder. El estado de todos los servicios debe ser **up**, como se muestra a continuación:

```
# cinder service-list
```

```
root@controller:~# cinder service-list
+-----+-----+-----+-----+-----+-----+-----+
| Binary          | Host       | Zone  | Status | State | Updated_at           | Disabled Reason |
+-----+-----+-----+-----+-----+-----+-----+
| cinder-scheduler | controller | nova  | enabled | up    | 2014-11-23T04:51:34.000000 | None            |
| cinder-volume    | compute   | nova  | enabled | up    | 2014-11-23T04:51:35.000000 | None            |
+-----+-----+-----+-----+-----+-----+-----+
root@controller:~#
```

- iv. Cargue las variables de entorno del proyecto y usuario demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- v. Cree un volumen de 1 GB de tamaño utilizando el siguiente comando:

```
# cinder create --display-name demo-volume1 1
```

```
root@controller:~# source /var/lib/keystone/demo-openrc.sh
root@controller:~# cinder create --display-name demo-volume1 1
+-----+-----+
| Property          | Value                                     |
+-----+-----+
| attachments       | []                                        |
| availability_zone  | nova                                      |
| bootable           | false                                    |
| created_at        | 2014-11-23T04:55:07.553084              |
| display_description | None                                      |
| display_name       | demo-volume1                             |
| encrypted          | False                                    |
| id                 | fc440efa-b80b-4885-88a4-eb4a3de0b5cc    |
| metadata           | {}                                        |
| size               | 1                                         |
| snapshot_id        | None                                      |
| source_volid       | None                                      |
| status             | creating                                  |
| volume_type        | None                                      |
+-----+-----+
root@controller:~#
```

- vi. Verifique que el volumen ha sido creado de manera exitosa y que se encuentra en estado **available**.

```
# cinder list
```

```
root@controller:~# cinder list
-----+-----+-----+-----+-----+-----+-----+-----+
| ID | Status | Display Name | Size | Volume Type | Bootable | Attached to |
|-----+-----+-----+-----+-----+-----+-----+
| fc440efa-b80b-4885-88a4-eb4a3de0b5cc | available | demo-volume1 | 1 | None | false | |
|-----+-----+-----+-----+-----+-----+-----+
root@controller:~#
```

- vii. En caso de presentar problemas puede revisar los logs de Cinder ubicados en el directorio `/var/log/cinder`.
- f) Una vez creado el volumen, se presenta en una máquina virtual existente. Dicho procedimiento se puede llevar a cabo desde Horizon o mediante la línea de comandos que verá en seguida:

```
# nova volume-attach demo-vm fc440efa-b80b-4885-88a4-eb4a3de0b5cc
```

Nota: reemplace **demo-vm** con el nombre de la máquina virtual a la que desee agregar el volumen y, **fc440efa-b80b-4885-88a4-eb4a3de0b5cc**, por el identificador del volumen deseado, que se puede encontrar con el comando del paso “**vi**”. Para más detalles, consulte la guía de Operaciones, en ella podrá encontrar instrucciones para gestionar volúmenes.

7.8 Solución de problemas

En caso de tener problemas durante la creación de instancias de máquinas virtuales, asegúrese de verificar que todos los servicios de nova están corriendo. Todos se deben encontrar en estado **up**. Durante la implementación de esta solución, al seguir paso a paso estas guías, ese fue el único problema detectado.

```

oot@controller:/tmp/imagenes# nova service-list
-----+-----+-----+-----+-----+-----+-----+-----+
Id | Binary          | Host       | Zone   | Status | State | Updated_at           | Disable
Reason |
-----+-----+-----+-----+-----+-----+-----+-----+
 1 | nova-cert       | controller | internal | enabled | up    | 2014-11-16T05:43:12.000000 | -
 2 | nova-consoleauth | controller | internal | enabled | up    | 2014-11-16T05:43:09.000000 | -
 3 | nova-scheduler  | controller | internal | enabled | up    | 2014-11-16T05:43:02.000000 | -
 4 | nova-conductor  | controller | internal | enabled | up    | 2014-11-16T05:43:02.000000 | -
 6 | nova-compute    | compute   | nova    | enabled | up    | 2014-11-16T05:43:11.000000 | -
 7 | nova-network    | compute   | internal | enabled | up    | 2014-11-16T05:43:08.000000 | -
-----+-----+-----+-----+-----+-----+-----+-----+

```

En caso de tener algún otro problema con la creación de instancias, puede consultar el siguiente enlace web que contiene una serie de recomendaciones: <http://docs.openstack.org/openstack-ops/content/instances.html>.

Recuerde revisar con detalle los logs de OpenStack. Los de Nova se encuentran en la ubicación `/var/log/nova/.log` de los nodos de virtualización y nodo controlador.*

En caso de tener problemas con Cinder puede revisar los logs ubicados en `/var/log/cinder/*`.

7.9 Pruebas

Esta sección identifica los elementos que necesitan ser probados para asegurarse de que la solución ha sido implementada adecuadamente.

La mejor manera de asegurarse de que su instalación funciona es utilizando OpenStack. Consulte la Guía de operaciones en la que podrá encontrar cómo realizar las siguientes tareas:

- Abra Horizon en su navegador y utilice los usuarios admin y demo para iniciar sesión (<http://controller/horizon>).

Nota: reemplace controller por el nombre o dirección IP del nodo controlador.

- Utilizando Horizon:
 - Cree una nueva imagen en Glance, puede ser de Ubuntu o Fedora. Utilice imágenes de 64 bits.

- Modifique el grupo de seguridad **default** o cree uno nuevo, que permita el acceso ICMP y SSH a las máquinas virtuales que se construyan bajo ese tenant.
 - Cree su propio key de SSH e impórtelo en OpenStack.
 - Cree su propio **flavor** utilizando los siguientes datos:
 - Nombre: VM-Peq
 - CPUs: 1
 - Memoria: 512 MB
 - Disco duro: 10 GB
 - Swap: 512 MB
 - Cree una máquina virtual. Use la imagen, así como el grupo de seguridad, la llave y el flavor creados durante este ejercicio.
 - Conéctese a la nueva máquina virtual por medio de la consola.
- Conéctese a la nueva máquina virtual por medio de SSH desde el nodo de virtualización.

Si todas estas tareas son realizadas de manera exitosa, puede considerar que la implementación de OpenStack también lo fue.

7.10 Mecanismo de respaldo

En ocasiones, existen factores imprevisibles que se presentan y hacen necesario regresar un sistema a su estado original.

Una vez en producción, es importante que todos los nodos que forman parte de la infraestructura (nodo controlador y nodo de virtualización), se sometan a un estricto control de respaldos. Dichos respaldos se deben realizar de manera periódica para evitar pérdidas importantes en sus sistemas y en sus instancias virtuales.

En el caso de las instancias virtuales, OpenStack cuenta con un mecanismo de **snapshots** que permite la creación de imágenes de respaldo de una máquina virtual específica, que se puede

utilizar para regresar la máquina virtual a su estado previo al momento de la creación del snapshot, o para crear otras máquinas virtuales a manera de clon.

8 Contactos

En seguida, se brinda información de las personas que puede contactar en caso de consultas, cambios técnicos, etc.

| Nombre | Rol | Dirección de correo-e |
|-----------------|---------------------------------------|------------------------------|
| Jennifer Vargas | Creación de la guía de implementación | jvargas@softwarelibre.cr.org |
| | | |



Guía de operaciones– Servicios de infraestructura en la nube con OpenStack

1.0 – Versión Final

25 de noviembre de 2014



Se permite el uso comercial de la obra y de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Licencia Creative Commons Atribución CompartirIgual 3.0 Costa Rica.

<http://creativecommons.org/licenses/by-sa/3.0/cr/legalcode>

Índice

| | |
|--|----------|
| 1 Control del documento..... | 3 |
| 1.1 Aprobación del documento..... | 3 |
| 1.2 Historial de cambios..... | 3 |
| 1.3 Documentación relacionada..... | 3 |
| 2 Introducción..... | 5 |
| 2.1 Objetivo..... | 5 |
| 2.2 Audiencia..... | 5 |
| 2.3 Terminología y acrónimos..... | 5 |
| 3 Resumen de los servicios operacionales..... | 7 |
| 3.1 Definición de los servicios incluidos..... | 7 |
| 3.2 Prerrequisitos para soportar esta solución..... | 7 |
| 4 Administración de OpenStack..... | 8 |
| 4.1 Configuraciones de apariencia de Horizon..... | 8 |
| 4.1.1 Establecer Horizon en español y ajustar la zona horaria desde Horizon..... | 8 |
| 4.1.2 Colocar los signos gráficos de la empresa..... | 9 |
| 4.2 Gestión de proyectos/tenants y usuarios en OpenStack..... | 13 |
| 4.2.1 Creación de proyectos/tenants y usuarios en OpenStack..... | 13 |
| 4.2.2 Eliminación de proyectos/tenants y usuarios en OpenStack..... | 15 |
| 4.3 Gestión de redes virtuales en OpenStack..... | 17 |
| 4.3.1 Crear redes..... | 18 |
| 4.3.2 Eliminar redes..... | 19 |
| 4.4 Creación y mantenimiento de imágenes en Glance..... | 19 |
| 4.4.1 Creación de imágenes en Glance..... | 20 |
| 4.4.1.1 Creación de imágenes utilizando línea de comandos..... | 20 |
| 4.4.1.2 Creación de imágenes utilizando Horizon..... | 22 |
| 4.4.2 Eliminación de imágenes en Glance..... | 25 |
| 4.4.2.1 Eliminar imágenes utilizando línea de comandos..... | 25 |
| 4.4.2.2 Eliminar imágenes utilizando Horizon..... | 26 |
| 4.5 Gestión de grupos de seguridad en OpenStack..... | 27 |
| 4.5.1 Crear de grupos de seguridad utilizando Horizon..... | 27 |
| 4.5.2 Eliminación de grupos de seguridad utilizando Horizon..... | 31 |
| 4.6 Gestión de una instancia de máquina virtual..... | 32 |
| 4.6.1 Definición de las plantillas de las máquinas virtuales..... | 32 |
| 4.6.2 Configuración de las políticas de los grupos de seguridad para habilitar acceso remoto a las máquinas virtuales..... | 34 |
| 4.6.3 Creación/Importación de llaves para asignar a las máquinas virtuales..... | 35 |
| 4.6.3.1 Importar una llave en OpenStack desde línea de comandos..... | 36 |
| 4.6.3.2 Importar una llave en OpenStack utilizando Horizon..... | 37 |
| 4.6.4 Creación de máquinas virtuales..... | 40 |
| 4.6.4.1 Creación de máquinas virtuales desde línea de comandos..... | 40 |
| 4.6.4.1.1 Tareas de Preparación..... | 40 |
| 4.6.4.1.2 Creación de la máquina virtual usando línea de comandos..... | 42 |
| 4.6.4.2 Creación de máquinas virtuales desde Horizon..... | 44 |

| | |
|---|-----------|
| 4.6.5 Conectarse a la consola de una máquina virtual..... | 47 |
| 4.6.6 Conectarse remotamente a una máquina virtual..... | 49 |
| 4.6.7 Reinstalar una máquina virtual..... | 51 |
| 4.6.8 Hacer snapshots de una máquina virtual..... | 53 |
| 4.6.9 Eliminar una máquina virtual..... | 54 |
| 4.6.9.1 Eliminar una máquina virtual desde línea de comandos..... | 55 |
| 4.6.9.2 Eliminar una máquina virtual desde Horizon..... | 56 |
| 4.7 Gestión de volúmenes utilizando Cinder..... | 57 |
| 4.7.1 Creación de volúmenes..... | 57 |
| 4.7.1.1 Creación de volúmenes utilizando línea de comandos..... | 57 |
| 4.7.1.2 Creación de volúmenes utilizando Horizon..... | 58 |
| 4.7.2 Eliminación de volúmenes..... | 61 |
| 4.7.2.1 Eliminación de volúmenes utilizando línea de comandos..... | 61 |
| 4.7.2.2 Eliminación de volúmenes utilizando Horizon..... | 62 |
| 4.7.3 Asignación de volúmenes a las máquinas virtuales..... | 62 |
| 4.7.3.1 Asignación de volúmenes a las máquinas virtuales mediante línea de comandos..... | 63 |
| 4.7.3.2 Asignación de volúmenes a las máquinas virtuales utilizando Horizon..... | 63 |
| 4.7.4 Desvinculación de volúmenes en las máquinas virtuales..... | 65 |
| 4.7.4.1 Desvinculación de volúmenes de las máquinas virtuales mediante línea de comandos..... | 65 |
| 4.7.4.2 Desvinculación de volúmenes de las máquinas virtuales utilizando Horizon..... | 66 |
| 5 Contactos..... | 68 |

1 Control del documento

1.1 Aprobación del documento

A continuación se mencionan las personas que revisaron este trabajo, verificaron su información y aprobaron el contenido con base en los requerimientos de los servicios definidos durante la creación del documento:

| Nombre | Rol | Fecha de aprobación |
|-------------------------------|-----------------|---------------------|
| M. Sc. Herson Esquivel Vargas | Profesor asesor | 25 nov 2014 |
| M. Sc. Jaime Gutiérrez Álfaro | Lector Interno | 25 nov 2014 |
| M. Sc. Jorge Ortega Pérez | Lector Externo | 25 nov 2014 |
| | | |
| | | |
| | | |
| | | |

1.2 Historial de cambios

La siguiente bitácora de cambios contiene un registro de las modificaciones realizadas:

| Publicado / Revisado | Versión # | Autor (opcional) | Sección / Naturaleza del cambio |
|----------------------|-----------|---|--|
| 25 nov 2014 | 1.0 | Ing. Jennifer Vargas | Versión final de la guía de operaciones. |
| 16 feb 2014 | 1.1 | Bach. Cristina Vargas Ramírez Ing. Jennifer Vargas | Revisión filológica y correcciones. |
| | | | |
| | | | |

1.3 Documentación relacionada

Los siguientes documentos son necesarios para utilizar apropiadamente esta guía. Asegúrese de tener acceso a ellos al utilizarla.

| Nombre del documento | Pequeña descripción |
|--|---|
| Guía de referencias técnicas – Servicios de infraestructura en la nube con OpenStack | Esta guía posee detalles sobre la arquitectura y diseño de esta solución. Incluye las especificaciones de los paquetes de software que se utilizarán, así como las características del equipo sugerido. |
| Guía de implementación – Servicios de infraestructura en la nube con OpenStack | Esta guía posee todos los detalles de configuración, instalación e implementación de la solución. |
| Documentación oficial de OpenStack | http://docs.openstack.org/ En el repositorio de documentación de OpenStack se puede encontrar documentación como guías de usuario, de operatividad. Además, documentos para aprender a utilizar los API de OpenStack, guías de instalación en las distintas distribuciones de Linux, entre otras. |
| Capacitación de OpenStack | http://docs.openstack.org/training-guides/content/ OpenStack ha puesto a disposición del público guías con ejercicios y capacitaciones que cualquier persona puede utilizar para aprender sobre el proyecto. |

2 Introducción

Esta sección describe el propósito, la audiencia y la terminología de este documento.

2.1 Objetivo

El propósito de esta guía es proporcionar información operativa sobre la solución para dar servicios de infraestructura en la nube utilizando OpenStack. Para ello, se brindan detalles operacionales sobre el uso de los distintos servicios de OpenStack, ya sea por medio de línea de comandos o desde el Portal de Administración Web de OpenStack (Horizon). En algunos casos, las instrucciones de uso estarán limitadas a líneas de comandos desde alguno de los nodos de OpenStack o a instrucciones que expliquen cómo realizar alguna tarea específica desde Horizon.

2.2 Audiencia

La guía ha sido creada con el fin de apoyar, principalmente, a pequeñas y medianas empresas de Costa Rica que estén interesadas en la implementación de una solución para proporcionar servicios de infraestructura en la nube, ya sea para uso interno o para dar servicios de hospedaje a empresas externas.

La guía se enfoca en el uso por parte de las PYMES costarricenses, pero no está limitada a ellas y puede ser accedida por otro tipo de audiencia. Estas guías se pueden distribuir y modificar libremente, siempre que se respeten los términos de su licencia (CC BY-SA 3.0 CR).

2.3 Terminología y acrónimos

En esta sección se define la terminología y los acrónimos comúnmente utilizados en este documento.

| Término/Acrónimo | Definición |
|-------------------|--|
| Nodo | Equipo virtual o físico en el cual se instala un sistema operativo con sus respectivos servicios. |
| Aprovisionamiento | Significa proveer, acondicionar y habilitar un servicio para que el usuario final se pueda beneficiar con él y, de este modo, satisfaga sus requerimientos con un servicio bajo demanda con la calidad acordada. |

| Término/Acrónimo | Definición |
|----------------------|---|
| Escalabilidad | Indica la habilidad de un sistema para reaccionar y adaptarse a las necesidades continuas del negocio sin perder calidad. Se debe manejar un crecimiento continuo de trabajo de manera fluida, o bien estar preparado para expandirse sin perder la calidad de los servicios ofrecidos. |
| Orquestación | Describe el arreglo, automatización, coordinación y gestión de sistemas de computadores complejos, así como servicios. Trata de alinear las solicitudes del negocio con las aplicaciones, los datos y la infraestructura. |
| Horizon | Es un servicio que proporciona un portal web que interactúa con los distintos servicios de OpenStack, por ejemplo: creación de instancias virtuales, asignación de direcciones IP y configuración de controles de acceso. |
| Nova/Compute | Se encarga de gestionar el ciclo de vida de las instancias virtuales en un ambiente OpenStack, así como del aprovisionamiento y deprovisionamiento de máquinas virtuales bajo demanda. |
| Neutrón | Este componente de OpenStack se encarga de proporcionar el servicio de Red para los otros servicios o componentes de OpenStack. Posee un API con el cual los usuarios puedan definir sus redes y cualquier dispositivo conectado a ellas. Soporta dispositivos de red de una gran cantidad de vendedores y tecnologías. |
| Swift | Este es el objeto de almacenamiento de OpenStack. Es altamente tolerante a fallos, posee una arquitectura escalable y redundante que realiza replicación de datos a través de todos sus nodos. |
| Cinder | Proporciona bloques de almacenamiento persistentes a instancias de máquinas virtuales en tiempo de ejecución. |
| Keystone | Este componente es el encargado de proveer servicios de autenticación y autorización a los demás componentes de OpenStack. |
| Glance | Se encarga de almacenar y obtener imágenes de las máquinas virtuales. OpenStack utiliza este componente durante la fase de aprovisionamiento. |
| Ceilometer/Telemetry | Se encarga de la monitorización y medición de la nube de OpenStack para propósitos financieros, de control de calidad, escalabilidad y propósitos estadísticos. |
| Heat | Es el servicio de orquestación de OpenStack. |
| Trove | Este es un nuevo componente de Openstack liberado en la versión Juno . Brinda el servicio de "Base de datos como un servicio" para bases de datos tanto relacionales como no relacionales. |

3 Resumen de los servicios operacionales

Esta sección describe los procesos necesarios para operar de manera específica los servicios incluidos como parte de esta solución.

3.1 Definición de los servicios incluidos

Este documento se basa en un conjunto de servicios que se definieron y seleccionaron durante la fase de diseño de la solución, los cuales fueron revisados y aprobados en la ejecución del proyecto. La definición de estos servicios la podrá encontrar en "**Guía de referencias técnicas – Servicios de infraestructura en la nube con OpenStack**".

3.2 Prerrequisitos para soportar esta solución

A continuación, encontrará los requerimientos de uso de este documento, con los cuales podrá soportar los servicios definidos.

Es importante leer y entender lo siguiente:

- El contenido de este documento.
- Leer el contenido de la Guía de referencias técnicas.
- Tener un concepto claro de todos los componentes o servicios de OpenStack utilizados en esta solución.
- Entender qué tipo de tareas deberá realizar como parte de esta solución.
- Leer el contenido de los documentos relacionados.
- Tener conocimiento básico de Administración Linux y del uso de la terminal.

Además, se recomienda que al menos una persona se dedique al mantenimiento del ambiente y otra a las labores operativas. A pesar de que ambos roles pueden ser realizados por la misma persona, no es recomendable.

4 Administración de OpenStack

En esta sección podrá encontrar las instrucciones necesarias para gestionar de manera exitosa cada uno de los servicios de OpenStack incluidos en la solución de Servicios de infraestructura en la nube.

En cada una de las secciones que se presentan a continuación, podrá encontrar una guía detallada sobre el uso de los servicios de OpenStack implementados de acuerdo con las instrucciones de la **"Guía de Implementación– Servicios de Infraestructura en la nube con OpenStack"**.

4.1 Configuraciones de apariencia de Horizon

4.1.1 Establecer Horizon en español y ajustar la zona horaria desde Horizon

Luego de instalar Horizon, la interfaz administrativa se encontrará, por defecto, en inglés. Para establecer la interfaz en español siga estas instrucciones:

- a) Utilice la dirección <http://controller/horizon> para abrir Horizon desde su navegador. Recuerde que **controller** es el nombre o dirección IP del nodo controlador.
- b) Inicie sesión utilizando el usuario admin, así como las credenciales de él.
- c) Haga clic sobre el "triángulo" que se encuentra a un lado del botón admin ubicado en la esquina superior derecha. Se desplegará un menú, haga clic sobre la opción "Settings".
- d) Se desplegará una ventana en la que puede cambiar tanto el idioma como la zona horaria. Seleccione "español" y la zona horaria de Costa Rica. Una vez hechas las modificaciones, haga clic sobre el botón "Save" para aplicar los cambios.

User Settings

Language *

Timezone *

Items Per Page * ?

Description:
 Modify dashboard settings for your user.

[Save](#)

e) De manera inmediata la interfaz web se podrá ver en español:

ubuntu OpenStack Dashboard admin

Ajustes de usuario

Ajustes de usuario

Idioma *

Zona horaria *

4.1.2 Colocar los signos gráficos de la empresa

OpenStack puede ser configurado a la medida para lucir el logo de su compañía de manera sencilla. Siga las siguientes instrucciones para cambiar el logo, el título en el “dashboard” o el enlace de la marca para que apunte al sitio web de su empresa.

- a) Inicie sesión en el nodo controlador por medio de SSH.
- b) Ubuntu instala por defecto su propio tema de Horizon, puede utilizar el siguiente comando para eliminarlo:

```
# apt-get remove --auto-remove openstack-dashboard-ubuntu-theme
```

- c) Para cambiar el título del sitio web puede modificar el archivo de configuración de Horizon ubicado en `/etc/openstack-dashboard/local_settings.py`. Al final del archivo, agregue la variable `SITE_BRANDING` con el nombre de su empresa.

```
# vi /etc/openstack-dashboard/local_settings.py
```

```
# Compress all assets offline as part of packaging installation
COMPRESS_OFFLINE = True
SITE_BRANDING = "TEC"
```

Nota: Reemplace TEC por el nombre que quiera que tenga el título del sitio web.

- d) También, puede cambiar el enlace del título y agregar el sitio web de su empresa al escribir la variable `SITE_BRANDING_LINK` al final del mismo archivo:

```
SITE_BRANDING_LINK = "http://www.tec.ac.cr"
```

- e) Para modificar el logo siga estas instrucciones:
 - i. Deberá crear dos logos con fondos transparentes de tamaño 365x50 (`logo_med`) y 216x50(`logo_peq`), los cuales serán utilizados en la pantalla de login y en la interfaz de un usuario cuando ha iniciado sesión.
 - ii. Una vez creados los logos, cópielos en el directorio `/usr/share/openstack-dashboard/openstack_dashboard/static/dashboard/img/` del nodo controlador.
 - iii. Asegúrese de que los nuevos logos pertenezcan al usuario y al grupo Horizon.

```
# chown horizon:horizon /usr/share/openstack-
dashboard/openstack_dashboard/static/dashboard/img/logo_peq.png
# chown horizon:horizon /usr/share/openstack-
dashboard/openstack_dashboard/static/dashboard/img/logo_med.png
```

Nota: Reemplace `logo_peq.png` y `logo_med.png` por cada uno de los logos creados.

```
root@controller:/# chown horizon:horizon /usr/share/openstack-dashboard/static/dashboard/img/logo_peq.png
root@controller:/# chown horizon:horizon /usr/share/openstack-dashboard/static/dashboard/img/logo_med.png
```

- iv. Cree una hoja de estilo **`/usr/share/openstack-dashboard/openstack_dashboard/static/dashboard/css/micss.css`** (reemplace **`micss`** por el nombre que guste asignarle a esta hoja de estilo) con el siguiente contenido:

```
/*
 * New theme colors for dashboard that override the defaults:
 * dark blue: #355796 / rgb(53, 87, 150)
 * light blue: #BAD3E1 / rgb(186, 211, 225)
 *
 * By Preston Lee <plee@tgen.org>
 */
h1.brand {
background: #355796 repeat-x top left;
border-bottom: 2px solid #BAD3E1;
}
h1.brand a {
background: url(../img/logo_peq.png) top left no-repeat;
}
#splash .login {
background: #355796 url(../img/logo_med.png) no-repeat center 35px;
}
#splash .login .modal-header {
border-top: 1px solid #BAD3E1;
}
.btn-primary {
background-image: none !important;
background-color: #355796 !important;
border: none !important;
```

```

box-shadow: none;
}
.btn-primary:hover,
.btn-primary:active {
border: none;
box-shadow: none;
background-color: #BAD3E1 !important;
text-decoration: none;
}

```

Nota: Reemplace **logo_peq.png** y **logo_med.png** por cada uno de los logos creados.

- v. Cambie el dueño del archivo de la hoja de estilo a Horizon:

```
# chown horizon:horizon /usr/share/openstack-
dashboard/openstack_dashboard/static/dashboard/css/micss.css
```

- vi. Abra el archivo */usr/share/openstack-dashboard/openstack_dashboard/templates/_stylesheets.html* y agregue las siguientes líneas para incluir la nueva hoja de estilo. La línea debe quedar dentro de la sección **compress** tal como se muestra a continuación:

```

<link href='{{ STATIC_URL }}bootstrap/css/bootstrap.min.css' media='screen'
rel='stylesheet' />
<link href='{{ STATIC_URL }}dashboard/css/{% choose_css %}' media='screen'
rel='stylesheet' />
<link href='{{ STATIC_URL }}dashboard/css/micss.css' media='screen'
rel='stylesheet' />

```

```

{% Load compress %}
{% compress css %}
<link href="{{ STATIC_URL }}dashboard/scss/horizon.scss" type="text/scss" media="screen" rel="stylesheet" />
</link>
<link href="{{ STATIC_URL }}dashboard/css/rickshaw.css" type="text/css" media="screen" rel="stylesheet" />
</link>
<link href="{{ STATIC_URL }}bootstrap/css/bootstrap.min.css" media="screen" rel="stylesheet" />
<link href="{{ STATIC_URL }}dashboard/css/{% choose_css %}" media="screen" rel="stylesheet" />
<link href="{{ STATIC_URL }}dashboard/css/micss.css" media="screen" rel="stylesheet" />
<link href="{{ STATIC_URL }}horizon/lib/bootstrap_datepicker/datepicker3.css" type="text/css" media="screen" rel="stylesheet" />
<link href="{{ STATIC_URL }}dashboard/scss/horizon_charts.scss" type="text/scss" media="screen" rel="stylesheet" />
<link href="{{ STATIC_URL }}dashboard/scss/horizon_workflow.scss" type="text/scss" media="screen" rel="stylesheet" />
<link href="{{ STATIC_URL }}horizon/lib/font-awesome/scss/font-awesome.scss" type="text/scss" media="screen" rel="stylesheet" />
{% endcompress %}
<link rel="shortcut icon" href="{{ STATIC_URL }}dashboard/img/favicon.ico"/>

```

vii. Si lo desea, puede hacer otras modificaciones a su nueva hoja de estilo.

f) Reinicie el servicio apache para aplicar los cambios.

```
# service apache2 restart
```

También, existe la opción de crear su propia versión de Horizon con la estructura deseada. En caso de tener interés en ese tipo de trabajo puede leer la documentación de OpenStack que contiene los detalles. Remítase al siguiente vínculo: <http://docs.openstack.org/developer/horizon/topics/tutorial.html>

4.2 Gestión de proyectos/tenants y usuarios en OpenStack

4.2.1 Creación de proyectos/tenants y usuarios en OpenStack

El proceso de creación de tenants (proyectos) y usuarios en OpenStack se debe hacer de manera manual desde el nodo controlador.

Para crear un tenant y un usuario nuevo, tome en cuenta las siguientes instrucciones:

a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@<controller>
```

Nota: reemplace <usuario> y <controller> por los valores correspondientes.

b) Cargue las variables de ambiente del tenant y usuario admin.

```
# source /var/lib/keystone/admin-openrc.sh
```

c) Cree el tenant/proyecto con el nombre deseado.

```
# keystone tenant-create --name tec --description "Tenant TEC"
```

Nota: reemplace los valores en rojo con los valores deseados.

```
root@controller:/tmp/imagenes# keystone tenant-create --name tec --description "Tenant TEC"
+-----+-----+
| Property | Value |
+-----+-----+
| description | Tenant TEC |
| enabled | True |
| id | a9ac10f2239f4885817f40993511d20b |
| name | tec |
+-----+-----+
root@controller:/tmp/imagenes#
```

d) Cree el usuario **tec** en Keystone (con el mismo nombre del tenant).

```
# keystone user-create --name tec --pass <ContraseñaUsuario> --email <DirecciónE-mail>
```

Nota: reemplace **tec**, **<ContraseñaUsuario>**, y **<DirecciónE-mail>** por el nombre de usuario que desee, una contraseña segura y una dirección de correo que quiera utilizar para estos efectos. Recuerde que puede generar contraseñas aleatorias utilizando el siguiente comando:

```
# openssl rand -hex 10
```

```
root@controller:/tmp/imagenes# keystone user-create --name tec --pass 123456 --email jvargas@softwarelibrecr.org
+-----+-----+
| Property | Value |
+-----+-----+
| email | jvargas@softwarelibrecr.org |
| enabled | True |
| id | 8e2779d920084609a298c45d3c02fb4b |
| name | tec |
| username | tec |
+-----+-----+
root@controller:/tmp/imagenes#
```

e) Asocie el usuario y el tenant creado con el rol **_member_** utilizando el siguiente comando:

```
# keystone user-role-add --tenant tec --user tec --role _member_
```

Nota: Este comando no producirá ninguna salida.

- f) En `/var/lib/keystone/tec-openrc.sh`, cree el script de openRC para este usuario.

```
# vi /var/lib/keystone/tec-openrc.sh
```

Agregue el siguiente contenido:

```
export OS_TENANT_NAME=tec
export OS_USERNAME=tec
export OS_PASSWORD=<ContraseñaUsuario>
export OS_AUTH_URL=http://controller:5000/v2.0
```

Nota: reemplace `tec`, `<ContraseñaUsuario>` y `controller` con los valores correspondientes.

- g) Para ejecutar operaciones desde el nodo controlador bajo este tenant, deberá cargar las variables de ambiente utilizando el siguiente comando, así como el script creado en el punto anterior.

```
# source /var/lib/keystone/tec-openrc.sh
```

- h) Para probar si su usuario funciona, puede iniciar sesión en Horizon utilizando `http://controller/horizon`.
- i) Recuerde que cada tenant tiene sus propios grupos de seguridad, llaves e instancias. Un tenant sin privilegios administrativos no puede ver las instancias existentes en otro tenant ni comunicarse con otro tenant.
- j) En caso de necesitar más tenants y usuarios o roles, puede seguir este mismo procedimiento.

4.2.2 Eliminación de proyectos/tenants y usuarios en OpenStack

Antes de eliminar un tenant o usuario debe tomar en cuenta que las máquinas virtuales asociadas a esos tenants no serán eliminadas. Los recursos seguirán siendo utilizados por las instancias virtuales creadas durante la operación de ese proyecto y las máquinas virtuales seguirán ejecutándose. Para liberar los recursos, las instancias deben ser eliminadas antes del tenant y los usuarios asociados, ya que una vez eliminado un tenant, los usuarios asociados no podrán tener los equipos desde Horizon, pero tendrán acceso remoto a los equipos.

- a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace **<usuario>** y **<controller>** por los valores correspondientes.

- b) Cargue las variables de ambiente del tenant y usuario admin.

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Si desea eliminar un tenant siga las instrucciones que se indican a continuación:

- i. Utilice el siguiente comando para ver la lista de tenants existentes en Keystone:

```
# keystone tenant-list
```

```
root@controller:/tmp/imagenes# keystone tenant-list
+-----+-----+-----+
|          id          | name | enabled |
+-----+-----+-----+
| 5a0d3c1c49cb4e67aada421c3b26539f | admin | True |
| e2e2410014c949e88892454493ed7f72 | demo | True |
| 5b9ffcbee982462d9b50224453f25c10 | service | True |
| a9ac10f2239f4885817f40993511d20b | tec | True |
+-----+-----+-----+
```

- ii. Obtenga el identificador del tenant que desea eliminar y ejecute el siguiente comando para eliminarlo:

```
# keystone tenant-delete a9ac10f2239f4885817f40993511d20b
```

Nota: Reemplace **a9ac10f2239f4885817f40993511d20b** con el identificador del tenant que desea eliminar.

- iii. Corrobore que el tenant ha sido eliminado de manera satisfactoria utilizando nuevamente el comando **keystone tenant-list**.

- d) Si desea eliminar un usuario:

- i. Utilice el siguiente comando para obtener la lista de usuarios existentes y el identificador del usuario que desea eliminar:

```
# keystone user-list
```

```

root@controller:/tmp/imagenes# keystone user-list
+-----+-----+-----+-----+
| id | name | enabled | email |
+-----+-----+-----+-----+
| e8a6038dd3214ebc977aafa0359a65d9 | admin | True | jenni.tec@gmail.com |
| 1ae3fd3603854bbcaf4bf93569c721 | demo | True | jenni.tec@gmail.com |
| 0c0367b5ed7a4b9eb5e0d98a4fb36fea | glance | True | |
| 4e0548285f6541148227dba0b1356d1d | nova | True | |
| 8e2779d920084609a298c45d3c02fb4b | tec | True | jvargas@softwarelibre.org |
+-----+-----+-----+-----+

```

- ii. Elimine el usuario deseado utilizando el siguiente comando:

```
# keystone user-delete 8e2779d920084609a298c45d3c02fb4b
```

Nota: Reemplace **8e2779d920084609a298c45d3c02fb4b** con el identificador del usuario que desea eliminar.

- iii. Corrobore que el usuario ha sido eliminado de manera satisfactoria utilizando nuevamente el comando **keystone user-list**.

4.3 Gestión de redes virtuales en OpenStack

Para empezar a crear instancias virtuales es necesario generar la infraestructura de la red virtual a la que se van a conectar esas instancias.

Como parte de esta solución se utiliza una red de tipo Multi-Host que será compartida por todas las instancias virtuales creadas en OpenStack, independientemente del tenant o usuario que la cree. Típicamente, esta red provee acceso a Internet a las máquinas virtuales mediante el uso de una IP en una red con acceso a Internet y una serie de reglas de seguridad. El tenant llamado **admin** es su dueño porque provee acceso externo a la red de muchos otros tenants. Además, comparte la misma subnet asociada con la red física a la que se conecta la tarjeta de red física en el nodo de virtualización. Es necesario que se especifique el rango de red que será usado por las instancias virtuales para evitar interferencia con otros dispositivos en la red externa.

Si cuenta con más tarjetas físicas con acceso externo, puede crear más redes virtuales en otras redes de acceso externo.

En caso de querer aislar las máquinas virtuales y asignarles distintas subnets a otras máquinas virtuales de distintos tenants/proyectos, puede valorar el uso y configuración de VLAN

networking. Para ello, debe contar, en su infraestructura, con un switch que soporte VLANeo. Los detalles de este tipo de configuración no se encuentran cubiertos en esta solución.

Esta operación se efectúa desde línea de comandos en el nodo controlador.

4.3.1 Crear redes

Siga las instrucciones que se detallan a continuación para la crear la red virtual. Los comandos se deben ejecutar en el nodo controlador.

Note que este proceso ya fue realizado durante el procedimiento de implementación de OpenStack y deberá repetirlo **solamente** en caso de que deba agregar más redes virtuales mediante otros dispositivo de red.

- a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace <usuario> y <controller> por los valores correspondientes.

- b) Cargue las variables de ambiente del tenant y usuario **admin** de Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Cree la red mediante el comando que se muestra a continuación:

```
# nova network-create virt-net --bridge br100 --multi-host T \
--fixed-range-v4 192.168.1.0/24
```

Nota: reemplace **192.168.1.0/24** con la subnet asociada a la red externa de la red física. Este comando no produce ninguna salida. Además, puede modificar el nombre de la red (virt-net) por el nombre que guste utilizar para identificarla.

- d) Verifique que la red ha sido creada de manera exitosa.

```
# nova net-list
```

4.3.2 Eliminar redes

Para eliminar una red virtual siga las instrucciones que se detallan a continuación:

- a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace **<usuario>** y **<controller>** por los valores correspondientes.

- b) Cargue las variables de ambiente del tenant y usuario **admin** de Keystone:

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Obtenga el identificador de la red que desea eliminar del sistema. Use el siguiente comando:

```
# nova net-list
```

```
root@controller:/tmp/imagenes# source /var/lib/keystone/admin-openrc.sh
root@controller:/tmp/imagenes# nova net-list
+-----+-----+-----+
| ID                | Label  | CIDR          |
+-----+-----+-----+
| d1e5410b-3641-4cc2-96ae-ab528d7399be | demo-net | 192.168.1.32/27 |
+-----+-----+-----+
```

- d) Para eliminar una red ejecute el siguiente comando:

```
# nova network-delete d1e5410b-3641-4cc2-96ae-ab528d7399be
```

Nota: reemplace **d1e5410b-3641-4cc2-96ae-ab528d7399be** con el identificador de la red que desea eliminar.

4.4 Creación y mantenimiento de imágenes en Glance

Para la creación de máquinas virtuales es importante contar con un catálogo de imágenes que puedan ser utilizadas en su instalación.

Primeramente, asegúrese de no exceder el espacio disponible en el sistema de ficheros `/var/lib/glance/images` del nodo controlar. En caso de necesitar más espacio puede extender dicho sistema de ficheros.

En segundo lugar, debe descargar o crear las imágenes que quiere utilizar en sus máquinas virtuales. Afortunadamente, hay una serie de repositorios en Internet con varias imágenes disponibles para sistemas en la nube que podrá utilizar.

En el siguiente enlace web podrá consultar sobre las imágenes disponibles y cómo obtenerlas: http://docs.openstack.org/image-guide/content/ch_obtaining_images.html

4.4.1 Creación de imágenes en Glance

Este proceso se puede llevar a cabo de manera manual, utilizando la línea de comandos desde el nodo controlador o mediante Horizon. A continuación, se detallan ambos métodos.

4.4.1.1 Creación de imágenes utilizando línea de comandos

A manera de ejemplo, descargue una de las imágenes disponibles en los repositorios de Ubuntu, Ubuntu server 14.04 de 64 bits.

- a) Inicie sesión en el nodo controlador.
- b) Identifique la imagen que quiere añadir en su sistema y descárguela.

```
# mkdir /tmp/imagenes
# cd /tmp/imagenes
# wget http://cloud-images.ubuntu.com/trusty/20141113.1/trusty-server-cloudimg-amd64-disk1.img
```

- c) Cargue las variables de entorno del tenant y usuario admin de Keystone.

```
# source /var/lib/keystone/admin-openrc.sh
```

- d) Suba la imagen descargada a Glance. Para ello, utilice el comando que aparece en el recuadro. Note que esta imagen podrá ser utilizada por cualquier proyecto y/o usuario, ya que la ha agregado como pública.

```
# glance image-create --name "trusty-server-cloudimg-amd64" --file trusty-server-cloudimg-amd64-disk1.img --disk-format qcow2 --container-format bare --is-public True --progress
```

```

root@controller:/tmp/imagenes# glance image-create --name "trusty-server-cloudimg-amd64" --file trusty-server-cloudimg-amd64-disk1.img --disk-format qcow2 --container-format bare --is-public True --progress
=====>] 100%
-----+-----+
Property | Value
-----+-----+
checksum | 31a60ca8cb67194b3a731bf62d6ec719
container_format | bare
created_at | 2014-11-15T20:29:50
deleted | False
deleted_at | None
disk_format | qcow2
id | a634fb10-28a0-4e05-ae13-bd2d30e45dc0
is_public | True
min_disk | 0
min_ram | 0
name | trusty-server-cloudimg-amd64
owner | 5a0d3c1c49cb4e67aada421c3b26539f
protected | False
size | 255853056
status | active
updated_at | 2014-11-15T20:29:51
virtual_size | None
-----+-----+

```

Nota: modifique los valores en **rojo** según corresponda.

- e) Para confirmar que la imagen ha sido subida a Glance ejecute el siguiente comando. En este caso hay dos imágenes; una de ellas es la que acabamos de subir.

```
# glance image-list
```

```

root@controller:/tmp/imagenes# glance image-list
-----+-----+
ID | Name | Disk Format | Container Format
-----+-----+
57b9849d-2ec1-4f0e-9094-5f1949a84869 | cirros-0.3.3-x86_64 | qcow2 | bare
a634fb10-28a0-4e05-ae13-bd2d30e45dc0 | trusty-server-cloudimg-amd64 | qcow2 | bare
-----+-----+

```

- f) Elimine el directorio temporal creado durante este proceso.

```
# rm -r /tmp/imagenes
```

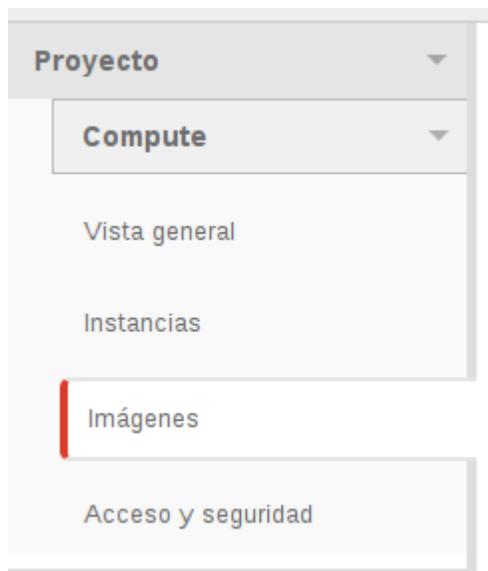
- g) La imagen está lista para que se utilice en alguna de las máquinas virtuales que se serán creadas. Repita este proceso para cualquier otra imagen que desee agregar a Glance.

4.4.1.2 Creación de imágenes utilizando Horizon

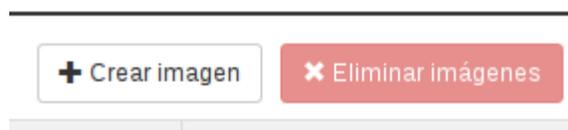
El proceso para agregar imágenes en Horizon es sumamente sencillo. En primer lugar, identifique la imagen que desea agregar en el sistema.

Tome en cuenta que en este ejemplo vamos a utilizar una de las imágenes de Cloud de Fedora. Haga clic en el vínculo para hallarla.
http://download.fedoraproject.org/pub/fedora/linux/updates/20/Images/x86_64/Fedora-x86_64-20-20140407-sda.qcow2

- Abra Horizon desde su navegador. Utilice la dirección `http://controller/horizon`. Recuerde que **controller** es el nombre o dirección IP del nodo controlador.
- Inicie sesión utilizando el usuario demo así como sus credenciales.
- Una vez dentro, diríjase a "Proyecto → Compute → Imágenes".



- Haga clic en el botón "Crear Imagen", que se encuentra en la esquina superior derecha.



- e) Se abrirá una ventana en la cual se solicitan todos los datos de la nueva imagen:
- i. Escriba el nombre con el que desea identificar esta imagen en Glance, así como una descripción.

Nombre ***Descripción**

- ii. Ahora será necesario proporcionar la dirección desde la cual se puede descargar la imagen. Seleccione "Ubicación de la imagen" en el campo "Origen de la imagen" o "Fichero de imagen" si se encuentra en su computadora. En este caso usaremos la opción de descarga directa desde los repositorios de Fedora.

Origen de la imagen**Ubicación de la imagen ?**

- iii. Elija el formato de la imagen. Recuerde, las imágenes de tipo cloud tienen el formato qcow2.

Formato *

- iv. Coloque la arquitectura de la imagen, en este caso es de 64 bits.

Arquitectura

- v. Proporcione los requerimientos mínimos para esta instalación. Usualmente, se encuentran disponibles en el sitio web del fabricante. En este caso, se necesita un mínimo de 10GB de disco duro y 1GB de memoria.

Disco mínimo (GB) ?

Memoria RAM mínima (MB) ?

- vi. Seleccione "Público" si quiere que esta imagen esté disponible para todos los usuarios o proyectos o "Protegido" si solo debe estar disponible para el usuario admin. En este caso, crearemos una imagen pública:

- Público
- Protegido

- vii. Cuando todos los datos estén completos, haga clic sobre el botón "Crear Imagen".

- viii. Inmediatamente, se desplegará la lista de imágenes disponibles en el sistema, allí podrá ver la nueva imagen agregada y su estado. Cuando finalice la creación de la imagen, su estado será "Active".

Imágenes

Proyecto (0)
Compartido conmigo (0)
Público (3)
+ Crear imagen
✖ Eliminar imagen

| <input type="checkbox"/> | Nombre de la imagen | Tipo | Estado | Público | Protegido | Formato | Tamaño | Actions |
|--------------------------|--|-------|--------|---------|-----------|---------|----------|---------------------------------------|
| <input type="checkbox"/> | Fedora-x86_64-20-20140407 | Image | Active | Sí | no | QCOW2 | 201,1 MB | <input type="button" value="Lanzar"/> |
| <input type="checkbox"/> | trusty-server-cloudimg-amd64 | Image | Active | Sí | no | QCOW2 | 244,0 MB | <input type="button" value="Lanzar"/> |

- f) La imagen está lista para ser utilizada. Repita este proceso para agregar imágenes adicionales en OpenStack utilizando Horizon.

4.4.2 Eliminación de imágenes en Glance

4.4.2.1 Eliminar imágenes utilizando línea de comandos

Para eliminar imágenes de manera manual debe ingresar al nodo controlador y cargar las variables de ambiente del tenant y usuario admin.

- a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace <usuario> y controller con los valores correspondientes.

- b) Cargue la variable de entorno del usuario **admin** utilizando el script correspondiente.

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Obtenga la lista de imágenes disponibles en el sistema.

```
# glance image-list
```

```
root@controller:/tmp/imagenes# glance image-list
-----+-----+-----+-----+
| ID                               | Name                               | Disk For
Size      | Status |                               |
-----+-----+-----+-----+
| 57b9849d-2ec1-4f0e-9094-5f1949a84869 | cirros-0.3.3-x86_64               | qcow2
13200896 | active |
| 1dfe7f7e-bda2-4424-9926-e7b71ddfc61d | Fedora-x86_64-20-20140407        | qcow2
210829312 | active |
| 7b5722f9-77ed-45c3-8445-24da1c6eb632 | remove-snapshot-image            | qcow2
19398656 | active |
| a634fb10-28a0-4e05-ae13-bd2d30e45dc0 | trusty-server-cloudimg-amd64     | qcow2
255853056 | active |
-----+-----+-----+-----+
root@controller:/tmp/imagenes#
```

- d) Elimine la imagen deseada utilizando el siguiente comando:

```
# glance image-delete remove-snapshot-image
```

Nota: este comando no produce ninguna salida.

- e) Corrobore que la imagen ha sido eliminada de Glance con el comando **glance image-list**.

```

root@controller:/tmp/imagenes# glance image-delete remove-snapshot-image
root@controller:/tmp/imagenes# glance image-list
-----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID                               | Name                               | Disk Format | Container Form |
| Size                             | Status                             |             |                 |
-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 57b9849d-2ec1-4f0e-9094-5f1949a84869 | cirros-0.3.3-x86_64               | qcow2      | bare           |
| 13200896                             | active                             |             |                 |
| 1dfe7f7e-bda2-4424-9926-e7b71ddfc61d | Fedora-x86_64-20-20140407        | qcow2      | bare           |
| 210829312                             | active                             |             |                 |
| a634fb10-28a0-4e05-ae13-bd2d30e45dc0 | trusty-server-cloudimg-amd64     | qcow2      | bare           |
| 255853056                             | active                             |             |                 |
-----+-----+-----+-----+-----+-----+-----+-----+-----+
root@controller:/tmp/imagenes#
    
```

4.4.2.2 Eliminar imágenes utilizando Horizon

Para eliminar imágenes desde Horizon siga las instrucciones que verá a continuación:

- a) Abra Horizon desde su navegador. Para ello, utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario admin.
- b) Vaya a **"Proyecto → Compute → Imágenes"**.
- c) Seleccione la imagen que desea eliminar y haga clic sobre el botón "Eliminar imágenes" que se ubica en la esquina superior derecha de la ventana.

Imágenes Proyecto (0) Compartido conmigo (0) Público (3) + Crear imagen ✖ Eliminar imágenes

| <input type="checkbox"/> | Nombre de la imagen | Tipo | Estado | Público | Protegido | Formato | Tamaño | Actions |
|-------------------------------------|--|-------|--------|---------|-----------|---------|----------|---------------------------------------|
| <input type="checkbox"/> | Fedora-x86_64-20-20140407 | Image | Active | Sí | no | QCOW2 | 201,1 MB | <input type="button" value="Lanzar"/> |
| <input type="checkbox"/> | trusty-server-cloudimg-amd64 | Image | Active | Sí | no | QCOW2 | 244,0 MB | <input type="button" value="Lanzar"/> |
| <input checked="" type="checkbox"/> | cirros-0.3.3-x86_64 | Image | Active | Sí | no | QCOW2 | 12,6 MB | <input type="button" value="Lanzar"/> |

- d) Se mostrará una ventana emergente en la cual se le solicitará su confirmación. Presione clic sobre el botón "Eliminar imágenes" para eliminar la que ha seleccionado o haga clic sobre el botón "Cancelar" para abortar esta acción.

Confirm Eliminar imágenes

You have selected "cirros-0.3.3-x86_64". Please confirm your selection. This action cannot be undone.

- e) Después de unos segundos, la imagen no aparecerá más en la lista de imágenes disponibles en Horizon.

4.5 Gestión de grupos de seguridad en OpenStack

Cada tenant posee sus propios grupos de seguridad que pueden ser creados para asignarlos a las instancias de máquinas virtuales que se crean en determinado tenant/proyecto.

Los grupos de seguridad permiten la creación de reglas de seguridad mediante Iptables. Estas reglas se crean en el nodo de virtualización para proteger el acceso a las instancias de máquinas virtuales.

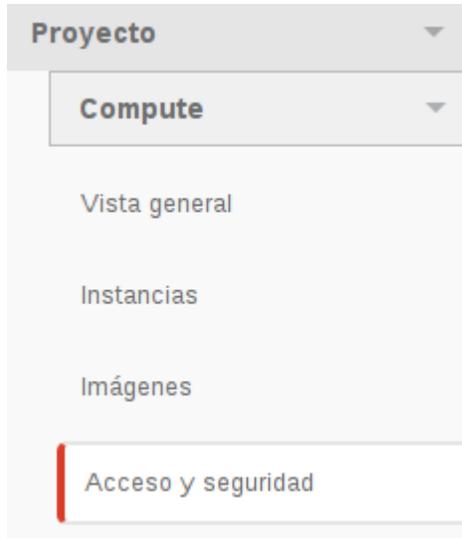
Al igual que en cualquier otro firewall, se pueden crear grupos de seguridad en los cuales se agregan reglas para restringir el tráfico de ciertos protocolos. Por defecto, OpenStack crea un grupo de seguridad llamado **default**, que no permite ningún tipo de acceso a las máquinas virtuales. Ese grupo se puede modificar como se explica en el apartado de gestión de máquinas virtuales, que se encuentra en la sección "**4.6.2 Configuración de las políticas de los grupos de seguridad para habilitar acceso remoto a las máquinas virtuales**". Además, es posible crear grupos de seguridad nuevos, los cuales tienen la capacidad de asociarse a cada máquina virtual y, de este modo también pueden contar con distintas reglas de seguridad para cada instancia virtual si así se desea.

A continuación, se brindan las instrucciones para crear y eliminar nuevos grupos de seguridad mediante Horizon.

4.5.1 Crear de grupos de seguridad utilizando Horizon

Para agregar un nuevo grupo de seguridad a su proyecto, tenga en cuenta las siguientes instrucciones:

- a) Abra Horizon desde su navegador. Para ello, utilice la dirección `http://controller/horizon` e inicie sesión con los credenciales del usuario correspondiente al tenant que desea modificar. En este caso, utilizaremos el usuario demo, asociado al tenant con el mismo nombre.
- b) Diríjase a **“Proyecto → Compute → Acceso y Seguridad”**.



- c) Para crear un nuevo grupo de seguridad haga clic sobre el botón “Crear grupo de seguridad”, ubicado en la esquina superior izquierda de la ventana.



- d) Se desplegará una ventana en la cual debe escribir el nombre que desee asignarle al nuevo grupo de seguridad y una descripción de su función. En este caso, a manera de ejemplo, vamos a crear un grupo de seguridad para un servidor web. Una vez digitados ambos datos, haga clic sobre el botón “Crear grupo de seguridad”, esto agregará un grupo de seguridad sin reglas, como se puede apreciar en el punto siguiente.

Crear grupo de seguridad

Nombre *

Descripción:

Los grupos de Seguridad son conjuntos de reglas de filtros de IP que son aplicadas a la configuración de red para la máquina virtual. Después de que el grupo de seguridad es creado, puedes agregar reglas al grupo de seguridad.

Descripción *

- e) Una vez creado el nuevo conjunto, aparecerá en la lista de grupos de seguridad existentes en este proyecto. Para agregarle reglas, haga clic sobre el botón “Administrar Reglas” ubicado al final de la fila del grupo de seguridad correspondiente y siga estas instrucciones:

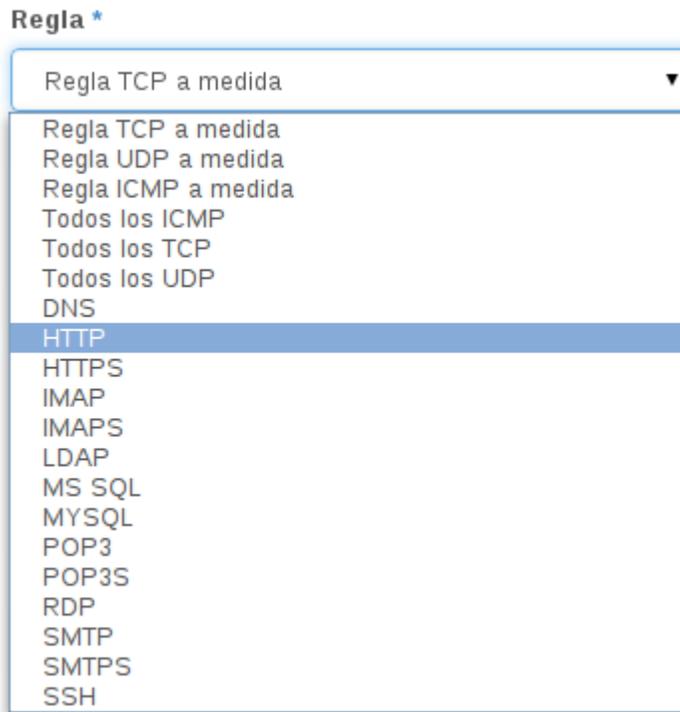
| | | | |
|-------------------------------------|-----------------------|-----------------------|----------------------|
| <input checked="" type="checkbox"/> | Nuevo-grupo-seguridad | Utilizado por VMs web | Administrar reglas ▾ |
|-------------------------------------|-----------------------|-----------------------|----------------------|

- i. El sistema lo redirigirá a una página donde podrá agregar reglas al grupo de seguridad. Haga clic sobre el botón “Agregar regla” ubicado en la esquina superior derecha.

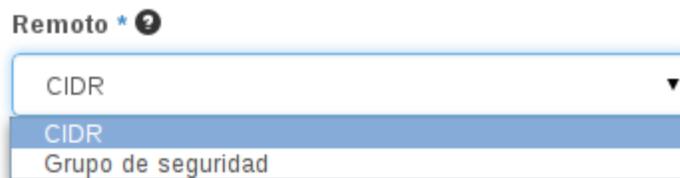
Reglas del grupo de seguridad

| | Dirección | Tipo Ethernet | Protocolo IP | Rango de puertos | Remoto | Actions |
|----------------------|-----------|---------------|--------------|------------------|--------|---------|
| No items to display. | | | | | | |
| Displaying 0 items | | | | | | |

- ii. Se abrirá una ventana para agregar la nueva regla. Además, existe la opción de agregar una regla de manera manual mediante las opciones “Regla TCP a medida”, “Regla UDP a medida” o “Regla ICMP a medida”. También, se encuentra disponible una lista de reglas predefinidas para los distintos tipos de servicios. En este caso utilizaremos la regla HTTP requerida para máquinas virtuales que proporcionen un servicio web.



- iii. Es necesario definir la fuente o entrada para permitir el tráfico web a nuestras máquinas virtuales. Existen dos opciones para especificar el tráfico de entrada: 1) mediante el uso de un bloque de direcciones IP (CIDR) donde limitamos el tráfico a ese grupo de direcciones y 2) mediante la opción "Grupo de seguridad". Al seleccionar esta opción le estamos indicando al sistema que permita el tráfico red solo a las máquinas que pertenezcan al mismo grupo de seguridad. En el siguiente ejemplo, vamos a utilizar un bloque de direcciones IP.



- iv. En el campo CIDR debe especificar la red a la que quiere proporcionar acceso al tráfico red. Para permitir el tráfico a cualquier red puede mantenerla en 0.0.0.0/0. Sin embargo, esta política no es recomendada por motivos de seguridad; en su lugar, es preferible que especifique una red.

CIDR 

0.0.0.0/0

- v. Una vez completados todos los campos, haga clic sobre el botón “Añadir”, ubicado en la esquina inferior derecha, para agregar la nueva regla al grupo de seguridad, o bien haga clic en “Cancelar” para no llevar a cabo esta acción.

Cancelar

Añadir

- vi. Repita los pasos de “i” a “v” para agregar más reglas de seguridad al grupo de seguridad creado.
- f) Repita los pasos de “a” a “g” para crear más grupos de seguridad en su proyecto/tenant.

4.5.2 Eliminación de grupos de seguridad utilizando Horizon

Para eliminar un grupo de seguridad existente en su proyecto, siga las instrucciones que se muestran a continuación:

- a) Abra Horizon desde su navegador. Para ello, utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario correspondiente al tenant que desea modificar. En este caso, utilizaremos el usuario demo, asociado al tenant con el mismo nombre.
- b) Vaya a “**Proyecto** → **Compute** → **Acceso y Seguridad**”.
- c) Se desplegará una lista con todos los grupos de seguridad existentes. Para eliminar un grupo de seguridad marque el “check” que se encuentra al inicio de la fila del grupo de seguridad correspondiente y haga clic sobre el botón “Borrar grupos de seguridad”, ubicado en la esquina superior derecha de la ventana.

| Grupos de seguridad | | | + Crear grupo de seguridad | ✖ Borrar grupos de seguridad |
|-------------------------------------|--------------------------|-----------------------|--|--|
| <input type="checkbox"/> | Nombre | Descripción | Actions | |
| <input checked="" type="checkbox"/> | Nuevo-grupo-de-seguridad | Utilizado por VMs web | Administrar reglas ▾ | |

- d) Se abrirá una nueva ventana en la cual se le pedirá confirmar. Haga clic sobre "Borrar grupos de seguridad" para llevar a cabo esta acción o en "Cancelar" para abortarla.

Confirm Borrar grupos de seguridad

You have selected "Nuevo-grupo-de-seguridad". Please confirm your selection. This action cannot be undone.

Borrar grupos de seguridad

Cancel

- e) Repita este proceso para eliminar otros grupos de seguridad.

4.6 Gestión de una instancia de máquina virtual

En esta sección podrá encontrar los detalles necesarios para la gestión de máquinas virtuales en OpenStack, ya sea mediante el uso de línea de comandos, o bien del portal web llamado Horizon.

Asimismo, encontrará cómo crear, modificar y eliminar instancias de máquinas virtuales desde el sistema llamado Nova.

4.6.1 Definición de las plantillas de las máquinas virtuales

En OpenStack hay plantillas llamadas **sabores**, las cuales se utilizan para definir las propiedades o atributos de una máquina virtual, por ejemplo: CPU virtuales, memoria virtual y tamaño del disco duro. Algunas de estas plantillas están predefinidas, como se muestra a continuación.

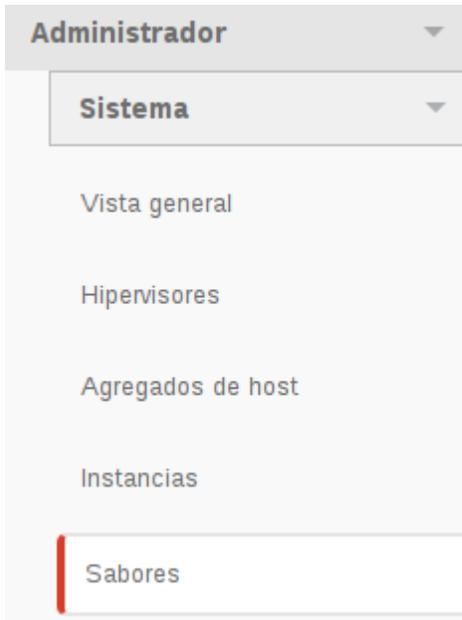
Sabores

| <input type="checkbox"/> | Nombre del sabor | VCPU | RAM | Disco raíz | Disco efímero | Disco de intercambio (swap) | ID | Público | Metadatos | Actions |
|--------------------------|------------------|------|---------|------------|---------------|-----------------------------|----|---------|-----------|----------------|
| <input type="checkbox"/> | m1.tiny | 1 | 512MB | 1GB | 0GB | 0MB | 1 | Sí | no | Editar sabor ▼ |
| <input type="checkbox"/> | m1.small | 1 | 2048MB | 20GB | 0GB | 0MB | 2 | Sí | no | Editar sabor ▼ |
| <input type="checkbox"/> | m1.medium | 2 | 4096MB | 40GB | 0GB | 0MB | 3 | Sí | no | Editar sabor ▼ |
| <input type="checkbox"/> | m1.large | 4 | 8192MB | 80GB | 0GB | 0MB | 4 | Sí | no | Editar sabor ▼ |
| <input type="checkbox"/> | m1.xlarge | 8 | 16384MB | 160GB | 0GB | 0MB | 5 | Sí | no | Editar sabor ▼ |

Displaying 5 items

Se pueden crear nuevas plantillas que se adapten a sus necesidades. Esto lo podemos lograr de la siguiente manera:

- f) Abra Horizon desde su navegador. Utilizando la dirección <http://controller/horizon>, inicie sesión con los credenciales del usuario admin.
- g) Diríjase a **“Administrador → Sistema → Sabores”**.



- h) Podrá ver la lista de sabores o plantillas disponibles en el sistema. Para crear un nuevo sabor pulse clic sobre el botón “Crear sabor”, ubicado en la esquina superior derecha.



- i) Complete los datos con las características de la nueva plantilla que desea crear y pulse clic sobre el botón “Crear sabor”, ubicado en la esquina inferior derecha de la ventana.

Información del sabor *
Acceso al sabor

Nombre *

ID [?]

VCPU *

RAM (MB) *

Disco raíz (GB) *

Disco efímero (GB) *

Disco de intercambio (MB) *

Los sabores definen los tamaños de memoria RAM, disco, número de cores y otros recursos que pueden ser seleccionados por los usuarios al desplegar instancias.

j) La nueva plantilla se desplegará en la lista de las existentes en el sistema.

| | | | | | | | | | | |
|--------------------------|--------|---|-------|------|-----|-------|--------------------------------------|----|----|----------------|
| <input type="checkbox"/> | VM-Peq | 1 | 512MB | 10GB | 1GB | 512MB | ffa13b06-bd73-4475-b68f-c23f55e5a18f | Sí | no | Editar sabor ▾ |
|--------------------------|--------|---|-------|------|-----|-------|--------------------------------------|----|----|----------------|

k) Repita este proceso para crear nuevos sabores o plantillas de máquinas virtuales.

4.6.2 Configuración de las políticas de los grupos de seguridad para habilitar acceso remoto a las máquinas virtuales

Por defecto, las políticas del grupo de seguridad **default** de OpenStack restringen el acceso a las máquinas virtuales por medio de cualquier protocolo. La única manera de conectarse a una máquina virtual es mediante el uso de la consola que utiliza VNC.

Afortunadamente, esto se puede modificar. Recordemos que contamos con dos usuarios en Keystone para ingresar a OpenStack (**admin** y **demo**). En este caso vamos a utilizar el usuario **demo** para crear las máquinas virtuales, por lo que debemos cargar sus variables de entorno. Asimismo, vamos a modificar su grupo de seguridad para permitir acceso a SSH y a ICMP (ping). Posteriormente, podremos crear una máquina virtual y asignar este grupo de seguridad.

f) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace **<usuario>** y **controller** con los valores correspondientes.

g) Cargue las variables de entorno del usuario **demo** utilizando el script correspondiente.

```
# source /var/lib/keystone/demo-openrc.sh
```

h) Modifique las políticas del grupo de seguridad para habilitar ICMP y SSH desde cualquier red. Utilice los siguientes comandos:

```
# nova secgroup-add-rule default icmp -1 -1 0.0.0.0/0
# nova secgroup-add-rule default tcp 22 22 0.0.0.0/0
```

4.6.3 Creación/Importación de llaves para asignar a las máquinas virtuales

Una vez creada una máquina virtual, necesitará credenciales para iniciar sesión. Usualmente, la imagen a utilizar provee detalles sobre cómo iniciar sesión luego de instalado el sistema operativo, pero OpenStack ofrece, además, la posibilidad de proporcionar una llave pública/privada para iniciar sesión desde el nodo de virtualización (o algún otro nodo que pueda tener contacto con la red de las máquinas virtuales).

Esta llave se puede crear por medio de Horizon o de línea de comandos, como mostraremos a continuación.

En seguida, observará los detalles sobre la creación de una llave en Openstack.

a) Primeramente, cree la llave SSH desde el servidor y usuario deseados. Recuerde asegurarse de que el servidor seleccionado tenga acceso a la red virtual que será

asignada a las máquinas virtuales. En este caso utilizaremos el servidor de virtualización y el usuario root para generar la llave SSH.

- i. Inicie sesión en el nodo de virtualización y hágase root.

```
# ssh <usuario>@compute
$ sudo su -
```

Nota: reemplace **<usuario>** y **compute** con los valores correspondientes.

- ii. Cree la llave SSH utilizando el siguiente comando:

```
# ssh-keygen
```

```
root@compute:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
4b:79:a6:26:ad:76:a2:65:cd:9b:c9:81:0a:00:2d:62 root@compute
The key's randomart image is:
+--[ RSA 2048 ]-----+
|
|+E.
|+.
| .          S o
| .          * =
| .          = 0
| .          +0=.
| .          oo.o=
|
+-----+
root@compute:~#
```

4.6.3.1 Importar una llave en OpenStack desde línea de comandos

- a) Copie la llave pública desde el nodo de virtualización al nodo controlador para agregarla a OpenStack.

```
# scp ~/.ssh/id_rsa.pub controller:/tmp/llaveroot_compute.pub
```

```

root@compute:~# scp ~/.ssh/id_rsa.pub controller:/tmp/llaveroot_compute.pub
The authenticity of host 'controller (192.168.0.11)' can't be established.
ECDSA key fingerprint is 5d:63:a4:ff:51:ea:18:73:32:33:29:9f:f8:ce:59:5b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'controller,192.168.0.11' (ECDSA) to the list of known
hosts.
root@controller's password:
id_rsa.pub                                100% 394      0.4KB/s   00:00
root@compute:~#

```

b) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace <usuario> y controller con los valores correspondientes.

c) Cargue las variables de entorno del usuario **demo** utilizando el script correspondiente.

```
# source /var/lib/keystone/demo-openrc.sh
```

d) Cree la llave SSH en el nodo de virtualización. Emplee el usuario de su preferencia y copie dicha llave.

```
# nova keypair-add --pub-key /tmp/llaveroot_compute.pub compute-key
```

Nota: puede reemplazar **compute-key** por uno nombre de su preferencia.

e) Al utilizar el siguiente comando, puede corroborar que la llave ha sido creada. Tome nota del nombre de la llave, pues va a necesitarlo para la creación de la máquina virtual.

```
# nova keypair-list
```

```

root@controller:~# nova keypair-list
-----+-----+-----+
Name      | Fingerprint |
-----+-----+-----+
compute-key | 4b:79:a6:26:ad:76:a2:65:cd:9b:c9:81:0a:00:2d:62 |
demo-key   | 4d:ec:be:1d:33:66:ce:bf:29:88:71:e5:2b:d3:03:e9 |
-----+-----+-----+
root@controller:~#

```

f) Repita este proceso para agregar otras llaves SSH al sistema.

4.6.3.2 Importar una llave en OpenStack utilizando Horizon

El proceso de crear llaves en OpenStack mediante Horizon es sencillo.

- a) Inicie sesión en el nodo de virtualización.

```
# ssh <usuario>@compute
$ sudo su -
```

Nota: reemplace **<usuario>** y **compute** con los valores correspondientes.

- b) Obtenga el contenido de la llave pública generada anteriormente y cópielo.

```
# cat ~/.ssh/id_rsa.pub
```

- c) Abra Horizon desde su navegador. Utilice la dirección <http://<controller>/horizon> e inicie sesión con los credenciales del usuario demo.
- d) Diríjase a **"Proyecto → Compute → Acceso y Seguridad"**.
- e) Seleccione la pestaña "Pares de claves", posteriormente, haga clic sobre el botón "Importar par de claves", ubicado en la esquina superior derecha de la pantalla.

Grupos de seguridad Pares de claves IPs flotantes Acceso a la API

Pares de claves

+ Crear p

| <input type="checkbox"/> | Nombre de par de claves | Fingerprint |
|---|-------------------------|-------------|
| <input type="button" value="Importar par de claves"/> | | |

- f) Escriba el nombre que quiere asignarle a la nueva llave y pegue el contenido de la llave pública que copió en el paso **"b"**.

Nombre de par de claves *

Clave pública *

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQBAQDZ3soJ
N294/d4XNQRWdH08cunhl2AIMR3im8O8Zlpxcne8
2kwixb7+ZVulZ9X1iu6/Jl5mKWa+WlJ1oO4jdWNSC
n0kXoFt7leVp1AdRnpLJ/k6wKp1Ufgv0WOBE8yaJA
1Fslp5VhDWyW4fbDtVOA0Rq1tpwsxPj08p2IkzRO/
AHRgGqR3c7kWOMyGP+mCgK0KBGqrQQ2hmM9
SI9P3BCj6Tf6DU4M1JIWxn/CNhKgGa/JmkN73BfO
KWmSO1fd08xpnRGNmLlq0EQq0ZDs/oXbbSpN/BQ
5IV5mVMm4Qrt2T/5vXKrF36+Gw8YXq/7xeTArNI5E
```

g) Seguidamente, pulse clic sobre el botón “Importar par de llaves”.

h) Seguidamente, se desplegará una lista con todas las llaves cargadas en el sistema.

Pares de claves

| <input type="checkbox"/> | Nombre de par de claves | Fingerprint | Actions |
|--------------------------|-------------------------|---|--|
| <input type="checkbox"/> | demo-key | 4d:ec:be:1d:33:66:ce:bf:29:88:71:e5:2b:d3:03:e9 | <input type="button" value="Borrar pareja de claves"/> |
| <input type="checkbox"/> | compute-key | 4b:79:a6:26:ad:76:a2:65:cd:9b:c9:81:0a:00:2d:62 | <input type="button" value="Borrar pareja de claves"/> |
| <input type="checkbox"/> | test | fa:a6:84:77:de:eb:86:28:75:01:e9:a4:31:77:31:9f | <input type="button" value="Borrar pareja de claves"/> |
| <input type="checkbox"/> | compute-key2 | 4b:79:a6:26:ad:76:a2:65:cd:9b:c9:81:0a:00:2d:62 | <input type="button" value="Borrar pareja de claves"/> |

i) Repita este proceso para agregar más llaves en OpenStack.

4.6.4 Creación de máquinas virtuales

Las máquinas virtuales se pueden crear desde línea de comandos o mediante Horizon. En esta sección podrá encontrar los detalles sobre ambos métodos.

4.6.4.1 Creación de máquinas virtuales desde línea de comandos

Para crear una máquina virtual desde línea de comandos, conéctese al nodo controlador y siga las instrucciones que se brindan a continuación:

4.6.4.1.1 Tareas de Preparación

Primeramente, asegúrese de que la red virtual haya sido creada de acuerdo con las instrucciones señaladas en la sección **“8.1.6.3 Creación de la red virtual en nova network”** de la Guía de implementación.

Además, se debe asegurar de que todos los servicios de OpenStack Nova corran de manera satisfactoria en el nodo controlador. Lo anterior puede corroborarlo con el siguiente comando. El estado de todos los servicios debe de ser **“up”**.

```
# source /var/lib/keystone/admin-openrc.sh
# nova service-list
```

```
root@controller:~# source /var/lib/keystone/admin-openrc.sh
root@controller:~# nova service-list
-----+-----+-----+-----+-----+-----+-----+
 Id | Binary           | Host       | Zone   | Status | State | Updated_at |
 Disabled Reason |
-----+-----+-----+-----+-----+-----+-----+
 1  | nova-cert        | controller | internal | enabled | up    | 2014-11-15T19:53:59.000000 |
 2  | nova-consoleauth | controller | internal | enabled | up    | 2014-11-15T19:54:06.000000 |
 3  | nova-scheduler   | controller | internal | enabled | up    | 2014-11-15T19:54:01.000000 |
 4  | nova-conductor   | controller | internal | enabled | up    | 2014-11-15T19:54:05.000000 |
 6  | nova-compute     | compute   | nova    | enabled | up    | 2014-11-15T19:54:06.000000 |
 7  | nova-network     | compute   | internal | enabled | up    | 2014-11-15T19:54:01.000000 |
-----+-----+-----+-----+-----+-----+-----+

```

Para la creación de una máquina virtual es necesario tener a mano los siguientes detalles: nombre o identificador de la red virtual, nombre de la llave SSH a utilizar, nombre de la imagen, sabor (flavor) a utilizar en la máquina virtual. A continuación, podrá leer las instrucciones para

obtener esos datos, los cuales se deben conseguir desde el nodo controlador, a través del tenant/usuario admin.

- a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace **<usuario>** y **controller** con los valores correspondientes.

- b) Cargue las variables de ambiente del usuario admin.

```
# source /var/lib/keystone/admin-openrc.sh
```

- c) Para obtener el identificador de la red virtual utilice el siguiente comando. Tome nota del identificador de esa red.

```
# nova net-list
```

```
root@controller:~# nova net-list
-----+-----+-----+
ID | Label | CIDR |
-----+-----+-----+
d1e5410b-3641-4cc2-96ae-ab528d7399be | demo-net | 192.168.1.32/27 |
-----+-----+-----+
```

- d) Como se mencionó en la sección “**4.6.1 Definición de plantillas de las máquinas virtuales**”, los “flavors” definen el tamaño que tendrá la máquina virtual. Por eso, debe escoger un flavor para la máquina virtual que va a crear. Para obtener la lista de plantillas disponibles en el sistema, ejecute el siguiente comando. Tome nota del nombre del flavor deseado

```
# nova flavor-list
```

```
root@controller:/tmp/imagenes# nova flavor-list
-----+-----+-----+-----+-----+-----+-----+
ID | Name | Memory_MB | Disk | Ephemeral | Swap | VCPUs | RXTX |
-----+-----+-----+-----+-----+-----+-----+
1 | m1.tiny | 512 | 1 | 0 | | 1 | 1.0 |
2 | m1.small | 2048 | 20 | 0 | | 1 | 1.0 |
3 | m1.medium | 4096 | 40 | 0 | | 2 | 1.0 |
4 | m1.large | 8192 | 80 | 0 | | 4 | 1.0 |
5 | m1.xlarge | 16384 | 160 | 0 | | 8 | 1.0 |
ffa13b06-bd73-4475-b68f-c23f55e5a18f | VM-Peq | 512 | 10 | 1 | 512 | 1 | 1.0 |
-----+-----+-----+-----+-----+-----+-----+
Is_Public |
-----+-----+-----+-----+-----+-----+
True |
-----+-----+-----+-----+-----+-----+-----+
```

- e) Obtenga la lista de llaves SSH cargadas en el sistema. Necesitará el nombre de la llave deseada durante el proceso de creación de la máquina virtual.

```
# nova keypair-list
```

```
root@controller:/tmp/imagenes# nova keypair-list
-----+-----+
Name          | Fingerprint          |
-----+-----+
controller-root-key | 4d:ec:be:1d:33:66:ce:bf:29:88:71:e5:2b:d3:03:e9 |
-----+-----+
root@controller:/tmp/imagenes#
```

4.6.4.1.2 Creación de la máquina virtual usando línea de comandos

La máquina virtual, desde línea de comandos, se crea de manera sencilla. Para ello, siga las instrucciones que se brindan a continuación:

- a) Cargue las variables de ambiente del tenant y usuario demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- b) Una vez con todos los datos necesarios a mano, cree la máquina virtual utilizando el siguiente comando:

```
# nova boot --flavor m1.tiny --image cirros-0.3.3-x86_64 --nic net-id=d1e5410b-3641-4cc2-96ae-ab528d7399be --security-group default --key-name compute-key vm-cirros
```

Como puede ver, hemos elegido el "flavor" **m1.tiny** que corresponde a una máquina virtual de 1 CPU virtual, 512 MB de memoria y 1GB de disco; la imagen **cirros-0.3.3-x86_64**, que es apta para sistemas pequeños; la red de identificador **d1e5410b-3641-4cc2-96ae-ab528d7399be**, que corresponde a la de nombre demo-net; el grupo de seguridad llamado **default** y la llave SSH llamada compute-key, correspondiente a la llave del nodo de virtualización. El nombre de la máquina virtual, en este caso, es **vm-cirros**.

- c) Inmediatamente después de escribir ese comando, se desplegarán como respuesta las propiedades de la nueva máquina virtual, como se muestra en la siguiente imagen:

```

Property | Value
-----|-----
OS-DCF:diskConfig | MANUAL
OS-EXT-AZ:availability_zone | nova
OS-EXT-STS:power_state | 0
OS-EXT-STS:task_state | scheduling
OS-EXT-STS:vm_state | building
OS-SRV-USG:launched_at | -
OS-SRV-USG:terminated_at | -
accessIPv4 |
accessIPv6 |
adminPass | 3TQkBnmNR8zA
config_drive |
created | 2014-11-16T01:11:53Z
flavor | m1.tiny (1)
hostId |
id | 110790fa-6d64-4349-8934-04ca424cb9e1
image | cirros-0.3.3-x86_64 (57b9849d-2ec1-4f0e-9094-5f1949a84869)
key_name | compute-key
metadata | {}
name | vm-cirros
os-extended-volumes:volumes_attached | []
progress | 0
security_groups | default
status | BUILD
tenant_id | e2e2410014c949e88892454493ed7f72
updated | 2014-11-16T01:11:54Z
user_id | 1ae3fd3603854bbcafcf4bf93569c721
    
```

- d) La máquina tardará varios minutos en el proceso de creación. Para revisar el estado de la máquina puede utilizar el comando que encontrará seguidamente. Durante el proceso de instalación, la máquina se mantendrá en estado "Building" y, una vez finalizada la creación, cambiará al estado "Active".

```
# nova list
```

```

root@controller:/tmp/imagenes# nova list
-----+-----+-----+-----+-----+-----+
 ID | Name | Status | Task State | Power State | Networks
-----+-----+-----+-----+-----+-----+
 25b0d8fe-2ae6-4b5e-a951-413671311b18 | demovmtest | ACTIVE | - | Running | demo-net=192.168.1.35 |
 110790fa-6d64-4349-8934-04ca424cb9e1 | vm-cirros | ACTIVE | - | Running | demo-net=192.168.1.36 |
-----+-----+-----+-----+-----+-----+
root@controller:/tmp/imagenes#
    
```

- e) Como puede apreciar en la imagen anterior, el sistema ha concluido la creación de la máquina virtual y le ha asignado una dirección IP en la red virtual especificada. Desde el nodo de virtualización se puede conectar a la máquina virtual por medio de SSH o consola, así como utilizando los credenciales correspondientes. Más adelante se explicará cómo conectarse a una máquina virtual por medio de SSH o consola.

4.6.4.2 Creación de máquinas virtuales desde Horizon

A continuación se proporcionan las instrucciones para crear máquinas virtuales mediante el dashboard de OpenStack.

- a) Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- b) Diríjase a **“Proyecto → Compute → Instancias”**.



- c) Pulse clic sobre el botón “Lanzar instancia”.



- d) Inmediatamente, se desplegará una ventana con varias opciones. A continuación, siga las instrucciones para crear la máquina virtual:
 - i. En la pestaña llamada “Detalles”, debe proporcionar los detalles básicos para crear la máquina virtual.
 - ii. En el campo “Nombre de la instancia”, coloque el nombre que le quiere asignar a la máquina virtual.

Nombre de la instancia *

Ubuntu-vm

- iii. Seguidamente, seleccione la plantilla (sabor) de la máquina virtual de su preferencia. En este caso, utilizaremos un sabor llamado **VM-Peq**, que cuenta con las características que se muestran a continuación:

Sabor * ?

VM-Peq ▾

- m1.tiny
- VM-Peq
- VM-Med
- m1.small
- m1.medium
- m1.large
- m1.xlarge

| | |
|----------------------|--------|
| Nombre | VM-Peq |
| VCPU | 1 |
| Disco raíz | 10 GB |
| Disco efímero | 1 GB |
| Disco total | 11 GB |
| RAM | 512 MB |

- iv. En "recuento de instancias" puede indicar cuántas máquinas virtuales quiere crear con estas mismas características. En este caso se creará solo una.

Recuento de instancias * ?

1

- v. En el campo "Origen de arranque de la instancia", puede seleccionar si desea instalar la máquina virtual mediante una imagen instantánea; una que se haya creado a partir de otra máquina virtual en el ambiente, o bien si quiere utilizar una imagen normal. En este caso utilizaremos una de las imágenes que fueron creadas en la sección **"4.3.1 Creación de imágenes en Glance"**.

Origen de arranque de la instancia * ?

Arrancar desde una imagen ▼

Escoger origen

Arrancar desde una imagen

Arrancar desde una instantánea

- vi. En la opción "Nombre de la imagen", seleccionaremos la imagen que deseamos utilizar para instalar la máquina virtual, en este caso, la de Ubuntu server:

Nombre de la imagen

trusty-server-cloudimg-amd64 (244,0 MB) ▼

Seleccionar imagen

Fedora-x86_64-20-20140407 (201,1 MB)

cirros-0.3.3-x86_64 (12,6 MB)

trusty-server-cloudimg-amd64 (244,0 MB)

Nota: es imprescindible asegurarse de que el sabor en uso cumpla con los requerimientos mínimos de CPU, memoria y disco de la distribución que desea instalar.

- vii. Pulse clic sobre la pestaña llamada "Acceso y seguridad". En esta seleccionará la llave que desea asignar a la máquina virtual y podrá asociarla con el grupo de seguridad que guste.

Detalles * Acceso y seguridad * Pos-crea

Par de claves * ?

compute-key ▼ +

Seleccione un par de claves

demo-key

compute-key

test

compute-key2

Grupos de seguridad * ?

Nuevo-grupo-de-seguridad

Nuevo-grupo-seguridad

default

viii. Una vez completados todos los datos, pulse clic sobre el botón "Lanzar", que se encuentra en la esquina inferior derecha de la ventana.



ix. Inmediatamente, se desplegará en pantalla una lista con todas las máquinas virtuales existentes en el sistema, una vez que la máquina virtual esté preparada, aparecerá en estado "Activo", tal como se muestra a continuación:

Instancias

Nombre de la instancia ▼ Filter Filtrar Lanzar instancia Instancias de reinicio suave Terminar instancias

| <input type="checkbox"/> | Nombre de la instancia | Nombre de la imagen | Dirección IP | Tamaño | Par de claves | Estado | Zona de disponibilidad | Tarea | Estado de energía | Tiempo desde su creación | Actions |
|--------------------------|------------------------|------------------------------|--------------|---------|---------------|--------|------------------------|-------|-------------------|--------------------------|---------------------|
| <input type="checkbox"/> | ubuntu-vm | trusty-server-cloudimg-amd64 | 192.168.1.37 | VM-PeQ | compute-key | Activo | nova | None | Ejecutando | 4 días, 3 horas | Crear instantánea ▼ |
| <input type="checkbox"/> | vm-cirros | cirros-0.3.3-x86_64 | 192.168.1.36 | m1.tiny | compute-key | Activo | nova | None | Ejecutando | 4 días, 4 horas | Crear instantánea ▼ |

Displaying 2 items

e) La máquina virtual **ubuntu-vm**, está lista para ser utilizada. Repita este proceso para crear otras máquinas virtuales.

4.6.5 Conectarse a la consola de una máquina virtual

Una vez creada una máquina virtual, puede conectarse a la consola. En OpenStack la consola utilizada se basa en VNC.

Para ingresar a la consola es preciso utilizar un navegador web. Se puede realizar directamente desde Horizon u obtener la información desde línea de comandos.

- a) Para obtener los detalles del acceso a la consola de una máquina virtual desde línea de comandos, siga estas instrucciones:
 - i. Inicie sesión en el nodo controlador.
 - ii. Cargue las variables de ambiente del tenant demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- iii. Para obtener los datos de la consola de una máquina virtual debe ejecutar el siguiente comando:

```
# nova get-vnc-console ubuntu-vm novnc
```

Nota: reemplace **ubuntu-vm** con el nombre de la máquina virtual que desee consultar.

- iv. Ese comando le proporcionará una URL que deberá utilizar en el navegador de su preferencia para tener acceso a la consola de la máquina virtual.

```
root@controller:/tmp/imagenes# nova get-vnc-console ubuntu-vm novnc
+-----+
| Type | Url |
+-----+
| novnc | http://controller:6080/vnc_auto.html?token=b4061a73-6bee-405a-984f-c20701a033dd |
+-----+
root@controller:/tmp/imagenes#
```

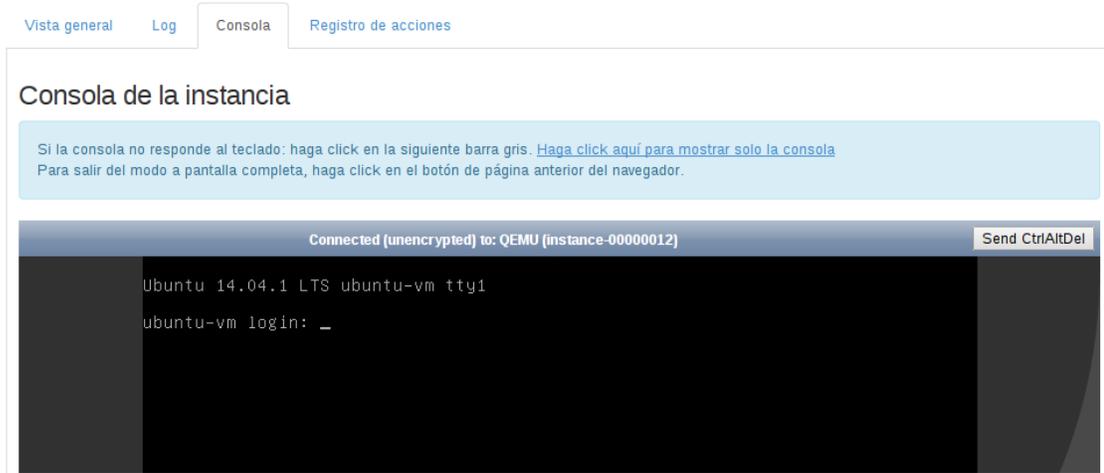
- b) Para ingresar a la consola desde Horizon, siga las instrucciones que se le brindan a continuación:

- i. Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- ii. Diríjase a **“Proyecto → Compute → Instancias”**.
- iii. Se desplegará una lista de máquinas virtuales. Ubique la máquina virtual a la que le desee abrir la consola y pulse clic sobre un “triángulo” de opciones ubicado al final de la fila de la máquina virtual correspondiente. Allí se desplegarán una serie de acciones que puede realizar en esa máquina virtual. Pulse clic sobre “Consola” y con esto será redirigido a la consola de la máquina virtual seleccionada.

| | | | | | | | | | | | |
|--------------------------|-----------|------------------------------|--------------|---------|-------------|--------|------|------|------------|-----------------|---|
| <input type="checkbox"/> | ubuntu-vm | trusty-server-cloudimg-amd64 | 192.168.1.37 | VM-PeQ | compute-key | Activo | nova | None | Ejecutando | 4 días, 3 horas | Crear instantánea Asociar IP flotante Desasociar IP flotante Editar instancia Editar grupos de seguridad Consola |
| <input type="checkbox"/> | vm-cirros | cirros-0.3.3-x86_64 | 192.168.1.36 | m1.tiny | compute-key | Activo | nova | None | Ejecutando | 4 días, 4 horas | |

Displaying 2 items

Detalles de la instancia:ubuntu-vm



- iv. Una vez abierta la consola, solo deberá proporcionar los credenciales para ingresar al servicio.

4.6.6 Conectarse remotamente a una máquina virtual

Para conectarse a una máquina virtual basada en Linux puede utilizar SSH desde el nodo de virtualización, tal como se muestra seguidamente.

- a) Obtenga la dirección IP de la máquina virtual a la que se quiere conectar. Esta información la puede obtener desde Horizon ("Proyecto → Compute → Instancias").

| <input type="checkbox"/> | Nombre de la instancia | Nombre de la imagen | Dirección IP |
|--------------------------|------------------------|------------------------------|--------------|
| <input type="checkbox"/> | ubuntu-vm | trusty-server-cloudimg-amd64 | 192.168.1.37 |

- b) Conéctese al nodo de virtualización y hágase root.

```
# ssh <usuario>@compute
```

```
# sudo su -
```

Nota: reemplace **<usuario>** y **compute** con los valores correspondientes.

- c) Desde el nodo de virtualización y utilizando el usuario root (la llave creada en esta documentación utiliza el usuario root), conéctese por medio de SSH.

```
# ssh 192.168.1.37
```

Nota: reemplace el valor en rojo por la dirección IP de la máquina virtual a la cual se quiere conectar.

- d) Escriba **yes** para confirmar que desea conectarse a la máquina virtual.

```
root@compute:~# ssh 192.168.1.37
The authenticity of host '192.168.1.37 (192.168.1.37)' can't be established.
ECDSA key fingerprint is 7d:2c:f2:9a:12:2d:d1:db:92:d1:68:a1:54:84:2b:75.
Are you sure you want to continue connecting (yes/no)? yes
```

- e) Seguidamente, el sistema le mostrará un mensaje que indica el usuario que debe utilizar para conectarse a la máquina virtual (en lugar de root). A continuación, algunos ejemplos con máquinas virtuales construidas, que utilizan distintas imágenes:

```
root@compute:~# ssh 192.168.1.37
The authenticity of host '192.168.1.37 (192.168.1.37)' can't be established.
ECDSA key fingerprint is 7d:2c:f2:9a:12:2d:d1:db:92:d1:68:a1:54:84:2b:75.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.37' (ECDSA) to the list of known hosts.
Please login as the user "ubuntu" rather than the user "root".

Connection to 192.168.1.37 closed.
root@compute:~#
```

```
root@compute:~# ssh 192.168.1.36
Please login as 'cirros' user, not as root
```

- f) Conéctese nuevamente, pero esta vez utilice el usuario indicado.

```
# ssh <usuario>@192.168.1.37
```

Nota: reemplace el valor en rojo por el nombre del usuario y la dirección IP de la máquina virtual a la cual se quiere conectar.

- g) Como podrá ver, se ha podido ingresar de manera exitosa a la máquina virtual **ubuntu-vm**, creada en secciones anteriores.

```

root@compute:~# ssh ubuntu@192.168.1.37
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-39-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

System information as of Sun Nov 16 03:33:31 UTC 2014

System load:  0.0          Processes:    70
Usage of /:   7.6% of 9.81GB Users logged in:  0
Memory usage: 10%         IP address for eth0: 192.168.1.37
Swap usage:   0%

Graph this data and manage this system at:
  https://landscape.canonical.com/

Get cloud support with Ubuntu Advantage Cloud Guest:
  http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

ubuntu@ubuntu-vm:~$ █

```

Para conectarse a una máquina virtual basada en otro tipo de sistema operativo, deberá habilitar el protocolo de comunicación pertinente. Para ello, modifique el grupo de seguridad correspondiente para habilitar el tráfico desde su red y utilizar el cliente adecuado para conectarse a la máquina virtual no basada en Linux.

4.6.7 Reinstalar una máquina virtual

En caso de que necesite reinstalar el sistema operativo de alguna máquina virtual existente, puede hacerlo desde Horizon siguiendo estas instrucciones:

- a) Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión mediante los credenciales del usuario demo.
- b) Diríjase a **"Proyecto → Compute → Instancias"**.

- c) Ubique la máquina virtual que desea reinstalar y pulse clic sobre el “triángulo” que se encuentra al final de la fila. Se desplegará una serie de opciones, entre las cuales se encuentra “Reconstruir Instancia” al final de la lista. Pulse clic sobre dicha opción.

| | | | | | | | | | | | |
|--------------------------|-----------|------------------------------|--------------|---------|-------------|--------|------|------|------------|-----------------|--|
| <input type="checkbox"/> | ubuntu-vm | trusty-server-cloudimg-amd64 | 192.168.1.37 | VM-Peq | compute-key | Activo | nova | None | Ejecutando | 4 días, 3 horas | Crear instantánea Asociar IP flotante Desasociar IP flotante Editar instancia Editar grupos de seguridad Consola Ver log Pausar instancia Instancia suspendida Redimensionar instancia Instancias de reinicio suave Instancias de reinicio duro Detener instancia Reconstruir instancia Terminar instancia |
| <input type="checkbox"/> | vm-cirros | cirros-0.3.3-x86_64 | 192.168.1.36 | m1.tiny | compute-key | Activo | nova | None | Ejecutando | 4 días, 4 horas | |

Displaying 2 items

- d) Se abrirá una nueva ventana donde debe seleccionar la imagen que utilizará para reconstruir la máquina virtual.

Reconstruir instancia

Seleccionar imagen *

trusty-server-cloudimg-amd64 (244,0 MB) ▼

- Seleccionar imagen
- Fedora-x86_64-20-20140407 (201,1 MB)
- trusty-server-cloudimg-amd64 (244,0 MB)
- cirros-0.3.3-x86_64 (12,6 MB)

Descripción:

Seleccione la imagen para reconstruir su instancia.

Cancelar Reconstruir instancia

- e) Una vez que haya seleccionado la imagen deseada, haga clic sobre el botón “Reconstruir instancia”.
- f) Lo anterior lo llevará de vuelta a la lista de máquinas virtuales, donde se mostrará el estado de la máquina virtual -que se está reinstalando- en estado “Reconstruyendo”.

Una vez finalizado el proceso, cambiará su estado a "Active" y podrá utilizar nuevamente la máquina virtual.

| | Nombre de la instancia | Nombre de la imagen | Dirección IP | Tamaño | Par de claves | Estado | Zona de disponibilidad | Tarea | Estado de energía | Tiempo desde su creación | Actions |
|--------------------------|------------------------|------------------------------|--------------|------------|---------------|-------------|------------------------|--|-------------------|--------------------------|--------------------------|
| <input type="checkbox"/> | ubuntu-vm | trusty-server-cloudimg-amd64 | 192.168.1.37 | VM-Pequeño | compute-key | Reconstruir | nova |  Reconstruyendo | Ejecutando | 4 días, 3 horas | Asociar IP flotante ▼ |

4.6.8 Hacer snapshots de una máquina virtual

Los snapshots o imágenes instantáneas son muy útiles para crear un respaldo de una máquina virtual y conservar su estado en un momento preciso. Especialmente, son útiles antes de operaciones intrusivas como parcheos o actualizaciones que podrían afectar el buen funcionamiento de una máquina virtual.

Puede crear una imagen instantánea de una máquina virtual si utiliza el dashboard de OpenStack. Siga las instrucciones que se brindan a continuación:

- Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- Diríjase a **"Proyecto → Compute → Instancias"**.
- Ubique la máquina virtual a la que le desea hacer un snapshot y puse clic sobre el botón "Crear Instantánea".

| | | | | | | | | | | | |
|--------------------------|-----------|------------------------------|--------------|------------|-------------|--------|------|------|------------|-----------------|---------------------|
| <input type="checkbox"/> | ubuntu-vm | trusty-server-cloudimg-amd64 | 192.168.1.37 | VM-Pequeño | compute-key | Activo | nova | None | Ejecutando | 4 días, 3 horas | Crear instantánea ▼ |
|--------------------------|-----------|------------------------------|--------------|------------|-------------|--------|------|------|------------|-----------------|---------------------|

- Se abrirá una nueva ventana en la cual deberá asignar un nombre para el snapshot. Escríbalo y haga clic sobre el botón "Crear Instantánea".

Crear instantánea

Nombre de la instantánea *

Descripción:

Las instantáneas conservan el estado de disco de una instancia en ejecución.

- e) Lo anterior lo llevará a la sección donde se mostrará la lista de imágenes disponibles en el sistema, así como otros snapshots que hayan sido creados. Una vez que haya terminado ese procedimiento,, podrá ver su estado en "Active", tal como se muestra a continuación. Tome en cuenta que el proceso puede tardar algunos minutos, según el tamaño de la máquina virtual.

Imágenes

| Imágenes | | | | | | | | | |
|--------------------------|---------------------|--------------|--------|------------------------|-----------|-------------|----------|--|--|
| | | Proyecto (1) | | Compartido conmigo (0) | | Público (3) | | <input type="button" value="+ Crear imagen"/> <input type="button" value="✖ Eliminar imágenes"/> | |
| <input type="checkbox"/> | Nombre de la imagen | Tipo | Estado | Público | Protegido | Formato | Tamaño | Actions | |
| <input type="checkbox"/> | snapshot-ubuntu-vm | Snapshot | Active | no | no | QCOW2 | 809,0 MB | Lanzar ▾ | |
| Displaying 1 item | | | | | | | | | |

- f) Ahora la imagen se podrá utilizar para la creación de otras máquinas virtuales mediante el usuario demo o para restablecer el estado de la máquina virtual desde la cual fue creada. Debe tener en cuenta que no se trata de una imagen pública, por lo que no está disponible para otros tenants o usuarios, solamente para demo.
- g) Repita el proceso para crear otros snapshots.

4.6.9 Eliminar una máquina virtual

Si ha dejado de necesitar una máquina virtual y desea eliminarla para liberar los recursos que tiene asignados, proceda desde línea de comandos, o bien desde Horizon, de las formas como lo indican las siguientes instrucciones según sea el caso:

4.6.9.1 Eliminar una máquina virtual desde línea de comandos

- a) Inicie sesión en el nodo controlador.

```
# ssh <usuario>@controller
```

Nota: reemplace **<usuario>** y **controller** con los valores correspondientes.

- b) Cargue las variables de ambiente del tenant/usuario con las cuales se creó la máquina virtual que desea eliminar. Para efectos de los ejemplos seguidos en este proyecto, utilizaremos el tenant y usuario demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- c) Obtenga la lista de máquinas virtuales existentes en el sistema. Utilice el siguiente comando:

```
# nova list
```

```
root@controller:/tmp/imagenes# nova list
+-----+-----+-----+-----+-----+-----+
| ID          | Name      | Status | Task State | Power State | Networks |
+-----+-----+-----+-----+-----+-----+
| 25b0d8fe-2ae6-4b5e-a951-413671311b18 | demovmtest | ACTIVE | -          | Running     | demo-net=192.168.1.35 |
| 7115ebe1-19bf-4801-a035-07043c932b57 | ubuntu-vm | ACTIVE | -          | Running     | demo-net=192.168.1.37 |
| 110790fa-6d64-4349-8934-04ca424cb9e1 | vm-cirros | ACTIVE | -          | Running     | demo-net=192.168.1.36 |
+-----+-----+-----+-----+-----+-----+
```

- d) Para eliminar una máquina virtual, ejecute el comando que observará a continuación:

```
# nova delete <nombre de la máquina virtual>
```

Nota: reemplace **<nombre de la máquina virtual>** con el valor de la máquina virtual que desea eliminar.

- e) Hecho esto, la máquina es eliminada del sistema de manera exitosa y sus recursos han sido liberados.

```

root@controller:/tmp/imagenes# nova delete demovmtest
Request to delete server demovmtest has been accepted.
root@controller:/tmp/imagenes# nova list
-----+-----+-----+-----+-----+-----+
| ID          | Name      | Status | Task State | Power State | Networks |
-----+-----+-----+-----+-----+-----+
| 7115ebeb1-19bf-4801-a035-07043c932b57 | ubuntu-vm | ACTIVE | -          | Running    | demo-net=192.168.1.37 |
| 110790fa-6d64-4349-8934-04ca424cb9e1 | vm-cirros | ACTIVE | -          | Running    | demo-net=192.168.1.36 |
-----+-----+-----+-----+-----+-----+
root@controller:/tmp/imagenes#

```

4.6.9.2 Eliminar una máquina virtual desde Horizon

- Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- Diríjase a **"Proyecto → Compute → Instancias"**.
- Seleccione de la lista la máquina virtual que desea eliminar. Marque el check al inicio de la fila.

| | | | |
|-------------------------------------|------------|---------------------|----|
| <input checked="" type="checkbox"/> | vm-cirros2 | cirros-0.3.3-x86_64 | 19 |
|-------------------------------------|------------|---------------------|----|

- Una vez seleccionada la máquina virtual que desea eliminar, haga clic sobre el botón "Terminar instancias", que se encuentra en la esquina superior derecha.

 Terminar instancias

- Inmediatamente, se despliega una ventana en la cual se solicita confirmar esta acción, haga clic sobre el botón "Terminar instancias" para eliminar la máquina virtual o "Cancelar", para anular la acción.

Confirm Terminar instancias

You have selected "vm-cirros2". Please confirm your selection. This action cannot be undone.

Terminar instancias

Cancel

- f) Después de unos minutos, la máquina virtual no aparecerá más en la lista de instancias virtuales disponibles en Horizon.

4.7 Gestión de volúmenes utilizando Cinder

Cinder es el servicio utilizado para la gestión de volúmenes de almacenamiento local que pueden ser presentados a las instancias de máquinas virtuales para crear sistemas de ficheros según los requerimientos del cliente o sus aplicaciones.

4.7.1 Creación de volúmenes

Los volúmenes se crean mediante LVM versión 2 sobre un grupo de volúmenes llamado **nova-volumes**. Luego, se pueden presentar a una máquina virtual existente al especificar el tamaño en GB deseado.

4.7.1.1 Creación de volúmenes utilizando línea de comandos

- a) Inicie sesión en el nodo controlador y hágase root.
- b) Cargue las variables de entorno del usuario y tenant en el cual prefiere crear el volumen. En este caso el usuario/proyecto utilizado es demo:

```
# source /var/lib/keystone/demo-openrc.sh
```

- c) Cree el volumen deseado. Utilice el siguiente comando:

```
# cinder create --display-name demo-volume1 1
```

Nota: reemplace **demo-volume1** por el nombre que desea asignar al volumen y **1**, por el tamaño en GB con el que desea crear dicho volumen.

```

root@controller:~# source /var/lib/keystone/demo-openrc.sh
root@controller:~# cinder create --display-name demo-volume1 1
+-----+-----+
| Property | Value |
+-----+-----+
| attachments | [] |
| availability_zone | nova |
| bootable | false |
| created_at | 2014-11-23T04:55:07.553084 |
| display_description | None |
| display_name | demo-volume1 |
| encrypted | False |
| id | fc440efa-b80b-4885-88a4-eb4a3de0b5cc |
| metadata | {} |
| size | 1 |
| snapshot_id | None |
| source_volid | None |
| status | creating |
| volume_type | None |
+-----+-----+

```

- d) Para verificar que el volumen se ha creado exitosamente, utilice el siguiente comando, que le mostrará la lista de volúmenes existentes.

```
# cinder list
```

```

root@controller:~# cinder list
+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | Status | Display Name | Size | Volume Type | Bootable | Attached to |
+-----+-----+-----+-----+-----+-----+-----+-----+
| fc440efa-b80b-4885-88a4-eb4a3de0b5cc | available | demo-volume1 | 1 | None | false | |
+-----+-----+-----+-----+-----+-----+-----+-----+

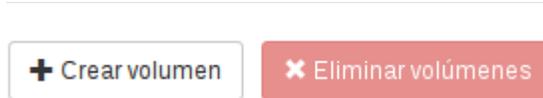
```

- e) El volumen está listo para ser vinculado con una máquina virtual. Para asociar un volumen con una máquina virtual existente siga las instrucciones de la sección **"4.7.3 Asignación de volúmenes a las máquinas virtuales"**.

4.7.1.2 Creación de volúmenes utilizando Horizon

- Abra Horizon desde su navegador. Para ello, utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- Diríjase a **"Proyecto → Compute → Volúmenes"**.

- c) Para crear un volumen haga clic sobre la opción "Crear volumen", ubicada en la esquina superior derecha de la ventana.



- d) Se abrirá una ventana en la que deberá proporcionar todos los detalles del volumen que desea crear.
- i. Escriba el nombre y la descripción del volumen.

Nombre del volumen *

Descripción

Este es un volumen de 2GB que será creado como prueba

- ii. Indique el origen del volumen. Si desea que el volumen posea el contenido de una imagen de las existentes en Glance, seleccione la opción **imagen**. Si desea que el volumen sea una copia de otro volumen existente en Cinder, seleccione la opción **volumen**. Si desea un volumen vacío, seleccione la opción **Sin origen**, tal como se muestra a continuación:

Origen del volumen

A dropdown menu with a blue border and a downward arrow on the right. The current selection is 'Volumen'. The menu is open, showing the following options: 'Sin origen, volumen vacío' (highlighted in blue), 'Imagen', and 'Volumen'.

iii. Indique el tipo del volumen.

Tipo

A dropdown menu with a blue border and a downward arrow on the right. The current selection is 'Sin tipo de volumen'.

iv. Proporcione el tamaño en GB que desea asignarle al volumen.

Tamaño (GB) *

A text input field with a light gray border containing the number '2'.

v. Puede asignar el volumen, ya sea para utilizarlo en una zona específica, o bien en cualquier zona.

Zona de disponibilidad

A dropdown menu with a blue border and a downward arrow on the right. The current selection is 'Cualquier zona de disponibilidad'. The menu is open, showing the following options: 'Cualquier zona de disponibilidad' (highlighted in blue) and 'nova'.

vi. Una vez completados todos los datos, podrá crear el volumen al escoger la opción "Crear volumen", ubicada en la esquina inferior derecha de la ventana. También, puede hacer clic sobre "Cancelar" para abortar la acción.

Two buttons are shown side-by-side. The left button is labeled 'Cancelar' and has a light gray background. The right button is labeled 'Crear volumen' and has a blue background.

- vii. El volumen está listo para ser vinculado con una máquina virtual. Para lograrlo, siga las instrucciones en la sección **"4.7.3 Asignación de volúmenes a las máquinas virtuales"**.

Volúmenes

Filter Filtrar + Crear volumen ✖ Eliminar volúmenes

| <input type="checkbox"/> | Nombre | Descripción | Tamaño | Estado | Tipo | Asociado a | Zona de disponibilidad | Arrancable | Cifrado | Actions |
|--------------------------|--------------|--|--------|-----------|------|------------|------------------------|------------|---------|------------------|
| <input type="checkbox"/> | volumen-demo | Este es un volumen de 2GB que será cr... | 2GB | Available | . | | nova | no | No | Editar volumen ▾ |

4.7.2 Eliminación de volúmenes

4.7.2.1 Eliminación de volúmenes utilizando línea de comandos

- Inicie sesión en el nodo controlador y hágase root.
- Cargue las variables de entorno del usuario y tenant en el cual quiere eliminar el volumen. En este caso el usuario/proyecto utilizado es demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- Obtenga el nombre o identificador del volumen que desea eliminar utilizando el siguiente comando. En este caso el volumen a eliminar será **demo-volume2**.

```
# cinder list
```

- Elimine el volumen. Para verificar la eliminación del volumen utilice el comando **cinder list** nuevamente.

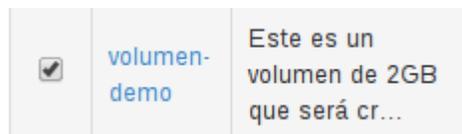
```
# cinder delete demo-volume2
# cinder list
```

Nota: reemplace **demo-volume2** con el nombre del volumen que desea eliminar.

```
root@controller:~# cinder delete demo-volume2
root@controller:~# cinder list
+-----+-----+-----+-----+-----+-----+-----+-----+
|          ID          | Status | Display Name | Size | Volume Type | Bootable | Attac
hed to |
+-----+-----+-----+-----+-----+-----+-----+-----+
| a46e0109-7d68-495f-b263-9dd4cd723d26 | deleting | demo-volume2 | 1 | None | false |
| fc440efa-b80b-4885-88a4-eb4a3de0b5cc | available | demo-volume1 | 1 | None | false |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

4.7.2.2 Eliminación de volúmenes utilizando Horizon

- a) Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- b) Diríjase a **"Proyecto → Compute → Volúmenes"**.
- c) Se desplegará la lista de volúmenes disponibles en el sistema. Para eliminar un volumen, haga clic sobre el "check" que se encuentra al inicio de la fila del volumen correspondiente y sobre el botón "Eliminar Volúmenes", ubicado en la esquina superior derecha de la ventana.



- d) Se desplegará una nueva ventana, en la cual se le pedirá confirmar la acción. Haga clic sobre "Eliminar Volúmenes" para eliminar el volumen seleccionado o sobre "Cancelar", para abortar la acción.

Confirm Eliminar volúmenes

You have selected "volumen-demo". Please confirm your selection. This action cannot be undone.

Eliminar volúmenes Cancel

4.7.3 Asignación de volúmenes a las máquinas virtuales

Para vincular un volumen a una máquina virtual, ambos deben haber sido creados con anterioridad. Una vez creados, el volumen deseado y la máquina virtual, se podrá proceder de acuerdo con cualquiera de las siguientes opciones para asignar un determinado volumen a una máquina virtual.

4.7.3.1 Asignación de volúmenes a las máquinas virtuales mediante línea de comandos

- a) Inicie sesión en el nodo controlador y hágase root.
- b) Cargue las variables de entorno del usuario y tenant al cual pertenece la máquina virtual. En este caso el usuario/proyecto utilizado es demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- c) Obtenga la lista de volúmenes disponibles en el sistema e identifique el ID del volumen que desea asignar a la máquina virtual. Utilice el siguiente comando

```
# cinder list
```

- d) Para asignar un volumen a una máquina virtual utilice el comando que se muestra a continuación:

```
# nova volume-attach vm-cirros fc440efa-b80b-4885-88a4-eb4a3de0b5cc
```

Nota: reemplace **vm-cirros** con el nombre de la máquina virtual a la que desee agregar el volumen y **fc440efa-b80b-4885-88a4-eb4a3de0b5cc**, por el identificador del volumen deseado, que se puede obtener con el comando **cinder list**.

- e) Inicie sesión en la máquina virtual en la cual fue vinculado el volumen. Hecho esto, podrá ver el nuevo disco al utilizar el comando **fdisk -l**.

```
# fdisk -l
```

4.7.3.2 Asignación de volúmenes a las máquinas virtuales utilizando Horizon

- a) Abra Horizon desde su navegador. Utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- b) Diríjase a **"Proyecto → Compute → Volúmenes"**.
- c) Para asignar un volumen haga clic sobre el "triángulo" de opciones que se encuentra al final del volumen deseado y, posteriormente, sobre la opción "Editar asociaciones".

| | | | | | | | | | |
|--------------------------|--------------|--|-----|-----------|---|------|----|----|--|
| <input type="checkbox"/> | volumen-demo | Este es un volumen de 2GB que será cr... | 2GB | Available | - | nova | no | No | Editar volumen |
| <input type="checkbox"/> | demo-volume1 | - | 1GB | Available | - | nova | no | No | Extender volumen Editar asociaciones Crear instantánea Subir Imagen Eliminar volúmen |

Displaying 2 items

- d) Se abrirá una nueva ventana para asignar el volumen a una instancia de máquina virtual en específico. Un volumen solo debe ser asociado o vinculado con una sola máquina virtual para evitar corrupción de datos.
- e) En la sección "Asociar a instancia", seleccione la máquina virtual con la que quiere asociar este volumen.

Asociar a instancia

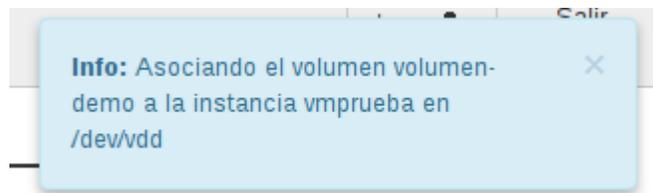
Asociar a instancia * ?

vmprueba (0bd79a1d-91f3-479e-a82a-da4daa2bdeda) ▼

Seleccione una instancia

vmprueba (0bd79a1d-91f3-479e-a82a-da4daa2bdeda)

- f) Una vez hecha la asociación, el volumen será presentado en la máquina virtual como un disco duro virtual, en la forma /dev/vdd#. Con ese disco virtual podrá crear un sistema de ficheros o un grupo de volúmenes.



- g) En la lista de volúmenes podrá ver si el volumen se encuentra asociado y cuál es el nombre virtual que se le asignó al nuevo dispositivo.

| <input type="checkbox"/> | Nombre | Descripción | Tamaño | Estado | Tipo | Asociado a | Zona de disponibilidad | Arrancable | Cifrado | Actions |
|--------------------------|--------------|--|--------|--------|------|---------------------------------|------------------------|------------|---------|------------------|
| <input type="checkbox"/> | volumen-demo | Este es un volumen de 2GB que será cr... | 2GB | In-Use | - | Asociado a vmprueba en /dev/vdd | nova | no | No | Editar volumen ▾ |

- h) Inicie sesión en la máquina virtual en la que se vinculó el volumen. Podrá ver el nuevo disco utilizando el comando **fdisk -l**.

```
# fdisk -l
```

```
root@vmprueba:~# fdisk -l
Disk /dev/vda: 10.7 GB, 10737418240 bytes
4 heads, 32 sectors/track, 163840 cylinders, total 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000a52f1

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1   *          2048     20971519     10484736   83   Linux

Disk /dev/vdb: 1073 MB, 1073741824 bytes
16 heads, 63 sectors/track, 2080 cylinders, total 2097152 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/vdb doesn't contain a valid partition table

Disk /dev/vdc: 536 MB, 536870912 bytes
16 heads, 63 sectors/track, 1040 cylinders, total 1048576 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/vdc doesn't contain a valid partition table

Disk /dev/vdd: 2147 MB, 2147483648 bytes
16 heads, 63 sectors/track, 4161 cylinders, total 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/vdd doesn't contain a valid partition table
root@vmprueba:~#
```

4.7.4 Desvinculación de volúmenes en las máquinas virtuales

4.7.4.1 Desvinculación de volúmenes de las máquinas virtuales mediante línea de comandos

- a) Inicie sesión en el nodo controlador y hágase root.

- b) Cargue las variables de entorno del usuario y tenant al cual pertenece la máquina virtual. En este caso el usuario/proyecto utilizado es demo.

```
# source /var/lib/keystone/demo-openrc.sh
```

- c) Obtenga la lista de volúmenes disponibles en el sistema. Utilice el siguiente comando e identifique el ID del volumen que desea asignar a la máquina virtual.

```
# cinder list
```

- d) Asegúrese de que el volumen no lo esté utilizando el sistema operativo de la máquina virtual a la que se encuentra asociado. El sistema de ficheros se debe desmontar y eliminar antes de proceder con la desvinculación. Para desvincular un volumen de una máquina virtual, utilice el comando que se le brinda seguidamente:

```
# nova volume-detach vm-cirros fc440efa-b80b-4885-88a4-eb4a3de0b5cc
```

Nota: reemplace **vm-cirros** con el nombre de la máquina virtual a la que desee quitar el volumen y **fc440efa-b80b-4885-88a4-eb4a3de0b5cc**, por el identificador del volumen que se desea desvincular.

4.7.4.2 Desvinculación de volúmenes de las máquinas virtuales utilizando Horizon

- a) Abra Horizon desde su navegador. Para ello, utilice la dirección <http://controller/horizon> e inicie sesión con los credenciales del usuario demo.
- b) Diríjase a **“Proyecto → Compute → Volúmenes”**.
- c) Para desvincular un volumen haga clic sobre el “triángulo” de opciones que se encuentra al final del volumen deseado y haga clic sobre la opción “Editar asociaciones”.

| | | | | | | | | | | |
|--------------------------|--------------|--|-----|-----------|---|--|------|----|----|--|
| <input type="checkbox"/> | volumen-demo | Este es un volumen de 2GB que será cr... | 2GB | Available | - | | nova | no | No | Editar volumen |
| <input type="checkbox"/> | demo-volume1 | - | 1GB | Available | - | | nova | no | No | Extender volumen Editar asociaciones Crear instantánea Subir Imagen Eliminar volumen |

Displaying 2 items

- d) Se abrirá una nueva ventana en la cual se mostrarán las asociaciones que posee el volumen. Para desvincularlo de la máquina virtual a la que se encuentra asociado, haga clic sobre la opción "Desasociar volumen". Antes de realizar este paso, asegúrese de que el volumen no lo esté utilizando el sistema operativo de la máquina virtual a la que se encuentra asociado. Recuerde que el sistema de ficheros debe ser desmontado y eliminado antes de proceder con la desvinculación.

Asociaciones

Desasociar Volúmenes

| <input type="checkbox"/> | Instancia | Dispositivo | Actions |
|--------------------------|-----------|-------------|--------------------|
| <input type="checkbox"/> | vmprueba | /dev/vdd | Desasociar Volumen |

Displaying 1 item

- e) Se abrirá una nueva ventana en la que deberá confirmar la acción. Haga clic sobre "Desasociar volumen" para proceder o "Cancelar", para abortar la acción.

Confirm Desasociar Volumen

You have selected "Volumen volumen-demo en instancia vmprueba". Please confirm your selection. This action cannot be undone.

Desasociar Volumen

Cancel

- f) Una vez completada la desvinculación, el volumen se encontrará disponible para ser asignado a otra instancia de máquina virtual, si así se desea.

| <input type="checkbox"/> | Nombre | Descripción | Tamaño | Estado | Tipo | Asociado a | Zid |
|--------------------------|--------------|--|--------|-----------|------|------------|-----|
| <input type="checkbox"/> | volumen-demo | Este es un volumen de 2GB que será cr... | 2GB | Available | - | | n |

5 Contactos

En seguida, se brinda información de las personas que puede contactar en caso de consultas, cambios técnicos, etc.

| Nombre | Rol | Dirección de correo-e |
|-----------------|---|------------------------------|
| Jennifer Vargas | Creadora de esta guía de operaciones de OpenStack | jvargas@softwarelibre.cr.org |
| | | |