

Instalación y configuración de OpenStack en Oracle® Solaris 11.2



Referencia: E56870
Abril de 2015

Referencia: E56870

Copyright © 2014, 2015, Oracle y/o sus filiales. Todos los derechos reservados.

Este software y la documentación relacionada están sujetos a un contrato de licencia que incluye restricciones de uso y revelación, y se encuentran protegidos por la legislación sobre la propiedad intelectual. A menos que figure explícitamente en el contrato de licencia o esté permitido por la ley, no se podrá utilizar, copiar, reproducir, traducir, emitir, modificar, conceder licencias, transmitir, distribuir, exhibir, representar, publicar ni mostrar ninguna parte, de ninguna forma, por ningún medio. Queda prohibida la ingeniería inversa, desensamblaje o descompilación de este software, excepto en la medida en que sean necesarios para conseguir interoperabilidad según lo especificado por la legislación aplicable.

La información contenida en este documento puede someterse a modificaciones sin previo aviso y no se garantiza que se encuentre exenta de errores. Si detecta algún error, le agradeceremos que nos lo comuniqué por escrito.

Si este software o la documentación relacionada se entrega al Gobierno de EE.UU. o a cualquier entidad que adquiera las licencias en nombre del Gobierno de EE.UU. entonces aplicará la siguiente disposición:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Este software o hardware se ha desarrollado para uso general en diversas aplicaciones de gestión de la información. No se ha diseñado ni está destinado para utilizarse en aplicaciones de riesgo inherente, incluidas las aplicaciones que pueden causar daños personales. Si utiliza este software o hardware en aplicaciones de riesgo, usted será responsable de tomar todas las medidas apropiadas de prevención de fallos, copia de seguridad, redundancia o de cualquier otro tipo para garantizar la seguridad en el uso de este software o hardware. Oracle Corporation y sus subsidiarias declinan toda responsabilidad derivada de los daños causados por el uso de este software o hardware en aplicaciones de riesgo.

Oracle y Java son marcas comerciales registradas de Oracle y/o sus subsidiarias. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

Intel e Intel Xeon son marcas comerciales o marcas comerciales registradas de Intel Corporation. Todas las marcas comerciales de SPARC se utilizan con licencia y son marcas comerciales o marcas comerciales registradas de SPARC International, Inc. AMD, Opteron, el logotipo de AMD y el logotipo de AMD Opteron son marcas comerciales o marcas comerciales registradas de Advanced Micro Devices. UNIX es una marca comercial registrada de The Open Group.

Este software o hardware y la documentación pueden proporcionar acceso a, o información sobre contenidos, productos o servicios de terceros. Oracle Corporation o sus filiales no son responsables y por ende desconocen cualquier tipo de garantía sobre el contenido, los productos o los servicios de terceros a menos que se indique otra cosa en un acuerdo en vigor formalizado entre Ud. y Oracle. Oracle Corporation y sus filiales no serán responsables frente a cualesquiera pérdidas, costos o daños en los que se incurra como consecuencia de su acceso o su uso de contenidos, productos o servicios de terceros a menos que se indique otra cosa en un acuerdo en vigor formalizado entre Ud. y Oracle.

Accesibilidad a la documentación

Para obtener información acerca del compromiso de Oracle con la accesibilidad, visite el sitio web del Programa de Accesibilidad de Oracle en <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Acceso a Oracle Support

Los clientes de Oracle que hayan adquirido servicios de soporte disponen de acceso a soporte electrónico a través de My Oracle Support. Para obtener información, visite <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> o <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> si tiene problemas de audición.

Contenido

Cómo utilizar esta documentación	9
1 Introducción a OpenStack en Oracle Solaris 11.2	11
Novedades de Oracle Solaris 11.2	11
Integración de OpenStack con Oracle Solaris	11
Cómo utilizar este manual	14
Requisitos de instalación de OpenStack	16
2 Instalación de una configuración de evaluación	19
Implementación de Unified Archive de OpenStack	19
Descarga del archivo de imagen	20
Instalación de un sistema único	21
Configuración del conmutador virtual elástico	26
Implementación del archivo unificado de Juno OpenStack	26
▼ Cómo realizar la instalación de Juno OpenStack en una zona de núcleo mediante el archivo de almacenamiento unificado	27
Uso del panel de control de OpenStack	28
▼ Cómo acceder al panel de control de OpenStack	28
Exploración del panel de control	29
Creación e inicio de una instancia de VM	33
3 Instalación en varios sistemas para una configuración de Havana OpenStack en varios nodos	41
Descripción general de la arquitectura de tres nodos	41
Configuración del nodo de controlador	45
▼ Cómo configurar el nodo de controlador	46
Instalación del Protocolo de hora de red	47
Instalación de MySQL	48
Instalación de Keystone	49
Instalación y configuración de Heat	50

Instalación de Cinder	51
Instalación de Glance	56
Instalación y configuración de Neutron en el nodo del controlador	57
Instalación de Nova	58
▼ Cómo configurar Horizon	59
Configuración del nodo de cálculo	60
▼ Cómo configurar el nodo de cálculo	61
Configuración del nodo de red	62
▼ Cómo configurar el nodo de red	64
Configuración del agente Neutron L3	66
4 Instalación en varios sistemas para una configuración de Juno de OpenStack en varios nodos	77
Descripción general de la arquitectura de tres nodos	77
Configuración del nodo de controlador	81
Pasos preliminares	81
Instalación del Protocolo de hora de red	82
Instalación de MySQL	83
Instalación de Keystone	85
Instalación de Glance	86
Instalación de Nova	88
Instalación de Horizon	89
Instalación de Cinder	91
Instalación y configuración de Neutron	94
Instalación y configuración de Heat	97
Configuración del nodo de cálculo	98
▼ Cómo configurar el nodo de cálculo	98
Configuración del nodo de almacenamiento	100
▼ Cómo configurar el nodo de almacenamiento de bloques	101
Configuración de una red interna en OpenStack	101
▼ Cómo crear una red interna	102
Configuración de OpenStack con una red externa	103
▼ Cómo configurar la red externa en OpenStack	104
▼ Cómo crear y asociar direcciones IP flotantes como usuario cliente	108
▼ Cómo observar la configuración de agente L3	109
5 Creación de instancias de máquinas virtuales	111
Gestión de tipos	111
Visualización de la información sobre tipos	112

Modificación de las especificaciones de tipos	113
Gestionar imágenes	114
Visualización de información sobre imágenes	114
Creación de una imagen	115
Agregación de una imagen al almacén de imágenes	116
Creación de una instancia de VM	117
▼ Cómo crear una instancia de VM mediante el uso de una interfaz de línea de comandos	117
6 Solución de problemas de OpenStack	119
Limitaciones conocidas	119
Examen de los archivos log	121
Investigación y resolución de problemas	122
Instalación y configuración de OpenStack	123
Instalación y configuración de una instancia de VM	123
Índice	127

Cómo utilizar esta documentación

- **Descripción general:** describe cómo instalar OpenStack e implementar máquinas virtuales de OpenStack en sistemas Oracle Solaris 11.2.
- **Destinatarios:** administradores de sistemas de instalaciones de gran tamaño.
- **Conocimientos necesarios:** red de Solaris y administración de sistemas de gran tamaño. Resulta útil estar familiarizado con OpenStack.

Biblioteca de documentación del producto

La documentación y los recursos para este producto y los productos relacionados se encuentran disponibles en <http://www.oracle.com/pls/topic/lookup?ctx=E56339>.

Comentarios

Envíenos comentarios acerca de esta documentación mediante <http://www.oracle.com/goto/docfeedback>.

◆◆◆ 1 C A P Í T U L O 1

Introducción a OpenStack en Oracle Solaris 11.2

En este capítulo, se describe cómo se usan las funciones de Oracle Solaris en OpenStack de Oracle Solaris 11.2.

En este capítulo, se tratan los temas siguientes:

- [“Novedades de Oracle Solaris 11.2” \[11\]](#)
- [“Integración de OpenStack con Oracle Solaris” \[11\]](#)
- [“Cómo utilizar este manual” \[14\]](#)
- [“Requisitos de instalación de OpenStack” \[16\]](#)

Novedades de Oracle Solaris 11.2

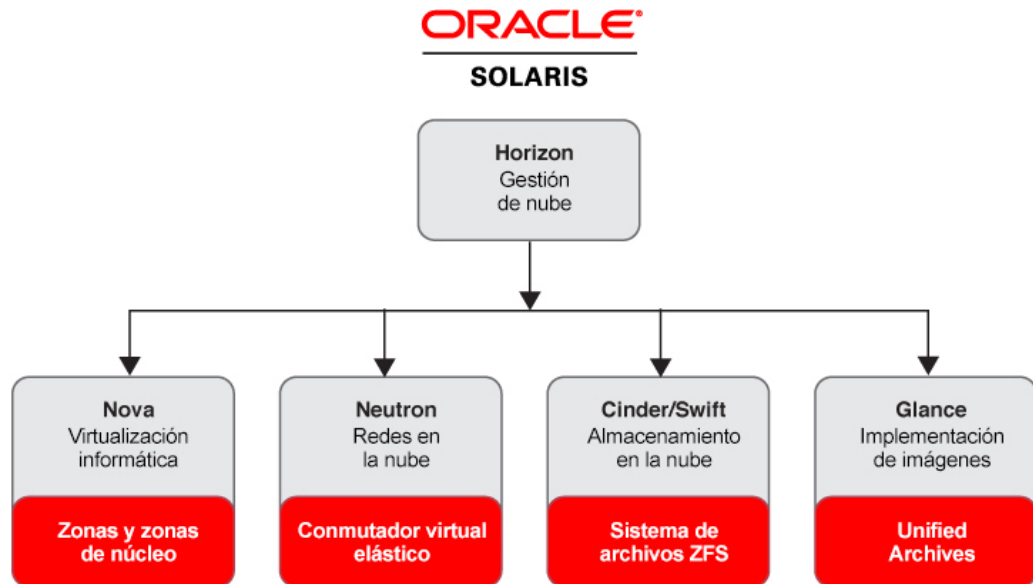
A partir de Oracle Solaris 11.2 SRU 10, se admite la versión Juno de OpenStack. Este documento contiene procedimientos para la configuración de OpenStack específica para Havana o Juno.

- [Capítulo 3, Instalación en varios sistemas para una configuración de Havana OpenStack en varios nodos](#)
- [Capítulo 4, Instalación en varios sistemas para una configuración de Juno de OpenStack en varios nodos](#)

Integración de OpenStack con Oracle Solaris

Oracle Solaris 11.2 incluye una distribución de OpenStack que está completamente integrada con las tecnologías centrales de Oracle Solaris 11. OpenStack en Oracle Solaris 11.2 le permite crear una nube privada de IaaS (Infraestructura como servicio) apta para empresas, de modo que los usuarios pueden crear rápidamente redes virtuales y realizar cálculos de los recursos mediante el uso del portal centralizado basado en Web.

En la siguiente figura, se muestran las funciones de Oracle Solaris 11.2 que se usan para implementar los servicios de OpenStack. Estas relaciones se describen debajo de la figura.

FIGURA 1-1 Integración de Oracle Solaris y OpenStack

Los siguientes servicios de OpenStack se proporcionan en Oracle Solaris 11.2:

- Nova

El servicio de virtualización informática de Nova proporciona un controlador de tejido informático en nube que admite una variedad de tecnologías de virtualización. En Solaris, las instancias de máquinas virtuales (VM) son zonas de núcleo o zonas no globales. Las zonas son entornos virtuales de alta densidad escalables con sobrecarga baja de virtualización. Las zonas de núcleo también proporcionan versiones independientes del núcleo, lo que permite la actualización independiente de las instancias de VM y resulta conveniente para una nube de varios clientes.

Para obtener más información sobre las zonas de Oracle Solaris, consulte la documentación sobre zonas en [Oracle Solaris 11.2 Library](#).

- Neutron

El servicio de virtualización de redes de Neutron proporciona conectividad de red para otros servicios de OpenStack en varios sistemas de OpenStack y para las instancias de VM. En Solaris, los servicios de virtualización de red se proporcionan mediante la función EVS, que actúa como un único punto de control para la creación, configuración y supervisión de los controladores virtuales en múltiples servidores físicos. Las aplicaciones pueden controlar su propio comportamiento para priorizar el tráfico de red en la nube. Neutron proporciona una API para que los usuarios soliciten y configuren redes virtuales de forma dinámica. Estas redes conectan interfaces, como las VNIC de instancias de VM de Nova.

Para obtener más información sobre los controladores EVS, consulte [Capítulo 5, “Acerca de los conmutadores virtuales elásticos”](#) de “Gestión de virtualización de red y recursos de red en Oracle Solaris 11.2”.

- Cinder

El servicio de almacenamiento de bloques de Cinder proporciona una infraestructura para gestionar los volúmenes de almacenamiento de bloques en OpenStack. Cinder le permite exponer los dispositivos de bloque y conectar los dispositivos de bloque a instancias de VM para un almacenamiento expandido, mayor rendimiento e integración con plataformas de almacenamiento empresarial. En Solaris, Cinder usa ZFS para almacenamiento y usa iSCSI o canal de fibra para acceso remoto. ZFS proporciona servicios de datos integrados, incluidos instantáneas, cifrado y anulación de duplicación. Un controlador Cinder también está disponible en el dispositivo de almacenamiento ZFS.

Para obtener más información sobre ZFS, consulte “Gestión de sistemas de archivos ZFS en Oracle Solaris 11.2”. La documentación sobre el dispositivo de almacenamiento ZFS está disponible en <https://docs.oracle.com/en/storage/>.

- Swift

El servicio de almacenamiento de objetos Swift proporciona servicios de almacenamiento de objetos redundantes y escalables para proyectos y usuarios de OpenStack. Swift almacena y recupera datos arbitrarios no estructurados mediante ZFS y luego es posible acceder a los datos mediante una API RESTful.

- Glance

El servicio del almacén de imágenes de Glance almacena imágenes de disco de máquinas virtuales, que se usan para implementar instancias de VM. En Solaris, las imágenes de Glance son Unified Archive. Las imágenes se pueden almacenar en una variedad de ubicaciones, desde sistemas simples de archivos hasta sistemas de almacenamiento de objetos, como OpenStack Swift. Glance tiene una API RESTful que le permite consultar metadatos y también recuperar la imagen.

Los archivos de Unified Archive permiten realizar implementaciones seguras, rápidas, compatibles y escalables. Se puede usar el mismo Unified Archive para implementar sistemas bare-metal o virtuales. Puede usar Unified Archive con Automated Installer (AI) para aprovisionar rápidamente varios sistemas.

Para obtener más información, consulte “Uso de Unified Archives para la clonación y la recuperación del sistema en Oracle Solaris 11.2”. Una instalación de AI es un método de realizar una instalación automatizada mediante una imagen AI en medios o un servidor. Para obtener más información, consulte [Capítulo 5, “Instalaciones automatizadas que se inician desde medios”](#) de “Instalación de sistemas Oracle Solaris 11.2”.

- Horizon

Horizon es el panel de control de OpenStack, donde puede gestionar la infraestructura de nube y la infraestructura informática para admitir varias instancias de VM. El panel de control proporciona una interfaz de usuario basada en Web para los servicios de OpenStack. Consulte “Uso del panel de control de OpenStack” [28] para ver un ejemplo.

- Keystone

El servicio de identidad Keystone proporciona servicios de autenticación y autorización entre usuarios, administradores y servicios de OpenStack.

- El motor de servicio de orquestación Heat permite que los desarrolladores automaticen la implementación de una infraestructura de OpenStack. El motor se basa en las plantillas que contienen información de configuración y publican las operaciones de instalación para desplegar una configuración personalizada.

Cada servicio de OpenStack está representado por uno o más servicios de Utilidad de gestión de servicios (SMF). Por ejemplo, consulte la lista de servicios de SMF en [Tabla 3-1, “Servicios SMF instalados en los nodos de controlador, red y cálculo”](#). SMF regula los servicios de OpenStack, por ejemplo, mediante la realización de un reinicio de servicio en caso de fallo o comprobación de dependencia de servicio completa para obtener un arranque más preciso y eficiente.

Image Packaging System (IPS) permite realizar implementaciones sencillas y actualizaciones rápidas del sistema OpenStack, protegidas contra fallas. Los entornos de inicio (BE) le permite mantener fácilmente un entorno de copia de seguridad cuando actualiza los sistemas OpenStack. Todos los servicios de OpenStack se entregan en su propio paquete IPS para proporcionar flexibilidad de instalación, incluida la minimización. Todos los paquetes de servicio de OpenStack proporcionan un usuario y un grupo únicos para ese servicio de OpenStack y un perfil de RBAC para administrar ese servicio de OpenStack.

Cómo utilizar este manual

En este manual, se trata principalmente información sobre OpenStack que es diferente entre Solaris y otras plataformas. Las funciones que funcionan de la misma manera en Solaris y en otras plataformas, y las operaciones que realiza de la misma manera en Solaris y en otras plataformas, en general, no se tratan en este manual.

[Capítulo 2, Instalación de una configuración de evaluación](#) describe cómo instalar OpenStack rápidamente en un único sistema Solaris para fines de evaluación. La instalación completa se entrega en un Unified Archive y se realiza la mayor parte de la configuración.

En el [Capítulo 3, Instalación en varios sistemas para una configuración de Havana OpenStack en varios nodos](#), se describe cómo instalar y configurar OpenStack en tres sistemas Solaris: un nodo de controlador, un nodo de red y un nodo de cálculo.

[Capítulo 5, Creación de instancias de máquinas virtuales](#) proporciona información específica de Solaris para creación y uso de una instancia de VM. No se analizan tareas como creación de clientes y usuarios asociados porque esas tareas son iguales en Solaris y en otras plataformas.

Para obtener información general sobre el OpenStack Havana en el que se basa OpenStack esta versión de Oracle Solaris, consulte los siguientes recursos y obtenga más información en el [sitio de documentación de OpenStack](#):

- *Guías de formación de OpenStack*
- *Guía del usuario final*, que incluye la “Hoja de referencia de la interfaz de línea de comandos de OpenStack”
- *Guía del Usuario Administrador*
- *Referencia de la interfaz de la línea de comandos*
- *Referencia de configuración*
- *Guía del Administrador de Nubes*

Para obtener más información acerca de Solaris, consulte [Oracle Solaris 11.2 Information Library](#). Para obtener más información acerca de OpenStack en Solaris, consulte [OpenStack for Oracle Solaris 11](#).

En la comunidad de OpenStack, en ocasiones, diferentes términos tienen el mismo significado. Por ejemplo, una máquina virtual en la nube se puede denominar servidor, instancia o VM de cálculo. Una parte funcional de OpenStack, como cálculos o redes se puede denominar módulo, componente o servicio. En OpenStack, los términos *proyecto* e *inquilino* se usan indistintamente. Este documento usa los siguientes términos:

servicio	Un servicio de OpenStack, como Nova o el servicio de cálculo.
servicio SMF	Un servicio de Solaris, como <code>svc:/application/openstack/nova/nova-compute:default</code> . Las frases como “active el servicio” hacen referencia a los servicios SMF.
nodo	Un sistema que hospeda servicios de OpenStack. Por ejemplo, un nodo de controlador hospeda servicios de Keystone, Glance y Horizon.
proyecto	En las zonas de Oracle Solaris, un proyecto es un identificador administrativo de toda la red de un trabajo relacionado. En este documento, sin embargo, el término se utiliza según la definición de OpenStack, que es una agrupación lógica de usuarios en el módulo de cálculo. El proyecto define las cuotas y el acceso a las imágenes de VM.
instancia de VM	Una máquina virtual en la nube. Una instancia de VM es una VM en ejecución, o una VM con un estado conocido, como suspendido, que se puede usar como un servidor de hardware.
zona	Tecnología en Oracle Solaris para virtualizar el sistema operativo y proporcionar entornos aislados y seguros para ejecutar aplicaciones. El término también puede hacer referencia al entorno virtualizado. En Oracle Solaris, la virtualización de cálculo de OpenStack está integrada en tecnología de zonas.

Para obtener más explicaciones de los términos de OpenStack, consulte <http://docs.openstack.org/glossary/content/glossary.html>.

Requisitos de instalación de OpenStack

Los sistemas Solaris en los que instala OpenStack deben cumplir con los siguientes requisitos:

- **Sistema operativo.** Los sistemas OpenStack deben ejecutar Oracle Solaris 11.2. Para instalar Oracle Solaris 11.2, consulte [“Instalación de sistemas Oracle Solaris 11.2”](#). Para descargar las imágenes de instalación, consulte la página [Oracle Solaris 11.2 Downloads](#). Para actualizar desde una versión anterior de Oracle Solaris 11 a Oracle Solaris 11.2, consulte [“Actualización a Oracle Solaris 11.2”](#) y [Capítulo 4, “Actualización de una imagen de Oracle Solaris”](#) de [“Agregación y actualización de software en Oracle Solaris 11.2”](#).
- **Hardware.** Para asegurarse de que los sistemas admiten Oracle Solaris 11.2, consulte [Oracle Solaris 11.2 System Requirements](#) o [“Requisitos de sistema para instalar Oracle Solaris 11.2”](#) de [“Notas de la versión de Oracle Solaris 11.2”](#). Necesitará hasta 5 GB de espacio adicional para instalar OpenStack, según cuáles servicios de OpenStack instale en cada nodo. Asegúrese de tener suficientes CPU, memoria y espacio en disco en los nodos de cálculo para admitir el número deseado de instancias de VM. Los sistemas deberán tener de 100 a 200 GB de almacenamiento ZFS para imágenes de instancias de VM y para creación de instancias de VM.
- **Compatibilidad con virtualización.** Los sistemas OpenStack deberán admitir zonas de núcleo. Las instancias de VM podrán ser zonas no globales o zonas de núcleo.

Para verificar si el sistema admite la virtualización, escriba el comando `virtinfo` en una ventana de terminal. La salida del comando debería mostrar la información como se muestra en el siguiente ejemplo:

```
# virtinfo
NAME          CLASS
non-global-zone supported
kernel-zone  supported
```

Para soporte de la zona del núcleo, el sistema debe cumplir los siguientes requisitos adicionales:

- Un mínimo de 8 GB de RAM física.
- Suficiente ajuste de ZFS Adaptive Replacement Cache (ARC) en el host para evitar errores de memoria. Para obtener más información, consulte [“Ajuste de la caché ZFS ARC del host en el host de una zona del núcleo”](#) de [“Creación y uso de zonas del núcleo de Oracle Solaris”](#).

Nota - Las zonas de núcleo no se pueden ejecutar en invitados Oracle VM Server for x86 o en Oracle VM VirtualBox.

Para obtener más información acerca de los requisitos de instalación de la zona de núcleo, consulte los siguientes recursos:

- [“Requisitos de hardware y software de las zonas del núcleo de Oracle Solaris”](#) de [“Creación y uso de zonas del núcleo de Oracle Solaris ”](#)
- El archivo README de la sección [“Unified Archive de Oracle Solaris 11.2 OpenStack”](#) de la página de descargas [Oracle Solaris 11.2 OpenStack Unified Archive](#)

Consulte [“Limitaciones conocidas” \[119\]](#) para obtener información adicional que puede afectar la instalación de OpenStack.

◆◆◆ CAPÍTULO 2

Instalación de una configuración de evaluación

Para fines de evaluación, puede instalar OpenStack en un único sistema Oracle Solaris. Este tipo de instalación de OpenStack también se denomina instalación de un único nodo. La versión de OpenStack puede ser Havana o Juno, en función del sistema Oracle Solaris.

Para obtener una descripción de la instalación de OpenStack en varios nodos, consulte el [Capítulo 3, Instalación en varios sistemas para una configuración de Havana OpenStack en varios nodos](#) o el [Capítulo 4, Instalación en varios sistemas para una configuración de Juno de OpenStack en varios nodos](#).

En este capítulo, se describe cómo instalar un único nodo mediante el uso de Unified Archive y cómo crear fácilmente instancias de VM mediante el uso de una interfaz de explorador. En este capítulo, se tratan los siguientes temas:

- [“Implementación de Unified Archive de OpenStack” \[19\]](#)
- [“Uso del panel de control de OpenStack” \[28\]](#)

Nota - El contenido de este capítulo se aplica tanto a la versión Havana como a la versión Juno, a menos que se indique lo contrario.

Implementación de Unified Archive de OpenStack

La [página de descarga de archivos unificados](#) brinda la opción de descargar solamente el archivo de Oracle Solaris o Oracle Solaris con el archivo de OpenStack.

El archivo de Oracle Solaris con OpenStack instala el sistema operativo y todos los servicios de OpenStack que se muestran en [“Integración de OpenStack con Oracle Solaris” \[11\]](#).

El archivo unificado de Oracle Solaris OpenStack proporciona las siguientes funciones:

- Servicios de OpenStack, en su mayoría, preconfigurados, lo que disminuye la cantidad de trabajo de configuración que necesita llevar a cabo.
- Secuencias de comandos que automatizan la configuración adicional que es necesario realizar.

- Plantillas del sistema Solaris.
- Un almacén de imágenes Glance precargado con dos imágenes: una zona no global y una zona de núcleo.

Puede implementar Unified Archive de OpenStack en un sistema bare-metal o en una zona de núcleo. Consulte [“Instalación de un sistema único” \[21\]](#) para conocer opciones.

Descarga del archivo de imagen

Use el siguiente procedimiento para descargar un archivo de imagen única que proporcione el sistema operativo Oracle Solaris y OpenStack.

▼ Cómo descargar el archivo de imagen

1. En un explorador de Internet, vaya a la página de descarga [Unified Archives](#).
2. Lea el acuerdo de licencia y haga clic en el botón Aceptar.
3. Descargue el archivo adecuado para el tipo de instalación que desea realizar.

- **Descargue el archivo Unified Archive para la arquitectura del sistema.**

Use este archivo para realizar uno de los siguientes tipos de instalaciones:

- Use este archivo `.uar` como argumento para que el comando `zoneadm install` instale directamente una zona de núcleo.
- Haga referencia a este archivo `.uar` en un manifiesto de instalación AI para realizar una instalación de AI mediante un servicio de instalación de AI.
- Use este archivo `.uar` para crear un medio de inicio AI.

- **Descargue el archivo de imagen de inicio USB para la arquitectura del sistema.**

Use este archivo `usb` para instalar el archivo unificado desde el medio de inicio.

4. **Verifique la integridad del archivo descargado.**

Haga clic en el enlace de sumas de comprobación MD5. Ejecute el siguiente comando `digest` y compare la salida con la suma de comprobación correspondiente para el archivo de sumas de comprobación:

```
$ digest -a md5 file
```

Instalación de un sistema único

En esta sección, se describe cómo instalar la configuración de evaluación de OpenStack de un único sistema. Los primeros tres métodos descritos realizan la instalación en bare-metal. El cuarto método realiza la instalación en una zona de núcleo.

- [Cómo realizar la instalación con el archivo USB descargado \[21\]](#)
- [Cómo realizar la instalación mediante el archivo Unified Archive y un servicio de instalación AI \[22\]](#)
- [Cómo realizar la instalación mediante un medio de inicio AI desde el archivo Unified Archive \[23\]](#)
- [Cómo realizar la instalación en una zona de núcleo mediante el archivo de almacenamiento unificado de Havana \[24\]](#)

▼ Cómo realizar la instalación con el archivo USB descargado

Este procedimiento describe cómo instalar Oracle Solaris y OpenStack en sistemas bare-metal mediante el archivo USB descargado. Este método no requiere un servidor AI y es el método más directo para instalaciones bare-metal.

1. **Descargue el archivo USB para la arquitectura del sistema de destino.**
Consulte [Cómo descargar el archivo de imagen \[20\]](#).
2. **Transfiera el archivo USB a una unidad flash USB.**
 - **Uso del comando `usbcopy`.**

Si tiene acceso a un sistema Oracle Solaris 11.2, use la utilidad `usbcopy`. Consulte la página del comando `man usbcopy(1M)`.

Debe usar el comando `usbcopy` de Oracle Solaris 11.2. No puede usar `usbcopy` de una versión anterior a Solaris.
 - **Uso del comando `dd`.**

Si no tiene acceso a un sistema Oracle Solaris 11.2, puede usar el comando `dd`. Asegúrese de identificar correctamente el disco adecuado (unidad flash) cuando use `dd`.
 - **En Oracle Solaris 11:**
 - a **Desactive el servicio HAL.**

```
# svcadm disable -t hal
```
 - b **Inserte la unidad flash y ubique el dispositivo adecuado.**

```
# rmformat
```

c Copie la imagen.

```
# dd if=/path/image.usb of=/dev/rdisk/device bs=16k
```

d Active el servicio HAL.

```
# svcadm enable hal
```

■ **En Linux:**

a Inserte la unidad flash y ubique el dispositivo adecuado.

```
# dmesg | tail
```

b Copie la imagen.

```
# dd if=/path/image.usb of=/dev/diskN bs=16k
```

■ **En MacOSX:**

a Identifique la unidad, /dev/diskN, donde N es el número de disco.

```
# diskutil list  
# diskutil unmountDisk /dev/diskN
```

b Copie la imagen.

```
# dd if=/path/image.usb of=/dev/diskN bs=16k
```

3. Inserte la unidad flash en el sistema e inicie el sistema desde USB.

Se mostrará la herramienta System Configuration Interactive (SCI). Si no ve la herramienta SCI, presione la tecla Intro o presione Ctrl-L para volver a generar la pantalla.

▼ **Cómo realizar la instalación mediante el archivo Unified Archive y un servicio de instalación AI**

Este procedimiento describe cómo instalar Oracle Solaris y OpenStack en un sistema bare-metal mediante el archivo de almacenamiento unificado descargado y AI.

- 1. Descargue el archivo Unified Archive para la arquitectura del sistema de destino.**
Consulte [Cómo descargar el archivo de imagen \[20\]](#).
- 2. Cree un manifiesto de AI.**

En el servidor de instalación AI de Oracle Solaris, copie y modifique el archivo `/usr/share/auto_install/manifest/default_archive.xml` según la configuración de AI. En la sección ARCHIVE software, especifique la ubicación del archivo `.uar` descargado.

3. Configure un servicio de instalación AI.

Configure un servicio de instalación AI mediante el manifiesto de AI del paso anterior. Consulte la [Parte III, “Instalación con un servidor de instalación”](#) de “[Instalación de sistemas Oracle Solaris 11.2](#)”.

4. Inicie el sistema desde la red.

```
ok boot net -install
```

5. Una vez finalizada la instalación, reinicie el sistema.

Se mostrará la herramienta SCI para configurar el sistema. Si no ve la herramienta SCI, presione la tecla Intro o presione Ctrl-L para volver a generar la pantalla.

▼ **Cómo realizar la instalación mediante un medio de inicio AI desde el archivo Unified Archive**

Este procedimiento describe cómo instalar Oracle Solaris 11.2 y OpenStack en sistemas bare-metal mediante la creación de una imagen AI de inicio. Se crea una imagen USB con capacidad de inicio desde el archivo Unified Archive descargado. Consulte [Capítulo 5, “Instalaciones automatizadas que se inician desde medios”](#) de “[Instalación de sistemas Oracle Solaris 11.2](#)” para obtener más información acerca de este método.

1. Descargue el archivo Unified Archive para la arquitectura del sistema de destino.

Consulte [Cómo descargar el archivo de imagen \[20\]](#).

2. Cree el USB de AI desde el archivo Unified Archive.

```
# archiveadm create-media -s http://pkg.oracle.com/solaris/release \  
-f usb -o workdir/usb-filename \  
\  
workdir/uar-file
```

donde `workdir` es la ubicación en donde descargó el archivo Unified Archive. El archivo USB AI se creará en el mismo directorio.

3. Transfiera el archivo USB a una unidad flash USB.

- Si ha descargado el archivo USB a un sistema Oracle Solaris 11.2, utilice el comando `usbcopy`.
- Si ha descargado el archivo USB a un sistema cuyo sistema operativo Oracle Solaris es anterior a 11.2, utilice el comando `dd` de la siguiente manera:

1. Desactive el servicio HAL.

```
# svcadm disable -t hal
```

2. Inserte la unidad flash y ubique el dispositivo adecuado.

```
# rmformat
```

3. Copie la imagen en la unidad flash.

```
# dd if=/path/image.usb of=/dev/rdisk/device bs=16k
```

4. Active el servicio HAL.

```
# svcadm enable hal
```

4. Revise el manifiesto AI predeterminado.

Puede usar el manifiesto predeterminado o crear un manifiesto personalizado. Si crea un manifiesto personalizado, almacene el manifiesto personalizado en una ubicación a la que pueda acceder el sistema que se instala.

5. Inserte la unidad flash USB en el sistema e inicie el sistema desde USB.

Se le pedirá que use el manifiesto de AI o que proporcione la ubicación del manifiesto personalizado.

Se mostrará la herramienta SCI para configurar el sistema. Si no ve la herramienta SCI, presione la tecla Intro o presione Ctrl-L para volver a generar la pantalla.

▼ Cómo realizar la instalación en una zona de núcleo mediante el archivo de almacenamiento unificado de Havana

Este procedimiento describe cómo instalar Oracle Solaris 11.2 y la versión Havana de OpenStack directamente en una zona de núcleo mediante el archivo de almacenamiento unificado descargado.

Antes de empezar Asegúrese de que el sistema que alojará la zona del núcleo satisface los requisitos de virtualización especificados en [“Requisitos de instalación de OpenStack” \[16\]](#).

1. **Descargue el archivo Unified Archive para la arquitectura del sistema de destino.**

Consulte [Cómo descargar el archivo de imagen \[20\]](#).

2. **Cree una zona de núcleo.**

```
# zonecfg -z OpenStackKZ create -t SYSsolaris-kz
```

3. **Configure la zona del núcleo.**

Asegúrese de que la zona de núcleo tenga suficientes CPU, RAM, almacenamiento y direcciones MAC virtuales. Ninguna zona global creada dentro de la zona de núcleo podrá consumir estas direcciones MAC adicionales automáticamente.

El siguiente ejemplo configura la zona con 8 CPU virtuales, un límite de 8 GB de memoria física y asignación automática de direcciones MAC. Consulte la página del comando [zonecfg\(1M\)](#) para obtener descripciones de los recursos configurables.

```
# zonecfg -z OpenStackKZ
zonecfg:OpenStackKZ> add virtual-cpu
zonecfg:OpenStackKZ:virtual-cpu> set ncpus=8
zonecfg:OpenStackKZ:virtual-cpu> end
zonecfg:OpenStackKZ> select capped-memory
zonecfg:OpenStackKZ:capped-memory> set physical=8g
zonecfg:OpenStackKZ:capped-memory> end
zonecfg:OpenStackKZ> select anet id=0
zonecfg:OpenStackKZ:anet> add mac
zonecfg:OpenStackKZ:anet:mac> set mac-address=auto
zonecfg:OpenStackKZ:anet:mac> end
zonecfg:OpenStackKZ:anet> end
zonecfg:OpenStackKZ> exit
```

4. Compruebe su configuración.

```
# zonecfg -z OpenStackKZ info
```

5. Instale la zona del núcleo.

El siguiente ejemplo muestra la instalación de la versión x86 del archivo `.uar`.

```
# zoneadm -z OpenStackKZ install -a path/uar-file
```

donde `uar-file` es la versión Oracle Solaris 11.2 con el archivo de Havana OpenStack.

6. Inicie la zona.

```
# zoneadm -z OpenStackKZ boot
```

7. Inicie sesión en la consola de la zona para completar la configuración.

```
# zlogin -C OpenStackKZ
```

Se mostrará la herramienta SCI para configurar el sistema. Si no ve la herramienta SCI, presione la tecla Intro o presione Ctrl-L para volver a generar la pantalla.

8. Asegúrese de que la zona de núcleo tenga una dirección IP.

Unified Archive espera que un servidor DHCP asigne una dirección IP a la zona de núcleo. Si usa DHCP, asegúrese de que la dirección MAC esté asignada a la zona de núcleo. La selección de Automático en la página de red de la herramienta SCI del paso anterior generará la asignación de una dirección MAC. Si ni usa DHCP, asegúrese de asignar una dirección IPv4 a la zona de núcleo.

La dirección IP es importante porque actualmente no existen repositorios de sistema para las zonas de núcleo. Con una dirección IP, la zona de núcleo puede conectarse a un repositorio de paquete IPS si se requiere.

Configuración del conmutador virtual elástico

Una vez que se haya completado la instalación y que se haya reiniciado el sistema, deberá configurar el Conmutador virtual elástico (EVS). EVS le permite crear, configurar y supervisar los conmutadores virtuales que abarcan servidores físicos y conecta VM que forman parte del mismo EVS. Para obtener más información acerca de EVS, consulte [Capítulo 5, “Acerca de los conmutadores virtuales elásticos”](#) de “[Gestión de virtualización de red y recursos de red en Oracle Solaris 11.2](#)”.

Se proporciona una secuencia de comandos para automatizar la configuración de EVS. Ejecute la secuencia de comandos con el siguiente comando:

```
# /usr/demo/openstack/configure_evs.py
```

Esta secuencia de comandos realiza la siguiente configuración:

- Crea claves de shell seguro (SSH) para todos los usuarios requeridos, como los usuarios de UNIX root, evsuser y neutron y anexa esas claves públicas al archivo `/var/user/evsuser/.ssh/authorized_keys` para el usuario evsuser.
- Configura EVS.
- Activa los servicios SMF `neutron-server:default` y `neutron-dhcp-agent:default`.
- Especifica cuál tecnología LAN virtual se usará (VLAN o VXLAN) y los ID o segmentos correspondientes.

Implementación del archivo unificado de Juno OpenStack

Para implementar el archivo de Juno OpenStack, primero, debe asegurarse de que el sistema esté ejecutando la versión Oracle Solaris 11.2 SRU 10. Las versiones de Oracle Solaris 11.2 anteriores no admiten Juno.

▼ Cómo realizar la instalación de Juno OpenStack en una zona de núcleo mediante el archivo de almacenamiento unificado

Antes de empezar El sistema que aloja la zona del núcleo debe cumplir los requisitos de virtualización especificados en [“Requisitos de instalación de OpenStack” \[16\]](#). Asegúrese también de que el archivo de almacenamiento unificado de OpenStack se haya descargado en el sistema. Consulte [Cómo descargar el archivo de imagen \[20\]](#).

1. Cree una zona de núcleo.

```
# zonecfg -z kzone-name create -t SYSsolaris-kz
```

Mediante este paso, se crea una zona de núcleo basada en la plantilla de Oracle Solaris denominada SYSsolaris-kz.

2. Configure la zona del núcleo.

En el siguiente ejemplo, se configura la zona con 8 CPU virtuales y una memoria física con un límite de 12 GB. Consulte la página del comando [man zonecfg\(1M\)](#) para obtener descripciones de los recursos configurables.

```
# zonecfg -z kzone-name
zonecfg:OpenStackKZ> add virtual-cpu
zonecfg:OpenStackKZ:virtual-cpu> set ncpus=8
zonecfg:OpenStackKZ:virtual-cpu> end
zonecfg:OpenStackKZ> select capped-memory
zonecfg:OpenStackKZ:capped-memory> set physical=12g
zonecfg:OpenStackKZ:capped-memory> end
zonecfg:OpenStackKZ> verify
zonecfg:OpenStackKZ> exit
```

3. (Opcional) Compruebe su configuración.

```
# zonecfg -z kzone-name info
```

4. Instale la zona del núcleo.

En el siguiente ejemplo, se utilizan 50 GB de espacio en disco para garantizar que la zona de núcleo tenga suficiente espacio para crear volúmenes para las instancias de VM.

```
# zoneadm -z kzone-name install -a archive-path -x install-size=50g
```

donde *archive-path* hace referencia al nombre de ruta completo de la ubicación del archivo unificado de Juno OpenStack.

5. Inicie la zona.

```
# zoneadm -z kzone-name boot
```

6. Inicie sesión en la consola de la zona para completar la configuración.

```
# zlogin -C kzone-name
```

Se mostrará la herramienta SCI para configurar el sistema. Si no ve la herramienta SCI, presione la tecla Intro o presione Ctrl-L para volver a generar la pantalla.

Uso del panel de control de OpenStack

Después de completar las tareas de configuración de instalación y posteriores a la instalación para OpenStack, inicie sesión en el panel de control de OpenStack para ver los recursos disponibles y para iniciar una instancia de VM.

▼ Cómo acceder al panel de control de OpenStack

- 1. Inicie sesión en cualquier sistema que se pueda conectar al sistema OpenStack.**
- 2. Configure el explorador.**
 - a. Active JavaScript.**
 - b. Mantenga las cookies.**

3. En el campo de ubicación o dirección del explorador, introduzca la siguiente ubicación:

```
http://system/horizon/
```

system es el nombre o la dirección IP del sistema OpenStack donde se instaló Unified Archive de OpenStack y donde se ejecuta el servicio Horizon OpenStack con el servidor web Apache.

Si realizó la instalación desde Unified Archive en una zona de núcleo, el sistema OpenStack es la zona del núcleo y *system* es el nombre o la dirección IP de la zona de núcleo.

4. Proporcione la siguiente información en la pantalla de inicio de sesión:

- Nombre de usuario: admin
- Contraseña: secrete

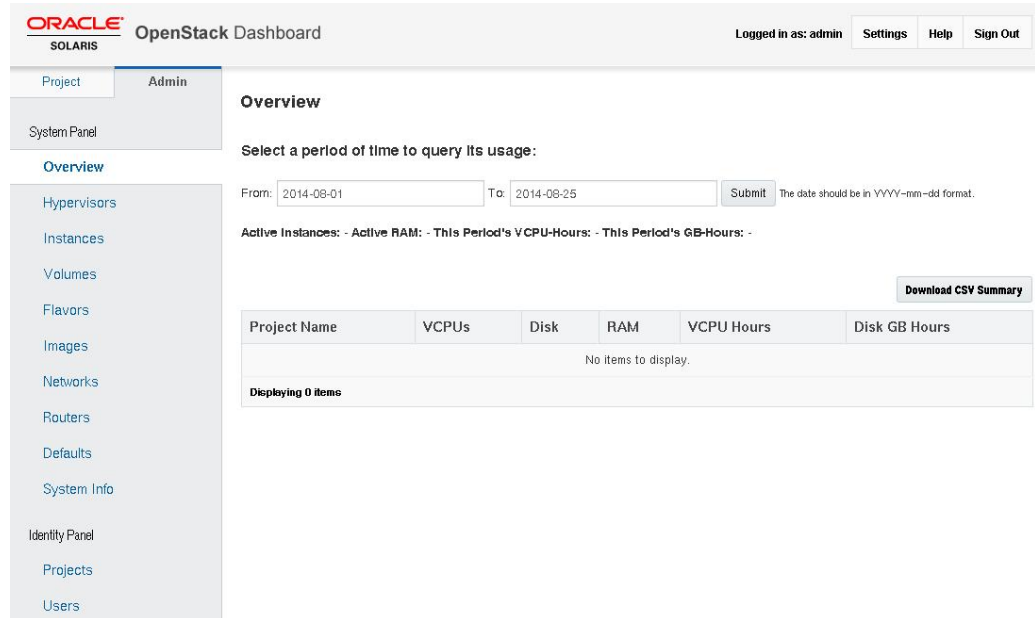
Las funciones disponibles en el panel de control dependen de los permisos de acceso o los roles del usuario con los que ha iniciado sesión.

Exploración del panel de control

Cuando inicia sesión como administrador de la nube, la ventana contiene un panel en el lado izquierdo de la pantalla con dos separadores: Proyecto y Administrador. El panel de Administrador es la vista de administrador de nube predeterminada. Las selecciones en el panel de Administrador proporcionan la siguiente funcionalidad:

- Una vista general de las instancias de Nova y los volúmenes de Cinder que se encuentran en uso dentro de la nube.
- La capacidad de ver y editar las definiciones de tipos que definen las características de instancia de VM, como las siguientes:
 - La cantidad de CPU virtuales.
 - La cantidad de memoria.
 - El espacio de disco asignado.
 - La marca de la zona de Solaris subyacente: `solaris` para zonas no globales y `solaris-kz` para zonas de núcleo.
- La capacidad de crear redes virtuales y enrutadores para uso por administradores de la nube.
- La capacidad de ver y editar proyectos mediante la agrupación y el aislamiento de la propiedad de recursos informáticos virtuales.
- La capacidad de ver y editar usuarios, que son las personas o los servicios que usan recursos en la nube.

FIGURA 2-1 Ventana Descripción general del administrador del panel de control de OpenStack

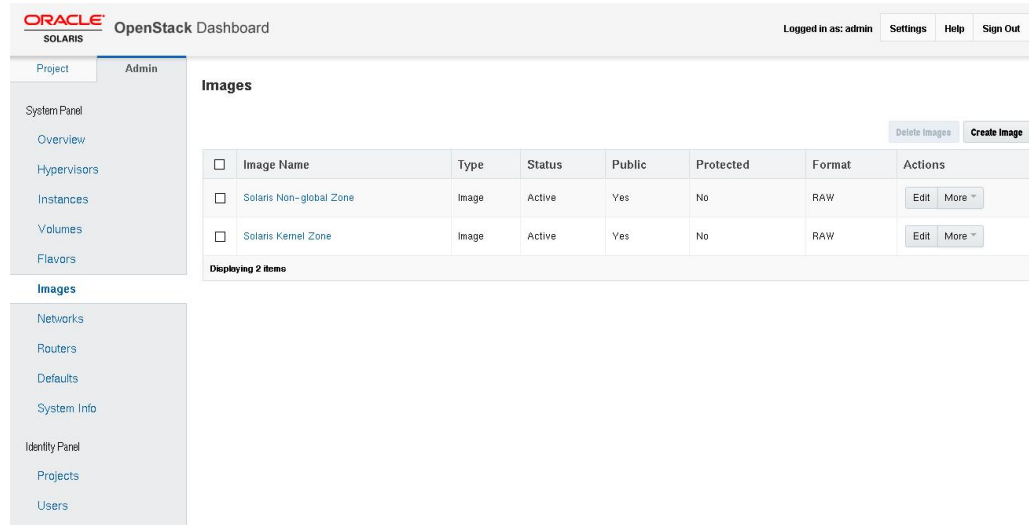


Si instaló OpenStack tal como se describe en “[Instalación de un sistema único](#)” [21], el sistema OpenStack está preconfigurado con los siguientes recursos:

- Dos imágenes: zona no global de Solaris y zona de núcleo de Solaris
- Dos proyectos: demo and service
- Diez tipos

La siguiente figura muestra las imágenes.

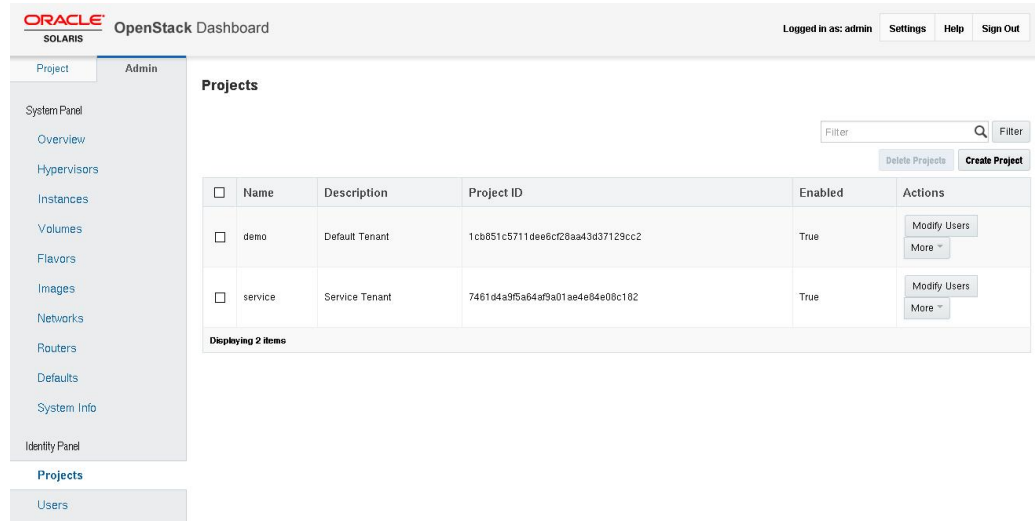
FIGURA 2-2 Pantalla Imágenes en el panel de control de OpenStack



La siguiente figura muestra los proyectos, también conocidos como clientes.

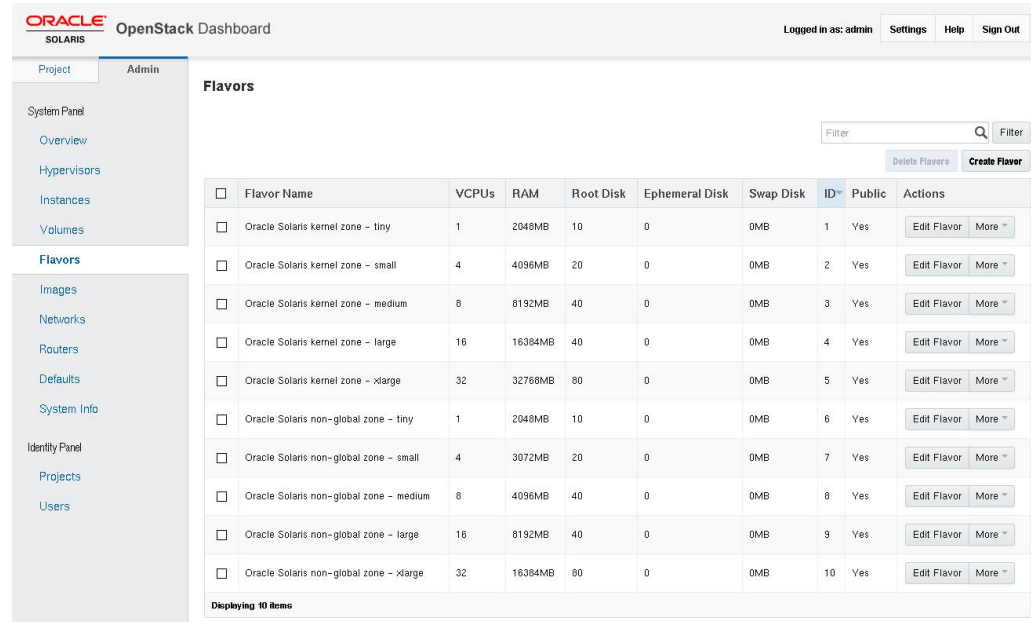
- El cliente demo es el cliente predeterminado. De manera predeterminada, el cliente demo se crea con un único usuario, `admin`, como miembro.
- El inquilino `service` es el que usa el administrador de red para crear recursos que se compartirán entre varios inquilinos. Por ejemplo, en los ejemplos y escenarios en el documento, el enrutador Neutron se crea en el inquilino `service` de modo que el enrutador es compartido por todos los inquilinos. No debe usar el inquilino `service` por ningún otro motivo en la configuración de OpenStack. Los servicios de OpenStack se comunican entre sí mediante usuarios específicos de servicio, que tienen el rol `admin` y forman parte del inquilino de servicio.

FIGURA 2-3 Pantalla Proyectos en el panel de control de OpenStack



La siguiente figura muestra los tipos. Consulte [“Visualización de la información sobre tipos” \[112\]](#) para obtener una lista de texto de tipos.

FIGURA 2-4 Pantalla Tipos en el panel de control de OpenStack



Creación e inicio de una instancia de VM

Use el panel Proyecto del panel de control para crear una instancia de VM.

▼ Cómo crear una instancia de VM mediante el panel de control

Antes de empezar Asegúrese de tener un par de claves SSH. Consulte [Cómo crear un par de claves SSH \[38\]](#).

Asegúrese de tener una red externa definida. Consulte [“Configuración del agente Neutron L3” \[66\]](#).

1. **Haga clic en el separador Proyecto que se encuentra sobre el lado izquierdo del panel de control.**

El proyecto que está usando el usuario se muestra en la parte superior del panel. De manera predeterminada, el usuario admin de esta configuración de OpenStack usa el proyecto demo.

2. **En la sección Gestionar cálculos del panel haga clic en Instancias.**

3. En el lado derecho del panel Instancias, haga clic en el botón Iniciar instancia.

Se mostrará el siguiente cuadro de diálogo Iniciar instancia, que le permite especificar el nombre de la nueva instancia de VM, y seleccionar el tipo de instancia y el tipo de imagen de la instancia.

FIGURA 2-5 Cuadro de diálogo Iniciar instancia

Launch Instance ✕

Details | Access & Security | Networking

Availability Zone
nova

Instance Name

Flavor
Oracle Solaris kernel zone - tiny

Instance Count
1

Instance Boot Source
--- Select source ---

Specify the details for launching an instance.
The chart below shows the resources used by this project.

Flavor Details

Name	Oracle Sola...
VCPUs	1
Root Disk	10 GB
Ephemeral Disk	0 GB
Total Disk	10 GB
RAM	2,048 MB

Project Limits

Number of Instances
[Progress bar]

Number of VCPUs
[Progress bar]

Total RAM
[Progress bar]

Cancel **Launch**

4. En el campo Nombre de instancia, introduzca un nombre para la nueva instancia de VM.

5. Seleccione un tipo de la lista desplegable Tipo.

Si este sistema OpenStack es una zona de núcleo y no un sistema bare-metal, deberá seleccionar un tipo de zona no global.

6. En Origen de inicio de instancia, seleccione Iniciar desde imagen.

Se mostrará una selección de Nombre de imagen. Seleccione la imagen que tenga el mismo tipo de zona que el tipo que seleccionó. Tanto el tipo como la imagen deben ser para una zona no global `solaris` o una zona de núcleo `solaris-kz`. Si este sistema OpenStack es una zona de núcleo y no un sistema bare-metal, deberá seleccionar una imagen de zona no global.

FIGURA 2-6 Cuadro de diálogo Iniciar instancia con selecciones

Launch Instance ✕

Details | Access & Security | Networking

Availability Zone
nova

Instance Name
VM1

Flavor
Oracle Solaris non-global zone - tiny

Instance Count
1

Instance Boot Source
Boot from image

Image Name
Solaris Non-global Zone (1.2 GB)

Specify the details for launching an instance.
The chart below shows the resources used by this project.

Flavor Details

Name	Oracle Sola...
VCPUs	1
Root Disk	10 GB
Ephemeral Disk	0 GB
Total Disk	10 GB
RAM	2,048 MB

Project Limits

Number of Instances
██████████

Number of VCPUs
██████████

Total RAM
██████████

7. **Haga clic en el separador Acceso y seguridad del cuadro de diálogo.**
Seleccione el par de claves SSH que se instalará en la nueva instancia de VM.
8. **Haga clic en el separador Redes del cuadro de diálogo.**
Seleccione la red a la que se deberá conectar la nueva instancia de VM.

9. Haga clic en el botón Iniciar que se encuentra en la parte inferior del cuadro de diálogo.

Se creará, se instalará y se iniciará la nueva instancia de VM.

El tiempo requerido para que una nueva instancia esté disponible depende de varios factores, entre ellos, el tamaño de la imagen, los recursos proporcionados en el tipo y el lugar donde OpenStack ha colocado el sistema de archivos raíz de la nueva instancia de VM.

10. Asocie una dirección IP flotante con la nueva instancia de VM.

Puede realizar estos pasos mientras se instala la nueva instancia de VM. La instancia de VM deberá tener una dirección IP flotante asociada, para que los usuarios puedan iniciar sesión.

a. Haga clic en el botón IP flotante asociada de la columna Acciones.

Se abrirá el cuadro de diálogo Gestionar asociaciones de IP flotante.

b. Seleccione una dirección del menú desplegable Dirección IP.

Si el campo Dirección IP muestra un mensajes que indica que no hay direcciones IP disponibles, haga clic en el botón +. Consulte [Cómo asociar una dirección IP flotante a un proyecto \[39\]](#).

c. Seleccione un puerto asociado.

La lista de puertos muestra las direcciones IP fijas de las instancias de VM.

d. Haga clic en el botón Asociar que se encuentra en la parte inferior del cuadro de diálogo.

Pasos siguientes

- Haga clic en Instancias y haga clic en el nombre de la instancia para ver información detallada acerca de la instancia y ver el registro de la consola de la instancia. Vuelva a cargar la página para ver las actualizaciones del registro.
- Haga clic en Volúmenes para ver los volúmenes de Cinder que se han creado.
- Haga clic en Topología de red para ver una representación de la red de la nube, incluidos todos los segmentos de subred, los enrutadores virtuales y las instancias activas.
- Haga clic en Imágenes e instantáneas para ver los archivos de Unified Archive que se han cargado en el almacén de imágenes de Glance.
- Cuando se haya finalizado la instalación de la nueva instancia de VM y se haya alcanzado el estado Activo, inicie sesión en la instancia. El siguiente comando inicia sesión en la zona como root con la clave del Paso 7 y la dirección IP flotante del Paso 10:

```
# ssh root@floating-ip-address
```

▼ **Cómo crear un par de claves SSH**

1. **Haga clic en el separador Proyecto que se encuentra sobre el lado izquierdo del panel de control.**
2. **En la sección Gestionar cálculos del panel haga clic en Acceso y seguridad.**
3. **Haga clic en el separador Pares de claves.**
4. **Haga clic en el botón Crear par de claves.**
Se mostrará el cuadro de diálogo Crear par de claves.
5. **Introduzca un nombre para el par de claves en el campo Nombre de par de claves.**
6. **Haga clic en el botón Crear par de claves en el cuadro de diálogo.**
El nuevo par de claves se deberá descargar automáticamente.

Si el nuevo par de claves no se descarga automáticamente, haga clic en el enlace Descargar par de claves que se proporciona.

El nuevo par de claves deberá aparecer en el separador Pares de claves del panel Acceso y seguridad.

▼ **Cómo crear una red para un cliente**

1. **Haga clic en el separador Proyecto que se encuentra sobre el lado izquierdo del panel de control.**
2. **En la sección Gestionar red del panel, haga clic en Redes.**
3. **Haga clic en el botón Crear red.**
Se mostrará el cuadro de diálogo Crear red.
4. **En el separador Red, introduzca un nombre para la red en el campo Nombre.**
5. **Proporcione la información solicitada en los separadores Subred y Detalles de subred.**
6. **Haga clic en el botón Crear que se encuentra en la parte inferior del cuadro de diálogo.**
Se mostrarán la nueva red y la nueva subred en el panel Redes.

▼ **Cómo asociar una dirección IP flotante a un proyecto**

1. **Haga clic en el separador Proyecto que se encuentra sobre el lado izquierdo del panel de control.**
2. **En la sección Gestionar cálculos del panel haga clic en Acceso y seguridad.**
3. **Haga clic en el separador IP flotantes.**
4. **Haga clic en el botón Asignar IP a proyecto.**
Se abrirá el cuadro de diálogo Asignar IP flotante.
5. **En el menú desplegable del cuadro de diálogo, seleccione una agrupación desde donde se asignará la IP flotante.**
6. **Haga clic en el botón Asignar IP en el cuadro de diálogo.**

◆◆◆ CAPÍTULO 3

Instalación en varios sistemas para una configuración de Havana OpenStack en varios nodos

En este capítulo, se describe cómo instalar una configuración de OpenStack de varios nodos. Cada nube necesita solamente una instancia del panel de control, un almacén de imágenes y un servicio de identidad. Cada nube puede tener cualquier cantidad de instancias de almacenamiento y cálculo. En un entorno de producción, estos servicios se configuran en varios nodos. Evalúe cada componente en relación con sus necesidades de una implementación de nube específica para determinar si se deberá instalar ese componente en un nodo separado y cuántos nodos de ese tipo necesitará.

Nota - Este capítulo se aplica a una configuración de OpenStack específica para Havana. La versión Juno de OpenStack sólo se admite a partir de la versión Oracle Solaris 11.2 SRU10.

- Para obtener información sobre cómo obtener la versión Oracle Solaris 11.2 SRU10 y actualizar una configuración existente de Havana a Juno, consulte [De Havana a Juno: procedimientos de actualización de OpenStack](#).
- Si está ejecutando la versión Oracle Solaris 11.2 SRU10 y desea realizar una nueva configuración de Juno, consulte el [Capítulo 4, Instalación en varios sistemas para una configuración de Juno de OpenStack en varios nodos](#).

En este capítulo, se describe una arquitectura que se implementa en un sistema físico separado. Para realizar la partición de un solo servidor de Oracle SPARC y configurar OpenStack de varios nodos en el servidor que ejecuta OVM Server for SPARC (LDoms), consulte [Multi-node Solaris 11.2 OpenStack on SPARC Servers](#).

Descripción general de la arquitectura de tres nodos

La arquitectura que se describe en este capítulo se implementa en los siguientes tres sistemas:

- Nodo de controlador. El nodo de controlador es el lugar donde se ejecutan la mayoría de los servicios de OpenStack compartidos y otras herramientas. El nodo de controlador suministra

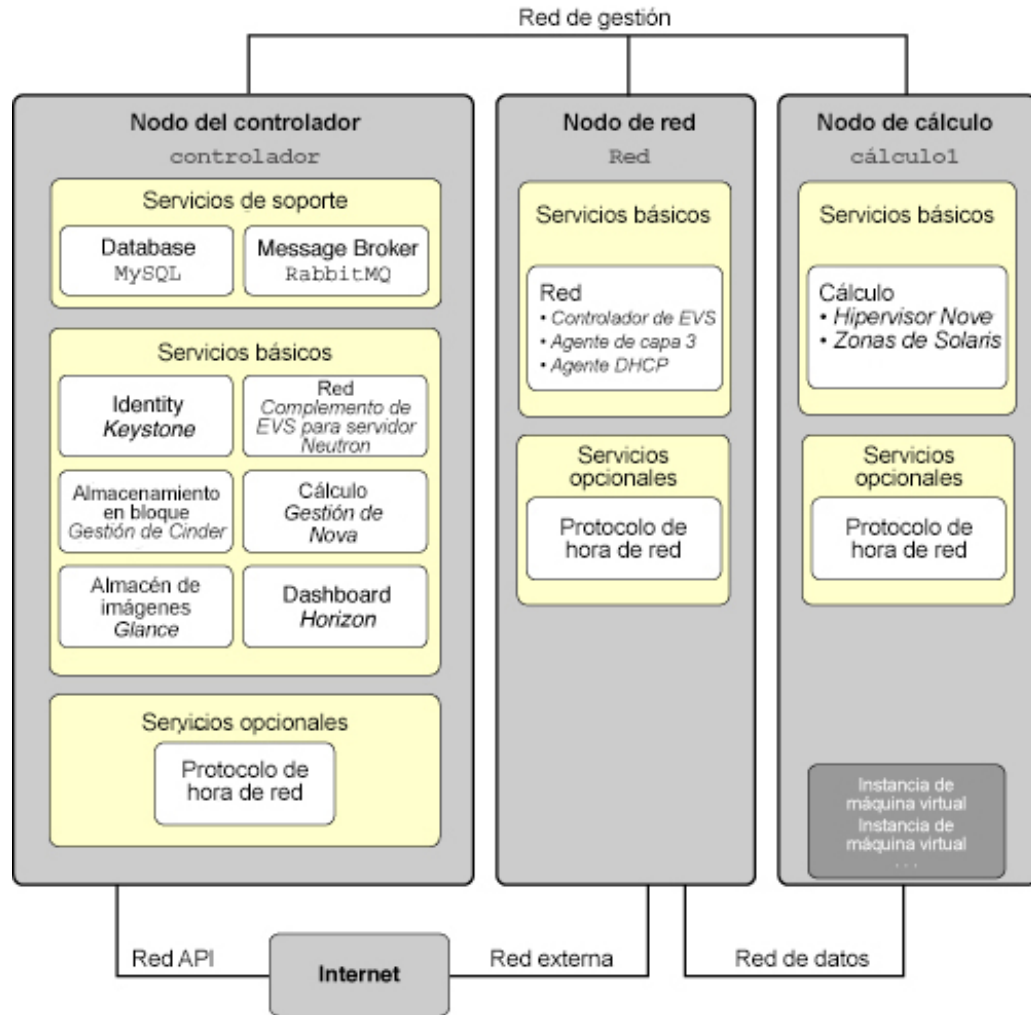
API, programación y otros servicios compartidos para la nube. El nodo de controlador tiene el panel de control, el almacén de imágenes y el servicio de identidad. Además, también se configuran en este nodo el servicio de gestión de cálculo Nova y el servidor Neutron

- **Nodo de red.** El nodo de red proporciona redes virtuales y servicios de red a las instancias de Nova mediante los servicios de red DHCP y Neutron de capa 3.
- **Nodo de cálculo.** El nodo de cálculo es el lugar donde se instalan las instancias de VM (instancias de cálculo de Nova). Las instancias de VM usan los destinos iSCSI proporcionados por el servicio de volúmenes Cinder.

En esta arquitectura, los tres nodos comparten una subred común denominada subred de gestión. El nodo de controlador y cada nodo de cálculo comparten una subred común separada denominada subred de datos. Todos los sistemas están conectados a la red de gestión mediante su interfaz física `net0`. El nodo de red y el nodo de cálculo están conectados a la red de datos mediante sus interfaces físicas `net1`.

La siguiente figura muestra una vista de alto nivel de la arquitectura que se describe en este capítulo.

FIGURA 3-1 Arquitectura de referencia de configuración de tres nodos



La siguiente tabla muestra cuáles servicios SMF relacionados con OpenStack están instalados en cada nodo. La lista muestra la parte más pequeña del nombre de cada servicio SMF que puede usar con comandos como `svcadm`. El nombre de la instancia del servicio SMF se muestra únicamente si el nombre resulta ambiguo sin el nombre de la instancia.

TABLA 3-1 Servicios SMF instalados en los nodos de controlador, red y cálculo

Nodo de controlador	Nodo de red	Nodo de cálculo
mysql	neutron-dhcp-agent	nova-compute
rabbitmq	neutron-l3-agent	ntp
keystone	evs-controller	
cinder-api	ntp	
cinder-db		
cinder-db		
cinder-scheduler		
cinder-volume:default		
cinder-volume:setup		
glance-api		
glance-db		
glance-registry		
glance-scrubber		
neutron-server		
evs		
nova-api-ec2		
nova-api-osapi-compute		
nova-cert		
nova-conductor		
nova-objectstore		
nova-scheduler		
http		
ntp		
heat-api		
heat-db		
heat-api-cfn		
heat-api-cloudwatch		
heat-engine		

Esta arquitectura de ejemplo no muestra el servicio de almacenamiento del objeto de Swift. Para obtener información general acerca de cómo configurar Swift, consulte la información del sitio de la comunidad de OpenStack, como [OpenStack Configuration Reference](#). Para obtener información acerca de cómo configurar los servicios de Swift en los sistemas de Solaris y para obtener otra información acerca de OpenStack en Solaris, consulte [OpenStack for Oracle Solaris 11](#).

Para obtener una lista de los parámetros de configuración de OpenStack que son útiles para las implementaciones de OpenStack en Oracle Solaris, consulte “Parámetros de configuración comunes para OpenStack” en [Getting Started with OpenStack on Oracle Solaris 11.2](#).

Para preparar la implementación de la configuración de OpenStack de tres nodos de ejemplo, asegúrese de tener la siguiente información:

- Dirección IP y nombre de host del nodo de controlador.
- Dirección IP y nombre de host del nodo de red.
- Dirección IP y nombre de host del nodo de cálculo.
- Contraseñas para los distintos usuarios del servicio según sea necesario.

Para la configuración de ejemplo, los nombres de los tres nodos son `controller`, `network` y `compute1`.

Configuración del nodo de controlador

Cada nodo de controlador necesita solamente un servicio del panel de control, un almacén de imágenes y un servicio de identidad. Este nodo también incluye MySQL, RabbitMQ, almacenamiento de bloques y cálculos, y servicios de redes.

Para gestionar mejor el uso de memoria entre ZFS y las aplicaciones de Oracle Solaris 11, establezca el parámetro `usr_reserve_hint_pct` en el nodo, como se muestra en el siguiente ejemplo:

```
# echo "set user_reserve_hint_pct=80" >>/etc/system.d/site:kernel-zones-reserve
# reboot
```

donde *site* puede hacer referencia a la compañía.

También debe establecer este parámetro en los diferentes nodos de OpenStack.

Para obtener más información sobre este parámetro, inicie sesión en su cuenta de MOS en <https://support.oracle.com> y consulte el documento 1663862.1, *Gestión de memoria entre ZFS y aplicaciones en Oracle Solaris 11.2*.

La comunicación entre servicios de OpenStack se realiza mediante el Protocolo avanzado de cola de mensajes (AMQP). En Solaris, AMQP es implementado por RabbitMQ. RabbitMQ

es un servicio necesario. Generalmente, se configura un solo nodo en la nube para ejecutar RabbitMQ. En esta arquitectura, RabbitMQ se configura para ejecución en el nodo de controlador.

▼ Cómo configurar el nodo de controlador

1. (Opcional) Instale y configure NTP.

Consulte [“Instalación del Protocolo de hora de red”](#) [47].

2. (Opcional) Instale y configure MySQL.

Consulte [“Instalación de MySQL”](#) [48].

3. Instale RabbitMQ.

a. Instale el paquete RabbitMQ.

```
controller# pkg install rabbitmq
```

b. Active el servicio SMF de RabbitMQ.

```
controller# svcadm enable rabbitmq
```

4. Instale y configure Keystone.

Consulte [“Instalación de Keystone”](#) [49].

5. Instale y configure Cinder.

Consulte [“Instalación de Cinder”](#) [51].

6. Instale y configure Glance.

Consulte [“Instalación de Glance”](#) [56].

7. Instale y configure Neutron.

Consulte [“Instalación y configuración de Neutron en el nodo del controlador”](#) [57].

8. Instale y configure Nova.

Consulte [“Instalación de Nova”](#) [58].

9. Configure Horizon.

Consulte [Cómo configurar Horizon](#) [59].

Instalación del Protocolo de hora de red

Se recomienda la instalación del Protocolo de hora de red (NTP), aunque es opcional. Si instala NTP, instálelo en cada nodo de servicio de la implementación en la nube.

NTP ayuda a garantizar una hora coherente del día en todos los nodos de servicio de la nube. Si activa NTP en una red, configure los nodos de servicio para obtener la hora por la red.

- Si la multidifusión IP está activada en las subredes IP donde residen los nodos de servicio puede aprovechar la multidifusión IP para configurar NTP.
- Si la multidifusión IP no está activada en las subredes IP donde residen los nodos de servicio, configure NTP manualmente.

▼ Cómo instalar y configurar el protocolo de hora de red

1. Instale el paquete NTP.

```
controller# pkg install ntp
```

2. Instale el archivo de configuración.

```
controller# cp /etc/inet/ntp.client /etc/inet/ntp.conf
```

3. Configure NTP.

La configuración necesaria depende de si la multidifusión IP está activada en las subredes del nodo.

- **Si la multidifusión IP está activada, no se necesita ninguna configuración adicional.**
- **Si la multidifusión IP no está activada, configure los nombres de host o las direcciones IP de los servidores NTP existentes:**

a. Comente la opción `multicastclient` del archivo `/etc/inet/ntp.conf`.

```
# multicastclient 224.0.1.1
```

b. Elimine el comentario de una o más de las opciones del servidor en el archivo `/etc/inet/ntp.conf`:

```
server ntp_server_1 iburst
server ntp_server_2 iburst
```

4. Active el servicio SMF del servidor NTP.

```
controller# svcadm enable ntp
```

Instalación de MySQL

Muchos servicios OpenStack mantienen una base de datos para realizar un seguimiento de los recursos críticos, el uso y otra información. De manera predeterminada, se especifican bases de datos individuales de SQLite para este fin y son útiles para la configuración de un solo nodo. Para configuraciones de varios nodos, se recomienda una base de datos de MySQL para almacenar esta información.

▼ Cómo instalar una base de datos MySQL

1. Determine el nombre primario asociado con el nodo de controlador.

Use la dirección IP primaria del nodo de controlador para determinar el nombre primario asociado con ese nodo.

```
controller# getent hosts controller-IP
controller-IP controller-name
```

2. Instale el paquete de servidor MySQL.

```
controller# pkg install mysql-55
```

3. Instale el paquete de cliente MySQL.

```
controller# pkg install mysql-55/client
```

4. Active el servicio SMF del servidor MySQL.

```
controller# svcadm enable mysql:version_55
```

5. Defina la contraseña root del servidor MySQL.

```
controller# mysqladmin -u root password MySQL-root-password
```

6. Configure el servidor MySQL.

Cree las tablas que usará OpenStack. Conceda privilegios a los servicios del nodo de controlador para proporcionar acceso exclusivo a estas bases de datos. Use el *controller-name* que generó el comando `getent hosts` anterior.

```
controller# mysql -u root -p
Enter password: MySQL-root-password
mysql> create database cinder;
mysql> grant all privileges on cinder.*
-> to 'cinder'@'controller-name'
-> identified by 'cinder';
mysql> create database glance;
mysql> grant all privileges on glance.*
-> to 'glance'@'controller-name'
```



```

-> identified by 'glance';
mysql> create database keystone;
mysql> grant all privileges on keystone.*
-> to 'keystone'@'controller-name'
-> identified by 'keystone';
mysql> create database nova;
mysql> grant all privileges on nova.*
-> to 'nova'@'controller-name'
-> identified by 'nova';
mysql> flush privileges;
mysql> quit

```

7. Instale el paquete de biblioteca de cliente Python de MySQL.

```
controller# pkg install python-mysql
```

Instalación de Keystone

El servicio Keystone se debe estar instalado y configurado en el nodo de controlador.

▼ Cómo instalar y configurar Keystone

1. Instale el paquete Keystone.

```
controller# pkg install keystone
```

2. Modifique el archivo de configuración de Keystone.

Elimine los comentarios y defina los siguientes dos parámetros en el archivo `/etc/keystone/keystone.conf`.

a. Defina el parámetro `admin_token`.

El parámetro `admin_token` es un “secreto compartido” entre Keystone y otros servicios de OpenStack. El valor de este parámetro puede ser cualquier cadena de caracteres, pero este valor no se debe revelar ni distribuir. Una manera de crear esta cadena es usar OpenSSL como se muestra en el siguiente comando:

```
controller# openssl rand -hex 10
random_string
```

Use este valor de salida para definir el parámetro `admin_token` en el archivo `/etc/keystone/keystone.conf`.

```
admin_token = random_string
```

b. Defina el parámetro `connection`.

El parámetro `connection` es un URI que representa la ubicación de la base de datos de Keystone y el tipo de base de datos que se usa.

Use el `controller-name` que generó el comando `getent hosts` anterior para definir el parámetro `connection` en el archivo `/etc/keystone/keystone.conf`.

```
connection = mysql://keystone:keystone@controller-name/keystone
```

3. Genere tokens de Infraestructura de clave pública (PKI).

```
controller# su - keystone -c "keystone-manage pki_setup"
```

4. Active el servicio SMF de Keystone.

```
controller# svcadm enable keystone
```

5. Rellene la base de datos de Keystone.

Puede realizar este paso manualmente o puede usar la secuencia de comandos `sample_data.sh` tal como se muestra en el siguiente ejemplo. Use el `controller-name` que generó el comando `getent hosts` anterior.

```
controller# su - keystone -c "env
CONTROLLER_ADMIN_ADDRESS=controller-name
CONTROLLER_INTERNAL_ADDRESS=controller-name
CONTROLLER_PUBLIC_ADDRESS=controller-name
/usr/demo/openstack/keystone/sample_data.sh"
```

La secuencia de comandos `sample_data.sh` admite variables de entorno que definen el nodo donde reside cada servicio API y la contraseña de cada servicio. Revise la secuencia de comandos para obtener más información acerca de los parámetros que se pueden definir desde el entorno. De manera predeterminada, se crea un usuario de Keystone para cada servicio en el cliente `service`, con una contraseña igual que la del nombre de usuario. Por ejemplo, se crea un usuario `nova` con la contraseña `nova`.

Instalación y configuración de Heat

Heat es el motor de orquestación de OpenStack que le permite desplegar las aplicaciones en la nube basadas en las plantillas de que cree. Heat se instala en el mismo nodo que Keystone.

▼ Cómo configurar Heat

Antes de empezar Primero debe configurar Keystone, como se describe en [Cómo instalar y configurar Keystone](#) antes de realizar esta tarea.

1. Instale el paquete de Heat.

```
controller# pkg install heat
```

2. Ejecute la secuencia de comandos de Heat.

```
# /usr/demo/openstack/keystone/heat-keystone-setup
```

3. Edite `/etc/heat/api-paste.ini` mediante la actualización de la siguiente información en el archivo.

```
# Auth middleware that validates token against keystone
[filter:authtoken]
paste.filter_factory = heat.common.auth_token:filter_factory
auth_uri = http://controller-IP:5000/v2.0
identity_uri = http://controller-IP:35357
admin_tenant_name = keystone
admin_user = heat
admin_password = heat-password
```

4. Active el servicio de Heat.

```
# svcadm enable -rs heat-api heat-db heat-engine heat-api-cfn heat-api-cloudwatch
```

Instalación de Cinder

La configuración de Cinder debe especificar al menos la siguiente información:

- Información de autorización para autenticación con Keystone.
- La clase de volúmenes que se crearán.

▼ Cómo instalar y configurar Cinder

1. Instale el paquete de Cinder.

```
controller# pkg install cinder
```

2. Especifique la información de configuración de autenticación.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/cinder/api-paste.ini`. Estos parámetros especifican la ubicación del servicio API de Keystone y la información de autenticación de Cinder.

```
auth_uri = http://controller-name:5000/v2.0
identity_uri = http://controller-name:35357
admin_tenant_name = service
admin_user = cinder
admin_password = cinder-password
```

3. Especifique las clases de volúmenes que deberá crear el servicio de volúmenes de Cinder.

En el archivo `/etc/cinder/cinder.conf`, elimine los comentarios del parámetro `volume_driver` adecuado. Se admiten las siguientes cuatro clases de volúmenes:

ZFSVolumeDriver

Admite la creación de volúmenes locales para uso de Nova en el mismo nodo que el servicio de volúmenes de Cinder.

ZFSISCSIDriver

Admite la creación y la exportación de destinos de iSCSI para uso por los nodos de cálculo remoto de Nova.

ZFSFCDriver

Admite la creación y la exportación de LUN de canal de fibra para uso por los nodos de cálculo remoto de Nova.

ZFSSAISCSIDriver

Admite la creación y la exportación de destinos de iSCSI desde un Oracle ZFS Storage Appliance para uso por nodos de cálculo remoto de Nova. Los parámetros adicionales de este controlador se deben definir en el archivo `/etc/cinder/cinder.conf`.

En el ejemplo que se describe en este capítulo, iSCSI se usa para proporcionar volúmenes para uso por instancias de Nova. Comente la selección predeterminada de `ZFSVolumeDriver` y elimine los comentarios de la selección de `ZFSISCSIDriver`.

```
# Driver to use for volume creation (string value)
# The local ZFS driver provides direct access to ZFS volumes that it
# creates. The other listed drivers provide access to ZFS volumes via
# iSCSI or Fibre Channel and are suitable for cases where block storage
# for Nova compute instances is shared.
#volume_driver=cinder.volume.drivers.solaris.zfs.ZFSVolumeDriver
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSISCSIDriver
#volume_driver=cinder.volume.drivers.solaris.zfs.ZFSFCDriver
#volume_driver=cinder.volume.drivers.zfssa.zfssaiscsi.ZFSSAISCSIDriver
```

4. Defina parámetros de configuración adicionales.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/cinder/cinder.conf`. Estos parámetros especifican la ubicación del servicio API de Glance API, la base de datos correspondiente para Cinder y el servicio RabbitMQ.

```
glance_host=controller-name
sql_connection=mysql://cinder:cinder@controller-name/cinder
rabbit_host=controller-name
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSISCSIDriver
```

5. Si se configurarán los destinos de iSCSI, active los servicios SMF correspondientes.

```
controller# svcadm enable iscsi/target stmf
```

6. Active los servicios SMF de Cinder.

```
controller# svcadm enable cinder-db
controller# svcadm enable cinder-api cinder-scheduler
controller# svcadm enable cinder-volume:default cinder-volume:setup
```

Véase también Consulte también [How to Build OpenStack Block Storage on ZFS](#).

▼ Cómo configurar el controlador iSCSI de Cinder de ZFS Storage Appliance

El controlador iSCSI de Cinder Oracle ZFS Storage Appliance activa Oracle ZFS Storage Appliance (ZFSSA) para uso ininterrumpido como recurso de almacenamiento de bloques para Cinder. El controlador proporciona la capacidad de crear volúmenes iSCSI que pueden ser asignados por un servidor de Cinder a cualquier instancia de máquina virtual creada por el servicio de Nova. El paquete `cloud/openstack/cinder` entrega el controlador. El dispositivo debe ejecutar al menos la versión 2013.1.2.0 del software ZFSSA.

Antes de empezar Configure una agrupación en Oracle ZFS Storage Appliance. Puede elegir usar una agrupación preexistente.

1. Ejecute el flujo de trabajo `cinder.akwf`.

Puede usar un usuario existente o crear un nuevo usuario con autorizaciones de roles para realizar las operaciones del controlador de Cinder.

El flujo de trabajo de `cinder.akwf` realiza las siguientes tareas:

- Crea el usuario si el usuario no existe.
- Define las autorizaciones de roles para realizar operaciones del controlador de Cinder.
- Activa el servicio RESTful si el servicio está desactivado.

Puede ejecutar el flujo de trabajo desde la Interfaz de línea de comandos (CLI) o desde la Interface de usuario del explorador (BUI) del dispositivo.

■ **Ejecute el flujo de trabajo desde la CLI.**

```
zfssa:maintenance workflows> download
zfssa:maintenance workflows download (uncommitted)> show
Properties:
    url = (unset)
    user = (unset)
```

```
password = (unset)

zfssa:maintenance workflows download (uncommitted)> set url="url to the cinder.akwf
file"
url = "url to the cinder.akwf file"
zfssa:maintenance workflows download (uncommitted)> commit
Transferred 2.64K of 2.64K (100%) ... done

zfssa:maintenance workflows> ls
Properties:
showhidden = false

Workflows:

WORKFLOW    NAME                                     OWNER SETID ORIGIN
VERSION
workflow-000 Clear locks                 root  false Oracle Corporation
1.0.0
workflow-001 Configuration for OpenStack Cinder Driver root  false Oracle Corporation
1.0.0

zfssa:maintenance workflows> select workflow-001

zfssa:maintenance workflow-001 execute (uncommitted)> set name=openstack
name = openstack
zfssa:maintenance workflow-001 execute (uncommitted)> set password=openstack-password
password = *****
zfssa:maintenance workflow-001 execute (uncommitted)> commit
User openstack created.
```

■ **Ejecute el flujo de trabajo desde la BUI.**

- a. **Seleccione Mantenimiento -> Flujos de trabajo y use el ícono más para cargar un nuevo flujo de trabajo.**
- b. **Haga clic en el botón Buscar y seleccione el archivo `cinder.akwf`.**
- c. **Haga clic en el botón CARGAR para terminar de cargar el flujo de trabajo.**
- d. **Haga clic en la nueva fila que aparece en la página Flujos de trabajo de BUI para ejecutar el controlador de Cinder.**

El flujo de trabajo solicita un nombre de usuario y una contraseña. El nombre de usuario y la contraseña también se usarán en el archivo `cinder.conf` como `zfssa_auth_user` y `zfssa_auth_password`.

2. Defina los parámetros en el archivo `cinder.conf`.

Especifique las siguientes propiedades requeridas en el archivo `cinder.conf`:

- `volume_driver` – Asegúrese de que `cinder.volume.drivers.zfssa.zfssaiscsi.ZFSSAISCSIDriver` no tenga comentarios. Asegúrese de que las otras tres selecciones tengan comentarios.
- `zfssa_host`: el nombre o la dirección IP del host de gestión de ZFSSA.
- `zfssa_auth_user`: el nombre de usuario del usuario de Cinder en ZFSSA.
- `zfssa_auth_password`: la contraseña del usuario de Cinder en ZFSSA.
- `zfssa_pool`: la agrupación que se usará para asignar volúmenes.
- `zfssa_target_portal`: el portal de destino iSCSI de ZFSSA (`data-ip:port`). El puerto predeterminado es 3260.
- `zfssa_project`: el nombre del proyecto ZFSSA. Si el proyecto no existe en el dispositivo, el controlador creará un proyecto con ese nombre a la hora de inicio. Este proyecto contiene todos los volúmenes creados por el controlador. Se proporcionan propiedades adicionales de ZFSSA para configuración de características de volumen (por ejemplo, tamaño de bloques) y acceso (por ejemplo, iniciadores, destinos, seguridad).
- `zfssa_initiator_group`: el nombre del grupo de iniciadores. Si el grupo de iniciadores no existe en el dispositivo, el controlador creará un grupo de iniciadores con ese nombre a la hora de inicio. Si desea usar el grupo de iniciadores `default`, defina el valor de este parámetro en `default`. El grupo de iniciadores `default` puede ser útil para fines de evaluación. El grupo de iniciadores `default` no debe usarse normalmente, porque puede causar la exposición de los volúmenes a iniciadores no deseados o conflictivos.
- `zfssa_target_interfaces`: interfaces de red de destino iSCSI de ZFSSA. Use el siguiente comando para ver las interfaces:

```
zfssa:configuration net interfaces> show
```

```
Interfaces:
```

INTERFACE	STATE	CLASS	LINKS	ADDRS	LABEL
e1000g0	up	ip	e1000g0	1.10.20.30/24	Untitled Interface

- `connection`: cambie `connection` por `sql_connection`.

Localice la siguiente línea:

```
connection=mysql://cinder:cinder...
```

Modifique esta línea como se muestra a continuación:

```
sql_connection=mysql://cinder:cinder...
```

3. Asegúrese de que el servicio iSCSI de ZFSSA esté en línea.

Si el servicio iSCSI de ZFSSA no está en línea, use la BUI o la CLI del dispositivo para activarlo. El siguiente ejemplo muestra cómo usar la CLI en el dispositivo:

```
zfssa:> configuration services iscsi
zfssa:configuration services iscsi> enable
```

```
zfssa:configuration services iscsi> show
Properties:
<status> = online
...
```

4. Active los servicios SMF del volumen de Cinder.

```
controller# svcadm enable cinder-volume:default cinder-volume:setup
```

Instalación de Glance

Al igual que con la configuración de Cinder, la configuración de Glance requiere la configuración de determinada información de autenticación, así como información sobre la ubicación de los servicios de MySQL y RabbitMQ.

▼ Cómo instalar y configurar Glance

1. Instale el paquete de Glance.

```
controller# pkg install glance
```

2. Defina los parámetros de configuración de Glance.

Elimine los comentarios y defina los siguientes parámetros en cada uno de los siguientes archivos:

- /etc/glance/glance-api.conf
- /etc/glance/glance-cache.conf
- /etc/glance/glance-registry.conf
- /etc/glance/glance-scrubber.conf

```
auth_uri = http://controller-name:5000/v2.0
identity_uri = http://controller-name:35357
admin_tenant_name = service
admin_user = glance-password
admin_password = glance
```

3. Especifique el URI de la base de datos MySQL.

En ambos archivos, /etc/glance/glance-api.conf y /etc/glance/glance-registry.conf, especifique el URI de la base de datos MySQL:

```
connection=mysql://glance:glance@controller-name/glance
```

4. Especifique la ubicación del servicio RabbitMQ.

En `/etc/glance/glance-api.conf`, defina la ubicación del servicio RabbitMQ.

```
rabbit_host = controller-name
```

5. Active los servicios SMF de Glance.

```
controller# svcadm enable glance-db
controller# svcadm enable glance-api glance-registry glance-scrubber
```

Instalación y configuración de Neutron en el nodo del controlador

En la arquitectura que se describe en este capítulo, el servicio API de Neutron se ejecuta en el nodo de controlador. Para activar este servicio para comunicación con el controlador de EVS instalado en el nodo de red, las claves SSH para el usuario de Neutron del controlador se deben depositar en el archivo `authorized_keys` del usuario `evsuser` en el nodo de red.

▼ Cómo instalar y configurar Neutron

1. Instale el paquete de Neutron.

```
controller# pkg install neutron
```

2. Cree una clave pública SSH en el nodo de controlador para el usuario de neutron.

Esta clave activará el servicio API de Neutron API para acceder al controlador de EVS.

Use el comando `ssh-keygen` como usuario `neutron` y cree la clave para el usuario `neutron`.

```
controller# su - neutron
-c "ssh-keygen -N '' -f /var/lib/neutron/.ssh/id_rsa -t rsa"
```

3. Copie la clave al nodo de controlador EVS.

Copie la clave pública SSH generada en el paso anterior, `/var/lib/neutron/.ssh/id_rsa.pub`, al nodo donde se ejecuta el controlador de EVS. Se hará referencia a esta clave cuando se configure el controlador EVS.

4. Defina los parámetros en el archivo de configuración de Neutron.

Especifique la información de autenticación de Keystone y la ubicación del servicio RabbitMQ. En el archivo `/etc/neutron/neutron.conf`, elimine los comentarios y defina los siguientes parámetros:

```
rabbit_host = controller-name
```

```
auth_uri = http://controller-name:5000/v2.0
identity_uri = http://controller-name:35357
admin_tenant_name = service
admin_user = neutron
admin_password = neutron-password
```

5. Especifique la ubicación del controlador de EVS.

a. Obtenga la ubicación del controlador de EVS.

Use la dirección IP del nodo de red con el comando `getent hosts` para obtener el nombre del nodo de red:

```
network# getent hosts network-IP
network-IP network-name
```

b. Defina la ubicación del controlador de EVS.

En el archivo `/etc/neutron/plugins/evs/evs_plugin.ini`, quite los comentarios de los siguientes parámetros o establezca dichos parámetros. Utilice la salida del comando `getent hosts` para establecer la ubicación del controlador de EVS:

```
evs_controller = ssh://evsuser@network-name
sql_connection = path-to-database
```

6. Active el servicio de servidor Neutron.

```
controller# svcadm enable neutron-server
```

Instalación de Nova

La configuración de Nova en el nodo de controlador también requiere la configuración de la autenticación usual y otra información de servicio.

▼ Cómo instalar y configurar Nova

1. Instale el paquete de Nova.

```
controller# pkg install nova
```

2. Especifique la información de configuración de autenticación.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/nova/api-paste.ini`. Estos parámetros especifican la ubicación del servicio API de Keystone y la información de autenticación de Nova.

```
auth_uri = http://controller-name:5000/v2.0
identity_uri = http://controller-name:35357
admin_tenant_name = service
admin_user = nova
admin_password = nova-password
```

3. Defina parámetros de configuración adicionales.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/nova/nova.conf`. Estos parámetros especifican puntos finales adicionales del servicio de autenticación de Keystone, el servicio API de Glance, el servicio API de Neutron API, el servicio RabbitMQ y el URI de la base de datos para la base de datos específica de Nova.

```
keystone_ec2_url=http://controller-name:5000/v2.0/ec2tokens
glance_host=controller-name
neutron_url=http://controller-name:9696
neutron_admin_username=neutron
neutron_admin_password=neutron-password
neutron_admin_tenant_name=service
neutron_admin_auth_url=http://controller-name:5000/v2.0
rabbit_host=controller-name
connection=mysql://nova:nova@controller-name/nova
```

4. Active los servicios SMF de Nova.

```
controller# svcadm enable nova-conductor
controller# svcadm enable nova-api-ec2 nova-api-osapi-compute
nova-cert nova-conductor nova-objectstore nova-scheduler
```

▼ Cómo configurar Horizon

1. Instale el paquete de Horizon.

```
horizon# pkg install horizon
```

2. Genere certificados para uso de Horizon.

Los siguientes comandos generan certificados autofirmados para uso por Horizon y copian el archivo de configuración del panel de control de OpenStack al directorio de archivos de configuración de Apache. Para obtener más información acerca de la creación de certificados autofirmados, consulte [SSL/TLS Strong Encryption: FAQ](#) de Apache.

```
controller# DASHBOARD=/etc/openstack_dashboard
controller# openssl req -new -x509 -nodes
-out horizon.crt -keyout horizon.key
controller# mv horizon.crt horizon.key ${DASHBOARD}
controller# chmod 0600 ${DASHBOARD}/horizon.*
controller# sed
```

```
-e "/SSLCertificateFile/s:/path.*:${DASHBOARD}/horizon.crt:"
-e "/SSLCACertificateFile/d"
-e "/SSLCertificateKeyFile/s:/path.*:${DASHBOARD}/horizon.key:"
< /etc/apache2/2.2/samples-conf.d/openstack-dashboard-tls.conf
> /etc/apache2/2.2/conf.d/openstack-dashboard-tls.conf
```

3. En el archivo `~/conf.d/openstack-dashboard-tls.conf`, especifique la dirección del sitio y el nombre del servidor del paquete de Horizon con los siguientes parámetros:

```
RedirectPermanent=site-address
ServerName=server-name
```

Nota - En la configuración de tres nodos de ejemplo actual, los dos parámetros especificarían el sistema de nodo de controlador.

4. Realice una de las siguientes operaciones para leer el nuevo archivo de configuración:

- Si el servicio Apache está desactivado, actívelo.

```
controller# svcadm enable apache22
```

- Si el servicio Apache está en línea, reinicielo.

```
controller# svcs apache22
STATE      STIME    FMRI
online     Jul_07   svc:/network/http:apache22
controller# svcadm restart apache22
```

Configuración del nodo de cálculo

El nodo de cálculo es donde se instalan las instancias de VM. Es probable que desee varios nodos de cálculo para la nube.

Para gestionar mejor el uso de memoria entre ZFS y las aplicaciones de Oracle Solaris 11, establezca el parámetro `usr_reserve_hint_pct` en el nodo, como se muestra en el siguiente ejemplo:

```
# echo "set user_reserve_hint_pct=80" >>/etc/system.d/site:kernel-zones-reserve
# reboot
```

donde *site* puede hacer referencia a la compañía.

También debe establecer este parámetro en los diferentes nodos de OpenStack.

Para obtener más información sobre este parámetro, inicie sesión en su cuenta de MOS en <https://support.oracle.com> y consulte el documento 1663862.1, *Gestión de memoria entre ZFS y aplicaciones en Oracle Solaris 11.2*.

▼ Cómo configurar el nodo de cálculo

1. (Opcional) Instale y configure NTP.

Consulte “[Instalación del Protocolo de hora de red](#)” [47].

2. Instale el paquete de Nova.

```
compute1# pkg install nova
```

3. Reinicie el Daemon de acceso remoto (RAD).

Nova usa el RAD para comunicación con la estructura de Oracle Solaris Zones.

```
compute1# svcadm restart rad:local
```

4. Especifique la información de configuración de autenticación.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/nova/api-paste.ini`. Estos parámetros especifican la ubicación del servicio API de Keystone y la información de autenticación de Nova.

```
auth_uri = http://controller-name:5000/v2.0
identity_uri = http://controller-name:35357
admin_tenant_name = service
admin_user = nova
admin_password = nova-password
```

5. Configure los parámetros de autenticación y de base de datos, además de los servicios relevantes.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/nova/nova.conf`. Estos parámetros especifican puntos finales adicionales del servicio de autenticación de Keystone, el servicio API de Glance, el servicio API de Neutron API, el servicio RabbitMQ y el URI de la base de datos para la base de datos específica de Nova.

```
keystone_ec2_url=http://controller-name:5000/v2.0/ec2tokens
glance_host=controller-name
neutron_url=http://controller-name:9696
neutron_admin_username=neutron
neutron_admin_password=neutron-password
neutron_admin_tenant_name=service
neutron_admin_auth_url=http://controller-name:5000/v2.0
rabbit_host=controller-name
connection=mysql://nova:nova@controller-name/nova
```

6. Instale el paquete EVS en el nodo de cálculo.

```
# pkg install evs
```

7. Cree una clave pública SSH en el nodo de cálculo para el usuario de root.

Esta clave activará la estructura Solaris Zones para acceder al controlador de EVS.

Use el comando `ssh-keygen` como usuario `root` y cree la clave para el usuario `root`.

```
compute1# su - root -c "ssh-keygen -N '' -f /root/.ssh/id_rsa -t rsa"
```

8. Copie la clave al nodo de controlador EVS.

Copie la clave pública SSH generada en el paso anterior, `/root/.ssh/id_rsa.pub`, al nodo donde se ejecuta el controlador EVS, que es el nodo de red de esta configuración. Se hará referencia a esta clave cuando se configure el controlador EVS en el nodo de red.

En este paso, puede registrar las imágenes Glance. Consulte [“Creación de una imagen” \[115\]](#) y [“Agregación de una imagen al almacén de imágenes” \[116\]](#).

9. Active el servicio de cálculo de Nova.

```
compute1# svcadm enable nova-compute
```

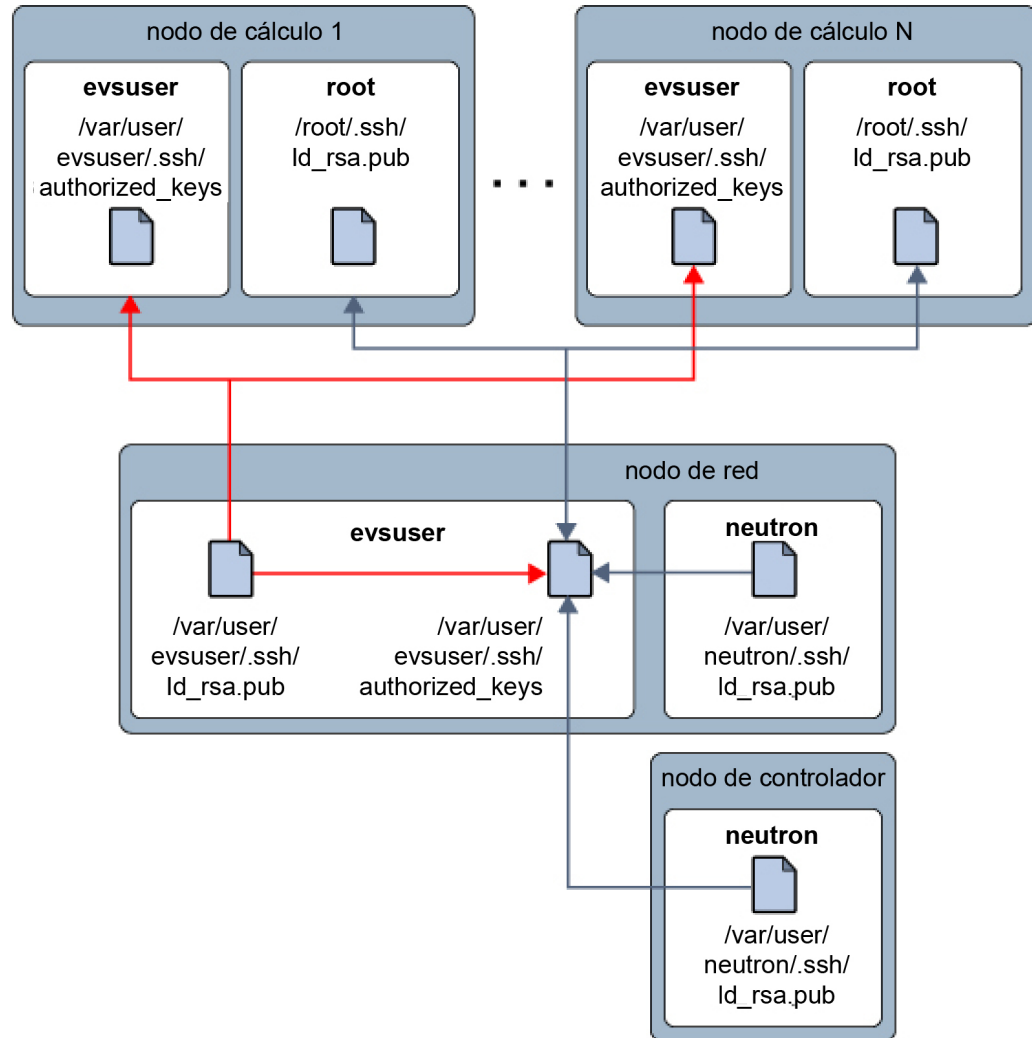
Configuración del nodo de red

La configuración del nodo de red requiere la configuración del conmutador virtual elástico (EVS) y del agente DHCP de Neutron. Si lo desea, puede configurar el agente Neutron de capa 3 (L3).

EVS forma el backend para las redes de OpenStack y facilita la comunicación entre las instancias de VM, ya sea si usan redes VLAN o VXLAN. Las instancias de VM pueden estar en el mismo nodo de cálculo o en varios nodos de cálculo. Para obtener más información acerca de EVS, consulte [Capítulo 5, “Acerca de los conmutadores virtuales elásticos” de “Gestión de virtualización de red y recursos de red en Oracle Solaris 11.2”](#).

Quando configure el nodo de red, asegúrese de copiar las claves públicas SSH de `evsuser` en cada archivo `authorized_keys` de `evsuser` en el nodo de cálculo y en el nodo de red. Consulte la imagen a continuación que muestra la distribución de las claves SSH públicas. En la imagen, se asume que se han configurado varios nodos de cálculo.

FIGURA 3-2 Distribución de claves SSH en controlador EVS



Para gestionar mejor el uso de memoria entre ZFS y las aplicaciones de Oracle Solaris 11, establezca el parámetro `usr_reserve_hint_pct` en el nodo, como se muestra en el siguiente ejemplo:

```
# echo "set user_reserve_hint_pct=80" >>/etc/system.d/site:kernel-zones-reserve
# reboot
```

donde `site` puede hacer referencia a la compañía.

También debe establecer este parámetro en los diferentes nodos de OpenStack.

Para obtener más información sobre este parámetro, inicie sesión en su cuenta de MOS en <https://support.oracle.com> y consulte el documento 1663862.1, *Gestión de memoria entre ZFS y aplicaciones en Oracle Solaris 11.2*.

▼ Cómo configurar el nodo de red

1. (Opcional) Instale y configure NTP.

Consulte “[Instalación del Protocolo de hora de red](#)” [47].

2. Instale el paquete de Neutron.

```
network# pkg install neutron
```

3. Cree claves públicas SSH en el nodo de red para los usuarios neutron y evsuser.

```
network# su - neutron \  
-c "ssh-keygen -N '' -f /var/lib/neutron/.ssh/id_rsa -t rsa"
```

```
network# su - neutron \  
-c "ssh-keygen -N '' -f /var/lib/neutron/.ssh/id_rsa -t rsa"
```

4. Combine los archivos de claves con el archivo de claves autorizadas de EVS.

Combine la claves públicas SSH creadas anteriormente desde los nodos de controlador y cálculo con estas dos nuevas claves y concatene la combinación del archivo de claves autorizadas para evsuser.

```
network# cat \  
path-to-neutron@controller/id_rsa.pub \  
\  
path-to-root@compute1/id_rsa.pub \  
/var/lib/neutron/.ssh/id_rsa.pub \  
/var/user/evsuser/.ssh/id_rsa.pub \  
>> /var/user/evsuser/.ssh/authorized_keys
```

5. Copie el /var/user/evsuser/.ssh/id_rsa.pub de la clave SSH pública de evsuser en cada archivo authorized_keys de evsuser del nodo del cálculo y nodo de red.

Consulte la [Figura 3-2, “Distribución de claves SSH en controlador EVS”](#) para obtener una descripción general de las claves SSH públicas de evsuser en el contexto de la configuración de nube general.

6. Verifique que la conectividad SSH funcione correctamente.

Use el comando ssh para conectarse al controlador de EVS y responda yes cuando se le solicite cada conexión. Efectúe este paso en todos los nodos de la arquitectura de OpenStack.


```

controller# su - neutron -c "ssh evsuser@network-name whoami"
compute1# su - root -c "ssh evsuser@network-name whoami"
network# su - neutron -c "ssh evsuser@network-name whoami"
network# su - root -c "ssh evsuser@network-name whoami"

```

7. Instale el paquete del controlador de EVS.

```
network# pkg install rad-evs-controller
```

8. Reinicie el RAD.

```
network# svcadm restart rad:local
```

9. Configure el controlador de EVS.

Defina si la red virtual en la que se comunicarán las instancias de Nova es una VLAN o una VXLAN. Este ejemplo de abajo realiza la siguiente configuración de VLAN:

- Un VLAN con ID 13 se configura para una red externa que se usa para conectarse fuera de la nube.
- El rango de ID de VLAN ID de 1000 a 2000 se configura para las subredes creadas por EVS.

```

network# evsadm set-prop -p controller=ssh://evsuser@network-name
network# evsadm
network# evsadm set-controlprop -p l2-type=vlan
network# evsadm set-controlprop -p uplink-port=net1
network# evsadm set-controlprop -p vlan-range=13,1000-2000

```

10. Configure los parámetros de autenticación y la ubicación del servicio RabbitMQ.

Elimine los comentarios y defina los siguientes parámetros en el archivo `/etc/neutron/neutron.conf`. Estos parámetros especifican puntos finales adicionales del servicio de autenticación de Keystone, el servicio API de Glance, el servicio API de Neutron API, el servicio RabbitMQ y el URI de la base de datos para la base de datos específica de Neutron.

```

[keystone_authtoken]
signing_dir = /var/lib/neutron/keystone-signing
auth_host = 127.0.0.1
auth_port = 35357
auth_protocol = http
auth_uri = http://controller-IP:5000/v2.0
admin_tenant_name = service
admin_user = neutron
admin_password = neutron-password
identity_uri = http://controller-IP:35357
.
[DEFAULT]
core_plugin = neutron.plugins.evs.plugin.EVSNeutronPluginV2
allow_overlapping_ips = False

quotas]

```

```
quota_driver = neutron.plugins.evs.db.quotas_db.EVSDbQuotaDriver
```

11. Configure el agente DHCP de Neutron.

Elimine los comentarios y defina el siguiente parámetro en el archivo `/etc/neutron/dhcp_agent.ini` para especificar la ubicación del controlador de EVS.

```
evs_controller = ssh://evsuser@network-name
```

12. (Opcional) Especifique el dominio predeterminado que se usará para resolver las solicitudes de DNS de las instancias de Nova.

En el archivo `/etc/neutron/dhcp_agent.ini`, elimine los comentarios y defina el parámetro `dhcp_domain` en el dominio predeterminado que se usará para resolver las solicitudes DNS de las instancias de Nova.

13. Active el agente DHCP.

```
network# svcadm enable neutron-dhcp-agent
```

14. Configure el agente Neutron L3.

Este paso es opcional pero aconsejable.

Configuración del agente Neutron L3

Esta sección muestra cómo crear una red virtual que representa una red externa. En esta red virtual, DHCP no se usa. En su lugar, se crean direcciones IP flotantes. Estas direcciones IP flotantes se asignan a un cliente particular y se pueden asignar a instancias de VM de Nova para uso por usuarios con ese cliente. El agente Neutron L3 crea automáticamente asignaciones NAT de uno a uno entre las direcciones asignadas a instancias de Nova y las direcciones IP flotantes.

La implementación de Oracle Solaris 11.2 de OpenStack Neutron admite un enrutador de proveedor con un modelo de implementación de redes privadas. Para obtener más información sobre este modelo, consulte los casos de uso de Neutron de OpenStack en [Chapter 7 of the Operator Training Guide](#). En este modelo de implementación, cada cliente tendrá una o más redes privadas y todas las redes cliente compartirán el mismo enrutador. El administrador de datos crea, posee y gestiona el enrutador. El enrutador no está visible en la vista de topología de red del cliente. Dado que hay solamente un único enrutador, las redes cliente no pueden usar direcciones IP superpuestas. El administrador crea las redes privadas en nombre de los inquilinos.

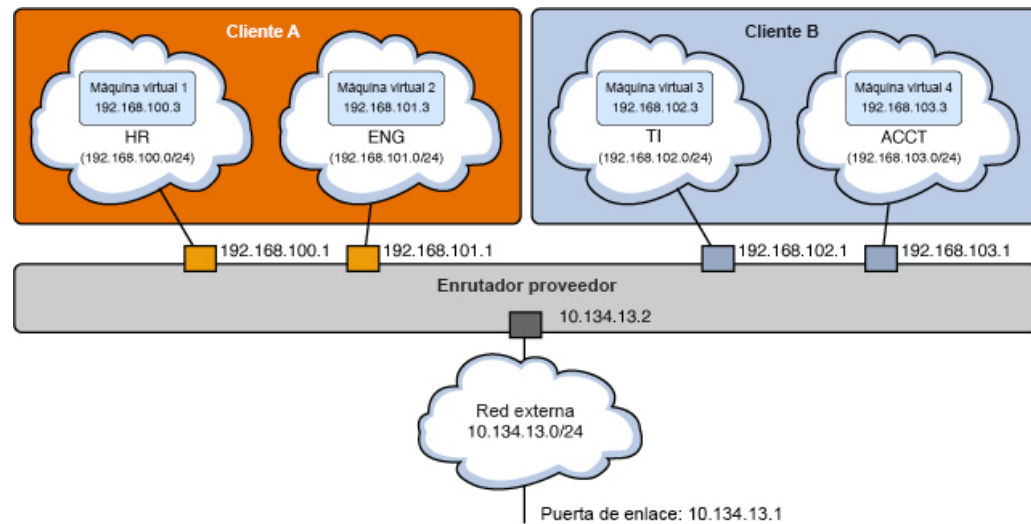
De manera predeterminada, el enrutador de este modelo evita el enrutamiento entre las redes privadas que son parte del mismo cliente: las instancias de VM dentro de una red privada no se pueden comunicar con las instancias de VM de otra red privada, aunque sean todas parte del mismo cliente. Para cambiar este comportamiento, defina

`allow_forwarding_between_networks` en True en el archivo de configuración `/etc/neutron/l3_agent.ini` y reinicie el servicio SMF `neutron-l3-agent`.

El enrutador de este modelo proporciona conectividad al exterior para las instancias de VM del cliente. El enrutador realiza NAT bidireccional en la interface que conecta el enrutador a la red externa. Los clientes crean tantas IP flotantes (IP públicas) como necesitan o como permite la cuota de IP flotante y, a continuación, asocian estas IP flotantes con las instancias de VM que requieren conectividad externa.

La siguiente figura ilustra el modelo de implementación de Oracle Solaris 11.2. Debajo de la figura se muestra una descripción.

FIGURA 3-3 Enrutador de proveedor con modelo de redes privadas



En el modelo que se muestra en la figura anterior, cada cliente tiene dos redes internas y dos instancias de VM.

- | | |
|---------|--|
| Cliente | <ul style="list-style-type: none"> ■ Red HR con subred 192.168.100.0/24 y puerta de enlace 192.168.100.1 ■ Red ENG con subred 192.168.101.0/24 y puerta de enlace 192.168.101.1 ■ VM1 conectado a HR con una dirección IP fija 192.168.100.3 ■ VM2 conectado a ENG con una dirección IP fija 192.168.101.3 |
| Cliente | <ul style="list-style-type: none"> ■ Red IT con subred 192.168.102.0/24 y puerta de enlace 192.168.102.1 |

- Red ACCT con subred 192.168.103.0/24 y puerta de enlace 192.168.103.1
- VM3 conectado a IT con una dirección IP fija 192.168.102.3
- VM4 conectado a ACCT con una dirección IP fija 192.168.103.3

La tabla NAT bidireccional se configura como se muestra en la siguiente tabla.

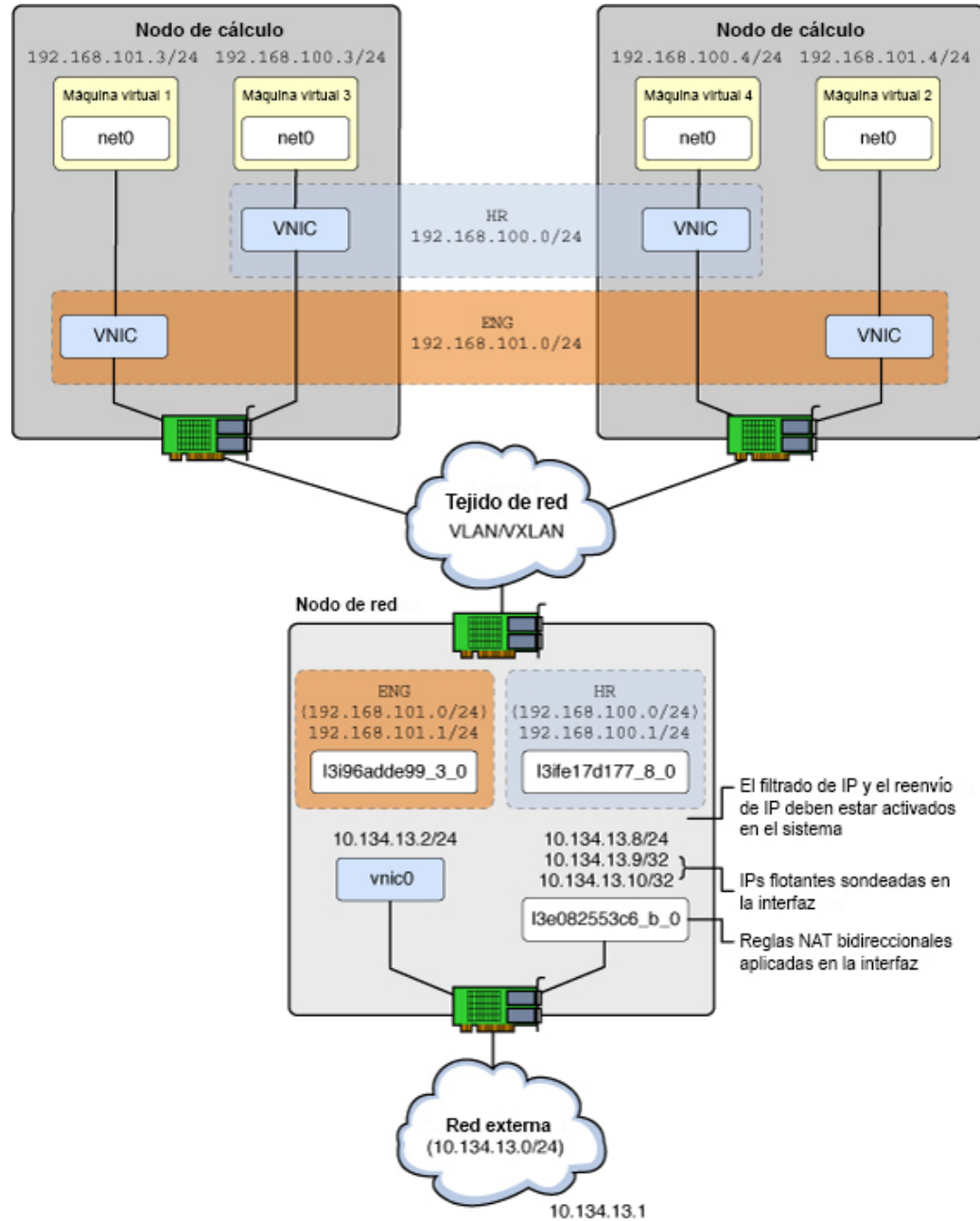
IP Interna	IP flotante
192.168.100.3	10.134.13.40
192.168.101.3	10.134.13.9

Todas las instancias de las interfaces de puerta de enlace se crean en el nodo que se ejecuta en el servicio SMF de `neutron-l3-agent`. Aunque una nube puede tener varias instancias de Neutron, necesita solamente un `neutron-l3-agent` por nube.

La red externa es una red de proveedor asociada con la subred 10.134.13.0/24, a la que se puede acceder desde el exterior. Los clientes crearán direcciones IP flotantes desde esta red y las asociarán a sus instancias de VM. VM1 y VM2 tienen las IP flotantes 10.134.13.40 y 10.134.13.9 asociadas, respectivamente. Se puede acceder a VM1 y VM2 desde el exterior mediante estas direcciones IP.

El siguiente diagrama muestra cómo se configuran estos recursos de cliente en un nodo de red y dos nodos de cálculo. Debajo de la figura, se muestran descripciones de algunas de las funciones de la figura.

FIGURA 3-4 Configuración del agente Neutron L3



VNIC	Interfaz de red virtual.
l3e...	El agente L3 agent creó una VNIC en la red externa (“e”), donde se produce la NAT bidireccional.
l3i...	El agente L3 creó una VNIC en la red interna (“i”) que tiene la dirección IP de puerta de enlace predeterminada.

La siguiente lista muestra cómo se usarán las direcciones IP en esta configuración de ejemplo.

10.134.13.1

Puerta de enlace predeterminada

10.134.13.2 - 10.134.13.7

Direcciones IP reservadas para mostrar API de OpenStack (como Nova, Cinder, Glance) a los clientes

10.134.13.9 - 10.134.13.254

Direcciones IP flotantes para las instancias de VM de cliente

▼ Cómo configurar el agente Neutron L3

Este procedimiento muestra cómo usar el cliente `service` para crear un enrutador, una red externa o una subred externa que será usada por todos los clientes del centro de datos.

El administrador del centro de datos lleva a cabo este procedimiento. Debe usar la línea de comandos para configurar el enrutador único compartido y asociar redes y subredes de distintos inquilinos porque el panel de control de OpenStack le permite gestionar los recursos solamente un inquilino uno por vez.

Consulte [Figura 3-4, “Configuración del agente Neutron L3”](#) a medida que realiza los siguientes pasos.

Antes de empezar Debe realizar la configuración de las redes internas antes de realizar este procedimiento.

- 1. Active la funcionalidad de filtro IP de Solaris.**

```
network# svcadm enable ipfilter
```

- 2. Active el reenvío de IP en todo el host.**

```
network# ipadm set-prop -p forwarding=on ipv4
network# ipadm set-prop -p forwarding=on ipv6
```

- 3. Asegúrese de que EVS esté configurado correctamente y de que tenga el ID de VLAN necesario para la red externa.**

En el siguiente ejemplo, el ID de VLAN y los valores de rango se basan en la configuración de EVS completada en el [Paso 9](#).

```
network# evsadm show-controlprop -p vlan-range,l2-type
PROPERTY    PERM    VALUE      DEFAULT    HOST
l2-type     rw     vlan       vlan       --
vlan-range  rw     13,1000-2000  --       --
```

4. Asegúrese de que exista el cliente service.

```
network# keystone tenant-list
```

5. Cree el enrutador de proveedor.

El enrutador de proveedor se crea como usuario neutron de OpenStack en el cliente service.

Observe el UUID id) del nuevo enrutador. Lo usará en el próximo paso.

```
network# export OS_USERNAME=neutron
network# export OS_PASSWORD=neutron-password
network# export OS_TENANT_NAME=service
network# export OS_AUTH_URL=http://controller-name:5000/v2.0
network# neutron router-create provider_router
Created a new router:
+-----+-----+
| Field                | Value                                     |
+-----+-----+
| admin_state_up       | True                                     |
| external_gateway_info |                                           |
| id                   | 181543df-40d1-4514-ea77-fddd78c389ff    |
| name                 | provider_router                         |
| status               | ACTIVE                                   |
| tenant_id            | f164220cb02465db929ce520869895fa       |
+-----+-----+
```

6. Actualice el archivo de configuración del agente L3.

Actualice el valor de `router_id` en el archivo `/etc/neutron/l3_agent.ini`, con el UUID id) del enrutador del paso anterior.

```
router_id = 181543df-40d1-4514-ea77-fddd78c389ff
```

7. Active el servicio SMF de neutron-l3-agent.

```
network# svcadm enable neutron-l3-agent
```

8. Cree una red externa.

La red virtual se crea como usuario neutron de OpenStack en el cliente service.

```
network# neutron net-create --provider:network_type=vlan
--provider:segmentation_id=13 --router:external=true external_network
Created a new network:
```

```

+-----+-----+
| Field          | Value          |
+-----+-----+
| admin_state_up | True           |
| id             | f67f0d72-0ddf-11e4-9d95-e1f29f417e2f |
| name           | external_network |
| provider:network_type | vlan          |
| provider:segmentation_id | 13           |
| router:external | True           |
| shared         | False          |
| status         | ACTIVE         |
| subnets       |                |
| tenant_id      | f164220cb02465db929ce520869895fa |
+-----+-----+

```

9. Asocie una subred a la red externa.

Cree una subred asociada para la red externa, especifique que DHCP está desactivado y que la subred se usará para enrutamiento externo. Especifique una subred IP enrutable desde donde se asignarán las direcciones IP flotantes. En este ejemplo, la subred está asociada con el ID VLAN 13.

```

network# neutron subnet-create --enable-dhcp=False \
--allocation-pool start=10.134.13.8,end=10.134.13.254 \
--name external_subnet external_network 10.134.13.0/24
Created a new subnet:

```

```

+-----+-----+
| Field          | Value          |
+-----+-----+
| allocation_pools | {"start": "10.134.13.8", "end": "10.134.13.254"} |
| cidr            | 10.134.13.0/24 |
| dns_nameservers |                |
| enable_dhcp     | False          |
| gateway_ip      | 10.134.13.1    |
| host_routes     |                |
| id             | 5d9c8958-0de0-11e4-9d96-e1f29f417e2f |
| ip_version      | 4              |
| name            | external_subnet |
| network_id      | f67f0d72-0ddf-11e4-9d95-e1f29f417e2f |
| tenant_id      | f164220cb02465db929ce520869895fa |
+-----+-----+

```

10. Agregue la red externa al enrutador.

En el siguiente comando, el primer UUID es el UUID `provider_router` y el segundo UUID es el UUID `external_network`.

```

network# neutron router-gateway-set
181543df-40d1-4514-ea77-fddd78c389ff
f67f0d72-0ddf-11e4-9d95-e1f29f417e2f
Set gateway for router 181543df-40d1-4514-ea77-fddd78c389ff
network# neutron router-list -c name -c external_gateway_info
+-----+-----+

```



```
| name          | external_gateway_info          |
+-----+-----+-----+
| provider_router | {"network_id": "f67f0d72-0ddf-11e4-9d95-e1f29f417e2f"} |
+-----+-----+-----+
```

11. Agregue las redes privadas del cliente al enrutador.

Las redes que muestra el comando `neutron net-list` se configuraron anteriormente.

```
network# keystone tenant-list
+-----+-----+-----+
| id          | name | enabled |
+-----+-----+-----+
| 511d4cb9ef6c40beadc3a664c20dc354 | demo | True |
| f164220cb02465db929ce520869895fa | service | True |
+-----+-----+-----+
network# neutron net-list --tenant-id=511d4cb9ef6c40beadc3a664c20dc354
+-----+-----+-----+
| id          | name | subnets |
+-----+-----+-----+
| c0c15e0a-0def-11e4-9d9f- | HR | c0c53066-0def-11e4-9da0- |
| e1f29f417e2f | | e1f29f417e2f 192.168.100.0/24 |
| ce64b430-0def-11e4-9da2- | ENG | ce693ac8-0def-11e4-9da3- |
| e1f29f417e2f | | e1f29f417e2f 192.168.101.0/24 |
+-----+-----+-----+
```

En el siguiente comando, el primer UUID es el UUID `provider_router` y el segundo UUID es el UUID de la subred HR.

```
network# neutron router-interface-add
181543df-40d1-4514-ea77-fddd78c389ff
c0c53066-0def-11e4-9da0-e1f29f417e2f (HR subnet UUID)
Added interface 7843841e-0e08-11e4-9da5-e1f29f417e2f to router 181543df-40d1-4514-ea77-
fddd78c389ff.
```

En el siguiente comando, el primer UUID es el UUID `provider_router` y el segundo UUID es el UUID de la subred ENG.

```
network# neutron router-interface-add
181543df-40d1-4514-ea77-fddd78c389ff
ce693ac8-0def-11e4-9da3-e1f29f417e2f
Added interface 89289b8e-0e08-11e4-9da6-e1f29f417e2f to router 181543df-40d1-4514-ea77-
fddd78c389ff.
```

- Véase también
- [Cómo observar la configuración de agente L3 \[75\]](#)
 - [“Limitaciones conocidas” \[119\]](#)

▼ **Cómo crear y asociar direcciones IP flotantes como usuario cliente**

Un usuario cliente realiza esta procedimiento mediante el panel de control de OpenStack Horizon.

1. Inicie sesión en el panel de control de OpenStack.

Inicie sesión tal como se describe en [Cómo acceder al panel de control de OpenStack \[28\]](#), con las credenciales del usuario cliente.

2. Seleccione Proyecto -> Acceso y seguridad -> IP flotantes.

3. Seleccione external_network.

4. Haga clic en el botón Asignar IP.

El separador IP flotantes muestra que la dirección IP flotante 10.134.13.9 está asignada.

5. Haga clic en el botón Asociar.

6. En el menú desplegable, seleccione el puerto de la instancia de VM.

La ventana Proyecto -> Instancias muestra que la dirección IP flotante está asociada con la instancia de VM.

Si seleccionó un par de claves (clave pública SSH) cuando inició una instancia de VM, entonces esa clave SSH se agregará al archivo `authorized_keys` para el usuario `root` en la instancia de VM.

7. Inicie sesión en la instancia de VM en ejecución.

```
global# ssh root@10.134.13.9
Last login: Fri Jul 18 00:37:39 2014 from 10.132.146.13
Oracle Corporation SunOS 5.11 11.2 June 2014
root@host-192-168-101-3:~# uname -a
SunOS host-192-168-101-3 5.11 11.2 i86pc i386 i86pc
root@host-192-168-101-3:~# zoneadm list -cv
ID NAME STATUS PATH BRAND IP
2 instance-00000001 running / solaris excl
root@host-192-168-101-3:~# ipadm
NAME CLASS/TYPE STATE UNDER ADDR
lo0 loopback ok -- --
lo0/v4 static ok -- 127.0.0.1/8
lo0/v6 static ok -- ::1/128
net0 ip ok -- --
net0/dhcp inherited ok -- 192.168.101.3/24
```

▼ Cómo observar la configuración de agente L3

Puede usar comandos de filtro IP, como `ipf`, `ippool` y `ipnat`, y comandos de redes, como `dladm` y `ipadm`, para observar y resolver el problema de la configuración realizada por `neturon-l3-agent`.

1. Muestre las VNIC creadas por `neutron-l3-agent`.

```
network# dladm show-vnic
LINK          OVER          SPEED  MACADDRESS      MACADDRTYPE  VIDS
l3i7843841e_0_0  net1          1000   2:8:20:42:ed:22  fixed        200
l3i89289b8e_0_0  net1          1000   2:8:20:7d:87:12  fixed        201
l3ed527f842_0_0  net0          100    2:8:20:9:98:3e   fixed
```

2. Muestre las direcciones IP creadas por `neutron-l3-agent`.

```
network# ipadm
NAME          CLASS/TYPE  STATE  UNDER  ADDR
l3ed527f842_0_0  ip          ok     --     --
  l3ed527f842_0_0/v4  static     ok     --     10.134.13.8/24
  l3ed527f842_0_0/v4a static     ok     --     10.134.13.9/32
l3i7843841e_0_0  ip          ok     --     --
  l3i7843841e_0_0/v4  static     ok     --     192.168.100.1/24
l3i89289b8e_0_0  ip          ok     --     --
  l3i89289b8e_0_0/v4  static     ok     --     192.168.101.1/24
```

3. Muestre las reglas de filtro IP.

```
network# ipfstat -io
empty list for ipfilter(out)
block in quick on l3i7843841e_0_0 from 192.168.100.0/24 to pool/4386082
block in quick on l3i89289b8e_0_0 from 192.168.101.0/24 to pool/8226578
network# ippool -l
table role = ipf type = tree number = 8226578
{ 192.168.100.0/24; };
table role = ipf type = tree number = 4386082
{ 192.168.101.0/24; };
```

4. Muestre las reglas de IP NAT.

```
network# ipnat -l
List of active MAP/Redirect filters:
bimap l3ed527f842_0_0 192.168.101.3/32 -> 10.134.13.9/32
List of active sessions:
BIMAP 192.168.101.3 22 <- -> 10.134.13.9 22 [10.132.146.13 36405]
```


◆◆◆ 4 C A P Í T U L O 4

Instalación en varios sistemas para una configuración de Juno de OpenStack en varios nodos

En este capítulo, se describe cómo instalar la configuración de OpenStack en varios nodos. Para obtener información sobre una instalación en un solo nodo, consulte el [Capítulo 2, Instalación de una configuración de evaluación](#).

En este capítulo, se tratan los siguientes temas:

- “Descripción general de la arquitectura de tres nodos” [77]
- “Configuración del nodo de controlador” [81]
- “Configuración del nodo de cálculo” [98]
- “Configuración del nodo de almacenamiento” [100]

Nota - Este capítulo se aplica a una configuración de OpenStack específica para Juno que sólo se admite a partir de la versión Oracle Solaris 11.2 SRU10.

- Para obtener información sobre cómo obtener la versión Oracle Solaris 11.2 SRU10 y actualizar una configuración existente de Havana a Juno, consulte [De Havana a Juno: procedimientos de actualización de OpenStack](#).
 - Si no está ejecutando la versión Oracle Solaris 11.2 SRU10, utilice las instrucciones de configuración de Havana. Consulte el [Capítulo 3, Instalación en varios sistemas para una configuración de Havana OpenStack en varios nodos](#).
-

Descripción general de la arquitectura de tres nodos

Por lo general, se instala y se configura OpenStack en varios sistemas o nodos. Las configuraciones en un solo nodo son útiles para probar OpenStack como producto y para familiarizarse con sus funciones. Sin embargo, la configuración en un solo nodo no es adecuada en un entorno de producción.

Cada nube necesita solamente una instancia del panel de control, un almacén de imágenes y un servicio de identidad. Cada nube puede tener cualquier cantidad de instancias de almacenamiento y cálculo. En un entorno de producción, estos servicios se configuran en varios nodos. Evalúe cada componente en relación con sus necesidades de una implementación de nube específica para determinar si se deberá instalar ese componente en un nodo separado y cuántos nodos de ese tipo necesitará.

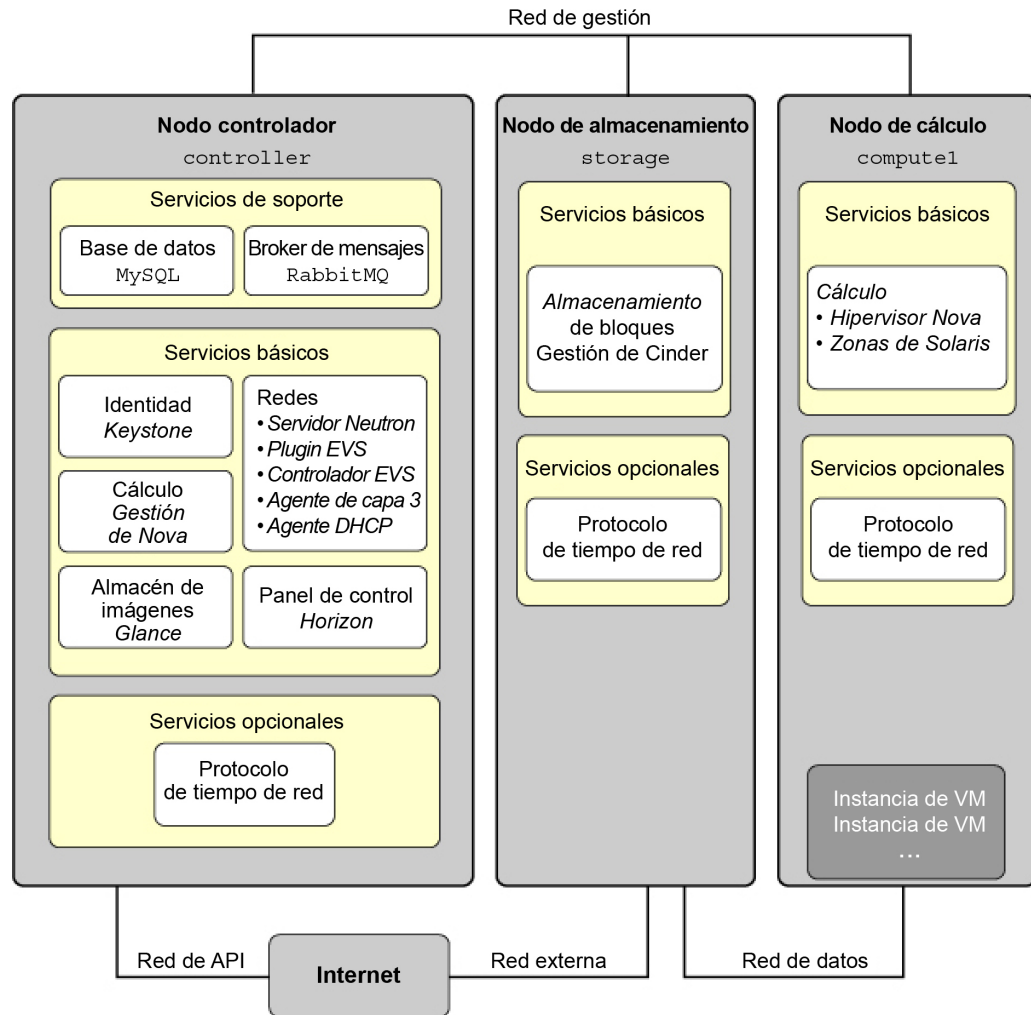
- **Nodo de controlador:** nodo donde se ejecuta la mayoría de los servicios de OpenStack compartidos y otras herramientas. El nodo de controlador suministra API, programación y otros servicios compartidos para la nube. El nodo de controlador tiene el panel de control, el almacén de imágenes y el servicio de identidad. Además, también se configuran en este nodo el servicio de gestión de cálculo Nova y el servidor Neutron
- **Nodo de cálculo:** nodo donde se instalan instancias de VM, también conocidas como instancias de cálculo de Nova. El nodo ejecuta el daemon de cálculo que gestiona estas instancias de VM.
- **Nodo de almacenamiento:** nodo que aloja los datos.

La arquitectura que se describe en este capítulo se implementa en los siguientes tres sistemas:

Nota - En esta documentación, se describe una arquitectura que se implementa en un sistema físico separado. Para realizar la partición de un solo servidor de Oracle SPARC y configurar OpenStack de varios nodos en el servidor que ejecuta OVM Server for SPARC (LDoms), consulte [Multi-node Solaris 11.2 OpenStack on SPARC Servers](#). El artículo hace referencia específicamente a la versión Havana de OpenStack. Sin embargo, los pasos generales también se aplican a la versión actual.

La siguiente figura muestra una vista de alto nivel de la arquitectura que se describe en este capítulo.

FIGURA 4-1 Arquitectura de referencia de configuración de tres nodos



Esta arquitectura de ejemplo no muestra el servicio de almacenamiento del objeto de Swift. Para obtener información general acerca de cómo configurar Swift, consulte la información del sitio de la comunidad de OpenStack, como [OpenStack Configuration Reference](#). Para obtener información sobre cómo configurar los servicios Swift en sistemas Oracle Solaris y para obtener otra información sobre OpenStack en Oracle Solaris, consulte [OpenStack para Oracle Solaris 11](#).

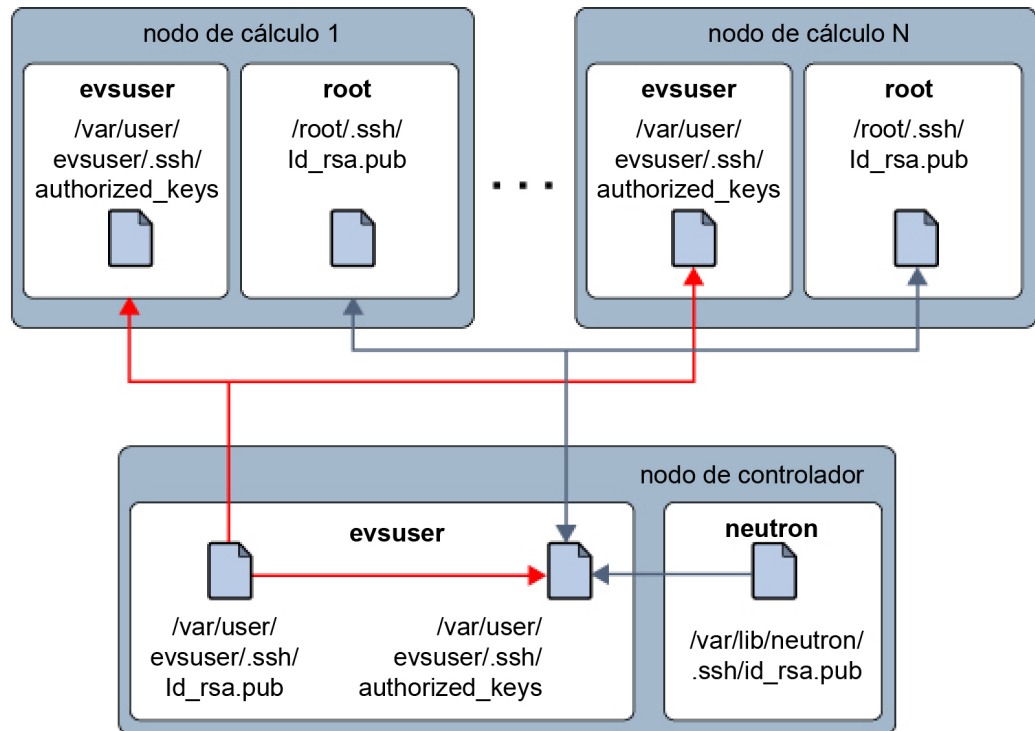
En Oracle Solaris, el conmutador virtual elástico (EVS) forma el backend de la red de OpenStack. EVS facilita la comunicación entre instancias de VM que están en redes VLAN o

VXLAN. Las instancias de VM pueden estar en el mismo nodo de cálculo o en varios nodos de cálculo. Para obtener más información acerca de EVS, consulte [Capítulo 5, “Acerca de los conmutadores virtuales elásticos”](#) de “Gestión de virtualización de red y recursos de red en Oracle Solaris 11.2”.

Para que los diferentes nodos puedan comunicarse entre sí, las claves públicas SSH de `evsuser`, `neutron` y `root` en el nodo de controlador deben estar en cada archivo `authorized_keys` de `evsuser` en todos los nodos de cálculo configurados. Consulte la imagen a continuación que muestra la distribución de las claves SSH públicas. En la imagen, se asume que se configuraron varios nodos de cálculo.

Para obtener una lista de los parámetros de configuración de OpenStack que son útiles para las implementaciones de OpenStack en sistemas Oracle Solaris, consulte "Parámetros de configuración comunes para OpenStack" en <http://www.oracle.com/technetwork/articles/servers-storage-admin/getting-started-openstack-os11-2-2195380.html>.

FIGURA 4-2 Distribución de claves SSH en controlador EVS



Configuración del nodo de controlador

El nodo de controlador tiene un servicio de panel de control, un almacén de imágenes y un servicio de identidad. Este nodo también incluye MySQL, RabbitMQ, almacenamiento de bloques y cálculos, y servicios de redes.

En la lista siguiente, se especifican las tareas para configurar el nodo de controlador:

- “Pasos preliminares” [81]
- “Instalación del Protocolo de hora de red” [82]
- “Instalación de MySQL” [83]
- “Instalación de Keystone” [85]
- “Instalación de Glance” [86]
- “Instalación de Nova” [88]
- “Instalación de Horizon” [89]
- “Instalación de Cinder” [91]
- “Instalación y configuración de Neutron” [94]
- “Instalación y configuración de Heat” [97]

Pasos preliminares

Para prepararse para la implementación de la configuración de OpenStack en varios nodos, tenga en cuenta lo siguiente:

- Si utiliza varias interfaces de red en un único nodo o sistema de OpenStack, cree nombres de host para esas interfaces de red.
 Por ejemplo, puede tener interfaces separadas para gestionar el tráfico administrativo y de API de OpenStack (red de OpenStack), el tráfico entre nodos de cálculo y el enrutador L3 (red de cliente) y el tráfico con la red más grande fuera de la configuración de la nube (red externa). Por lo tanto, se deben crear nombres de host correspondientes para cada host, por ejemplo, host-on, host-tn y host-en.
 Asegúrese de que todos estos nombres de host y direcciones IP se incluyan en el archivo /etc/hosts del nodo o en la configuración de DNS.
- Para facilitar la configuración de servicios de OpenStack, establezca variables, como las siguientes:
 - `$CONTROLLER_ADMIN_NODE`: el nombre de host de la interfaz o la dirección IP en el nodo de controlador al que están conectados los servicios administrativos de OpenStack.
 - `$CONTROLLER_ADMIN_NODE_IP`: la dirección IP del puerto de controlador que maneja los servicios administrativos y el tráfico de OpenStack.
 - `$COMPUTE_ADMIN_NODE_IP`: la dirección IP del puerto de cálculo que maneja los servicios administrativos y el tráfico de OpenStack.

- \$VOLUME_IP: el nombre de host del nodo de almacenamiento.
- Asegúrese, además, de que las contraseñas estén disponibles para asignarse según sea necesario.

Para gestionar mejor el uso de memoria entre ZFS y las aplicaciones de Oracle Solaris 11, establezca el parámetro `usr_reserve_hint_pct` en el nodo, como se muestra en el siguiente ejemplo:

```
# echo "set user_reserve_hint_pct=80" >>/etc/system.d/site:kernel-zones-reserve
# reboot
```

donde *site* puede hacer referencia a la compañía.

También debe establecer este parámetro en los diferentes nodos de OpenStack.

Para obtener más información sobre este parámetro, inicie sesión en su cuenta de MOS en <https://support.oracle.com> y consulte el documento 1663862.1, *Gestión de memoria entre ZFS y aplicaciones en Oracle Solaris 11.2*.

Instalación del Protocolo de hora de red

Se recomienda la instalación del Protocolo de hora de red (NTP), aunque es opcional. Si instala NTP, instálelo en cada nodo de servicio de la implementación en la nube.

NTP ayuda a garantizar una hora coherente del día en todos los nodos de servicio de la nube. Si activa NTP en una red, configure los nodos de servicio para obtener la hora por la red.

- Si la multidifusión IP está activada en las subredes IP donde residen los nodos de servicio puede aprovechar la multidifusión IP para configurar NTP.
- Si la multidifusión IP no está activada en las subredes IP donde residen los nodos de servicio, configure NTP manualmente.

▼ Cómo instalar y configurar el protocolo de hora de red

Antes de empezar Si no desea utilizar la multidifusión para NTP, debe configurar un servidor NTP.

1. Instale el paquete NTP.

```
controller# pkg install ntp
```

2. Instale el archivo de configuración.

```
controller# cp /etc/inet/ntp.client /etc/inet/ntp.conf
```

3. **(Opcional) Configure el archivo `/etc/inet/ntp.conf` si desea utilizar la multidifusión.**
 - a. **Elimine los comentarios de la opción `multicastclient`.**
 - b. **Elimine los comentarios de una o varias de las opciones del servidor y proporcione el nombre específico del servidor NTP o su dirección IP.**

El archivo de configuración podría ser similar al ejemplo siguiente:

```
# multicastclient 224.0.1.1
...
server system1.example.com iburst
server system2.example.com iburst
# server server_name3 iburst
```

4. **Active el servicio SMF del servidor NTP.**

```
controller# svcadm enable ntp
```

Instalación de MySQL

Muchos servicios OpenStack mantienen una base de datos para realizar un seguimiento de los recursos críticos, el uso y otra información. De manera predeterminada, se especifican bases de datos individuales de SQLite para este fin y son útiles para la configuración de un solo nodo. Para las configuraciones en varios nodos, en especial en un entorno de producción, se recomiendan otras bases de datos, como MySQL, para almacenar esta información.

La comunicación entre servicios de OpenStack se realiza a través del protocolo avanzado de cola de mensajes (AMQP). En Oracle Solaris, AMQP es implementado por RabbitMQ. RabbitMQ es un servicio necesario. Generalmente, se configura un solo nodo en la nube para ejecutar RabbitMQ. En esta arquitectura, RabbitMQ se configura para ejecutarse en el nodo de controlador.

▼ Cómo instalar una base de datos MySQL

1. **Instale el servidor MySQL y los paquetes.**

```
controller# pkg install mysql-55 mysql-55/client python-mysql \
rabbitmq markupsafe rad-evs-controller
```

2. **Active los servicios RabbitMQ.**

```
controller# svcadm enable rabbitmq
controller# svcadm restart rad:local
```

3. **(Opcional) Si utiliza una dirección IP dedicada para el tráfico de administración y de API, agregue esa dirección en `/etc/mysql/5.5/my.cnf`:**

```
bind-address=$CONTROLLER_ADMIN_NODE_IP
```

4. **Active el servicio MySQL.**

```
controller# svcadm enable mysql
```

5. **Defina la contraseña root del servidor MySQL.**

```
controller# mysqladmin -u root password MySQL-root-password
```

6. **Configure MySQL.**

Cree las tablas que usará OpenStack. Conceda privilegios a los servicios del nodo de controlador para proporcionar acceso exclusivo a estas bases de datos.

```
controller# mysql -u root -p
Enter password: MySQL-root-password
mysql> drop database if exists nova;
mysql> drop database if exists cinder;
mysql> drop database if exists glance;
mysql> drop database if exists keystone;
mysql> drop database if exists neutron;
mysql> drop database if exists heat;
mysql> create database cinder;
mysql> default character set utf8
mysql> default collate utf8_general_ci;
mysql> grant all privileges on cinder.* to 'cinder'@$CONTROLLER_ADMIN_NODE' \
identified by service-password';
mysql> grant all privileges on cinder.* to 'cinder'@$VOLUME_IP' \
identified by service-password';
mysql> create database glance;
mysql> default character set utf8
mysql> default collate utf8_general_ci;
mysql> grant all privileges on glance.* to 'glance'@$CONTROLLER_ADMIN_NODE' \
identified by service-password';
mysql> create database keystone;
mysql> default character set utf8
mysql> default collate utf8_general_ci;
mysql> grant all privileges on keystone.* to 'keystone'@$CONTROLLER_ADMIN_NODE' \
identified by service-password';
mysql> create database nova;
mysql> default character set utf8
mysql> default collate utf8_general_ci;
mysql> grant all privileges on nova.* to 'nova'@$CONTROLLER_ADMIN_NODE' \
identified by service-password';
mysql> create database neutron;
mysql> default character set utf8
mysql> default collate utf8_general_ci;
mysql> grant all privileges on neutron.* to 'neutron'@$CONTROLLER_ADMIN_NODE' \
```

```

identified by service-password';
mysql> create database heat
mysql> default character set utf8
mysql> default collate utf8_general_ci;
mysql> grant all privileges on heat.* to 'heat'@'$CONTROLLER_ADMIN_NODE' \
mysql> identified by service-password';
mysql> flush privileges;
mysql> quit

```

Instalación de Keystone

El servicio Keystone debe estar instalado y configurado en el nodo de controlador.

Ejemplo de secuencia de comandos de Keystone

Para rellenar rápidamente la base de datos de Keystone, puede utilizar una secuencia de comandos de ejemplo, `/usr/demo/openstack/keystone/sample_data.sh`. La secuencia de comandos crea los siguientes inquilinos iniciales:

- `service`, en el cual se crea un usuario de Keystone para cada servicio de OpenStack.
- `demo`, en el que se crea el usuario `admin` con una contraseña predeterminada.

Además, la secuencia de comandos también establece las variables de entorno que definen el nodo donde reside cada servicio API, así como la contraseña para cada servicio. De manera predeterminada, para cada uno de los servicios del inquilino `service`, el nombre del servicio, el nombre de usuario y la contraseña son idénticos. Por ejemplo, para el servicio Nova, se crea el usuario `nova` con la contraseña `nova`.

Puede cambiar las variables antes de ejecutar la secuencia de comandos para personalizar los nombres de usuario del servicio y sus contraseñas correspondientes, así como el nombre de los inquilinos iniciales. Asegúrese de revisar la secuencia de comandos para obtener más información acerca de los parámetros que se pueden establecer para el entorno.

▼ Cómo instalar y configurar Keystone

1. **Instale el paquete Keystone.**

```
controller# pkg install keystone
```

2. **Cree el token compartido para Keystone y otros servicios de OpenStack.**

El token se compone de una cadena aleatoria de caracteres.

```
controller# openssl rand -hex 10  
token-string
```

3. Establezca el token en una variable de entorno.

```
controller# export SERVICE_TOKEN=token-string
```

4. Modifique parámetros en el archivo `/etc/keystone/keystone.conf`.

La configuración debe ser similar al siguiente ejemplo:

```
[DEFAULT]  
admin_token = token-string  
qpid_hostname=$CONTROLLER_ADMIN_NODE  
rabbit_host=$CONTROLLER_ADMIN_NODE  
...  
[database]  
connection = mysql://keystone:service-password@$CONTROLLER_ADMIN_NODE/keystone
```

5. Active el servicio SMF de Keystone.

```
controller# svcadm enable keystone
```

6. Genere tokens de Infraestructura de clave pública (PKI).

```
controller# su - keystone -c "keystone-manage pki_setup"
```

7. Rellene la base de datos de Keystone.

Para utilizar la secuencia de comandos de ejemplo, emita el siguiente comando.

```
controller# CONTROLLER_PUBLIC_ADDRESS=$CONTROLLER_ADMIN_NODE \  
CONTROLLER_ADMIN_ADDRESS=$CONTROLLER_ADMIN_NODE \  
CONTROLLER_INTERNAL_ADDRESS=$CONTROLLER_ADMIN_NODE \  
SERVICE_TOKEN=token-string \  
/usr/demo/openstack/keystone/sample_data.sh
```

Instalación de Glance

La configuración de Glance requiere configurar cierta información para la autenticación, así como especificar la ubicación de los servicios MySQL y RabbitMQ.

▼ Cómo instalar y configurar Glance

1. Instale el paquete de Glance.

```
controller# pkg install glance
```

2. Configure Glance quitando los comentarios de los parámetros de estos archivos de configuración o estableciendo dichos parámetros:

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

```
[DEFAULT]
registry_host = $CONTROLLER_ADM_NODE
admin_user = glance
admin_password = service-password
admin_tenant_name = tenant
auth_url = http://$CONTROLLER_ADM_NODE:5000/v2.0
auth_strategy = keystone
default_publisher_id = image.$CONTROLLER_ADM_NODE
rabbit_host = $CONTROLLER_ADM_NODE
qpid_hostname = $CONTROLLER_ADM_NODE

[database]
connection = mysql://glance:service-password@$CONTROLLER_ADM_NODE/glance

[keystone_authtoken]
auth_uri= http://$CONTROLLER_ADM_NODE:5000/v2.0
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = glance
admin_password = service-password
```

- `/etc/glance/glance-cache.conf`

```
[DEFAULT]
auth_url = http://$CONTROLLER_ADM_NODE:5000/v2.0/
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = glance
admin_password = service-password
```

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

```
[DEFAULT]
default_publisher_id = image.$CONTROLLER_ADM_NODE
rabbit_host = $CONTROLLER_ADM_NODE
qpid_hostname = $CONTROLLER_ADM_NODE

[database]
connection = mysql://glance:glance@$CONTROLLER_ADM_NODE/glance

[keystone_authtoken]
```

```
auth_uri = http://$CONTROLLER_ADM_NODE:5000/v2.0
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = glance
admin_password = service-password
```

- /etc/glance/glance-api-paste.ini

```
[filter:authtoken]
auth_uri = http://$CONTROLLER_ADM_NODE:5000/v2.0/
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = glance
admin_password = service-password
```

- /etc/glance/glance-registry-paste.ini

```
[filter:authtoken]
auth_uri = http://$CONTROLLER_ADM_NODE:5000/v2.0/
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = glance
admin_password = service-password
```

- /etc/glance/glance-scrubber.conf

```
[DEFAULT]
auth_url = http://$CONTROLLER_ADM_NODE:5000/v2.0/
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = glance
admin_password = service-password
```

```
[database]
connection=mysql://glance:glance@$CONTROLLER_ADM_NODE/glance
```

3. Active los servicios SMF de Glance.

```
controller# svcadm enable -rs glance-api glance-db glance-registry glance-scrubber
```

Instalación de Nova

La configuración de Nova en el nodo de controlador también requiere la configuración de la autenticación usual y otra información de servicio. Tenga en cuenta que esta sección pertenece a la configuración de servicios de punto final de Nova, no al nodo de cálculo.

▼ Cómo instalar y configurar Nova

1. Instale el paquete de Nova.

```
controller# pkg install nova
```

2. Configure Nova quitando los comentarios de los parámetros o estableciendo dichos parámetros en el archivo `/etc/nova/nova.conf`:

```
[DEFAULT]
qpid_hostname=${CONTROLLER_ADM_NODE}
rabbit_host=${CONTROLLER_ADM_NODE}
my_ip=${CONTROLLER_ADMIN_NODE_IP}
host=${CONTROLLER_ADMIN_NODE}
firewall_driver=nova.virt.firewall.NoopFirewallDriver

[database]
connection = mysql://nova:nova@${CONTROLLER_ADM_NODE}/nova

[glance]
host=${CONTROLLER_ADM_NODE}

[keystone_authtoken]
auth_uri=http://${CONTROLLER_ADM_NODE}:5000/v2.0/
identity_uri=http://${CONTROLLER_ADM_NODE}:35357/
admin_user=nova
admin_password=service-password
admin_tenant_name=tenant

[neutron]
url=http://${CONTROLLER_ADM_NODE}:9696
admin_username=neutron
admin_password=service-password
admin_tenant_name=tenant
admin_auth_url=http://${CONTROLLER_ADM_NODE}:5000/v2.0
```

3. Active los servicios SMF de Nova.

```
controller# svcadm enable -rs nova-conductor
controller# svcadm enable -rs nova-api-osapi-compute
nova-cert nova-scheduler
```

Instalación de Horizon

Horizon sirve como el portal web para OpenStack. En el siguiente procedimiento, se supone que hay acceso de HTTP normal a Horizon, en lugar de SSL/TLS.

▼ Cómo configurar Horizon

1. Instale el paquete de Horizon.

```
controller# pkg install horizon
```

2. Realice uno de los siguientes grupos de pasos en función de lo que se aplica a su configuración.

■ Su configuración utiliza HTTP.

1. Proporcione valores para la secuencia de comandos `/etc/openstack_dashboard/local_settings.py`.

```
controller# gsed -i -e s@SECURE_PROXY_SSL_HEADER@#SECURE_PROXY_SSL_HEADER@ \
-e s@CSRF_COOKIE_SECURE@#CSRF_COOKIE_SECURE@ \
-e s@SESSION_COOKIE_SECURE@#SESSION_COOKIE_SECURE@ \
/etc/openstack_dashboard/local_settings.py
```

2. Copie la versión HTTP del archivo `http.conf` para OpenStack.

```
controller# cp /etc/apache2/2.2/samples-conf.d/openstack-dashboard-http.conf \
/etc/apache24/2.2/conf.d/
```

■ Su configuración utiliza SSL/TLS.

1. Genere certificados para uso de Horizon.

Los siguientes comandos generan certificados autofirmados para uso por Horizon y copian el archivo de configuración del panel de control de OpenStack al directorio de archivos de configuración de Apache. Para obtener más información acerca de la creación de certificados autofirmados, consulte [SSL/TLS Strong Encryption: FAQ](#) de Apache.

```
controller# export DASHBOARD=/etc/openstack_dashboard
controller# openssl req -new -x509 -nodes \
-out horizon.crt -keyout horizon.key
```

```
controller# mv horizon.crt horizon.key ${DASHBOARD}
controller# chmod 0600 ${DASHBOARD}/horizon.*
```

```
controller# sed \
-e "/SSLCertificateFile/s:/path.*:${DASHBOARD}/horizon.crt:" \
-e "/SSLCACertificateFile/d" \
-e "/SSLCertificateKeyFile/s:/path.*:${DASHBOARD}/horizon.key:" \
< /etc/apache2/2.2/samples-conf.d/openstack-dashboard-tls.conf \
> /etc/apache2/2.2/conf.d/openstack-dashboard-tls.conf
```

2. En el archivo `~/conf.d/openstack-dashboard-tls.conf`, especifique la dirección del sitio y el nombre del servidor del paquete de Horizon con los siguientes parámetros:

```
RedirectPermanent=controller-IP
ServerName=controller-name
```

3. Inicie el servicio Apache con uno de los siguientes comandos aplicables.

```
controller# svcadm enable apache22
```

Instalación de Cinder

La configuración de Cinder debe especificar al menos la siguiente información:

- Información de autorización para autenticación con Keystone.
- La clase de volúmenes que se crearán.

▼ Cómo instalar y configurar Cinder

Los pasos de este procedimiento hacen referencia a la configuración de servicios de punto final de Cinder, no al nodo de volumen o Cinder.

1. Instale el paquete de Cinder.

```
controller# pkg install cinder
```

2. Configure Cinder quitando los comentarios de los parámetros o estableciendo dichos parámetros en el archivo `/etc/cinder/cinder.conf`.

```
[DEFAULT]
qpid_hostname=${CONTROLLER_ADM_NODE}
rabbit_host=${CONTROLLER_ADM_NODE}
my_ip=${CONTROLLER_ADM_NODE}

[database]
connection = mysql://cinder:cinder@${CONTROLLER_ADM_NODE}/cinder

[keystone_authtoken]
auth_uri = http://${CONTROLLER_ADM_NODE}:5000/v2.0
identity_uri = http://${CONTROLLER_ADM_NODE}:35357
admin_tenant_name = tenant
admin_user = cinder
admin_password = service-password
```

3. Active los servicios SMF de Cinder.

```
controller# svcadm enable -rs cinder-db
```

```
controller# svcadm enable -rs cinder-api cinder-scheduler
```

Véase también Consulte también [How to Build OpenStack Block Storage on ZFS](#).

▼ Cómo configurar el controlador iSCSI de Cinder de ZFS Storage Appliance

El controlador iSCSI de Cinder Oracle ZFS Storage Appliance activa Oracle ZFS Storage Appliance (ZFSSA) para uso ininterrumpido como recurso de almacenamiento de bloques para Cinder. El controlador proporciona la capacidad de crear volúmenes iSCSI que pueden ser asignados por un servidor de Cinder a cualquier instancia de máquina virtual creada por el servicio de Nova. El paquete `cloud/openstack/cinder` entrega el controlador. El dispositivo debe ejecutar al menos la versión 2013.1.2.0 del software ZFSSA.

Antes de empezar Configure una agrupación en Oracle ZFS Storage Appliance. Puede elegir usar una agrupación preexistente.

1. Ejecute el flujo de trabajo `cinder.akwf`.

Puede usar un usuario existente o crear un nuevo usuario con autorizaciones de roles para realizar las operaciones del controlador de Cinder.

El flujo de trabajo de `cinder.akwf` realiza las siguientes tareas:

- Crea el usuario si el usuario no existe.
- Define las autorizaciones de roles para realizar operaciones del controlador de Cinder.
- Activa el servicio RESTful si el servicio está desactivado.

Puede ejecutar el flujo de trabajo desde la Interfaz de línea de comandos (CLI) o desde la Interface de usuario del explorador (BUI) del dispositivo.

■ Ejecute el flujo de trabajo desde la CLI.

```
zfssa:maintenance workflows> download
zfssa:maintenance workflows download (uncommitted)> show
Properties:
    url = (unset)
    user = (unset)
    password = (unset)

zfssa:maintenance workflows download (uncommitted)> set url="url to the cinder.akwf
file"
    url = "url to the cinder.akwf file"
zfssa:maintenance workflows download (uncommitted)> commit
Transferred 2.64K of 2.64K (100%) ... done

zfssa:maintenance workflows> ls
Properties:
```

```

showhidden = false

Workflows:

WORKFLOW      NAME                                     OWNER SETID ORIGIN
VERSION
workflow-000  Clear locks                                root  false Oracle Corporation
1.0.0
workflow-001  Configuration for OpenStack Cinder Driver  root  false Oracle Corporation
1.0.0

zfssa:maintenance workflows> select workflow-001

zfssa:maintenance workflow-001 execute (uncommitted)> set name=openstack
name = openstack
zfssa:maintenance workflow-001 execute (uncommitted)> set password=openstack-password
password = *****
zfssa:maintenance workflow-001 execute (uncommitted)> commit
User openstack created.

```

- **Ejecute el flujo de trabajo desde la BUI.**
 - a. **Seleccione Mantenimiento -> Flujos de trabajo y use el ícono más para cargar un nuevo flujo de trabajo.**
 - b. **Haga clic en el botón Buscar y seleccione el archivo `cinder.akwf`.**
 - c. **Haga clic en el botón CARGAR para terminar de cargar el flujo de trabajo.**
 - d. **Haga clic en la nueva fila que aparece en la página Flujos de trabajo de BUI para ejecutar el controlador de Cinder.**

El flujo de trabajo solicita un nombre de usuario y una contraseña. El nombre de usuario y la contraseña también se usarán en el archivo `cinder.conf` como `san_login` y `san_password`.

2. Establezca parámetros en el archivo `/etc/cinder/cinder.conf`.

Especifique las siguientes propiedades requeridas en el archivo `cinder.conf`:

- `volume_driver` – Asegúrese de que `cinder.volume.drivers.zfssa.zfssaiscsi.ZFSSAISC SIDriver` no tenga comentarios. Asegúrese de que las otras tres selecciones tengan comentarios.
- `san_ip`: el nombre o la dirección IP del host de gestión de ZFSSA.
- `san_login`: el nombre de usuario del usuario de Cinder en ZFSSA.
- `san_password`: la contraseña del usuario de Cinder en ZFSSA.
- `zfssa_pool`: la agrupación que se usará para asignar volúmenes.

- `zfssa_target_portal`: el portal de destino iSCSI de ZFSSA (`data-ip:port`). El puerto predeterminado es 3260.
- `zfssa_project`: el nombre del proyecto ZFSSA. Si el proyecto no existe en el dispositivo, el controlador creará un proyecto con ese nombre a la hora de inicio. Este proyecto contiene todos los volúmenes creados por el controlador. Se proporcionan propiedades adicionales de ZFSSA para configuración de características de volumen (por ejemplo, tamaño de bloques) y acceso (por ejemplo, iniciadores, destinos, seguridad).
- `zfssa_initiator_group`: el nombre del grupo de iniciadores. Si el grupo de iniciadores no existe en el dispositivo, el controlador creará un grupo de iniciadores con ese nombre a la hora de inicio. Si desea usar el grupo de iniciadores `default`, defina el valor de este parámetro en `default`. El grupo de iniciadores `default` puede ser útil para fines de evaluación. El grupo de iniciadores `default` no debe usarse normalmente, porque puede causar la exposición de los volúmenes a iniciadores no deseados o conflictivos.
- `zfssa_target_interfaces`: interfaces de red de destino iSCSI de ZFSSA. Use el siguiente comando para ver las interfaces:

```
zfssa:configuration net interfaces> show
```

```
Interfaces:
```

```
INTERFACE STATE CLASS LINKS   ADDR          LABEL
e1000g0  up   ip   e1000g0  1.10.20.30/24  Untitled Interface
```

- `connection`: establezca el parámetro de la siguiente manera.

```
connection=mysql://cinder:service-password@controller-fqn/cinder
```

3. Asegúrese de que el servicio iSCSI de ZFSSA esté en línea.

Si el servicio iSCSI de ZFSSA no está en línea, use la BUI o la CLI del dispositivo para activarlo. El siguiente ejemplo muestra cómo usar la CLI en el dispositivo:

```
zfssa:> configuration services iscsi
zfssa:configuration services iscsi> enable
zfssa:configuration services iscsi> show
Properties:
<status> = online
...
```

4. Active los servicios SMF del volumen de Cinder.

```
controller# svcadm enable cinder-volume:default cinder-volume:setup
```

Instalación y configuración de Neutron

En la arquitectura que se describe en este capítulo, el servicio API de Neutron se ejecuta en el nodo de controlador.

▼ Cómo instalar y configurar Neutron

1. Instale el paquete de Neutron.

```
controller# pkg install neutron
```

2. Configure Neutron quitando los comentarios de los parámetros de estos archivos de configuración o estableciendo dichos parámetros:

■ /etc/neutron/neutron.conf

```
qpid_hostname=${CONTROLLER_ADM_NODE}
rabbit_host=${CONTROLLER_ADM_NODE}

# Host to locate redis. (string value)
# host=127.0.0.1
host=${CONTROLLER_ADM_NODE}

[keystone_authtoken]
auth_uri = http://${CONTROLLER_ADM_NODE}:5000/v2.0
identity_uri = http://${CONTROLLER_ADM_NODE}:35357
admin_tenant_name = tenant
admin_user = neutron
admin_password = service-password

[database]
connection = mysql://neutron:neutron@${CONTROLLER_ADM_NODE}/neutron
```

■ /etc/neutron/plugins/evs/evs_plugin.ini

```
[EVS]
evs_controller = ssh://evsuser@${CONTROLLER_ADM_NODE}
```

■ /etc/neutron/dhcp_agent.ini

```
[DEFAULT]
evs_controller = ssh://evsuser@${CONTROLLER_ADM_NODE}
```

3. Configure el conmutador virtual elástico (EVS).

a. Establezca la propiedad de EVS para especificar la ubicación del controlador de EVS.

```
controller# evsadm set-prop -p controller=ssh://evsuser@${CONTROLLER_ADM_NODE}
```

b. Cree pares de claves SSH para usuarios evsuser, neutron y root.

```
controller# su - evsuser -c "ssh-keygen -N '' \
-f /var/user/evsuser/.ssh/id_rsa -t rsa"
```

```
controller# su - neutron -c "ssh-keygen -N '' -f /var/lib/neutron/.ssh/id_rsa -t rsa"
controller# ssh-keygen -N '' -f /root/.ssh/id_rsa -t rsa
```

- c. Combine las claves SSH de los usuarios evsuser, neutron y root en el archivo `authorized_keys` de evsuser.**

```
controller# cat /var/user/evsuser/.ssh/id_rsa.pub \
/var/lib/neutron/.ssh/id_rsa.pub /root/.ssh/id_rsa.pub >> \
/var/user/evsuser/.ssh/authorized_keys
```

- d. Pruebe las conexiones SSH para aceptar las huellas que se almacenarán en el archivo `known_host`.**

Especifique Yes en cada solicitud de confirmación.

```
controller# su - evsuser -c "ssh evsuser@$CONTROLLER_ADM_NODE true"
controller# su - neutron -c "ssh evsuser@$CONTROLLER_ADM_NODE true"
controller# ssh evsuser@$CONTROLLER_ADM_NODE true
```

- e. Establezca la propiedad de los directorios `.ssh`.**

```
controller# chown -R evsuser:evsgroup /var/user/evsuser/.ssh
controller# chown -R neutron:neutron /var/lib/neutron/.ssh
```

- f. Configure las propiedades `l2-type`, `uplink-port` y `vlan-range` del controlador de EVS.**

```
controller# evsadm set-controlprop -p property=value
```

En el siguiente ejemplo, se muestra cómo establecer estas propiedades. Opcionalmente, utilice el comando final para mostrar todas las propiedades de EVS:

```
controller# evsadm set-controlprop -p l2-type=vlan
controller# evsadm set-controlprop -p vlan-range=1,200-300
controller# evsadm set-controlprop -p uplink-port=net0

controller# evsadm show-controlprop -o all
```

- 4. Active el reenvío IP.**

```
controller# ipadm set-prop -p forwarding=on ipv4
```

- 5. Inicie el servicio de filtro IP.**

```
controller# svcadm enable -rs ipfilter
```

- 6. Active el servicio de servidor Neutron.**

```
controller# svcadm enable -rs neutron-server neutron-dhcp-agent
```


Instalación y configuración de Heat

Heat es el motor de orquestación de OpenStack que le permite desplegar las aplicaciones en la nube basadas en las plantillas de que cree. Heat se instala en el mismo nodo que Keystone.

▼ Cómo configurar Heat

Antes de empezar Primero debe configurar Keystone, como se describe en [Cómo instalar y configurar Keystone](#) antes de realizar esta tarea.

1. Instale el paquete de Heat.

```
controller# pkg install heat
```

2. Ejecute la secuencia de comandos de Heat.

```
controller# OS_SERVICE_ENDPOINT=http://$CONTROLLER_ADM_NODE \
SERVICE_HOST=$CONTROLLER_ADM_NODE \
OS_AUTH_URL=http://$CONTROLLER_ADM_NODE:5000/v2.0 \
OS_USERNAME=admin OS_PASSWORD=secrete OS_TENANT_NAME=demo \
/usr/demo/openstack/keystone/heat-keystone-setup
```

3. Configure Heat quitando los comentarios de los parámetros de estos archivos de configuración o estableciendo dichos parámetros:

- /etc/heat/heat.conf

```
[database]
connection = mysql://heat:heat@$CONTROLLER_ADM_NODE/heat
```

```
[keystone_authtoken]
auth_uri = http://$CONTROLLER_ADM_NODE:5000/v2.0
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = heat
admin_password = service-password
```

- /etc/heat/api-paste.ini

```
[filter:authtoken]
auth_uri = http://$CONTROLLER_ADM_NODE:5000/v2.0/
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_tenant_name = tenant
admin_user = heat
admin_password = service-password
```

4. Active el servicio de Heat.

```
controller# svcadm enable -rs heat-api heat-db heat-engine \  
heat-api-cfn heat-api-cloudwatch
```

Configuración del nodo de cálculo

Puede instalar instancias de VM en el nodo de cálculo, así como en el daemon nova-compute. Las instancias de VM proporcionan una amplia gama de servicios, como análisis y aplicaciones web. Puede configurar tantos nodos de cálculo como sea necesario para la nube.

Para gestionar mejor el uso de memoria entre ZFS y las aplicaciones de Oracle Solaris 11, establezca el parámetro `usr_reserve_hint_pct` en el nodo, como se muestra en el siguiente ejemplo:

```
# echo "set user_reserve_hint_pct=80" >>/etc/system.d/site:kernel-zones-reserve  
# reboot
```

donde *site* puede hacer referencia a la compañía.

También debe establecer este parámetro en los diferentes nodos de OpenStack.

Para obtener más información sobre este parámetro, inicie sesión en su cuenta de MOS en <https://support.oracle.com> y consulte el documento 1663862.1, *Gestión de memoria entre ZFS y aplicaciones en Oracle Solaris 11.2*.

▼ Cómo configurar el nodo de cálculo

1. **(Opcional) Instale y configure NTP.**

Consulte “[Instalación del Protocolo de hora de red](#)” [47].

2. **Instale el paquete de Nova.**

```
compute1# pkg install nova
```

3. **Reinicie el Daemon de acceso remoto (RAD).**

Nova usa el RAD para comunicación con la estructura de Oracle Solaris Zones.

```
compute1# svcadm restart rad:local
```

4. **Configure Nova quitando los comentarios de los siguientes parámetros o estableciendo dichos parámetros en el archivo `/etc/nova/nova.conf`:**

```
[DEFAULT]
rabbit_host=${CONTROLLER_ADM_NODE}
my_ip=${COMPUTE_ADMIN_NODE_IP}
host=${COMPUTE_ADMIN_NODE_X}
firewall_driver=nova.virt.firewall.NoopFirewallDriver
keystone_ec2_url=http://${CONTROLLER_ADM_NODE}:5000/v2.0/ec2tokens

[database]
connection = mysql://nova:nova@${CONTROLLER_ADM_NODE}/nova

[glance]
host=${CONTROLLER_ADM_NODE}

[keystone_authtoken]
auth_uri=http://${CONTROLLER_ADM_NODE}:5000/v2.0/
identity_uri=http://${CONTROLLER_ADM_NODE}:35357/
admin_usr=nova
admin_password=service-password
admin_tenant_name=tenant

[neutron]
url=http://${CONTROLLER_ADM_NODE}:9696
admin_username=neutron
admin_password=service-password
admin_tenant_name=tenant
admin_auth_url=http://${CONTROLLER_ADM_NODE}:5000/v2.0
```

5. Configure EVS en el nodo de cálculo.

a. Instale el paquete de EVS.

```
compute1# pkg install evs
```

b. Establezca la propiedad de EVS para especificar la ubicación del controlador de EVS.

```
compute1# evsadm set-prop -p controller=ssh://evsuser@${CONTROLLER_ADM_NODE}
```

6. Configure las comunicaciones entre los nodos de controlador y de cálculo.

a. Cree una clave pública SSH en el nodo de cálculo para el usuario root.

```
compute1# su - root -c "ssh-keygen -N '' -f /root/.ssh/id_rsa -t rsa"
```

b. (Opcional) Compruebe el contenido de la clave SSH.

```
compute1# cat /root/.ssh/id_rsa.pub
```

c. Copie la clave SSH /root/.ssh/id_rsa.pub en una ubicación del nodo de controlador.

- d. **En el nodo de controlador, agregue la clave SSH al archivo `authorized_keys` para `evsuser`.**

```
controller# cat location/id_rsa.pub >> /var/user/evsuser/.ssh/authorized_keys
```

- e. **(Opcional) Verifique que la clave SSH del nodo de cálculo se haya agregado al archivo `authorized_keys`.**

```
controller# cat /var/user/evsuser/.ssh/authorized_keys
```

La salida debe incluir el contenido de la clave SSH que usted generó en el nodo de cálculo y que se mostró en el [Paso 6.b](#).

- f. **En el nodo de cálculo, pruebe las conexiones SSH del nodo de cálculo al controlador y acepte las huellas que se almacenarán en el archivo `known_host`.**

Especifique Yes en la solicitud de confirmación.

```
compute1# ssh evsuser@$CONTROLLER_ADM_NODE true
```

7. Active el servicio de cálculo de Nova.

```
compute1# svcadm enable nova-compute
```

Configuración del nodo de almacenamiento

El nodo de almacenamiento es el repositorio de todos los datos dentro de la configuración de OpenStack.

Para gestionar mejor el uso de memoria entre ZFS y las aplicaciones de Oracle Solaris 11, establezca el parámetro `usr_reserve_hint_pct` en el nodo, como se muestra en el siguiente ejemplo:

```
# echo "set user_reserve_hint_pct=80" >>/etc/system.d/site:kernel-zones-reserve
# reboot
```

donde *site* puede hacer referencia a la compañía.

También debe establecer este parámetro en los diferentes nodos de OpenStack.

Para obtener más información sobre este parámetro, inicie sesión en su cuenta de MOS en <https://support.oracle.com> y consulte el documento 1663862.1, *Gestión de memoria entre ZFS y aplicaciones en Oracle Solaris 11.2*.

▼ Cómo configurar el nodo de almacenamiento de bloques

1. Instale los paquetes adecuados.

```
storage# pkg install cinder python-mysql mysql-55/client
```

2. Configure Cinder quitando los comentarios de los parámetros o estableciendo dichos parámetros en el archivo `/etc/cinder/cinder.conf`.

```
[DEFAULT]
san_is_local=true
my_ip=storage-IP
rabbit_host=controller-fqdn
glance_host=controller-IP
zfs_volume_base=cinder/cinder

[database]
connection = mysql://cinder:service-password@controller-fqdn/cinder

[DEFAULT]
san_is_local=true
my_ip=$VOLUME_IP
rabbit_host=$CONTROLLER_ADM_NODE
glance_host=$CONTROLLER_ADM_NODE
zfs_volume_base=cinder/cinder

[database]
connection = mysql://cinder:cinder@$CONTROLLER_ADM_NODE/cinder

[keystone_authtoken]
auth_uri = http://$CONTROLLER_ADM_NODE:5000/v2.0
identity_uri = http://$CONTROLLER_ADM_NODE:35357
admin_user = cinder
admin_password = service-password
admin_tenant_name = tenant
```

3. Inicie los servicios Cinder.

```
storage# svcadm enable -rs cinder-db cinder-volume:default cinder-volume:setup
storage# svcadm enable -rs iscsi/target
```

Configuración de una red interna en OpenStack

En una red interna, las instancias de VM que cree posteriormente sólo pueden comunicarse entre sí. Estas instancias no se pueden conectar a la red más amplia ni se puede acceder a ellas de forma externa.

Nota - Para configurar una red externa para un inquilino, consulte [“Configuración de OpenStack con una red externa” \[103\]](#).

▼ Cómo crear una red interna

Realice estos pasos en el nodo en el que se ejecutan servicios Neutron, que, en este caso, es el nodo de controlador.

1. Muestre los inquilinos existentes.

```
controller# keystone tenant-list
```

La lista de inquilinos incluye los inquilinos y sus ID correspondientes. Seleccione el ID del inquilino para el que desea crear la red interna.

2. Cree la red para el inquilino seleccionado.

```
controller# neutron net-create --tenant-id tenant-ID network-name
```

donde *tenant-ID* se obtiene de la lista de inquilinos en el paso anterior.

3. Cree la subred para el mismo inquilino.

```
controller# neutron subnet-create --name subnet-name \
--tenant-id tenant-ID network-name subnet-IP
```

ejemplo 4-1 Creación de la red interna

En este ejemplo, se muestra cómo crear la red interna para el inquilino demo, creado de manera predeterminada por la secuencia de comandos de Keystone de ejemplo en [“Ejemplo de secuencia de comandos de Keystone” \[85\]](#).

```
controller# keystone tenant-list
+-----+-----+-----+
|   id   |  name | enabled |
+-----+-----+-----+
| abcde12345 | demo | True |
| fghij67890 | service | True |
+-----+-----+-----+

controller# neutron net-create --tenant-id abcde12345 demo_internal_net
Created a new network:
+-----+-----+-----+
| Field                | Value                |
+-----+-----+-----+
| admin_state_up       | True                 |
| id                   | 9999                 |
| name                 | demo_internal_net   |
```

```

| provider:network_type | vlan          |
| provider:segmentation_id | 300          |
| router:external      | False        |
| shared               | False        |
| status               | ACTIVE       |
| subnets             |              |
| tenant_id            | abcde12345   |
+-----+-----+

```

```

controller# neutron subnet-create --name demo_int_subnet --tenant-id abcde12345 \
demo_internal_net 192.168.1.0/24
Created a new subnet:

```

```

+-----+-----+
|      Field      |              Value              |
+-----+-----+
| allocation_pools | {"start": "192.168.1.2", "end": "192.168.1.254"} | |
| cidr             | 192.168.1.0/24                 |
| dns_nameservers  |                                  |
| enable_dhcp      | True                            |
| gateway_ip       | 192.168.1.1                    |
| host_routes      |                                  |
| id               | 07f9b37c-ae4e-11e4-8000-db57d0041a2c |
| ip_version       | 4                                |
| name             | demo_int_subnet                 |
| network_id       | 99999|                          |
| tenant_id        | abcde12345                     |
+-----+-----+

```

Configuración de OpenStack con una red externa

Puede crear una red externa para permitir que las redes privadas en la nube se comuniquen con la red más amplia. En una nube, un inquilino puede tener una o más redes privadas. Al crear una red externa para la nube, se crea un enrutador de proveedor que es compartido por todas las redes del inquilino. Usted, el administrador, crea, posee y gestiona este enrutador. El enrutador no está visible en la vista de topología de red del cliente. Dado que hay solamente un único enrutador, las redes cliente no pueden usar direcciones IP superpuestas.

La creación de una red externa también implica la configuración del agente L3 de Neutron. El agente Neutron L3 crea automáticamente asignaciones NAT de uno a uno entre las direcciones asignadas a instancias de Nova y las direcciones IP flotantes. El agente L3 también permite la comunicación entre redes privadas. De manera predeterminada, el enrutamiento entre redes privadas que forman parte del mismo inquilino está desactivado. Para cambiar este comportamiento, defina `allow_forwarding_between_networks` en `True` en el archivo de configuración `/etc/neutron/l3_agent.ini` y reinicie el servicio SMF `neutron-l3-agent`.

El enrutador proporciona conectividad al exterior para las instancias de VM del inquilino. El enrutador realiza NAT bidireccional en la interface que conecta el enrutador a la red externa. Los clientes crean tantas IP flotantes (IP públicas) como necesitan o como permite la cuota de

IP flotante y, a continuación, asocian estas IP flotantes con las instancias de VM que requieren conectividad externa.

Para ver una ilustración de la relación entre redes internas y la red externa en OpenStack, consulte la [Figura 3-3, “Enrutador de proveedor con modelo de redes privadas”](#).

▼ Cómo configurar la red externa en OpenStack

En este procedimiento, se muestra cómo crear una red virtual que representa una red externa. En esta red virtual, DHCP no se usa. En su lugar, se crean direcciones IP flotantes. Estas direcciones IP flotantes se asignan a un cliente particular y se pueden asignar a instancias de VM de Nova para uso por usuarios con ese cliente.

Debido a que el controlador tiene el servicio Neutron en el ejemplo de arquitectura de varios nodos en este capítulo, puede realizar los siguientes pasos en el nodo de controlador.

Antes de empezar Para realizar este procedimiento, ya deben existir redes internas para el inquilino. Para conocer los procedimientos para crear una red privada, consulte [Cómo crear una red interna \[102\]](#).

Además, debe haber completado la configuración del conmutador virtual elástico, es decir, la configuración de sus propiedades `l2-type` y `vlan-range`. En el siguiente ejemplo, se muestra cómo visualizar estos valores de propiedades al configurarlos en el [Paso 9](#):

```
controller# evsadm show-controlprop -p l2-type -p vlan-range
PROPERTY  PERM  VALUE      DEFAULT  HOST
l2-type   rw    vlan       vlan     --
vlan-range rw    1,200-300  --      --
```

También debe tener la información sobre el inquilino cuyas redes privadas se conectarán a la red externa. Puede mostrar información del inquilino en cualquier momento mediante el siguiente comando:

```
keystone tenant-list
```

1. Active el filtro IP de Solaris.

```
controller# svcadm enable ipfilter
```

2. Active el reenvío de IP en todo el host.

```
controller# ipadm set-prop -p forwarding=on ipv4
```

3. Establezca las variables de entorno necesarias.

```
controller# export OS_USERNAME=neutron
controller# export OS_PASSWORD=service-password
controller# export OS_TENANT_NAME=service-name
controller# export OS_AUTH_URL=http://controller-name:5000/v2.0
```


4. Cree el enrutador de proveedor.

```
controller# neutron router-create router-name
```

El comando muestra el nombre de enrutador con un ID correspondiente. Utilícelo para actualizar el archivo de configuración en el siguiente paso.

5. Actualice el archivo de configuración del agente L3.

En el archivo `/etc/neutron/l3_agent.ini`, establezca el valor del parámetro `router_id` en el UUID de enrutador en el paso anterior.

```
router_id = router-ID
```

6. Active el servicio SMF de neutron-l3-agent.

```
controller# svcadm enable neutron-l3-agent
```

7. Cree una red externa.

```
controller# neutron net-create --provider:network_type=vlan \
--provider:segmentation_id=VLAN-nbr \
--router:external=true network-name
```

donde el valor de `segmentation_id` es el primer número del rango de VLAN.

8. Cree una subred y asíciela a la red externa.

Para la configuración de ejemplo de este documento, el DHCP está desactivado. La agrupación de asignaciones consta de un rango de direcciones IP flotantes que está asignado a la subred.

```
controller# neutron subnet-create --enable-dhcp=false --name subnet-name \
--allocation-pool start=start-IP, end=end-IP network-name subnet-IP
```

9. Agregue la red externa al enrutador.

```
controller# neutron router-gateway-set router-ID network-ID \
```

Nota - Puede obtener `router-ID` del archivo `/etc/neutron/l3_agent.ini`. Si es necesario, puede utilizar el comando `neutron net-list` para obtener `network-ID`.

10. Agregue la red privada del inquilino al enrutador.

Para este paso, necesita el ID de enrutador y el ID de subred del inquilino. Puede obtener la información de la siguiente manera:

a. Para obtener el ID de subred del inquilino, visualice primero los inquilinos y sus ID y, luego, las subredes del ID del inquilino específico.

```
# keystone tenant-list
# neutron net-list --tenant-id tenant-ID
```

b. Agregue la red privada al enrutador.

Puede repetir este paso en función de cuántas de las redes privadas del inquilino desea agregar a la red externa.

```
controller# neutron router-interface-add router-ID subnet-ID
```

ejemplo 4-2 Creación de la red externa para el inquilino service

El siguiente ejemplo utiliza parte de la [Figura 3-3, “Enrutador de proveedor con modelo de redes privadas”](#) como referencia. En la figura, el inquilino A tiene dos instancias de VM y cada una pertenece a una red privada. Las dos redes privadas son HR y ENG. Estas dos subredes se agregarán a la red externa para activar las dos instancias de VM con el fin de comunicarse con la red más amplia. El ejemplo asume que se utilizan el nombre de usuario y la contraseña predeterminados para el servicio Neutron se utiliza.

```
controller# svcadm enable ipfilter
controller# ipadm set-prop -p forwarding=on ipv4

controller# export OS_USERNAME=neutron
controller# export OS_PASSWORD=neutron
controller# export OS_TENANT_NAME=TenantA
controller# export OS_AUTH_URL=http://controller-name:5000/v2.0
```

```
controller# neutron router-create ext-router
Created a new router:
+-----+-----+
| Field          | Value          |
+-----+-----+
| admin_state_up | True           |
| external_gateway_info |              |
| id             | 97ro5-ut3er   |
| name           | ext-router    |
| status         | ACTIVE        |
| tenant_id      | abcde12345    |
+-----+-----+
```

En este punto, puede actualizar el archivo /etc/neutron/l3_agent.ini estableciendo el parámetro router_ID en 97ro5-ut3er.

Después de actualizar el archivo, continúe con los pasos restantes.

```
controller# svcadm enable neutron-l3-agent

controller# neutron net-create --provider:network_type=vlan \
--provider:segmentation_id=1 --router:external=true ext_network
Created a new network:
+-----+-----+
| Field          | Value          |
+-----+-----+
| admin_state_up | True           |
```

```

| id                | 555ext-net555    |
| name              | ext_network      |
| provider:network_type | vlan            |
| provider:segmentation_id | 1              |
| router:external   | True             |
| shared            | False           |
| status            | ACTIVE          |
| subnets          |                 |
| tenant_id         | abcde12345      |
+-----+-----+

```

```

controller# neutron subnet-create --enable-dhcp=False \
--name ext_subnet --allocation-pool start=10.134.13.8,end=10.134.13.254 \
ext_network 10.134.13.0/24
Created a new subnet:

```

```

+-----+-----+
| Field          | Value           |
+-----+-----+
| allocation_pools | {"start": "10.134.13.8", "end": "10.134.13.254"} |
| cidr            | 10.134.13.0/24 |
| dns_nameservers |                 |
| enable_dhcp     | False          |
| gateway_ip      | 10.134.13.1    |
| host_routes     |                 |
| id              | 444sub-net444  |
| ip_version      | 4              |
| name            | ext_subnet     |
| network_id      | 555ext-net555  |
| tenant_id       | abcde12345     |
+-----+-----+

```

```

controller# neutron router-gateway-set 97ro5-ut3er 555ext-net555
Set gateway for router 97ro5-ut3er

```

```

controller# keystone tenant-list

```

```

+-----+-----+-----+
| id    | name | enabled |
+-----+-----+-----+
| 12345abcde | TenantA | True |
| 67890fghij | TenantB | True |
+-----+-----+-----+

```

```

controller# neutron net-list --tenant-id 12345abcde

```

```

+-----+-----+-----+
| id    | name | subnets |
+-----+-----+-----+
| 1a3b5c7d9e | HR | xyz-123-uvw |
| 2f4g6h8i0j | ENG | 098-r2d2-56 |
+-----+-----+-----+

```

```

controller# neutron router-interface-add 97ro5-ut3er xyz-123-uvw HR added to the router.
Added interface xyz-123-uvw to router 97ro5-ut3er.

```

```
controller# neutron router-interface-add 97ro5-ut3er 098-r2d2-56    ENG added to the router.  
Added interface 098-r2d2-56 to router 97ro5-ut3er.
```

- Véase también
- [Cómo observar la configuración de agente L3 \[75\]](#)
 - [“Limitaciones conocidas” \[119\]](#)

▼ Cómo crear y asociar direcciones IP flotantes como usuario cliente

Un usuario cliente realiza esta procedimiento mediante el panel de control de OpenStack Horizon.

1. Inicie sesión en el panel de control de OpenStack.

Inicie sesión tal como se describe en [Cómo acceder al panel de control de OpenStack \[28\]](#), con las credenciales del usuario cliente.

2. Seleccione Proyecto -> Acceso y seguridad -> IP flotantes.

3. Seleccione el nombre de la red externa.

4. Haga clic en el botón Asignar IP.

El separador IP flotantes muestra una dirección IP flotante asignada.

5. Haga clic en el botón Asociar.

6. En el menú desplegable, seleccione el puerto de la instancia de VM.

La ventana Proyecto -> Instancias muestra que la dirección IP flotante está asociada con la instancia de VM.

Si seleccionó un par de claves (clave pública SSH) cuando inició una instancia de VM, entonces esa clave SSH se agregará al archivo `authorized_keys` para el usuario `root` en la instancia de VM.

7. Inicie sesión en la instancia de VM en ejecución.

```
# ssh root@IP-address
```

donde *IP-address* es la dirección IP flotante que está asociada a la instancia de VM.

Después de iniciar sesión, puede utilizar la instancia, como lo haría normalmente con otros sistemas, como instalar aplicaciones, etc.

```
global# ssh root@10.134.13.9  
Last login: Fri Jul 18 00:37:39 2014 from 10.132.146.13
```

```

Oracle Corporation      SunOS 5.11      11.2      June 2014
root@host-192-168-101-3:~# uname -a
SunOS host-192-168-101-3 5.11 11.2 i86pc i386 i86pc
root@host-192-168-101-3:~# zoneadm list -cv
ID NAME                STATUS      PATH                BRAND      IP
 2 instance-00000001 running    /                  solaris    excl
root@host-192-168-101-3:~# ipadm
NAME                CLASS/TYPE STATE    UNDER    ADDR
lo0                  loopback  ok      --        --
  lo0/v4             static   ok      --        127.0.0.1/8
  lo0/v6             static   ok      --        ::1/128
net0                 ip        ok      --        --
  net0/dhcp          inherited ok      --        192.168.101.3/24

```

▼ Cómo observar la configuración de agente L3

Puede usar comandos de filtro IP, como `ipf`, `ippool` y `ipnat`, y comandos de redes, como `dladm` y `ipadm`, para observar y resolver el problema de la configuración realizada por `neturon-l3-agent`.

1. Muestre las VNIC creadas por `neutron-l3-agent`.

```

network# dladm show-vnic
LINK                OVER      SPEED  MACADDRESS          MACADDRTYPE  VIDS
l3i7843841e_0_0     net1     1000   2:8:20:42:ed:22     fixed        200
l3i89289b8e_0_0     net1     1000   2:8:20:7d:87:12     fixed        201
l3ed527f842_0_0     net0     100    2:8:20:9:98:3e     fixed

```

2. Muestre las direcciones IP creadas por `neutron-l3-agent`.

```

network# ipadm
NAME                CLASS/TYPE STATE    UNDER    ADDR
l3ed527f842_0_0     ip        ok      --        --
  l3ed527f842_0_0/v4 static   ok      --        10.134.13.8/24
  l3ed527f842_0_0/v4a static   ok      --        10.134.13.9/32
l3i7843841e_0_0     ip        ok      --        --
  l3i7843841e_0_0/v4 static   ok      --        192.168.100.1/24
l3i89289b8e_0_0     ip        ok      --        --
  l3i89289b8e_0_0/v4 static   ok      --        192.168.101.1/24

```

3. Muestre las reglas de filtro IP.

```

network# ipfstat -io
empty list for ipfilter(out)
block in quick on l3i7843841e_0_0 from 192.168.100.0/24 to pool/4386082
block in quick on l3i89289b8e_0_0 from 192.168.101.0/24 to pool/8226578
network# ippool -l
table role = ipf type = tree number = 8226578
{ 192.168.100.0/24; };
table role = ipf type = tree number = 4386082

```

```
{ 192.168.101.0/24; };
```

4. Muestre las reglas de IP NAT.

```
network# ipnat -l
List of active MAP/Redirect filters:
bimap l3ed527f842_0_0 192.168.101.3/32 -> 10.134.13.9/32
List of active sessions:
BIMAP 192.168.101.3 22 <- -> 10.134.13.9 22 [10.132.146.13 36405]
```

◆◆◆ 5 C A P Í T U L O 5

Creación de instancias de máquinas virtuales

En este capítulo, se describe cómo aprovisionar instancias de máquinas virtuales (instancias de VM) en la nube de OpenStack. Cada instancia de VM pertenece a un cliente. Los usuarios pueden crear una o más instancias de VM, en uno o más clientes, y trabajar en ellas. Para obtener información general acerca de cómo crear y gestionar instancias de VM de OpenStack, consulte [OpenStack End User Guide](#).

Para crear una instancia de VM, necesita un tipo y una imagen. En este capítulo, se tratan los temas siguientes:

- “Gestión de tipos” [111]
- “Gestionar imágenes” [114]
- “Creación de una instancia de VM” [117]

Es posible que desee crear un archivo donde definirá las variables de entorno para los comandos de OpenStack que se usarán. Si no define las variables del entorno, deberá especificar las opciones en cada comando. Consulte la [OpenStack Command-Line Interface Reference](#) para obtener más información.

Gestión de tipos

Un *tipo* es un tipo de instancia de VM o una plantilla de hardware virtual. Un tipo especifica un conjunto de recursos de máquinas virtuales, como la cantidad de CPU virtuales, la cantidad de memoria y el espacio en disco asignado a una instancia de VM. En Solaris, el tipo también incluye la marca de la zona subyacente: `solaris` para zonas no globales y `solaris-kz` para zonas de núcleo. Un ejemplo de un tipo de instancia es una zona de kernel con 16 CPU virtuales y 16384 MB de RAM.

Para obtener información general acerca de los tipos, consulte la sección [Flavors](#)” de la *Guía del administrador de nubes de OpenStack*.

Visualización de la información sobre tipos

el comando `nova flavor-list` muestra una lista de los tipos disponibles. Cuando cree una instancia de VM, usará el nombre o el ID del tipo. Para incluir `extra-specs` en la lista de tipos, especifique la opción `--extra-specs` con el comando `nova flavor-list`. Para obtener más información acerca del valor `extra_specs`, consulte [“Modificación de las especificaciones de tipos” \[113\]](#).

En el siguiente ejemplo, las columnas `RXTX_Factor` y `Is_Public` se quitan del resultado para ahorrar espacio. Consulte la [OpenStack Command-Line Interface Reference](#) para obtener más información acerca de estas columnas.

```
$ nova flavor-list
+-----+-----+-----+-----+-----+-----+
| ID | Name                               | Memory_MB | Disk | Ephemeral | Swap | VCPUs |
+-----+-----+-----+-----+-----+-----+
| 1  | Oracle Solaris kernel zone - tiny | 2048      | 10  | 0         |      | 1     |
| 10 | Oracle Solaris non-global zone - xlarge | 16384    | 80  | 0         |      | 32    |
| 2  | Oracle Solaris kernel zone - small  | 4096     | 20  | 0         |      | 4     |
| 3  | Oracle Solaris kernel zone - medium | 8192     | 40  | 0         |      | 8     |
| 4  | Oracle Solaris kernel zone - large  | 16384    | 40  | 0         |      | 16    |
| 5  | Oracle Solaris kernel zone - xlarge | 32768    | 80  | 0         |      | 32    |
| 6  | Oracle Solaris non-global zone - tiny | 2048     | 10  | 0         |      | 1     |
| 7  | Oracle Solaris non-global zone - small | 3072    | 20  | 0         |      | 4     |
| 8  | Oracle Solaris non-global zone - medium | 4096    | 40  | 0         |      | 8     |
| 9  | Oracle Solaris non-global zone - large | 8192     | 40  | 0         |      | 16    |
+-----+-----+-----+-----+-----+-----+
```

Los siguientes comandos muestran información detallada acerca del tipo especificado. La primera versión del comando especifica el nombre de tipo y la segunda versión especifica el ID de tipo. Esta salida es idéntica para ambos comandos.

```
$ nova flavor-show 'Oracle Solaris kernel zone - large'
$ nova flavor-show
+-----+-----+
| Property          | Value                               |
+-----+-----+
| name              | Oracle Solaris kernel zone - large |
| ram               | 16384                               |
| OS-FLV-DISABLED:disabled | False                               |
| vcpus             | 16                                   |
+-----+-----+
```



```

| extra_specs          | {'zonecfg:brand': 'solaris-kz'} |
| swap                |                                  |
| os-flavor-access:is_public | True                             |
| rxtx_factor         | 1.0                              |
| OS-FLV-EXT-DATA:ephemeral | 0                                |
| disk                | 40                               |
| id                  | 4                                |
+-----+-----+

```

Modificación de las especificaciones de tipos

Para cambiar las especificaciones de tipos, use el subcomando `flavor-key` para modificar el valor `extra_specs`.

```
nova flavor-key flavor action key=value key=value ...]
```

flavor El nombre o el ID del tipo.

action set o unset

key=value *key* es el nombre de la especificación. *value* es el nuevo valor de esa especificación. unset

```
$ nova flavor-key 4 set zonecfg:bootargs=-v
```

```
$ nova flavor-show
```

```

+-----+-----+
| Property          | Value                               |
+-----+-----+
| name              | Oracle Solaris                     |
|                  | kernel zone - large               |
| ram               | 16384                              |
| OS-FLV-DISABLED:disabled | False                             |
| vcpus             | 16                                  |
| extra_specs      | {'zonecfg:brand':                  |
|                  |   u'solaris-kz', u'zonecfg:bootargs': u'-v'} |
| swap             |                                     |
| os-flavor-access:is_public | True                             |
| rxtx_factor       | 1.0                              |
| OS-FLV-EXT-DATA:ephemeral | 0                                |
| disk              | 40                               |
| id                | 4                                |
+-----+-----+

```

Se admiten las siguientes propiedades en zonas de núcleo y en zonas no globales:

- `zonecfg:bootargs`
- `zonecfg:brand`
- `zonecfg:hostid`

Las siguientes propiedades se admiten únicamente en zonas no globales:

- `zonecfg:file-mac-profile`
- `zonecfg:fs-allowed`
- `zonecfg:limitpriv`

No se admiten otras propiedades de configuración en OpenStack. Consulte la página del comando `man zonecfg(1M)` para obtener información acerca de estas propiedades de configuración de zona.

Use la clave `sc_profile` para especificar un perfil de configuración del sistema:

```
$ nova flavor-key 4 set sc_profile=/system/volatile/profile/sc_profile.xml
```

El comando `nova flavor-key` le permite modificar o agregar únicamente los valores `extra_specs`. Para modificar otras especificaciones de un tipo existente, como la cantidad de RAM, debe suprimir el tipo y crear un tipo modificado con el mismo nombre. Para obtener información general acerca de cómo suprimir y crear tipos, consulte [OpenStack Admin User Guide](#).

Gestionar imágenes

Una imagen de máquina virtual, o *imagen*, es un único archivo que contiene un disco virtual que tiene instalado un sistema operativo de inicio. Una imagen proporciona una plantilla para sistemas de archivos de máquinas virtuales.

Además del panel de control, puede usar los clientes de línea de comandos `glance` y `nova` o las API de cálculo y servicio de imágenes para gestionar las imágenes.

Visualización de información sobre imágenes

El comando `nova image-list` muestra una lista de la imágenes disponibles. Cuando cree una instancia de VM, usará el nombre o el ID de la imagen.

```
$ nova image-list
+-----+-----+-----+-----+
| ID                | Name                | Status | Server |
+-----+-----+-----+-----+
| e422aae1-b0ba-618c-85d3-a214059800e2 | Solaris Kernel Zone | ACTIVE |        |
| e82aa857-ec92-4859-f530-deb89274863e | Solaris Non-global Zone | ACTIVE |        |
+-----+-----+-----+-----+
```

El comando `glance image-list` muestra información adicional, incluido el formato del disco, el formato del contenedor y el tamaño de imagen:

```
$ glance image-list --human-readable
```

Los comandos `nova image-show` y `glance image-show` muestran más información acerca de una imagen especificada:

```
$ nova image-show 'Solaris Kernel Zone'
+-----+
| Property          | Value                               |
+-----+
| OS-EXT-IMG-SIZE:size | 1547458560                          |
| created           | 2014-06-29T15:40:49Z                |
| id                | e422aae1-b0ba-618c-85d3-a214059800e2 |
| metadata architecture | x86_64                              |
| metadata hypervisor_type | solariszones                        |
| metadata vm_mode   | solariszones                        |
| minDisk           | 0                                    |
| minRam            | 0                                    |
| name              | Solaris Kernel Zone                 |
| progress          | 100                                  |
| status            | ACTIVE                              |
| updated           | 2014-06-29T15:40:55Z                |
+-----+

$ glance image-show 'Solaris Kernel Zone'
+-----+
| Property          | Value                               |
+-----+
| Property 'architecture' | x86_64                              |
| Property 'hypervisor_type' | solariszones                        |
| Property 'vm_mode'      | solariszones                        |
| checksum               | b2fc9560c15603c7663326db82d5ddaa  |
| container_format      | bare                                 |
| created_at             | 2014-06-29T15:40:49.108578         |
| deleted                 | False                               |
| disk_format            | raw                                  |
| id                     | e422aae1-b0ba-618c-85d3-a214059800e2 |
| is_public               | True                                 |
| min_disk                | 0                                    |
| min_ram                 | 0                                    |
| name                    | Solaris Kernel Zone                 |
| owner                   | 7461d4a9f5a64af9a01ae4e84e08c182  |
| protected               | False                               |
| size                    | 1547458560                          |
| status                  | active                              |
| updated_at              | 2014-06-29T15:40:55.769756         |
+-----+
```

Creación de una imagen

En Solaris, las imágenes de OpenStack son Unified Archive y se deben ejecutar en Oracle Solaris 11.2. Mediante el uso del comando `archiveadm`, puede crear nuevos Unified Archive desde zonas globales, no globales y de núcleo que ejecutan Oracle Solaris 11.2. Cargue la imagen en el repositorio de Glance para uso con OpenStack.

Unified Archive puede ser un archivo de clonación o un archivo de recuperación. Cree un archivo de clonación sobre la base del entorno de inicio activo actual o cree un archivo de recuperación que incluya todos los entornos de inicio y la información de configuración del sistema. Un archivo de clonación no incluye ninguna información de configuración del sistema de la instancia del SO. Para un archivo de clonación, los instaladores fuerzan la reconfiguración o puede proporcionar la configuración en un perfil de configuración del sistema (SC). Un archivo de clonación tampoco incluye, por ejemplo, entornos de inicio. Use Unified Archive de recuperación si desea el sistema completo. Para obtener más información acerca de Unified Archive, consulte [“Uso de Unified Archives para la clonación y la recuperación del sistema en Oracle Solaris 11.2”](#).

Los siguientes comandos capturan un Unified Archive de una zona no global en ejecución que se denomina myzone:

```
global# zonecfg -z myzone create
global# zoneadm -z myzone install
global# zlogin myzone
'sed /^PermitRootLogin/s/no$/without-password/
< /etc/ssh/sshd_config > /system/volatile/sed.$$ ;
cp /system/volatile/sed.$$ /etc/ssh/sshd_config'
global# archiveadm create -z myzone /var/tmp/myzone.uar
```

También puede crear una imagen de OpenStack mediante la creación de una *instantánea* de una instancia de VM existente. Use el comando `nova image-create` para crear una imagen mediante la toma de una instantánea desde una instancia de VM en ejecución.

Además de crear una imagen para usar en la creación de instancias de VM, es posible que desee usar imágenes personalizadas para copias de seguridad de datos o para rescatar una instancia de VM. Una *zona de rescate* es un tipo especial de imagen que se inicia cuando se coloca una imagen de VM en el modo `rescue`. Una imagen de rescate permite al administrador montar el sistema de archivos para que la instancia de VM corrija el problema.

Agregación de una imagen al almacén de imágenes

El servicio de imágenes de OpenStack, Glance, proporciona servicios de almacenamiento, detección, registro y entrega para imágenes de disco y servidor. Un *servidor de registros* es un servicio de imágenes que proporciona información de metadatos de imágenes a los clientes. El servicio de imágenes usa la *caché de imágenes* para obtener imágenes en el host local, en lugar de volver a descargarlas desde el servidor de imágenes cada vez que se solicita una imagen.

El siguiente comando carga el Unified Archive creado en la sección anterior en el repositorio de Glance. Use `raw` como tipo de formato. Asegúrese de especificar la propiedad `architecture`.

```
global# glance image-create --container-format bare --disk-format raw
--is-public true --name "Oracle Solaris 11.2 x86 NGZ"
```

```
--property architecture=x86_64
--property hypervisor_type=solariszones
--property vm_mode=solariszones < /var/tmp/myzone.uar
```

El comando `glance image-create` puede cargar la imagen y definir todos los valores de propiedad a la vez. La siguiente secuencia de comandos muestra cómo asegurarse de que cargará la imagen con la propiedad `architecture` definido en la arquitectura del host actual:

```
#!/bin/ksh

# Upload Unified Archive image to glance with proper Solaris decorations

arch=$(archiveadm info -p $1|grep ^archive|cut -d '|' -f 4)

if [[ "$arch" == "i386" ]]; then
    imgarch=x86_64
else
    imgarch=sparc64
fi

name=$(basename $1 .uar)

export OS_USERNAME=glance
export OS_PASSWORD=glance
export OS_TENANT_NAME=service
export OS_AUTH_URL=http://controller-name:5000/v2.0

glance image-create --name $name --container-format bare --disk-format raw --owner service
--file $1 --is-public True --property architecture=$imgarch --property
hypervisor_type=solariszones
--property vm_mode=solariszones --progress
```

Creación de una instancia de VM

Se requiere un tipo, una imagen y una red para crear una instancia de VM.

▼ Cómo crear una instancia de VM mediante el uso de una interfaz de línea de comandos

1. Seleccione el cliente para el que desea crear la nueva instancia de VM.

Deberá especificar el nombre del cliente o el ID en el comando para crear la instancia de VM.

```
$ keystone tenant-list
+-----+-----+-----+
|          id          | name | enabled |
```

```
+-----+-----+-----+
| 6ea34f7dafa5ce3c9a1b9de659e59d77 | demo | True |
| 0bda9b63b800ca808031a38637d50f3e | service | True |
+-----+-----+-----+
```

2. Seleccione la imagen para la que desea crear la nueva instancia de VM.

Deberá especificar el nombre de la imagen o el ID en el comando para crear la instancia de VM. Consulte [“Visualización de información sobre imágenes” \[114\]](#) para obtener instrucciones acerca de cómo mostrar especificaciones de cada imagen, incluido el ID de imagen.

Si implementa una instancia de VM en una zona no global con un nivel de versión anterior que la zona global, la instancia de VM se actualizará automáticamente al nivel de versión de la zona global durante la instalación. Si se intenta implementar una instancia de VM en una zona no global con un nivel de versión posterior que la zona global, fracasará.

3. Seleccione un tipo desde el que desea crear la nueva instancia de VM.

Asegúrese de tener un tipo con las especificaciones que necesita. Agregue las especificaciones requeridas a un tipo o cree un tipo nuevo con las especificaciones que necesita. Consulte [“Visualización de la información sobre tipos” \[112\]](#) para obtener instrucciones acerca de cómo mostrar especificaciones de cada tipo, incluido el ID de tipo y extra-specs.

4. Seleccione una red para que use la nueva instancia de VM.

Deberá especificar el nombre de la red o el ID en el comando para crear la instancia de VM. Si el comando `neutron net-list` no muestra una red para el cliente que eligió en el Paso 1, use el comando `neutron net-create` para crear una red para este cliente. Consulte la [OpenStack Command-Line Interface Reference](#) para obtener detalles. Observe el ID de la red.

5. Cree la instancia.

Use el comando `nova boot` para crear una instancia de cálculo e iniciarla. *imageID* corresponde al Paso 2, *flavorID* al Paso 3 y *nicID* al Paso 4. Consulte [OpenStack Command-Line Interface Reference](#) para obtener detalles.

```
# nova boot --image imageID --flavor flavorID --nic net-id=nicID
```

6. Seleccione una dirección IP flotante para que use la nueva instancia de VM.

Use el comando `neutron floatingip-list` para mostrar las direcciones IP flotantes para el cliente que eligió en el Paso 1. Si es necesario, use el comando `neutron floatingip-create` para crear una dirección IP flotante para este cliente. Observe el ID de la dirección IP flotante.

7. Asocie la dirección IP flotante con la nueva instancia de VM.

Use el comando `neutron floatingip-associate` para asociar la dirección IP flotante del Paso 6 con la nueva instancia de VM.

Solución de problemas de OpenStack

Este capítulo proporciona la información siguiente:

- Problemas conocidos de esta versión
- Uso de archivos log asociados con OpenStack
- Investigación y resolución de problemas

Limitaciones conocidas

Los siguientes son problemas conocidos con OpenStack (Havana 2013.2.3) en Oracle Solaris 11.2:

- No se admite el acceso de la consola remota a instancias de VM mediante el uso del panel de control de OpenStack. En su lugar use, el panel de control para cargar un par de claves SSH. Este par de claves se inserta en el archivo `authorized_keys` de `root` para esa instancia de VM.
- Solamente se admiten por completo los nodos de Nova que ejecutan Solaris porque Neutron admite únicamente un complemento único para virtualización de red.
- Actualmente, el anexo del volumen Cinder no se admite en las zonas no globales.
- Las instancias de VM deben ejecutar Oracle Solaris 11.2.
- No se admite el cambio de tamaño de las instancias de VM.

No se admite el comando `nova resize`. Es posible que la salida del comando `nova resize` indique que se ha completado el comando, pero el comando `nova resize-confirm` informa que el cambio de tamaño de la instancia no se puede confirmar y el comando `nova show` muestra que no se ha cambiado el tamaño de la instancia.

- No se admite la migración de VM en vivo.

No se admite el comando `nova live-migration`.

- No se admite la copia de seguridad de Cinder.

El servicio `cinder-backup` se instala cuando instala el paquete `cinder`, pero el servicio está desactivado y deje dejarlo desactivado.

- En el cuadro de diálogo Iniciar instancia del panel de control, solamente sea admite Iniciar desde imagen para Origen de inicio de instancia. En el menú Proyecto ->

Imágenes e instantáneas -> Acciones, no se admite `CreateVolumeFromImage`. Consulte [Solaris OpenStack Horizon customizations](#) para obtener información acerca de otras personalizaciones del panel de control de Solaris OpenStack.

- No se admiten enlaces de datos VXLAN como valor para la opción `external_network_dataLink` del archivo `/etc/neutron/l3_agent.ini`. Si define un enlace de datos VXLAN como valor para la opción `external_network_dataLink`, el agente Neutron L3 no podrá crear ni asociar una VNIC en la red externa.

- Deberá usar la línea de comandos para modificar la cuota del recurso de red de un proyecto.

La cuota de un recurso de red no se puede modificar desde Horizon. Puede usar el panel de control de Horizon para crear un proyecto o modificar los recursos que no pertenecen a la red de un proyecto existente. Para modificar la cuota para redes, subredes, puertos, enrutadores o direcciones IP flotantes de un proyecto, deberá usar el comando `neutron quota-update`.

Incluso cuando modifique un recurso que no pertenece a la red, se mostrará el siguiente mensaje de error. Puede ignorar este mensaje. Contrariamente a lo que indica este mensaje, la cuota para el recurso que no pertenece a la red se ha aplicado.

Error: Modified project information and members, but unable to modify project quotas.

- SMF y OpenStack podrán informar un estado de servicio diferente.

El siguiente ejemplo muestra que el servicio `nova-cert` está desactivado en OpenStack, aunque SMF muestra el servicio como `online`:

```
root@c190-133:~# nova service-disable c190-133 nova-cert
+-----+-----+-----+
| Host      | Binary   | Status   |
+-----+-----+-----+
| c190-133 | nova-cert | disabled |
+-----+-----+-----+
root@c190-133:~# svcs nova-cert
STATE          STIME      FMRI
online         21:14:11  svc:/application/openstack/nova/nova-cert:default
```

- El servicio SMF `neutron-l3-agent` entra en mantenimiento cuando se inicia.

Solución: reinicie el servicio `ipfilter` y borre `neutron-l3-agent`.

```
network# svcadm restart ipfilter:default
network# svcadm clear neutron-l3-agent:default
```

- En determinadas configuraciones, se elimina la puerta de enlace predeterminada para el nodo de red.

Si la dirección IP del nodo de red se deriva del espacio de direcciones `external_network` y si usa el comando `neutron router-gateway-clear` para eliminar `external_network` de `provider_router`, se suprime la puerta de enlace predeterminada para el nodo de red y no es posible acceder al nodo de red.


```
network# neutron router-gateway-clear router_UUID
```

Solución: conecte el nodo de red mediante la consola y agregue la puerta de enlace predeterminada nuevamente.

- La base de datos sqlite de Nova se bloquea cuando se crean varias instancias simultáneamente.

Cuando se crea una gran cantidad de instancias (por ejemplo, 10 o más) simultáneamente, el comando `nova list` deja de funcionar durante un período y muestra el siguiente mensaje de error:

```
$ nova list
```

```
ERROR: The server has either erred or is incapable of performing the
requested operation. (HTTP 500) (Request-ID:
req-0ad63452-6753-c9fc-8275-e80604d42569)
```

Horizon tampoco puede encontrar las instancias.

La causa de este problema es que se bloquea la base de datos sqlite de Nova. Después de un tiempo, la base de datos se restablece y `nova list` y Horizon funcionan correctamente.

Solución: use la base de datos de MySQL en lugar de sqlite. Consulte [Cómo configurar el nodo de cálculo \[61\]](#).

Examen de los archivos log

Los servicios SMF y varios procesos de Solaris producen archivos log donde pueden buscar mensajes de error o recopilar más información acerca de los mensajes que se muestran en la pantalla. Los archivos log del servicio SMF contienen información de depuración valiosa.

Según el problema, los archivos log del servicio SMF `nova-compute`, `nova-scheduler` y `neutron-server` generalmente son útiles. Use el comando `svcs -L` para encontrar el nombre de un archivo log del servicio SMF.

```
$ svcs -L neutron-server
```

```
/var/svc/log/application-openstack-neutron-neutron-server:default.log
```

Como usuario con privilegios, puede usar el comando `svcs -Lv` para ver el archivo log del servicio.

```
# svcs -Lv neutron-server
```

El comando `svcs -xv` muestra el estado del servicio, así como el nombre del archivo log.

```
$ svcs -xv neutron-server
```

```
svc:/application/openstack/neutron/neutron-server:default (OpenStack Neutron Network Service)
State: online since Fri Jul 25 12:11:16 2014
See: /var/svc/log/application-openstack-neutron-neutron-server:default.log
Impact: None.
```

Además de los archivos log del servicio SMF, los servicios de OpenStack producen archivos log y varios procesos de Solaris tienen sus propios archivos log. Algunos servicios de OpenStack registran información con el nombre de servicio de OpenStack en el directorio `/var/log`. Por ejemplo, los archivos log del almacén de imágenes de OpenStack están en `/var/log/glance`. Si tiene problemas para crear e iniciar una instancia de VM, busque en el directorio `/var/log/zones`.

La mayoría de los archivos de configuración de OpenStack tienen el nombre de servicio OpenStack en el directorio `/etc`. Por ejemplo, los archivos de configuración de red de OpenStack están en `/etc/neutron`. Los archivos de configuración de Horizon están en `/etc/openstack_dashboard`.

Para recibir más información en el archivo log para un servicio de OpenStack, configure la opción `verbose` en el archivo de configuración para ese servicio. Es posible que el archivo de configuración ya tenga la opción `verbose` configurada en `false` o que tenga comentarios. Elimine los comentarios o agregue la opción `verbose` y configure `verbose=true`. De manera similar, puede configurar `debug=true` en el archivo de configuración para ver más acerca de la salida de las operaciones afectadas por ese archivo de configuración. Consulte las tablas de las opciones de configuración en “Parámetros de configuración comunes para OpenStack” en [Getting Started with OpenStack on Oracle Solaris 11.2](#) y en la *Referencia de configuración de OpenStack* del [sitio de documentación de OpenStack](#).

De forma similar a la configuración de `debug=true` en un archivo de configuración, comandos de servicio individuales de OpenStack pueden usar una opción `--debug`.

Asuma el perfil de RBAC adecuado para ver los archivos log del servicio OpenStack o use el comando `pfedit` para modificar los archivos de configuración del servicio OpenStack. Se pueden asignar los siguientes perfiles:

- Gestión de almacenamiento de bloques de OpenStack
- Gestión de cálculos de OpenStack
- Gestión de identidades de OpenStack
- Gestión de imágenes de OpenStack
- Gestión de redes de OpenStack
- Gestión de almacenamiento de objetos de OpenStack
- Gestión de OpenStack

Investigación y resolución de problemas

El comando `svcs -x` sin nombre de servicio muestra los servicios con el estado `maintenance`. Use el comando `svcs` en todos los nodos de OpenStack para asegurarse de que todos los servicios necesarios tienen el estado `online`. Si un servicio necesario no tiene el estado `online`, compruebe los archivos log tal como se indica en [“Examen de los archivos log” \[121\]](#).

Para obtener ayuda general para la solución de problemas con servicios SMF, consulte [“Resolución de problemas de servicio” de “Gestión de los servicios del sistema en Oracle Solaris 11.2”](#).

Si es posible que su problema sea un problema de red, ejecute el comando `evsadm` en todos los nodos.

Si tiene problemas para completar correctamente una operación mientras usa el panel de control de OpenStack, intente realizar la misma operación en la línea de comandos. Es posible que reciba un mensaje de error más útil en la línea de comandos. Compruebe si el comando tiene una opción de salida detallada.

Si tiene un problema en un nodo que no es el nodo de controlador, ejecute algunos comandos simples, como `nova list` en la línea de comandos de ese nodo para verificar que el nodo se puede comunicar con el nodo de controlador.

Instalación y configuración de OpenStack

En esta sección, se analizan los errores que se pueden presentar cuando se instala y se configura OpenStack.

Errores del panel de control

Si ve mensajes de error “No autorizado” como los siguientes, compruebe si se ha modificado la clave de host RSA.

```
Error: Unauthorized: Unable to retrieve usage information.  
Error: Unauthorized: Unable to retrieve quota information.  
Error: Unauthorized: Unable to retrieve project list information.  
Error: Unauthorized: Unable to retrieve instance list information.
```

Instalación y configuración de una instancia de VM

Los problemas que se analizan en esta sección están específicamente relacionados con las instancias de VM.

La instancia de VM tiene un estado de error

Un motivo por el que una instancia de VM puede tener un estado de error es que ha intentado instalar una instancia de VM con una arquitectura diferente a la del sistema host. En este caso, es posible que no reciba un mensaje de error que indique específicamente una discrepancia

entre las arquitecturas. Para evitar este problema, asegúrese de configurar correctamente la propiedad `architecture` de la imagen cuando cargue la imagen al almacén de imágenes `glance`. Cuando usa Horizon para cargar una imagen, debe definir las propiedades de la imagen después de la carga. De manera alternativa, puede usar la línea de comandos para cargar la imagen y definir los valores de propiedad en un comando `glance image-create`. Consulte [“Agregación de una imagen al almacén de imágenes” \[116\]](#) para ver un ejemplo.

Los valores de propiedad de la instancia de VM no coinciden con los valores de propiedad de la zona

Parte de la información que OpenStack informa acerca de una instancia de VM no coincide con la información que Solaris informa acerca de la zona correspondiente. Es posible que la información que se muestra en Horizon o que muestra el comando `nova` no coincida con la información que muestra el comando `zoneadm` u otros comandos de Solaris.

Nombre	El nombre de una instancia de VM tal como se muestra en Horizon o como lo muestra el comando <code>nova list</code> es el nombre que asignó cuando creó la instancia, como <code>example-instance</code> . El nombre de la zona que muestra el comando <code>zoneadm list</code> es similar a <code>instance-00000001</code> . Use el comando <code>nova show</code> para determinar cuál zona está asociada con cuál instancia de VM. En la salida de <code>nova show</code> , el valor de la propiedad <code>OS-EXT-SRV-ATTR:instance_name</code> es el nombre de la zona y el valor de la propiedad <code>name</code> es el nombre de la instancia de VM.
UUID	El UUID de una instancia de VM como se muestra en Horizon o como lo muestra el comando <code>nova show</code> no coincide con el UUID de la misma zona, como lo muestra el comando <code>zoneadm list -p</code> . El UUID que muestra el comando <code>zoneadm</code> es un identificador diferente del identificador que se usa para Nova.
CPU	El número de VCPU de una instancia de VM, como se muestra en Horizon, es el número de CPU <i>limitadas</i> que se virtualizan únicamente según la cantidad de CPU en fracciones que puede usar la instancia. Este número no proporciona observabilidad dentro de la instancia que se limitó. El comando <code>psrinfo</code> informa las CPU dedicadas asignadas a la zona.
Memoria	La cantidad de memoria para una instancia de VM, tal como se muestra en Horizon, puede ser diferente de la cantidad de memoria que muestra el comando <code>prtconf</code> cuando ha iniciado sesión en esa instancia de VM. Horizon muestra la cantidad de memoria especificada por el tipo que se usó para crear la instancia de VM. El comando <code>prtconf command</code> informa acerca de toda la memoria del sistema.

Almacenamiento	La cantidad de almacenamiento para una instancia de VM, tal como se muestra en Horizon, puede ser diferente de la cantidad de almacenamiento que se muestra cuando ha iniciado sesión en esa instancia de VM, a menos que la instancia de VM sea una zona no global que usa Zonas en almacenamiento compartido (ZOSS).
----------------	--

Desmantelamiento de redes

Si tiene problemas con la configuración de Neutron en el nodo de red y es posible que necesite desmantelar la configuración para volver a empezar, siga este procedimiento. Según el punto en el que deba volver a iniciar la configuración, siga la secuencia como que se brinda en el procedimiento.

▼ Cómo eliminar la configuración de red en Neutron

1. Realice este paso en el panel de control de Horizon.

- a. Disocie todas las direcciones IP flotantes.
- b. Elimine todas las direcciones IP flotantes.

2. En la ventana de terminal, escriba los siguientes comandos:

```
# neutron router-gateway-clear router-id external-network-id
```

```
# neutron router-interface-delete router-id subnet-id
```

- a. Para eliminar la interfaz del enrutador de puerta de enlace, escriba el siguiente comando:

```
# neutron router-gateway-interface-delete router-id external-network-id
```

- b. Para eliminar las interfaces del enrutador restantes, escriba el siguiente comando:

```
# neutron router-interface-delete router-id subnet-id
```

3. Realice lo siguiente en el panel de control de Horizon.

- a. Termine todas las instancias de VM.
- b. Suprima las subredes.

Si ocurren problemas mientras se suprimen las subredes, consulte [Cómo eliminar Vports \[126\]](#).

- c. **Suprima la red.**

▼ **Cómo eliminar Vports**

Utilice este procedimiento si se producen problemas que impiden que suprima subredes.

1. **Determine qué vports están en uso actualmente.**

```
# evsadm
```

2. **Restablezca los vports que esté utilizando.**

```
# evsadm reset-vport vport
```

3. **Elimine los vports.**

```
# evsadm delete-vport vport
```

Índice

A

- agente DHCP, 66
- agente L3, 66, 103
- AI, 13, 20
 - medio de inicio, 23
 - servicio de instalación, 22
- almacén de imágenes *Ver* Glance
- almacenamiento, 13, 13
 - Ver también* Cinder
 - Ver también* Swift
- almacenamiento de bloques *Ver* Cinder
- almacenamiento de objetos *Ver* Swift
- AMQP, 45, 83
- archivo unificado
 - instalación de OpenStack en un solo nodo
 - servicio AI bare-metal, 22
- archivo unificado de OpenStack *Ver* archivos unificados
- archivos unificados
 - instalación de OpenStack (Juno) en un solo nodo, 27
 - instalación de OpenStack en un solo nodo
 - USB bare-metal, 21
 - zona de núcleo, 24
 - instalación de OpenStack en un solo nodo (Juno), 26
- autenticación *Ver* Keystone
- Automated Installer *Ver* AI

B

- base de datos, 48, 83

C

- Cinder, 13

- copia de seguridad, 119
 - instalación, 51, 91
 - ZFS Storage Appliance, 53, 92
- clientes, 31
- comando `archiveadm`, 23
- comando `archiveadm command`, 115
- comando `evsadm`, 65
- comando `usbcopy`, 21
- comando `zlogin`, 25, 28
- comando `zoneadm`
 - subcomando `boot`, 25, 27
 - subcomando `install`, 20
- comando `zonecfg`
 - subcomando `add`, 24, 27
 - subcomando `create`, 24, 27
 - subcomando `select`, 24, 27
 - subcomando `set`, 24, 27
- conmutador virtual elástico *Ver* EVS

D

- Daemon de acceso remoto (RAD), 61, 98
- direcciones IP flotantes, 74, 108

E

- entornos de inicio (BE), 14
- estructura de zonas, 61, 98
- EVS, 12
 - Ver también* Neutron
 - comando `evsadm`, 65, 70, 104
 - configuración, 26
 - configuración en el nodo de red, 65
 - secuencia de comandos de configuración
 - `configure_evs.py`, 26

G

Glance, 13, 116
 instalación, 56, 86

H

Heat, 14
 instalación, 50, 97
 herramienta SCI, 22
 herramienta System Configuration Interactive *Ver*
 herramienta SCI
 Horizon, 13
 configurar acceso SSL, 59, 90
 errores, 121

I

Image Packaging System (IPS), 14
Ver también paquetes
 imagen de rescate, 116
 imágenes, 30, 114
Ver también instancias de VM
 caché de imágenes, 116
 copia de seguridad, 116
 instantáneas, 116
 rescate, 116
 servidor de registro, 116
 inicio de sesión en la consola de la zona, 28
 inicio de sesión en la consola de zona, 25
 instalación
 requisitos, 16
 instalación de Juno OpenStack
 uso de un archivo unificado, 26
 instalación de OpenStack
 configuración de evaluación, 19
 configuración de varios nodos, 41
 configuración en un solo nodo, 19
 configuración en varios nodos, 77
 uso de un Unified Archive, 19
 instalación de OpenStack de un solo nodo
 uso de OVM Server for SPARC, 41
 instalación de OpenStack de varios nodos, 41
 instalación de OpenStack en un solo nodo
 uso de archivo unificado, 19
 uso del servidor OVM para SPARC, 78

instalación de OpenStack en varios nodos, 77
 instancia *Ver* instancias de VM
 instancias de VM, 12, 13, 15
Ver también Nova
 copia de seguridad, 116
 creación, 33, 111, 117
 imágenes, 30, 114
 inicio, 33, 117
 inicio de sesión, 37, 74, 108
 instantáneas, 116
 rescate, 116
 tipos, 29, 32, 111
 instantáneas, 116

K

Keystone, 13
 instalación, 49, 85

L

LDoms, 41, 78

M

máquina virtual (VM), 13 *Ver* instancias de VM
 máquina virtual en la nube *Ver* instancias de VM
 MySQL, 48, 83

N

Neutron, 12
 agente DHCP, 66
 agente L3, 66, 103
 direcciones IP flotantes, 74, 108
 instalación, 57, 94
 nodo de almacenamiento, 77
 configuración, 100
 nodo de cálculo, 41, 77
 configuración, 60, 98
 nodo de controlador, 41, 77
 configuración, 45, 81
 nodo de red, 41
 configuración, 62

Nova, 12
 errores, 121
 instalación en el nodo de controlador, 58
 instalación en nodo de controlador, 88
NTP, 47, 82

O

opciones de depuración, 122
OVM Server for SPARC, 41

P

panel de control, 13
 Ver también Horizon
 creación de una instancia de VM, 33
 inicio de sesión, 28
 inicio de una instancia de VM, 33
 visualización de imágenes, 30
 visualización de proyectos, 31
 visualización de tipos, 32
perfil de configuración del sistema *Ver* perfil SC
perfil de SC, 114
perfil SC, 116
perfiles de RBAC, 14
plantillas
 hardware *Ver* tipos
plantillas de hardware *Ver* tipos
plantillas de instancias *Ver* tipos
Protocolo avanzado de cola de mensajes (AMQP), 45
protocolo avanzado de cola de mensajes (AMQP) *Ver*
AMQP
Protocolo de hora de red (NTP), 47, 82
proyectos, 31

R

RabbitMQ, 45, 83
RAD, 61, 98
red externa
 creación, 103, 104
red interna
 creación, 102
requisitos de instalación del sistema, 16

S

SDN, 12
secuencia de comandos de configuración EVS
 `configure_evs.py`, 26
secuencia de comandos `sample_data.sh`, 85
servicio de cálculo *Ver* Nova
servicio de identidad *Ver* Keystone
servicio de imágenes *Ver* Glance
servicio de orquestación *Ver* Heat
servicio SMF de `neutron-dhcp-agent`, 66
servicio SMF de `neutron-l3-agent`, 71, 105
servicios de OpenStack, 12
servidor de registro, 116
servidor OVM para SPARC, 78
Software Defined Networking (SDN) *Ver* Neutron
SPARC, 41, 78
SQLite, 48, 83
Swift, 13, 45, 79

T

tipos, 32, 111
 Ver también instancias de VM
 extra-specs, 112

U

Unified Archive, 13
 Ver también Glance
 creación de imágenes de OpenStack desde, 115
 descarga, 20
 instalación de OpenStack de un único nodo
 AI de inicio bare-metal, 23
 instalación OpenStack de un único nodo, 19
Unified Archive de OpenStack *Ver* Unified Archive
Utilidad de gestión de servicios (SMF), 14

V

virtualización de red *Ver* Neutron

Z

ZFS, 13, 13

Ver también Cinder
Ver también Swift
ZFS Storage Appliance (ZFSSA), 53, 92
ZFSSA, 53, 92
zonas, 12
zonas de núcleo, 12
zonas no globales, 12