

Instalación y configuración de OpenStack (Juno) en Oracle® Solaris

ORACLE®

Referencia: E62828
Junio de 2016

Referencia: E62828

Copyright © 2014, 2016, 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 (Juno)	11
Novedades de la versión Juno para Oracle Solaris	11
Notas de actualización	13
Cómo utilizar este manual	13
2 Instalación de OpenStack en un solo nodo	17
Implementación de Unified Archive de OpenStack	17
Acerca de la imagen de UA de OpenStack	17
Descarga del archivo de imagen	18
Instalación de la configuración de sistema único	19
▼ Cómo realizar la instalación mediante el archivo de almacenamiento unificado y un servicio de instalación AI	19
▼ Cómo realizar la instalación en una zona de núcleo mediante el archivo Unified Archive	20
Using the OpenStack Dashboard	22
▼ Cómo acceder al panel de control de OpenStack	22
Exploración del panel de control	23
3 Instalación en varios sistemas para una configuración de OpenStack en varios nodos	25
Visión general de la arquitectura de tres nodos	25
Pasos preliminares	28
Preparación de nombres de host, variables y contraseñas	29
Ejemplo de secuencia de comandos de Keystone	30
Acerca de la edición de archivos de configuración	31
Optimización del uso de la memoria	31
Configuración del servidor NTP	31
▼ Cómo configurar el servidor NTP	32

Configuración del nodo de controlador	33
Configuración del cliente NTP	33
Instalación de MySQL	34
Instalación de Keystone	35
Instalación de Glance	36
Instalación de Nova	38
Instalación de Horizon	39
Instalación de Cinder	40
Instalación de Neutron	41
Configuración del nodo de cálculo	44
▼ Cómo configurar el nodo de cálculo	45
▼ Cómo activar el acceso a la consola	48
Configuración del nodo de almacenamiento	49
▼ Cómo configurar el nodo de almacenamiento de bloques	50
Configuración del almacenamiento de objetos Swift	51
▼ Cómo configurar el nodo de servicio de controlador proxy de Swift	52
▼ Cómo configurar un nodo de almacenamiento de objetos	53
4 Tareas posteriores a la instalación y tareas de configuración	55
Preparación de la red externa para proyectos de OpenStack	55
Acerca del enrutador de proveedor	55
Creación de red externa	57
Preparación de imágenes para el repositorio de Glance	61
Creación de imágenes	62
Visualización de información sobre imágenes	63
Uso de secuencia de comandos de creación de imágenes de Glance	65
5 Gestión de la nube	67
Creación de proyectos y usuarios	67
▼ Cómo crear un proyecto y asignar usuarios	67
▼ Cómo agregar usuarios existentes a un proyecto	68
Creación de redes internas para proyectos	70
▼ Cómo configurar una red para un proyecto	70
▼ Cómo asociar una dirección IP flotante a un proyecto	73
▼ Cómo proporcionar conectividad externa a las redes internas	73
Visualización de información de configuración del agente L3	74
Creación e inicio de una instancia de VM	75
▼ Cómo crear un par de claves SSH	75
▼ Cómo crear una instancia de VM	76

▼ Cómo agregar usuarios a una instancia de VM	79
Gestión de tipos	81
Visualización de la información sobre tipos	81
Modificación de las especificaciones de tipos	82
Gestión de instancias de VM	84
Migración y evacuación de instancias de VM	84
Cambio de tamaño de una instancia de VM	86
6 Opciones para la configuración y el despliegue de Cinder	91
Despliegue de sistemas remotos para almacenamiento	91
Configuración del archivo <code>cinder.conf</code>	92
Otorgamiento de derechos al usuario designado	95
Activación del host remoto como destino	95
Uso de OpenStack con Oracle ZFS Storage Appliance	96
Acerca de Oracle ZFS Storage Appliance	96
Configuración de OpenStack con Oracle ZFSSA	96
▼ Cómo configurar Oracle ZFSSA for OpenStack	98
7 Cómo trabajar con el componente Heat	101
Acerca del componente Heat	101
Instalación de Heat	102
▼ Cómo configurar Heat	102
Acerca de las plantillas HOT	102
8 Cómo trabajar con el componente Ironic	105
Acerca del componente Ironic	105
Instalación y configuración de Ironic	106
▼ Cómo instalar y configurar Ironic	106
Visión general: Implementación de reconstrucción completa con Ironic	111
Uso de Ironic para la implementación de reconstrucción completa	113
▼ Cómo implementar la reconstrucción completa a partir de un archivo UAR	113
▼ Cómo interrumpir la comunicación de un nodo	116
9 Solución de problemas de OpenStack	119
Ayuda de la línea de comandos	119
Limitaciones conocidas	120
Examen de los archivos log	121

Investigación y resolución de problemas	123
Instalación y configuración de una instancia de VM	124
Problema relacionado con Horizon	127
Problema relacionado con Heat	128
Problemas de escalabilidad	129
Desmantelamiento de redes	129
▼ Cómo eliminar la configuración de red en Neutron	129
▼ Cómo eliminar Vports	130
Consejos y trucos generales sobre depuración	130
Sitios útiles	131
A Archivos y servicios de configuración comunes de OpenStack	133
Archivos de configuración	133
Archivos cinder	133
Archivos glance	133
Archivos de Keystone	133
Archivos neutron	134
Archivos nova	134
Archivos de Horizon	134
Archivos de Swift	134
Servicios SMF de OpenStack	135
Cinder	135
Glance	135
Keystone	135
Neutron	135
Nova *	135
Swift	136
Índice	137

Cómo utilizar esta documentación

- **Visión general:** describe cómo instalar OpenStack e implementar máquinas virtuales de OpenStack en sistemas Oracle Solaris 11.3.
- **Destinatarios:** administradores de sistemas de instalaciones de gran tamaño.
- **Conocimientos necesarios:** red de Oracle 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=E69400>.

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 (Juno)

En este capítulo, se describen las funciones que se introdujeron en la versión Juno de OpenStack. También proporciona información sobre cómo usar este manual.

Para obtener una visión general de las tecnologías de OpenStack y Oracle Solaris, consulte *Overview: OpenStack Components and Oracle Solaris Technologies*. Además, consulte *Planning for an OpenStack Configuration* para ver las indicaciones y los requisitos de planificación para instalar OpenStack en Oracle Solaris.

En este capítulo, se tratan los temas siguientes:

- “Novedades de la versión Juno para Oracle Solaris” [11]
- “Notas de actualización” [13]
- “Cómo utilizar este manual” [13]

Novedades de la versión Juno para Oracle Solaris

Para conocer las funciones que se introdujeron en los componentes principales de OpenStack en esta versión, consulte las siguientes notas de la versión y, opcionalmente, la versión anterior:

- [Notas de la versión de Juno \(https://wiki.openstack.org/wiki/ReleaseNotes/Juno\)](https://wiki.openstack.org/wiki/ReleaseNotes/Juno)
- [Notas de la versión de Icehouse \(https://wiki.openstack.org/wiki/ReleaseNotes/Icehouse\)](https://wiki.openstack.org/wiki/ReleaseNotes/Icehouse)

Si desea ver una lista de los problemas de la versión Juno actual, consulte el archivo Readme de la versión correspondiente en [My Oracle Support \(https://support.oracle.com\)](https://support.oracle.com).

Las siguientes funcionalidades de Oracle Solaris se encuentran en Juno OpenStack:

- Compatibilidad de la consola para instancias de VM, donde se puede acceder a la consola de cada instancia desde un explorador en función de una solicitud del usuario.
- Para Horizon, Apache 2.4 se usa de manera predeterminada si está disponible en el sistema. Por lo tanto, se incluyen fragmentos `httpd.conf` para TLS y HTTP en el directorio `samples-conf.d` de Apache 2.4.

- Para Neutron, compatibilidad con IPv6 total para redes de inquilino, así como compatibilidad para NAT de origen (SNAT).

De manera predeterminada, SNAT está activado en la interfaz de la puerta de enlace de un enrutador de Neutron. Para desactivar esta función, ejecute el comando `neutron router-gateway-set --disable-snat`.

Además, en Neutron, se admite lo siguiente en Juno:

- Metadata Services
- Tipo de red de capa 2 plana (sin etiquetas)
- Nuevos subcomandos de Neutron

Para obtener más información sobre estas funciones de Neutron, consulte [Novedades de Solaris OpenStack Juno Neutron](#).

- Se introduce un servicio SMF agregado para asociar con un servicio de actualización en Swift 2.2.2, lo que reconcilia objetos en contenedores de Swift después de la recuperación desde una partición de red.
- Compatibilidad con OpenStack y servicios de metadatos compatibles con EC2.
- A partir de la versión SRU 2 de Oracle Solaris 11.3, se agregaron las siguientes funciones relacionadas con Cinder:
 - Uso de almacenamiento SAN remoto
Además de funcionar en volúmenes ZFS gestionados localmente, con compatibilidad para Paramiko, Cinder también puede aprovisionar estos volúmenes ZFS en nodos Solaris remotos como almacenamiento SAN a través del protocolo iSCSI o de canal de fibra. Para orquestar los sistemas remotos, se los debe configurar en varios backends de almacenamiento.
 - Compatibilidad con operaciones de copia de seguridad y restauración de volúmenes
El servicio SMF de copia de seguridad de Cinder ahora está activado en Oracle Solaris. Por lo tanto, se pueden hacer operaciones de copia de seguridad y restauración de volúmenes no conectados entre backends configurados. Actualmente, Swift es el único backend admitido.
 - Compatibilidad para migración de volúmenes de Cinder
Cinder aprovecha las operaciones de ZFS para migrar volúmenes de Cinder. Los procesos de envío y recepción de ZFS hacen que sea posible la migración de volúmenes entre backends de Cinder con configuraciones distintas. Si el destino de la migración reside en el mismo zpool que el origen, por lo tanto se usa una operación de cambio de nombre de ZFS. Actualmente, la migración de volúmenes de Cinder se limita a un único sistema.
- A partir de la versión SRU 6 de Oracle Solaris 11.3, se agregan las siguientes funciones relacionadas con Nova:
 - Migración en directo
Puede iniciar la migración en directo utilizando el panel de control del servicio de Horizon o el comando `nova`. El programador selecciona al host de destino de la

migración entre los nodos participantes. Estos nodos deben tener las claves públicas de SSH del resto. Consulte “[Migración y evacuación de instancias de VM](#)” [84].

El cifrado garantiza que la operación de migración en directo sea segura. Por medio de un parámetro en el archivo `/etc/nova.conf`, puede seleccionar manualmente el cifrado que desea usar o puede permitir que el proceso seleccione un cifrado adecuado.

- Asistencia para evacuación de instancia

La funcionalidad `nova evacuate`. En caso de que falle un host o se desactiven los servicios en un host, la instancia puede transferirse de forma segura a un nodo diferente y recuperarse desde el nodo. Tenga en cuenta que la asistencia para evacuación está disponible solo si los dispositivos raíz están en un almacenamiento compartido. Además, se admite la evacuación solo para zonas de núcleo, no para zonas no globales.

- Cambio de tamaño de instancias de VM

Se admite el cambio de tamaño de instancias de VM a nuevos tipos. Para una instancia de VM, puede incrementar o reducir el número de CPU y la cantidad de memoria disponible. Además, también puede aumentar el tamaño del volumen raíz disponible para una instancia. Sin embargo, no se admite reducir el tamaño de los volúmenes raíz.

Notas de actualización

La versión Juno de OpenStack se admite en las siguientes versiones de Oracle Solaris:

- Oracle Solaris 11.2 SRU 10 y versiones posteriores
- Oracle Solaris 11.3

Si tiene una configuración de Havana existente en los sistemas que ejecutan la versiones de Oracle Solaris 11.2 SRU 10 o SRU 11, siga los pasos del manual documentados en <https://community.oracle.com/docs/DOC-910993> para actualizar a la versión Juno. A partir de Oracle Solaris 11.2 SRU 12, el proceso de actualización de Havana a Juno es automático.

Nota - Actualmente existe un problema con Neutron si actualiza una configuración existente de Havana a Juno. En ciertos casos, debido a un error de configuración, una base de datos de Neutron se actualiza para usar SQLite en lugar de MySQL. Debido a que SQLite no es compatible y no es adecuado para usar en modo de producción, debe migrar manualmente la base de datos a MySQL. Para obtener más información, consulte [Migración de la base de datos de Neutron de SQLite a MySQL para Oracle OpenStack para Oracle Solaris](#).

Cómo utilizar este manual

En este manual, se incluye principalmente información de OpenStack que es específica para Oracle Solaris. Por lo general, las funciones y las operaciones que son comunes a todas las plataformas no se tratan en este manual.

Para obtener información general sobre la versión Juno en la que se basa OpenStack en esta versión de Oracle Solaris, consulte los siguientes recursos de Juno 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 sobre Oracle Solaris, acceda a la [biblioteca de información](#) de la versión de Oracle Solaris. También hay información adicional disponible en el sitio [Cloud Management](#).

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 Oracle 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>.

◆◆◆ CAPÍTULO 2

Instalación de OpenStack en un solo nodo

En este capítulo, se describe cómo empezar a usar rápidamente OpenStack al instalarlo en un solo sistema y crear una instalación en un solo nodo.

En este capítulo, se tratan los siguientes temas:

- [“Implementación de Unified Archive de OpenStack” \[17\]](#)
- [“Instalación de la configuración de sistema único” \[19\]](#)
- [“Using the OpenStack Dashboard” \[22\]](#)

Tenga en cuenta que el mejor uso de una instalación de OpenStack en un solo nodo es para evaluación y para familiarizarse con su funcionamiento. Para uso de producción real, OpenStack se debe instalar y configurar en varios sistemas o nodos. Para obtener una descripción de la instalación de OpenStack de varios nodos, consulte [Capítulo 3, Instalación en varios sistemas para una configuración de OpenStack en varios nodos](#).

Implementación de Unified Archive de OpenStack

Oracle Solaris proporciona un archivo unificado de OpenStack que le permite obtener rápidamente una configuración de OpenStack en ejecución.

Acerca de la imagen de UA de OpenStack

Cuando instala el archivo, todos los servicios esenciales de OpenStack se instalan y preconfiguran en el sistema y quedan listos para usar. Puede crear máquinas virtuales (VM), también conocidas como instancias de Nova, a las cuales les puede asignar almacenamiento de bloques. Puede conectar estas VM a redes virtuales y, luego, gestionar las VM mediante una interfaz de explorador web.

El archivo unificado de OpenStack proporciona las siguientes características:

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

Descarga del archivo de imagen

Desde la página de archivos unificados de Oracle Solaris, puede descargar una imagen USB o una imagen de UAR. Con cualquiera de las imágenes, puede usar una variedad de métodos de instalación para crear una configuración en un solo nodo según el destino de la instalación:

- Reconstrucción completa.
 - Instalación mediante un servicio de red de Automated Installer.
 - Instalación mediante una imagen USB que se genera a partir de un archivo unificado con el comando `archiveadm` (1M).
 - Instalación indirecta mediante la imagen de inicio de Automated Installer de Oracle Solaris, combinada con el archivo unificado.
- Instalación en zonas de núcleo mediante los comandos estándar de la zona.

Por motivos de simplificación, este capítulo se centra en dos métodos de instalación. Cada método tiene un destino específico de instalación.

- Instalación del archivo unificado sin sistema operativo mediante un servicio de red de Automated Installer.
- Instalación del archivo unificado en una zona de núcleos mediante comandos de zona.

Nota - En este capítulo, se proporcionan los procedimientos generales de instalación. Para obtener instrucciones detalladas, consulte el archivo README que está asociado al archivo. Para obtener más información sobre los métodos de instalación de Oracle Solaris, consulte los manuales de instalación en la [biblioteca](#) de la versión de Oracle Solaris.

▼ Cómo descargar el archivo de imagen

1. En un explorador de Internet, vaya a la página de descarga [Archivos unificados](#).
2. En el separador Descargas, seleccione Solaris y luego Archivos unificados.
3. Lea el acuerdo de licencia y haga clic en el botón Aceptar.
4. Descargue el archivo *.uar de OpenStack adecuado para la arquitectura de su sistema.

5. 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 la configuración de sistema único

En esta sección, se describe cómo instalar el archivo unificado que descargó para crear la configuración de OpenStack en un solo nodo. En esta sección, se describen dos métodos:

- Instalación del archivo unificado sin sistema operativo mediante un servicio de red de Automated Installer.

Para obtener más información sobre Automated Installer, consulte los manuales de instalación en la [biblioteca](#) de la versión de Oracle Solaris.

- Instalación del archivo unificado en una zona de núcleos.

Para obtener más información sobre las zonas del núcleo, consulte *Creación y uso de zonas del núcleo de Oracle Solaris* en la misma [biblioteca](#).

▼ Cómo realizar la instalación mediante el archivo de almacenamiento unificado y un servicio de instalación AI

Antes de empezar Para realizar este procedimiento, debe haber descargado el archivo Archivo unificado en el servidor de instalación de AI. Consulte [Cómo descargar el archivo de imagen \[18\]](#).

1. Realice lo siguiente en el servidor de instalación de AI.

a. Cree o modifique un manifiesto de AI.

Por ejemplo, puede usar el archivo `/usr/share/auto_install/manifest/default_archive.xml` del servidor de instalación de AI y modificarlo de acuerdo con la configuración de AI. Asegúrese de que el manifiesto incluya el siguiente fragmento que se aplica a esta instalación de OpenStack:

```
<software type="ARCHIVE">
  <source>
    <file uri="full-path-to-the-UAR-file" />
  </source>
  <software_data action="install">
    <name>*</name>
```

```
</software_data>
</software>
```

b. Configure un servicio de instalación AI que utilice el manifiesto de AI del paso anterior.

Suponga que descargó el archivo unificado para un sistema SPARC. Después de crear el manifiesto, debe asociarlo con el alias del servicio de AI existente: `default-sparc`. En el siguiente ejemplo, se muestra cómo, en primer lugar, debe mostrar opcionalmente los servicios disponibles y, luego, asociar el manifiesto con el servicio seleccionado. Como paso final, muestra los servicios nuevamente para verificar el resultado de la asociación.

```
# installadm list -m
Service Name      Manifest Name    Type      Status   Criteria
-----
default-sparc    orig_default    derived   default  none
solaris11_3-sparc orig_default    derived   default  none

# installadm create-manifest -n default-sparc \
-m os_manifest -f openstack_manifest.xml -d
Created Manifest: 'os_manifest'

# installadm list -m
Service Name      Manifest Name    Type      Status   Criteria
-----
default-sparc    os_manifest     xml       default  none
                  orig_default    derived   default  none
solaris11_3-sparc orig_default    derived   default  none
```

2. Inicie el sistema de destino.

```
{0} ok boot net - install
```

El proceso de instalación demora un tiempo en completarse.

3. 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.

4. Proporcione la información según lo solicitado por la herramienta SCI.

5. Inicie sesión en el panel de control de Horizon.

Consulte [“Using the OpenStack Dashboard” \[22\]](#).

▼ Cómo realizar la instalación en una zona de núcleo mediante el archivo Unified Archive

Antes de empezar Asegúrese de lo siguiente:

- El sistema de destino que aloja la zona del núcleo cumple con los requisitos de virtualización.
Los requisitos para la instalación se describen en [Planning for an OpenStack Configuration](#).
- El archivo de almacenamiento unificado de OpenStack se ha descargado en el sistema.
Consulte [Cómo descargar el archivo de imagen \[18\]](#).

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.

Consulte la página del comando man [zonecfg\(1M\)](#) para obtener descripciones de los recursos configurables.

En el siguiente ejemplo, se configura la zona con 8 CPU virtuales y una memoria física con un límite de 12 GB.

```
# zonecfg -z kzone-name
zonecfg:OpenStackKZ> select 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 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.

7. Inicie sesión en el panel de control de Horizon.

Consulte [“Using the OpenStack Dashboard” \[22\]](#).

Using the OpenStack Dashboard

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 en la nube.

▼ 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.

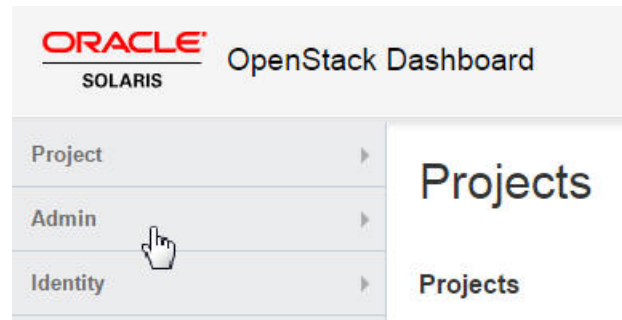
Si realizó la instalación desde Unified Archive en una zona de núcleo, el sistema OpenStack es la zona del núcleo y *system* es el nombre o la dirección IP de la zona de núcleo.

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

- Nombre de usuario: admin
- Contraseña: secrete

Exploración del panel de control

Inicialmente inicia sesión en el panel de control de Horizon como usuario `admin`, lo cual abre la página de llegada demo del proyecto. Debido a que `admin` tiene privilegios administrativos en `demo`, la página muestra 3 separadores en el panel izquierdo: Proyecto, Admin e Identidad. Sin privilegios administrativos, un usuario solo vería los separadores Proyecto e Identidad.



Nota - El nombre de usuario `admin` es independiente de su rol. Por ejemplo, `admin` tiene privilegios administrativos en `demo`. Si agrega `admin` como un usuario a un proyecto nuevo, los privilegios administrativos no se transfieren a ese proyecto. En su lugar, `admin` solo tendría privilegios de miembro y el separador Admin sería inaccesible.

La página Visión general de resumen de uso del panel de Administrador es la vista de administrador de nube por defecto.

FIGURA 1 Ventana Visión general del administrador del panel de control de OpenStack



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 Oracle 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.

El UA de OpenStack de la configuración de un solo nodo proporciona los siguientes recursos preconfigurados:

- Dos imágenes: zona no global de Solaris y zona de núcleo de Solaris
- Dos proyectos o inquilinos: `demo` y `service`

El inquilino `demo` es el inquilino por defecto con `admin` como su miembro único.

El inquilino `service` es el que usa el administrador de nube para crear recursos que se compartirán entre varios inquilinos. Por ejemplo, en el documento, el enrutador Neutron se crea en el inquilino `service` de modo que el enrutador es compartido por todos los inquilinos. No use 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 privilegios administrativos en el inquilino `service`.

- Diez tipos

Para ver las imágenes preconfiguradas en el repositorio que se incluyó en el Unified Archive de OpenStack, haga clic en uno de los siguientes separadores:

- Separador Admin > Sistema > Imágenes.
- Separador Proyecto > Cálculo > Imágenes.

Para mostrar los tipos de Oracle Solaris que están disponibles automáticamente, haga clic en el separador Admin > Sistema > Tipos.

Las siguientes presentaciones de video proporcionan una visión general completa del panel de control:

- [El panel de control de OpenStack: Parte 1](#)
- [El panel de control de OpenStack: Parte 2](#)

Para obtener información acerca de las tareas que puede realizar en el panel de control, consulte [Capítulo 5, Gestión de la nube](#).

◆◆◆ 3 C A P Í T U L O 3

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

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

En este capítulo, se tratan los siguientes temas:

- “Visión general de la arquitectura de tres nodos” [25]
- “Pasos preliminares” [28]
- “Configuración del nodo de controlador” [33]
- “Configuración del nodo de cálculo” [44]
- “Configuración del nodo de almacenamiento” [49]
- “Configuración del almacenamiento de objetos Swift” [51]

Visión general de la arquitectura de tres nodos

Las configuraciones en un solo nodo son útiles para probar OpenStack como producto y para familiarizarse con sus funciones. Sin embargo, la configuración de un solo nodo no es adecuada en un entorno de producción. Para este entorno, debe instalar y configurar OpenStack en varios sistemas o 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. 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.

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

- **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

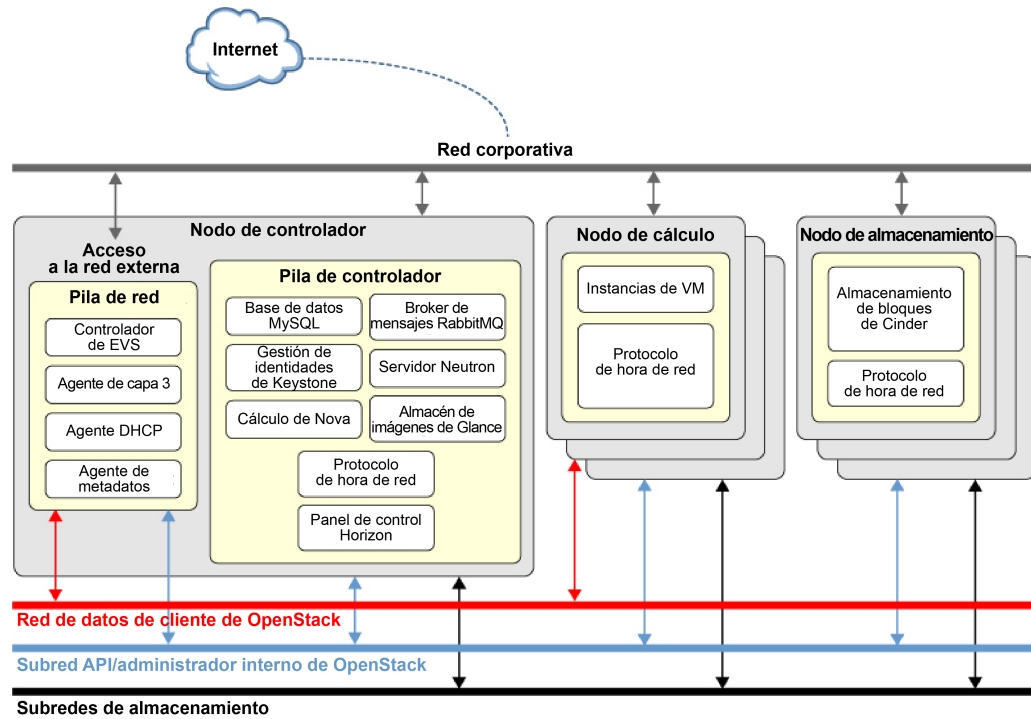
- **Nodo de cálculo:** nodo donde se instalan instancias de VM, también conocidas como instancias de cálculo de Nova. El nodo ejecuta el daemon de cálculo que gestiona estas instancias de VM.
- **Nodo de almacenamiento:** nodo que aloja los datos.

Esta arquitectura de tres nodos es la única manera de implementar OpenStack en varios sistemas. Debido a su flexibilidad, puede distribuir componentes de OpenStack de maneras distintas que esta arquitectura. Por lo tanto, debe planificar la configuración de la nube antes de comenzar la instalación. Para obtener información para orientarlo en la planificación, consulte [Planificación de una configuración de OpenStack](#).

Nota - Para particionar un solo servidor Oracle SPARC y configurar OpenStack de varios nodos en el servidor que ejecuta OVM Server for SPARC (LDoms), consulte [Solaris 11.2 OpenStack de varios nodos en servidores SPARC](#). 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 2 Arquitectura de referencia de configuración de tres nodos

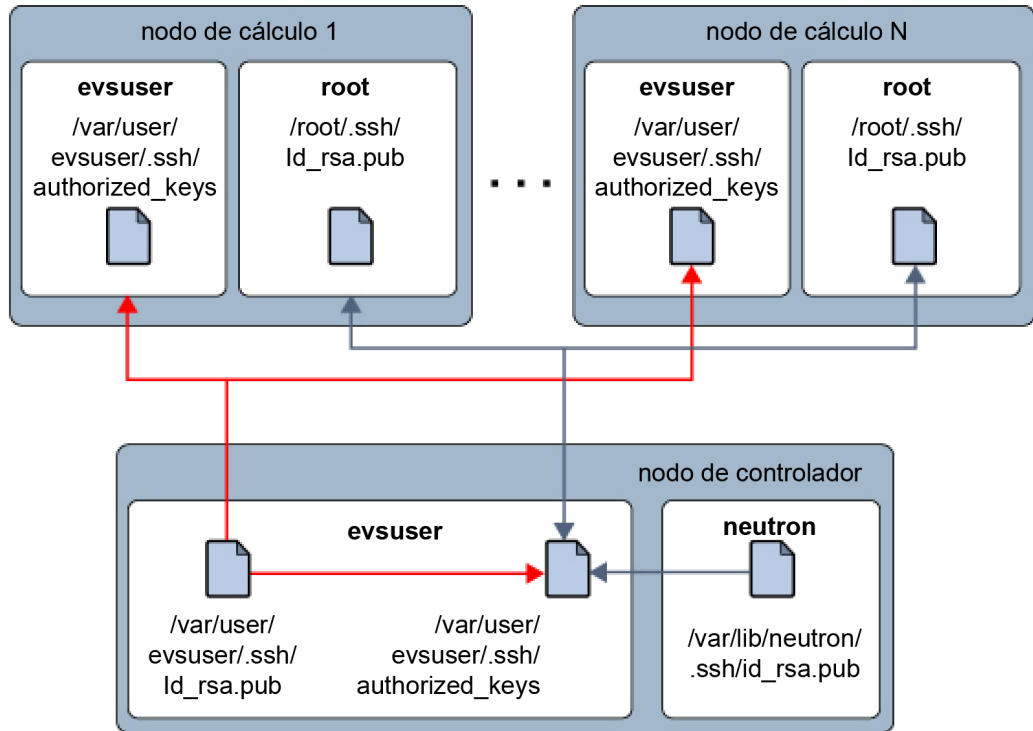


Esta figura utiliza Cinder para mostrar el nodo de almacenamiento. No obstante, el servicio de almacenamiento de objetos Swift se puede configurar de manera similar.

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 sobre el EVS, consulte *Gestión de virtualización de red y recursos de red en Oracle Solaris* para conocer más sobre el conmutador virtual. Este manual se encuentra en la biblioteca de la versión de Oracle Solaris en [Documentación de sistemas operativos](#).

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 han configurado varios nodos de cálculo.

FIGURA 3 Distribución de claves SSH en controlador EVS



Para obtener una lista de los parámetros de configuración de OpenStack que son útiles para los despliegues de OpenStack en sistemas Oracle Solaris, consulte [Apéndice A, Archivos y servicios de configuración comunes de OpenStack](#).

Pasos preliminares

En esta sección, se tratan algunas consideraciones preliminares antes de implementar la configuración de OpenStack en varios nodos.

Preparación de nombres de host, variables y contraseñas

En una configuración en varios nodos, debe usar varias interfaces de red para prestar servicio a las distintas subredes que crea para la nube. Asegúrese de haber preparado nombres de host para estas interfaces. Incluya estos nombres y las correspondientes direcciones IP en el archivo `/etc/hosts` de los sistemas o en la configuración de DNS.

Por ejemplo, puede crear los siguientes nombres de host para manejar diferentes tipos de tráfico de red.

- `host-on` para la red de OpenStack que aloja tráfico administrativo y de API.
- `host-tn` para la red del proyecto que aloja el tráfico entre nodos de cálculo y el enrutador L3.
- `host-en` para el tráfico de red externo.

Cuando configura los servicios de OpenStack en distintos nodos, cree variables para facilitar la tarea, como en los siguientes ejemplos:

- `$CONTROLLER_ADMIN_NODE`: el nombre de host de la interfaz o la dirección IP en el nodo de controlador al que están conectados servicios administrativos de OpenStack.
- `$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`: nombre de host del nodo de almacenamiento.

En el proceso de configuración también se necesitan contraseñas. A continuación, se muestra una lista de contraseñas que debe preparar.

- Contraseña de usuario root para la base de datos de MySQL
- Contraseña del usuario invitado de RabbitMQ
- Contraseña de usuario admin
- Contraseñas de base de datos de los servicios de OpenStack:
 - Servicio de identidad
 - Servicio de imagen
 - Servicio de cálculo
 - Base de datos de panel de control
 - Base de datos de almacenamiento de bloques
 - Base de datos de red
 - Base de datos de orquestación
- Contraseñas de usuarios del servicio de OpenStack:

- glance
- nova
- cinder
- neutron
- heat

Nota - También puede asignar una contraseña común para un grupo de usuarios o servicios. Cualquiera sea el sistema que adopte para asignar contraseñas, asegúrese de cumplir con las mejores prácticas para proteger su entorno. Consulte *Protección de sistemas y dispositivos conectados en Oracle Solaris* en la biblioteca que corresponda a su versión.

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 realiza las siguientes tareas básicas para ayudarlo a comenzar:

- Crea los siguientes proyectos iniciales:
 - `service` en el cual se crean servicios básicos o principales.
 - `demo` en la que se crea el usuario `admin` con la contraseña predeterminada `secrete`.
- Completa la base de datos de Keystone.
- Crea los siguientes servicios principales con los nombres de usuario y contraseñas correspondientes.
 - `cinder`
 - `ec2`
 - `glance`
 - `keystone`
 - `neutron`
 - `nova`
 - `swift`

Los nombres de usuario y contraseñas correspondientes son idénticos a los nombres de los servicios. Por ejemplo, para el servicio de Cinder, el nombre de usuario y contraseña es `cinder`, para el servicio `ec2` es `ec2` y así sucesivamente. Estas contraseñas son de conocimiento público y, por lo tanto, inseguras. Para una configuración de producción, debería crear contraseñas personalizadas para reemplazar las contraseñas por defecto en la secuencia de comandos. Si lo desea, puede establecer una sola contraseña para todos los servicios en la secuencia de comandos. Aplique todos los cambios necesarios a la secuencia de comandos antes de ejecutarla en el arranque de Keystone.

Nota - Asegúrese de revisar la secuencia de comandos para obtener más información acerca de los parámetros que se pueden establecer para el entorno. Sustituya la configuración por defecto en la secuencia de comandos según sus preferencias.

A lo largo de este documento, los procedimientos suponen que, excepto para las contraseñas, se usa la secuencia de comandos de datos de ejemplo sin revisión y se aplican todas las configuraciones por defecto de la secuencia de comandos a la configuración de la nube.

Acerca de la edición de archivos de configuración

Una gran parte de la configuración de OpenStack involucra la edición de los archivos de configuración de los componentes. En este documento, solo se identifican los parámetros seleccionados para su configuración en cada archivo `*.conf` o `*.ini`. Estos parámetros seleccionados son el mínimo requerido para que funcione la configuración de la nube. De todos modos, revise todos los contenidos de cada archivo de configuración para garantizar que todos los parámetros relacionados con la configuración de su nube específica estén configurados adecuadamente.

Optimización del uso de la memoria

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*.

Configuración del servidor NTP

Se recomienda la instalación del Protocolo de hora de red (NTP), aunque es opcional. 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.

El uso de NTP implica la configuración del servidor NTP y del cliente NTP. Normalmente, el servidor NTP es un sistema independiente de los demás sistemas donde configura OpenStack. El cliente NTP se instala y configura en los nodos o sistemas que alojan los componentes de OpenStack.

Para obtener más información acerca de NTP, consulte la documentación que se incluye en <http://www.ntp.org/documentation.html>.

▼ Cómo configurar el servidor NTP

El servidor NTP se encuentra en un sistema distinto que el de los nodos de OpenStack.

1. Instale el paquete NTP.

```
ntp-server# pkg install ntp
```

2. Instale el archivo de configuración.

```
ntp-server# cp /etc/inet/ntp.server /etc/inet/ntp.conf
```

3. Edite el archivo `/etc/inet/ntp.conf` mediante la configuración de las palabras clave `server` y `driftfile`.

Por ejemplo:

```
server 127.127.1.0 prefer
...
driftfile /var/ntp/ntp.drift
```

Nota - `127.127.1.0` **no** es una dirección IP. Es un formato que se utiliza para consultar un reloj que proporciona la hora exacta al servidor. Asegúrese de leer los comentarios que se incluyen en el archivo `ntp.conf`, donde se explica la palabra clave `server`.

4. Cree el archivo `/var/ntp/ntp.drift` como lo definió en el paso anterior.

```
ntp-server# touch /var/ntp/ntp.drift
```

5. Inicie el servicio `ntp`.

```
ntp-server# svcadm enable ntp
```


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 configurar el nodo de controlador, instale los servicios y los componentes de OpenStack en el sistema con el siguiente comando:

```
controller# pkg install openstack
```

Una vez que se ha completado la instalación del paquete, configure los servicios que desea ejecutar en el nodo. En la lista siguiente, se especifican las tareas para configurar el nodo de controlador:

- “Configuración del cliente NTP” [33].
- “Instalación de MySQL” [34].
- “Instalación de Keystone” [35].
- “Instalación de Glance” [36].
- “Instalación de Nova” [38].
- “Instalación de Horizon” [39].
- “Instalación de Cinder” [40].
- “Instalación de Neutron” [41].
- “Instalación de Heat” [102].

Configuración del cliente NTP

Debe instalar el servicio de cliente NTP en cada nodo de servicio en la implementación de la nube.

▼ Cómo configurar el cliente NTP

En esta sección, se supone que ya ha configurado el servidor NTP, como se explica en [Cómo configurar el servidor NTP](#) [32].

1. **Cree el archivo de configuración de cliente.**

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

2. **En el archivo de configuración de cliente, elimine los comentarios de una o varias de las opciones del servidor y proporcione el nombre específico o la dirección IP del servidor NTP.**

Por ejemplo, si el nombre de host del servidor NTP que configuró es `system1`, el archivo de configuración se parecerá al siguiente ejemplo:

```
# multicastclient 224.0.1.1
...
server system1.example.com iburst
# server server_name2 iburst
# server server_name3 iburst
```

3. Active el servicio 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 de 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 ejecución en el nodo de controlador.

▼ Cómo instalar una base de datos MySQL

1. Active los servicios RabbitMQ.

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

2. (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
```

3. Active el servicio MySQL.

```
controller# svcadm enable mysql
```

4. Defina la contraseña root del servidor MySQL.

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

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

```

Instalación de Keystone

El servicio Keystone se debe estar instalado y configurado en el nodo de controlador. Este procedimiento usa la secuencia de comandos de ejemplo que se describe en [“Ejemplo de secuencia de comandos de Keystone” \[30\]](#). Lea la sección anterior usando la secuencia de comandos.

▼ Cómo instalar y configurar Keystone

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

El token se compone de una cadena aleatoria de caracteres. Tenga en cuenta que el comando `openssl` le pide los componentes que constituyen la clave, como país, estado, etc.

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

2. Establezca el token en una variable de shell.

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

3. 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
```

4. Active el servicio SMF de Keystone.

```
controller# svcadm enable keystone
```

5. Complete la base de datos de Keystone usando la secuencia de comandos de ejemplo de Keystone.

Asegúrese de haber revisado y modificado la secuencia de comandos a su preferencia antes de ejecutarla. Los procedimientos suponen que la secuencia de comandos de ejemplo no está personalizada.

```
controller# CONTROLLER_PUBLIC_ADDRESS=$CONTROLLER_ADMIN_NODE \
CONTROLLER_ADMIN_ADDRESS=$CONTROLLER_ADMIN_NODE \
CONTROLLER_INTERNAL_ADDRESS=$CONTROLLER_ADMIN_NODE \
SERVICE_TOKEN=$MY_SERVICE_TOKEN \
ADMIN_PASSWORD=admin-password \
SERVICE_PASSWORD=service-password \
/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. Configure Glance eliminando los comentarios de los parámetros de estos archivos de configuración o estableciendo dichos parámetros:

- /etc/glance/glance-api.conf

```
[DEFAULT]
registry_host = $CONTROLLER_ADMIN_NODE

auth_strategy = keystone
default_publisher_id = image.$CONTROLLER_ADMIN_NODE
rabbit_host = $CONTROLLER_ADMIN_NODE
qpidd_hostname = $CONTROLLER_ADMIN_NODE

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

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

- /etc/glance/glance-cache.conf

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

- /etc/glance/glance-registry.conf

```
[DEFAULT]
default_publisher_id = image.$CONTROLLER_ADMIN_NODE
rabbit_host = $CONTROLLER_ADMIN_NODE
qpidd_hostname = $CONTROLLER_ADMIN_NODE

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

[keystone_authtoken]
auth_uri = http://$CONTROLLER_ADMIN_NODE:5000/v2.0
identity_uri = http://$CONTROLLER_ADMIN_NODE:35357
admin_tenant_name = service
admin_user = glance
```

```
admin_password = service-password
```

- `/etc/glance/glance-scrubber.conf`

```
[DEFAULT]
```

```
auth_url = http://$CONTROLLER_ADMIN_NODE:5000/v2.0/
```

```
identity_uri = http://$CONTROLLER_ADMIN_NODE:35357
```

```
admin_tenant_name = service
```

```
admin_user = glance
```

```
admin_password = service-password
```

```
[database]
```

```
connection=mysql://glance:service-password@$CONTROLLER_ADMIN_NODE/glance
```

2. **Active los servicios SMF de Glance.**

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

Instalación de Nova

Esta sección pertenece a la configuración de servicios de punto final de Nova, no al nodo de cálculo.

▼ Cómo instalar y configurar Nova

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

```
[DEFAULT]
```

```
qpid_hostname=$CONTROLLER_ADMIN_NODE
```

```
rabbit_host=$CONTROLLER_ADMIN_NODE
```

```
my_ip=$CONTROLLER_ADMIN_NODE_IP
```

```
host=$CONTROLLER_ADMIN_NODE
```

```
firewall_driver=nova.virt.firewall.NoopFirewallDriver
```

```
[database]
```

```
connection = mysql://nova:service-password@$CONTROLLER_ADMIN_NODE/nova
```

```
[glance]
```

```
host=$CONTROLLER_ADMIN_NODE
```

```
[keystone_authtoken]
```

```
auth_uri=http://$CONTROLLER_ADMIN_NODE:5000/v2.0/
```

```
identity_uri=http://$CONTROLLER_ADMIN_NODE:35357/
```

```
admin_user=nova
```

```
admin_password=service-password
```

```
admin_tenant_name=service
```

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

2. Defina los parámetros en el archivo `/etc/nova/api-paste.ini`.

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

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.

▼ Cómo configurar Horizon

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

■ Su configuración utiliza HTTP.

a. 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
```

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

```
controller# cp /etc/apache2/2.4/samples-conf.d/openstack-dashboard-http.conf \
/etc/apache2/2.4/conf.d/
```

■ Su configuración utiliza SSL/TLS.

a. 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 0644 ${DASHBOARD}/*
controller# chown websrvd:websrvd ${DASHBOARD}/*

controller# sed \
-e "/SSLCertificateFile/s:/path.*:${DASHBOARD}/horizon.crt:" \
-e "/SSLCertificateFile/d" \
-e "/SSLCertificateKeyFile/s:/path.*:${DASHBOARD}/horizon.key:" \
< /etc/apache2/2.4/samples-conf.d/openstack-dashboard-tls.conf \
> /etc/apache2/2.4/conf.d/openstack-dashboard-tls.conf
```

b. 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 /horizon https://controller-fqdn/horizon
ServerName controller-fqdn
```

2. Inicie el servicio Apache.

```
controller# svcadm enable apache24
```

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

Para el parámetro `volume_driver`, hay disponibles varios controladores para seleccionar. En el siguiente ejemplo, se muestra solo el controlador seleccionado para `volume_driver`. Los otros controladores disponibles que tienen un comentario se excluyen.

```
[DEFAULT]
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSISCSIDriver
my_ip=$CONTROLLER_ADMIN_NODE

[database]
connection = mysql://cinder:service-password@$CONTROLLER_ADMIN_NODE/cinder

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

2. Configure los parámetros en el archivo `/etc/cinder/api-paste.ini`.

```
[filter:authtoken]
admin_tenant_name = service
admin_user = cinder
admin_password = service-password
```

3. Si se configuran los destinos de iSCSI, active los servicios SMF correspondientes.

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

4. 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](#).

Instalación 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. Configure Neutron eliminando los comentarios de los parámetros de estos archivos de configuración o estableciendo dichos parámetros:

- /etc/neutron/neutron.conf

```
qpid_hostname=${CONTROLLER_ADMIN_NODE}
rabbit_host=${CONTROLLER_ADMIN_NODE}

host=${CONTROLLER_ADMIN_NODE}

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

[database]
connection = mysql://neutron:service-password@${CONTROLLER_ADMIN_NODE}/neutron
```

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

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

- /etc/neutron/dhcp_agent.ini

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

- /etc/neutron/l3_agent.ini

```
evs_controller = ssh://evsuser@${CONTROLLER_ADMIN_NODE}
```

2. Configure los pares de claves SSH que se deben utilizar.

- a. 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
```

- b. 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
```

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

3. Configure el conmutador virtual elástico (EVS).

Nota - A continuación, se describen los pasos secundarios para configurar los EVS para una red basada en VLAN.

Para configurar una red basada en VLAN, vaya a *Gestión de virtualización de red y recursos de red en Oracle Solaris* en la biblioteca de la versión de Oracle Solaris en [Documentación de sistemas operativos](#). En ese manual, consulte específicamente la sección *Caso de uso: configuración de un conmutador virtual elástico para un inquilino*.

Para ver un ejemplo de cómo se configuran redes planas, consulte https://blogs.oracle.com/openstack/entry/configuring_the_neutron_l3_agent.

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

```
controller# evsadm set-prop -p controller=ssh://evsuser@$CONTROLLER_ADMIN_NODE
```

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

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

Como se muestra en [Figura 2, “Arquitectura de referencia de configuración de tres nodos”](#), generalmente tendrá varias interfaces de red para prestar servicio a distintas subredes.

Cuando configura la propiedad `uplink-port`, puede separar las VLAN en los distintos puertos de red que prestan servicio a las subredes.

En el siguiente ejemplo, se muestra cómo definir las propiedades de EVS, incluida la división de las VLAN. Opcionalmente, utilice el comando final para mostrar todas las propiedades de EVS.

Nota - Debe definir el rango de VLAN antes de distribuir las VLAN por los puertos de red. De lo contrario, la propiedad `uplink-port` no se puede configurar.

```
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,vlan-range=1
controller# evsadm set-controlprop -p uplink-port=net1,vlan-range=200-250
controller# evsadm set-controlprop -p uplink-port=net1,vlan-range=251-300

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
```

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 configurar el nodo de cálculo, instale los servicios y los componentes de OpenStack en el sistema con el siguiente comando:

```
compute# pkg install openstack
```

Una vez que se ha completado la instalación del paquete, configure los servicios que desea ejecutar en el nodo.

Nota - 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* es un identificador aleatorio, como el nombre de su 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. Configure el cliente NTP.

Consulte “Configuración del cliente NTP” [33].

2. 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
```

3. Para configurar Nova, elimine los comentarios de los siguientes parámetros o establezca dichos parámetros en el archivo `/etc/nova/nova.conf`.

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

[database]
connection = mysql://nova:service-password@$CONTROLLER_ADMIN_NODE/nova

[glance]
host=$CONTROLLER_ADMIN_NODE

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

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

4. Defina los parámetros en el archivo `/etc/nova/api-paste.ini`.

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

5. Configure EVS en el nodo de cálculo.

- a. Asegúrese de que se haya instalado el paquete de EVS.

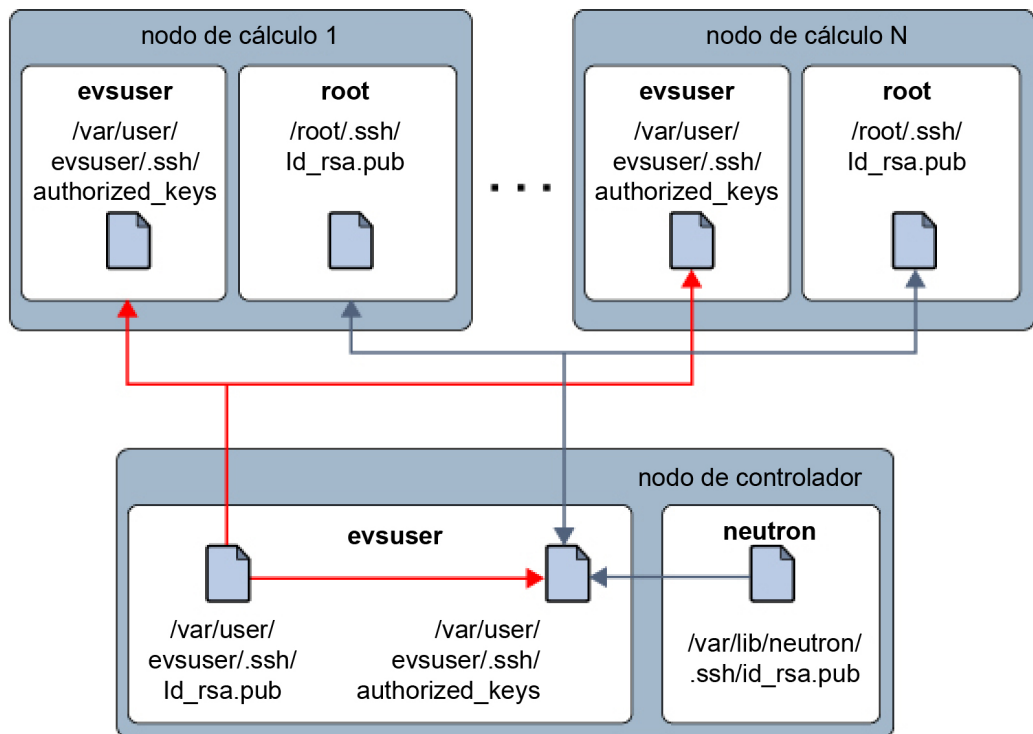
```
compute1# pkg info evs
```

- b. Especifique la ubicación del controlador de EVS.

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

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

La distribución de las claves SSH que establece la comunicación entre los nodos será similar a la de la siguiente figura:



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

```
compute1# 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 `/root/.ssh/id_rsa.pub` que generó en el nodo de cálculo.

- f. **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_ADMIN_NODE true
```

7. Otorgue el acceso del servicio Nova al nodo de cálculo.

- a. **Acceda al nodo de controlador.**

- b. **Use el siguiente comando:**

```
controller# mysql -u root -p
Enter password: MySQL-root-password
mysql> grant all privileges on nova.* to 'nova'@'nova-compute' identified by 'service-
password';
mysql> flush privileges
mysql> quit
```

donde `nova-compute` es el nombre de sistema del nodo de cálculo o el nombre de dominio completo.

8. En el nodo de cálculo, active el servicio de cálculo de Nova.

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

▼ Cómo activar el acceso a la consola

Use este procedimiento para poder usar una consola de instancia de VM desde un explorador en función de una solicitud del usuario.

1. **En cada nodo de cálculo, realice los siguientes pasos según el escenario que corresponda.**

- **Si es posible acceder a las direcciones IP del nodo de cálculo desde la red dirigida al público, establezca los siguientes parámetros en la sección [DEFAULT] del archivo `/etc/nova/nova.conf`.**

```
[DEFAULT]
...
vnc_enabled = true
vncserver_listen = 0.0.0.0
novncproxy_port = 6080
novncproxy_base_url = http://FQDN:6080/vnc_auto.html
novncproxy_host = 0.0.0.0
...
```

Donde *FQDN* representa el nombre de dominio completo o la dirección IP del nodo de cálculo.

- **Si el nodo de cálculo se encuentra en una red privada, establezca los siguientes parámetros en la sección [DEFAULT] del archivo `/etc/nova/nova.conf`.**

```
[DEFAULT]
...
vnc_enabled = true
vncserver_listen = internal-IP
novncproxy_port=6080
novncproxy_base_url = http://public-IP:6080/vnc_auto.html
vncserver_proxycient_address = internal-IP
```

- *internal-IP*: dirección IP del nodo de cálculo en la red interna.
- *public-IP*: dirección IP pública del host del controlador.

2. **Realice los siguientes pasos secundarios si se puede acceder a las direcciones IP del nodo de cálculo desde la red dirigida al público. De lo contrario, salte al siguiente paso.**

a. **Active el servicio `nova-novncproxy`.**

```
compute# svcadm enable nova-novncproxy
```

b. **Reinicie el servicio `nova-compute`.**


```
compute# svcadm restart nova-compute
```

3. En el nodo de controlador, realice los siguientes pasos según el escenario que corresponda.

- **Si es posible acceder a las direcciones IP del nodo de cálculo desde la red dirigida al público, active el servicio nova-consoleauth.**

```
controller# svcadm enable nova-consoleauth
```

- **Si el nodo de cálculo se encuentra en una red privada, realice los siguientes pasos:**

- a. **Establezca los siguientes parámetros en la sección [DEFAULT] del archivo `/etc/nova/nova.conf`.**

```
novncproxy_base_url=http://public-IP:6080/vnc_auto.html
```

donde *public-IP* es la dirección IP pública del host del controlador.

- b. **Active los servicios de Nova como se indica a continuación:**

```
controller# svcadm enable nova-consoleauth
```

```
controller# svcadm enable nova-novncproxy
```

Configuración del nodo de almacenamiento

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

Para configurar el nodo de cálculo, instale los servicios y los componentes de OpenStack en el sistema con el siguiente comando:

```
storage# pkg install openstack
```

Una vez que se ha completado la instalación del paquete, configure los servicios que desea ejecutar en el nodo.

Nota - 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* es un identificador aleatorio, como el nombre de su 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

Este procedimiento describe una configuración típica de un almacenamiento de bloques. Para obtener información acerca de otras opciones para configurar el componente de almacenamiento, consulte [Capítulo 6, Opciones para la configuración y el despliegue de Cinder](#).

1. Configure el cliente NTP.

Consulte [“Configuración del cliente NTP”](#) [33].

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

```
[DEFAULT]
san_is_local=true
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSISCSIDriver
my_ip=$VOLUME_IP
rabbit_host=$CONTROLLER_ADMIN_NODE
glance_host=$CONTROLLER_ADMIN_NODE
zfs_volume_base=cinder/cinder

[database]
connection = mysql://cinder:service-password@$CONTROLLER_ADMIN_NODE/cinder

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

3. Defina los parámetros en el archivo `/etc/cinder/api-paste.ini`.

```
[filter:authtoken]
admin_tenant_name = service
admin_user = cinder
admin_password = service-password
```

4. Otorgue el acceso del servicio Cinder al almacenamiento de bloques o el nodo de volumen.

a. Acceda al nodo de controlador.

b. Use el siguiente comando:

```
controller# mysql -u root -p
Enter password: MySQL-root-password
mysql> grant all privileges on cinder.* to 'cinder'@'$VOLUME_IP' identified by
'service-password';
mysql> flush privileges
mysql> quit
```

5. En el nodo de almacenamiento, inicie los servicios de Cinder.

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

Configuración del almacenamiento de objetos Swift

Swift es el proyecto de almacenamiento de objetos de OpenStack. Ofrece software de almacenamiento en la nube para permitirle almacenar y recuperar grandes volúmenes de datos con una API simple. El servicio es ideal para almacenar datos no estructurados que pueden aumentar sin límites.

Para obtener más información sobre Swift, consulte el capítulo de almacenamiento de objetos en la [Guía del administrador de nubes de OpenStack](#) de la comunidad de OpenStack.

En la documentación de la comunidad de OpenStack, se recomienda un mínimo de seis nodos para una implementación de Swift en modo de producción. Estos nodos constan de un controlador proxy de Swift y cinco nodos de almacenamiento de Swift. No obstante, en esta guía, se describe una implementación de tres nodos que sea acorde a la arquitectura de tres nodos en la que se basan los procedimientos de configuración anteriores. Puede agregar nodos de almacenamiento posteriormente según sea necesario.

▼ Cómo configurar el nodo de servicio de controlador proxy de Swift

Para realizar esta tarea, se supone que ya ha instalado los paquetes de OpenStack en el nodo designado para Swift. En [“Configuración del nodo de almacenamiento” \[49\]](#), consulte el comando de instalación que debe usar.

1. Instale los paquetes de Swift.

```
proxy-node # pkg install swift swiftclient
```

2. Cree las bases de datos de ZFS.

```
proxy-node # /usr/sbin/zfs create -o mountpoint=none rpool/export/swift
proxy-node # /usr/sbin/zfs create -o mountpoint=/srv rpool/export/swift/srv
proxy-node # /usr/sbin/zfs create -p rpool/export/swift/srv/node/disk0
proxy-node # /usr/bin/chown -R swift:swift /srv
```

3. Realice los siguientes vaciados octales.

Retenga los valores de los vaciados. Estos dos valores se conocen como \$OD_1 y \$OPD_2 en pasos posteriores.

```
proxy-node # od -t x8 -N 8 -A n < /dev/random
proxy-node # od -t x8 -N 8 -A n < /dev/random
```

4. Edite el archivo `/etc/swift/swift.conf` con los siguientes parámetros.

```
[swift-hash]
swift_hash_path_suffix = $OD_1
swift_hash_path_prefix = $OD_2
```

5. Edite el archivo `/etc/swift/proxy-server.conf` con los siguientes parámetros.

```
[DEFAULT]
bind_port = 8080

[filter:tempauth]
use = egg:swift#tempauth

operator_roles = admin, swiftoperator

[filter:authtoken]
auth_uri = http://$CONTROLLER_IP:5000/
identity_uri = http://$CONTROLLER_IP:35357
admin_tenant_name = service
admin_user = swift
admin_password = swiftpass

[filter:cache]
```

```
memcache_servers = $CONTROLLER_IP:11211
```

6. Active el daemon memcached.

```
proxy-node # svcadm enable -rs memcached
```

7. Genere los anillos.

```
proxy-node # cd /etc/swift
proxy-node # swift-ring-builder account.builder create 18 3 1
proxy-node # swift-ring-builder container.builder create 18 3 1
proxy-node # swift-ring-builder object.builder create 18 3 1
proxy-node # swift-ring-builder account.builder add r1z1-$STORAGE_IP_1:6002/disk0 100
proxy-node # swift-ring-builder container.builder add r1z1-$STORAGE_IP_1:6001/disk0 100
proxy-node # swift-ring-builder object.builder add r1z1-$STORAGE_IP_1:6000/disk0 100
proxy-node # swift-ring-builder account.builder add r1z1-$STORAGE_IP_2:6002/disk0 100
proxy-node # swift-ring-builder container.builder add r1z1-$STORAGE_IP_2:6001/disk0 100
proxy-node # swift-ring-builder object.builder add r1z1-$STORAGE_IP_2:6000/disk0 100
proxy-node # swift-ring-builder account.builder rebalance
proxy-node # swift-ring-builder container.builder rebalance
proxy-node # swift-ring-builder object.builder rebalance
proxy-node # chown -R swift:swift /etc/swift
```

8. Active el servicio Swift.

```
proxy-node # svcadm enable swift-proxy-server
```

▼ Cómo configurar un nodo de almacenamiento de objetos

Repita este procedimiento en cada nodo de almacenamiento de objetos que desea configurar.

1. Instale los paquetes de Swift.

```
storage-node # pkg install swift swiftclient
```

2. Cree las bases de datos de ZFS.

```
storage-node # /usr/sbin/zfs create -o mountpoint=none rpool/export/swift
storage-node # /usr/sbin/zfs create -o mountpoint=/srv rpool/export/swift/srv
storage-node # /usr/sbin/zfs create -p rpool/export/swift/srv/node/disk0
storage-node # /usr/bin/chown -R swift:swift /srv
```

3. Copie archivos desde el nodo del servidor proxy como se indica a continuación:

- a. **Copie el archivo `/etc/swift/swift.conf` del nodo del servidor proxy en el directorio `/etc/swift` del nodo actual.**

b. Copie los siguientes archivos del nodo del servidor proxy en el directorio `/etc/swift` del nodo actual.

- `account.ring.gz`
- `container.ring.gz`
- `object.ring.gz`

4. Active el servicio de replicador de Swift.

```
storage-node # svcadm enable swift-replicator-rsync
```

5. Establezca la propiedad del directorio `/etc/swift` del nodo actual.

```
storage-node # chown -R swift:swift /etc/swift
```

6. Active todos los servicios Swift.

```
storage-node # for x in `svcs -a -o SVC | fgrep swift | \
egrep "account|container|object" | sort` \
do \
    echo Starting $x \
    svcadm enable $x \
done
```

7. En el nodo de controlador, active el acceso y la operación de los servicios Swift para los usuarios.

a. Establezca las variables de shell para Swift.

```
controller# export OS_USERNAME=swift
controller# export OS_PASSWORD=service-password
controller# export OS_TENANT_NAME=service
controller# export OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE:5000/v2.0
```

b. Agregue el rol `swiftoperator` de Keystone.

```
controller# keystone role-create --name swiftoperator
```

c. Asigne el rol `swiftoperator` a los usuarios autorizados de los servicios Swift.

```
controller# keystone user-role-add --user user-name \
--role swiftoperator --tenant tenant-id
```

Pasos siguientes Para permitir que los usuarios usen la nube, complete los preparativos que se describen en [Capítulo 4, Tareas posteriores a la instalación y tareas de configuración.](#)

◆◆◆ 4 C A P Í T U L O 4

Tareas posteriores a la instalación y tareas de configuración

En este capítulo, se describen los procedimientos para completar la instalación y configuración inicial de OpenStack. En este capítulo, se tratan los siguientes temas:

- “Preparación de la red externa para proyectos de OpenStack” [55]
- “Preparación de imágenes para el repositorio de Glance” [61]

Preparación de la red externa para proyectos de OpenStack

La red externa proporciona la conexión entre las redes privadas en la nube y la red pública.

Acerca del enrutador de proveedor

El enrutador proporciona conectividad para proyectar instancias de VM con redes más amplias. El enrutador es compartido por todas las redes de proyecto. Dado que hay solamente un único enrutador, las redes de proyecto no pueden usar direcciones IP superpuestas.

El enrutador realiza traducción de direcciones de red (NAT) bidireccional en la interfaz que conecta el enrutador a la red externa. Los proyectos pueden tener tantas IP flotantes (IP públicas) como necesitan o como permite la cuota de IP flotante. Estas IP flotantes se asocian con las instancias de VM que requieren conectividad externa.

La creación del enrutador requiere la configuración del agente Neutron L3. Este agente 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.

Por defecto, el enrutamiento entre redes privadas del mismo proyecto está desactivado. Para cambiar este comportamiento, defina `allow_forwarding_between_networks` en `True` en el archivo de configuración `/etc/neutron/l3_agent.ini`. Reinicie el servicio SMF `neutron-l3-agent` después de definir el parámetro.

▼ Cómo configurar el enrutador para la red externa

En este procedimiento, se muestra cómo crear un enrutador para la red externa. Parte de los pasos requieren editar un archivo de configuración. Por lo tanto, para este procedimiento es más conveniente usar la ventana de terminal que el panel de control de Horizon.

Debe realizar los siguientes pasos en el nodo donde está instalado el servicio Neutron. Este documento localiza el servicio en el nodo de controlador basado en la arquitectura de muestra descrita en los capítulos anteriores.

Antes de empezar Asegúrese de que se haya completado la configuración de Neutron como se describe en [Cómo instalar y configurar Neutron \[42\]](#).

- 1. Active el filtro IP de Solaris, si está desactivado.**

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

- 2. Active reenvío de IP si está desactivado en el host.**

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

- 3. Establezca las variables de shell global para Neutron.**

```
controller# export OS_USERNAME=neutron
controller# export OS_PASSWORD=service-password
controller# export OS_TENANT_NAME=service
controller# export OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE: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. En el archivo `/etc/neutron/l3_agent.ini`, establezca el valor de `router_id` en el ID de enrutador en el paso anterior.**

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

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

- 7. (Opcional) Muestra la información sobre el enrutador.**

Después de agregar la red externa al enrutador, se agrega más información sobre el enrutador.

```
controller# neutron router-show router-name
```


ejemplo 1 Creación de enrutador

En este ejemplo, se muestra cómo crear un enrutador para la red externa.

```

controller# svcadm enable -rs ipfilter

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

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

controller# neutron router-create ext-router
Created a new router:
+-----+-----+
| Field          | Value                               |
+-----+-----+
| admin_state_up | True                                 |
| external_gateway_info |                                     |
| id              | f89b24ed-42dd-48b0-8f4b-fd41887a3370 |
| name            | ext-router                           |
| status          | ACTIVE                               |
| tenant_id      | 7d1caf0854b24becb28df5c5cabf72cc    |
+-----+-----+

```

En este punto, debe actualizar `router_id` en el archivo `/etc/neutron/l3_agent.ini`.

```
router_id = f89b24ed-42dd-48b0-8f4b-fd41887a3370
```

Luego, debe activar el servicio de agente L3.

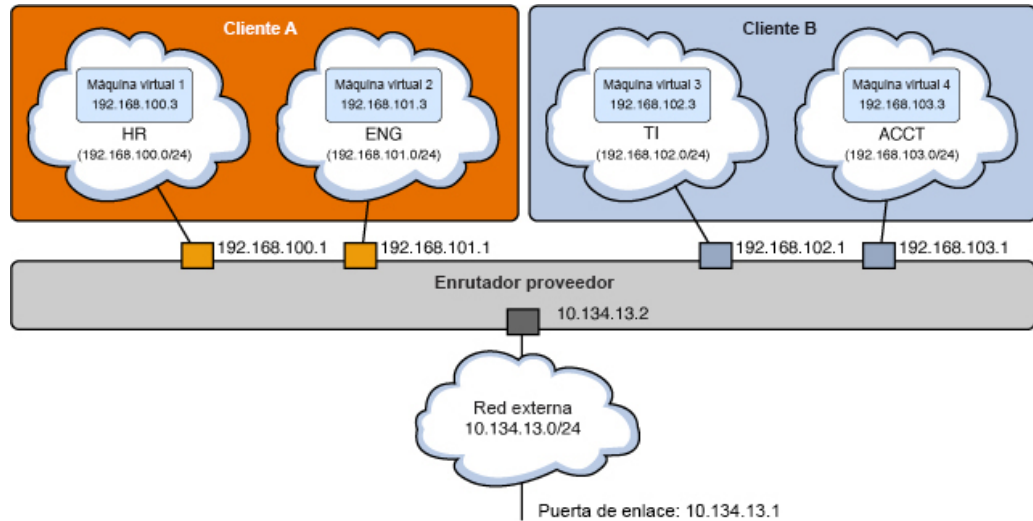
```
controller# svcadm enable neutron-l3-agent
```

Creación de red externa

Después de crear el enrutador, debe configurar la red externa. Mediante el gateway de la red externa, los usuarios en las redes internas de la nube pueden conectarse a Internet.

En la siguiente figura, se muestra cómo se usa un único enrutador para las comunicaciones de red de los proyectos de la nube.

FIGURA 4 Enrutador de proveedor con modelo de redes privadas



La figura muestra lo siguiente:

- Dos proyectos (Tenant A y Tenant B)
- Cuatro VM (VM1, VM2, VM3 y VM4)
- Cuatro subredes (HR, ENG, IT y ACCT)
- Enrutador
- Red externa

Cuando permite que una red interna tenga acceso a la red externa, la dirección IP de la VM se asigna a una de las direcciones flotantes que se asignan a la red externa.

▼ Cómo crear la red externa

En este procedimiento, se muestra cómo crear una red virtual que representa una red externa. Esta red virtual no usa DHCP. En cambio, se crean y asignan direcciones IP flotantes a los proyectos, y las usan las instancias de VM de Nova de esos proyectos. Estos pasos crean un tipo de red VLAN, pero el procedimiento se aplica para crear otros tipos de red, como una red plana.

Puede crear la red externa de forma independiente a la creación de redes internas.

Antes de empezar Complete la configuración del conmutador virtual elástico. Para obtener más información, consulte [Cómo instalar y configurar Neutron \[42\]](#), particularmente los pasos para configurar EVS.

1. Establezca las variables de shell global para Neutron.

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

2. (Opcional) Obtenga el rango de VLAN.

```
controller# evsadm show-controlprop -p vlan-range
```

3. 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 generalmente 1, el primer número del rango de VLAN definido durante la configuración de EVS.

Nota - Si está creando una red plana, no es necesario que especifique un ID de segmentación.

4. Cree una subred de la red externa.

La agrupación de asignaciones consta de un rango de direcciones IP flotantes que está asignado a la subred.

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

5. Agregue la red externa al enrutador.

```
controller# neutron router-gateway-set router-name ext-network-id
```

Nota - Por defecto, SNAT se activa cuando ejecuta este comando. Con SNAT activado, las VM de la red privada pueden acceder a la red externa. No obstante, no es posible acceder a las instancias en sí desde el exterior de la nube. Para desactivar SNAT, especifique la opción `--disable-snat` con el subcomando `neutron router-gateway-set`.

6. (Opcional) Muestra la información sobre el enrutador.

```
controller# neutron router-show router-name
```

ejemplo 2 Creación de red externa

En este ejemplo, se muestra cómo crear la red externa para prepararla para que las redes internas de la nube la usen.

Para crear una red plana, consulte también el ejemplo proporcionado en la sección 2 de <https://blogs.oracle.com/openstack/tags/juno>.

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

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

```
controller# neutron net-create --router:external=True \
--provider:network_type=vlan --provider:segmentation_id=1 ext-network
Created a new network:
```

```
+-----+-----+
| Field          | Value          |
+-----+-----+
| admin_state_up | True           |
| id             | 08cf49c8-f28f-49c1-933d-bdb1017e0294 |
| name           | ext-network    |
| provider:network_type | vlan          |
| provider:segmentation_id | 1            |
| router:external | True           |
| shared         | False         |
| status         | ACTIVE        |
| subnets       |               |
| tenant_id      | 7d1caf0854b24becb28df5c5cabf72cc |
+-----+-----+
```

```
controller# neutron subnet-create --name ext-subnet --disable-dhcp \
--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             | fce503ff-f483-4024-b122-f2524e3edae1 |
| ip_version      | 4             |
| ipv6_address_mode |               |
| ipv6_ra_mode    |               |
| name           | ext-subnet    |
| network_id      | 08cf49c8-f28f-49c1-933d-bdb1017e0294 |
| tenant_id      | 7d1caf0854b24becb28df5c5cabf72cc |
+-----+-----+
```

```
controller# neutron router-gateway-set ext-router 08cf49c8-f28f-49c1-933d-bdb1017e0294
```

```
Set gateway for router ext-router
```

```
controller# neutron router-show ext-router
+-----+-----+
| Field          | Value                                |
+-----+-----+
| admin_state_up | True                                  |
| external_gateway_info | {"network_id": "08cf49c8-f28f-49c1-933d-bdb1017e0294", |
|                  | "enable_snat": true,                |
|                  | "external_fixed_ips":               |
|                  | [{"subnet_id": "fce503ff-f483-4024-b122-f2524e3edae1", |
|                  | "ip_address": "10.134.13.8"}]} |
| id             | f89b24ed-42dd-48b0-8f4b-fd41887a3370 |
| name           | ext-router                           |
| status         | ACTIVE                                |
| tenant_id      | 7d1caf0854b24becb28df5c5cabf72cc    |
+-----+-----+
```

Véase también ■ [“Limitaciones conocidas” \[120\]](#).

Preparación de imágenes para el repositorio de Glance

Una imagen es la base de las instancias de VM en la nube. Una imagen es un único archivo que contiene un disco virtual que tiene instalado un sistema operativo iniciable. La imagen proporciona una plantilla para la creación de una o varias máquinas virtuales. Por lo tanto, para aprovisionar una VM en la nube, debe crear una imagen primero.

Glance, el servicio de imágenes de OpenStack, 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.

Puede cargar varias imágenes en el repositorio de Glance. Como mejor práctica, cargue imágenes de diferentes tipos de sistema que implementaría en la nube. Por ejemplo, cree imágenes archivadas de zonas no globales, zonas de núcleo y zonas globales. Luego puede implementar una VM de uno de estos tipos rápidamente seleccionando la plantilla adecuada.

Nota - Si siguió las instrucciones en [Capítulo 2, Instalación de OpenStack en un solo nodo](#), los procedimientos crearon automáticamente dos imágenes. En este capítulo, se supone que hay una configuración de varios nodos sin imágenes preconfiguradas.

Creación de imágenes

Para crear imágenes de OpenStack en Oracle Solaris, debe usar la función archivos unificados. Con el comando `archiveadm`, puede crear nuevos archivos unificados desde zonas globales, no globales y de núcleo.

El UA puede ser un archivo de clonación o un archivo de recuperación. Un archivo de *clonación* se basa en el entorno de inicio activo actual. Este archivo de clonación no incluye información de configuración del sistema de la instancia del sistema operativo, como los entornos de inicio inactivos. En cambio, los instaladores fuerzan la reconfiguración o usan la información de configuración proporcionada en un perfil de configuración del sistema (SC). Un archivo de *recuperación* incluye toda la información de configuración del sistema y los entornos de inicio. Por lo tanto, si desea incluir toda la información del sistema en un UA, cree un archivo de recuperación. Para obtener más información sobre los archivos unificados, consulte el manual correspondiente *Uso de archivos unificados para la clonación y la recuperación del sistema en Oracle Solaris* en la [biblioteca](#) de la versión de Oracle Solaris.

Luego, cuando la configuración esté completamente operativa, puede crear una imagen creando una instantánea de una instancia de VM existente. En este caso, la instancia de VM ya está en la nube. Por lo tanto, el comando que debe usar es `nova image-create` en lugar de `archiveadm`. El comando `nova` toma una instantánea de una instancia de VM en ejecución para crear la imagen.

También puede usar imágenes personalizadas para realizar 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.

En Oracle Solaris, puede crear una imagen de OpenStack en tres fases:

1. Cree la zona.
2. Cree el UA de la zona.
3. Cargue el UA a Glance.

Estas fases se combinan en el siguiente procedimiento.

▼ Cómo crear una imagen para OpenStack

A excepción de la sintaxis del comando para crear la zona, se puede usar el resto de los pasos para crear y cargar imágenes tanto de las zonas no globales como de las zonas de núcleo.

El paso para la creación de zonas proporciona solo los comandos básicos. Para obtener instrucciones completas sobre la creación de zonas, consulte la documentación sobre zonas en la [biblioteca](#) de la versión de Oracle Solaris.

- 1. En cualquier sistema, cree la zona y, luego, inicie sesión en ella.**

Luego de iniciar sesión, proporcione la información solicitada.

```
global# zonecfg -z zone-name create
global# zoneadm -z zone-name install
global# zoneadm -z zone-name boot
global# zlogin -C zone-name
```

Nota - Este paso puede tardar un poco.

2. Active SSH root para el acceso de inicio de sesión root de OpenStack.

```
global# zlogin zone-name
root@zone-name# sed /^PermitRootLogin/s/no$/without-password/ < /etc/ssh/sshd_config
> /system/volatile/sed.$$
root@zone-name# cp /etc/ssh/sshd_config /etc/ssh/sshd_config.orig
root@zone-name# cp /system/volatile/sed.$$ /etc/ssh/sshd_config
root@zone-name# exit
```

3. Cree el UA para la zona.

```
global# archiveadm create -z zone-name /var/tmp/archive-name.uar
```

4. Transfiera el UA al sistema donde está instalado Glance.

Este documento supone que Glance está en el nodo de controlador.

5. Establezca las variables de shell global para Glance.

```
controller# export OS_USERNAME=glance
controller# export OS_PASSWORD=service-password
controller# export OS_TENANT_NAME=service
controller# export OS_AUTH_URL=$CONTROLLER_ADMIN_NODE:5000/v2.0
```

6. Cargue el UA al repositorio de Glance.

```
controller# glance image-create --container-format bare --disk-format raw \
--is-public true --name "image-name" \
--property architecture=system-arch --property hypervisor_type=solariszones \
--property vm_mode=solariszones < path-to-archive-file
```

donde *system-arch* hace referencia a la arquitectura del sistema que puede ser *x86_64* o *sparc64*.

Visualización de información sobre imágenes

Para visualizar información de una imagen, puede usar el comando `nova` o el comando `glance`.

```
$ nova image-list
```

```
+-----+-----+-----+-----+
```

ID	Name	Status	Server
4dfbfd4f-2de5-4251-832c-e35a4a4145ee	Solaris Non-global Zone	ACTIVE	

El comando `glance image-list` muestra información adicional, como el formato del disco, el formato del contenedor y el tamaño de las diferentes imágenes.

Los comandos `nova image-show` y `glance image-show` muestran información acerca de una imagen específica. Cada comando genera una salida diferente sobre la imagen.

```
$ nova image-show 'Solaris Non-global Zone'
```

Property	Value
OS-EXT-IMG-SIZE:size	845025280
created	2015-11-19T14:46:38Z
id	4dfbfd4f-2de5-4251-832c-e35a4a4145ee
metadata architecture	x86_64
metadata hypervisor_type	solariszones
metadata vm_mode	solariszones
minDisk	0
minRam	0
name	Solaris Non-global Zone
progress	100
status	ACTIVE
updated	2015-11-19T14:46:42Z

```
$ glance image-show 'Solaris Non-global Zone'
```

Property	Value
Property 'architecture'	x86_64
Property 'hypervisor_type'	solariszones
Property 'vm_mode'	solariszones
checksum	ba9b9eeddb467833d725c8750a46e004
container_format	bare
created_at	2015-11-19T14:46:38
deleted	False
disk_format	raw
id	4dfbfd4f-2de5-4251-832c-e35a4a4145ee
is_public	True
min_disk	0
min_ram	0
name	Solaris Non-global Zone
owner	7d1caf0854b24becb28df5c5cabf72cc
protected	False
size	845025280
status	active
updated_at	2015-11-19T14:46:42

Nota - Puede obtener la misma información de la imagen en el panel de control de Horizon.

Uso de secuencia de comandos de creación de imágenes de Glance

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
```


◆◆◆ 5 CAPÍTULO 5

Gestión de la nube

En este capítulo, se describe cómo realizar diferentes tareas administrativas en la nube. Puede usar el panel de control o las líneas de comandos para realizar estas tareas. En el panel de control, las tareas en el separador Proyecto requieren solo el rol de miembro mientras que las tareas del separador Admin requieren privilegios administrativos. Para realizar todas las tareas de un proyecto en una sola sesión en el panel de control, debe tener tanto rol de miembro como de administrador para ese proyecto.

En este capítulo, se tratan los siguientes temas:

- “Creación de proyectos y usuarios” [67]
- “Creación de redes internas para proyectos” [70]
- “Creación e inicio de una instancia de VM” [75]
- “Gestión de tipos” [81]
- “Gestión de instancias de VM” [84]

Creación de proyectos y usuarios

El primer inicio de sesión a OpenStack como usuario admin lo lleva a la página de llegada del proyecto demo. A partir de este proyecto, puede continuar creando otros proyectos.

▼ Cómo crear un proyecto y asignar usuarios

Use este procedimiento para crear proyectos o inquilinos nuevos, y completarlos con usuarios nuevos.

1. **En su explorador, inicie sesión como administrador de la nube en la URL similar al siguiente enlace:**

`http://system/horizon/`

Se muestra la página de llegada del proyecto demo.

Para obtener una visión general del panel de control, consulte [“Exploración del panel de control” \[23\]](#)

2. **En el panel izquierdo, seleccione el separador Identidad > Proyectos.**
Se muestran los proyectos por defecto para demo y service.
3. **Haga clic en Crear proyecto.**
4. **En el separador de información de proyecto, especifique el nombre y la descripción del proyecto.**
Después de su creación, el proyecto se agrega a la lista de proyectos.
5. **Seleccione el separador Identidad > Usuarios.**
Se muestran los usuarios por defecto para demo y service.
6. **Haga clic en Crear usuario.**
7. **Proporcione la información necesaria en los campos adecuados.**
 - a. **Especifique el nombre de usuario nuevo y la contraseña asignada.**
 - b. **Desde la lista desplegable Proyecto principal, seleccione el proyecto al que pertenece el usuario nuevo.**
 - c. **(Opcional) Desde la lista desplegable Rol, seleccione el rol para el usuario del proyecto.**
Por defecto, un usuario nuevo de un proyecto tiene rol de miembro.

▼ **Cómo agregar usuarios existentes a un proyecto**

Use este procedimiento para agregar usuarios existentes a un proyecto nuevo.

1. **En el panel izquierdo de la pantalla por defecto, seleccione el separador Identidad > Proyectos.**
2. **Haga clic en Modificar usuarios del proyecto al que desea agregar usuarios existentes.**
Aparece el cuadro de diálogo Editar proyecto.
3. **En la lista Todos los usuarios, haga clic en el signo más (+) en el lado derecho del nombre de usuario que desee agregar al proyecto.**

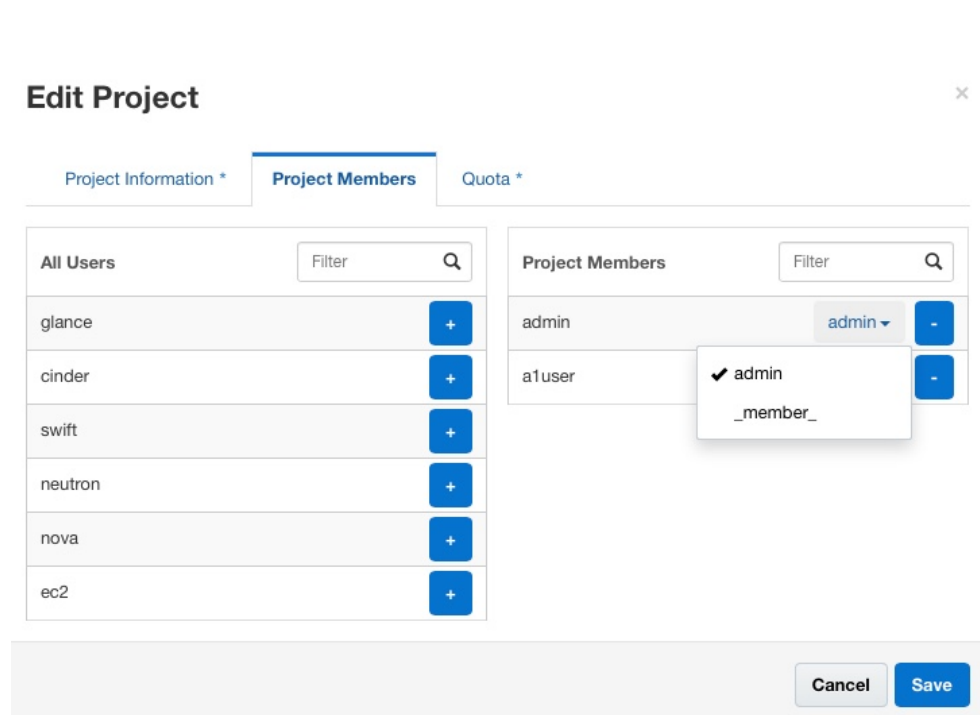
Por defecto, el usuario agregado tiene rol de miembro en ese proyecto.

Nota - No agregue estos usuarios del proyecto service a otros proyectos:

- glance
 - cinder
 - swift
 - neutron
 - nova
 - ec2
-

4. **(Opcional) Para modificar el rol de cualquier usuario del proyecto, siga los pasos siguientes:**
 - a. **En la lista de miembros del proyecto, abra la lista desplegable del usuario cuyo rol desea cambiar.**
 - b. **Seleccione el rol nuevo que desea asignar al usuario.**

En esta figura de ejemplo, los miembros de proyecto del proyecto actual son a1user y admin. Al usuario admin se le otorgan privilegios administrativos al proyecto. Se pueden asignar ambos roles a un usuario (administrador y miembro).



Creación de redes internas para proyectos

Un proyecto puede tener varias redes internas, cada una presta servicio a una instancia de máquina virtual. Por defecto, la comunicación de usuario está limitada solo dentro de la red. Su sitio debe tener una red operativa antes de poder configurar las redes de la nube.

▼ Cómo configurar una red para un proyecto

Para crear redes internas para un proyecto, es necesario ser al menos un miembro del proyecto. No es necesario tener privilegios administrativos para realizar este procedimiento.

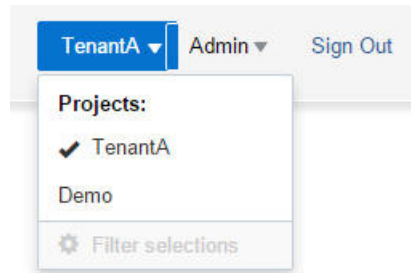
Por defecto, esta red no tiene acceso a la red externa. Para conectar la red interna a la red pública, consulte [Cómo proporcionar conectividad externa a las redes internas \[73\]](#).

1. **En su explorador, inicie sesión en el panel de control de Horizon usando una dirección URL similar a la siguiente:**

`http://system/horizon/`

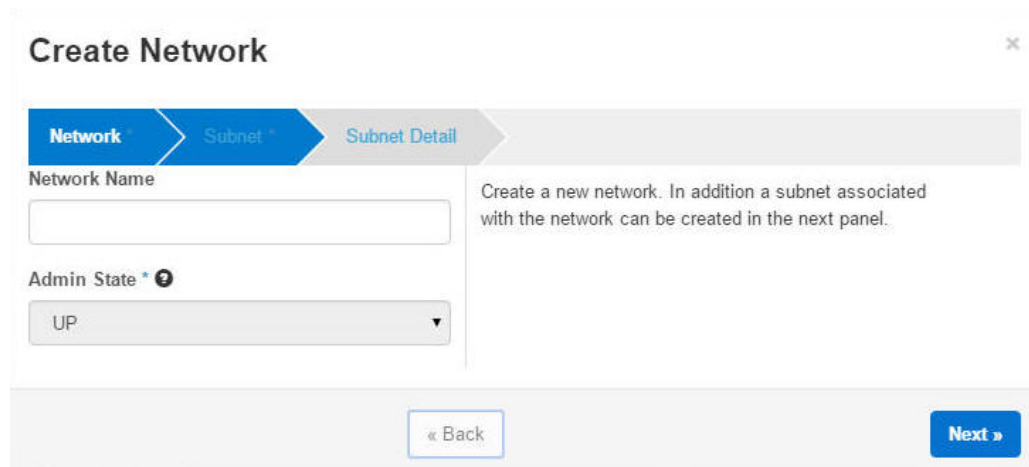
2. **Asegúrese de haber iniciado sesión en el proyecto correcto comprobando el nombre del proyecto en la esquina superior derecha de la pantalla por defecto.**

El siguiente ejemplo muestra que está seleccionado TenantA entre dos proyectos disponibles. El usuario admin ha iniciado sesión.



3. **En el panel izquierdo, seleccione el separador Proyecto > Red > Redes, luego haga clic en Crear red.**

Se muestra el cuadro de diálogo Crear red.



4. **Pase los separadores haciendo clic para proporcionar la información necesaria.**

Nota - Haciendo clic en Siguiete se muestra la pantalla para cada separador.

Cada separador le solicita la siguiente información:

- Separador Red
 - Nombre de red
 - Estado de administrador (acepte el valor por defecto).
- Separador Subred
 - Nombre de subred
 - Dirección de red
 - Versión de IP
 - IP de gateway (deje en blanco para aceptar el valor por defecto).
- Separador Detalle de subred
 - Usar DHCP (anule la selección si no usa DHCP).
 - Agrupaciones de asignación
 - Servidor de nombres DNS
 - Rutas de host

Después de proporcionar la información y hacer clic en el botón Crear, la pantalla de redes muestra las redes y sus subredes asociadas de forma similar al siguiente ejemplo:

Networks

Networks

<input type="checkbox"/>	Name	Subnets Associated
<input type="checkbox"/>	HR	hr_subnet 10.132.30.0/24
<input type="checkbox"/>	ENG	eng_subnet 10.132.35.0/24

Displaying 2 items

▼ Cómo asociar una dirección IP flotante a un proyecto

Parte de la configuración de la red es la creación de direcciones IP flotantes. Consulte [Cómo crear la red externa \[58\]](#). Use este procedimiento para asignar algunas de las direcciones IP a un proyecto.

1. **En el panel izquierdo, seleccione el separador Proyecto > Cálculo > Acceso y seguridad.**
2. **Haga clic en el separador IP flotantes.**
3. **Haga clic en el botón Asignar IP a proyecto.**
Se abrirá el cuadro de diálogo Asignar IP flotante.
4. **En el menú desplegable, seleccione una agrupación desde donde se asignará la IP flotante.**
5. **Haga clic en el botón Asignar IP en el cuadro de diálogo.**
Se agrega una dirección IP a la lista de IP flotantes. Puede asignar todos los IP que desee o que permita la cuota.

▼ Cómo proporcionar conectividad externa a las redes internas

Use este procedimiento para activar una red interna para acceder a la red pública más amplia. Este procedimiento supone que ya existen redes internas para proyectos específicos. Para crear una red interna de proyecto utilizando el panel de control, consulte [“Creación de redes internas para proyectos” \[70\]](#).

Antes de empezar Antes de continuar, debe obtener el nombre de subred que necesita acceso público.

1. **Establezca las variables de shell global para Neutron.**

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

2. **Identifique el ID de la subred que requiere acceso externo.**

```
controller# neutron subnet-list | grep subnet-name
```

3. **(Opcional) Obtenga el nombre del enrutador.**

```
controller# neutron router-list
```

4. Agregue el ID de la subred como interfaz al enrutador.

```
controller# neutron router-interface-add router-name subnetID
```

ejemplo 3 Conexión de una red interna a una red externa

Este ejemplo usa la red interna HR creada en [Cómo configurar una red para un proyecto \[70\]](#). La red HR, cuya subred es HR_Subnet, requiere acceso a la red pública.

```
controller# neutron subnet-list | grep HR_Subnet
| b6feff42-36aa-4235- | HR_Subnet | 10.132.20.0/24 | {"start": "10.132.20.2", |
| 9fe0-ac5de6b43af3 | | | "end": "10.132.20.254"} |
```

```
controller# neutron router-list
+-----+-----+-----+
| id | name | external_gateway_info |
+-----+-----+-----+
| f89b24ed-42dd-48b0- | ext-router | {"network_id": "6c4c1823-a203- |
| 8f4b-fd41887a3370 | | 43b1-9674-ddb5ff4185fc", |
| | | "enable_snat": true, |
| | | "external_fixed_ips": |
| | | [{"subnet_id": "83d9b40f-cc61- |
| | | 4696-b22e-b4cbc2aa3872", |
| | | "ip_address": "10.132.10.8"}]} |
+-----+-----+-----+
```

```
controller# neutron router-interface-add ext-router b6feff42-36aa-4235-9fe0-ac5de6b43af3
Added interface b6feff42-36aa-4235-9fe0-ac5de6b43af3 to router ext-router.
```

Visualización de información de configuración del 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`.

- Para mostrar las VNIC creadas por `neturon-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         0
```

- Para mostrar las direcciones IP creadas por `neturon-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
```

- Para mostrar 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; };
```

- Para mostrar las reglas NAT de IP:

```
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]
```

Creación e inicio de una instancia de VM

Para realizar los procedimientos en esta sección, como mínimo debe ser un miembro válido del proyecto. No se necesitan privilegios administrativos.

▼ Cómo crear un par de claves SSH

1. En su explorador, inicie sesión en el panel de control de Horizon usando una dirección URL similar a la siguiente:

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

2. Asegúrese de haber iniciado sesión en el proyecto correcto comprobando el nombre del proyecto en la esquina superior derecha de la pantalla por defecto.

3. **En el panel izquierdo, haga clic en el separador Proyecto > Cálculo > Acceso y seguridad.**
4. **En el separador Pares de claves, determine si debe crear o importar un par de claves.**
 - **Cree un par de claves.**
 - a. **Haga clic en el botón Crear par de claves.**
 - b. **Especifique un nombre en el campo Nombre de par de claves.**
 - c. **Haga clic en el botón Crear par de claves.**

El par de claves nuevo se descarga automáticamente. Si no es así, haga clic en el enlace que se muestra para descargarlo.

El par de claves nuevo se muestra en el separador Pares de claves del panel Acceso y seguridad.
 - **Importar un par de claves.**
 - a. **Haga clic en el botón Importar par de claves.**
 - b. **Especifique un nombre en el campo Nombre de par de claves.**
 - c. **Desde la ventana terminal, copie el contenido del archivo `.ssh/id_rsa.pub` del usuario `root` y péguelo en el campo Clave pública.**
 - d. **Haga clic en el botón Importar par de claves.**

El par de claves nuevo se muestra en el separador Pares de claves del panel Acceso y seguridad.

▼ **Cómo crear una instancia de VM**

Antes de empezar Asegúrese de tener un par de claves SSH. Consulte [Cómo crear un par de claves SSH \[75\]](#).

Asegúrese de tener una red interna definida. Consulte [“Creación de redes internas para proyectos” \[70\]](#).

1. **En su explorador, inicie sesión en el panel de control de Horizon usando una dirección URL similar a la siguiente:**

`http://system/horizon/`

2. **Asegúrese de haber iniciado sesión en el proyecto correcto comprobando el nombre del proyecto en la esquina superior derecha de la pantalla por defecto.**
3. **En el panel izquierdo, haga clic en el separador Proyecto > Cálculo > Instancias y, luego, haga clic en Iniciar instancia.**
Se mostrará el siguiente cuadro de diálogo Iniciar instancia.

Launch Instance
✕

Details

Access & Security

Networking

Availability Zone

Instance Name

Flavor

Instance Count

Instance Boot 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

Number of VCPUs

Total RAM

Cancel

Launch

4. **Proporcione la información que se le solicita en cada separador.**
Especifique la información de los siguientes campos:

- Separador Detalles
 - Nombre de la Instancia
 - Tipo: de la lista desplegable, seleccione el tipo adecuado. Si este sistema OpenStack es una zona de núcleo y no un sistema bare-metal, deberá seleccionar un tipo de zona no global.
 - Origen de inicio de instancia: de la lista desplegable, seleccione el inicio desde la imagen. Luego, seleccione el nombre de la imagen que desea utilizar. Si este sistema OpenStack es una zona de núcleo y no un sistema bare-metal, deberá seleccionar una imagen de zona no global.

El tipo y el tipo de imagen deben coincidir. Por ejemplo, si la imagen es de una zona no global, el tipo debe ser también una zona no global.
 - Separador Acceso y seguridad: seleccione el par de claves que desea usar.
 - Separador Red: de las redes disponibles, seleccione la red a la que conectará la VM nueva.
- 5. **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.

Este paso tarda un poco.

- 6. **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. **De la lista desplegable en la columna Acciones, seleccione Asociar IP flotante.**

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

- b. **Seleccione una dirección del menú desplegable Dirección IP.**

Si no hay direcciones IP disponibles, haga clic en el botón +. Consulte [Cómo asociar una dirección IP flotante a un proyecto \[73\]](#).

- c. **Seleccione el puerto que corresponda a la VM creada.**

- 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 archivos unificados 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 usuario root con el par de claves y la dirección IP flotante:

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

▼ Cómo agregar usuarios a una instancia de VM

En Oracle Solaris, la instancia de VM usa la tecnología de zonas de Oracle Solaris para permitirle aprovisionar máquinas virtuales en la nube. Para agregar usuarios a la instancia de VM, debe ejecutar los comandos como administrador de zona. Los pasos no se admiten en el panel de control. Por lo tanto, debe acceder a una ventana de terminal.

Antes de empezar Obtenga la dirección IP flotante de la red externa para la que está asociada la instancia de VM haciendo clic en el separador Admin > Sistema > Instancias del panel de control.

1. En una ventana de terminal, muestre las instancias de VM.

```
# zoneadm list -cv
```

Los nombres de las VM en la nube tienen el prefijo `instance`.

2. Inicie sesión en la zona específica.

```
# zlogin zonename
```

3. Cree un directorio de inicio para el usuario.

```
root@zone# mkdir -p /export/home/username
```

4. Cree el usuario.

```
root@zone# useradd -d home-dir options
```

donde `home-dir` es el directorio creado para el usuario. Para obtener otras opciones que pueda usar con el comando `useradd`, consulte la página del comando [man useradd\(1M\)](#).

5. Para crear la contraseña de usuario, ejecute este comando y siga las peticiones de datos.

```
root@zone# passwd username
```

6. (Opcional) Verifique que se cree la contraseña.

```
root@zone# grep username /etc/passwd
```

7. Para crear la contraseña de usuario root, ejecute este comando y siga las peticiones de datos.

```
zone# passwd root
```

8. Salga de la zona y cierre la sesión.

9. Inicie sesión con shell seguro en la máquina virtual.

```
# ssh username@floating-IP
```

donde *floating-IP* es la dirección IP flotante asociada de la VM.

ejemplo 4 Agregación de un usuario a una instancia de VM

En este ejemplo, se agrega el nombre de usuario *jsmith* como usuario de VM1.

```
# zoneadm list -cv
ID NAME          STATUS  PATH                                BRAND  IP
0  global         running /                                    solaris shared
6  instance-00000006 running /system/zones/instance-00000006 solaris excl
-  myzone         installed /system/zones/myzone             solaris excl
```

```
# zlogin instance-00000006
[Connected to zone 'instance-00000006' pts/3]
Last login: Wed Jan 6 14:31:18 2016 on pts/2
Oracle Corporation SunOS 5.11 11.3 September 2015
```

```
root@VM1# mkdir -p /export/home/jsmith
root@VM1# useradd -d /export/home/jsmith -m -s /usr/bin/bash jsmith
```

Se crea el usuario *jsmith* con *bash* como el shell por defecto.

```
root@VM1# passwd jsmith
New Password: password
Re-enter new Password: password
passwd: password successfully changed for jsmith
```

```
root@VM1# passwd root
New Password: password
Re-enter new Password: password
passwd: password successfully changed for root
```

```
root@VM1# exit
logout
```

```
[Connection to zone 'instance-00000006' pts/3 closed]
```



```
# ssh jsmith@10.132.10.9
```

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 Oracle 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

Cuando inicia sesión en el panel de control como administrador de nube, puede ver los tipos disponibles en el separador Admin > Sistema > Tipos.

FIGURA 5 Tipos en Oracle Solaris for Oracle OpenStack

<input type="checkbox"/>	Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk	Swap Disk	ID	Public	Metadata	Actions
<input type="checkbox"/>	Oracle Solaris kernel zone - tiny	1	2048MB	10GB	0GB	0MB	1	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris non-global zone - tiny	1	2048MB	10GB	0GB	0MB	6	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris non-global zone - small	4	3072MB	20GB	0GB	0MB	7	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris kernel zone - small	4	4096MB	20GB	0GB	0MB	2	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris non-global zone - medium	8	4096MB	40GB	0GB	0MB	8	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris kernel zone - medium	8	8192MB	40GB	0GB	0MB	3	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris non-global zone - large	16	8192MB	40GB	0GB	0MB	9	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris kernel zone - large	16	16384MB	40GB	0GB	0MB	4	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris non-global zone - xlarge	32	16384MB	80GB	0GB	0MB	10	Yes	Yes	Edit Flavor ▾
<input type="checkbox"/>	Oracle Solaris kernel zone - xlarge	32	32768MB	80GB	0GB	0MB	5	Yes	Yes	Edit Flavor ▾

Displaying 10 items

Para obtener información sobre estas columnas, consulte la [Referencia de la interfaz de línea de comandos de OpenStack](#).

Modificación de las especificaciones de tipos

En la columna Acciones de cada tipo, al hacer clic en una de las primeras tres opciones se muestran las propiedades del tipo y también le permite modificar las propiedades. Hay tres acciones disponibles:

- `Edit Flavor` muestra información acerca del tipo y las propiedades que se pueden modificar. Con el separador Acceso a tipo, puede restringir los proyectos que pueden acceder al tipo. La configuración por defecto es ninguno, lo que significa que el tipo es público y todos los proyectos pueden acceder a él.
- `Modify Access` abre directamente el separador Acceso a tipo para que pueda modificar la configuración de acceso.
- `Update Metadata` le permite modificar los metadatos del tipo.

No todas las modificaciones de tipo se pueden realizar en el panel de control. Por ejemplo, las claves de la propiedad `extra_specs` solo se pueden revisar con la línea de comandos. Las claves de la propiedad refieren al conjunto de propiedades de zona que generalmente se configura con el comando `zonecfg` y que se admite en OpenStack.

Se admiten las siguientes claves en tipos de zonas de núcleo y zonas no globales:

- `zonecfg:bootargs`
- `zonecfg:brand`
- `zonecfg:hostid`
- `zonecfg:cpu-arch`

Las siguientes claves se admiten únicamente en tipos de zonas no globales:

- `zonecfg:file-mac-profile`
- `zonecfg:fs-allowed`
- `zonecfg:limitpriv`

Para obtener una descripción de estas propiedades de configuración de zona, consulte la página del comando `man zonecfg(1M)`.

Nota - No se admiten todas las propiedades de configuración de zona en OpenStack.

La clave `sc_profile` también se puede modificar solo desde la línea de comandos. Esta clave se usa para especificar un perfil de configuración de sistema para el tipo.

Para modificar un tipo desde la línea de comandos, utilice la sintaxis siguiente:

```
nova flavor-key flavor action key=value [key=value ...]
```

flavor El nombre o el ID del tipo.

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

```
controller# nova flavor-key 8 set zonecfg:bootargs=-v
controller# nova flavor-show 8
```

Property	Value
OS-FLV-DISABLED:disabled	False
OS-FLV-EXT-DATA:ephemeral	0
disk	40
extra_specs	{"zonecfg:brand": "solaris"} u'zonecfg:bootargs': u'-v'}
<i>modificados</i>	
id	8
name	Oracle Solaris non-global zone - medium
os-flavor-access:is_public	True
ram	4096
rxtx_factor	1.0
swap	
vcpus	8

bootargs

Gestión de instancias de VM

Esta sección describe cómo modificar las instancias de VM que creó en la nube. Se incluyen los siguientes temas:

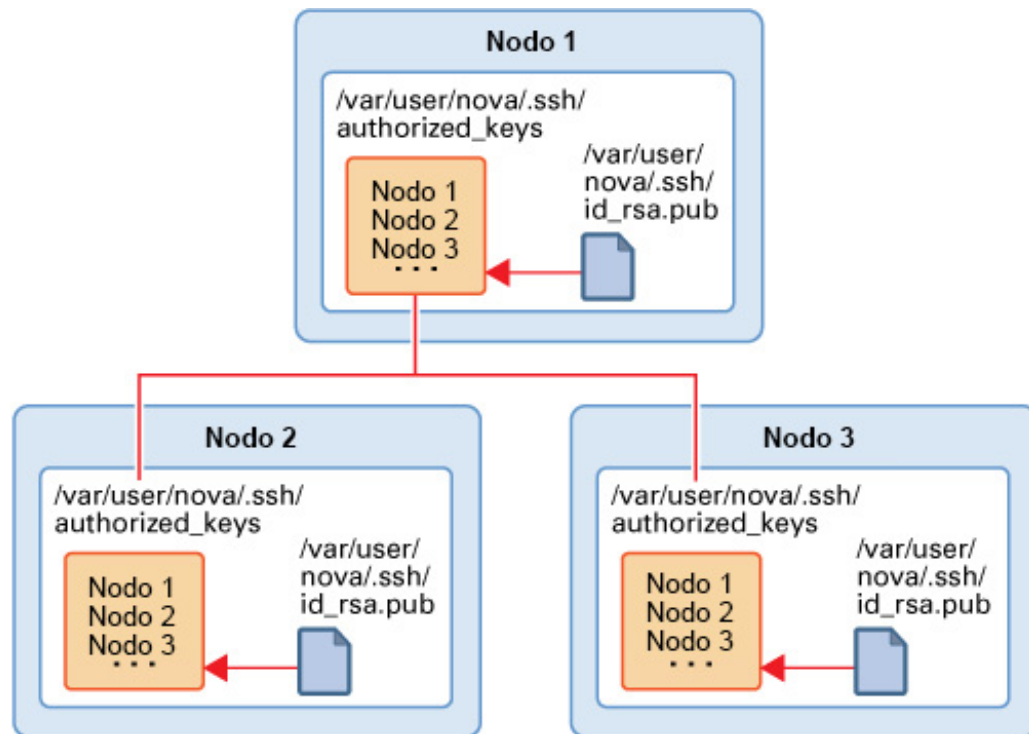
- [“Migración y evacuación de instancias de VM” \[84\]](#).
- [“Cambio de tamaño de una instancia de VM” \[86\]](#).

Migración y evacuación de instancias de VM

En Oracle OpenStack for Oracle Solaris, las instancias de VM son zonas de núcleo o no globales, que son entornos virtuales de alta densidad escalables de Oracle Solaris.

Para obtener más información sobre la migración en directo de zonas, consulte *Creación y uso de zonas del núcleo de Oracle Solaris* para conocer más sobre la migración de zonas de núcleo. Este manual se encuentra en la biblioteca de la versión de Oracle Solaris en [Documentación de sistemas operativos](#). Consulte también las páginas del comando `man zoneadm(1M)` y `solaris-kz(5)`.

Para que la migración de nodos se realice correctamente, debe asegurarse de que la clave SSH de cada nodo de cálculo se agregue al archivo de claves autorizadas de cada nodo. Por lo tanto, los nodos comparten archivos idénticos de claves autorizadas, como se ilustra en la siguiente figura:



Prepare la migración de instancia de VM completando los siguientes pasos:

1. En cada nodo, cree una clave SSH.

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

2. Lleve todos los archivos de claves de los diferentes nodos a una ubicación común en uno de los nodos.
3. Combine todas las claves en un archivo `authorized_keys`.

Por ejemplo:

```
# cat nova(1)/id_rsa.pub nova(n)/id_rsa.pub >> /var/user/nova/.ssh/authorized_keys
```

donde *nova(1)* a *nova(n)* representa las claves SSH de los nodos participantes.

4. Distribuya el archivo `authorized_keys` a los directorios `/var/user/nova/.ssh` de todos los nodos que participan.
5. Opcionalmente, especifique el cifrado que se debe usar durante la migración en el parámetro `live_migration_cipher` en el archivo `/etc/nova/nova.conf` de cada nodo de cálculo.

Sin embargo, si prefiere que el proceso seleccione automáticamente un cifrado adecuado, deje el parámetro sin definir.

Para migrar una instancia, use la siguiente sintaxis después de establecer las variables de shell globales:

Para realizar una migración en directo de un servidor en funcionamiento a una máquina distinta, utilice la siguiente sintaxis después de establecer las variables de shell globales:

```
# nova live-migration server [host]
```

donde *server* puede ser tanto el nombre del servidor como el ID del servidor, y el *host* opcional es el nombre del servidor de destino.

Si falla el nodo de la instancia actual o si el servicio Nova en sí mismo se desactiva por un período, puede mover la instancia y volver a crearla en otro nodo utilizando el comando `nova evacuate`. Por lo tanto, puede recuperar el nodo.

Tenga en cuenta que solo puede evacuar zonas de núcleo. Se admite la evacuación en configuraciones en las que los dispositivos raíz están en almacenamiento compartido.

Para realizar una migración en directo de todas las instancias de VM de un host a otro, utilice la siguiente sintaxis después de establecer las variables de shell globales.

```
# nova host-evacuate-live [--target-host target] server
```

Cambio de tamaño de una instancia de VM

El tamaño de una VM se indica en el tipo en cuya base se inicia la VM. Consulte [Cómo crear una instancia de VM \[76\]](#) para conocer los pasos para crear una instancia de VM. En la siguiente figura, se muestran detalles sobre un ejemplo de VM `hr_vm1` como se muestra en un panel de control de Horizon.

FIGURA 6 Tamaño de instancia de VM

Instances

Instances

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Size
<input type="checkbox"/>	hr_vm1	Solaris Non-global Zone	10.132.20.5 10.132.10.10	Oracle Solaris non-global zone - tiny

Displaying 1 item

La figura muestra que el tamaño de hr_vm1 es el tamaño definido para el tipo Oracle Solaris non-global zone - tiny. Haciendo clic en el nombre de la instancia, puede ver los detalles del tipo específico:

- RAM = 2 GB
- VCPU = 1 VCPU
- Disco = 10 GB

Cambiar el tamaño de una instancia significa usar un tipo diferente para la instancia.

Por defecto, el proceso de cambio de tamaño crea una instancia con un tamaño nuevo en un nodo nuevo. Sin embargo, si está cambiando el tamaño de un tipo de zona no global, debe configurar el proceso de cambio de tamaño para que se complete en el mismo nodo. Se requiere que las zonas no globales sean de la misma versión que sus zonas globales originales. Cambiar de tamaño a un tipo de zona no global y tener creada la instancia en un nodo diferente genera el riesgo de poner la instancia en un nodo que tenga una versión de zona global diferente. Para evitar el riesgo y crear la instancia con el cambio de tamaño en el mismo host, establezca `allow_resize_to_same_host` en `true` en el archivo `/etc/nova/nova.conf`.

Sin embargo, las zonas de núcleo no tienen esta limitación de las zonas no globales. Por lo tanto, puede cambiar el tamaño de una zona de núcleo en otro nodo.

▼ Cómo cambiar de tamaño una instancia de VM

Antes de empezar Debe saber el tamaño actual de la instancia de VM que desea modificar. Puede obtener esta información desde el panel de control. Para obtener un ejemplo, consulte [Figura 6, “Tamaño de instancia de VM”](#).

1. Establezca las variables de shell global.

```
# export OS_USERNAME=nova
# export OS_PASSWORD=service-password
# export OS_TENANT_NAME=service
# export OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE:5000/v2.0"
```

2. Si está realizando un cambio de tamaño a un tipo de zona no global, edite el parámetro que se muestra a continuación en el archivo `/etc/nova/nova.conf` de la siguiente manera:

```
allow_resize_to_same_host=true
```

3. Visualice la lista de tipos.

```
# nova flavor-list
```

4. Anote el número de ID del tipo nuevo que desea que use la instancia de VM.

5. Cambie la instancia de VM para usar el tipo nuevo haciendo referencia al número de ID del tipo.

```
# nova resize instance-name flavor-id
```

Por ejemplo, imagine que actualmente el tamaño de la instancia `hr_vm1` está establecido en el tipo `Oracle Solaris non-global zone - tiny`, cuyo número de ID es 6. Desea cambiar el tamaño de la instancia por el tipo `Oracle Solaris non-global zone - medium`, cuyo número de ID es 8. Debe escribir lo siguiente:

```
# nova resize --poll hr_vm1 8
```

La opción `--poll` le permite ver el progreso del proceso de cambio de tamaño hasta su finalización.

6. (Opcional) Verifique en el panel de control que el tamaño de la instancia ha cambiado por el tipo nuevo.

También puede realizar la misma verificación usando el siguiente comando:

```
# nova show instance-name
```

Compruebe que el valor del campo `flavor` especifique el tipo nuevo en el paso anterior.

A continuación, se presenta un ejemplo de una salida extraída. Esta instancia `hr_vm1` se crea en un proyecto llamado `TenantA` y se le ha cambiado el tamaño a `Oracle Solaris non-global zone - medium`.

```
# export OS_USERNAME=admin
# export OS_PASSWORD=admin-password
# export OS_TENANT_NAME=TenantA
# export OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE:5000/v2.0"
```



```
# nova show hr_vm1
+-----+-----+
| Property      | Value                                     |
+-----+-----+
| ...           |                                           |
| created       | 2016-01-26T12:38:47Z                     |
| flavor        | Oracle Solaris non-global zone - medium (8) |
| ...           |                                           |
+-----+-----+
```


Opciones para la configuración y el despliegue de Cinder

En este capítulo, se describen las maneras alternativas en las que puede configurar Cinder y el componente de almacenamiento de su configuración de OpenStack.

Nota - Los procedimientos que se describen en [“Configuración del nodo de almacenamiento” \[49\]](#) para configurar el almacenamiento de bloques sigue siendo el método por defecto para configurar el nodo de almacenamiento. Las opciones que se ofrecen en este capítulo son maneras alternativas de desplegar el componente de almacenamiento. Sin embargo, actualmente no se recomienda usar la configuración de almacenamiento por defecto y la configuración alternativa simultáneamente para configurar Cinder en la estructura de nube.

En este capítulo, se tratan los siguientes temas:

- [“Despliegue de sistemas remotos para almacenamiento” \[91\]](#)
- [“Uso de OpenStack con Oracle ZFS Storage Appliance” \[96\]](#)

Despliegue de sistemas remotos para almacenamiento

El despliegue de Cinder que se describe en esta sección es aplicable a configuraciones de Juno que ejecuten Oracle Solaris 11.3 SRU 2 o versiones posteriores.

En versiones anteriores de OpenStack sin compatibilidad con SAN, había que configurar el servicio de volúmenes de Cinder para que se ejecutara en el host de destino si se usaba el controlador iSCSI de ZFS.

Con la compatibilidad de redes de área de almacenamiento (SAN) en Cinder, puede usar SSH para configurar varios backends de hosts de almacenamiento y definir el tipo de volumen de estos hosts para determinar cómo se los usa. En este tipo de despliegues, tanto el paquete de Cinder como todos los servicios de Cinder se instalan solo en el host iniciador, que normalmente es el host que también funciona como nodo de cálculo.

No es necesario instalar el paquete de OpenStack en los hosts de destino remotos. Estos hosts simplemente proporcionan los discos LUN al host iniciador basado en COMSTAR.

El despliegue correcto de sistemas remotos para almacenamiento tiene los siguientes requisitos:

- Configuración del archivo `/etc/cinder/cinder.conf`.
- Otorgamiento del perfil de derechos adecuado al usuario designado.
- Instalación manual de paquetes adicionales.

En las siguientes secciones se describen estos requisitos en detalle.

Configuración del archivo `cinder.conf`

Después de instalar el paquete de Cinder en el host iniciador, edite la sección [DEFAULT] del archivo `/etc/cinder/cinder.conf`.

Consulte el siguiente ejemplo como guía para comprender las definiciones del archivo de configuración.

```
[DEFAULT]
my_ip = localhost
osapi_volume_workers = 1
auth_strategy = keystone
#num_volume_device_scan_tries = 6
os-volume-api-version = 2
scheduler_driver=cinder.scheduler.filter_scheduler.FilterScheduler

enabled_backends=zfsdriver-1, zfsdriver-2, zfsdriver-3, zfsdriver-4, zfsdriver-5

[zfsdriver-1]
volume_group=zvolumes-1
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSVolumeDriver
volume_backend_name=ZFS_LOCAL
zfs_volume_base = rpool/cinder
san_is_local = True
debug=true
verbose=true
[zfsdriver-2]
volume_group=zvolumes-2
volume_driver=cinder.volume.drivers.solaris.zfs.ZFISCSIDriver
volume_backend_name=ZFS_REMOTE
zfs_volume_base = rpool/cinder
san_ip = 10.134.13.38
san_login = jdoe
san_password = password
debug=true
verbose=true
[zfsdriver-3]
volume_group=zvolumes-3
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSFCDriver
volume_backend_name=ZFS_REMOTE_FC
zfs_volume_base = rpool/fc
san_ip = 10.134.13.38
```

```

san_login = jdoe
san_password = password
debug=true
verbose=true
[zfsdriver-4]
volume_group=zvolumes-4
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSISCSIDriver
volume_backend_name=ZFS_REMOTE
zfs_volume_base = rpool/cinder/zq
san_ip = 10.134.13.38
san_login = jdoe
san_password = password
debug=true
verbose=true
[zfsdriver-5]
volume_group=zvolumes-5
volume_driver=cinder.volume.drivers.solaris.zfs.ZFSISCSIDriver
volume_backend_name=ZFS_REMOTE
zfs_volume_base = rpool/zq
san_ip = 10.134.63.182
san_login = jdoe
san_password = password
debug=true
verbose=true

```

enabled_backends= Muestra los hosts de backend activados. En este ejemplo, hay 5 hosts de backend definidos.

zfsdriver-1,
zfsdriver-2,
zfsdriver-3,
zfsdriver-4,
zfsdriver-5

Los hosts de backend usan 3 controladores ZFS (ZFSVolumeDriver, ZFSISCSIDriver y ZFSFCDriver) en 3 hosts. De estos hosts, uno es local (localhost) y los otros son remotos (10.134.13.38 y 10.134.63.182).

volume_backend_name= Define un tipo de volumen con el nombre especificado. Los siguientes comandos de ejemplo realizan estas acciones, respectivamente:

ZFS_REMOTE

```

# cinder type-create type-remote
# cinder type-key type-remote set volume_backend_name=ZFS_REMOTE
# cinder create --display-name t1 --volume-type type-remote

```

En el ejemplo anterior, cada comando realiza las siguientes acciones:

- Crea un nuevo tipo de volumen.
- Asigna el nombre de backend al nuevo tipo de volumen.
- Crea un nuevo volumen según el nuevo tipo de volumen.

Una vez que se ejecuta el último comando, se crea el nuevo volumen `t1` en uno de los backends con el nombre `ZFS_REMOTE` según las reglas de filtros.

zfs_volume_base = rpool/zq

Especifica el juego de datos base para los nuevos volúmenes de ZFS en cada backend de volumen.

`san_is_local` Como todos los controladores de ZFS se basan en el controlador de SAN, debe configurar los parámetros `san_*`.
`san_ip san_login`
`san_password`

Los comandos se ejecutan en un host de backend, ya sea de manera local o remota con SSH. De esta manera, para cada backend, debe configurar el parámetro `san_is_local = True` o el parámetro `san_ip*`, pero no ambos.

`debug=true` Parámetros opcionales para depuración. Puede omitir la configuración de estos parámetros.
`verbose=true`

Después de haber configurado Cinder correctamente, al mostrar estos servicios se despliega el estado de cada servicio.

```
# cinder service-list
+-----+-----+-----+-----+-----+-----+-----+
| Binary | Host | Zone | Status | State | Updated_at | Disabled Reason |
+-----+-----+-----+-----+-----+-----+-----+
| cinder-backup | host-2 | nova | enabled | up | 2015-10-13 T19:22:45.000000 | None |
| cinder-scheduler | host-2 | nova | enabled | up | 2015-10-13 T19:22:43.000000 | None |
| cinder-volume | host-2 | nova | enabled | down | 2015-10-13 T18:31:41.000000 | None |
| cinder-volume | @zfsdriver-1 | nova | enabled | up | 2015-10-13 T19:22:46.000000 | None |
| cinder-volume | host-2 | nova | enabled | up | 2015-10-13 T19:22:47.000000 | None |
| cinder-volume | @zfsdriver-2 | nova | enabled | up | 2015-10-13 T19:22:47.000000 | None |
| cinder-volume | host-2 | nova | enabled | up | 2015-10-13 T19:22:48.000000 | None |
| cinder-volume | @zfsdriver-3 | nova | enabled | up | 2015-10-13 T19:22:48.000000 | None |
| cinder-volume | host-2 | nova | enabled | up | 2015-10-13 T19:22:47.000000 | None |
| cinder-volume | @zfsdriver-4 | nova | enabled | up | 2015-10-13 T19:22:47.000000 | None |
| cinder-volume | host-2 | nova | enabled | up | 2015-10-13 T19:22:48.000000 | None |
| cinder-volume | @zfsdriver-5 | nova | enabled | up | 2015-10-13 T19:22:48.000000 | None |
| cinder-volume | host-2 | nova | enabled | down | 2015-10-13 T18:32:55.000000 | None |
| cinder-volume | @zfsdriver-6 | nova | enabled | down | 2015-10-13 T18:32:55.000000 | None |
+-----+-----+-----+-----+-----+-----+-----+
```

Otorgamiento de derechos al usuario designado

Para permitir que el usuario definido mediante `san_login` use los destinos remotos, se le debe otorgar el perfil de derechos adecuado. En el siguiente ejemplo, se muestra cómo crear el perfil de derechos para el usuario.

```
# useradd -s /usr/bin/pfbash -m jdoe
# passwd jdoe password
# profiles -p "Cinder Storage management"
profiles:Cinder Storage management> set desc="Cinder Storage management on target host"
profiles:Cinder Storage management> add profiles="File System Management"
profiles:Cinder Storage management> add auths="solaris.smf.modify.stmf"
profiles:Cinder Storage management> add cmd=/usr/sbin/itadm
profiles:Cinder Storage management:itadm> set euid=0
profiles:Cinder Storage management:itadm> end
profiles:Cinder Storage management> add cmd=/usr/sbin/stmfadm
profiles:Cinder Storage management:stmfadm> set euid=0
profiles:Cinder Storage management:stmfadm> end
profiles:Cinder Storage management> add cmd=/usr/sbin/zfs
profiles:Cinder Storage management:zfs> set priv=sys_config,sys_mount
profiles:Cinder Storage management:zfs> end
profiles:Cinder Storage management> exit

# usermod -P "Cinder Storage management" jdoe
```

Para obtener más información sobre perfiles y derechos, consulte [Protección de los usuarios y los procesos en Oracle Solaris 11.3](#).

Activación del host remoto como destino

Para activar un host remoto como destino, siga estos pasos en el host remoto:

1. Instale el paquete `group/feature/storage-server` en el host remoto.

```
# pkg install storage-server
```

2. Active los siguientes servicios en el host:
 - `svc:/system/stmf:default`
 - `svc:/network/iscsi/target:default`

Por ejemplo:

```
remote-host# svcadm enable stmf
```

```
remote-host# svcadm enable -r svc:/network/iscsi/target:default
```

3. Inicialice y configure la lista de control de acceso (ACL) para el `zfs_volume_base` definido.

Por ejemplo, supongamos que tiene la definición `zfs_volume_base=rpool/fc` en el archivo de configuración. A continuación, debe ejecutar los siguientes comandos:

```
# chmod A+user:cinder:add_subdirectory:allow /rpool/fc
# zfs allow cinder clone,create,destroy,mount,snapshot rpool/fc
```

Uso de OpenStack con Oracle ZFS Storage Appliance

En esta sección, se describe una configuración de OpenStack que usa Oracle ZFS Storage Appliance (ZFSSA) como almacenamiento de backend.

Acerca de Oracle ZFS Storage Appliance

La familia de productos Oracle ZFS Storage Appliance presta servicios eficaces de datos de bloque y archivos para clientes de red, además de un conjunto de servicios de datos que se puede aplicar a los datos almacenados en el sistema que incluyen las siguientes tecnologías:

- **Analytics (Análisis):** le permite observar de forma dinámica el comportamiento del sistema en tiempo real y visualizar los datos gráficamente.
- **La agrupación de almacenamiento híbrido de ZFS:** le permite gestionar una serie de dispositivos, como dispositivos opcionales de memoria Flash, discos de baja potencia y alta capacidad, y memoria DRAM como única jerarquía de datos
- **Compatibilidad con una variedad de hardware**

El paquete `cloud/openstack/cinder` de Oracle OpenStack for Oracle Solaris incluye el controlador Oracle ZFSSA iSCSI Cinder. El controlador activa el dispositivo para uso ininterrumpido como recurso de almacenamiento de bloques para el componente Cinder. En concreto, el controlador le permite crear volúmenes iSCSI que un servidor de Cinder puede asignar a cualquier instancia de máquina virtual creada por el servicio de Nova. Para usar Oracle ZFSSA para almacenamiento, asegúrese primero de que el dispositivo tenga al menos la versión 2013.1.2.0 del software ZFSSA.

Configuración de OpenStack con Oracle ZFSSA

Esta sección no cubre en detalle los procedimientos para configurar Oracle ZFSSA. Para obtener más información acerca del dispositivo, incluidos pasos de configuración, consulte las siguientes fuentes:

- Documentación del producto Oracle ZFSSA en la [página de documentación de almacenamiento del centro de ayuda de Oracle](#)

- [Uso de Oracle ZFS Storage Appliance como backend de almacenamiento para OpenStack Cinder](http://www.oracle.com/technetwork/server-storage/sun-unified-storage/documentation/openstack-cinder-zfssa-120915-2813178.pdf) (<http://www.oracle.com/technetwork/server-storage/sun-unified-storage/documentation/openstack-cinder-zfssa-120915-2813178.pdf>). Esta documentación técnica detalla cómo configurar el dispositivo de almacenamiento para usarlo en la nube.

Antes de los procedimientos para configurar OpenStack con Oracle ZFSSA, asegúrese de haber creado la subred que alojará el tráfico de almacenamiento de datos de host. El escenario en este capítulo usa [Figura 2, “Arquitectura de referencia de configuración de tres nodos”](#) como base para este punto de partida pero con una diferencia. Aquí, el tráfico de almacenamiento de datos se aloja en dos subredes, con net2 y net3 como interfaces para ese tráfico. Las dos subredes garantizan un rendimiento eficiente y disponibilidad del servicio de almacenamiento de datos.

En relación con ZFSSA

Oracle ZFSSA usa iSCSI para intercambiar comandos SCSI mediante la red. Para configurar la comunicación entre nodos iSCSI, debe tener la siguiente información disponible:

- Dirección IP o nombre DNS del nodo SCSI. El nodo SCSI consta de los nodos de iniciador y de destino.
- Puerto de Internet TCP iSCSI en el nodo de destino. Por defecto, el número de puerto es 3260.
- Nombres completos de iSCSI (IQN) de los nodos de iniciador y destino.
- Información de autenticación opcional con protocolo de autenticación por desafío mutuo (autenticación CHAP).

Para obtener información relevante de iSCSI para este paso preparatorio, use el comando `iscsiadm` de Oracle Solaris.

Este documento no describe los pasos para configurar el dispositivo. Para obtener información sobre los procedimientos para preparar el dispositivo para su uso operativo, consulte la documentación de la versión en la [Documentación de almacenamiento](#) de Oracle.

En relación con OpenStack

El controlador OpenStack ZFSSA de Cinder crea y gestiona los volúmenes creando los LUN relacionados en el dispositivo, configurando las propiedades adecuadas de estos LUN y gestionando la visibilidad de los LUN para los nodos de cálculo e instancias de invitado de OpenStack en los que los volúmenes se están usando.

Después de configurar Oracle ZFSSA, puede configurar el controlador ZFSSA iSCSI de Cinder.

▼ Cómo configurar Oracle ZFSSA for OpenStack

El procedimiento utiliza el flujo de trabajo `cinder.akwf`, que 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.

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` utilizando uno de los siguientes métodos:

- **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-cinder.akwf-file"
    url = "url-to-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
zfssa:maintenance workflow-001 execute (uncommitted)>

zfssa:maintenance workflow-001 execute (uncommitted)> set name=user
    name = user
zfssa:maintenance workflow-001 execute (uncommitted)> set password=password
    password = password
zfssa:maintenance workflow-001 execute (uncommitted)> commit
```

User openstack created.

Tenga en cuenta que para *usuario* y *contraseña*, los valores son los que se definen para los parámetros `san_login` y `san_password` en el archivo `cinder.conf`.

- **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.ZFSSAISCSIDriver` 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	ADDRS	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-fqdn/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
```

Cómo trabajar con el componente Heat

En este capítulo, se describe el componente Heat según se encuentra implementado y según se admite en Oracle Solaris. Se incluyen los siguientes temas:

- “Acerca del componente Heat” [101]
- “Instalación de Heat” [102]
- “Acerca de las plantillas HOT” [102]

Acerca del componente Heat

Heat es el motor de orquestación de OpenStack que le permite implementar las aplicaciones basadas en las plantillas de orquestación de Heat que cree. Estas plantillas también se conocen como plantillas HOT.

Con una plantilla HOT, puede crear distintos tipos de recursos de OpenStack, como instancias, direcciones IP flotantes, volúmenes, usuarios, etc. Las plantillas también le permiten implementar funcionalidades avanzadas, como alta disponibilidad de instancias, autoescalamiento de instancias y pilas anidadas. Por lo tanto, con Heat, todos los proyectos principales de OpenStack pueden recibir una base de usuarios más grande. Mediante las plantillas, Heat le proporciona una manera de sustituir la implementación por defecto de recursos mediante el uso de parámetros que se pasan de la plantilla al motor.

Heat presta sus servicios mediante una API de servicios web RESTful. Al igual que con todas las aplicaciones OpenStack, se utilizan las interfaces WSGI de Python, y las aplicaciones se configuran juntas usando Paste. Los puntos finales HTTP de la aplicación están formados por pipelines de middleware de la interfaz de puerta de enlace de servidor web (WSGI). Heat utiliza específicamente dos puntos finales: el puerto 8004 para la API de Heat y el puerto 8000 para Heat CloudFormation.

El archivo de configuración `/etc/heat/heat.conf` controla la configuración del propio Heat. En este momento, no existen parámetros de configuración específicos de Solaris en el archivo de configuración principal de Heat.

Para obtener más información sobre el componente Heat, consulte la [documentación de Heat](#) en la comunidad de OpenStack.

Instalación de Heat

En una configuración típica, los servicios Heat se deben instalar en el mismo nodo que Keystone. El paquete Heat se incluye automáticamente si instaló OpenStack en el nodo con el siguiente comando:

```
# pkg install openstack
```

▼ 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. **Configure Heat eliminando los comentarios de los parámetros o estableciendo dichos parámetros en el archivo `/etc/heat/heat.conf`:**

```
[database]
connection = mysql://heat:heat-service-password@$CONTROLLER_ADMIN_NODE/heat

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

2. **Active el servicio de Heat.**

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

Acerca de las plantillas HOT

Para usar Heat para orquestar varias aplicaciones compuestas en la nube en la configuración de OpenStack, debe definir una plantilla de orquestación de Heat (HOT). Una plantilla HOT contiene especificaciones que debe completar. Los parámetros que proporciona se leen cuando ejecuta el proceso para crear tipos de recursos y otras funcionalidades avanzadas.

Para conocer las especificaciones de la plantilla HOT y sus descripciones, consulte http://docs.openstack.org/developer/heat/template_guide/hot_spec.html#hot-spec.

Para obtener información sobre cómo escribir una plantilla HOT, consulte http://docs.openstack.org/developer/heat/template_guide/hot_guide.html#hot-guide.

Para implementar una plantilla HOT, use el siguiente comando:

```
# heat stack-create -f template
```

Donde *template* incluye la ruta completa de la ubicación de la plantilla.

El siguiente ejemplo muestra el contenido de una plantilla HOT denominada *stack3* que crea tres redes privadas con subredes.

Nota - La línea `heat_template_version: 2013-05-23` debe aparecer en la parte superior de la plantilla.

```
heat_template_version: 2013-05-23
description: Create a few networks.

resources:
  heat_net1:
    type:OS::Neutron::Net
    properties:
      name:heat_net1

  heat_subnet1:
    type:OS::Neutron::Subnet
    properties:
      name:heat_subnet1
      network_id: { get_resource: heat_net1 }
      cidr: 192.168.50.0/24

  heat_net2:
    type:OS::Neutron::Net
    properties:
      name: heat_net2

  heat_subnet2:
    type:OS::Neutron::Subnet
    properties:
      name:heat_subnet2
      network_id: { get_resource: heat_net2 }
      cidr: 192.168.51.0/24

  heat_net3:
    type:OS::Neutron::Net
    properties:
      name: heat_net3

  heat_subnet3:
    type:OS::Neutron::Subnet
    properties:
      name:heat_subnet3
      network_id: { get_resource: heat_net3 }
      cidr: 192.168.52.0/24
```


Cómo trabajar con el componente Ironic

En este capítulo, se describe Ironic según se encuentra implementado y según se admite en Oracle Solaris. Se incluyen los siguientes temas:

- [“Acerca del componente Ironic” \[105\]](#)
- [“Instalación y configuración de Ironic” \[106\]](#)
- [“Visión general: Implementación de reconstrucción completa con Ironic” \[111\]](#)
- [“Uso de Ironic para la implementación de reconstrucción completa” \[113\]](#)

Acerca del componente Ironic

En los capítulos anteriores se describían los componentes principales de OpenStack para crear una nube. Los componentes adicionales ofrecen otros servicios que están relacionados con la administración de la nube. En este capítulo, se trata el componente Ironic que está disponible en la versión Juno.

Si bien los componentes principales de OpenStack le permiten aprovisionar máquinas virtuales o instancias de VM, Ironic presta servicios para registrar y aprovisionar instancias o nodos de reconstrucción completa e interrumpir la comunicación de estos. Ironic usa tecnologías comunes, como el inicio PXE e IPMI para admitir una gran variedad de hardware que puede aprovisionar. Además, un mecanismo de controlador conectable le permite a Ironic gestionar y admitir hardware específico del proveedor.

Para obtener más información sobre Ironic y las ventajas que ofrece, consulte la [documentación del desarrollador](#) de Ironic en el sitio web de la comunidad de OpenStack.

Ironic consta de tres componentes principales. En Oracle Solaris, estos componentes se suministran como servicios SMF. En la siguiente tabla, se muestran y se describen estos componentes.

Componente	Descripción	Servicio SMF
Servicio API de OpenStack Ironic	Servicio que proporciona una API RESTful mediante la cual los operadores y otros	svc:/application/openstack/ironic/ironic-api

Componente	Descripción	Servicio SMF
	servicios pueden interactuar con los nodos de reconstrucción completa gestionados.	
Servicio Conductor de OpenStack Ironic	Controlador principal que realiza el aprovisionamiento real de los nodos de reconstrucción completa mediante el uso de los controladores de referencia y específicos del proveedor. Los servicios API y conductor se comunican mediante RPC.	svc:/application/openstack/ironic/ironic-conductor
Servicio de base de datos de OpenStack Ironic	Un servicio SMF transitorio para crear y sincronizar la base de datos de backend de Ironic.	svc:/application/openstack/ironic/ironic-db

Instalación y configuración de Ironic

Puede usar Ironic como componente independiente sin otros servicios de OpenStack. O bien, puede implementarlo junto con otros componentes de OpenStack, generalmente, en el nodo de cálculo. La configuración depende de cómo implementa Ironic.

El aprovisionamiento de instancias de reconstrucción completa de Oracle Solaris requiere Automated Installer (AI). El AI se puede ubicar en el mismo nodo que el servicio Ironic. De manera alternativa, puede usar AI en un servidor remoto para operaciones con Ironic.

Para obtener más información sobre cómo configurar y usar AI, consulte los manuales de instalación en la biblioteca correspondiente en [Documentación de sistemas operativos](#).

▼ Cómo instalar y configurar Ironic

Antes de empezar Si está implementando Ironic junto con el resto de los componentes de OpenStack, asegúrese de configurar los componentes principales en primer lugar. Como mínimo, las configuraciones de Keystone y Glance se deben finalizar antes de trabajar con Ironic.

De manera similar, asegúrese de que la base de datos para Ironic esté configurada. Este procedimiento sigue la arquitectura de referencia de tres nodos que se muestra en [Figura 2, “Arquitectura de referencia de configuración de tres nodos”](#) donde la base de datos se configura en el nodo de controlador, junto con el resto de las bases de datos de OpenStack.

1. **Cree la base de datos de Ironic.**
 - a. **Establezca la variable de shell global para el nodo de administrador de controlador.**

```
controller# export $CONTROLLER_ADMIN_NODE=controller-node
```

donde *controller-node* puede ser la dirección IP o el nombre de host del controlador. Para obtener más información acerca de la configuración de variables, consulte [“Preparación de nombres de host, variables y contraseñas” \[29\]](#).

b. Cree la base de datos con los siguientes comandos:

```
controller# mysql -u root -p
Enter password: MySQL-root-password
mysql> create database ironic default character set utf8 default collate
utf8_general_ci;
mysql> grant all privileges on ironic.* to 'ironic'@'$CONTROLLER_ADMIN_NODE'
identified by 'service-password';
mysql> flush privileges;
mysql> quit
```

2. Instale el paquete de Ironic de OpenStack.

```
# pkg install ironic ironicclient rabbitmq
# pkg update stevedore stevedore-27
```

3. Instale las herramientas de administración y el servidor de AI.

Use el mismo comando, ya sea que esté instalando el paquete del servidor de AI de manera local con Ironic o de manera remota en otro host.

```
# pkg install pkg:/install/installadm
```

4. Si AI se ubica de manera remota, configure el usuario de Ironic en ese servidor.

```
remote-AI# useradd -d /var/lib/ironic -m -g 88 -u 88 \
-P "Install Service Management" ironic
remote-AI# password ironic
New Password: password
Re-enter new Password: password
```

Nota - Si AI se instala de manera local con Ironic, el proceso de instalación de Ironic crea automáticamente el usuario *ironic* en el sistema.

5. Cree y administre la clave SSH para el usuario de Ironic.

- Si el AI está instalado de manera local con Ironic, ejecute los siguientes comandos en ese sistema:

```
ironic-localhost# su - ironic
ironic-localhost# mkdir /var/lib/ironic/.ssh
ironic-localhost# ssh-keygen -N '' -t rsa \
-f /var/lib/ironic/.ssh/id_rsa
ironic-localhost# cat /var/lib/ironic/.ssh/id_rsa.pub > \
```

```
/var/lib/ironic/.ssh/authorized_keys
```

■ **Si AI está instalado de manera remota, lleve a cabo los siguientes pasos:**

a. **En el AI remoto, ejecute los siguientes comandos:**

```
remote-AI# su - ironic
remote-AI# mkdir /var/lib/ironic/.ssh
remote-AI# ssh-keygen -N '' -t rsa \
-f /var/lib/ironic/.ssh/id_rsa
remote-AI# cat /var/lib/ironic/.ssh/id_rsa.pub > \
/var/lib/ironic/.ssh/authorized_keys
```

b. **En el host de Ironic, ejecute los siguientes comandos:**

```
ironic-localhost# mkdir /var/lib/ironic/.ssh
ironic-localhost# scp ironic@AI-server:~/.ssh/id_rsa /var/lib/ironic/.ssh
ironic-localhost# scp ironic@AI-server:~/.ssh/id_rsa.pub /var/lib/ironic/.ssh
ironic-localhost# cat /var/lib/ironic/.ssh/id_rsa.pub > \
/var/lib/ironic/.ssh/authorized_keys
ironic-localhost# chown -R ironic:ironic /var/lib/ironic/.ssh
```

Donde *AI-server* puede ser la dirección IP o el nombre de host del servidor de AI.

6. **Edite el archivo `/etc/ironic/ironic.conf`.**

Tenga en cuenta las siguientes consideraciones relacionadas con la configuración de ciertos parámetros en el archivo:

<code>auth_strategy</code>	En la sección <code>DEFAULT</code> , especifique <code>noauth</code> si usa Ironic como componente independiente. Especifique <code>keystone</code> si está implementado Ironic con otros componentes de OpenStack.
<code>server</code>	En la sección <code>ai</code> , especifique <code>localhost</code> si Automated Installer (AI) está instalado de manera local con Ironic. Si AI está instalado de manera remota, especifique la dirección IP o el nombre de host del servidor correspondiente.
<code>connection</code>	En la sección <code>[database]</code> , especifique la conexión con la base de datos MySQL.
<code>glance_host</code>	En la sección <code>glance</code> , especifique la dirección IP o el nombre de host del servidor Glance si el parámetro <code>auth_strategy</code> está establecido en <code>keystone</code> .
<code>glance_api_servers</code>	Además, en la sección <code>glance</code> , especifique los nombres de host o las direcciones IP con sus correspondientes números de puerto, como <code>192.168.0.150:9292</code> . No especifique <code>localhost</code> .

Use el siguiente archivo de ejemplo como guía para configurar todos los parámetros requeridos para que funcione Ironic. Tenga en cuenta que en el archivo de ejemplo, la variable *glance-serverIP* representa la dirección IP o el nombre de host del servidor Glance.

```
[DEFAULT]
enabled_drivers=solaris
auth_strategy=     este valor depende de si Ironic es o no independiente
pybasedir = /usr/lib/python2.6/vendor-packages/ironic
bindir = /usr/lib/ironic
host = ironic

[ai]
server=     este valor depende de si AI es local o remota
username=ironic
port=22
timeout=10
ssh_key_file=/var/lib/ironic/.ssh/id_rsa
deploy_interval=30

[api]
port=6385

[conductor]
api_url=http://localhost:6385/
heartbeat_timeout=60
heartbeat_interval=60
sync_power_state_interval=300
check_provision_state_interval=120

[database]
connection= mysql://ironic:ironic-service-password@$CONTROLLER_ADMIN_NODE/ironic

[solaris_ipmi]
imagecache_dirname = /var/lib/ironic/images
imagecache_lock_timeout = 60

[glance]
glance_host = glance-serverIP
glance_port = 9292
glance_protocol = http
glance_api_servers = glance-serverIP:port
auth_strategy =     para que Ironic use Keystone para interactuar con Glance, especifique keystone

[keystone_authtoken]     configure si está en DEFAULT, auth_strategy = keystone
auth_host = localhost
auth_uri = http://$CONTROLLER_ADMIN_NODE:5000/v2.0
identity_uri = http://$CONTROLLER_ADMIN_NODE:35357
admin_user = ironic
admin_password = ironic-service-password
admin_tenant_name = tenant
signing_dir = /var/lib/ironic/keystone-signing

[neutron]
```

`auth_strategy =` *para que Ironic use Keystone para interactuar con Neutron, especifique keystone*

7. Asegúrese de que se estén ejecutando los servicios SMF para Ironic.

```
ironic-localhost# svcs -a | grep rabbitmq
ironic-localhost# svcs -a | grep ironic
```

Si Ironic se encuentra en modo independiente, es posible que deba activar los servicios manualmente.

```
ironic-localhost# svcadm enable rabbitmq
ironic-localhost# svcadm enable ironic-db
ironic-localhost# svcadm enable ironic-api ironic-conductor
```

8. (Opcional) Pruebe la utilidad de línea de comandos.

a. Establezca las variables de shell global para Ironic.

- Si `auth_strategy` está establecido en `noauth`, establezca las variables de shell que se indican a continuación:

```
ironic-localhost# export OS_AUTH_TOKEN=fake-token
ironic-localhost# export IRONIC_URL=http://localhost:6385
```

- Si `auth_strategy` está establecido en `keystone`, establezca las variables de shell que se indican a continuación.

Si el servicio Keystone está en un nodo distinto de Ironic, para `OS_AUTH_URL`, especifique el nombre de host o la dirección IP donde está instalado Keystone.

```
ironic-localhost# export OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE:5000/v2.0"
ironic-localhost# export OS_TENANT_NAME=service
ironic-localhost# export OS_USERNAME=ironic
ironic-localhost# export OS_PASSWORD=ironic-service-password
ironic-localhost# export IRONIC_URL=http://localhost:6385/
```

b. Ejecute una línea de comandos de Ironic.

```
# ironic driver-list
+-----+-----+
| Supported driver(s) | Active host(s) |
+-----+-----+
| solaris             | ironic         |
+-----+-----+
```

Actualmente, `solaris` es el único controlador activado y probado en Ironic. No obstante, puede agregar controladores a la lista incluyendo sus nombres en la sección `[enabled_drivers]` del archivo `/etc/ironic/ironic.conf`. Después de agregar los controladores, debe reiniciar los servicios SMF `ironic-api` e `ironic-conductor`.

Visión general: Implementación de reconstrucción completa con Ironic

Con el controlador `solaris` de Ironic, puede aprovisionar un nodo de reconstrucción completa mediante la función del archivo unificado (UAR) de Oracle Solaris o mediante Image Packaging System (IPS). Cuando crea un nodo, transfiere información al controlador `solaris` mediante elementos configurables de nodo.

En la siguiente tabla, se muestran los elementos para crear nodos. El elemento `driver_info/archive_uri` corresponde al aprovisionamiento de nodos con un archivo UAR. Los elementos restantes de la tabla corresponden al aprovisionamiento de nodos mediante IPS.

Elemento	Descripción	Ejemplo
<code>driver_info/archive_uri</code>	URI de archivo unificado con el cual se realiza el aprovisionamiento de la reconstrucción completa.	<code>http://host.example.com/sol-11_3-x86.uar</code>
<code>driver_info/ai_service</code>	Nombre del servicio AI que se debe utilizar.	<code>default-x86</code>
<code>driver_info/publishers</code>	Lista de editores IPS separados por el signo más (+) y que utiliza el formato de asignación de nombres <code>publisher-name@origin</code> .	<code>solaris@http://ipkg.us.oracle.com/solarisN/dev+userland@http://my.example.repo</code>
<code>driver_info/fmri</code>	Lista de FMRI de paquete que se deben instalar separados por el signo más (+).	<code>pkg:/group/system/solaris-small-server+pkg:/cloud/openstack/nova</code>
<code>driver_info/install_profiles</code>	Lista de URI de los perfiles de configuración que se deben aplicar en el entorno de instalación. Los URI están separados por el signo más (+).	<code>http://host.example.com/profile1.xml+glance://glance-image</code>
<code>driver_info/sc_profiles</code>	Lista de URI de los perfiles de configuración del sistema que se deben aplicar en el sistema instalado. Los URI están separados por el signo más (+).	<code>http://host.example.com/profile1.xml+glance://glance-image</code>
<code>driver_info/manifest</code>	URI del manifiesto de AI que se debe utilizar para aprovisionar la reconstrucción completa.	<code>http://host.example.com/my-manifest.xml</code>
<code>driver_info/ipmi_address</code>	Dirección IP o nombre de host de la consola serie que se debe conectar al ILOM del nodo que se debe aprovisionar.	<code>192.168.2.200</code>
<code>driver_info/ipmi_username</code>	Nombre de usuario para la conexión a IPMI.	<code>root</code>
<code>driver_info/ipmi_password</code>	Contraseña para la conexión a IPMI.	<code>password</code>

Si aprovisiona un nodo mediante un archivo UAR, solo debe proporcionar la información para `driver_info/archive_uri`. El URI de archivo solo puede usar uno de los esquemas que se muestran en la siguiente lista. Las mismas opciones de esquema se aplican a la especificación de perfiles si decide aprovisionar el nodo con IPS.

- `file://`
- `http://`
- `https://`
- `glance://`

Nota - Si bien Glance se usa generalmente para almacenar imágenes de instalación, la agregación de perfiles al almacenamiento es aceptable.

Si aprovisiona un nodo mediante IPS, en primer lugar, asegúrese de que exista, al menos, un servicio AI predeterminado para la arquitectura del nodo, como se muestra en el siguiente ejemplo:

```
# installadm list
Service Name          Status Arch  Type Alias Aliases Clients Profiles Manifests
-----
default-i386          on   i386  pkg  yes   0     0     0     1
default-sparc         on   i386  pkg  yes   0     0     0     1
ironic-x86            on   i386  pkg  no    0     0     0     1
ironic-sparc         on   i386  pkg  no    0     0     0     1

# installadm list -vcn ironic-x86
There are no clients configured for service 'ironic-x86'.

# installadm list -vmn ironic-x86
Service Name      Manifest Name Status  Criteria
-----
ironic-x86       orig_default  default none
ironic-sparc     orig_default  default none
```

Cuando usa IPS para aprovisionar nodos, la especificación de `driver_info/ai_service` es opcional. Si omite el nombre del servicio AI, se utiliza el servicio AI predeterminado para la arquitectura del nodo.

Si especifica paquetes personalizados para `driver_info/fmri`, también debe especificar los editores para el elemento `driver_info/publishers`.

Uso de Ironic para la implementación de reconstrucción completa

En la siguiente tarea, se proporcionan los pasos básicos, desde la creación del nodo hasta la implementación.

Para obtener más información sobre la función de archivos unificados de Oracle Solaris, consulte *Uso de archivos unificados para la recuperación y la clonación del sistema*. Este manual se encuentra en la biblioteca de la versión de Oracle Solaris en [Documentación de sistemas operativos](#).

▼ Cómo implementar la reconstrucción completa a partir de un archivo UAR

Para obtener un ejemplo concreto en el que se utilizan los siguientes pasos, consulte [Ejemplo 6, “Implementación de un nodo mediante un archivo UAR”](#).

Antes de empezar Asegúrese de que ya exista el archivo UAR que se debe utilizar.

- 1. Cree las siguientes variables para facilitar el uso de la utilidad de la línea de comandos de Ironic.**
 - IP: la dirección IP que se utiliza para conectarse al ILOM de los nodos.
 - USER: generalmente, el usuario es root.
 - PASS: la contraseña root.
 - HOST_MAC: la dirección MAC del sistema.
- 2. Cree el nodo de Ironic.**
`# ironic node-create options`
Se muestran las propiedades del nuevo nodo, incluido el UUID del nodo.
- 3. Cree una variable para el UUID del nodo para contar con una referencia sencilla.**
- 4. Actualice el nodo especificando la ubicación del UAR que se debe instalar.**
`# ironic node-update options`
- 5. Cree un puerto asociado para este nodo.**
`# ironic port-create options`
- 6. (Opcional) Valide los campos que especificó para el nodo.**

```
# ironic node-validate options
```

7. Aprovechone el nodo.

```
# ironic node-set-provision-state options
```

8. (Opcional) Muestre el estado de la implementación.

```
# ironic node-show options
```

Nota - La salida es distinta si ejecuta el comando mientras el aprovisionamiento está en curso, en lugar de hacerlo una vez que el proceso ha finalizado.

ejemplo 6 Implementación de un nodo mediante un archivo UAR

En este ejemplo, se supone que se cumplen las siguientes condiciones:

- Información básica de la máquina que alojará el nodo
 - Nombre del host: mynewnode.example.com
 - Arquitectura: x86
 - Dirección IP: 1.1.1.1
 - Dirección MAC: 01:02:03:04:05:06
- Información básica del host de ILOM:
 - Nombre del host: mynewnode-aa.example.com
 - Dirección IP: 2.2.2.2
 - Usuario: root
 - Contraseña: password
- Nombre del archivo UAR: myuar.server/sol11-3-x86.uar

```
# export ILOM_IP=2.2.2.2
# export ILOM_USER=root
# export ILOM_PASS=password
# export HOST_MAC=01:02:03:04:05:06

# ironic node-create -d solaris -i ipmi_address=$ILOM_IP \
  -i ipmi_username=$ILOM_USER -i ipmi_password=$ILOM_PASS
```

Property	Value
uuid	4eacbfde-4977-4d8c-8043-8cbe8f876187
driver_info	{u'ipmi_address': u'2.2.2.2', u'ipmi_username': u'root', u'ipmi_password': u'password'}
extra	{}
driver	solaris
chassis_uuid	None
properties	{}

Se crea el nodo

```
# export NODE=4eacbfde-4977-4d8c-8043-8cbe8f876187    Se guarda el UUID

# ironic node-update $NODE \
  add driver_info/archive_uri=http://myuar.server/sol11-3-x86.uar
+-----+
| Property          | Value                                     |
+-----+-----+
| instance_uuid     | None                                     |
| target_power_state | None                                     |
| properties        | {}                                       |
| maintenance       | False                                    |
| driver_info       | {u'archive_uri': u'http://myuar.server/sol11-3-x86.uar', |
|                   | u'ipmi_address': u'2.2.2.2', u'ipmi_username': u'root', |
|                   | u'ipmi_password': u'password'}         |
| extra             | {}                                       |
| last_error        | None                                     |
| created_at        | 2014-10-03T15:38:43+00:00              |
| target_provision_state | None                                     |
| driver            | solaris                                  |
| updated_at        | 2014-10-03T15:53:04+00:00              |
| instance_info     |                                           |
| chassis_uuid      | None                                     |
| provision_state   | None                                     |
| reservation       | None                                     |
| power_state       | None                                     |
| console_enabled   | False                                    |
| uuid              | 4eacbfde-4977-4d8c-8043-8cbe8f876187   |
+-----+-----+

# ironic port-create -n $NODE -a $HOST_MAC
+-----+-----+
| Property | Value                                     |
+-----+-----+
| node_uuid | 4eacbfde-4977-4d8c-8043-8cbe8f876187 |
| extra     | {}                                       |
| uuid      | 4c765ab0-2529-4463-a51b-e5544dd15a32 |
| address   | 01:02:03:04:05:06                       |
+-----+-----+

# ironic node-validate $NODE
+-----+-----+
| Interface | Result | Reason          |
+-----+-----+
| console   | None   | not supported   |
| deploy    | True   |                 |
| management | True   |                 |
| power     | True   |                 |
+-----+-----+

# ironic node-set-provision-state $NODE active    Se aprovisiona el nodo

# ironic node-show $NODE
+-----+-----+

```

Property	Value
instance_uuid	None
target_power_state	None
properties	{}
maintenance	False
driver_info	{'archive_uri': u'http://myuar.server/sol11-3-x86.uar', u'ipmi_address': u'2.2.2.2', u'ipmi_username': u'root', u'ipmi_password': u'password'}
extra	{}
last_error	None
created_at	2014-10-03T15:38:43+00:00
target_provision_state	deploy_complete
driver	solaris
updated_at	2014-10-03T15:53:04+00:00
instance_info	
chassis_uuid	None
provision_state	active
reservation	None
power_state	power on
console_enabled	False
uuid	4eacbfde-4977-4d8c-8043-8cbe8f876187

Si ejecuta el comando `ironic node-show` mientras el aprovisionamiento está en curso, `provision_state` indica un estado distinto de `active`.

▼ Cómo interrumpir la comunicación de un nodo

También puede usar este procedimiento en los casos en que falla una operación de aprovisionamiento.

1. Configure el nodo en un estado suprimido.

```
# ironic node-set-provision-state $NODE deleted
```

2. Muestre la información del nodo.

En el siguiente ejemplo, las propiedades con asteriscos (*) indican el estado suprimido del nodo.

```
# ironic node-show $NODE
```

Property	Value
instance_uuid	None
target_power_state	None
properties	{}
maintenance	True
driver_info	{'archive_uri': u'http://myuar.server/sol11-3-x86.uar',

```

|          | u'ipmi_address': u'2.2.2.2', u'ipmi_username': u'root', |
|          | u'ipmi_password': u'password'} |
|extra     | {} |
|last_error| None |
|created_at| 2014-10-03T15:38:43+00:00 |
|target_provision_state| None | *
|driver     | solaris |
|updated_at| 2014-10-03T15:53:04+00:00 |
|instance_info| {} |
|chassis_uuid| None | *
|provision_state| None | *
|reservation| None |
|power_state| power off | *
|console_enabled| False |
|uuid       | 4eacbfde-4977-4d8c-8043-8cbe8f876187 |
+-----+-----+-----+

```

3. Si está solucionando los problemas de un proceso de aprovisionamiento fallido, valide los campos del nodo.

Los problemas con cualquiera de los elementos `driver_info` que se especificaron para el nodo se identifican en la columna Reason.

```

# ironic node-validate $NODE
+-----+-----+-----+
| Interface | Result | Reason |
+-----+-----+-----+
| console   | None   | not supported |
| deploy    | True   | |
| management| True   | |
| power     | True   | |
+-----+-----+-----+

```


Solución de problemas de OpenStack

En este capítulo, se describen los problemas de OpenStack que están asociados con la versión actual. También se proporcionan consejos para la resolución de problemas y soluciones alternativas para resolver problemas básicos que puedan surgir durante la configuración. Se tratan los temas siguientes:

- “Ayuda de la línea de comandos” [119]
- “Limitaciones conocidas” [120]
- “Examen de los archivos log” [121]
- “Investigación y resolución de problemas” [123]
- “Consejos y trucos generales sobre depuración” [130]
- “Sitios útiles” [131]

Ayuda de la línea de comandos

En OpenStack, los comandos corresponden a los componentes de OpenStack. Por ejemplo, el comando `nova` se aplica a las operaciones de cálculo, `cinder` a almacenamiento y `neutron` a redes.

Para obtener ayuda sobre cómo usar estos comandos con la sintaxis correcta, los subcomandos admitidos, las opciones posibles, etc., use el comando `command-component help`, por ejemplo `nova help` o `neutron help`. Puede filtrar la lista de subcomandos posibles para usar con el comando `command-component` usando el comando `grep`. Por ejemplo, para mostrar los subcomandos de `neutron` relacionados con los enrutadores, debe escribir el siguiente comando:

```
# neutron help | grep router
l3-agent-list-hosting-router  List L3 agents hosting a router.
l3-agent-router-add          Add a router to a L3 agent.
l3-agent-router-remove       Remove a router from a L3 agent.
net-gateway-connect          Add an internal network interface to a router.
router-create                 Create a router for a given tenant.
router-delete                 Delete a given router.
router-gateway-clear          Remove an external network gateway from a router.
router-gateway-set            Set the external network gateway for a router.
router-interface-add          Add an internal network interface to a router.
```

<code>router-interface-delete</code>	Remove an internal network interface from a router.
<code>router-list</code>	List routers that belong to a given tenant.
<code>router-list-on-l3-agent</code>	List the routers on a L3 agent.
<code>router-port-list</code>	List ports that belong to a given tenant, with specified router.
<code>router-show</code>	Show information of a given router.
<code>router-update</code>	Update router's information.

Entonces, para obtener detalles específicos acerca de un subcomando, como `router-list` que identifica los enrutadores en la nube, debe escribir el siguiente comando:

```
# neutron help router-list
```

Limitaciones conocidas

Los siguientes son problemas conocidos con OpenStack (Juno) en Oracle Solaris 11.3:

- Solo se admiten por completo los nodos de Nova que ejecutan Oracle Solaris, porque Neutron admite únicamente un solo plugin 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 11.3.
- 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 al instalar el paquete `cinder`. Sin embargo, en un despliegue por defecto de Cinder como el que se describe en [“Configuración del nodo de almacenamiento” \[49\]](#), el servicio actualmente no funciona para `backup_volume`.

- 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 `UploadToImage`.
- 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
```

Examen de los archivos log

Los servicios SMF y varios procesos de Oracle Solaris producen archivos log donde se 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.

Debido a que OpenStack se instala generalmente en varios sistemas, los archivos log que debe consultar también están en distintas ubicaciones. Para obtener una resolución de problemas más sistemática, examine logs por nodo.

Para obtener información general sobre la resolución de problemas relacionados con los servicios SMF, consulte la sección de resolución de problemas de *Gestión de servicios del sistema en Oracle Solaris* que se aplique a la versión del sistema operativo. Vaya a la biblioteca correspondiente en <http://docs.oracle.com/en/operating-systems/>.

Para ver logs de servicio, asegúrese de tener la autorización adecuada. 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

Para la resolución del problema, utilice los siguientes comandos generales:

- Para averiguar qué servicios de OpenStack se están ejecutando en un nodo concreto:

```
# svcs -a | grep openstack
```

- Para mostrar los servicios que pueden estar en modo de mantenimiento:

```
# svcs -x
svc:/application/openstack/swift/swift-replicator-rsync:
    default (OpenStack Swift Replication Service)
State: maintenance since Fri May 22 04:06:11 2015
Reason: Start method exited with $SMF_EXIT_ERR_FATAL.
    See: http://support.oracle.com/msg/SMF-8000-KS
    See: rsync(1)
    See: rsyncd.conf(5)
    See: /var/svc/log/application-openstack-swift-swift-replicator-rsync:default.log
Impact: This service is not running.
```

Si un servicio está en modo de mantenimiento, consulte el archivo log del servicio.

- Para identificar el log de un servicio de OpenStack específico:

```
# svcs -L openstack-service
```

Por ejemplo:

```
# svcs -L neutron-server
/var/svc/log/application-openstack-neutron-neutron-server:default.log
```

Con la autorización adecuada, puede combinar opciones, como `-Lv`, para mostrar y ver el log del servicio.

- Para identificar de forma inmediata instancias de error que se registran en un archivo log específico, puede utilizar comandos de UNIX comunes, como `grep`.

```
# grep keyword `svcs -L openstack-service`
```

Puede buscar las instancias de error, `warning` y otras palabras clave críticas para leer directamente los mensajes de error.

- Para comprobar las propiedades de EVS al solucionar problemas de red, use los diversos subcomandos de `evsadm`, como `evsadm show-prop`.
- Una línea de comandos `list` de OpenStack puede proporcionar información sobre el estado de los componentes. Por ejemplo, `nova list` muestra instancias activas en el nodo de

cálculo, mientras que `nova image-list` muestra las imágenes disponibles y su estado. Para obtener un ejemplo de salida, consulte “[Visualización de información sobre imágenes](#)” [63].

Los siguientes logs suelen contener información útil para la resolución de problemas:

- `nova-compute`
- `nova-scheduler`
- `cinder-scheduler`
- `neutron-server`

Además de los archivos log del servicio SMF, también puede consultar los logs en el directorio `/var/log`. Al igual que otros procesos de Oracle Solaris, los servicios de OpenStack también generan sus propios archivos log en los directorios `/var/log/openstack-service`.

Por ejemplo, los archivos log del almacén de imágenes de OpenStack están en `/var/log/glance`. Es posible que los problemas con la creación y el inicio de una instancia de VM se registren en el directorio `/var/log/zones`. Los logs de mensajes se almacenan como `/var/log/rabbitmq/rabbit@hostname.log`.

La mayoría de los archivos de configuración de OpenStack tienen el nombre de servicio de 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`. Los que corresponden a Nova se encuentran en `/etc/nova`, etc. Puede utilizar estos archivos para la resolución de problemas de un servicio específico mediante la definición de los siguientes parámetros en el archivo de configuración del servicio o eliminando los comentarios de dichos parámetros:

- `debug=true`
- `verbose=true`

Estos parámetros permiten ver más 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 <http://www.oracle.com/technetwork/articles/servers-storage-admin/getting-started-openstack-os11-2-2195380.html> y en la *Referencia de configuración de OpenStack* en el [sitio de documentación de OpenStack](#).

Nota - Los comandos de servicio de OpenStack individuales también pueden tomar la opción `--debug`. Esta opción es el equivalente a establecer `debug=true` en un archivo de configuración.

Investigación y resolución de problemas

En esta sección, se describen algunos problemas que pueden surgir al instalar y configurar OpenStack.

En el ejemplo siguiente, se muestra un error relativo al panel de control.

```
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.
```

Estos mensajes pueden indicar que la clave de host RSA se cambió y no se propagó a todos los componentes. Para obtener más información sobre la configuración de claves RSA, consulte [Cómo instalar y configurar Neutron](#) y [Cómo configurar el nodo de cálculo](#).

El siguiente informe de error puede incluirse en el log nova-scheduler.

```
controller# grep error `svcs -L nova-scheduler`
2014-12-03 12:49:19.271 3475 TRACE
nova.openstack.common.rpc.common error: [Errno 32] Broken pipe
```

Un error de canalización rota normalmente se notifica al refrescar un servicio de OpenStack y no los otros. Si realizó cambios en los archivos de configuración en un nodo, refresque todos los servicios en el nodo. El siguiente comando reinicia los servicios que están en línea, pero necesitan refrescarse.

```
controller# svcs `*openstack*` | grep online \
| awk -e '{print $3}' | xargs svcadm restart
```

Un error también puede deberse a la falta de recursos. Durante la creación de una instancia de VM, el log nova-compute puede mostrar mensajes parecidos al siguiente:

```
[abc-123-def-456] Build of instance
abc-123-def-456 aborted: Image
xyz-987-uvw-654 is unacceptable: Image query failed.
Possibly invalid or corrupt. Log file location: controller:/tmp/archive_log.4249
```

Además, el log también indicaría `out of space/storage`. Utilice el comando `top` para ver los recursos del sistema. Si el sistema tiene menos de 1 GB de memoria, es posible que deba agregar más.

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 [“Preparación de imágenes para el repositorio de Glance” \[61\]](#) para obtener 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 Oracle 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 Oracle 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).

Problemas de credenciales

En ciertas ocasiones, pueden aparecer mensajes de error relacionados con credenciales incorrectas que le impiden ejecutar comandos de servicio. Por ejemplo, cuando ejecuta un comando glance, es posible que se genere el siguiente mensaje de error:

```
Invalid OpenStack Identity credentials.
```

La causa raíz de este mensaje puede ser diferente cada vez. Por lo tanto, debe examinar los logs para restringir las causas posibles. Con el servicio glance como ejemplo, mire los contenidos de los log del servicio SMF de Glance. El archivo `/var/log/glance/api.log` puede informar lo siguiente:

```
WARNING keystonemiddleware.auth_token [-] Authorization failed for token
```

Si estableció `Debug = True` y `Verbose = True` en los archivos de configuración de glance, se proporcionan más detalles en el archivo `/var/svc/log/application-openstack-glance-glance-api:default.log` como los que se detallan a continuación:

```
DEBUG keystonemiddleware.auth_token [-] Received request from user:
  user_id None, project_id None, roles None service: user_id None,
  project_id None, roles None
__call__/_usr/lib/python2.7/vendor-packages/keystonemiddleware/auth_token.py:821
```

Puede explorar las siguientes áreas para resolver el problema.

- Compruebe sus archivos de configuración de servicio para garantizar que los parámetros pertinentes estén definidos correctamente.
- Compruebe que las variables de shell globales del servicio sean correctas. Para el servicio Glance, por ejemplo, se deben establecer las siguientes variables:
 - `OS_USERNAME=glance`
 - `OS_PASSWORD=service-password`
 - `OS_TENANT_NAME=service`
 - `OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE:5000/v2.0`

Si el comando sigue fallando con los mismos mensajes de error, quizá necesite volver a crear el usuario del servicio para generar credenciales nuevas. Estudie el ejemplo siguiente:

```
# export OS_USERNAME=keystone
# export OS_PASSWORD=service-password
# export OS_TENANT_NAME=service
```

```
# export OS_AUTH_URL=http://$CONTROLLER_ADMIN_NODE:5000/v2.0

# keystone user-list
```

De la salida de este comando, anote el número de ID del usuario de servicio dañado. Deberá suprimir el usuario y luego volver a crearlo con las credenciales correctas:

```
# keystone user-delete user-ID

# keystone user-create --name glance --password service-password
# keystone user-role-add --user=glance --tenant=service --role=admin
```

Problema relacionado con Horizon

Después de iniciar las instancias de VM, el panel de control de Horizon puede quedar inaccesible y aparece el mensaje de error 404 Not Found. El log de los servicios de Apache incluiría la siguiente entrada:

```
Oct 13 16:13:53 Executing start method (" /lib/svc/method/http-apache24 start"). ]
Apache version is 2.4
(125) Address already in use: AH000/2: make_sock: could not bind to address [::]:80
Oct 13 16:13:55 Method "start" exited with status 0. ]
```

El log indica que la dirección no se puede enlazar al puerto 80 porque el puerto está ocupado.

A partir de la versión Juno, OpenStack usa Apache 2.4 en lugar de Apache 2.2, que se usaba en las versiones anteriores de OpenStack. Asegúrese de que la versión correcta de Apache esté activada y esté llevando a cabo la recepción en ese puerto.

Para liberar el puerto, siga estos pasos:

1. Obtenga el ID del proceso que está llevando a cabo la recepción actualmente en el puerto.

```
# netstat -uan -f inet | grep "*.80"
```

En este procedimiento, se supone que se están usando direcciones IPv4 en la configuración y que el proceso que retiene el puerto 80 está llevando a cabo la recepción en todas esas direcciones. Si el proceso está llevando a cabo la recepción del tráfico IPv6, el comando podría no dar ningún resultado.

2. En función del ID del proceso, identifique el proceso o el servicio real. Puede utilizar uno de los siguientes comandos:

```
# svcs -p | egrep "online|pid http" | ggrep -B1 pid
```

O

```
# ps -lf -p pid
```

3. Si la versión incorrecta de Apache está usando el puerto, desactive el servicio.

4. Active la versión correcta de Apache para la versión de Juno.

Si la versión correcta de Apache se encuentra en el modo de mantenimiento, borre el servicio antes de activarlo.

En el siguiente ejemplo, se ilustra cómo liberar el puerto 80 y pasar a la versión correcta de Apache.

```
# netstat -uan -f inet | grep "*.80"
*.80      *.*      root      5560 httpd    0      0 128000    0 LISTEN
*.8080    *.*      webservd   1124 java     0      0 128000    0 LISTEN
*.8009    *.*      webservd   1124 java     0      0 128000    0 LISTEN

# svcs -p | egrep "online|5560 http" | ggrep -B1 5560
online    Aug_31   svc:/network/http:apache22
          Sep_09   5560 httpd

# svcadm disable apache22
# svcadm clear apache24
# svcadm enable apache24
```

Problema relacionado con Heat

Nota - Este problema es aplicable solo si está usando la versión SRU 2 de Oracle Solaris 11.3 o versiones anteriores.

Puede producirse un error al crear una pila de Heat si se usa el comando normal `heat stack-create`, similar a lo que se muestra en el siguiente ejemplo:

```
# heat stack-create -f heat-template -P parameters stack-name
```

donde *heat-template* es una plantilla HOT definida de manera válida.

El log `neutron-server` incluiría entradas como las del siguiente ejemplo. Tenga en cuenta que en el ejemplo se muestra un log con ocultación que incluye solo la información relevante para el problema:

```
stack_name          | stack2          |
stack_owner         | admin           |
stack_status        | CREATE_FAILED   |
stack_status_reason | Resource CREATE failed: InternalServerError: Request |
                   | Failed: internal server error while processing your |
                   | request         |
```

El problema es que el puerto DHCP para la red y el puerto normal se están creando al mismo tiempo, lo que produce una condición de timeout. Como solución alternativa, puede desactivar DHCP para la subred. Una mejor solución sería cambiar a una versión de SRU de Oracle Solaris 11.3 posterior.

Problemas de escalabilidad

Por defecto, RabbitMQ impone un límite de descriptor de archivo de 255. Este límite puede fácilmente impedir que se siga escalando la nube después de haber creado algunos nodos de cálculo. Para evitar este bloqueo, aumente el valor del límite en el archivo `/etc/rabbitmq/rabbitmq-env.conf`:

```
# Increase soft limit on file descriptors for RabbitMQ
ulimit -n 8192
```

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 \[130\]](#).
- 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 remove-vport vport
```

Consejos y trucos generales sobre depuración

Los consejos generales a continuación pueden ayudar a comenzar con la resolución de problemas en OpenStack:

- Establezca `debug = true` y `verbose = true` en los diferentes archivos de configuración para ayudar a diagnosticar problemas. En algunos archivos de configuración, puede buscar un valor de marcador de posición que tenga un comentario y que puede cambiar para activar el registro detallado.
- Cuando realiza cambios a un archivo de configuración del componente, reinicie los servicios del componente para que se hagan efectivos.
- Use el comando `tail -30 `svcs -L service-name`` que le proporciona información del log de SMF. Si se activó la depuración además del registro detallado, quizá necesite aumentar la cantidad de líneas que especifica para el comando `tail`.
- Los procesos de Horizon pasan por Apache. Por lo tanto, para diagnosticar Horizon, active `debug = True` en el archivo `/etc/openstack_dashboard/local_settings.py`. Los errores de Django se generan la página web.

- Los componentes de Nova se crean sobre las zonas de Oracle Solaris. Por lo tanto, también puede consultar los logs en `/var/log/zones` para resolver los problemas de Nova.

Sitios útiles

Consulte los siguientes sitios para obtener consejos de resolución de problemas para solucionar los problemas de OpenStack.

- <https://blogs.oracle.com/openstack/>
- <https://ask.openstack.org/en/questions/>
- <https://raymii.org/s/tags/openstack.html>

Archivos y servicios de configuración comunes de OpenStack

En este apéndice, se muestran los archivos de configuración típicos de los componentes principales de OpenStack además de los servicios SMF de OpenStack.

Archivos de configuración

Archivos cinder

- `/etc/cinder/api_paste.ini`
- `/etc/cinder/cinder.conf`

Archivos glance

- `/etc/glance/glance-api.conf`
- `/etc/glance/glance-cache.conf`
- `/etc/glance/glance-registry.conf`
- `/etc/glance/glance-scrubber.conf`
- `/etc/glance/glance-registry-paste.ini`
- `/etc/glance/glance-api-paste.ini`

Archivos de Keystone

- `/etc/keystone/keystone.conf`
- `/etc/keystone/keystone-paste.ini`

Archivos neutron

- /etc/neutron/neutron.conf
- /etc/neutron/dhcp_agent.ini
- /etc/neutron/l3_agent.ini
- /etc/neutron/api-paste.ini
- /etc/neutron/metadata_agent.ini
- /etc/neutron/plugins/evs/evs_plugin.ini

Archivos nova

- /etc/nova/nova.conf
- /etc/nova/api_paste.ini

Archivos de Horizon

- /etc/openstack_dashboard/local_settings.py
- /etc/apache2/2.4/httpd.conf
- /etc/apache2/2.4/conf.d/openstack-dashboard-http.conf
- o
 /etc/apache2/2.4/conf.d/openstack-dashboard-tls.conf

Archivos de Swift

- /etc/swift/swift.conf
- /etc/swift/account-server.conf
- /etc/swift/container-server.conf
- /etc/swift/object-server.conf
- /etc/swift/proxy-server.conf
- /etc/swift/rsyncd.conf

Servicios SMF de OpenStack

Cinder

```
svc:/application/openstack/cinder/cinder-db:default
svc:/application/openstack/cinder/cinder-backup:default
svc:/application/openstack/cinder/cinder-scheduler:default
svc:/application/openstack/cinder/cinder-api:default
svc:/application/openstack/cinder/cinder-volume:setup
svc:/application/openstack/cinder/cinder-volume:default
```

Glance

```
svc:/application/openstack/glance/glance-db:default
svc:/application/openstack/glance/glance-registry:default
svc:/application/openstack/glance/glance-scrubber:default
svc:/application/openstack/glance/glance-api:default
```

Keystone

```
svc:/application/openstack/keystone:default
```

Neutron

```
svc:/application/openstack/neutron/neutron-server:default
svc:/application/openstack/neutron/neutron-dhcp-agent:default
svc:/application/openstack/neutron/neutron-metadata-agent:default
svc:/application/openstack/neutron/neutron-l3-agent:default
```

Nova *

```
svc:/application/openstack/nova/nova-objectstore:default
svc:/application/openstack/nova/nova-consoleauth:default
svc:/application/openstack/nova/nova-novncproxy:default
svc:/application/openstack/nova/nova-api-metadata:default
svc:/application/openstack/nova/nova-api-ec2:default
svc:/application/openstack/nova/nova-api-osapi-compute:default
svc:/application/openstack/nova/nova-conductor:default
svc:/application/openstack/nova/nova-cert:default
svc:/application/openstack/nova/nova-compute:default
svc:/application/openstack/nova/nova-scheduler:default
```

* Es posible que se muestren otros servicios de Nova según la configuración del nodo de cálculo.

Swift

```
svc:/application/openstack/swift/swift-object-expirer:default
svc:/application/openstack/swift/swift-account-reaper:default
svc:/application/openstack/swift/swift-container-replicator:default
svc:/application/openstack/swift/swift-account-replicator:default
svc:/application/openstack/swift/swift-object-auditor:default
svc:/application/openstack/swift/swift-container-updater:default
svc:/application/openstack/swift/swift-container-sync:default
svc:/application/openstack/swift/swift-object-updater:default
svc:/application/openstack/swift/swift-account-auditor:default
svc:/application/openstack/swift/swift-replicator-rsync:default
svc:/application/openstack/swift/swift-container-auditor:default
svc:/application/openstack/swift/swift-object-replicator:default
svc:/application/openstack/swift/swift-container-reconciler:default
svc:/application/openstack/swift/swift-container-server:default
svc:/application/openstack/swift/swift-object-server:default
svc:/application/openstack/swift/swift-proxy-server:default
svc:/application/openstack/swift/swift-account-server:default
```


Índice

A

- acceso a la consola, 48
- agente L3, 55
- archivos unificados
 - creación de imágenes de OpenStack desde, 62
 - descarga, 18
 - instalación de OpenStack en un solo nodo, 20
 - servicio AI de reconstrucción completa, 19
 - instalación OpenStack de un único nodo, 17
- archivos unificados de OpenStack *Ver* archivos unificados
- Automated Installer (AI), 19

B

- base de datos, 34

C

- Cinder
 - archivos de configuración, 40, 50, 92, 99
 - compatibilidad de SAN, 91
 - configuración de listas de control de acceso en hosts remotos, 95
 - configuración de sistemas de almacenamiento remotos, 91
 - controlador iSCSI de Cinder para ZFSSA, 96
 - copia de seguridad, 120
 - hosts de almacenamiento de destino, 91
 - instalación, 40
 - opciones de despliegue, 91
 - perfil de derechos de usuario, 95
 - uso de ZFSSA como almacenamiento de backend, 98
- claves SSH

- en configuración de EVS, 46
- migración de instancia, 85
- comando `archiveadm`, 62
- comando `zlogin`, 21
- comando `zoneadm`
 - subcomando `boot`, 21
- comando `zonecfg`
 - subcomando `add`, 21
 - subcomando `create`, 21
 - subcomando `select`, 21
 - subcomando `set`, 21
- compatibilidad de SAN en Cinder, 91
- conmutador virtual elástico
 - comando `evsadm`, 56
- conmutador virtual elástico (EVS)
 - claves SSH, 46
 - configuración, 45

D

- Daemon de acceso remoto (RAD), 45
- direcciones IP flotantes, 73
 - Ver también* red

E

- enrutador
 - creación para red externa, 56
- enrutador de proveedor *Ver* Neutron
- escalabilidad, 129
- estructura de zonas, 45
- EVS
 - comando `evsadm`, 58

G

- Glance, 61
 - archivos de configuración, 37
 - instalación, 36
 - secuencia de comandos de creación y carga, 65
 - visualización de información de imagen, 63

H

- Heat
 - instalación, 102
- Horizon
 - configurar acceso SSL, 39
 - visualización de tipos, 81

I

- imagen de rescate, 62
- imágenes, 24, 61
 - Ver también* instancias de VM
 - caché de imágenes, 61
 - copia de seguridad, 62
 - instantáneas, 62
 - rescate, 62
 - servidor de registro, 61
 - visualización de información, 63
- implementación de reconstrucción completa *Ver* reconstrucción completa
- inicio de sesión en la consola de zona, 21
- inquilinos, 24 *Ver* proyectos
- instalación de OpenStack
 - configuración de evaluación, 17
 - configuración en un solo nodo, 17
 - configuración en varios nodos, 25
 - uso de un archivo unificado, 17
- instalación de OpenStack de un solo nodo
 - uso de OVM Server for SPARC, 26
- instalación de OpenStack en un solo nodo
 - uso de un archivo unificado, 17
- instalación de OpenStack en varios nodos
 - arquitectura de referencia de tres nodos, 25
 - preparación, 28
- instancia *Ver* instancias de VM
- instancias de VM, 14, 75
 - Ver también* Nova

- agregación de usuarios, 79
- copia de seguridad, 62
- creación, 75
- imágenes, 24, 61
- inicio de sesión, 79
- instantáneas, 62
- migración, 84
- pares de claves, 75
- rescate, 62
- tipos, 24, 24, 81

instantáneas, 62

Ironic

- archivo `/etc/ironic/ironic.conf`, 108
- componentes, 105
- configuración de AI, 106
- fallos durante el aprovisionamiento, 116
- implementación de reconstrucción completa, 111
- implementación de reconstrucción completa a partir de UAR, 113
- instalación y configuración, 106
- interrupción de la comunicación de un nodo, 116
- servicio, 105
- uso de UAR e IPS, 112

K

- Keystone
 - archivos de configuración, 36
 - instalación, 35
 - secuencia de comandos `sample_data.sh`, 30

L

- LDoms, 26

M

- máquina virtual (VM) *Ver* instancias de VM
- máquina virtual de nube *Ver* instancias de VM
- migración en directo, 84
- MySQL, 34

N

- NAT *Ver* traducción de direcciones de red

- Neutron
- agente L3, 55, 74
 - archivos de configuración, 42
 - creación de un enrutador, 56
 - enrutador, 55
 - instalación, 41
 - red externa, 57
- neutron-l3-agent, 74
- nodo de almacenamiento, 25
- configuración, 49
 - varios hosts de backend, 91
- nodo de cálculo, 25, 25
- Ver también* Nova
 - configuración, 44
 - recuperación, 86
- nodo de controlador, 25
- configuración, 33
- Nova
- archivos de configuración, 38, 45
 - configuración, 44
 - creación de instancias de VM, 75
 - evacuar, 86
 - instalación en nodo de controlador, 38
 - migración de instancias, 84
 - recuperación de nodos de cálculo, 86
 - visualización de información de imagen, 63
- O**
- opciones de depuración, 123
 - optimización de la memoria, 31
 - OVM Server for SPARC, 26
- P**
- panel de control
- creación de instancia de VM, 75
 - creación de redes internas, 70
 - inicio de sesión, 22
 - visualización de imágenes, 24
 - visualización de proyectos, 24
 - visualización de tipos, 24, 81
- perfil de configuración del sistema *Ver* perfil SC
- perfil de privilegios para usuario de Cinder, 95
- perfil de SC, 82
- perfil SC, 62
- plantillas
- hardware *Ver* tipos
- plantillas de hardware *Ver* tipos
- plantillas de instancias *Ver* tipos
- protocolo avanzado de cola de mensajes (AMQP), 34
- protocolo de hora de red (NTP)
- archivo de configuración de cliente, 33
 - archivo de configuración de servidor, 32
 - cliente, 33
 - servidor, 31
- proyectos, 24
- agregación de usuarios, 67, 68
 - creación, 67
 - modificación de rol de usuario, 68
- R**
- RabbitMQ, 34
- RAD, 45
- reconstrucción completa, 111
- Ver también* Ironic
 - elementos configurables de nodo, 111
 - implementación con Ironic, 111
 - implementación con UAR, 113
 - interrupción de la comunicación, 116
 - uso de UAR e IPS, 112
- red
- asociación de direcciones IP flotantes, 73
 - conexión de redes internas a red externa, 73
 - creación de enrutador, 56
 - creación de red externa, 55, 58
 - interna, 70
 - subredes, 70
- red plana, 58
- S**
- secuencia de comandos para crear y cargar una imagen, 65
- secuencia de comandos `sample_data.sh`, 30
- servicio SMF de `neutron-l3-agent`, 56
- servidor de registro, 61
- sistemas de almacenamiento remotos, 91
- Ver también* Cinder

- activación como destinos, 95
- listas de control de acceso (ACL), 95
- servicios de SMF requeridos, 95

SPARC, 26

SQLite, 34

Swift, 27

- archivos de configuración, 52

T

tipos, 24, 81

- Ver también* instancias de VM
- modificación de propiedades, 82
- propiedad extra-specs, 82

TLS

- archivo de configuración, 40

traducción de direcciones de red, 55

traducción de direcciones de red segura (SNAT), 59

- desactivación, 59

U

user_reserve_hint_pct, 31

utilidad de gestión de servicios (SMF)

- servicios requeridos para hosts de destino, 95

Z

ZFS Storage Appliance (ZFSSA), 96

- Ver también* Cinder
- almacenamiento de backend para OpenStack, 96
- utilidad de flujo de trabajo, 98

zonecfg:bootargs, 83