

# Installation et configuration d'OpenStack dans Oracle® Solaris 11.2

**ORACLE**

Référence: E56869-03  
Avril 2015



**Référence: E56869-03**

Copyright © 2014, 2015, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf stipulation expresse de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, accorder de licence, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est livré sous licence au Gouvernement des Etats-Unis, ou à quiconque qui aurait souscrit la licence de ce logiciel pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique

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.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer un risque de dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour des applications dangereuses.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers, sauf mention contraire stipulée dans un contrat entre vous et Oracle. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation, sauf mention contraire stipulée dans un contrat entre vous et Oracle.

**Accessibilité de la documentation**

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité à la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

**Accès aux services de support Oracle**

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> ou le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> si vous êtes malentendant.



# Table des matières

---

<b>Utilisation de la présente documentation</b> .....	9
<b>1 Introduction à OpenStack dans Oracle Solaris 11.2</b> .....	11
Nouveautés d'Oracle Solaris 11.2 .....	11
Intégration d'OpenStack à Oracle Solaris .....	11
Utilisation de la présente documentation .....	14
Configuration requise pour l'installation d'OpenStack .....	16
<b>2 Installation d'une configuration d'évaluation</b> .....	19
Déploiement d'OpenStack Unified Archives .....	19
Téléchargement du fichier image .....	20
Installation du système unique .....	21
Configuration du commutateur virtuel élastique .....	26
Déploiement de Juno OpenStack Unified Archives .....	26
▼ Installation de Juno OpenStack dans une zone de noyau à l'aide du fichier Unified Archives .....	27
Utilisation du tableau de bord OpenStack .....	28
▼ Accès au tableau de bord OpenStack .....	28
Exploration du tableau de bord .....	29
Création et initialisation d'une instance de machine virtuelle .....	33
<b>3 Installation sur plusieurs systèmes pour une configuration Havana multinoeud OpenStack</b> .....	41
Présentation de l'architecture à trois noeuds .....	41
Configuration du noeud de contrôleur .....	45
▼ Configuration d'un noeud de contrôleur .....	46
Installation NTP (Network Time Protocol) .....	47
Installation de MySQL .....	48
Installation de Keystone .....	49
Installation et configuration de Heat .....	50

Installation de Cinder .....	51
Installation de Glance .....	56
Installation et configuration de Neutron sur le noeud du contrôleur .....	57
Installation de Nova .....	58
▼ Configuration d'Horizon .....	59
Configuration du noeud de calcul .....	60
▼ Configuration du noeud de calcul .....	61
Configuration du noeud de réseau .....	62
▼ Configuration du noeud de réseau .....	64
Configuration de l'agent Neutron L3 .....	66
<b>4 Installation sur plusieurs systèmes pour une configuration Juno multinoeud OpenStack .....</b>	<b>77</b>
Présentation de l'architecture à trois noeuds .....	77
Configuration du noeud de contrôleur .....	81
Etapes préliminaires .....	81
Installation NTP (Network Time Protocol) .....	82
Installation de MySQL .....	83
Installation de Keystone .....	85
Installation de Glance .....	86
Installation de Nova .....	88
Installation de Horizon .....	89
Installation de Cinder .....	91
Installation et configuration de Neutron .....	94
Installation et configuration de Heat .....	97
Configuration du noeud de calcul .....	98
▼ Configuration du noeud de calcul .....	98
Configuration du noeud de stockage .....	100
▼ Configuration du noeud de stockage de bloc .....	100
Configuration d'un réseau interne dans OpenStack .....	101
▼ Création d'un réseau externe .....	102
Configuration d'OpenStack avec un réseau externe .....	103
▼ Configuration du réseau externe dans OpenStack .....	104
▼ Création et association des adresses IP flottantes en tant qu'utilisateur locataire .....	108
▼ Respect de la configuration de l'agent L3 .....	109
<b>5 Création d'instances de machine virtuelle .....</b>	<b>111</b>
Gestion des variantes .....	111

---

Affichage des informations sur les variantes .....	112
Modification des spécifications de variante .....	113
Gestion des images .....	114
Affichage des informations sur les images .....	114
Création d'une image .....	115
Ajout d'une image au magasin d'images .....	116
Création d'une instance de machine virtuelle .....	117
▼ Création d'une instance de machine virtuelle à l'aide de l'interface de ligne de commande .....	117
<b>6 Dépannage d'OpenStack .....</b>	<b>121</b>
Limitations connues .....	121
Examen des fichiers journaux .....	123
Examen et résolution des problèmes .....	124
Installation et configuration d'OpenStack .....	125
Installation et configuration de l'instance de machine virtuelle .....	125
<b>Index .....</b>	<b>129</b>



## Utilisation de la présente documentation

---

- **Présentation** – Décrit la procédure d'installation OpenStack et de déploiement de machines virtuelles OpenStack sur des systèmes Oracle Solaris 11.2.
- **Public visé** – Administrateurs de grands systèmes d'installation.
- **Connaissances requises** – Administration réseau et grands systèmes Solaris. Etre familiarisé avec OpenStack est très utile.

## Bibliothèque de documentation produit

La documentation et les ressources de ce produit et des produits associés sont disponibles sur le site Web <http://www.oracle.com/pls/topic/lookup?ctx=E56338>.

## Commentaires

Faites part de vos commentaires sur cette documentation à l'adresse : <http://www.oracle.com/goto/docfeedback>.



# ◆◆◆ 1 CHAPITRE 1

## Introduction à OpenStack dans Oracle Solaris 11.2

---

Ce chapitre décrit la procédure d'utilisation des caractéristiques Oracle Solaris dans OpenStack sur Oracle Solaris 11.2

Ce chapitre présente les sections suivantes :

- [“Nouveautés d'Oracle Solaris 11.2” à la page 11](#)
- [“Intégration d'OpenStack à Oracle Solaris” à la page 11](#)
- [“Utilisation de la présente documentation” à la page 14](#)
- [“Configuration requise pour l'installation d'OpenStack” à la page 16](#)

### Nouveautés d'Oracle Solaris 11.2

La version Juno d'OpenStack est prise en charge depuis Oracle Solaris 11.2 SRU 10. Ce document décrit les procédures de configuration d'OpenStack pour Havana ou Juno.

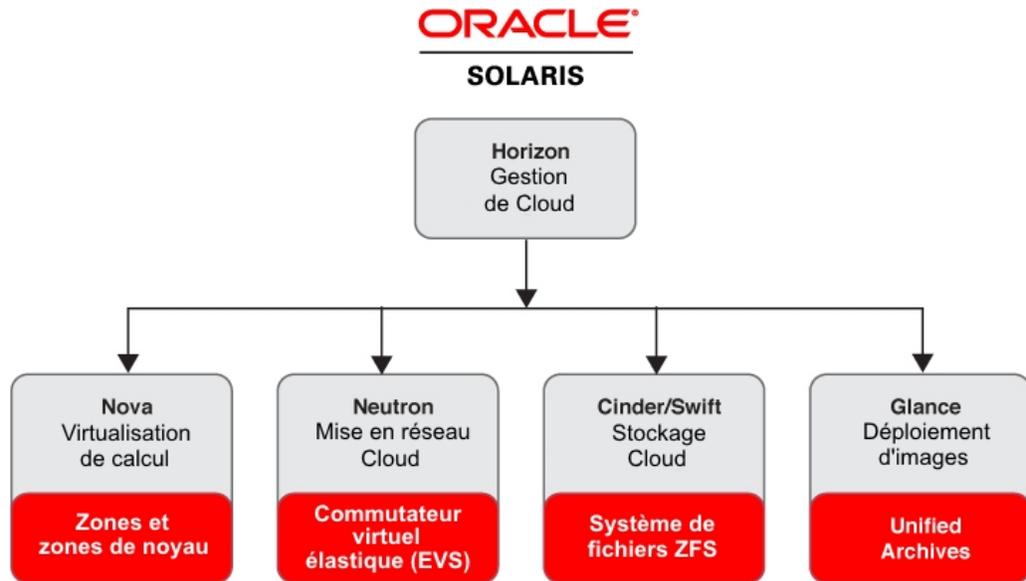
- [Chapitre 3, Installation sur plusieurs systèmes pour une configuration Havana multinoeud OpenStack](#)
- [Chapitre 4, Installation sur plusieurs systèmes pour une configuration Juno multinoeud OpenStack](#)

### Intégration d'OpenStack à Oracle Solaris

Oracle Solaris 11.2 inclut une distribution OpenStack qui est totalement intégrée aux technologies de base de Oracle Solaris 11. OpenStack dans Oracle Solaris 11.2 vous permet de créer une IaaS (Infrastructure as a service) professionnelle en Cloud privé, de sorte que les utilisateurs puissent rapidement créer une mise en réseau virtuel des ressources à l'aide d'un portail Web centralisé.

La figure suivante illustre les fonctions Oracle Solaris 11.2 utilisées pour la mise en oeuvre des services OpenStack. Ces relations sont décrites sous la figure.

**FIGURE 1-1** Intégration Oracle Solaris et OpenStack



Les services OpenStack suivants sont fournis dans Oracle Solaris 11.2

- Nova

Le service de virtualisation de calcul Nova fournit un contrôleur Fabric de cloud computing qui prend en charge différentes technologies de virtualisation. Dans Solaris, les instances de machine virtuelle sont des zones de noyau ou des zones non globales. Les zones sont des environnements virtuels denses et évolutifs, avec de faibles coûts de virtualisation. Les zones de noyau fournissent des versions de noyau indépendantes, permettant une mise à niveau indépendante des instances de machine virtuelle, ce qui est souhaitable pour un Cloud multi-locataire.

Pour plus d'informations sur les zones Oracle Solaris, reportez-vous à la documentation des différentes zones dans la [bibliothèque Oracle Solaris 11.2](#).

- Neutron

Le service de virtualisation du réseau Neutron assure la connectivité réseau pour les autres services OpenStack sur plusieurs systèmes OpenStack et pour les instances de machine virtuelle. Dans Solaris, les services de virtualisation de réseau sont fournis par le biais de la fonctionnalité EVS (Commutateur virtuel élastique), qui fait office de point de contrôle unique pour la création, la configuration et la surveillance des commutateurs virtuels qui étendent plusieurs serveurs physiques. Les applications peuvent générer leur propre comportement en matière de gestion de priorité de trafic réseau sur l'ensemble du Cloud.

Neutron fournit une API permettant aux utilisateurs de demander et de configurer des réseaux virtuels de manière dynamique. Ces réseaux connectent des interfaces telles que des VNIC à partir d'instances VM Nova.

Pour plus d'informations sur les commutateurs virtuels élastiques, reportez-vous au [Chapitre 5, “ A propos des commutateurs virtuels élastiques ” du manuel “ Gestion de la virtualisation réseau et des ressources réseau dans Oracle Solaris 11.2 ”](#).

- Cinder

Le service de stockage en mode bloc Cinder offre une infrastructure permettant la gestion de volumes de stockage en mode bloc dans OpenStack. Cinder vous permet d'exposer les périphériques en mode bloc et de les connecter à des instances de machine virtuelle pour obtenir un stockage étendu, une meilleure performance, et l'intégration avec les plateformes de stockage de l'entreprise. Dans Solaris, Cinder utilise ZFS pour la gestion du stockage et iSCSI ou la Fibre Channel pour l'accès distant. ZFS fournit des services de données intégrées, comprenant les clichés, le chiffrement, et l'élimination de doublons. Un pilote Cinder est également disponible sur le dispositif de stockage ZFS.

Pour plus d'informations sur le système ZFS, reportez-vous à la section [“ Gestion des systèmes de fichiers ZFS dans OracleSolaris 11.2 ”](#). La documentation du dispositif de stockage ZFS est disponible sur <https://docs.oracle.com/en/storage/>.

- Swift

Le service de stockage d'objets Swift fournit des services de stockage d'objets pour les projets et utilisateurs OpenStack. SWIFT stocke et récupère des données arbitraires non-structurées à l'aide de ZFS, les données sont alors accessibles via l'API de RESTful.

- Glance

Le service de magasin d'images Glance stocke les images des disques des machines virtuelles utilisés pour déployer les instances de machine virtuelle. Dans Solaris les images Glance sont des archives d'ensemble. Les images peuvent être stockées à différents emplacements de systèmes de fichiers simples jusqu'à des systèmes de stockage d'objets, tels que OpenStack SWIFT. Glance dispose d'une API RESTful vous permettant d'interroger les métadonnées d'image, ainsi que d'extraire l'image.

Unified Archives permet un déploiement sécurisé, conforme, rapide et évolutif. La même archive d'ensemble peut également servir à déployer des systèmes à chaud ou virtuels Vous pouvez utiliser les archives d'ensemble avec le programme d'installation automatisée (AI) pour créer de nombreux systèmes.

Pour plus d'informations, reportez-vous à la section [“ Utilisation de Unified Archives pour la récupération du système et le clonage dans Oracle Solaris 11.2 ”](#) Une installation AI est une méthode d'installation automatique à l'aide d'une image AI sur un support ou un serveur. Pour plus d'informations, reportez-vous au [Chapitre 5, “ Installations automatisées initialisées à partir d'un média ” du manuel “ Installation des systèmes Oracle Solaris 11.2 ”](#).

- Horizon

Horizon est le tableau de bord OpenStack où vous pouvez gérer l'infrastructure du Cloud et l'infrastructure de calcul pour assurer la prise en charge plusieurs instances de machine virtuelle. Le tableau de bord fournit une interface utilisateur Web aux services

OpenStack. Pour obtenir un exemple, reportez-vous à [“Utilisation du tableau de bord OpenStack” à la page 28](#).

- Keystone

Le service d'identité Keystone fournit des services d'authentification et d'autorisation entre les utilisateurs, les administrateurs et les services OpenStack.

- Le moteur du service d'orchestration Heat permet aux développeurs d'automatiser l'implémentation d'une infrastructure OpenStack. Le moteur est défini par les modèles qui contiennent les informations de configuration et les opérations post-installation pour déployer une configuration personnalisée.

Chaque service OpenStack est représenté par au moins un service SMF (Service Management Facility). Par exemple, reportez-vous à la liste des services SMF dans le [Tableau 3-1, “Services SMF installés sur les noeuds de contrôleur, de réseau et de calcul”](#). SMF régule les services OpenStack, par exemple, l'exécution du redémarrage automatique des services en cas de panne ou le contrôle complet des dépendances des services pour un démarrage plus précis et efficace.

L'IPS (Image Packaging System) permet un déploiement de système OpenStack aisé, et une mise à niveau rapide protégée contre l'échec. Les environnements d'initialisation (BE) vous permettent de facilement conserver un environnement de sauvegarde lorsque vous mettez les systèmes OpenStack à jour. Chaque service OpenStack est fourni dans son propre package IPS permettant souplesse et minimisation de l'installation. Chaque package de service OpenStack fournit un utilisateur et un groupe uniques pour ce service OpenStack ainsi qu'un profil RBAC pour l'administration de ce service OpenStack.

## Utilisation de la présente documentation

Ce document aborde principalement les informations OpenStack qui diffèrent entre Solaris et les autres plateformes. Les fonctionnalités ayant le même principe opérationnel sur Solaris que sur les autres plateformes ainsi que les opérations qui s'effectuent de façon identique sur Solaris et sur les autres plateformes ne sont pas traitées dans ce document

Le [Chapitre 2, Installation d'une configuration d'évaluation](#) décrit comment installer rapidement OpenStack sur un système Solaris unique à des fins d'évaluation. L'installation complète est fournie dans une archive d'ensemble et la plus grande partie de la configuration est effectuée pour vous.

Le [Chapitre 3, Installation sur plusieurs systèmes pour une configuration Havana multinoeud OpenStack](#) décrit comment installer et configurer OpenStack sur trois systèmes Solaris : un noeud Contrôleur, un noeud Réseau et un noeud Calcul.

Le [Chapitre 5, Création d'instances de machine virtuelle](#) fournit des informations spécifiques à Solaris pour la création et l'utilisation d'une instance de machine virtuelle. Des tâches telles que la création de locataires et utilisateurs associés ne sont pas abordées ici, car elles sont identiques sur Solaris et sur les autres plateformes.

Pour obtenir des informations générales sur le OpenStack Havana sur lequel est basé OpenStack dans cette version de Oracle Solaris, reportez-vous aux ressources suivantes et bien d'autres sur le [site de documentation OpenStack](#).

- *Manuels de formation OpenStack*
- *Guide de l'utilisateur final*, comprenant "l'aide-mémoire de l'interface de ligne de commande OpenStack".
- *Guide de l'administrateur*
- *Référence de l'interface de ligne de commande*
- *Référence de configuration*
- *Guide de l'administrateur de Cloud*

Pour plus d'informations sur Solaris, reportez-vous à la [Bibliothèque d'informations Oracle Solaris 11.2](#). Pour plus d'informations sur OpenStack pour Solaris, reportez-vous à [OpenStack pour Oracle Solaris 11](#).

Dans la communauté OpenStack, des termes différents peuvent avoir la même signification. Par exemple, une machine virtuelle dans le Cloud peut être appelée un serveur, une instance ou une machine virtuelle de calcul. Une fonctionnalité OpenStack, comme le calcul ou la mise en réseau peut être appelée module, composant, ou service. Dans OpenStack, les termes *projet* et *locataire* sont employés de manière interchangeable. Ce document utilise les termes suivants

service	Un service OpenStack tel que Nova ou un service de calcul.
Service SMF	Un service OpenStack tel que <code>svc:/application/openstack/nova/nova-compute:default</code> . Les expressions telles que "activer un service" fait référence aux services SMF.
noeud	Système hébergeant des services OpenStack. Par exemple, un noeud de contrôleur héberge les services Keystone, Glance et Horizon.
projet	Dans les zones Oracle Solaris, un projet est un identifiant administratif au niveau du réseau pour une tâche associée. Dans ce document, cependant, le terme est employé selon la définition OpenStack, qui désigne un regroupement logique d'utilisateurs au sein du module de calcul. Le projet définit des quotas et l'accès aux images VM.
Instance de machine virtuelle	Une machine virtuelle dans le Cloud. Une instance de machine virtuelle est une machine virtuelle en cours d'exécution, ou une machine virtuelle dont l'état est connu, tel que suspendu, qui peut être utilisée comme un serveur matériel.
zone	Technologie employée dans Oracle Solaris pour virtualiser le système d'exploitation et fournir des environnements isolés et sécurisés où exécuter les applications. Le terme peut également faire référence à l'environnement virtualisé proprement dit. Dans Oracle Solaris, la virtualisation du calcul OpenStack repose sur la technologie des zones.

Pour obtenir des explications supplémentaires sur les termes OpenStack, reportez-vous au <http://docs.openstack.org/glossary/content/glossary.html>.

## Configuration requise pour l'installation d'OpenStack

Les systèmes Solaris sur lesquels vous installez OpenStack doivent répondre aux exigences suivantes

- **Système d'exploitation.** Vos systèmes OpenStack doivent exécuter Oracle Solaris 11.2. Pour l'installation de Oracle Solaris 11.2, reportez-vous à la section “ [Installation des systèmes Oracle Solaris 11.2](#) ”. Pour télécharger des images d'installation, reportez-vous à la page [Téléchargements Oracle Solaris 11.2](#). Pour une mise à niveau à partir d'une version antérieure de Oracle Solaris 11 vers Oracle Solaris 11.2, reportez-vous à “ [Mise à niveau vers Oracle Solaris 11.2](#) ” et Chapitre 4, “ [Mise à jour ou mise à niveau d'une image Oracle Solaris](#) ” du manuel “ [Ajout et mise à jour de logiciels dans Oracle Solaris 11.2](#) ”.
- **Matériel.** Pour s'assurer que vos systèmes prennent en charge Oracle Solaris 11.2, reportez-vous à [Oracle Solaris 11.2 System Requirements](#) ou “ [Configuration système requise pour l'installation d'Oracle Solaris 11.2](#) ” du manuel “ [Notes de version Oracle Solaris 11.2](#) ”. Jusqu'à 5 Go d'espace supplémentaire peuvent être requis pour installer OpenStack, en fonction des services OpenStack que vous souhaitez installer sur chaque noeud. Assurez-vous que vous disposez de suffisamment de CPU, de mémoire et d'espace disque sur vos noeuds. Calcul pour prendre en charge le nombre souhaité d'instances VM. Vos systèmes doivent disposer de 100-200 Go de stockage ZFS pour les images d'instances de machine virtuelle et pour la création des instances de machine virtuelle.
- **Prise en charge de la virtualisation.** Vos systèmes OpenStack doivent prendre en charge les zones de noyau. Les instances de machine virtuelles peuvent être des zones non globales ou des zones de noyau.

Pour vérifier si votre système prend en charge la virtualisation, saisissez la commande `virtinfo` dans une fenêtre de terminal. Le résultat de la commande devrait afficher les informations comme illustré dans l'exemple suivant

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

Pour la prise en charge des zones de noyau, votre système doit satisfaire les conditions requises supplémentaires suivantes

- 8 Go minimum de RAM physique.
- Réglage suffisant de l'ARC (ZFS Adaptive Replacement Cache) sur l'hôte afin d'éviter les erreurs de mémoire. Pour plus d'informations, reportez-vous à la section “ [Réglage du cache ZFS ARC de l'hôte sur un hôte de zone de noyau](#) ” du manuel “ [Création et utilisation des zones de noyau d'Oracle Solaris](#) ”.

---

**Remarque** - Les zones de noyau ne peuvent pas s'exécuter dans des invités Oracle VM Server pour x86 or sur Oracle VM VirtualBox.

---

Pour plus d'informations sur la configuration minimale requise par l'installation des zones de noyau, reportez-vous aux ressources suivantes

- [“ Configuration matérielle et logicielle pour les zones de noyau Oracle Solaris ” du manuel “ Création et utilisation des zones de noyau d’Oracle Solaris ”](#)
- Le fichier README se trouve dans la section "Oracle Solaris 11.2 OpenStack Unified Archive" de la page de téléchargement [Oracle Solaris 11.2 OpenStack Unified Archive](#)

Reportez-vous aux [“Limitations connues”](#) à la [page 121](#) pour plus d'informations pouvant avoir un impact sur l'installation. OpenStack.



## Installation d'une configuration d'évaluation

---

A des fins d'évaluation, vous pouvez installer OpenStack sur un système Oracle Solaris unique. Ce type d'installation OpenStack est également qualifiée d'installation mononoeud. La version d'OpenStack peut être Havana ou Juno, suivant l'application Oracle Solaris sur le système.

Pour la description de l'installation d'un multinoeud OpenStack, reportez-vous au [Chapitre 3, Installation sur plusieurs systèmes pour une configuration Havana multinoeud OpenStack](#) ou [Chapitre 4, Installation sur plusieurs systèmes pour une configuration Juno multinoeud OpenStack](#).

Ce chapitre décrit l'installation d'un noeud unique à l'aide d'une archive d'ensemble, et la procédure pour créer des instances de machine virtuelle à l'aide d'une interface du navigateur. Ce chapitre comprend les sections suivantes :

- “Déploiement d'OpenStack Unified Archives” à la page 19
- “Utilisation du tableau de bord OpenStack” à la page 28

---

**Remarque** - Le contenu de ce chapitre concerne les versions Havana et Juno, sauf indication contraire.

---

## Déploiement d'OpenStack Unified Archives

La page [Unified Archives download page](#) offre la possibilité de télécharger uniquement l'archive Oracle Solaris ou l'archive Oracle Solaris avec OpenStack.

L'archive Oracle Solaris avec OpenStack installe le système d'exploitation et tous les services OpenStack répertoriés dans “[Intégration d'OpenStack à Oracle Solaris](#)” à la page 11.

L'archive d'ensemble Oracle Solaris OpenStack offre les fonctionnalités suivantes :

- Des services OpenStack le plus souvent préconfigurés, allégeant la configuration vous restant à effectuer
- Des scripts permettant l'automatisation de la configuration restant à effectuer
- Des modèles de systèmes Solaris

- Un magasin d'images Glance dans lequel deux images sont préchargées : une zone non globale et une zone de noyau

Vous pouvez déployer l'archive d'ensemble OpenStack sur un système à chaud ou sur une zone de noyau. Reportez-vous à “[Installation du système unique](#)” à la page 21 pour plus d'options.

## Téléchargement du fichier image

Utilisez la procédure suivante pour télécharger un fichier image unique qui fournit à la fois le système d'exploitation Oracle Solaris et OpenStack.

### ▼ Téléchargement du fichier image

1. Dans un navigateur Internet, accédez à la page de téléchargement [Unified Archives](#).
2. Lisez le contrat de licence et cliquez sur le bouton **Accepter**.
3. Téléchargez le fichier correspondant au type d'installation que vous voulez effectuer.
  - **Téléchargez le fichier Unified Archives correspondant à votre architecture système.**

Utilisez ce fichier afin d'effectuer l'un des types d'installation suivants

    - Utilisez ce fichier `.uar` en tant qu'argument à la commande `zoneadm install` pour installer directement une zone de noyau.
    - Référencez ce fichier `.uar` dans un manifeste d'installation AI pour effectuer une installation AI à l'aide d'un service d'installation AI.
    - Utilisez ce fichier `.uar` afin de créer un média AI amorçable.
  - **Téléchargez le fichier d'image d'initialisation USB correspondant à l'architecture de votre système.**

Utilisez ce fichier `usb` pour installer l'archive d'ensemble à partir du média d'initialisation.
4. Vérifiez l'intégrité du fichier téléchargé.

Cliquez sur le lien des sommes de contrôle MD5. Exécutez la commande `digest` et comparez la sortie avec la somme de contrôle appropriée à partir du fichier de sommes de contrôle

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

## Installation du système unique

Cette section décrit la procédure d'installation de la configuration d'évaluation OpenStack en système unique. Les trois premières méthodes décrites effectuent l'installation à chaud. La quatrième méthode effectue l'installation dans une zone de noyau.

- [“Installation à l'aide du fichier USB téléchargé” à la page 21](#)
- [“Installation en utilisant le fichier Unified Archives et un service d'installation AI” à la page 22](#)
- [“Installation en utilisant le média amorçable AI créé à partir du fichier d'archive d'ensemble” à la page 23](#)
- [“Installation dans une zone de noyau à l'aide du fichier Havana Unified Archives” à la page 24](#)

### ▼ Installation à l'aide du fichier USB téléchargé

Cette procédure décrit l'installation d'Oracle Solaris et d'OpenStack à chaud avec le fichier USB téléchargé. Cette méthode ne nécessite pas de serveur AI, il s'agit de la méthode d'installation à chaud la plus directe.

1. **Téléchargez le fichier USB correspondant à votre architecture système cible.**  
Reportez-vous à [“Téléchargement du fichier image” à la page 20](#).
2. **Transférez le fichier USB sur un lecteur flash USB.**
  - **Utilisation de la commande `usbcopy`.**  
Si vous avez accès à un système Oracle Solaris 11.2, servez-vous de l'utilitaire `usbcopy`. Reportez-vous à la page de manuel [`usbcopy\(1M\)`](#).  
  
Vous devez utiliser la commande `usbcopy` à partir de Oracle Solaris 11.2. Vous ne pouvez pas utiliser `usbcopy` à partir d'une version antérieure de Solaris.
  - **Utilisation de la commande `dd`.**  
Si vous n'avez pas accès à un système Oracle Solaris 11.2, vous pouvez exécuter la commande `dd`.  
  
Soyez extrêmement attentif à identifier correctement le disque approprié (disque flash) lors de l'utilisation de `dd`.
    - **Sur Oracle Solaris 11 :**
      - a **Désactivez le service HAL.**

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

**b Insérez le lecteur flash et localisez l'appareil approprié.**

```
# rmformat
```

**c Copiez l'image.**

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

**d Activez le service HAL.**

```
# svcadm enable hal
```

■ **Sur Linux :**

**a Insérez le lecteur flash et localisez l'appareil approprié.**

```
# dmesg | tail
```

**b Copiez l'image.**

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

■ **Sur MacOSX**

**a Identifiez le lecteur, /dev/diskN, où N est le numéro du disque.**

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

**b Copiez l'image.**

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

**3. Insérez le lecteur flash USB dans le système et initialisez-le à partir du lecteur USB.**

L'outil System Configuration Interactive (SCI) s'affiche. Si vous ne le voyez pas, appuyez sur la touche Entrée ou appuyez sur Ctrl-L pour le recréer l'écran.

## ▼ **Installation en utilisant le fichier Unified Archives et un service d'installation AI**

Cette procédure décrit l'installation d'Oracle Solaris et d'OpenStack à chaud à l'aide du fichier Unified Archives et de AI.

**1. Téléchargez le fichier d'archive d'ensemble correspondant à votre architecture système cible.**

Reportez-vous à [“Téléchargement du fichier image” à la page 20.](#)

**2. Créez un manifeste AI.**

Sur votre serveur d'installation AI Oracle Solaris, copiez et modifiez le fichier `/usr/share/auto_install/manifest/default_archive.xml` en fonction de votre configuration AI. Dans la section `ARCHIVE software`, indiquez l'emplacement du fichier `.uar` téléchargé.

**3. Configurez un service d'installation AI.**

Configurez un service d'installation AI avec le manifeste AI de l'étape précédente. Reportez-vous à la [Partie III, “ Installation à l'aide d'un serveur d'installation ” du manuel “ Installation des systèmes Oracle Solaris 11.2 ”.](#)

**4. Initialisez le système sur le réseau.**

```
ok boot net -install
```

**5. Une fois l'installation terminée, réinitialisez le système.**

L'outil SCI Tool s'affiche pour configurer le système. Si vous ne le voyez pas, appuyez sur la touche Entrée ou appuyez sur Ctrl-L pour le recréer l'écran.

**▼ Installation en utilisant le média amorçable AI créé à partir du fichier d'archive d'ensemble**

Cette procédure décrit l'installation de Oracle Solaris 11.2 et OpenStack à chaud en créant une image AI amorçable. Une image USB amorçable est créée à partir du fichier d'archive d'ensemble téléchargé. Reportez-vous au [Chapitre 5, “ Installations automatisées initialisées à partir d'un média ” du manuel “ Installation des systèmes Oracle Solaris 11.2 ”](#) pour plus d'informations sur cette méthode..

**1. Téléchargez le fichier d'archive d'ensemble correspondant à votre architecture système cible.**

Reportez-vous à [“Téléchargement du fichier image” à la page 20.](#)

**2. Créez l'USB AI à partir du fichier Unified Archives.**

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

où `workdir` correspond à l'emplacement où vous avez téléchargé le fichier Unified Archive. Le fichier AI USB sera créé dans le même répertoire.

### 3. Transférez le fichier USB sur un lecteur flash USB.

- Si vous avez téléchargé le fichier USB sur un système Oracle Solaris 11.2, utilisez la télécommande `usbcopy`.
- Si vous avez téléchargé le fichier USB sur un système dont le système d'exploitation Oracle Solaris est antérieur à Oracle Solaris 11.2, utilisez la commande `dd` comme suit :

1. Désactivez le service HAL.

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

2. Insérez le lecteur flash et localisez l'appareil approprié.

```
# rmformat
```

3. Copiez l'image sur le lecteur flash.

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

4. Activez le service HAL.

```
# svcadm enable hal
```

### 4. Vérifiez le manifeste AI par défaut.

Vous pouvez utiliser le manifeste par défaut ou créer un manifeste personnalisé. Si vous créez un manifeste personnalisé, stockez-le dans un emplacement accessible par le système en cours d'installation.

### 5. Insérez le lecteur flash USB dans le système et effectuez l'initialisation à partir du lecteur USB.

Vous êtes invité à utiliser le manifeste AI par défaut ou à indiquer l'emplacement du manifeste personnalisé.

L'outil SCI Tool s'affiche pour configurer le système. Si vous ne le voyez pas, appuyez sur la touche Entrée ou appuyez sur Ctrl-L pour le recréer l'écran.

## ▼ Installation dans une zone de noyau à l'aide du fichier Havana Unified Archives

Cette procédure décrit l'installation d'Oracle Solaris 11.2 et de Havana OpenStack directement dans une zone de noyau en utilisant le fichier Unified Archives téléchargé.

#### Avant de commencer

Assurez-vous que le système qui hébergera la zone de noyau est conforme aux exigences relatives à la virtualisation indiquées dans [“Configuration requise pour l'installation d'OpenStack”](#) à la page 16

### 1. Téléchargez le fichier d'archive d'ensemble correspondant à votre architecture système cible.

Reportez-vous à [“Téléchargement du fichier image”](#) à la page 20.

## 2. Créez une zone de noyau.

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

## 3. Configurez la zone de noyau.

Assurez-vous que votre zone de noyau dispose de suffisamment de CPU virtuelles, de RAM, d'espace disque et d'adresses MAC. Les zones non globales créées à l'intérieur de la zone de noyau pourront consommer ces adresses MAC supplémentaires automatiquement.

L'exemple suivant permet de configurer la zone avec 8 CPU virtuelles, une limite de 8 Go de mémoire, et une affectation d'adresses MAC automatique. Reportez-vous à la page de manuel [zonecfg\(1M\)](#) pour obtenir les descriptions des ressources configurables.

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

## 4. Vérifiez votre configuration.

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

## 5. Installez la zone de noyau.

L'exemple suivant illustre l'installation de la version x86 du fichier `.uar`.

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

où `uar-file` est l'archive Oracle Solaris 11.2 avec Havana OpenStack.

## 6. Amorcez la zone.

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

## 7. Connectez-vous à la console de la zone pour terminer la configuration.

```
# zlogin -C OpenStackKZ
```

L'outil SCI Tool s'affiche pour configurer le système. Si vous ne le voyez pas, appuyez sur la touche Entrée ou appuyez sur Ctrl-L pour le recréer l'écran.

### 8. Assurez-vous que la zone de noyau dispose d'une adresse IP.

L'archive d'ensemble s'attend à ce qu'un serveur DHCP affecte une adresse IP à la zone de noyau. Si vous utilisez DHCP, assurez-vous qu'une adresse MAC est affectée à la zone de noyau. Si vous avez sélectionné Automatique sur la page réseau de l'outil SCI Tool à l'étape précédente, l'adresse MAC doit être affectée. Si vous n'utilisez pas DHCP, veillez à attribuer une adresse IPv4 à la zone de noyau.

L'adresse IP est importante car il n'existe actuellement aucun référentiel système pour les zones de noyau. L'adresse IP permet à la zone de noyau de se connecter à un référentiel de package IPS, si nécessaire.

## Configuration du commutateur virtuel élastique

Après que l'installation a été exécutée et que le système a été réinitialisé, vous devez configurer le commutateur virtuel élastique (EVS). EVS vous permet de créer, de configurer et de contrôler les commutateurs virtuels qui étendent des serveurs physiques et connectent les machines virtuelles faisant partie d'un même EVS. Pour plus d'informations sur EVS, reportez-vous au [Chapitre 5, “ A propos des commutateurs virtuels élastiques ”](#) du manuel “ [Gestion de la virtualisation réseau et des ressources réseau dans Oracle Solaris 11.2](#) ”.

Un script est fourni pour automatiser la configuration EVS. Exécutez le script à l'aide de la commande suivante

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

Ce script réalise la configuration suivante

- Création de clés SSH (Secure Shell) pour tous les utilisateurs requis, tels que l'utilisateur root, l'utilisateur evsuser et les utilisateurs UNIX neutron, et ajout de ces clés publiques au fichier `/var/user/evsuser/.ssh/authorized_keys` pour l'utilisateur evsuser.
- Configuration EVS.
- Activation des services SMF `neutron-server:default` et `neutron-dhcp-agent:default`.
- Indication de la technologie de LAN virtuel à utiliser (VLAN ou VXLAN) et les Id ou segments correspondants.

## Déploiement de Juno OpenStack Unified Archives

Pour déployer l'archive Juno OpenStack, vous devez d'abord vous assurer que le système utilise la version Oracle Solaris 11.2 SRU 10. Les versions antérieures d'Oracle Solaris 11.2 ne prennent pas en charge Juno.

## ▼ Installation de Juno OpenStack dans une zone de noyau à l'aide du fichier Unified Archives

Avant de commencer

Le système qui héberge la zone de noyau doit satisfaire aux conditions de virtualisation indiquées dans [“Configuration requise pour l'installation d'OpenStack” à la page 16](#). Assurez-vous également que le fichier OpenStack Unified Archives a été téléchargé dans le système. Reportez-vous à [“Téléchargement du fichier image” à la page 20](#).

### 1. Créez une zone de noyau.

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

Cette étape crée une zone de noyau à partir du modèle Oracle Solaris appelé SYSsolaris-kz.

### 2. Configurez la zone de noyau.

L'exemple suivant permet de configurer la zone avec 8 CPU virtuelles et une limite de 12 Go de mémoire. Reportez-vous à la page de manuel [zonecfg\(1M\)](#) pour obtenir les descriptions des ressources configurables.

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

### 3. (Facultatif) Vérifiez votre configuration.

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

### 4. Installez la zone de noyau.

Dans l'exemple suivant, 50 Go d'espace disque sont utilisés pour la zone de noyau afin de garantir un espace suffisant pour créer des volumes pour les instances de machine virtuelle.

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

où *archive-path* correspond au chemin d'accès de l'emplacement de l'archive d'ensemble Juno OpenStack téléchargée.

### 5. Amorcez la zone.

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

### 6. Connectez-vous à la console de la zone pour terminer la configuration.

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

L'outil SCI Tool s'affiche pour configurer le système. Si vous ne le voyez pas, appuyez sur la touche Entrée ou appuyez sur Ctrl-L pour le recréer l'écran.

## Utilisation du tableau de bord OpenStack

Une fois les tâches de configuration d'installation et suivant l'installation de OpenStack terminées, connectez-vous au tableau de bord OpenStack pour visualiser les ressources disponibles et pour créer et initialiser une instance de machine virtuelle.

### ▼ Accès au tableau de bord OpenStack

1. **Connectez-vous à n'importe quel système pouvant se connecter au système OpenStack.**
2. **Configurez votre navigateur.**
  - a. **Activez JavaScript.**
  - b. **Conservez les cookies.**
3. **Dans le champ d'emplacement ou d'adresse du navigateur, entrez l'emplacement suivant**

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

Le *system* est soit le nom soit l'adresse IP du système OpenStack où l'archive d'ensemble OpenStack a été installée, et qui exécute le service Horizon OpenStack sous le serveur web Apache.

Si vous avez installé l'archive d'ensemble sur une zone de noyau, le système OpenStack est la zone de noyau, et *system* est le nom ou l'adresse IP de la zone de noyau.

4. **Indiquez les informations suivantes sur l'écran de connexion**
  - Nom utilisateur : admin
  - Mot de passe : secreta

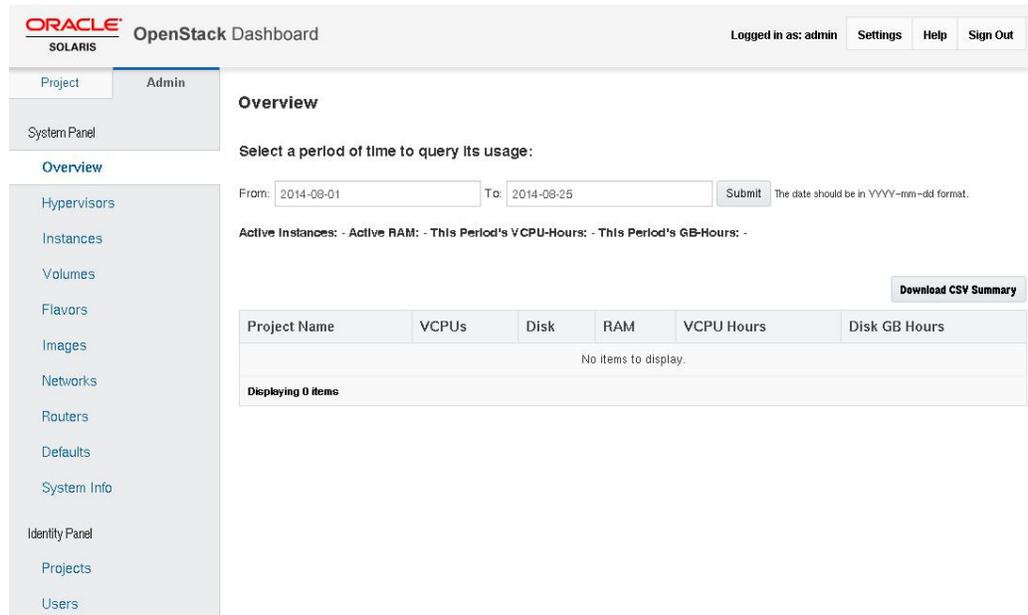
Les fonctions disponibles dans le tableau de bord dépendent des autorisations d'accès, ou rôles, de l'utilisateur sous lequel vous êtes connecté.

## Exploration du tableau de bord

Lorsque vous vous connectez en tant qu'administrateur de Cloud, la fenêtre comporte un panneau sur le côté gauche de l'écran avec deux onglets : projet et admin. Le panneau admin est l'affichage par défaut de l'administrateur du Cloud. Les choix proposés sur le panneau de l'administrateur permettent la fonctionnalité suivante

- Une vue globale des instances Nova et des volumes Cinder utilisés à l'intérieur du Cloud
- La capacité à afficher et modifier les définitions de variantes qui définissent les caractéristiques de l'instance de machine virtuelle telles que :
  - Le nombre de CPU virtuelles
  - La quantité de mémoire
  - L'espace disque affecté
  - La marque de la zone sous-jacente Solaris : `solaris` pour les zones non globales, et `solaris-kz` pour les zones de noyau.
- La possibilité de créer des routeurs et des réseaux virtuels en vue de leur utilisation par les administrateurs du cloud
- La possibilité d'afficher et de modifier des projets en regroupant et en isolant l'appartenance des ressources de calcul virtuelles.
- La possibilité d'afficher et de modifier les utilisateurs, qui sont les personnes ou les services qui utilisent les ressources du Cloud.

FIGURE 2-1 Fenêtre de présentation d'administration du tableau de bord OpenStack

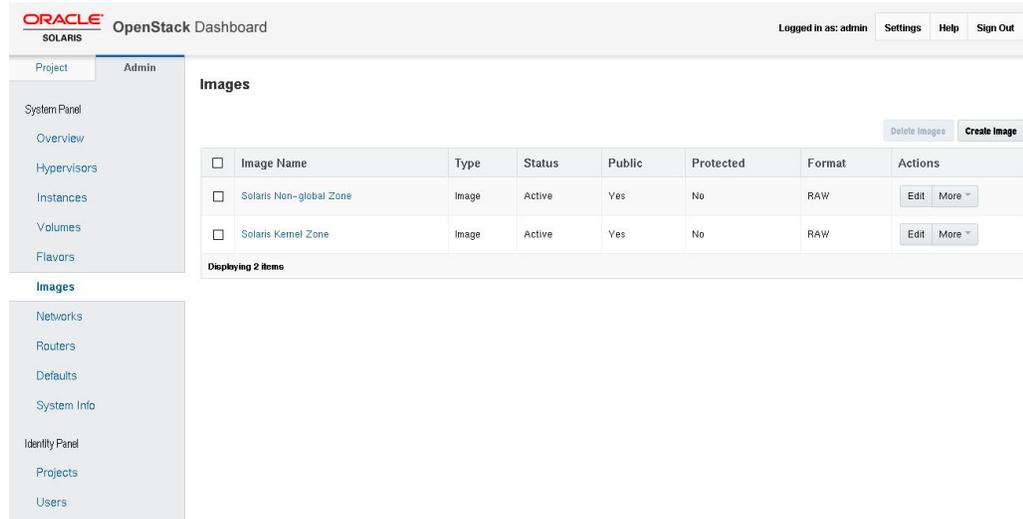


Si vous avez installé OpenStack conformément à [“Installation du système unique” à la page 21](#), votre système OpenStack est préconfiguré avec les ressources suivantes

- Deux images : Zone non globale Solaris et Zone de noyau Solaris
- Deux projets : demo et service
- Dix variantes

La figure suivante présente les images.

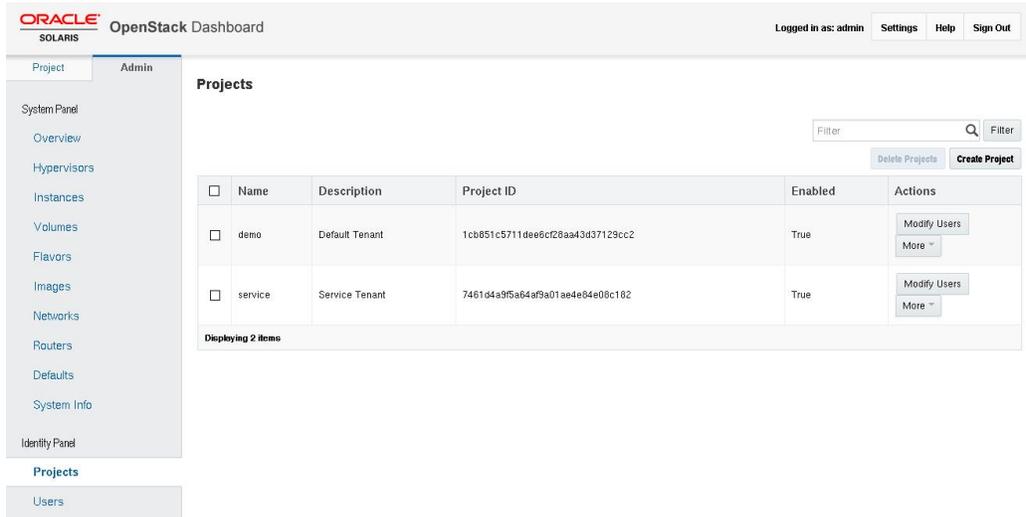
FIGURE 2-2 Ecran d'images dans le tableau de bord OpenStack



La figure suivante présente les projets, aussi appelés locataires.

- Le locataire *demo* est le locataire par défaut. Par défaut, le locataire *demo* est créé avec un utilisateur unique, *admin*, en tant que membre.
- Le locataire *service* est un locataire utilisé par l'administrateur du cloud pour créer des ressources qui seront partagées entre plusieurs locataires. Par exemple, dans les exemples et les scénarios présentés dans le document, le routeur Neutron est créé dans le locataire *service* de sorte que le routeur soit partagé par tous les locataires. Vous ne devez pas utiliser le locataire *service* à toute autre fin dans votre configuration OpenStack. Les services OpenStack communiquent les uns avec les autres au moyen d'utilisateurs spécifiques au service, qui ont tous le rôle *admin* dans le cadre du locataire *service*.

FIGURE 2-3 Ecran de projets dans le tableau de bord OpenStack



La figure suivante affiche les variantes. Reportez-vous à [“Affichage des informations sur les variantes”](#) à la page 112 pour une liste de texte des variantes.

FIGURE 2-4 Ecran des variantes dans le tableau de bord OpenStack

ORACLE SOLARIS OpenStack Dashboard

Logged in as: admin Settings Help Sign Out

Project Admin

System Panel

Overview

Hypervisors

Instances

Volumes

Flavors

Images

Networks

Routers

Defaults

System Info

Identity Panel

Projects

Users

Flavors

Filter

Delete Flavors Create Flavor

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

Displaying 10 items

## Création et initialisation d'une instance de machine virtuelle

Utiliser le panneau Projet dans le tableau de bord pour créer une instance de machine virtuelle.

### ▼ Création d'une instance de machine virtuelle avec le tableau de bord

Avant de commencer

Assurez-vous d'avoir une paire de clés SSH. Reportez-vous à [“Création d'une paire de clés SSH”](#) à la page 39.

Assurez-vous d'avoir un réseau externe défini. Voir [“Configuration de l'agent Neutron L3”](#) à la page 66.

1. Cliquez sur l'onglet **Projet** dans le panneau situé sur le côté gauche du tableau de bord.

Le projet en cours d'utilisation par l'utilisateur est affiché en haut de ce panneau. Par défaut, l'utilisateur admin dans cette configuration OpenStack utilise le projet demo.

**2. Dans la section Gestion de calcul du panneau, cliquez sur Instances.**

**3. Sur le côté droit du panneau Instances, cliquez sur le bouton Lancer.**

La boîte de dialogue Lancer l'instance suivante s'affiche pour que vous indiquiez le nom de la nouvelle instance de machine virtuelle et que vous sélectionniez la variante et le type d'image de l'instance.

FIGURE 2-5 Boîte de dialogue Lancer l'instance

**Launch Instance**
✕

**Details**

Access & Security

Networking

**Availability Zone**

nova
▼

**Instance Name**

**Flavor**

Oracle Solaris kernel zone - tiny
▼

**Instance Count**

1
▼

**Instance Boot Source**

--- Select source ---
▼

Specify the details for launching an instance.

The chart below shows the resources used by this project

**Flavor Details**

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

**Project Limits**

**Number of Instances**

**Number of VCPUs**

**Total RAM**

Cancel

Launch

4. Dans le champ Nom d'instance, entrez le nom de la nouvelle instance de machine virtuelle.
5. Sélectionnez une variante à partir de la liste déroulante de variantes.

Si ce système OpenStack est une zone de noyau et non un système à chaud, vous devez alors sélectionner une variante de zone non globale.

**6. Sous Source d'initialisation de l'instance, sélectionnez Initialisation à partir de l'image.**

Une sélection de noms d'images s'affiche. Sélectionner l'image qui est le même type de zone que la variante sélectionnée. La variante et l'image doivent être toutes deux soit une zone non globale `solaris`, soit une zone de noyau `solaris-kz`. Si ce système OpenStack est une zone de noyau et non un système à chaud, vous devez alors sélectionner une image de zone non globale.

**FIGURE 2-6** Boîte de dialogue Lancer l'instance avec les sélections

**Launch Instance**
✕

**Details**

Access & Security

Networking

**Availability Zone**

nova

**Instance Name**

VM1

**Flavor**

Oracle Solaris non-global zone - tiny

**Instance Count**

1

**Instance Boot Source**

Boot from image

**Image Name**

Solaris Non-global Zone (1.2 GB)

Specify the details for launching an instance.

The chart below shows the resources used by this project.

**Flavor Details**

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

**Project Limits**

**Number of Instances**

**Number of VCPUs**

**Total RAM**

Cancel

Launch

7. **Cliquez sur l'onglet Accès et Sécurité dans la boîte de dialogue.**  
Sélectionnez la paire de clés SSH à installer dans la nouvelle instance.
8. **Cliquez sur l'onglet Mise en réseau dans la boîte de dialogue.**  
Sélectionner le réseau auquel la nouvelle instance de machine virtuelle devra être associée.

Chapitre 2. Installation d'une configuration d'évaluation

37

**9. Cliquez sur le bouton Lancer dans la partie inférieure de la boîte de dialogue.**

La nouvelle instance de machine virtuelle est créée, installée et amorcée.

Le temps requis pour rendre une nouvelle instance disponible dépend de différents facteurs, incluant la taille de l'image, les ressources à la disposition de la variante, et l'emplacement où OpenStack a placé le système de fichiers root de la nouvelle instance de machine virtuelle.

**10. Associez l'adresse IP flottante à la nouvelle instance de machine virtuelle.**

Vous pouvez effectuer les opérations suivantes pendant que la nouvelle instance de machine virtuelle est en cours d'installation. L'instance de machine virtuelle doit avoir une adresse IP flottante associée pour que les utilisateurs puissent s'y connecter.

**a. Cliquez sur le bouton Associer IP flottante dans la colonne Actions.**

La boîte de dialogue Gérer les associations d'IP flottantes s'ouvre.

**b. Sélectionnez une adresse dans le menu déroulant d'adresses IP.**

Si le champ d'adresses IP affiche un message indiquant qu'aucune adresse IP n'est disponible, cliquez sur le bouton +. Reportez-vous à [“Association d'une adresse IP flottante à un projet”](#) à la page 39.

**c. Sélectionnez un port associé.**

La liste de ports présente les adresses IP fixes des instances de machine virtuelle.

**d. Cliquez sur le bouton Associer dans la partie inférieure de la boîte de dialogue.**

- Étapes suivantes**
- Cliquez sur Instances puis sur le nom de l'instance pour afficher les informations détaillées de l'instance et le journal de console de l'instance. Rechargez la page pour voir les mises à jour du journal.
  - Cliquez sur Volumes pour afficher les volumes Cinder qui ont été créés.
  - Cliquez sur Topologie réseau pour obtenir une représentation du réseau du Cloud, y compris tous les segments de sous-réseau, les routeurs virtuels et les instances actives.
  - Cliquez sur Images & Clichés pour visualiser les archives d'ensemble qui ont été téléchargés dans le magasin d'images Glance.
  - Lorsque l'installation d'une nouvelle instance de machine virtuelle est terminée et est en statut Actif, connectez-vous à l'instance. La commande suivante se connecte à la zone en tant que root à l'aide de la clé générée à l'étape 7 et des adresses IP flottantes de l'étape 10

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

## ▼ Création d'une paire de clés SSH

1. Cliquez sur l'onglet **Projet** dans le panneau situé sur le côté gauche du tableau de bord.
2. Dans la section **Gestion de calcul** du panneau, cliquez sur **Accès et Sécurité**.
3. Cliquez sur l'onglet **Paires de clés**.
4. Cliquez sur le bouton **Créer paire de clés**.  
La boîte de dialogue **Créer paire de clés** s'ouvre.
5. Indiquez le nom de la paire de clés dans le champ **Nom de la paire de clés** dans la boîte de dialogue.
6. Cliquez sur le bouton **Créer paire de clés** dans la boîte de dialogue.  
La nouvelle paire de clés doit se télécharger automatiquement.  
Si le nouvel paire de clés n'est pas téléchargée automatiquement, cliquez sur le lien fourni **Télécharger paire de clés**.  
La nouvelle paire de clés doit être répertoriée dans l'onglet **Paire de clés** du panneau **Sécurité & Accès**.

## ▼ Création d'un réseau pour un locataire

1. Cliquez sur l'onglet **Projet** dans le panneau situé sur le côté gauche du tableau de bord.
2. Dans la section **Gestion du réseau** du panneau, cliquez sur **Réseaux**.
3. Cliquez sur le bouton **Créer un réseau**.  
La boîte de dialogue correspondante apparaît.
4. Indiquez le nom pour le réseau dans le champ **Nom** de l'onglet **Réseau**.
5. Fournissez les informations demandées dans les onglets **Sous-réseau** et **Détails du sous-réseau**.
6. Cliquez sur le bouton **Créer** dans la partie inférieure de la boîte de dialogue.  
Le nouveau réseau et sous-réseau doivent être répertoriés dans le panneau **Réseaux**.

## ▼ Association d'une adresse IP flottante à un projet

1. Cliquez sur l'onglet **Projet** dans le panneau situé sur le côté gauche du tableau de bord.

2. **Dans la section Gestion de calcul du panneau, cliquez sur Accès et Sécurité.**
3. **Cliquez sur l'onglet IP flottantes.**
4. **Cliquez sur le bouton Allouer IP au projet.**  
La boîte de dialogue correspondante apparaît.
5. **Dans le menu déroulant de la boîte de dialogue, sélectionnez un pool à partir duquel allouer les IP flottantes.**
6. **Cliquez sur le bouton Allouer IP.**

## Installation sur plusieurs systèmes pour une configuration Havana multinoeud OpenStack

---

Ce chapitre décrit l'installation d'une configuration OpenStack multinoeud. Chaque Cloud n'a besoin que d'une instance de tableau de bord, d'un magasin d'images et d'un service d'identité. Chaque Cloud peut avoir n'importe quel nombre d'instances de stockage et de calcul. Dans un environnement de production, ces services sont configurés sur plusieurs noeuds. Évaluez chaque composant par rapport à vos besoins pour un déploiement Cloud particulier afin de déterminer si ce composant doit être installé sur un noeud distinct, ainsi que le nombre de noeuds de chaque type dont vous avez besoin.

---

**Remarque** - Ce chapitre concerne une configuration OpenStack Havana. La version Juno d'OpenStack est prise en charge uniquement à partir de la version Oracle Solaris 11.2 SRU10.

- Pour plus d'informations sur l'installation de la version d'Oracle Solaris 11.2 SRU10 et de la mise à niveau d'une configuration Havana existante vers Juno, reportez-vous au document [Havana to Juno: OpenStack Upgrade Procedures](#).
- Si vous exécutez la version d'Oracle Solaris 11.2 SRU10 et souhaitez effectuer une nouvelle configuration de Juno, reportez-vous au [Chapitre 4, Installation sur plusieurs systèmes pour une configuration Juno multinoeud OpenStack](#)

---

Ce chapitre décrit une architecture déployée sur des systèmes physiques distincts. Pour partitionner un serveur Oracle SPARC unique et configurer OpenStack en multinoeud sur le serveur exécutant OVM Server for SPARC (LDoms), reportez-vous à [Multi-node Solaris 11.2 OpenStack on SPARC Servers](#).

## Présentation de l'architecture à trois noeuds

L'architecture décrite dans ce chapitre est déployée sur les trois systèmes suivants :

- Noeud de contrôleur. Le noeud de contrôleur est le noeud où la plus grande partie des services partagés OpenStack et d'autres outils sont exécutés. Le noeud de contrôleur fournit les API, la planification et d'autres services partagés pour le Cloud. Le noeud de contrôleur

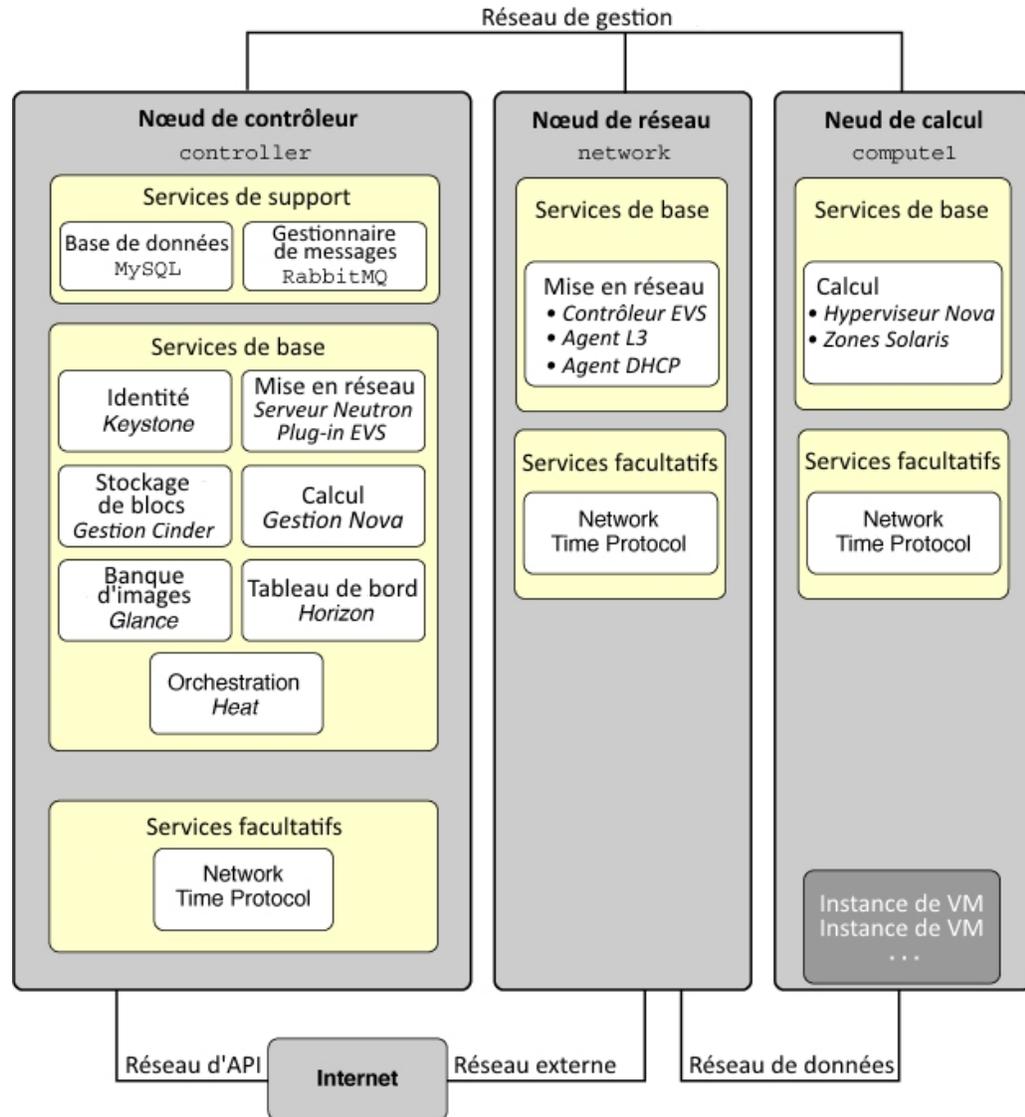
contient le tableau de bord, le magasin d'images et le service d'identité. En outre, le service de gestion des calculs Nova ainsi que le serveur Neutron sont également configurés dans ce noeud.

- Noeud de réseau. Le noeud de réseau fournit les services de mise en réseau et la mise en réseau virtuelle aux instances Nova au moyen de la couche 3 Neutron et des services de réseau DHCP.
- Noeud de calcul. Le noeud de calcul est l'endroit où les instances de machine virtuelle (instances de calcul Nova) sont installées. Les instances de machine virtuelle utilisent les cibles iSCSI fournies par le service de volume Cinder.

Dans cette architecture, les trois noeuds partagent un sous-réseau commun appelé sous-réseau de gestion. Le noeud de contrôleur et chaque noeud de calcul partagent un sous-réseau distinct, appelé sous-réseau de données. Chaque système est lié au réseau de gestion par son interface physique `net0`. Le noeud de réseau et le noeud de calcul sont liés au réseau de données par leurs interfaces physiques `net1`.

La figure suivante illustre une vue de haut niveau de l'architecture décrite dans ce chapitre.

FIGURE 3-1 Architecture de référence de la configuration à trois noeuds



Le tableau ci-dessous indique les services SMF associés à OpenStack installés sur chaque noeud. La liste indique la plus petite partie du nom de chaque partie du service SMF que vous pouvez utiliser avec des commandes telles que `svcadm`. Le nom de l'instance du service SMF n'est listé que si le nom risque d'être ambigu sans le nom de l'instance.

**TABLEAU 3-1** Services SMF installés sur les noeuds de contrôleur, de réseau et de calcul

Noeud de contrôleur.	Noeud de réseau	Noeud de calcul
mysql	neutron-dhcp-agent	nova-compute
rabbitmq	neutron-l3-agent	ntp
keystone	evs-controller	
cinder-api	ntp	
cinder-db		
cinder-db		
cinder-scheduler		
cinder-volume:default		
cinder-volume:setup		
glance-api		
glance-db		
glance-registry		
glance-scrubber		
neutron-server		
evs		
nova-api-ec2		
nova-api-osapi-compute		
nova-cert		
nova-conductor		
nova-objectstore		
nova-scheduler		
ntp		
heat-api		
heat-db		
heat-api-cfn		
heat-api-cloudwatch		
heat-engine		

Cet exemple d'architecture ne présente pas le service de stockage d'objets Swift. Pour obtenir des informations générales sur la configuration de Swift, voir les informations sur le site de la communauté OpenStack telles que la [Référence de configuration OpenStack](#). Pour plus d'informations sur la procédure de configuration des services SWIFT sur les systèmes Solaris, et pour plus d'informations sur OpenStack sur Solaris, reportez vous à [OpenStack pour Oracle Solaris 11](#).

Pour une liste des paramètres de configuration utiles pour les déploiements de {ENT OST} sur les systèmes Oracle Solaris, reportez-vous à « Paramètres communs de configuration pour OpenStack dans la rubrique [Démarrage de OpenStack sur Oracle Solaris 11.2](#).

Pour préparer l'implémentation de l'exemple de configuration OpenStack à trois noeuds, assurez-vous que vous disposez des informations suivantes

- Adresse IP et nom d'hôte du noeud de contrôleur.
- Adresse IP et nom d'hôte du noeud de réseau.
- Adresse IP et nom d'hôte du noeud de calcul.
- Mots de passe des différents utilisateurs du service selon les besoins.

Pour l'exemple de configuration, les noms des trois noeuds sont `controller`, `network` et `compute1`.

## Configuration du noeud de contrôleur

Le noeud de contrôleur comporte un service de tableau de bord, un magasin d'images et un service d'identité. Ce noeud inclut également MySQL, RabbitMQ, ainsi que des services de calcul, de stockage de bloc et de mise en réseau.

Pour une meilleure gestion de l'utilisation de la mémoire entre ZFS et les applications dans Oracle Solaris 11, définissez le paramètre `usr_reserve_hint_pct` sur le noeud, comme indiqué dans l'exemple suivant :

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

où *site* peut correspondre à votre entreprise.

Vous devez également définir ce paramètre sur les différents noeuds OpenStack.

Pour plus d'informations sur ce paramètre, connectez-vous à votre compte MOS à l'adresse <https://support.oracle.com> et consultez le Document 1663862.1, *Memory Management Between ZFS and Applications in Oracle Solaris 11.2*.

La communication entre les services OpenStack est effectuée par AMQP (Advanced Message Queuing Protocol). Sous Solaris, AMQP est implémenté par RabbitMQ. Le service RabbitMQ

est obligatoire. Un seul noeud du Cloud est généralement configuré pour exécuter RabbitMQ. Dans cette architecture, RabbitMQ est configuré pour s'exécuter sur le noeud de contrôleur.

## ▼ Configuration d'un noeud de contrôleur

1. **(Facultatif) Installez et configurez NTP.**  
Reportez-vous à la section [“Installation NTP \(Network Time Protocol\)”](#) à la page 47.
2. **(Facultatif) Installez et configurez MySQL.**  
Reportez-vous à la section [“Installation de MySQL”](#) à la page 48.
3. **Installez RabbitMQ.**
  - a. **Installez le package RabbitMQ.**  

```
controller# pkg install rabbitmq
```
  - b. **Activez le service SMF RabbitMQ.**  

```
controller# svcadm enable rabbitmq
```
4. **Installez et configurez Keystone.**  
Reportez-vous à la section [“Installation de Keystone”](#) à la page 49.
5. **Installez et configurez Cinder.**  
Reportez-vous à la section [“Installation de Cinder”](#) à la page 51.
6. **Installez et configurez Glance.**  
Reportez-vous à la section [“Installation de Glance”](#) à la page 56.
7. **Installez et configurez Neutron.**  
Voir [“Installation et configuration de Neutron sur le noeud du contrôleur”](#) à la page 57.
8. **Installez et configurez Nova.**  
Reportez-vous à la section [“Installation de Nova”](#) à la page 58.
9. **Configurez Horizon.**  
Reportez-vous à la section [“Configuration d'Horizon”](#) à la page 59.

## Installation NTP (Network Time Protocol)

L'installation de NTP (Network Time Protocol) est facultative, mais vivement recommandée. Si vous installez NTP service, installez-le sur chaque noeud de service de votre déploiement Cloud.

NTP permet de garantir la cohérence de l'heure du jour sur tous les noeuds du Cloud. Si vous activez NTP dans un réseau, configurez les noeuds de service pour obtenir l'heure sur laquelle ils sont sur le réseau.

- Si la multidiffusion est activée sur les sous-réseaux IP sur lesquels résident les noeuds de service, vous pouvez tirer parti du réseau IP pour utiliser la multidiffusion afin de configurer NTP.
- Si la multidiffusion n'est pas activée sur les sous-réseaux IP sur lesquels résident les noeuds de service, configurez NTP manuellement.

### ▼ Installation et configuration de NTP (Network Time Protocol)

#### 1. Installez le package NTP.

```
controller# pkg install ntp
```

#### 2. Installez le fichier de configuration.

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

#### 3. Configurez NTP.

La configuration nécessaire dépend de l'activation de la multidiffusion sur les sous-réseaux du noeud.

- **Si la multidiffusion est activée, aucune configuration supplémentaire n'est requise.**
- **Si la multidiffusion IP n'est pas activée, configurez les noms d'hôte ou les adresses IP des serveurs NTP existants**
  - a. **Mettez en commentaire l'option `multicastclient` dans le fichier `/etc/inet/ntp.conf`.**

```
# multicastclient 224.0.1.1
```
  - b. **Annulez le commentaire d'au moins une option de serveur dans le fichier `/etc/inet/ntp.conf` :**

```
server ntp_server_1 iburst
```

```
server ntp_server_2 iburst
```

**4. Activez le service SMF du serveur NTP.**

```
controller# svcadm enable ntp
```

## Installation de MySQL

De nombreux services OpenStack permettent à une base de données d'assurer le suivi des ressources critiques, leur utilisation et d'autres informations. Par défaut, les bases de données individuelles SQLite sont indiquées dans ce but, et s'avèrent utiles pour la configuration monoeud. Pour les configurations multinoeuds, une base de données MySQL est recommandée pour stocker ces informations.

### ▼ Installation d'une base de données MySQL

**1. Déterminez le nom principal associé au noeud de contrôleur.**

Utilisez l'adresse IP principale du noeud de contrôleur pour déterminer le nom principal associé à ce noeud.

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

**2. Installez le package du serveur MySQL.**

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

**3. Installez le pack client MySQL.**

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

**4. Activez le service SMF du serveur MySQL.**

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

**5. Définissez le mot de passe root du serveur MySQL.**

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

**6. Configurez le serveur MySQL.**

Créez les tables qui seront utilisées par OpenStack. Accordez les privilèges aux services sur le noeud de contrôleur afin de fournir un accès exclusif à ces bases de données. Utilisez le *controller-name* généré par la commande `getent hosts` précédente.

```
controller# mysql -u root -p
Enter password: MySQL-root-password
```

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

## 7. Installez le package de bibliothèque client MySQL Python.

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

# Installation de Keystone

Le service Keystone doit être installé et configuré sur le noeud de contrôleur.

## ▼ Installation et configuration de Keystone

### 1. Installez le package Keystone.

```
controller# pkg install keystone
```

### 2. Modifiez le fichier de configuration Keystone.

Annulez le commentaire et définissez deux paramètres suivants dans le fichier `/etc/keystone/keystone.conf`.

#### a. Définissez le paramètre `admin_token`.

Le paramètre `admin_token` est un "secret partagé" entre Keystone et OpenStack. La valeur de ce paramètre peut être n'importe quelle chaîne de caractères, mais cette valeur ne doit être ni divulguée ni distribuée. Pour créer ce type de chaîne, utilisez OpenSSL comme indiqué dans la commande suivante

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

Utilisez cette valeur de sortie pour définir le paramètre `admin_token` dans le fichier `/etc/keystone/keystone.conf`

```
admin_token = random_string
```

**b. Définissez le paramètre `connection`.**

Le paramètre de `connection` est un URI qui représente l'emplacement de la base de données Keystone et le type de base de données utilisée.

Utilisez le `controller-name` généré par la commande `getent hosts` précédente pour définir le paramètre `connection` dans le fichier `/etc/keystone/keystone.conf`.

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

**3. Générez des jetons PKI (Public Key Infrastructure).**

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

**4. Activez le service MSF Keystone.**

```
controller# svcadm enable keystone
```

**5. Alimenter la base de données Keystone.**

Cette étape peut être effectuée manuellement ou vous pouvez utiliser le script `sample_data.sh` comme indiqué dans l'exemple suivant. Utilisez le `controller-name` généré par la commande `getent hosts` précédente.

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

Le script `sample_data.sh` prend en charge les variables d'environnement qui définissent le nœud sur lequel réside chaque service d'API ainsi que le mot de passe de chaque service. Consultez le script pour plus d'informations sur les paramètres qui peuvent être définis à partir de l'environnement. Par défaut, un utilisateur Keystone est créé pour chaque service, sous le locataire `service`, avec un mot de passe identique au nom de l'utilisateur. Par exemple, un utilisateur `nova` est créé avec le mot de passe `nova`.

## Installation et configuration de Heat

Heat est un moteur d'orchestration OpenStack qui permet de déployer des applications pour le cloud basées sur des modèles que vous créez. Vous installez Heat sur le même nœud que Keystone.

## ▼ Configuration de Heat

Avant de commencer

Vous devez d'abord configurer Keystone comme décrit à la section [“Installation et configuration de Keystone”](#) avant d'effectuer cette tâche.

### 1. Installez le pack Heat.

```
controller# pkg install heat
```

### 2. Exécutez le script de configuration de Heat.

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

### 3. Modifiez `/etc/heat/api-past.ini` en mettant à jour les informations suivantes dans le fichier.

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

### 4. Activez le service Heat.

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

## Installation de Cinder

La configuration Cinder doit indiquer au moins les informations suivantes

- Informations d'autorisations pour authentification auprès de Keystone.
- Classe des volumes à créer.

## ▼ Installation et configuration de Cinder

### 1. Installez le package Cinder.

```
controller# pkg install cinder
```

### 2. Précisez les informations de configuration d'authentification.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/cinder/api-paste.ini`. Ces paramètres indiquent l'emplacement du service de l'API Keystone et les informations d'authentification Cinder.

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

### 3. Indiquez les classes des volumes à créer par le service de volume Cinder.

Dans le fichier `/etc/cinder/cinder.conf`, annulez le commentaire du paramètre `volume_driver` approprié. Les quatre classes de volumes ci-dessous sont prises en charge

ZFSVolumeDriver

Prend en charge la création de volumes locaux que Nova utilisera sur le même noeud que le service de volume Cinder.

ZFISISCSIDriver

Prend en charge la création et l'exportation des cibles iSCSI utilisés par les noeuds de calcul distants Nova.

ZFSFCDriver

Prend en charge la création et l'exportation des Luns Fibre Channel utilisés par les noeuds de calcul distants Nova.

ZFSSAISCSIDriver

Prend en charge la création et l'exportation des cibles iSCSI à partir d'Oracle ZFS Storage Appliance utilisées par les noeuds de calcul distants Nova. Des paramètres supplémentaires de ce pilote doivent être définis dans le fichier `/etc/cinder/cinder.conf`.

Dans l'exemple décrit dans ce chapitre, iSCSI permet de fournir les volumes utilisés par les instances Nova. Mettez en commentaire la sélection par défaut de `ZFSVolumeDriver`, et annulez le commentaire de la sélection `ZFISISCSIDriver`.

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

### 4. Définissez les paramètres de configuration supplémentaires.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/cinder/cinder.conf`. Ces paramètres précisent l'emplacement du service de l'API Glance, la base de données correspondante pour Cinder et le service RabbitMq.

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

### 5. Si les cibles iSCSI seront configurées, activez les services correspondants.

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

### 6. Activez les services SMF Cinder.

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

Voir aussi Reportez-vous également à [How to Build OpenStack Block Storage on ZFS](#).

## ▼ Configuration du pilote ZFS Storage Appliance iSCSI Cinder

Le pilote Oracle ZFS Storage Appliance iSCSI Cinder permet au ZFSSA (Oracle ZFS Storage Appliance) d'être utilisé de façon transparente sous la forme d'une ressource de stockage en mode bloc pour Cinder. Le pilote permet de créer des volumes iSCSI pouvant être alloués par un serveur Cinder à toute machine virtuelle instanciée par le service Nova. Le pilote est distribué par le package `cloud/openstack/cinder`. Votre appareil doit exécuter au moins la version logicielle 2013.1.2.0. de ZFSSA.

**Avant de commencer** Configuration d'un pool sur Oracle ZFS Storage Appliance. Vous pouvez décider d'utiliser un pool préexistant.

### 1. Exécutez le workflow `cinder.akwf`.

Vous pouvez utiliser un utilisateur existant ou créer un nouvel utilisateur avec les autorisations de rôle pour effectuer les opérations du pilote Cinder.

Le workflow `cinder.akwf` effectue les tâches suivantes

- Création de l'utilisateur s'il n'existe pas.
- Définition des autorisations de rôle pour effectuer les opérations du pilote Cinder.
- Activation du service RESTful si le service est désactivé.

Vous pouvez exécuter le workflow dans l'interface de ligne de commande (CLI) ou à partir de l'interface utilisateur de navigateur (BUI) de votre appareil.

### ■ Exécutez le workflow à partir de la CLI.

```
zfssa:maintenance workflows> download
zfssa:maintenance workflows download (uncommitted)> show
Properties:
```

```

        url = (unset)
        user = (unset)
        password = (unset)

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

zfssa:maintenance workflows> ls
Properties:
        showhidden = false

Workflows:

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

zfssa:maintenance workflows> select workflow-001

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

```

■ **Exécutez le workflow à partir de la BUI.**

- a. **Sélectionnez Maintenance -> Workflow, et utilisez l'icône plus (+) pour télécharger un nouveau workflow.**
- b. **Cliquez sur le bouton Parcourir et sélectionnez le fichier cinder.akwf.**
- c. **Cliquez sur le bouton TELECHARGER pour terminer le téléchargement du workflow.**
- d. **Cliquez sur la nouvelle ligne qui apparaît sur la page Workflows BUI pour exécuter le workflow du pilote Cinder.**

Le workflow invite à entrer un nom d'utilisateur et un mot de passe. Ce nom d'utilisateur et ce mot de passe seront également utilisés dans le fichier `cinder.conf` en tant que `zfssa_auth_user` et `zfssa_auth_password`.

**2. Définissez les paramètres dans le fichier `cinder.conf`.**

Indiquez les propriétés obligatoires suivantes dans le fichier `cinder.conf`

- `volume_driver` – Assurez-vous que `cinder.volume.drivers.zfssa.zfssaiscsi.ZFSSAISCSIDriver` n'est pas commentée. Assurez-vous que les trois autres sélections sont commentées.
- `zfssa_host` – Nom ou adresse IP de l'hôte de gestion ZFSSA.
- `zfssa_auth_user` – Nom d'utilisateur de l'utilisateur Cinder sur le ZFSSA.
- `zfssa_auth_password` – Mot de passe de l'utilisateur Cinder sur le ZFSSA.
- `zfssa_pool` – Le pool à utiliser pour allouer les volumes.
- `zfssa_target_portal` – Portail cible iSCSI ZFSSA (`data-ip:port`). Le port par défaut est 3260.
- `zfssa_project` – Nom du projet ZFSSA. Si le projet n'existe pas sur l'appareil, un projet avec le même nom est créé par le pilote au démarrage. Ce projet contient tous les volumes créés par le pilote. Des propriétés ZFSSA supplémentaires sont utilisées pour définir les caractéristiques des volumes (la taille de bloc, par exemple) et leur accès (par exemple, initiateurs, cibles, sécurité).
- `zfssa_initiator_group` – Nom du groupe d'initiateurs. Si le groupe d'initiateurs n'existe pas sur l'appareil, un groupe d'initiateurs avec le même nom est créé par le pilote au moment du démarrage. Si vous souhaitez utiliser le groupe d'initiateurs `default`, définissez la valeur de ce paramètre avec sa valeur `default`. Le groupe d'initiateurs `default` peut s'avérer utile à des fins d'évaluation. Le groupe d'initiateurs `default` ne doit normalement pas être utilisé, car cela risque d'exposer les volumes à des initiateurs non souhaités ou incompatibles.
- `zfssa_target_interfaces` – Les interfaces de réseau des cibles iSCSI ZFSSA. Exécutez la commande suivante pour afficher les interfaces

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

```
Interfaces:
```

```

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

- `connection` – Modifiez la connection en `sql_connection`.

Recherchez la ligne suivante :

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

Comme indiqué ci-dessous modifiez cette ligne

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

### 3. Vérifiez que le service ZFSSA iSCSI est en ligne.

Si le service ZFSSA iSCSI n'est pas en ligne, utiliser la BUI ou la CLI dans l'appareil pour l'activer. L'exemple suivant montre comment utiliser la CLI dans l'appareil

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

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

**4. Activez les services SMF de volume Cinder.**

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

## Installation de Glance

De même que pour Cinder, la configuration de Glance nécessitera de configurer certaines informations d'authentification ainsi que des informations sur l'emplacement des services MySQL et RabbitMQ.

### ▼ Configuration et installation de Glance

**1. Installez le package Glance.**

```
controller# pkg install glance
```

**2. Définissez les paramètres de configuration Glance.**

Annulez le commentaire et définissez les paramètres suivants dans chacun des fichiers ci-dessous

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

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

**3. Indiquez l'URI de la base de données MySQL.**

Dans les deux des fichiers et /etc/glance/glance-api.conf et /etc/glance/glance-registry.conf, indiquez l'URI de la base de données MySQL :

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

**4. Indiquez l'emplacement du service RabbitMQ.**

Dans `/etc/glance/glance-api.conf`, définissez l'emplacement du service RabbitMQ.

```
rabbit_host = controller-name
```

#### 5. Activez les services SMF Glance.

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

## Installation et configuration de Neutron sur le noeud du contrôleur

Dans l'architecture décrite dans ce chapitre, le service de l'API Neutron s'exécute sur le noeud de contrôleur. Pour permettre à ce service de communiquer avec le contrôleur EVS installé sur le noeud Réseau, les clés SSH publiques de l'utilisateur Neutron du contrôleur doivent être déposées dans le fichier `authorized_keys` de l'utilisateur `evsuser`.

### ▼ Installation et configuration de Neutron

#### 1. Installez le package Neutron.

```
controller# pkg install neutron
```

#### 2. Créez une clé publique SSH sur le noeud de contrôleur pour l'utilisateur neutron.

Cette clé permettra au service de l'API Neutron d'accéder au contrôleur EVS.

Utilisez la commande `ssh-keygen` en tant qu'utilisateur `neutron`, et créez la clé pour l'utilisateur `neutron`.

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

#### 3. Copiez la clé vers le noeud de contrôleur EVS.

Copiez la clé publique SSH générée à l'étape précédente, `/var/lib/neutron/.ssh/id_rsa.pub`, sur le noeud sur lequel le contrôleur EVS est en cours d'exécution. Cette clé est référencée lors de la configuration du contrôleur EVS.

#### 4. Définissez les paramètres dans le fichier de configuration Neutron.

Indiquez les informations d'authentification et l'emplacement du service RabbitMQ. Dans le fichier `/etc/neutron/neutron.conf`, annulez le commentaire et définissez les paramètres suivants

```
rabbit_host = controller-name
```

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

## 5. Indiquez l'emplacement du contrôleur EVS.

### a. Obtenez l'emplacement du contrôleur EVS.

Utilisez l'adresse IP du noeud de réseau avec la commande `getent hosts` pour obtenir le nom du noeud de réseau

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

### b. Définissez l'emplacement du contrôleur EVS.

Dans le fichier `/etc/neutron/plugins/evs/evs_plugin.ini`, supprimez les commentaires ou définissez les paramètres suivants. Utilisez le résultat de la commande `getent hosts` pour définir l'emplacement du contrôleur EVS :

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

## 6. Activez le service de serveur Neutron.

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

# Installation de Nova

La configuration de Nova sur le noeud de contrôleur nécessite également de configurer les informations habituelles d'authentification et de service.

## ▼ Installation et configuration de Nova

### 1. Installez le package Nova.

```
controller# pkg install nova
```

### 2. Précisez les informations de configuration d'authentification.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/nova/api-paste.ini`. Ces paramètres spécifient l'emplacement du service de l'API Keystone et les informations d'authentification Nova.

```
auth_uri = http://controller-name:5000/v2.0
```

```
identity_uri = http://controller-name:35357
admin_tenant_name = service
admin_user = nova
admin_password = nova-password
```

### 3. Définissez les paramètres de configuration supplémentaires.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/nova/nova.conf`. Ces paramètres indiquent d'autres points d'extrémité du service d'authentification Keystone, le service de l'API Glance, le service de l'API Neutron, le service RabbitMQ et l'URI de la base de données spécifique à Nova.

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

### 4. Activez les services SMF Nova.

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

## ▼ Configuration d'Horizon

### 1. Installez le pack Horizon.

```
horizon# pkg install horizon
```

### 2. Générez les certificats à utiliser par Horizon.

Les commandes suivantes génèrent des certificats auto-signés pour Horizon et copient le fichier de configuration du tableau de bord OpenStack sur le répertoire de fichiers de configuration Apache. Pour plus d'informations sur la création de certificats autosignés, reportez-vous à la foire aux questions Apache : [SSL/TLS Strong Encryption: FAQ](#).

```
controller# DASHBOARD=/etc/openstack_dashboard
controller# openssl req -new -x509 -nodes
-out horizon.crt -keyout horizon.key
controller# mv horizon.crt horizon.key ${DASHBOARD}
controller# chmod 0600 ${DASHBOARD}/horizon.*
controller# sed
-e "/SSLCertificateFile/s:/path.*:${DASHBOARD}/horizon.crt:"
-e "/SSLCACertificateFile/d"
```

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

3. **Dans le fichier `~/conf.d/openstack-dashboard-tls.conf`, indiquez l'adresse du site et le nom de serveur du pack Horizon sur les paramètres suivants**

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

---

**Remarque** - Dans l'exemple actuel de configuration à trois noeuds, les deux paramètres doivent indiquer le système de noeud du contrôleur.

---

4. **Effectuez l'une des opérations suivantes pour lire le nouveau fichier de configuration**

- Si le service Apache est désactivé, activez-le

```
controller# svcadm enable apache22
```

- Si le service Apache est en ligne, redémarrez-le.

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

## Configuration du noeud de calcul

Le noeud de calcul est l'endroit où les instances de machine virtuelle sont installées. Vous souhaitez peut-être de nombreux noeuds de calcul pour votre Cloud.

Pour une meilleure gestion de l'utilisation de la mémoire entre ZFS et les applications dans Oracle Solaris 11, définissez le paramètre `usr_reserve_hint_pct` sur le noeud, comme indiqué dans l'exemple suivant :

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

où *site* peut correspondre à votre entreprise.

Vous devez également définir ce paramètre sur les différents noeuds OpenStack.

Pour plus d'informations sur ce paramètre, connectez-vous à votre compte MOS à l'adresse <https://support.oracle.com> et consultez le Document 1663862.1, *Memory Management Between ZFS and Applications in Oracle Solaris 11.2*.

## ▼ Configuration du noeud de calcul

### 1. (Facultatif) Installez et configurez NTP.

Reportez-vous à la section “[Installation NTP \(Network Time Protocol\)](#)” à la page 47.

### 2. Installez le package Nova.

```
compute1# pkg install nova
```

### 3. Redémarrez RAD (Démon d'accès distant).

Nova utilise RAD pour communiquer avec la structure Oracle Solaris Zones.

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

### 4. Précisez les informations de configuration d'authentification.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/nova/api-paste.ini`. Ces paramètres spécifient l'emplacement du service de l'API Keystone et les informations d'authentification Nova.

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

### 5. Configurez les paramètres d'authentification et de base de données ainsi que les services pertinents.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/nova/nova.conf`. Ces paramètres indiquent d'autres points d'extrémité du service d'authentification Keystone, le service de l'API Glance, le service de l'API Neutron, le service RabbitMQ et l'URI de la base de données spécifique à Nova.

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

### 6. Installez le pack EVS sur le noeud de calcul.

```
# pkg install evs
```

### 7. Créez une clé publique SSH sur le noeud de calcul pour l'utilisateur root.

Cette clé permettra à la structure des zones Solaris d'accéder au contrôleur EVS.

Utilisez la commande `ssh-keygen` tant qu'utilisateur `root`, et créez la clé pour l'utilisateur `root`.

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

#### 8. Copiez la clé vers le noeud de contrôleur EVS.

Copiez la clé publique SSH générée à l'étape précédente, `/root/.ssh/id_rsa.pub`, vers le noeud exécutant le contrôleur EVS, c'est-à-dire le noeud de réseau dans cette configuration réseau. Cette clé sera référencée lors de la configuration du contrôleur EVS sur le noeud de réseau.

Après cette étape, vous pouvez enregistrer des images Glance. Reportez-vous à la section [“Création d'une image” à la page 115](#) et [“Ajout d'une image au magasin d'images” à la page 116](#).

#### 9. Activez le service de calcul Nova.

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

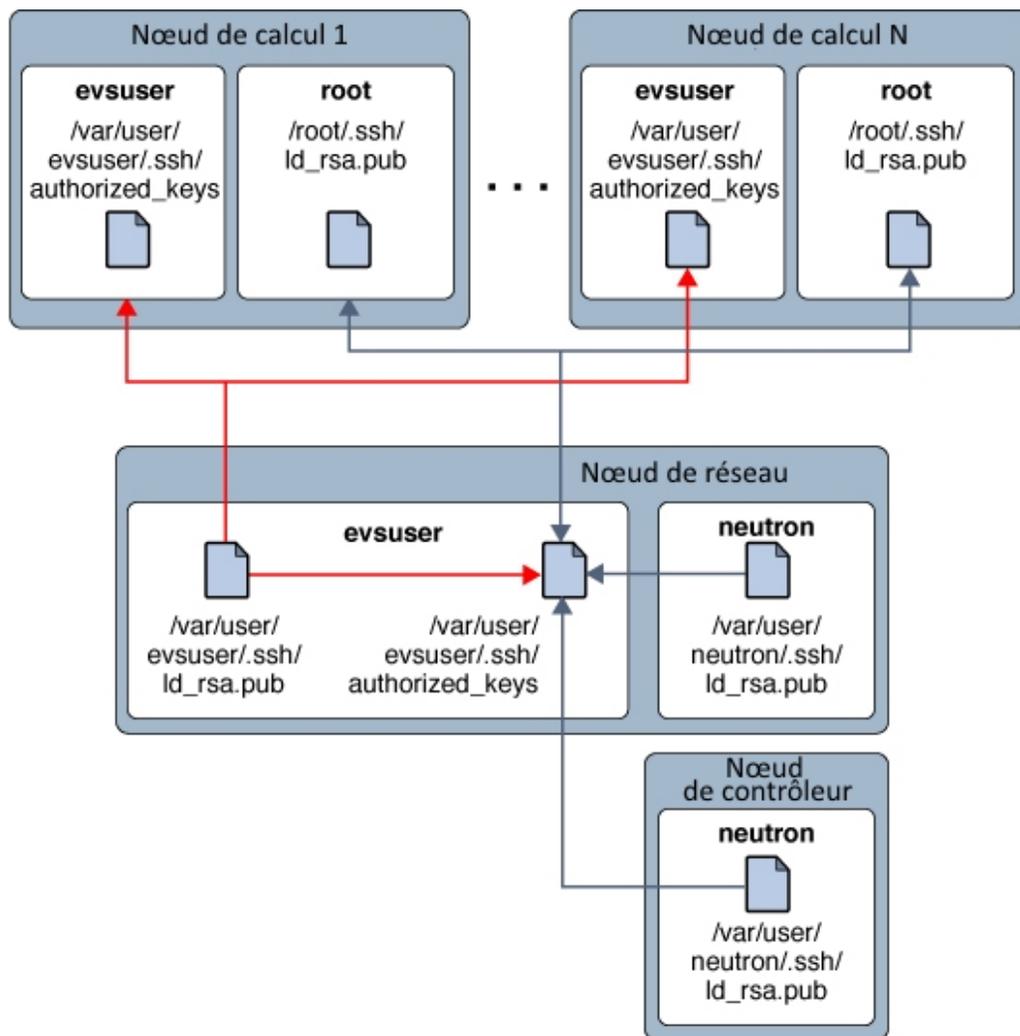
## Configuration du noeud de réseau

La configuration du noeud de réseau nécessite la configuration à la fois du commutateur virtuel élastique (EVS) et l'agent DHCP Neutron. Vous pouvez éventuellement configurer l'agent Neutron de couche 3 (L3).

EVS forme le backend du réseau OpenStack et facilite la communication entre les instances de machine virtuelle, soit par les VLAN, soit par les VXLAN. Les instances VM peuvent se trouver sur le même noeud de calcul ou sur plusieurs noeuds de calcul. Pour plus d'informations sur EVS, reportez-vous au [Chapitre 5, “ A propos des commutateurs virtuels élastiques ” du manuel “ Gestion de la virtualisation réseau et des ressources réseau dans Oracle Solaris 11.2 ”](#).

Lorsque vous configurez le noeud Réseau, veillez à copier les clés publiques SSH de l'utilisateur `evsuser` dans le fichier `authorized_keys` de chaque utilisateur `evsuser` sur les noeuds Calcul et Réseau. Reportez-vous à l'image suivante qui illustre la répartition des clés publiques SSH. L'image suppose que plusieurs noeuds de calcul ont été configurés.

FIGURE 3-2 Répartition des clés SSH du contrôleur EVS



Pour une meilleure gestion de l'utilisation de la mémoire entre ZFS et les applications dans Oracle Solaris 11, définissez le paramètre `usr_reserve_hint_pct` sur le noeud, comme indiqué dans l'exemple suivant :

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

où *site* peut correspondre à votre entreprise.

Vous devez également définir ce paramètre sur les différents noeuds OpenStack.

Pour plus d'informations sur ce paramètre, connectez-vous à votre compte MOS à l'adresse <https://support.oracle.com> et consultez le Document 1663862.1, *Memory Management Between ZFS and Applications in Oracle Solaris 11.2*.

## ▼ Configuration du noeud de réseau

### 1. (Facultatif) Installez et configurez NTP.

Reportez-vous à la section “[Installation NTP \(Network Time Protocol\)](#)” à la page 47.

### 2. Installez le package Neutron.

```
network# pkg install neutron
```

### 3. Créez des clés publiques SSH sur le noeud Réseau pour les utilisateurs neutron et evsuser.

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

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

### 4. Combinez les fichiers de clés dans le fichier de clés autorisées EVS.

Combinez les clés publiques SSH précédemment générées à partir des noeud de contrôleur et de calcul avec ces deux nouvelles clés et concaténez la combinaison dans le fichier de clés autorisées pour evsuser.

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

### 5. Copiez la clé publique SSH /var/user/evsuser/.ssh/id\_rsa.pub de l'utilisateur evsuser dans le fichier authorized\_keys de chacun des noeuds Calcul et Réseau de l'utilisateur evsuser.

Reportez-vous à la [Figure 3-2, “Répartition des clés SSH du contrôleur EVS”](#) pour obtenir un aperçu de la répartition de la clé publique SSH de l'utilisateur evsuser's 'dans le contexte de la configuration globale du cloud.

### 6. Vérifiez la bonne connectivité SSH.

Utilisez la commande `ssh` pour établir la connexion au contrôleur EVS, et répondez `yes` lorsque vous y êtes invité, pour chaque connexion. Effectuez cette étape sur chaque noeud de l'architecture OpenStack.

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

## 7. Installez le package du contrôleur EVS.

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

## 8. Redémarrez RAD.

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

## 9. Configurez le contrôleur EVS.

Décidez si le réseau virtuel sur lequel les instances Nova communiqueront est un VLAN ou un VXLAN. L'exemple ci-dessous effectue la configuration de VLAN suivant

- Un VLAN avec un ID de 13 est configuré pour un réseau externe utilisé pour la connexion réseau en dehors du Cloud.
- La plage d'ID du VLAN de 1000-2000 est configurée pour les sous-réseaux créés par EVS.

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

## 10. Configurez les paramètres d'authentification et l'emplacement du service RabbitMQ.

Annulez le commentaire et définissez les paramètres suivants dans le fichier `/etc/neutron/neutron.conf`. Ces paramètres indiquent d'autres points d'extrémité du service d'authentification Keystone, le service de l'API Glance, le service de l'API Neutron, le service RabbitMQ et l'URI de la base de données spécifique à Neutron.

```
[keystone_authtoken]
signing_dir = /var/lib/neutron/keystone-signing
auth_host = 127.0.0.1
auth_port = 35357
auth_protocol = http
auth_uri = http://controller-IP:5000/v2.0
admin_tenant_name = service
admin_user = neutron
admin_password = neutron-password
identity_uri = http://controller-IP:35357
.
```

```
[DEFAULT]
core_plugin = neutron.plugins.evs.plugin.EVSNeutronPluginV2
allow_overlapping_ips = False

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

**11. Configurez l'agent DHCP Neutron.**

Annulez le commentaire et définissez le paramètre suivant dans le fichier `/etc/neutron/dhcp_agent.ini` pour indiquer l'emplacement du contrôleur EVS.

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

**12. (Facultatif) Indiquez le domaine par défaut à utiliser pour résoudre des demandes DNS des instances Nova.**

Dans le fichier `/etc/neutron/dhcp_agent.ini`, annulez le commentaire et définissez le paramètre `dhcp_domain`, sur le domaine par défaut à utiliser pour résoudre les demandes DNS des instances Nova.

**13. Activez l'agent DHCP.**

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

**14. Configurez l'agent Neutron L3.**

Cette étape est facultative, mais vivement recommandée.

## Configuration de l'agent Neutron L3

Cette section explique comment créer un réseau virtuel qui représente un réseau externe. Sur ce réseau virtuel, le DHCP n'est pas utilisé. A la place, des adresses IP flottantes sont créées. Ces adresses IP flottantes sont affectées à un locataire spécifique et peuvent être affectées aux instances des instances de machine virtuelle Nova pour que leurs utilisateurs sous ce locataire puissent les utiliser. L'agent Neutron L3 crée automatiquement des mappages NAT bi-univoques entre les adresses affectées aux instances Nova et les adresses IP flottantes.

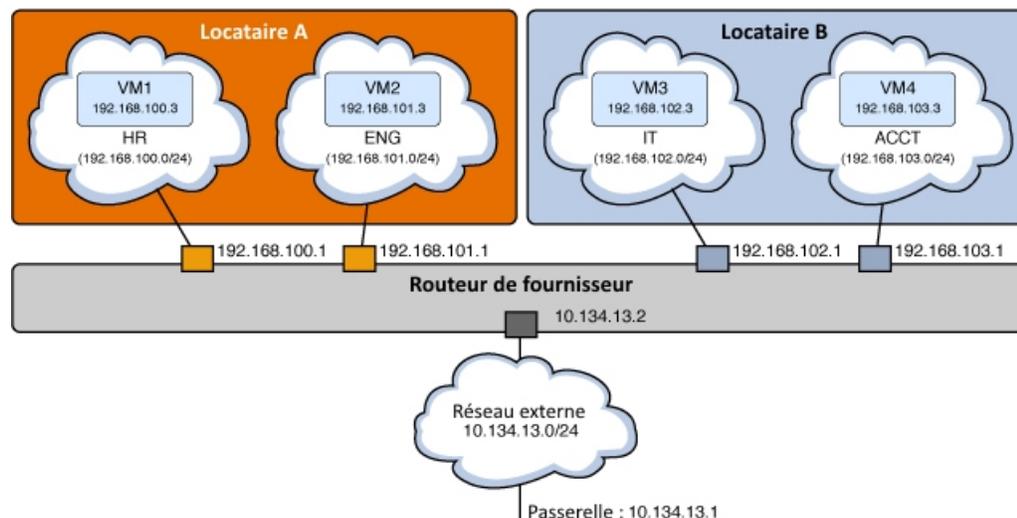
L'implémentation Oracle Solaris 11.2 de OpenStack Neutron prend en charge un routeur de fournisseur avec modèle de déploiement de réseaux privés. Pour plus d'informations sur ce modèle, reportez-vous aux cas d'emploi d'OpenStack Neutron au [Chapitre 7 du Guide de formation de l'opérateur](#). Dans cet un modèle de déploiement, chaque locataire peut disposer d'au moins un réseau privé, et tous les réseaux locataires partagent le même routeur. L'administrateur de données crée, possède et gère le routeur. Le routeur n'est pas visible dans la vue de la topologie réseau du locataire. Etant donné qu'il n'y a qu'un routeur, les réseaux locataires ne peuvent pas utiliser le chevauchement des adresses IP. L'administrateur crée les réseaux privés pour le compte des locataires.

Par défaut, le routeur, dans ce modèle, empêche le routage entre les réseaux privés qui font partie d'un même locataire : les instances de machine virtuelle à l'intérieur d'un réseau privé ne peuvent pas communiquer avec les instances de machine virtuelle d'un autre réseau privé, même en faisant partie du même locataire. Pour modifier ce comportement, définissez `allow_forwarding_between_networks` sur `True` dans le fichier de configuration `/etc/neutron/l3_agent.ini` et redémarrez le service SMF `neutron-l3-agent`.

Dans ce modèle, le routeur fournit la connectivité des instances de machine virtuelle du locataire au monde extérieur. Le routeur effectue une NAT bidirectionnelle sur l'interface qui connecte le routeur au réseau externe. Les locataires créent autant d'adresses IP flottantes que nécessaire ou qu'autorisées par le quota d'IP flottantes, puis associent ces IP flottantes aux instances de machine virtuelle qui requièrent une connectivité avec l'extérieur.

La figure suivante illustre le modèle de déploiement Neutron {ENT :OS11u2}. Une description suit la figure.

**FIGURE 3-3** Routeur de fournisseur avec modèle de réseaux privés



Dans le modèle présenté dans la figure précédente, chaque locataire dispose de deux réseaux internes et de deux instances de machine virtuelle.

- |           |  |
|-----------|--|
| Locataire | <ul style="list-style-type: none"> <li>■ Réseau HR avec sous-réseau 192.168.100.0/24 et passerelle 192.168.100.1</li> <li>■ Réseau ENG sous-réseau 192.168.101.0/24 et passerelle 192.168.101.1</li> </ul> |
|-----------|--|

- VM1 connectée à HR avec une adresse IP fixe de 192.168.100.3
  - VM2 connectée à ENG avec une adresse IP fixe de 192.168.101.3
- Locataire
- Réseau IT avec sous-réseau 192.168.102.0/24 et passerelle 192.168.102.1
  - Réseau ACCT avec sous-réseau 192.168.103.0/24 et passerelle 192.168.103.1
  - VM3 connectée à IT avec une adresse IP fixe de 192.168.102.3
  - VM4 connectée à ACCT avec une adresse IP fixe de 192.168.103.3

Le tableau NAT bidirectionnelle est configuré comme indiqué dans le tableau suivant.

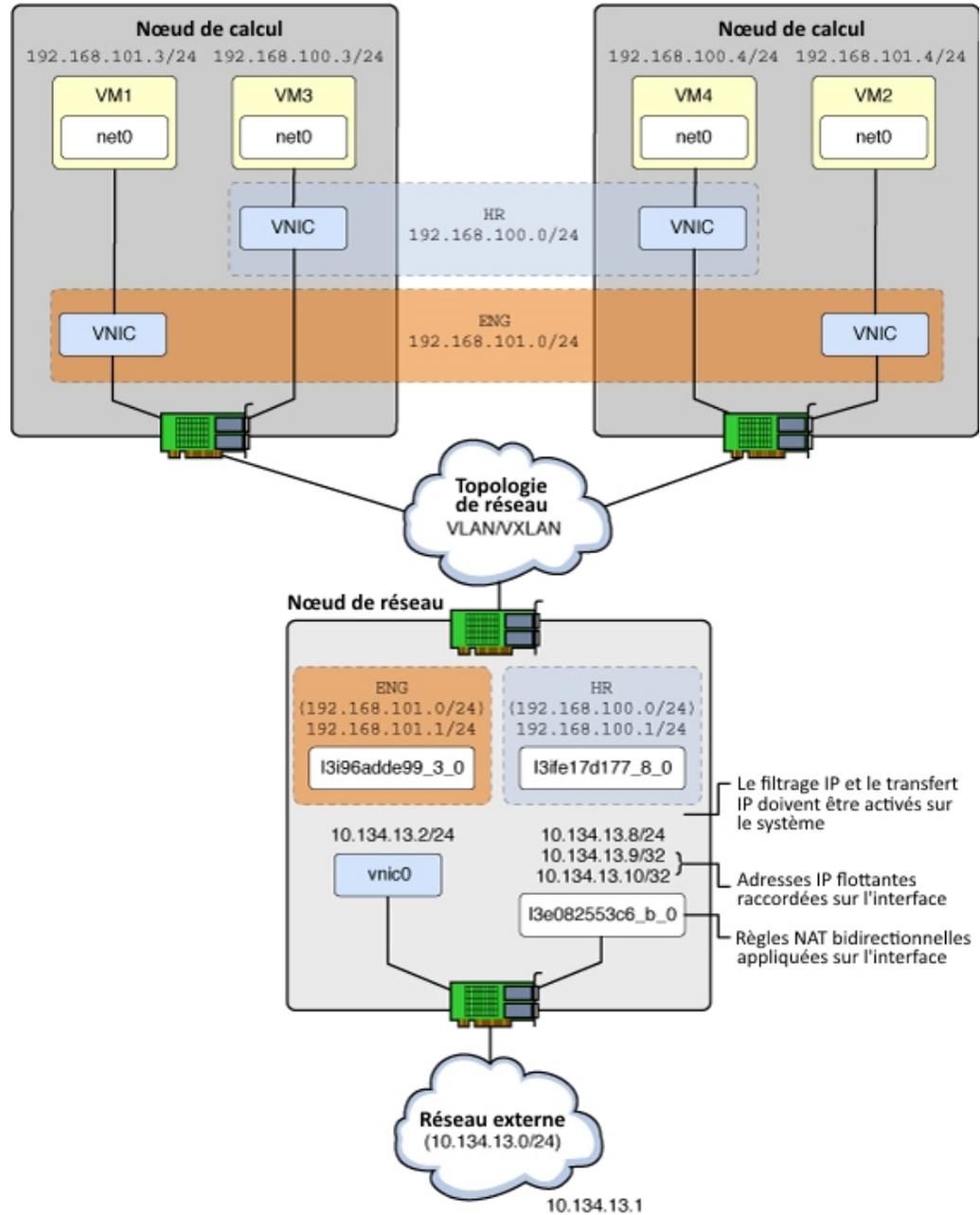
IP interne	IP flottante
192.168.100.3	10.134.13.40
192.168.101.3	10.134.13.9

Toutes les interfaces de passerelle sont instanciées sur le noeud qui exécute le service SMF neutron-l3-agent. Bien qu'un Cloud puisse avoir plusieurs instances Neutron, seul un neutron-l3-agent par Cloud est nécessaire.

Le réseau externe est un réseau fournisseur associé au sous-réseau qui 10.134.13.0/24, qui est accessible par l'extérieur. Les locataires créeront les adresses IP flottantes à partir de ce réseau et les associeront à leurs instances de machine virtuelle. VM1 et VM2 ont des adresses IP flottantes 10.134.13.40 et 10.134.13.9 qui leur sont associées respectivement. VM1 et VM2 sont accessibles par l'extérieur via ces adresses IP.

Le diagramme suivant illustre la façon dont les ressources de locataire sont configurées dans un noeud de réseau et deux noeuds de calcul. Les descriptions de certaines fonctions suivent la figure.

FIGURE 3-4 Configuration de l'agent Neutron L3



VNIC	Interface réseau virtuelle.
l3e...	L'agent L3 a créé une VNIC sur le réseau externe ("e") où la NAT bidirectionnelle se produit.
l3i...	L'agent L3 a créé une VNIC sur le réseau interne ("i") qui dispose de l'adresse IP de la passerelle par défaut.

La liste ci-dessous présente les modalités d'utilisation des adresses IP dans cet exemple de configuration.

10.134.13.1

Passerelle par défaut

10.134.13.2 - 10.134.13.7

Adresses IP réservées pour exposer les APIs {ENT :OST} (telles que Nova, Cinder, Glance) aux locataires.

10.134.13.9 - 10.134.13.254

Adresses IP flottantes pour les instances de machine virtuelle du locataire

## ▼ Configuration de l'agent Neutron L3

Cette procédure explique comment utiliser le locataire de service afin de créer un routeur, un réseau externe et un sous-réseau externe qui seront utilisés par tous les locataires du centre de données.

Cette procédure est exécutée par l'administrateur de centre de données. Vous devez utiliser la ligne de commande pour configurer le routeur unique partagé et associer des réseaux et des sous-réseaux de différents locataires, car le tableau de bord OpenStack vous permet de gérer les ressources d'un seul locataire à la fois.

Reportez-vous à la [Figure 3-4, "Configuration de l'agent Neutron L3"](#) au fur et à mesure que vous effectuez les étapes ci-après.

**Avant de commencer**

Vous devez terminer la configuration des réseaux internes avant d'effectuer cette procédure.

### 1. Activez la fonctionnalité filtre IP de Solaris.

```
network# svcadm enable ipfilter
```

### 2. Activez le transfert d'IP sur tout l'hôte.

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

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

### 3. Assurez-vous qu'EVS est correctement configuré et qu'il dispose de l'ID du VLAN requise pour le réseau externe.

Dans l'exemple suivant, l'ID et les valeurs de plage VLAN sont basées sur la configuration EVS effectuée à l'Étape 9.

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

### 4. Assurez-vous que le locataire de service existe.

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

### 5. Créez le routeur de fournisseur.

Le routeur de fournisseur est créé comme l'utilisateur neutron OpenStack sous le locataire de service.

Notez l'UUID id) du nouveau routeur. Vous en aurez besoin à l'étape suivante.

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

### 6. Mettez à jour le fichier de configuration de l'agent L3.

Mettez à jour la valeur de `routeur_id` dans le fichier `/etc/neutron/l3_agent.ini`, à l'aide de l'UUID id) de routeur de l'étape précédente.

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

### 7. Activez le service SMF neutron-l3-agent.

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

### 8. Créez un réseau externe.

Le réseau virtuel est créé en tant qu'utilisateur neutron {ENT :OST} sous le locataire de service.

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

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

### 9. Associez un sous-réseau au réseau extérieur.

Créez un réseau de sous-réseau associé pour le réseau extérieur, en indiquant que le DHCP est désactivé et que le sous-réseau sera utilisé pour le routage externe. Indiquez un sous-réseau IP routable à partir de l'emplacement où les adresses IP seront affectées. Dans cet exemple, le sous-réseau est associé à l'ID de VLAN 13.

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

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

### 10. Ajoutez le réseau externe au routeur.

Dans la commande suivante, la première UUID est l'UUID du provider\_router, et la seconde UUID est celle du external\_network.

```

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

```

## 11. Ajoutez les réseaux privés du locataire au routeur.

Les réseaux affichés par la commande `neutron net-list` ont été configurés précédemment.

```

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

```

Dans la commande suivante, la première UUID est celle du `provider_router`, et la seconde UUID est celle du sous-réseau HR.

```

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

```

Dans la commande suivante, la première UUID est celle du `provider_router`, et la seconde UUID celle du sous-réseau ENG.

```

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

```

- Voir aussi
- [“Respect de la configuration de l'agent L3” à la page 75](#)
  - [“Limitations connues” à la page 121](#)

## ▼ Création et association des adresses IP flottantes en tant qu'utilisateur locataire

Cette procédure est effectuée par un utilisateur locataire avec le tableau de bord Horizon {ENT : OST}.

### 1. Connectez-vous au tableau de bord OpenStack.

Connectez-vous, comme indiqué dans la section [“Accès au tableau de bord OpenStack” à la page 28](#), avec les informations d'identification de l'utilisateur locataire.

### 2. Sélectionnez Projet -> Accès et sécurité -> IP flottantes.

### 3. Sélectionnez external\_network.

### 4. Cliquez sur le bouton Allouer IP.

L'onglet IP flottantes indique que l'adresse IP flottante 10.134.13.9 est allouée.

### 5. Cliquez sur le bouton Associer.

### 6. Sélectionnez le port de l'instance de machine virtuelle dans le menu déroulant.

La fenêtre Projet -> Instances indique que l'adresse IP flottante est associée à l'instance de machine virtuelle.

Si vous avez sélectionné une paire de clés (clé publique SSH) lors du lancement d'une instance de machine virtuelle, cette clé SSH sera ajoutée dans le fichier `authorized_keys` pour l'utilisateur `root` de l'instance de machine virtuelle.

### 7. Connectez-vous à l'instance de machine virtuelle en cours d'exécution.

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

## ▼ Respect de la configuration de l'agent L3

Vous pouvez utiliser des commandes de filtre IP, telles que `ipf`, `ippool`, et `ipnat` et des commandes de mise en réseau, telles que `dladm` et `ipadm` pour respecter et résoudre les problèmes de la configuration effectuée par `neutron-l3-agent`.

### 1. Affichez les VNIC créées par `neutron-l3-agent`.

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

### 2. Affichez les adresses IP créée par `neutron-l3-agent`.

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

### 3. Affichez les règles de filtrage IP.

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

### 4. Affichez les règles de filtrage NAT.

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



## Installation sur plusieurs systèmes pour une configuration Juno multinoeud OpenStack

---

Ce chapitre décrit la procédure d'installation d'une configuration OpenStack multinoeud. Pour plus d'informations sur une installation monoeud, reportez-vous au [Chapitre 2, Installation d'une configuration d'évaluation](#).

Ce chapitre comprend les sections suivantes :

- “Présentation de l'architecture à trois noeuds” à la page 77
- “Configuration du noeud de contrôleur” à la page 81
- “Configuration du noeud de calcul” à la page 98
- “Configuration du noeud de stockage” à la page 100

---

**Remarque** - Ce chapitre concerne une configuration OpenStack spécifique de Juno qui est prise en charge uniquement à compter de la version Oracle Solaris 11.2 SRU10.

- Pour plus d'informations sur l'installation de la version d'Oracle Solaris 11.2 SRU10 et de la mise à niveau d'une configuration Havana existante vers Juno, reportez-vous au document [Havana to Juno: OpenStack Upgrade Procedures](#).
  - Si vous n'exécutez pas la version d'Oracle Solaris 11.2 SRU10, utilisez les instructions de la configuration Havana à la place. Reportez-vous au [Chapitre 3, Installation sur plusieurs systèmes pour une configuration Havana multinoeud OpenStack](#)
- 

### Présentation de l'architecture à trois noeuds

En général, vous installez et configurez OpenStack sur plusieurs systèmes ou noeuds. Les configurations monoeud permettent de tester OpenStack comme produit et de vous familiariser avec ses fonctionnalités. Toutefois, une configuration monoeud n'est pas adaptée à un environnement de production.

Chaque Cloud n'a besoin que d'une instance de tableau de bord, d'un magasin d'images et d'un service d'identité. Chaque Cloud peut avoir n'importe quel nombre d'instances de stockage et de

calcul. Dans un environnement de production, ces services sont configurés sur plusieurs noeuds. Évaluez chaque composant par rapport à vos besoins pour un déploiement Cloud particulier afin de déterminer si ce composant doit être installé sur un noeud distinct, ainsi que le nombre de noeuds de chaque type dont vous avez besoin.

- Noeud de contrôleur – noeud où la plus grande partie des services partagés OpenStack et d'autres outils sont exécutés. Le noeud de contrôleur fournit les API, la planification et d'autres services partagés pour le Cloud. Le noeud de contrôleur contient le tableau de bord, le magasin d'images et le service d'identité. En outre, le service de gestion des calculs Nova ainsi que le serveur Neutron sont également configurés dans ce noeud.
- Noeud de calcul - noeud où les instances de machine virtuelle (instances de calcul Nova) sont installées. Le noeud exécute le démon de calcul qui gère ces instances de machine virtuelle.
- Noeud de stockage – noeud qui héberge les données.

L'architecture décrite dans ce chapitre est déployée sur les trois systèmes suivants :

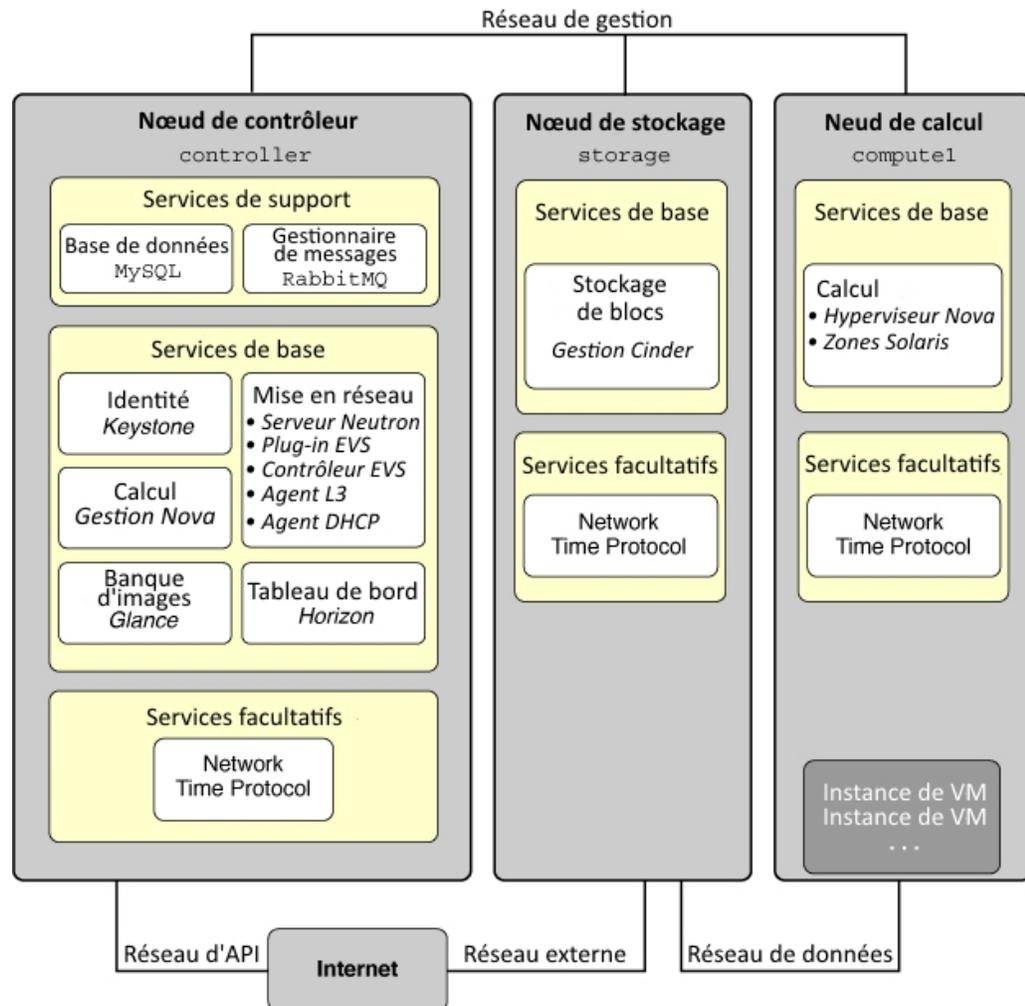
---

**Remarque** - Cette documentation décrit une architecture déployée sur des systèmes physiques distincts. Pour partitionner un serveur Oracle SPARC unique et configurer OpenStack en multinoeud sur le serveur exécutant OVM Server for SPARC (LDoms), reportez-vous à [Multi-node Solaris 11.2 OpenStack on SPARC Servers](#). L'article traite particulièrement de la version Havana d'OpenStack. Toutefois, les étapes générales s'appliquent également à la version actuelle.

---

La figure suivante illustre une vue de haut niveau de l'architecture décrite dans ce chapitre.

FIGURE 4-1 Architecture de référence de la configuration à trois noeuds



Cet exemple d'architecture ne présente pas le service de stockage d'objets Swift. Pour obtenir des informations générales sur la configuration de Swift, voir les informations sur le site de la communauté OpenStack telles que la [Référence de configuration OpenStack](#). Pour plus d'informations sur la procédure de configuration des services SWIFT sur les systèmes Oracle Solaris, et pour des informations supplémentaires sur OpenStack sur Oracle Solaris, reportez vous à [OpenStack for Oracle Solaris 11](#).

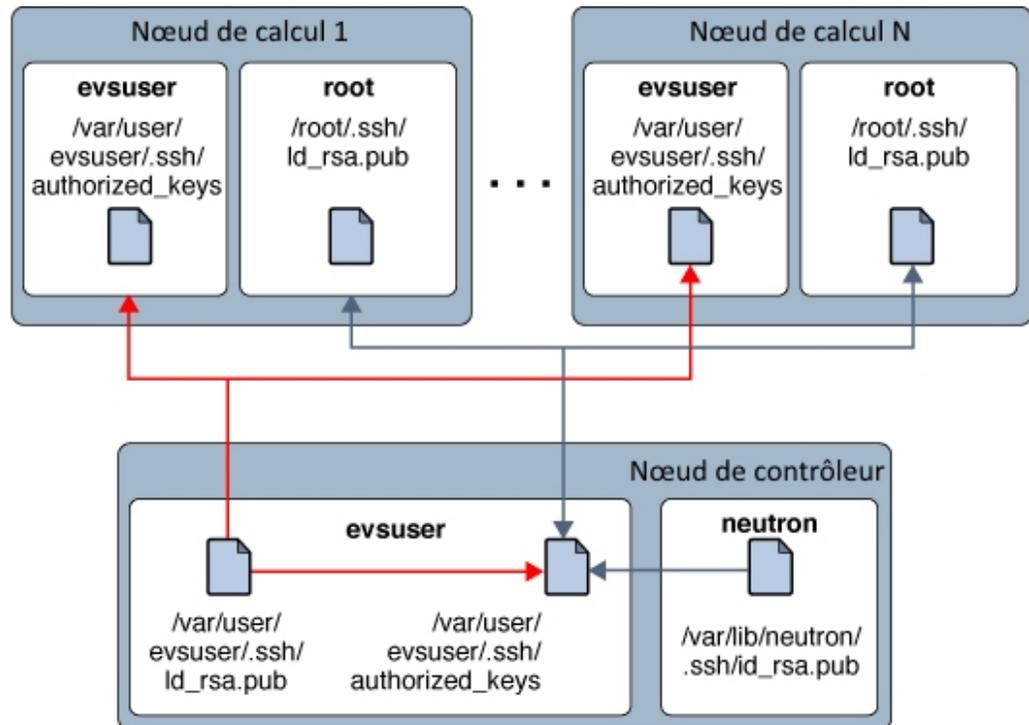
Dans Oracle Solaris, le commutateur virtuel élastique (EVS) forme le back-end des réseaux OpenStack. EVS facilite la communication entre les instances de machine virtuelle qui sont

sur des réseaux locaux virtuels ou des réseaux locaux virtuels extensibles. Les instances VM peuvent se trouver sur le même noeud de calcul ou sur plusieurs noeuds de calcul. Pour plus d'informations sur EVS, reportez-vous au [Chapitre 5, “ A propos des commutateurs virtuels élastiques ”](#) du manuel “ [Gestion de la virtualisation réseau et des ressources réseau dans Oracle Solaris 11.2](#) ”.

Pour permettre aux différents noeuds de communiquer les uns avec les autres, les clés publiques SSH de `evsuser`, `neutron` et `root` dans le noeud de contrôleur doivent se trouver dans chaque fichier `authorized_keys` de l'utilisateur `evsuser` dans tous les noeuds de calcul configurés. Reportez-vous à l'image suivante qui illustre la répartition des clés publiques SSH. L'image suppose que plusieurs noeuds de calcul ont été configurés.

Pour une liste des paramètres de configuration utiles pour les déploiements de {ENT OST} sur les systèmes Oracle Solaris, reportez-vous à « Paramètres communs de configuration pour OpenStack » dans la rubrique <http://www.oracle.com/technetwork/articles/servers-storage-admin/getting-started-openstack-os11-2-2195380.html>.

**FIGURE 4-2** Répartition des clés SSH du contrôleur EVS



## Configuration du noeud de contrôleur

Le noeud de contrôleur comporte un service de tableau de bord, un magasin d'images et un service d'identité. Ce noeud inclut également MySQL, RabbitMQ, ainsi que des services de calcul, de stockage de bloc et de mise en réseau.

La liste suivante indique les tâches de configuration du noeud de contrôleur :

- [“Étapes préliminaires” à la page 81](#)
- [“Installation NTP \(Network Time Protocol\)” à la page 82](#)
- [“Installation de MySQL” à la page 83](#)
- [“Installation de Keystone” à la page 85](#)
- [“Installation de Glance” à la page 86](#)
- [“Installation de Nova” à la page 88](#)
- [“Installation de Horizon” à la page 89](#)
- [“Installation de Cinder” à la page 91](#)
- [“Installation et configuration de Neutron” à la page 94](#)
- [“Installation et configuration de Heat” à la page 97](#)

### Étapes préliminaires

Pour préparer l'implémentation de la configuration OpenStack multinoeud, tenez compte des éléments suivants :

- Si vous utilisez plusieurs interfaces de réseau dans un noeud ou système OpenStack unique, créez des noms d'hôte pour ces interfaces de réseau.

Par exemple, vous aurez peut-être besoin d'interfaces distinctes pour gérer les services administratifs d'OpenStack et le trafic des API (réseau OpenStack), le trafic entre les noeuds de calcul et le routeur L3 (réseau client) et le trafic avec le réseau de plus grande taille hors de votre configuration Cloud (réseau externe). Vous devrez, par conséquent, créer les noms d'hôte correspondants, tels que `host-on`, `host-tn` et `host-en`.

Assurez-vous que tous ces noms d'hôte et adresses IP sont inclus dans le fichier `/etc/hosts` du noeud ou dans la configuration du système de noms de domaine.

- Pour faciliter la configuration des services d'OpenStack, définissez les variables telles que :
  - `$CONTROLLER_ADMIN_NODE` - nom d'hôte de l'interface ou de l'adresse IP dans le noeud de contrôleur auquel les services administratifs d'OpenStack sont associés.
  - `$CONTROLLER_ADMIN_NODE_IP` - adresse IP du port de contrôleur qui gère les services administratifs et le trafic d'OpenStack.
  - `$COMPUTE_ADMIN_NODE_IP` - adresse IP du port de calcul qui gère les services administratifs et le trafic d'OpenStack.

- \$VOLUME\_IP - nom d'hôte du noeud de stockage.
- Assurez-vous également que les mots de passe sont disponibles pour affectation le cas échéant.

Pour une meilleure gestion de l'utilisation de la mémoire entre ZFS et les applications dans Oracle Solaris 11, définissez le paramètre `usr_reserve_hint_pct` sur le noeud, comme indiqué dans l'exemple suivant :

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

où *site* peut correspondre à votre entreprise.

Vous devez également définir ce paramètre sur les différents noeuds OpenStack.

Pour plus d'informations sur ce paramètre, connectez-vous à votre compte MOS à l'adresse <https://support.oracle.com> et consultez le Document 1663862.1, *Memory Management Between ZFS and Applications in Oracle Solaris 11.2*.

## Installation NTP (Network Time Protocol)

L'installation de NTP (Network Time Protocol) est facultative, mais vivement recommandée. Si vous installez NTP service, installez-le sur chaque noeud de service de votre déploiement Cloud.

NTP permet de garantir la cohérence de l'heure du jour sur tous les noeuds du Cloud. Si vous activez NTP dans un réseau, configurez les noeuds de service pour obtenir l'heure sur laquelle ils sont sur le réseau.

- Si la multidiffusion est activée sur les sous-réseaux IP sur lesquels résident les noeuds de service, vous pouvez tirer parti du réseau IP pour utiliser la multidiffusion afin de configurer NTP.
- Si la multidiffusion n'est pas activée sur les sous-réseaux IP sur lesquels résident les noeuds de service, configurez NTP manuellement.

### ▼ Installation et configuration de NTP (Network Time Protocol)

**Avant de commencer**

Si vous ne voulez pas utiliser la multidiffusion pour le protocole NTP, vous devez configurer un serveur NTP.

#### 1. Installez le package NTP.

```
controller# pkg install ntp
```

#### 2. Installez le fichier de configuration.

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

**3. (Facultatif) Configurez le fichier `/etc/inet/ntp.conf` si vous voulez utiliser la multidiffusion.**

**a. Mettez en commentaire l'option `multicastclient`.**

**b. Annulez les commentaires sur une ou plusieurs options du serveur et indiquez le nom du serveur NTP ou son adresse IP.**

Le fichier de configuration doit ressembler à l'exemple suivant :

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

**4. Activez le service SMF du serveur NTP.**

```
controller# svcadm enable ntp
```

## Installation de MySQL

De nombreux services OpenStack permettent à une base de données d'assurer le suivi des ressources critiques, leur utilisation et d'autres informations. Par défaut, les bases de données individuelles SQLite sont indiquées dans ce but, et s'avèrent utiles pour la configuration mononoeud. Pour les configurations multinoeuds, notamment dans un environnement de production, d'autres bases de données, telles que la base de données MySQL, sont recommandées pour le stockage de ces informations.

La communication entre les services OpenStack est effectuée par le protocole AMQP (Advanced Message Queuing Protocol). Sous Oracle Solaris, AMQP est implémenté par RabbitMQ. Le service RabbitMQ est obligatoire. Un seul noeud du Cloud est généralement configuré pour exécuter RabbitMQ. Dans cette architecture, RabbitMQ est configuré pour s'exécuter sur le noeud de contrôleur.

### ▼ Installation d'une base de données MySQL

**1. Installez le serveur et les packages MySQL.**

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

**2. Activez les services RabbitMQ.**

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

- 3. (Facultatif) Si vous utilisez une adresse IP dédiée pour l'administration et le trafic des API, ajoutez cette adresse dans le fichier /etc/mysql/5.5/my.cnf :**

```
bind-address=${CONTROLLER_ADMIN_NODE_IP}
```

- 4. Activez le service MySQL.**

```
controller# svcadm enable mysql
```

- 5. Définissez le mot de passe root du serveur MySQL.**

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

- 6. Configurez MySQL.**

Créez les tables qui seront utilisées par OpenStack. Accordez les privilèges aux services sur le noeud de contrôleur afin de fournir un accès exclusif à ces bases de données.

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

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

## Installation de Keystone

Le service Keystone doit être installé et configuré sur le noeud de contrôleur.

### Exemple de script Keystone

Pour alimenter rapidement la base de données Keystone, vous pouvez utiliser un exemple de script `/usr/demo/openstack/keystone/sample_data.sh`. Le script crée les locataires initiaux suivants :

- `service`, sous lequel un utilisateur Keystone est créé pour chaque service OpenStack.
- `demo`, sous lequel l'utilisateur `admin` est créé avec un mot de passe par défaut.

En outre, le script configure les variables d'environnement qui définissent le noeud sur lequel réside chaque service d'API ainsi que le mot de passe de chaque service. Par défaut, pour chacun des services du locataire `service`, le nom du service, le nom d'utilisateur et le mot de passe sont identiques. Par exemple, pour le service Nova, l'utilisateur `nova` est créé avec le mot de passe `nova`.

Vous pouvez modifier les variables avant d'exécuter le script pour personnaliser les noms des utilisateurs du service et les mots de passe correspondants, ainsi que le nom des locataires initiaux. Consultez le script pour plus d'informations sur les paramètres qui peuvent être définis pour l'environnement.

## ▼ Installation et configuration de Keystone

### 1. Installez le package Keystone.

```
controller# pkg install keystone
```

**2. Créez le jeton partagé de Keystone et d'autres services OpenStack.**

Le jeton est composé d'une chaîne aléatoire de caractères.

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

**3. Configurez le jeton dans une variable d'environnement.**

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

**4. Modifiez les paramètres dans le fichier /etc/keystone/keystone.conf.**

La configuration doit ressembler à l'exemple suivant :

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

**5. Activez le service MSF Keystone.**

```
controller# svcadm enable keystone
```

**6. Générez des jetons PKI (Public Key Infrastructure).**

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

**7. Alimenter la base de données Keystone.**

Pour utiliser l'exemple de script, exécutez la commande suivante.

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

## Installation de Glance

Le paramétrage de Glance nécessite la configuration de certaines informations d'authentification, ainsi que l'emplacement des services MySQL et RabbitMQ.

### ▼ Configuration et installation de Glance

**1. Installez le package Glance.**

```
controller# pkg install glance
```

## 2. Configurez Glance en supprimant les commentaires ou en définissant les paramètres dans les fichiers de configuration suivants :

- /etc/glance/glance-api.conf

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

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

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

- /etc/glance/glance-cache.conf

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

- /etc/glance/glance-registry.conf

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

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

[keystone_authtoken]
```

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

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

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

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

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

- /etc/glance/glance-scrubber.conf

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

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

### 3. Activez les services SMF Glance.

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

## Installation de Nova

La configuration de Nova sur le noeud de contrôleur nécessite également de configurer les informations habituelles d'authentification et de service. Cette section traite de la configuration des services de point d'extrémité Nova et non du noeud de calcul.

## ▼ Installation et configuration de Nova

### 1. Installez le package Nova.

```
controller# pkg install nova
```

### 2. Configurez Nova en supprimant les commentaires ou en définissant les paramètres dans le fichier `/etc/nova/nova.conf` :

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

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

[glance]
host=${CONTROLLER_ADM_NODE}

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

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

### 3. Activez les services SMF Nova.

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

## Installation de Horizon

Horizon sert de portail Web pour OpenStack. La procédure suivante utilise l'accès HTTP standard plutôt que SSL/TLS.

## ▼ Configuration d'Horizon

### 1. Installez le pack Horizon.

```
controller# pkg install horizon
```

### 2. Exécutez l'un des groupes d'étapes suivants en fonction de votre configuration.

#### ■ Votre configuration utilise HTTP.

1. Indiquez des paramètres pour le script `/etc/openstack_dashboard/local_settings.py`.

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

2. Copiez la version HTTP du fichier `http.conf` pour OpenStack.

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

#### ■ Votre configuration utilise SSL/TLS.

1. Générez les certificats à utiliser par Horizon.

Les commandes suivantes génèrent des certificats auto-signés pour Horizon et copient le fichier de configuration du tableau de bord OpenStack sur le répertoire de fichiers de configuration Apache. Pour plus d'informations sur la création de certificats autosignés, reportez-vous à la foire aux questions Apache : [SSL/TLS Strong Encryption: FAQ](#).

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

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

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

2. Dans le fichier `~/conf.d/openstack-dashboard-tls.conf`, indiquez l'adresse du site et le nom de serveur du package Horizon sur les paramètres suivants :

```
RedirectPermanent=controller-IP
```

```
ServerName=controller-name
```

### 3. Démarrez le service Apache avec l'une des commandes suivantes.

```
controller# svcadm enable apache22
```

## Installation de Cinder

La configuration Cinder doit indiquer au moins les informations suivantes

- Informations d'autorisations pour authentification auprès de Keystone.
- Classe des volumes à créer.

### ▼ Installation et configuration de Cinder

Les étapes de cette procédure s'appliquent à la configuration des services de point d'extrémité Cinder et non au noeud Cinder ou de volume.

#### 1. Installez le package Cinder.

```
controller# pkg install cinder
```

#### 2. Configurez Cinder en supprimant les commentaires ou en définissant les paramètres dans le fichier `/etc/cinder/cinder.conf`.

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

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

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

#### 3. Activez les services SMF Cinder.

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

Voir aussi Reportez-vous également à [How to Build OpenStack Block Storage on ZFS](#).

## ▼ Configuration du pilote ZFS Storage Appliance iSCSI Cinder

Le pilote Oracle ZFS Storage Appliance iSCSI Cinder permet au ZFSSA (Oracle ZFS Storage Appliance) d'être utilisé de façon transparente sous la forme d'une ressource de stockage en mode bloc pour Cinder. Le pilote permet de créer des volumes iSCSI pouvant être alloués par un serveur Cinder à toute machine virtuelle instanciée par le service Nova. Le pilote est distribué par le package `cloud/openstack/cinder`. Votre appareil doit exécuter au moins la version logicielle 2013.1.2.0. de ZFSSA.

### Avant de commencer

Configuration d'un pool sur Oracle ZFS Storage Appliance. Vous pouvez décider d'utiliser un pool préexistant.

#### 1. Exécutez le workflow `cinder.akwf`.

Vous pouvez utiliser un utilisateur existant ou créer un nouvel utilisateur avec les autorisations de rôle pour effectuer les opérations du pilote Cinder.

Le workflow `cinder.akwf` effectue les tâches suivantes

- Création de l'utilisateur s'il n'existe pas.
- Définition des autorisations de rôle pour effectuer les opérations du pilote Cinder.
- Activation du service RESTful si le service est désactivé.

Vous pouvez exécuter le workflow dans l'interface de ligne de commande (CLI) ou à partir de l'interface utilisateur de navigateur (BUI) de votre appareil.

#### ■ Exécutez le workflow à partir de la CLI.

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

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

zfssa:maintenance workflows> ls
Properties:
    showhidden = false

Workflows:

WORKFLOW   NAME                               OWNER SETID ORIGIN
VERSION
```

```

workflow-000 Clear locks                                root false Oracle Corporation
1.0.0
workflow-001 Configuration for OpenStack Cinder Driver root false Oracle Corporation
1.0.0

zfssa:maintenance workflows> select workflow-001

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

```

- **Exécutez le workflow à partir de la BUI.**
  - a. **Sélectionnez Maintenance -> Workflow, et utilisez l'icône plus (+) pour télécharger un nouveau workflow.**
  - b. **Cliquez sur le bouton Parcourir et sélectionnez le fichier cinder.akwf.**
  - c. **Cliquez sur le bouton TELECHARGER pour terminer le téléchargement du workflow.**
  - d. **Cliquez sur la nouvelle ligne qui apparaît sur la page Workflows BUI pour exécuter le workflow du pilote Cinder.**

Le workflow invite à entrer un nom d'utilisateur et un mot de passe. Ce nom d'utilisateur et ce mot de passe seront également utilisés dans le fichier `cinder.conf` en tant que `san_login` et `san_password`.

## 2. Définissez les paramètres dans le fichier `/etc/cinder/cinder.conf`.

Indiquez les propriétés obligatoires suivantes dans le fichier `cinder.conf`

- `volume_driver` – Assurez-vous que `cinder.volume.drivers.zfssa.zfssaiscsi.ZFSSAISCSIDriver` n'est pas commentée. Assurez-vous que les trois autres sélections sont commentées.
- `san_ip` – Nom ou adresse IP de l'hôte de gestion ZFSSA.
- `san_login` – Nom de l'utilisateur Cinder sur le ZFSSA.
- `san_password` – Mot de passe de l'utilisateur Cinder sur le ZFSSA.
- `zfssa_pool` – Le pool à utiliser pour allouer les volumes.
- `zfssa_target_portal` – Portail cible iSCSI ZFSSA (`data-ip:port`). Le port par défaut est 3260.
- `zfssa_project` – Nom du projet ZFSSA. Si le projet n'existe pas sur l'appareil, un projet avec le même nom est créé par le pilote au démarrage. Ce projet contient tous les volumes

créés par le pilote. Des propriétés ZFSSA supplémentaires sont utilisées pour définir les caractéristiques des volumes (la taille de bloc, par exemple) et leur accès (par exemple, initiateurs, cibles, sécurité).

- `zfssa_initiator_group` – Nom du groupe d'initiateurs. Si le groupe d'initiateurs n'existe pas sur l'appareil, un groupe d'initiateurs avec le même nom est créé par le pilote au moment du démarrage. Si vous souhaitez utiliser le groupe d'initiateurs `default`, définissez la valeur de ce paramètre avec sa valeur `default`. Le groupe d'initiateurs `default` peut s'avérer utile à des fins d'évaluation. Le groupe d'initiateurs `default` ne doit normalement pas être utilisé, car cela risque d'exposer les volumes à des initiateurs non souhaités ou incompatibles.
- `zfssa_target_interfaces` – Les interfaces de réseau des cibles iSCSI ZFSSA. Exécutez la commande suivante pour afficher les 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` – Définissez le paramètre comme suit :

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

### 3. Vérifiez que le service ZFSSA iSCSI est en ligne.

Si le service ZFSSA iSCSI n'est pas en ligne, utiliser la BUI ou la CLI dans l'appareil pour l'activer. L'exemple suivant montre comment utiliser la CLI dans l'appareil

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

### 4. Activez les services SMF de volume Cinder.

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

## Installation et configuration de Neutron

Dans l'architecture décrite dans ce chapitre, le service de l'API Neutron s'exécute sur le noeud de contrôleur.

### ▼ Installation et configuration de Neutron

#### 1. Installez le package Neutron.

```
controller# pkg install neutron
```

## 2. Configurez Neutron en supprimant les commentaires ou en définissant les paramètres dans les fichiers de configuration suivants :

- /etc/neutron/neutron.conf

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

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

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

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

- /etc/neutron/plugins/evs/evs\_plugin.ini

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

- /etc/neutron/dhcp\_agent.ini

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

## 3. Configurez le commutateur virtuel élastique (EVS).

- a. Configurez la propriété EVS pour indiquer l'emplacement du contrôleur EVS.

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

- b. Créez la paire de clés SSH pour les utilisateurs evsuser, neutron et root.

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

- c. Combinez les clés SSH des utilisateurs evsuser, neutron et root dans le fichier authorized\_keys de l'utilisateur evsuser.

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

**d. Testez les connexions SSH pour accepter les empreintes qui seront stockées dans le fichier `known_host`.**

Sélectionnez Oui à chaque invite de confirmation.

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

**e. Configurez le responsable des répertoires `.ssh`.**

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

**f. Configurez les propriétés `l2-type`, `uplink-port` et `vlan-range` du contrôleur EVS.**

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

L'exemple suivant décrit la configuration de ces propriétés. Vous pouvez éventuellement utiliser la commande finale pour afficher toutes les propriétés EVS :

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

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

**4. Activez le transfert IP.**

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

**5. Démarrez le service de filtre IP.**

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

**6. Activez le service de serveur Neutron.**

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

## Installation et configuration de Heat

Heat est un moteur d'orchestration OpenStack qui permet de déployer des applications pour le cloud basées sur des modèles que vous créez. Vous installez Heat sur le même noeud que Keystone.

### ▼ Configuration de Heat

Avant de commencer

Vous devez d'abord configurer Keystone comme décrit à la section [“Installation et configuration de Keystone”](#) avant d'effectuer cette tâche.

#### 1. Installez le pack Heat.

```
controller# pkg install heat
```

#### 2. Exécutez le script de configuration de Heat.

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

#### 3. Configurez Heat en supprimant les commentaires ou en définissant les paramètres dans les fichiers de configuration suivants :

- /etc/heat/heat.conf

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

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

- /etc/heat/api-paste.ini

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

#### 4. Activez le service Heat.

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

## Configuration du noeud de calcul

Vous installez les instances de machine virtuelle dans le noeud de calcul, tout comme le démon nova-compute. Les instances VM fournissent une large gamme de services telles que des applications Web et des analyses. Vous pouvez configurer autant de noeuds de calcul que nécessaire dans votre Cloud.

Pour une meilleure gestion de l'utilisation de la mémoire entre ZFS et les applications dans Oracle Solaris 11, définissez le paramètre `usr_reserve_hint_pct` sur le noeud, comme indiqué dans l'exemple suivant :

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

où *site* peut correspondre à votre entreprise.

Vous devez également définir ce paramètre sur les différents noeuds OpenStack.

Pour plus d'informations sur ce paramètre, connectez-vous à votre compte MOS à l'adresse <https://support.oracle.com> et consultez le Document 1663862.1, *Memory Management Between ZFS and Applications in Oracle Solaris 11.2*.

### ▼ Configuration du noeud de calcul

1. **(Facultatif) Installez et configurez NTP.**  
Reportez-vous à la section “[Installation NTP \(Network Time Protocol\)](#)” à la page 47.

2. **Installez le package Nova.**

```
compute1# pkg install nova
```

3. **Redémarrez RAD (Démon d'accès distant).**  
Nova utilise RAD pour communiquer avec la structure Oracle Solaris Zones.

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

4. **Configurez Nova en supprimant les commentaires ou en définissant les paramètres suivants dans le fichier `/etc/nova/nova.conf`.**

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

```

my_ip=$COMPUTE_ADMIN_NODE_IP
host=$COMPUTE_ADMIN_NODE_X
firewall_driver=nova.virt.firewall.NoopFirewallDriver
keystone_ec2_url=http://$CONTROLLER_ADM_NODE:5000/v2.0/ec2tokens

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

[glance]
host=$CONTROLLER_ADM_NODE

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

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

```

## 5. Paramétrez EVS dans le noeud de calcul.

### a. Installez le package EVS.

```
compute1# pkg install evs
```

### b. Configurez la propriété EVS pour indiquer l'emplacement du contrôleur EVS.

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

## 6. Configurez les communications entre le contrôleur et les noeuds de calcul.

### a. Créez une clé publique SSH sur le noeud de calcul pour l'utilisateur root.

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

### b. (Facultatif) Vérifiez le contenu de la clé SSH.

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

### c. Copiez la clé SSH /root/.ssh/id\_rsa.pub dans un emplacement du noeud de contrôleur.

### d. Sur le noeud de contrôleur, ajoutez la clé SSH au fichier authorized\_keys pour l'utilisateur evsuser.

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

- e. **(Facultatif) Vérifiez que la clé SSH issue du noeud de calcul a été ajoutée au fichier `authorized_keys`.**

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

La sortie doit inclure le contenu de la clé SSH que vous avez générée dans le noeud de calcul et que vous avez affichée à l'[Étape 6.b](#).

- f. **Testez les connexions SSH du noeud de calcul au contrôleur et acceptez le stockage des empreintes dans le fichier `known_host`.**

Sélectionnez Oui à l'invite de confirmation.

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

7. **Activez le service de calcul Nova.**

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

## Configuration du noeud de stockage

Le noeud de stockage constitue le référentiel de toutes les données traitées dans le paramétrage d'OpenStack.

Pour une meilleure gestion de l'utilisation de la mémoire entre ZFS et les applications dans Oracle Solaris 11, définissez le paramètre `usr_reserve_hint_pct` sur le noeud, comme indiqué dans l'exemple suivant :

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

où `site` peut correspondre à votre entreprise.

Vous devez également définir ce paramètre sur les différents noeuds OpenStack.

Pour plus d'informations sur ce paramètre, connectez-vous à votre compte MOS à l'adresse <https://support.oracle.com> et consultez le Document 1663862.1, *Memory Management Between ZFS and Applications in Oracle Solaris 11.2*.

### ▼ Configuration du noeud de stockage de bloc

1. **Installez les packages appropriés.**

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

## 2. Configurez Cinder en supprimant les commentaires ou en définissant les paramètres dans le fichier `/etc/cinder/cinder.conf`.

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

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

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

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

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

## 3. Démarrez les services Cinder.

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

# Configuration d'un réseau interne dans OpenStack

Sur un réseau interne, les instances de machine virtuelle que vous créez ultérieurement ne peuvent communiquer qu'entre elles. Ces instances ne peuvent pas se connecter au réseau élargi ou sont accessibles de façon externe.

---

**Remarque** - Pour configurer un réseau externe pour un locataire, reportez-vous à la rubrique [“Configuration d'OpenStack avec un réseau externe”](#) à la page 103.

---

## ▼ Création d'un réseau externe

Effectuez ces étapes sur le noeud où sont exécutés les services Neutron, qui est, dans le cas présent, le noeud de contrôleur.

### 1. Affichez la liste des locataires existants.

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

La liste des locataires contient les locataires et leurs identifiants. Sélectionnez l'identifiant du locataire pour lequel vous allez créer le réseau interne.

### 2. Créez le réseau pour le locataire sélectionné.

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

où *tenant-ID* est issu de la liste des locataires générée à l'étape précédente.

### 3. Créez le sous-réseau pour le même locataire.

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

#### Exemple 4-1 Création du réseau interne

Cet exemple illustre la création du réseau interne pour le locataire demo qui a été créé par défaut par l'exemple de script Keystone dans [“Exemple de script Keystone”](#) à la page 85.

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

controller# neutron net-create --tenant-id abcde12345 demo_internal_net
Created a new network:
+-----+-----+
| Field                | Value                |
+-----+-----+
| admin_state_up       | True                 |
| id                   | 9999                 |
| name                 | demo_internal_net   |
| provider:network_type | vlan                 |
| provider:segmentation_id | 300                 |
| router:external      | False                |
| shared               | False                |
| status               | ACTIVE               |
| subnets             |                      |
+-----+-----+
```

```

| tenant_id          | abcde12345          |
+-----+-----+
controller# neutron subnet-create --name demo_int_subnet --tenant-id abcde12345 \
demo_internal_net 192.168.1.0/24
Created a new subnet:
+-----+-----+
| Field              | Value               |
+-----+-----+
| allocation_pools  | {"start": "192.168.1.2", "end": "192.168.1.254"} | |
| cidr              | 192.168.1.0/24     |
| dns_nameservers   |                     |
| enable_dhcp       | True                |
| gateway_ip        | 192.168.1.1        |
| host_routes       |                     |
| id                | 07f9b37c-ae4e-11e4-8000-db57d0041a2c |
| ip_version        | 4                   |
| name              | demo_int_subnet     |
| network_id        | 99999|              |
| tenant_id         | abcde12345         |
+-----+-----+

```

## Configuration d'OpenStack avec un réseau externe

Vous créez un réseau externe pour permettre aux réseaux privés du Cloud de communiquer avec le réseau élargi. Dans le Cloud, un locataire peut avoir un ou plusieurs réseaux privés. Lorsque vous créez un réseau externe pour le Cloud, vous créez un routeur de fournisseur partagé par tous les réseaux du locataire. En tant qu'administrateur, vous créez, possédez et gérez ce routeur. Le routeur n'est pas visible dans la vue de la topologie réseau du locataire. Etant donné qu'il n'y a qu'un routeur, les réseaux locataires ne peuvent pas utiliser le chevauchement des adresses IP.

La création d'un réseau externe implique également la configuration de l'agent Neutron L3. L'agent Neutron L3 crée automatiquement des mappages NAT bi-univoques entre les adresses affectées aux instances Nova et les adresses IP flottantes. L'agent L3 assure également la communication entre les réseaux privés. Par défaut, le routage entre des réseaux privés qui appartiennent au même locataire est désactivé. Pour modifier ce comportement, définissez `allow_forwarding_between_networks` sur `True` dans le fichier de configuration `/etc/neutron/l3_agent.ini` et redémarrez le service SMF `neutron-l3-agent`.

Le routeur fournit la connectivité des instances de machine virtuelle du locataire au monde extérieur. Le routeur effectue une NAT bidirectionnelle sur l'interface qui connecte le routeur au réseau externe. Les locataires créent autant d'adresses IP flottantes que nécessaire ou qu'autorisées par le quota d'IP flottantes, puis associent ces IP flottantes aux instances de machine virtuelle qui requièrent une connectivité avec l'extérieur.

Pour illustrer la relation entre les réseaux internes et le réseau externe dans OpenStack, reportez-vous à la [Figure 3-3, "Routeur de fournisseur avec modèle de réseaux privés"](#).

## ▼ Configuration du réseau externe dans OpenStack

Cette procédure explique comment créer un réseau virtuel qui représente un réseau externe. Sur ce réseau virtuel, le DHCP n'est pas utilisé. A la place, des adresses IP flottantes sont créées. Ces adresses IP flottantes sont affectées à un locataire spécifique et peuvent être affectées aux instances des instances de machine virtuelle Nova pour que leurs utilisateurs sous ce locataire puissent les utiliser.

Comme le contrôleur inclut le service Neutron dans l'architecture multinoeud de démonstration de ce chapitre, vous effectuez les étapes suivantes dans le noeud de contrôleur.

**Avant de commencer**

Pour exécuter cette procédure, des réseaux internes doivent déjà exister pour le locataire. Pour consulter les procédures de création d'un réseau privé, reportez-vous à la rubrique [“Création d'un réseau externe” à la page 102](#).

En outre, vous devez avoir terminé la configuration du commutateur virtuel élastique, notamment les propriétés `l2-type` et `vlan-range`. L'exemple suivant décrit comment afficher ces paramètres de propriété une fois que vous les avez configurés à l'[Étape 9](#) :

```

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

Vous devez également fournir les informations relatives au locataire dont vous connectez les réseaux privés au réseau externe. Vous pouvez afficher ces informations à tout moment à l'aide de la commande suivante :

```
keystone tenant-list
```

### 1. Activez le filtre IP Solaris.

```
controllor# svcadm enable ipfilter
```

### 2. Activez le transfert d'IP sur tout l'hôte.

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

### 3. Configurez les variables d'environnement adéquates.

```

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

### 4. Créez le routeur de fournisseur.

```
controllor# neutron router-create router-name
```

La commande affiche le nom du routeur avec le code correspondant. Utilisez ce code pour mettre à jour le fichier de configuration à l'étape suivante.

**5. Mettez à jour le fichier de configuration de l'agent L3.**

Dans le fichier `/etc/neutron/l3_agent.ini`, configurez la valeur du paramètre `router_id` avec l'identificateur unique universel du routeur à l'étape précédente.

```
router_id = router-ID
```

**6. Activez le service SMF `neutron-l3-agent`.**

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

**7. Créez un réseau externe.**

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

où la valeur de `segmentation_id` est le premier chiffre de la plage de VLAN.

**8. Créez un sous-réseau et associez-le au réseau externe.**

Pour l'exemple de configuration de ce document, DHCP est désactivé. Le pool d'allocation correspond à une plage d'adresses IP flottantes affectée au sous-réseau.

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

**9. Ajoutez le réseau externe au routeur.**

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

---

**Remarque** - Vous pouvez récupérer `router-ID` dans le fichier `/etc/neutron/l3_agent.ini`. Au besoin, vous pouvez utiliser la commande `neutron net-list` pour récupérer `network-ID`.

---

**10. Ajoutez le réseau privé du locataire au routeur.**

Pour cette étape, vous devez vous munir du code du routeur et du code du sous-réseau du locataire. Vous pouvez obtenir ces informations comme suit :

- a. Pour récupérer le code du sous-réseau du locataire, affichez d'abord les locataires et leurs codes, puis les sous-réseaux du code du locataire concerné.**

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

- b. Ajoutez le réseau privé au routeur.**

Répétez cette étape pour chaque réseau privé du locataire que vous voulez ajouter au réseau externe.

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

**Exemple 4-2** Création du réseau externe pour le locataire service

L'exemple suivant utilise une partie de la [Figure 3-3, "Routeur de fournisseur avec modèle de réseaux privés"](#) comme référence. Dans la figure, le locataire Tenant A est associé à deux instances de machine virtuelle, chacune appartenant à un réseau privé. Les deux réseaux privés sont HR et ENG. Ces deux sous-réseaux seront ajoutés au réseau externe pour permettre aux deux instances de machine virtuelle de communiquer avec le réseau élargi. Dans cet exemple, le nom d'utilisateur et le mot de passe par défaut sont utilisés pour le service Neutron.

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

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

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

A ce stade, mettez à jour le fichier `/etc/neutron/l3_agent.ini` en définissant le paramètre `router_ID` avec la valeur `97ro5-ut3er`.

Après avoir modifié le fichier, poursuivez avec les étapes restantes.

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

controller# neutron net-create --provider:network_type=vlan \
--provider:segmentation_id=1 --router:external=true ext_network
Created a new network:
+-----+-----+
| Field          | Value          |
+-----+-----+
| admin_state_up | True           |
| id             | 555ext-net555  |
| name           | ext_network    |
| provider:network_type | vlan          |
+-----+-----+
```

```
| provider:segmentation_id | 1 |
| router:external         | True |
| shared                  | False |
| status                  | ACTIVE |
| subnets                | |
| tenant_id               | abcde12345 |
+-----+-----+-----+
```

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

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

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

```
controller# keystone tenant-list
+-----+-----+-----+
| id      | name | enabled |
+-----+-----+-----+
| 12345abcde | TenantA | True |
| 67890fghij | TenantB | True |
+-----+-----+-----+
```

```
controller# neutron net-list --tenant-id 12345abcde
+-----+-----+-----+
| id      | name | subnets |
+-----+-----+-----+
| 1a3b5c7d9e | HR | xyz-123-uvw |
| 2f4g6h8i0j | ENG | 098-r2d2-56 |
+-----+-----+-----+
```

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

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

- Voir aussi
- [“Respect de la configuration de l'agent L3” à la page 75](#)
  - [“Limitations connues” à la page 121](#)

## ▼ Création et association des adresses IP flottantes en tant qu'utilisateur locataire

Cette procédure est effectuée par un utilisateur locataire avec le tableau de bord Horizon {ENT : OST}.

- 1. Connectez-vous au tableau de bord OpenStack.**  
Connectez-vous, comme indiqué dans la section [“Accès au tableau de bord OpenStack” à la page 28](#), avec les informations d'identification de l'utilisateur locataire.
- 2. Sélectionnez Projet -> Accès et sécurité -> IP flottantes.**
- 3. Sélectionnez le nom du réseau externe.**
- 4. Cliquez sur le bouton Allouer IP.**  
L'onglet IP flottantes affiche l'adresse IP flottante allouée.
- 5. Cliquez sur le bouton Associer.**
- 6. Sélectionnez le port de l'instance de machine virtuelle dans le menu déroulant.**  
La fenêtre Projet -> Instances indique que l'adresse IP flottante est associée à l'instance de machine virtuelle.  
Si vous avez sélectionné une paire de clés (clé publique SSH) lors du lancement d'une instance de machine virtuelle, cette clé SSH sera ajoutée dans le fichier `authorized_keys` pour l'utilisateur `root` de l'instance de machine virtuelle.
- 7. Connectez-vous à l'instance de machine virtuelle en cours d'exécution.**

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

où *IP-address* est l'adresse IP flottante associée à l'instance de machine virtuelle.

Après vous être connecté, vous pouvez utiliser l'instance comme n'importe quel autre système (par exemple, pour installer des applications).

```
global# ssh root@10.134.13.9
Last login: Fri Jul 18 00:37:39 2014 from 10.132.146.13
Oracle Corporation SunOS 5.11 11.2 June 2014
root@host-192-168-101-3:~# uname -a
SunOS host-192-168-101-3 5.11 11.2 i86pc i386 i86pc
root@host-192-168-101-3:~# zoneadm list -cv
```

```

ID NAME          STATUS   PATH          BRAND   IP
  2 instance-0000001 running /          solaris excl
root@host-192-168-101-3:~# ipadm
NAME          CLASS/TYPE STATE   UNDER   ADDR
lo0           loopback  ok     --      --
  lo0/v4      static    ok     --      127.0.0.1/8
  lo0/v6      static    ok     --      ::1/128
net0          ip        ok     --      --
  net0/dhcp   inherited ok     --      192.168.101.3/24

```

## ▼ Respect de la configuration de l'agent L3

Vous pouvez utiliser des commandes de filtre IP, telles que `ipf`, `ippool`, et `ipnat` et des commandes de mise en réseau, telles que `dladm` et `ipadm` pour respecter et résoudre les problèmes de la configuration effectuée par `neutron-l3-agent`.

### 1. Affichez les VNIC créées par `neutron-l3-agent`.

```

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

```

### 2. Affichez les adresses IP créée par `neutron-l3-agent`.

```

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

```

### 3. Affichez les règles de filtrage IP.

```

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

```

### 4. Affichez les règles de filtrage NAT.

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

## Création d'instances de machine virtuelle

---

Ce chapitre décrit la façon de fournir des instances de machines virtuelles (instances VM) au Cloud OpenStack. Chaque instance de machine virtuelle appartient à un locataire. Les utilisateurs peuvent créer et travailler sur au moins une instance de machine virtuelle dans au moins un locataire. Pour des informations générales sur la création et la gestion d'instances de machine virtuelle OpenStack, reportez-vous au [OpenStack End User Guide](#).

Pour créer une instance de machine virtuelle, vous avez besoin d'une variante et d'une image. Ce chapitre présente les sections suivantes :

- “Gestion des variantes” à la page 111
- “Gestion des images” à la page 114
- “Création d'une instance de machine virtuelle” à la page 117

Vous pourrez souhaiter créer un fichier dans lequel vous définirez les variables d'environnement des commandes OpenStack à utiliser. Si vous ne le faites pas, vous devrez spécifier les options à chaque commande. Reportez-vous à [OpenStack Command-Line Interface Reference](#) pour plus d'informations.

### Gestion des variantes

Une *variante* est un type d'instance de machine virtuelle, ou modèle matériel virtuel. Une variante spécifie un ensemble de ressources d'une machine virtuelle, telles que le nombre de CPU virtuelles, la quantité de mémoire, et l'espace disque, affectées à une instance de machine virtuelle. Sur Solaris, la variante inclut la marque de la zone sous-jacente : `solaris` pour les zones non globales et `solaris-kz` pour les zones de noyau. Un exemple de variante d'instance est une zone de noyau avec 16 CPU virtuelles et 16384 Mo de RAM.

Pour obtenir des informations générales sur les variantes, reportez-vous à la section [Variantes](#) du *OpenStack Cloud Administrator Guide*.

## Affichage des informations sur les variantes

La commande `nova flavor-list` affiche la liste des variantes disponibles. Vous utiliserez le nom ou l'ID de la variante lors de la création de l'instance de machine virtuelle. Pour inclure les extra-specs sur la liste de variantes, indiquez l'option `--extra-specs` avec la commande `nova flavor-list`. Pour de plus amples informations sur les `extra_specs`, reportez-vous à [“Modification des spécifications de variante” à la page 113](#).

Dans l'exemple suivant, les colonnes `RXTX_Factor` et `Is_Public` sont omises des sorties pour économiser de l'espace. Reportez-vous à [OpenStack Command-Line Interface Reference](#) pour plus d'informations sur ces colonnes.

```
$ nova flavor-list
+-----+-----+-----+-----+-----+-----+
| ID | Name                               | Memory_MB | Disk | Ephemeral | Swap | VCPUs |
+-----+-----+-----+-----+-----+-----+
| 1  | Oracle Solaris                     |           |     |           |     |      |
|   |   kernel zone - tiny               |    2048   |    10 |    0     |     |     1 |
| 10 | Oracle Solaris                     |           |     |           |     |      |
|   |   non-global zone - xlarge        |   16384   |    80 |    0     |     |    32 |
| 2  | Oracle Solaris                     |           |     |           |     |      |
|   |   kernel zone - small             |    4096   |    20 |    0     |     |     4 |
| 3  | Oracle Solaris                     |           |     |           |     |      |
|   |   kernel zone - medium            |    8192   |    40 |    0     |     |     8 |
| 4  | Oracle Solaris                     |           |     |           |     |      |
|   |   kernel zone - large             |   16384   |    40 |    0     |     |    16 |
| 5  | Oracle Solaris                     |           |     |           |     |      |
|   |   kernel zone - xlarge            |   32768   |    80 |    0     |     |    32 |
| 6  | Oracle Solaris                     |           |     |           |     |      |
|   |   non-global zone - tiny         |    2048   |    10 |    0     |     |     1 |
| 7  | Oracle Solaris                     |           |     |           |     |      |
|   |   non-global zone - small        |    3072   |    20 |    0     |     |     4 |
| 8  | Oracle Solaris                     |           |     |           |     |      |
|   |   non-global zone - medium       |    4096   |    40 |    0     |     |     8 |
| 9  | Oracle Solaris                     |           |     |           |     |      |
|   |   non-global zone - large        |    8192   |    40 |    0     |     |    16 |
+-----+-----+-----+-----+-----+-----+
```

Les commandes suivantes affichent des informations détaillées sur la variante spécifique. La première version de la commande indique le nom de la variante, et la seconde version indique l'ID de la variante. La sortie est identique pour ces deux commandes.

```
$ nova flavor-show 'Oracle Solaris kernel zone - large'
$ nova flavor-show
+-----+-----+-----+-----+-----+-----+
| Property                | Value                               |
+-----+-----+-----+-----+-----+-----+
| name                    | Oracle Solaris kernel zone - large |
| ram                     | 16384                               |
| OS-FLV-DISABLED:disabled | False                               |
| vcpus                   | 16                                  |
+-----+-----+-----+-----+-----+-----+
```

extra_specs	{'zonecfg:brand': 'solaris-kz'}
swap	
os-flavor-access:is_public	True
rxtx_factor	1.0
OS-FLV-EXT-DATA:ephemeral	0
disk	40
id	4

## Modification des spécifications de variante

Pour modifier les spécifications de variante, utilisez la sous-commande `flavor-key` pour modifier la valeur `extra_specs`.

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

*flavor* Le nom ou l'ID de la variante.

*action* set ou unset

*key=value* La *key* est le nom de la spécification. La *value* est la nouvelle valeur pour cette spécification. Si l'*action* est unset, n'indiquez que la *key*.

```
$ nova flavor-key 4 set zonecfg:bootargs=-v
```

```
$ nova flavor-show
```

Property	Value
name	Oracle Solaris
	kernel zone - large
ram	16384
OS-FLV-DISABLED:disabled	False
vcpus	16
extra_specs	{'zonecfg:brand': u'solaris-kz', 'zonecfg:bootargs': u'-v'}
swap	
os-flavor-access:is_public	True
rxtx_factor	1.0
OS-FLV-EXT-DATA:ephemeral	0
disk	40
id	4

Les propriétés suivantes sont prises en charge dans les zones de noyau et les zones non globales

- `zonecfg:bootargs`
- `zonecfg:brand`
- `zonecfg:hostid`

Les propriétés suivantes ne sont prises en charge que dans les zones non globales

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

Les autres propriétés de configuration de zone ne sont pas prises en charge dans OpenStack. Reportez-vous à la page de manuel [zonecfg\(1M\)](#) pour plus d'informations sur ces propriétés de configuration de zone.

Utilisez la clé `sc_profile` pour spécifier un profil de configuration système

```
$ nova flavor-key 4 set sc_profile=/system/volatile/profile/sc_profile.xml
```

La commande `nova flavor-key` vous permet de modifier ou d'ajouter uniquement des valeurs `extra_specs`. Pour modifier d'autres spécifications d'une variante existante, tel que la quantité de RAM, vous devez supprimer la variante et créer une variante modifiée portant le même nom. Pour plus d'informations sur la suppression et la création de variantes, reportez-vous au [OpenStack Admin User Guide](#).

## Gestion des images

Une image de machine virtuelle, ou *image*, est un fichier unique contenant un disque virtuel qui porte un système d'exploitation amorçable installé sur celui-ci. Une image fournit un modèle pour les systèmes de fichiers de machine virtuelle.

En plus du tableau de bord, vous pouvez utiliser les clients de ligne de commande de `glance` et `nova` ou les API de service d'images et de calcul pour gérer les images.

## Affichage des informations sur les images

La commande `nova image-list` affiche une liste des images disponibles. Vous utiliserez le nom ou l'ID de la variante pour créer l'instance de machine virtuelle.

```
$ nova image-list
+-----+-----+-----+-----+
| ID                | Name                | Status | Server |
+-----+-----+-----+-----+
| e422aae1-b0ba-618c-85d3-a214059800e2 | Solaris Kernel Zone | ACTIVE |        |
| e82aa857-ec92-4859-f530-deb89274863e | Solaris Non-global Zone | ACTIVE |        |
+-----+-----+-----+-----+
```

La commande `glance image-list` affiche des informations supplémentaires, y compris le format de disque, le format de conteneur et la taille de l'image

```
$ glance image-list --human-readable
```

Les commandes `nova image-show` et `glance image-show` offrent plus d'informations sur une image spécifique

```
$ nova image-show 'Solaris Kernel Zone'
+-----+
| Property          | Value                               |
+-----+
| OS-EXT-IMG-SIZE:size | 1547458560                          |
| created           | 2014-06-29T15:40:49Z                |
| id                | e422aae1-b0ba-618c-85d3-a214059800e2 |
| metadata architecture | x86_64                              |
| metadata hypervisor_type | solariszones                        |
| metadata vm_mode    | solariszones                        |
| minDisk            | 0                                    |
| minRam             | 0                                    |
| name               | Solaris Kernel Zone                 |
| progress           | 100                                  |
| status             | ACTIVE                              |
| updated            | 2014-06-29T15:40:55Z                |
+-----+

$ glance image-show 'Solaris Kernel Zone'
+-----+
| Property          | Value                               |
+-----+
| Property 'architecture' | x86_64                              |
| Property 'hypervisor_type' | solariszones                        |
| Property 'vm_mode'      | solariszones                        |
| checksum               | b2fc9560c15603c7663326db82d5ddaa  |
| container_format       | bare                                 |
| created_at             | 2014-06-29T15:40:49.108578          |
| deleted                 | False                                |
| disk_format            | raw                                  |
| id                     | e422aae1-b0ba-618c-85d3-a214059800e2 |
| is_public              | True                                 |
| min_disk               | 0                                    |
| min_ram                | 0                                    |
| name                   | Solaris Kernel Zone                 |
| owner                  | 7461d4a9f5a64af9a01ae4e84e08c182  |
| protected              | False                                |
| size                   | 1547458560                          |
| status                 | active                              |
| updated_at             | 2014-06-29T15:40:55.769756         |
+-----+
```

## Création d'une image

Dans Solaris, les images OpenStack sont des archives d'ensemble et doivent exécuter Oracle Solaris 11.2. Avec la commande `archiveadm`, vous pouvez créer de nouvelles archives d'ensemble à partir de zones globales, de zones non globales et de zones de noyau exécutant Oracle Solaris 11.2. Téléchargez l'image vers le référentiel Glance en vue d'une utilisation avec OpenStack.

Votre archive d'ensemble peut être soit une archive de clone soit une archive de récupération. Créez une archive de clone basée sur l'environnement d'initialisation actif en cours, ou créez une archive de récupération qui inclut tous les environnements d'initialisation et les informations de configuration système. Un clone d'archive n'inclut aucune des informations de configuration système à partir de l'instance du SE. En ce qui concerne une archive de clone, les programmes d'installation forcent la reconfiguration, ou vous pouvez fournir la configuration dans un profil de configuration système (SC). De plus, une archive de clone ne contient pas les environnement d'initialisation inactifs, par exemple. Utilisez une archive d'ensemble de récupération si vous souhaitez le système dans son intégralité. Pour plus d'informations sur Unified Archives, reportez-vous à “ [Utilisation de Unified Archives pour la récupération du système et le clonage dans Oracle Solaris 11.2](#) ”.

Les commandes suivantes capturent une archive d'ensemble d'une zone non globale en cours d'exécution nommée myzone

```
global# zonecfg -z myzone create
global# zoneadm -z myzone install
global# zlogin myzone
'sed /^PermitRootLogin/s/no$/without-password/
< /etc/ssh/sshd_config > /system/volatile/sed.$$ ;
cp /system/volatile/sed.$$ /etc/ssh/sshd_config'
global# archiveadm create -z myzone /var/tmp/myzone.uar
```

Vous pouvez également créer une image OpenStack en créant un *cliché* d'une instance de machine virtuelle existante. Utilisez la commande `nova image-create` pour créer une image en prenant un cliché d'une instance de machine virtuelle en cours d'exécution.

En plus de la création d'une image pour créer des instances de machine virtuelle, vous pourrez souhaiter utiliser des images personnalisées pour les sauvegardes de données ou pour assurer le secours d'une instance de machine virtuelle. Une *image de secours* est un type spécial d'image qui est initialisée lorsqu'une instance de machine virtuelle est placée en mode secours. Une image de secours permet à un administrateur de monter les systèmes de fichiers pour l'instance de machine virtuelle afin de résoudre le problème.

## Ajout d'une image au magasin d'images

Le service d'images de OpenStack, Glance, fournit un service de stockage, de détection, d'inscription et de livraison d'images de disque et de serveur. Un *serveur de registre* est un service d'image fournissant les informations de métadonnées de l'image aux clients. L'*image cache* est utilisée par le service d'image pour obtenir des images sur l'hôte local au lieu de devoir effectuer un nouveau téléchargement à partir du serveur d'images chaque fois qu'une image est demandée.

La commande suivante télécharge l'archive d'ensemble créée dans la section précédente vers le référentiel Glance. Utilisez le format de type `raw`. Veillez à spécifier la propriété de l'architecture.

```
global# glance image-create --container-format bare --disk-format raw
--is-public true --name "Oracle Solaris 11.2 x86 NGZ"
--property architecture=x86_64
--property hypervisor_type=solariszones
--property vm_mode=solariszones < /var/tmp/myzone.uar
```

La commande `glance image-create` permet de télécharger l'image et de définir toutes les valeurs de propriété à la fois. Le script suivant montre comment s'assurer que vous avez téléchargé l'image avec la propriété `architecture` définie sur l'architecture de l'hôte en cours

```
#!/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
```

## Création d'une instance de machine virtuelle

Une variante, une image et un réseau sont nécessaires pour créer une instance de machine virtuelle.

### ▼ Création d'une instance de machine virtuelle à l'aide de l'interface de ligne de commande

1. Sélectionnez le locataire pour lequel vous souhaitez créer la nouvelle instance de machine virtuelle.

Vous devez spécifier le nom ou l'ID de locataire dans la commande pour créer l'instance de machine virtuelle.

```
$ keystone tenant-list
+-----+-----+-----+
|          id          | name | enabled |
+-----+-----+-----+
| 6ea34f7dafa5ce3c9a1b9de659e59d77 | demo | True |
| 0bda9b63b800ca808031a38637d50f3e | service | True |
+-----+-----+-----+
```

## 2. Sélectionnez l'image à partir de laquelle vous souhaitez créer la nouvelle instance de machine virtuelle.

Vous devrez spécifier le nom ou l'ID de l'image dans la commande pour créer l'instance de machine virtuelle. Reportez-vous à [“Affichage des informations sur les images” à la page 114](#) pour plus d'informations sur la procédure d'affichage des spécifications de l'image, y compris l'ID de l'image.

Si vous déployez une instance de machine virtuelle de zone non globale d'une version inférieure à celle de la zone globale, l'instance de machine virtuelle sera automatiquement mise à niveau vers le niveau de version de la zone globale lors de l'installation. Une tentative de déploiement d'une instance de machine virtuelle de zone non globale d'une version supérieure à celle de la zone globale échouera.

## 3. Sélectionnez la variante à partir de laquelle vous souhaitez créer la nouvelle instance de machine virtuelle.

Assurez-vous de disposer d'une variante avec les spécifications dont vous avez besoin. Ajoutez les spécifications requises à une variante ou créez une variante avec les spécifications dont vous avez besoin. Reportez-vous à [“Affichage des informations sur les variantes” à la page 112](#) pour plus d'informations sur la procédure d'affichage des spécifications de chaque variante, y compris l'ID de variante et les extra-specs.

## 4. Sélectionnez un réseau à utiliser par la nouvelle instance de machine virtuelle.

Vous devrez indiquer le nom ou l'ID du réseau dans la commande pour créer l'instance de machine virtuelle. Si la commande `neutron net-list` n'affiche pas un réseau pour le locataire sélectionné lors de l'étape 1, utilisez la commande `neutron net-create` pour créer un réseau pour ce locataire. Reportez-vous [OpenStack Command-Line Interface Reference](#) pour plus de détails. Notez l'ID du réseau.

## 5. Créez l'instance.

Utilisez la commande `nova boot` pour créer et initialiser une instance de calcul. L'*imageID* vient de l'étape 2, la *flavorID* de l'étape 3, et la *nicID* de l'étape 4. Reportez - vous [OpenStack Command-Line Interface Reference](#) pour plus de détails.

```
# nova boot --image imageID --flavor flavorID --nic net-id=nicID
```

**6. Sélectionnez une adresse IP flottante à utiliser par la nouvelle instance de machine virtuelle.**

Utilisez la commande `neutron floatingip-list` pour afficher les adresses IP flottantes pour le locataire sélectionné à l'étape 1. Si nécessaire, utilisez la commande `neutron floatingip-create` pour créer une adresse IP flottante pour ce locataire. Notez l'ID de cette adresse IP flottante.

**7. Associez l'adresse IP flottante à la nouvelle instance de machine virtuelle.**

Utilisez la commande `neutron floatingip-associate` pour associer l'adresse IP flottante de l'étape 6 à la nouvelle instance VM.



## Dépannage d'OpenStack

---

Ce chapitre présente les sections suivantes :

- Problèmes connus dans cette version
- Utilisation des fichiers journaux associés à OpenStack
- Examen et résolution des problèmes

### Limitations connues

Problèmes connus avec OpenStack (Havana 2013.2.3) dans Oracle Solaris 11.2:

- L'accès distant par la console à des instances de machine virtuelle à l'aide du tableau de bord OpenStack n'est pas pris en charge. Utilisez le tableau de bord pour télécharger une paire de clés SSH. Cette paire de clés sera ajoutée au fichier `authorized_keys` de l'utilisateur `root` de cette instance de machine virtuelle.
- Seuls les noeuds Nova exécutant Solaris sont entièrement pris en charge, car Neutron ne prend en charge qu'un seul module d'extension pour la virtualisation du réseau.
- L'ajout de volumes Cinder en pièce jointe n'est actuellement pas pris en charge dans les zones non globales.
- Les instances de machine virtuelle doivent exécuter Oracle Solaris 11.2.
- Le redimensionnement des instances de machine virtuelle n'est pas pris en charge.

La commande `nova resize` n'est pas prise en charge. La sortie de la commande `nova resize` pourra indiquer que la commande a été exécutée, mais la commande `nova resize-confirm` signale que le redimensionnement de l'instance ne peut pas être confirmé, et la commande `nova show` indique que l'instance n'a pas été redimensionnée.

- La migration en direct des instances de machine virtuelle n'est pas prise en charge.

La commande `nova live-migration` n'est pas prise en charge.

- La sauvegarde de Cinder n'est pas prise en charge.

Le service `cinder-backup` est installé en même temps que le package `cinder`, mais le service est désactivé, et doit le rester.

- Sur la boîte de dialogue du lancement de l'instance, seule l'initialisation à partir de l'image est prise en charge pour la source d'initialisation de l'instance. Sur le menu Projet -> Clichés

d'images & -> Menu d'actions, CreateVolumeFromImage n'est pas pris en charge. Reportez-vous à la section [Solaris OpenStack Horizon customizations](#) pour plus d'informations sur d'autres personnalisations du tableau de bord Solaris OpenStack.

- Les liaisons de données VXLAN ne sont pas prises en charge en tant que valeur pour l'option `external_network_dataLink` dans le fichier `/etc/neutron/l3_agent.ini`. Si vous définissez une liaison de données VXLAN en tant que valeur de l'option `external_network_dataLink`, l'agent Neutron L3 ne parviendra pas à créer et à raccorder une VNIC sur le réseau externe.
- Vous devez utiliser la ligne de commande pour modifier le quota d'une ressource réseau pour un projet.

Le quota d'une ressource réseau ne peut pas être modifié dans Horizon. Vous pouvez utiliser le tableau de bord Horizon pour créer un projet ou modifier des ressources non-réseau d'un projet existant. Pour modifier le quota des réseaux, des sous-réseaux, des ports, des routeurs ou des adresses IP flottantes, vous devez utiliser la commande `neutron quota-update`.

Même lorsque vous modifiez une ressource non-réseau, vous pouvez voir le message d'erreur suivant. Vous pouvez ignorer ce message. Contrairement à ce qu'indique le message, le quota de la ressource non-réseau a été appliqué.

Error: Modified project information and members, but unable to modify project quotas.

- SMF et Openstack peuvent renvoyer des états de service différents.

L'exemple suivant montre que le service `nova-cert` est désactivé dans OpenStack, même si SMF présente ce service comme étant `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
```

- Le service SMF `neutron-l3-agent` se met en maintenance lors de son redémarrage. Solution de contournement : Redémarrez le service `ipfilter` et effacez le `neutron-l3-agent`.

```
network# svcadm restart ipfilter:default
network# svcadm clear neutron-l3-agent:default
```

- La passerelle par défaut du noeud de réseau est supprimée pour certaines configurations. Si l'adresse IP du noeud de réseau est calculée à partir de l'espace d'adressage `external_network`, et que vous utilisez la commande `neutron router-gateway-clear` pour supprimer le `external_network` du `provider_router`, la passerelle par défaut du noeud de réseau sera supprimée et le noeud de réseau ne sera plus accessible.

```
network# neutron router-gateway-clear router_UUID
```

Solution de contournement : Connectez-vous au noeud de réseau par la console et renouvelez l'ajout de la passerelle par défaut.

- La base de données `sqlite` de Nova se bloque lorsque plusieurs instances sont créées simultanément.

Lorsqu'un grand nombre d'instances (par exemple, 10 ou plus), sont créées simultanément, la commande `nova list` cesse de fonctionner pendant un certain temps avec le message d'erreur suivant

```
$ nova list
```

```
ERROR: The server has either erred or is incapable of performing the
requested operation. (HTTP 500) (Request-ID:
req-0ad63452-6753-c9fc-8275-e80604d42569)
```

Il est également impossible à Horizon de trouver les instances.

Le problème est causé par le blocage de la base de données `sqlite`. Après un certain temps, la base de données se remet à fonctionner, et `nova list` comme Horizon fonctionnent normalement.

Solution de contournement : Utiliser la base de données MySQL au lieu de `sqlite`. Reportez-vous à [“Configuration du noeud de calcul” à la page 61](#).

## Examen des fichiers journaux

Les services SMF ainsi que divers processus Solaris produisent des fichiers journaux que vous pouvez consulter pour y trouver les messages d'erreur ou encore collecter plus d'informations sur les messages affichée à l'écran. Les fichiers journaux du service SMF contiennent d'utiles informations de débogage.

Selon le problème, les fichiers journaux des services SMF `nova-compute`, `nova-scheduler`, et `neutron-server` sont souvent utiles. Utilisez la commande `svcs -L` pour rechercher le nom d'un fichier journal de service SMF.

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

En tant qu'utilisateur privilégié, vous pouvez utiliser la commande `svcs -Lv` pour visualiser le fichier journal du service.

```
# svcs -Lv neutron-server
```

La commande `svcs -xv` indique l'état du service et le nom du fichier journal.

```
$ svcs -xv neutron-server
svc:/application/openstack/neutron/neutron-server:default (OpenStack Neutron Network Service)
```

```
State: online since Fri Jul 25 12:11:16 2014
  See: /var/svc/log/application-openstack-neutron-neutron-server:default.log
Impact: None.
```

Outre les fichiers journaux de service SMF, les services OpenStack produisent également des fichiers journaux, et de nombreux processus Solaris ont leurs propres fichiers journaux. Certains services OpenStack journalisent les informations sous leur nom de service OpenStack dans le répertoire `/var/log`. Par exemple, les fichiers journaux du magasin d'images OpenStack sont dans `/var/log/glance`. Si vous rencontrez des problèmes lors de la création et l'initialisation d'une instance de machine virtuelle, consultez le répertoire `/var/log/zones`.

La plupart des fichiers de configuration sont sous le nom du service OpenStack dans le répertoire `/etc`. Ainsi, les fichiers de configuration de mise en réseau OpenStack sont dans `/etc/neutron`. Les fichiers de configuration de Horizon sont dans `/etc/openstack_dashboard`.

Pour recevoir plus d'informations dans le fichier journal d'un service OpenStack, définissez l'option `verbose` dans le fichier de configuration de ce service. L'option `verbose` du fichier de configuration peut déjà être définie sur `false` ou être commentée. Annulez le commentaire ou ajoutez l'option `verbose` et définissez `verbose = true`. De la même façon, vous pouvez définir `debug=true` dans le fichier de configuration pour afficher un plus grand nombre de sorties des diverses opérations affectées par ce fichier de configuration. Reportez-vous aux tableaux d'options de configuration dans "Common Configuration Parameters for OpenStack" dans la rubrique [Getting Started with OpenStack on Oracle Solaris 11.2](#) et dans la *Référence de configuration d'OpenStack* sur le site de documentation [OpenStack](#).

Comme pour la définition de `debug=true` dans un fichier de configuration, les commandes de service OpenStack peuvent prendre une option `--debug`.

Prenez le profile RBAC approprié afin de visualiser les fichiers journaux de service OpenStack ou d'utiliser la commande `pfedit` pour modifier les fichiers de configuration du service OpenStack. Les profils pouvant être affectés sont les suivants

- Gestion OpenStack en mode "block storage"
- Gestion de calcul OpenStack
- Gestion des identités OpenStack
- Gestion des images OpenStack
- Gestion du réseau OpenStack
- Gestion de stockage d'objets OpenStack
- Gestion OpenStack

## Examen et résolution des problèmes

La commande `svcs -x` sans nom de service affiche tous les services en statut maintenance. Utilisez la commande `svcs` sur tous les noeuds OpenStack pour vous assurer que tous

les services requis sont en statut online. Si un service requis n'est pas en statut online, consultez les fichiers journaux, comme indiqué dans la section [“Examen des fichiers journaux”](#) à la page 123.

Pour une aide générale concernant la résolution des problèmes pour les services SMF, reportez-vous à la section [“Dépistage des pannes de services”](#) du manuel [“Gestion des services système dans Oracle Solaris 11.2”](#).

Si votre problème est probablement un problème de réseau, exécutez la commande `evsadm` sur tous les noeuds.

Si vous rencontrez des difficultés à effectuer une opération avec le tableau de bord OpenStack, essayez d'effectuer la même opération avec la ligne de commande. Le message d'erreur que vous recevrez sur la ligne de commande pourra s'avérer plus utile. Vérifiez que la commande possède une option de sortie détaillée.

Si vous rencontrez un problème dans un autre noeud que le noeud de contrôleur, exécutez de simples commandes, telles que `nova list` sur la ligne de commande de ce noeud pour vérifier que le noeud peut communiquer avec le noeud de contrôleur.

## Installation et configuration d'OpenStack

Cette section présente les erreurs que vous risquez de rencontrer lorsque vous installez et configurez OpenStack.

### Erreurs du tableau de bord

Si vous voyez des messages d'erreur "Non autorisé" tels que les messages suivants, vérifiez si la clé d'hôte RSA a été modifiée.

```
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.
```

## Installation et configuration de l'instance de machine virtuelle

Les problèmes abordés dans cette section sont spécifiques aux instances de machine virtuelle.

## Instance de machine virtuelle en statut d'erreur

Une instance de machine virtuelle pourra être en statut d'erreur car vous tentez de l'installer sur un système hôte qui a une architecture différente. Dans ce cas, vous risquez de ne pas recevoir un message d'erreur vous indiquant précisément que les architectures ne correspondent pas. Pour éviter ce problème, assurez-vous d'avoir correctement défini la propriété `architecture` de l'image lorsque vous téléchargez celle-ci depuis le magasin d'images  `glance`. Si vous utilisez Horizon pour télécharger une image, vous devez définir les propriétés de l'image après les téléchargements. Vous pouvez également utiliser la ligne de commande pour télécharger l'image et définir les valeurs de propriété en une seule commande  `glance image-create`. Reportez-vous à [“Ajout d'une image au magasin d'images”](#) à la page 116 pour obtenir un exemple.

## Les valeurs de propriété d'instance de machine virtuelle ne correspondent pas aux valeurs de propriété de la zone

Certaines informations qu'OpenStack retourne sur une instance de machine virtuelle ne correspondent pas aux informations retournées par Solaris sur la zone correspondante. Les informations affichées dans Horizon ou par la commande `nova` peuvent ne pas correspondre aux informations affichées par la commande `zoneadm` ou par d'autres commandes Solaris.

Nom	Le nom d'une instance de machine virtuelle tel qu'affiché dans Horizon ou par la commande <code>nova list</code> est le nom que vous avez affecté lors de la création de l'instance, tel que <code>example-instance</code> . Le nom de la zone affiché par la commande <code>zoneadm list</code> est identique à <code>instance-00000001</code> . Utilisez la commande <code>nova show</code> pour déterminer quelles zones sont associées à quelles instances de machine virtuelle. Dans la sortie <code>nova show</code> , la valeur de la propriété <code>OS-EXT-SRV-ATTR:instance_name</code> est le nom de la zone, et la valeur de la propriété <code>name</code> est le nom de l'instance de machine virtuelle.
UUID	L'UUID d'une instance de machine virtuelle telle qu'affichée dans Horizon ou par la commande <code>nova show</code> ne correspond pas à l'UUID de la même zone telle qu'affichée par la commande <code>zoneadm list -p</code> . L'UUID affichée par la commande <code>zoneadm</code> est un identificateur différent de celui utilisé pour Nova.
CPU	Le nombre de CPU virtuelles d'une instance de machine virtuelle tel qu'affiché dans Horizon est le nombre de CPU <i>plafonnées</i> qui sont virtualisées uniquement dans la mesure du nombre de fractions de CPU pouvant être utilisées par l'instance. Ce chiffre ne fournit pas d'observabilité à l'intérieur de l'instance plafonnée. La commande <code>psrinfo</code> signale les CPU dédiées qui sont allouées à la zone.

Mémoire	La quantité de mémoire d'une instance de machine virtuelle telle qu'affichée dans Horizon peut être différente de la quantité de mémoire que la commande <code>prtconf</code> affiche lorsque vous êtes connecté à cette instance de machine virtuelle. Horizon indique la quantité de mémoire spécifiée par la variante utilisée pour créer l'instance de machine virtuelle. La commande <code>prtconf</code> signale l'ensemble de la mémoire système.
Stockage	La quantité de stockage d'une instance de machine virtuelle telle qu'affichée dans Horizon peut être différente de la quantité de stockage affichée lorsque vous êtes connecté à cette instance de machine virtuelle, sauf si l'instance de machine virtuelle est une zone non globale utilisant ZOSS (Zones on Shared Storage).

## Démontage de réseaux

Si vous rencontrez des problèmes lors de la configuration de Neutron sur le noeud Réseau et devez peut-être annuler la configuration pour recommencer, suivez la procédure ci-dessous. En fonction du point auquel vous devez commencer à annuler la configuration, suivez la séquence décrite dans la procédure.

### ▼ Suppression de la configuration réseau sur Neutron

#### 1. Effectuez cette étape sur le tableau de bord Horizon.

- a. Dissociez toutes les adresses IP flottantes.
- b. Supprimez toutes les adresses IP flottantes.

#### 2. Dans une fenêtre du terminal, tapez les commandes suivantes

```
# neutron router-gateway-clear router-id external-network-id
```

```
# neutron router-interface-delete router-id subnet-id
```

- a. Pour supprimer l'interface de passerelle du routeur, tapez la commande suivante

```
# neutron router-gateway-interface-delete router-id external-network-id
```

- b. Pour supprimer les interfaces de routeur restantes, tapez la commande suivante

```
# neutron router-interface-delete router-id subnet-id
```

**3. Effectuez les tâches suivantes sur le tableau de bord Horizon.**

**a. Mettez fin à toutes les instances VM.**

**b. Supprimez les sous-réseaux.**

Si des problèmes se produisent lors de la suppression des sous-réseaux, reportez-vous à la section [“Suppression des Vports” à la page 128.](#)

**c. Supprimez le réseau.**

**▼ Suppression des Vports**

Utilisez cette procédure si des problèmes vous empêchent de supprimer des sous-réseaux.

**1. Déterminez les vports qui sont en cours d'utilisation.**

```
# evsadm
```

**2. Réinitialisez les vports en cours d'utilisation.**

```
# evsadm reset-vport vport
```

**3. Supprimez les vports.**

```
# evsadm delete-vport vport
```

# Index

---

## A

Adresses IP flottantes, 74, 108  
Advanced Message Queuing Protocol (AMQP) *Voir* AMQP  
Agent DHCP, 66  
Agent L3, 66, 103  
AI, 13, 20

- média amorçable, 23
- Service d'installation, 22

AMQP, 45, 83  
AMQP (Advanced Message Queuing Protocol), 45  
Archive d'ensemble OpenStack *Voir* Unified Archives  
archiveadm, commande, 23, 115  
Authentification *Voir* Keystone

## B

Base de données, 48, 83

## C

Cinder, 13

- Installation, 51, 91
- Sauvegarde, 121
- ZFS Storage Appliance, 53, 92

Clichés, 116  
Commande `zlogin`, 27  
Commutateur virtuel élastique *Voir* EVS  
Configuration du système requise pour l'installation, 16  
Connexion à la console de la zone, 25, 27

## E

Environnements d'initialisation (BE), 14  
EVS, 12

- Voir aussi* Neutron

commande `evsadm`, 104  
Configuration, 26  
configuration sur le noeud de réseau, 65  
`evsadm`, commande, 65, 71  
Script de configuration `configure_evs.py`, 26  
`evsadm`, commande, 65

## G

Glance, 13, 116

- Installation, 56, 86

## H

Heat, 14

- installation, 50, 97

Horizon, 13

- Configuration d'accès SSL, 59, 90
- Erreurs, 123

## I

Image de secours, 116  
Images, 30, 114

- Voir aussi* Instances de machine virtuelle
- Clichés, 116
- Image cache, 116
- Sauvegarde, 116
- Secours, 116
- Serveur de registre, 116

Installation

- Exigences, 16

Installation d'OpenStack

- Configuration multinoeud, 77

Installation de Juno OpenStack

- Avec une archive d'ensemble, 26
- Installation OpenStack
  - Avec une archive d'ensemble, 19
  - Configuration d'évaluation, 19
  - Configuration monoeud, 19
  - Configuration multinoeud, 41
- Installation OpenStack monoeud
  - Avec OVM Server for SPARC, 41, 78
  - Avec une archive d'ensemble, 19
- Installation OpenStack multi-noeud, 41
- Installation OpenStack multinoeud, 77
- Instance *Voir* Instances de machine virtuelle
- Instances de machine virtuelle, 12, 13, 15
  - Voir aussi* Nova
  - Clichés, 116
  - Connexion, 74, 108
  - Création, 33, 111, 117
  - Images, 30
  - Initialisation, 33, 117
  - Sauvegarde, 116
  - Secours, 116
  - Variantes, 111
- Instances de machine virtuelles
  - connexion, 38
- Instances de machine virtuelle
  - Images, 114
- Instances de machines virtuelles
  - Variantes, 29, 32
- IPS (Image Packaging System), 14
  - Voir aussi* Packages

## K

- Keystone, 14
  - Installation, 49, 85

## L

- LDoms, 41, 78
- Locataires, 31

## M

- Machine virtuelle (VM), 13 *Voir* Instances de machine virtuelle

- Machine virtuelle du Cloud *Voir* Instances de machine virtuelle
- Magasin d'images *Voir* Glance
- Modèles
  - Matériel *Voir* Variantes
- Modèles d'instances *Voir* Variantes
- Modèles matériels *Voir* Variantes
- MySQL, 48, 83

## N

- Network Time Protocol (NTP), 47, 82
- Neutron, 12
  - Adresses IP flottantes, 74, 108
  - Agent DHCP, 66
  - Agent L3, 66, 103
  - Installation, 57, 94
- Noeud de calcul, 41, 77
  - Configuration, 60, 98
- Noeud de contrôleur, 41, 77
  - Configuration, 45, 81
- Noeud de réseau, 41
  - Configuration, 62
- Noeud de stockage, 77
  - Configuration, 100
- Nova, 12
  - Erreurs, 123
  - Installation sur le noeud de contrôleur, 58, 88
- NTP, 47, 82

## O

- Options de débogage, 124
- Outil SCI, 22
- Outil System Configuration Interactive (SCI) *Voir*
- Outil SCI Tool
- OVM Server for SPARC, 41, 78

## P

- Profil de configuration système *Voir* profil SC
- Profil SC, 114, 116
- Profils RBAC, 14
- Programme d'installation automatisée *Voir* AI

Projets, 31

## R

RabbitMQ, 45, 83  
 RAD, 61, 98  
 RAD (Démon d'accès distant), 61, 98  
 Réseau externe  
   Création, 103, 104  
 Réseau interne  
   Création, 102

## S

sample\_data.sh script, 85  
 Script de configuration eVSconfigure\_evs.py, 26  
 SDN, 12  
 SDN (Software Defined Networking) *Voir* Neutron  
 Serveur de registre, 116  
 Service d'identité *Voir* Keystone  
 Service d'image *Voir* Glance  
 service d'orchestration *Voir* Heat  
 Service de calcul *Voir* Nova  
 Service SMF neutron-dhcp-agent , 66  
 Service SMF neutron-l3-agent , 71, 105  
 Services OpenStack, 12  
 SMF (Service Management Facility), 14  
 SPARC, 41, 78  
 SQLite, 48, 83  
 Stockage, 13, 13  
   *Voir aussi* Cinder  
   *Voir aussi* Swift  
 Stockage d'objets *Voir* Swift  
 Stockage en mode bloc *Voir* Cinder  
 Structure de zones, 61, 98  
 Swift, 13, 45, 79

## T

Tableau de bord, 13  
   *Voir aussi* Horizon  
 Affichage des images, 30  
 Affichage des projets, 31  
 Affichage des variantes, 32  
 Connexion, 28

Création d'instances de machine virtuelle, 33  
 Initialisation d'une instance de machine virtuelle, 33

## U

Unified Archives, 13  
   *Voir aussi* Glance  
 Création d'images OpenStack à partir de , 115  
 Installation OpenStack mononoeud, 19  
   Service AI à chaud, 22  
   USB à chaud, 21  
   Zone de noyau, 24  
 installation OpenStack mononoeud  
   AI amorçable à chaud, 23  
 Installation OpenStack mononoeud (Juno), 26, 27  
 Téléchargement, 20  
 usbcopy, commande, 21

## V

Variantes, 32, 111  
   *Voir aussi* Instances de machine virtuelle  
   extra-specs, 112  
 Virtualisation du réseau *Voir* Neutron

## Z

ZFS, 13, 13  
   *Voir aussi* Cinder  
   *Voir aussi* Swift  
 ZFS Storage Appliance (ZFSSA), 53, 92  
 ZFSSA, 53, 92  
 zlogin, commande, 25  
 zoneadm, commande  
   boot, sous-commande, 25, 27  
   install, sous-commande, 20  
 zonecfg, commande  
   add, sous-commande, 25, 27  
   create, sous-commande, 25, 27  
   select, sous-commande, 25, 27  
   set, sous-commande, 25, 27  
 Zones, 12  
 Zones de noyau, 12  
 Zones non globales, 12

