



Red Hat Enterprise Linux 9

Configuration d'un cluster Red Hat High Availability sur Red Hat OpenStack Platform

Installation et configuration de clusters HA et de ressources de clusters sur des instances RHOSP

Red Hat Enterprise Linux 9 Configuration d'un cluster Red Hat High Availability sur Red Hat OpenStack Platform

Installation et configuration de clusters HA et de ressources de clusters sur des instances RHOSP

Notice légale

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

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

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Résumé

Vous pouvez utiliser Red Hat High Availability Add-On pour configurer un cluster de haute disponibilité (HA) sur les instances de Red Hat OpenStack Platform (RHOSP). Ce titre fournit des instructions pour l'installation des paquets et des agents requis, ainsi que des exemples qui configurent un cluster de base, des ressources de clôture et des ressources de cluster HA.

Table des matières

RENDRE L'OPEN SOURCE PLUS INCLUSIF	3
FOURNIR UN RETOUR D'INFORMATION SUR LA DOCUMENTATION DE RED HAT	4
CHAPITRE 1. PRÉFACE	5
CHAPITRE 2. CONFIGURATION RECOMMANDÉE DU GROUPE DE SERVEURS RHOSP	6
CHAPITRE 3. INSTALLATION DES PAQUETS ET AGENTS HAUTE DISPONIBILITÉ ET RHOSP	7
CHAPITRE 4. MÉTHODES D'AUTHENTIFICATION POUR RHOSP	9
4.1. AUTHENTIFICATION AVEC UN FICHER CLOUDS.YAML	9
4.2. AUTHENTIFICATION AVEC UN SCRIPT D'ENVIRONNEMENT OPENRC	10
4.3. AUTHENTIFICATION AVEC UN NOM D'UTILISATEUR ET UN MOT DE PASSE	10
CHAPITRE 5. CRÉER UN CLUSTER DE BASE SUR RED HAT OPENSTACK PLATFORM	11
CHAPITRE 6. CONFIGURATION DE LA CLÔTURE POUR UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM	13
CHAPITRE 7. CONFIGURATION DES RESSOURCES D'UN CLUSTER HA SUR LA PLATEFORME RED HAT OPENSTACK	15
7.1. CONFIGURATION D'UNE RESSOURCE OPENSTACK-INFO (OBLIGATOIRE)	15
7.2. CONFIGURER UNE ADRESSE IP VIRTUELLE DANS UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM	16
7.3. CONFIGURER UNE ADRESSE IP FLOTTANTE DANS UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM	17
7.4. CONFIGURATION D'UNE RESSOURCE DE STOCKAGE PAR BLOCS DANS UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM	19

RENDRE L'OPEN SOURCE PLUS INCLUSIF

Red Hat s'engage à remplacer les termes problématiques dans son code, sa documentation et ses propriétés Web. Nous commençons par ces quatre termes : master, slave, blacklist et whitelist. En raison de l'ampleur de cette entreprise, ces changements seront mis en œuvre progressivement au cours de plusieurs versions à venir. Pour plus de détails, voir le [message de notre directeur technique Chris Wright](#).

FOURNIR UN RETOUR D'INFORMATION SUR LA DOCUMENTATION DE RED HAT

Nous apprécions vos commentaires sur notre documentation. Faites-nous savoir comment nous pouvons l'améliorer.

Soumettre des commentaires sur des passages spécifiques

1. Consultez la documentation au format **Multi-page HTML** et assurez-vous que le bouton **Feedback** apparaît dans le coin supérieur droit après le chargement complet de la page.
2. Utilisez votre curseur pour mettre en évidence la partie du texte que vous souhaitez commenter.
3. Cliquez sur le bouton **Add Feedback** qui apparaît près du texte en surbrillance.
4. Ajoutez vos commentaires et cliquez sur **Submit**.

Soumettre des commentaires via Bugzilla (compte requis)

1. Connectez-vous au site Web de [Bugzilla](#).
2. Sélectionnez la version correcte dans le menu **Version**.
3. Saisissez un titre descriptif dans le champ **Summary**.
4. Saisissez votre suggestion d'amélioration dans le champ **Description**. Incluez des liens vers les parties pertinentes de la documentation.
5. Cliquez sur **Submit Bug**.

CHAPITRE 1. PRÉFACE

Ce document fournit des procédures pour configurer un cluster Red Hat High Availability (HA) et ses agents de clôture et de ressources sur les instances Red Hat OpenStack Platform (RHOSP).

Pour la documentation générale de RHOSP, voir [Documentation produit pour Red Hat Openstack Platform](#).

Pour les politiques, exigences et limitations de Red Hat applicables à l'utilisation d'instances RHOSP en tant que membres d'un cluster RHEL High Availability, voir Support [Policies for RHEL High Availability Clusters - OpenStack Virtual Machines as Cluster Members - Red Hat Customer Portal](#) (Politiques de support pour les clusters RHEL High Availability - Machines virtuelles OpenStack en tant que membres de clusters).

CHAPITRE 2. CONFIGURATION RECOMMANDÉE DU GROUPE DE SERVEURS RHOSP

Avant de créer les instances RHOSP à utiliser comme nœuds de cluster HA, créez un groupe de serveurs d'instance pour regrouper les instances par une politique d'affinité, avec un groupe de serveurs par cluster lorsque vous configurez plusieurs clusters.

Lors de la création d'un groupe de serveurs, la stratégie d'affinité que vous définissez pour le groupe de serveurs détermine la résilience du cluster en cas de défaillance de l'hyperviseur.

- Avec la politique d'affinité par défaut de **affinity**, tous les nœuds du cluster pourraient être créés sur le même hyperviseur RHOSP. Dans ce cas, si l'hyperviseur tombe en panne, tout le cluster tombe en panne. Pour cette raison, Red Hat vous recommande de définir une stratégie d'affinité pour le groupe de serveurs de **anti-affinity** ou **soft-anti-affinity**.
- Avec une politique d'affinité de **anti-affinity**, le groupe de serveurs n'autorise qu'un seul nœud de cluster par nœud de calcul. Toute tentative de création d'un nombre de nœuds de cluster supérieur à celui des nœuds de calcul génère une erreur. Bien que cette configuration offre la meilleure protection contre les défaillances de l'hyperviseur RHOSP, elle peut nécessiter plus de ressources que celles dont vous disposez pour déployer des clusters de grande taille.
- Avec une stratégie d'affinité de **soft-anti-affinity**, le groupe de serveurs distribuera les nœuds de cluster aussi uniformément que possible sur tous les nœuds de calcul. Bien que cela offre moins de protection contre les défaillances de l'hyperviseur qu'une stratégie de **anti-affinity**, cela offre un niveau plus élevé de haute disponibilité qu'une stratégie d'affinité de **affinity**.

Pour déterminer la politique d'affinité des groupes de serveurs pour votre déploiement, vous devez équilibrer les besoins de votre cluster avec les ressources dont vous disposez en prenant en compte les composants suivants du cluster :

- Le nombre de nœuds dans la grappe
- Le nombre de nœuds de calcul RHOSP disponibles
- Nombre de nœuds requis pour le quorum de la grappe afin de maintenir les opérations de la grappe

Pour plus d'informations sur l'affinité et la création d'un groupe de serveurs d'instance, les [filtres du planificateur de calcul](#) et la [référence de l'interface de ligne de commande](#) .

CHAPITRE 3. INSTALLATION DES PAQUETS ET AGENTS HAUTE DISPONIBILITÉ ET RHOSP

Sur chacun des nœuds que vous utiliserez pour un cluster HA sur la plateforme OpenStack, utilisez la procédure suivante pour installer les paquets nécessaires à la configuration d'un cluster Red Hat High Availability sur Red Hat OpenStack Platform (RHOSP).

Conditions préalables

- Un groupe de serveurs pour les instances RHOSP à utiliser comme nœuds de cluster HA, comme décrit dans [Configuration recommandée des groupes de serveurs RHOSP](#)
- Une instance RHOSP pour chaque nœud de cluster HA
 - Les instances sont membres d'un groupe de serveurs
 - Les instances sont configurées comme des nœuds exécutant RHEL 9.1 ou une version ultérieure

Procédure

1. Activez les dépôts RHEL HA et le canal d'outils RHOSP.

```
# subscription-manager repos --enable=rhel-9-for-x86_64-highavailability-rpms
# subscription-manager repos --enable=openstack-17-tools-for-rhel-9-x86_64-rpms
```

2. Installez les paquetages logiciels de Red Hat High Availability Add-On, ainsi que les paquetages requis pour les agents de ressources de cluster RHOSP et les agents de clôture RHOSP.

```
# dnf install pcs pacemaker python3-openstackclient python3-novaclient fence-agents-openstack
```

3. L'installation des paquets **pcs** et **pacemaker** sur chaque nœud crée l'utilisateur **hacluster**, qui est le compte d'administration **pcs**. Créez un mot de passe pour l'utilisateur **hacluster** sur tous les nœuds du cluster. Il est recommandé d'utiliser le même mot de passe pour tous les nœuds.

```
# passwd hacluster
```

4. Si **firewalld.service** est installé, ajoutez le service de haute disponibilité au pare-feu RHEL.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --add-service=high-availability
```

5. Démarrer le service **pcs** et l'autoriser à démarrer au démarrage.

```
# systemctl start pcsd.service
# systemctl enable pcsd.service
```

6. Modifiez le fichier **/etc/hosts** et ajoutez les noms d'hôtes RHEL et les adresses IP internes. Voir [Comment le fichier /etc/hosts doit-il être configuré sur les nœuds de cluster RHEL ?](#) pour plus d'informations.

7. Vérifiez que le service **pcs** est en cours d'exécution.

systemctl status pcsd.service

pcsd.service - PCS GUI and remote configuration interface

Loaded: loaded (/usr/lib/systemd/system/pcsd.service; enabled; vendor preset: disabled)

Active: active (running) since Thu 2018-03-01 14:53:28 UTC; 28min ago

Docs: man:pcsd(8)

man:pcs(8)

Main PID: 5437 (pcsd)

CGroup: /system.slice/pcsd.service

└─5437 /usr/bin/ruby /usr/lib/pcsd/pcsd > /dev/null &

Mar 01 14:53:27 ip-10-0-0-48.ec2.internal systemd[1]: Starting PCS GUI and remote configuration interface...

Mar 01 14:53:28 ip-10-0-0-48.ec2.internal systemd[1]: Started PCS GUI and remote configuration interface.

Ressources supplémentaires

- Pour de plus amples informations sur la configuration et la gestion des clusters de haute disponibilité Red Hat, voir [Configuration et gestion des clusters de haute disponibilité](#) .

CHAPITRE 4. MÉTHODES D'AUTHENTIFICATION POUR RHOSP

Les agents de clôture à haute disponibilité et les agents de ressources prennent en charge trois méthodes d'authentification distinctes pour communiquer avec RHOSP :

- Un fichier de configuration **clouds.yaml**
- Un script d'environnement OpenRC
- Authentification par nom d'utilisateur et mot de passe via Pacemaker

Après avoir déterminé la méthode d'authentification à utiliser pour le cluster, spécifiez les paramètres d'authentification appropriés lors de la création d'une clôture ou d'une ressource de cluster.

4.1. AUTHENTIFICATION AVEC UN FICHER CLOUDS.YAML

Pour utiliser un fichier **clouds.yaml** afin de s'authentifier avec RHOSP, procédez comme suit.

Procédure

1. Sur chaque nœud qui fera partie de votre cluster, configurez un fichier **clouds.yaml**. Pour plus d'informations sur la création d'un fichier **clouds.yaml**, voir le [Guide de gestion des utilisateurs et des identités](#).

Le fichier **clouds.yaml** pour les procédures de ce document qui utilisent un fichier **clouds.yaml** pour l'authentification est le suivant. Ces procédures spécifient **ha-example** pour le fichier **cloud= parameter**, tel qu'il est défini dans ce fichier.

```
$ cat .config/openstack/clouds.yaml
clouds:
  ha-example:
    auth:
      auth_url: https://<ip_address>:13000/
      project_name: rainbow
      username: unicorns
      password: <password>
      user_domain_name: Default
      project_domain_name: Default
    <. . . additional options . . .>
    region_name: regionOne
    verify: False
```

2. Testez si l'authentification est réussie et si vous avez accès à l'API RHOSP avec la commande RHOSP de base suivante, en remplaçant le nom du nuage que vous avez spécifié dans le fichier **clouds.yaml** que vous avez créé pour **ha-example**. Si cette commande ne permet pas d'afficher une liste de serveurs, contactez votre administrateur RHOSP.

```
$ openstack --os-cloud=ha-example server list
```

3. Spécifiez le paramètre de nuage lors de la création d'une ressource de cluster ou d'une ressource de clôture.

4.2. AUTHENTIFICATION AVEC UN SCRIPT D'ENVIRONNEMENT OPENRC

Pour utiliser un script d'environnement OpenRC afin de s'authentifier avec RHOSP, procédez comme suit.

Procédure

1. Sur chaque nœud qui fera partie de votre cluster, configurez un script d'environnement OpenRC. Pour plus d'informations sur la création d'un script d'environnement OpenRC, voir [Définir des variables d'environnement à l'aide du fichier OpenStack RC](#) .
2. Testez si l'authentification est réussie et si vous avez accès à l'API RHOSP à l'aide de la commande RHOSP de base suivante. Si cette commande ne permet pas d'afficher une liste de serveurs, contactez votre administrateur RHOSP.

```
$ openstack server list
```

3. Spécifiez le paramètre **openrc** lors de la création d'une ressource cluster ou d'une ressource fencing.

4.3. AUTHENTIFICATION AVEC UN NOM D'UTILISATEUR ET UN MOT DE PASSE

Pour s'authentifier auprès de RHOSP au moyen d'un nom d'utilisateur et d'un mot de passe, spécifiez les paramètres **username**, **password**, et **auth_url** pour une ressource de cluster ou une ressource de clôture lorsque vous créez la ressource. D'autres paramètres d'authentification peuvent être nécessaires, en fonction de la configuration de RHOSP. L'administrateur RHOSP fournit les paramètres d'authentification à utiliser.

CHAPITRE 5. CRÉER UN CLUSTER DE BASE SUR RED HAT OPENSTACK PLATFORM

Cette procédure permet de créer un cluster de haute disponibilité sur une plateforme RHOSP.

Conditions préalables

- Une instance RHOSP pour chaque nœud de cluster HA
- Le nœud HAcluster fonctionne sous RHEL 9.1 ou une version ultérieure
- Les paquets haute disponibilité et RHOSP sont installés sur chaque nœud, comme décrit dans la section [Installation des paquets et agents haute disponibilité et RHOSP](#).

Procédure

1. Sur l'un des nœuds du cluster, entrez la commande suivante pour authentifier l'utilisateur **pcs hacluster**. Dans la commande, indiquez le nom de chaque nœud de la grappe. Dans cet exemple, les nœuds de la grappe sont **node01**, **node02** et **node03**.

```
[root@node01 ~]# pcs host auth node01 node02 node03
Username: hacluster
Password:
node01: Authorized
node02: Authorized
node03: Authorized
```

2. Créez la grappe. Dans cet exemple, la grappe est nommée **newcluster**.

```
[root@node01 ~]# pcs cluster setup newcluster node01 node02 node03
...
Synchronizing pcsd certificates on nodes node01, node02, node03...
node02: Success
node03: Success
node01: Success
Restarting pcsd on the nodes in order to reload the certificates...
node02: Success
node03: Success
node01: Success
```

Vérification

1. Activer le cluster.

```
[root@node01 ~]# pcs cluster enable --all
node01: Cluster Enabled
node02: Cluster Enabled
node03: Cluster Enabled
```

2. Démarrer le cluster.

```
[root@node01 ~]# pcs cluster start --all  
node02: Starting Cluster...  
node03: Starting Cluster...  
node01: Starting Cluster...
```


CHAPITRE 6. CONFIGURATION DE LA CLÔTURE POUR UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM

La configuration de la clôture permet d'isoler automatiquement un nœud défaillant de votre cluster HA, ce qui empêche le nœud de consommer les ressources du cluster ou d'en compromettre le fonctionnement.

Utilisez l'agent de clôture **fence_openstack** pour configurer un dispositif de clôture pour un cluster HA sur RHOSP. Exécutez la commande suivante pour afficher les options de l'agent fence RHOSP.

```
# pcs stonith describe fence_openstack
```

Conditions préalables

- Un cluster HA configuré fonctionnant sous RHOSP
- Accès aux API RHOSP, à l'aide de la méthode d'authentification RHOSP que vous utiliserez pour la configuration du cluster, comme décrit dans [Méthodes d'authentification pour RHOSP](#)
- La propriété de cluster **stonith-enabled** est définie sur **true**, qui est la valeur par défaut. Red Hat ne prend pas en charge les clusters lorsque la clôture est désactivée, car cela ne convient pas à un environnement de production. Exécutez la commande suivante pour afficher les valeurs des propriétés du cluster.

```
# pcs property config --all
Cluster Properties:
...
stonith-enabled: true
```

Procédure

Effectuez les étapes suivantes à partir de n'importe quel nœud de la grappe.

1. Déterminez l'UUID de chaque nœud de votre cluster.

La commande suivante affiche la liste complète de tous les noms d'instance RHOSP au sein du projet **ha-example**, ainsi que l'UUID du nœud de cluster associé à cette instance RHOSP. Le nom d'hôte du nœud peut ne pas correspondre au nom de l'instance RHOSP.

```
# openstack --os-cloud="ha-example" server list
...
| ID                               | Name                               |...
| 6d86fa7d-b31f-4f8a-895e-b3558df9decb|testnode-node03-vm|...
| 43ed5fe8-6cc7-4af0-8acd-a4fea293bc62|testnode-node02-vm|...
| 4df08e9d-2fa6-4c04-9e66-36a6f002250e|testnode-node01-vm|...
```

2. Créez le dispositif de clôture, en utilisant le site **pcmk_host_map parameter** pour associer chaque nœud du cluster à l'UUID de ce nœud.
 - a. La commande suivante crée un dispositif de clôture **fence_openstack** pour un cluster de 3 nœuds, en utilisant un fichier de configuration **clouds.yaml** pour l'authentification. Pour **cloud= parameter**, spécifiez le nom du nuage dans votre fichier **clouds.yaml**.

```
# pcs stonith create fenceopenstack fence_openstack
pcmk_host_map="node01:4df08e9d-2fa6-4c04-9e66-
```

```
36a6f002250e;node02:43ed5fe8-6cc7-4af0-8acd-a4fea293bc62;node03:6d86fa7d-
b31f-4f8a-895e-b3558df9decb" power_timeout="240" pcmk_reboot_timeout="480"
pcmk_reboot_retries="4" cloud="ha-example"
```

- b. La commande suivante crée un dispositif de clôture **fence_openstack**, en utilisant un script d'environnement OpenRC pour l'authentification.

```
# pcs stonith create fenceopenstack fence_openstack
pcmk_host_map="node01:4df08e9d-2fa6-4c04-9e66-
36a6f002250e;node02:43ed5fe8-6cc7-4af0-8acd-a4fea293bc62;node03:6d86fa7d-
b31f-4f8a-895e-b3558df9decb" power_timeout="240" pcmk_reboot_timeout="480"
pcmk_reboot_retries="4" openrc="/root/openrc"
```

- c. La commande suivante crée un dispositif de clôture **fence_openstack**, en utilisant un nom d'utilisateur et un mot de passe pour l'authentification. Les paramètres d'authentification, notamment **username**, **password**, **project_name** et **auth_url**, sont fournis par l'administrateur RHOSP.

```
# pcs stonith create fenceopenstack fence_openstack
pcmk_host_map="node01:4df08e9d-2fa6-4c04-9e66-
36a6f002250e;node02:43ed5fe8-6cc7-4af0-8acd-a4fea293bc62;node03:6d86fa7d-
b31f-4f8a-895e-b3558df9decb" power_timeout="240" pcmk_reboot_timeout="480"
pcmk_reboot_retries="4" username="XXX" password="XXX"
project_name="rhelha" auth_url="XXX" user_domain_name="Default"
```

Vérification

1. À partir d'un nœud de la grappe, clôturez un autre nœud de la grappe et vérifiez l'état de la grappe pour voir si le nœud que vous avez clôturé est hors ligne.

```
[root@node01 ~] # pcs stonith fence node02
[root@node01 ~] # pcs status
```

2. Redémarrez le nœud que vous avez clôturé et vérifiez l'état pour vous assurer que le nœud a démarré.

```
[root@node01 ~] # pcs cluster start node02
[root@node01 ~] # pcs status
```

CHAPITRE 7. CONFIGURATION DES RESSOURCES D'UN CLUSTER HA SUR LA PLATEFORME RED HAT OPENSTACK

Utilisez les agents de ressources suivants lors de la configuration des ressources pour un cluster HA sur RHOSP :

- **openstack-info**: Vous devez configurer une ressource **openstack-info** en tant que ressource clonée pour votre cluster afin d'exécuter tout autre agent de ressource spécifique à RHOSP que l'agent de clôture **fence_openstack**.
- **openstack-virtual-ip**: Configure une ressource d'adresse IP virtuelle
- **openstack-floating-ip**: Configure une ressource d'adresses IP flottantes
- **openstack-cinder-volume**: Configure une ressource de stockage en bloc

Pour configurer les autres ressources du cluster, utilisez les agents de ressources standard de Pacemaker.

7.1. CONFIGURATION D'UNE RESSOURCE OPENSTACK-INFO (OBLIGATOIRE)

Une ressource **openstack-info** est nécessaire pour exécuter tout autre agent de ressource spécifique à RHOSP que l'agent de clôture **fence_openstack**.

La procédure suivante crée une ressource **openstack-info**. Cette procédure utilise un fichier **clouds.yaml** pour l'authentification RHOSP.

Conditions préalables

- Un cluster HA configuré fonctionnant sous RHOSP
- Accès aux API RHOSP, à l'aide de la méthode d'authentification RHOSP que vous utiliserez pour la configuration du cluster, comme décrit dans [Méthodes d'authentification pour RHOSP](#)

Procédure

Effectuez les étapes suivantes à partir de n'importe quel nœud de la grappe.

1. Pour afficher les options de l'agent de ressources **openstack-info**, exécutez la commande suivante.

```
# pcs resource describe openstack-info
```

2. Créez la ressource **openstack-info** en tant que ressource clone. Dans cet exemple, la ressource est également nommée **openstack-info**. Cet exemple utilise un fichier de configuration **clouds.yaml** et le paramètre **cloud=** est défini sur le nom du nuage dans votre fichier **clouds.yaml**.

```
# pcs resource create openstack-info openstack-info cloud="ha-example" clone
```

3. Vérifier l'état de la grappe pour s'assurer que la ressource est en cours d'exécution

```
# pcs status
```

Full List of Resources:

- * Clone Set: openstack-info-clone [openstack-info]:
- * Started: [node01 node02 node03]

7.2. CONFIGURER UNE ADRESSE IP VIRTUELLE DANS UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM

La procédure suivante crée une ressource d'adresse IP virtuelle RHOSP pour un cluster HA sur une plate-forme RHOSP. Cette procédure utilise un fichier **clouds.yaml** pour l'authentification RHOSP.

La ressource IP virtuelle RHOSP fonctionne conjointement avec une ressource de cluster **IPaddr2**. Lorsque vous configurez une ressource d'adresse IP virtuelle RHOSP, l'agent de ressource s'assure que l'infrastructure RHOSP associe l'adresse IP virtuelle à un nœud de cluster sur le réseau. Cela permet à une ressource IPadr2 de fonctionner sur ce nœud.

Conditions préalables

- Un cluster HA configuré fonctionnant sous RHOSP
- Une adresse IP assignée à utiliser comme adresse IP virtuelle
- Accès aux API RHOSP, à l'aide de la méthode d'authentification RHOSP que vous utiliserez pour la configuration du cluster, comme décrit dans [Méthodes d'authentification pour RHOSP](#)

Procédure

Effectuez les étapes suivantes à partir de n'importe quel nœud de la grappe.

1. Pour afficher les options de l'agent de ressources **openstack-virtual-ip**, exécutez la commande suivante.

```
# pcs resource describe openstack-virtual-ip
```

2. Exécutez la commande suivante pour déterminer l'ID de sous-réseau de l'adresse IP virtuelle que vous utilisez. Dans cet exemple, l'adresse IP virtuelle est 172.16.0.119.

```
# openstack --os-cloud=ha-example subnet list
+-----+ ... +-----+
| ID                | ... | Subnet      |
+-----+ ... +-----+
| 723c5a77-156d-4c3b-b53c-ee73a4f75185 | ... | 172.16.0.0/24 |
+-----+ ... +-----+
```

3. Créer la ressource adresse IP virtuelle RHOSP.
La commande suivante crée une ressource d'adresse IP virtuelle RHOSP pour une adresse IP de 172.16.0.119, en spécifiant l'ID de sous-réseau que vous avez déterminé à l'étape précédente.

```
# pcs resource create ClusterIP-osp ocf:heartbeat:openstack-virtual-ip cloud=ha-example ip=172.16.0.119 subnet_id=723c5a77-156d-4c3b-b53c-ee73a4f75185
```

- Configurez les contraintes d'ordre et d'emplacement pour vous assurer que la ressource **openstack-info** démarre avant la ressource Adresse IP virtuelle et que la ressource Adresse IP virtuelle s'exécute sur le même nœud que la ressource **openstack-info**.

```
# pcs constraint order start openstack-info-clone then ClusterIP-osp
Adding openstack-info-clone ClusterIP-osp (kind: Mandatory) (Options: first-action=start
then-action=start)
# pcs constraint colocation add ClusterIP-osp with openstack-info-clone
score=INFINITY
```

- Créez une ressource **IPAddr2** pour l'adresse IP virtuelle.

```
# pcs resource create ClusterIP ocf:heartbeat:IPAddr2 ip=172.16.0.119
```

- Configurez les contraintes d'ordre et d'emplacement pour vous assurer que la ressource **openstack-virtual-ip** démarre avant la ressource **IPAddr2** et que la ressource **IPAddr2** s'exécute sur le même nœud que la ressource **openstack-virtual-ip**.

```
# pcs constraint order start ClusterIP-osp then ClusterIP
Adding ClusterIP-osp ClusterIP (kind: Mandatory) (Options: first-action=start then-
action=start)
# pcs constraint colocation add ClusterIP with ClusterIP-osp
```

Vérification

- Vérifier la configuration des contraintes de ressources.

```
# pcs constraint config
Location Constraints:
Ordering Constraints:
  start ClusterIP-osp then start ClusterIP (kind:Mandatory)
  start openstack-info-clone then start ClusterIP-osp (kind:Mandatory)
Colocation Constraints:
  ClusterIP with ClusterIP-osp (score:INFINITY)
  ClusterIP-osp with openstack-info-clone (score:INFINITY)
```

- Vérifiez l'état de la grappe pour vous assurer que les ressources fonctionnent.

```
# pcs status
...

Full List of Resources:
* fenceopenstack (stonith:fence_openstack): Started node01
* Clone Set: openstack-info-clone [openstack-info]:
  * Started: [ node01 node02 node03 ]
* ClusterIP-osp (ocf::heartbeat:openstack-virtual-ip): Started node03
* ClusterIP (ocf::heartbeat:IPAddr2): Started node03
```

7.3. CONFIGURER UNE ADRESSE IP FLOTTANTE DANS UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM

La procédure suivante crée une ressource d'adresse IP flottante pour un cluster HA sur RHOSP. Cette procédure utilise un fichier **clouds.yaml** pour l'authentification de RHOSP.

Conditions préalables

- Un cluster HA configuré fonctionnant sous RHOSP
- Une adresse IP sur le réseau public à utiliser comme adresse IP flottante, attribuée par l'administrateur RHOSP
- Accès aux API RHOSP, à l'aide de la méthode d'authentification RHOSP que vous utiliserez pour la configuration du cluster, comme décrit dans [Méthodes d'authentification pour RHOSP](#)

Procédure

Effectuez les étapes suivantes à partir de n'importe quel nœud de la grappe.

1. Pour afficher les options de l'agent de ressources **openstack-floating-ip**, exécutez la commande suivante.

```
# pcs resource describe openstack-floating-ip
```

2. Trouvez l'ID de sous-réseau de l'adresse du réseau public que vous utiliserez pour créer la ressource d'adresse IP flottante.
 - a. Le réseau public est généralement celui qui possède la passerelle par défaut. Exécutez la commande suivante pour afficher l'adresse de la passerelle par défaut.

```
# route -n | grep ^0.0.0.0 | awk '{print $2}'
172.16.0.1
```

- b. Exécutez la commande suivante pour trouver l'ID du sous-réseau du réseau public. Cette commande génère un tableau avec des titres d'ID et de sous-réseau.

```
# openstack --os-cloud=ha-example subnet list
+-----+-----+
| ID                | | Subnet
+-----+-----+
| 723c5a77-156d-4c3b-b53c-ee73a4f75185 | | 172.16.0.0/24 |
+-----+-----+
```

3. Créez la ressource adresse IP flottante, en spécifiant l'adresse IP publique pour la ressource et l'ID de sous-réseau pour cette adresse. Lorsque vous configurez la ressource Adresse IP flottante, l'agent de ressource configure une adresse IP virtuelle sur le réseau public et l'associe à un nœud de cluster.

```
# pcs resource create float-ip openstack-floating-ip cloud="ha-example"
ip_id="10.19.227.211" subnet_id="723c5a77-156d-4c3b-b53c-ee73a4f75185"
```

4. Configurez une contrainte d'ordre pour garantir que la ressource **openstack-info** démarre avant la ressource d'adresse IP flottante.

```
# pcs constraint order start openstack-info-clone then float-ip
Adding openstack-info-clone float-ip (kind: Mandatory) (Options: first-action=start then-action=start)
```

5. Configurez une contrainte d'emplacement pour vous assurer que la ressource d'adresse IP flottante s'exécute sur le même nœud que la ressource **openstack-info**.

```
# pcs constraint colocation add float-ip with openstack-info-clone score=INFINITY
```

Vérification

1. Vérifier la configuration des contraintes de ressources.

```
# pcs constraint config
Location Constraints:
Ordering Constraints:
  start openstack-info-clone then start float-ip (kind:Mandatory)
Colocation Constraints:
  float-ip with openstack-info-clone (score:INFINITY)
```

2. Vérifiez l'état de la grappe pour vous assurer que les ressources fonctionnent.

```
# pcs status
...
Full List of Resources:
* fenceopenstack (stonith:fence_openstack): Started node01
* Clone Set: openstack-info-clone [openstack-info]:
  * Started: [ node01 node02 node03 ]
* float-ip (ocf::heartbeat:openstack-floating-ip): Started node02
```

7.4. CONFIGURATION D'UNE RESSOURCE DE STOCKAGE PAR BLOCS DANS UN CLUSTER HA SUR RED HAT OPENSTACK PLATFORM

La procédure suivante crée une ressource de stockage en bloc pour un cluster HA sur RHOSP. Cette procédure utilise un fichier **clouds.yaml** pour l'authentification de RHOSP.

Conditions préalables

- Un cluster HA configuré fonctionnant sous RHOSP
- Un volume de stockage en bloc créé par l'administrateur RHOSP
- Accès aux API RHOSP, à l'aide de la méthode d'authentification RHOSP que vous utiliserez pour la configuration du cluster, comme décrit dans [Méthodes d'authentification pour RHOSP](#)

Procédure

Effectuez les étapes suivantes à partir de n'importe quel nœud de la grappe.

1. Pour afficher les options de l'agent de ressources **openstack-cinder-volume**, exécutez la commande suivante.

```
# pcs resource describe openstack-cinder-volume
```

2. Déterminez l'ID du volume de stockage par blocs que vous configurez en tant que ressource de cluster.
Exécutez la commande suivante pour afficher un tableau des volumes disponibles comprenant l'UUID et le nom de chaque volume.

```
# openstack --os-cloud=ha-example volume list
| ID | Name |
| 23f67c9f-b530-4d44-8ce5-ad5d056ba926 | testvolume-cinder-data-disk |
```

Si vous connaissez déjà le nom du volume, vous pouvez exécuter la commande suivante, en spécifiant le volume que vous êtes en train de configurer. Cette commande affiche un tableau avec un champ ID.

```
# openstack --os-cloud=ha-example volume show testvolume-cinder-data-disk
```

3. Créer la ressource de stockage en bloc, en spécifiant l'ID du volume.

```
# pcs resource create cinder-vol openstack-cinder-volume volume_id="23f67c9f-b530-4d44-8ce5-ad5d056ba926" cloud="ha-example"
```

4. Configurez une contrainte d'ordre pour garantir que la ressource **openstack-info** démarre avant la ressource de stockage en bloc.

```
# pcs constraint order start openstack-info-clone then cinder-vol
Adding openstack-info-clone cinder-vol (kind: Mandatory) (Options: first-action=start then-action=start)
```

5. Configurez une contrainte d'emplacement pour vous assurer que la ressource de stockage de blocs s'exécute sur le même nœud que la ressource **openstack-info**.

```
# pcs constraint colocation add cinder-vol with openstack-info-clone score=INFINITY
```

Vérification

1. Vérifier la configuration des contraintes de ressources.

```
# pcs constraint config
Location Constraints:
Ordering Constraints:
  start openstack-info-clone then start cinder-vol (kind:Mandatory)
Colocation Constraints:
  cinder-vol with openstack-info-clone (score:INFINITY)
```

2. Vérifiez l'état de la grappe pour vous assurer que la ressource est en cours d'exécution.

```
# pcs status
...
Full List of Resources:
* Clone Set: openstack-info-clone [openstack-info]:
  * Started: [ node01 node02 node03 ]
* cinder-vol (ocf::heartbeat:openstack-cinder-volume): Started node03
* fenceopenstack (stonith:fence_openstack): Started node01
```