Red Hat Enterprise Linux 6

Administration de clusters

Configurer et gérer le module complémentaire High Availability

Last Updated: 2017-10-13

Red Hat Enterprise Linux 6 Administration de clusters

Configurer et gérer le module complémentaire High Availability

Red Hat Engineering Content Services
docs-need-a-fix@redhat.com

Notice légale

Résumé

Configurer et gérer le module complémentaire High Availabilitydécrit la configuration et la gestion du module complémentaire High Availability sur Red Hat Enterprise Linux 6.

Table des matières

INTRODUCTION

Ce document fournit des informations sur l'installation, la configuration et la gestion des modules complémentaires Red Hat High Availability. Les modules complémentaires Red Hat High Availability vous permettent de connecter un groupe d'ordinateurs (appelés des *nœuds* ou des *membres*) de manière à fonctionner ensemble en tant que cluster. Dans ce document, l'utilisation du mot *cluster(s)*, ou *grappe(s)*, est utilisé en faisant référence à un groupe d'ordinateurs exécutant le module complémentaire Red Hat High Availability.

Les lecteurs de ce document devraient posséder une maîtrise avancée du fonctionnement de Red Hat Enterprise Linux et comprendre les concepts des clusters, du stockage, et de l'informatique de serveurs.

Pour obtenir plus d'informations surRed Hat Enterprise Linux 6, reportez-vous aux ressources suivantes :

- *Guide d'installation Red Hat Enterprise Linux* — Fournit des informations sur l'installation de Red Hat Enterprise Linux 6.
- *Guide de déploiement Red Hat Enterprise Linux* — Fournit des informations sur le déploiement, la configuration et l'administration de Red Hat Enterprise Linux 6.

Pour obtenir plus d'informations sur le module complémentaire High Availability et les autres produits qui y sont liés pour Red Hat Enterprise Linux 6, reportez-vous aux ressources suivantes :

- *Aperçu du module complémentaire High Availability* — Fournit un aperçu de haut niveau du module complémentaire High Availability.
- *Administration du gestionnaire de volume logiques (LVM)* — Fournit une description du gestionnaire de volumes logiques LVM, y compris des informations sur l'exécution de LVM dans un environnement clusterisé.
- *Global File System 2 : Configuration et administration* — Fournit des informations sur l'installation, la configuration et la maintenance de Red Hat GFS2 (Red Hat Global File System 2), qui est inclus dans le module complémentaire Resilient Storage.
- *DM Multipath* — Fournit des informations sur l'utilisation de la fonctionnalité DM Multipath (Device-Mapper Multipath) de Red Hat Enterprise Linux 6.
- *Administration de l'Équilibreur de charges* — Fournit des informations sur la configuration de systèmes et services de haute performance avec le module complémentaire Équilibreur de charges, un ensemble de composants logiciels fournissant des serveurs virtuels Linux (LVS, de l'anglais « Linux Virtual Server ») pour équilibrer les charges IP sur un ensemble de serveurs réels.
- *Notes de publication* — Fournit des informations sur la version actuelle des produits Red Hat.

La documentation sur le module complémentaire High Availability et les autres documents de Red Hat sont disponibles sous les formats HTML, PDF et RPM sur le CD Red Hat Enterprise Linux Documentation et en ligne sur http://docs.redhat.com/docs/en-US/index.html.

1. COMMENTAIRES

Si vous identifiez une erreur typographique, ou si vous pensez à un façon d'améliorer ce manuel, faîtes-nous en part. Veuillez soumettre un rapport dans Bugzilla (http://bugzilla.redhat.com/bugzilla/) sous le composant **doc-Cluster_Administration**.

Assurez-vous de bien mentionner l'identifiant du manuel :

```
Cluster_Administration(EN)-6 (2013-2-15T16:26)
```

En mentionnant l'identifiant de ce manuel, nous pouvons voir la version exacte du guide que vous possédez.

Si vous avez des suggestions pour améliorer la documentation, essayez d'être aussi précis que possible. Si vous avez trouvé une erreur, veuillez inclure le numéro de la section ainsi que des portions du texte qui l'entoure afin que nous puissions la retrouver plus facilement.

CHAPITRE 1. APERÇU DE LA GESTION ET DE LA CONFIGURATION DU MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

Le module complémentaire Red Hat High Availability vous permet de connecter un groupe d'ordinateurs (appelés des *nœuds* ou des *membres*) pour qu'ils fonctionnent ensemble en tant que cluster. Vous pouvez utiliser le module complémentaire Red Hat High Availability afin de répondre à vos besoins en clustering (par exemple, installer un cluster pour partager des fichiers sur un système de fichiers GFS2 ou installer un basculement de service).

NOTE

Pour obtenir des informations sur les meilleures pratiques pour déployer et mettre à jour des clusters Red Hat Enterprise Linux à l'aide des modules complémentaires High Availability (Haute disponibilité) et GFS2 (« Red Hat Global File System 2 »), reportez-vous à l'article « Red Hat Enterprise Linux Cluster, High Availability, and GFS Deployment Best Practices » sur le Portail client Red Hat à l'adresse : . https://access.redhat.com/kb/docs/DOC-40821.

Ce chapitre fournit un résumé des fonctionnalités et mises à jour ajoutées au module complémentaire Red Hat High Availability depuis la publication initiale de Red Hat Enterprise Linux 6, suivi par un aperçu de la configuration et de la gestion du module complémentaire Red Hat High Availability.

1.1. NOUVELLES FONCTIONNALITÉS ET FONCTIONNALITÉS MODIFIÉES

Cette section répertorie les fonctionnalités nouvelles et modifiées de la documentation du module complémentaire Red Hat High Availability qui ont été ajoutées depuis la sortie initiale de Red Hat Enterprise Linux 6.

1.1.1. Nouvelles fonctionnalités et fonctionnalités modifiées de Red Hat Enterprise Linux 6.1

Red Hat Enterprise Linux 6.1 inclut la documentation et les mises à jour et modifications des fonctionnalités suivantes :

- À partir de Red Hat Enterprise Linux 6.1 et ses versions plus récentes, le module complémentaire Red Hat High Availability offre maintenant la prise en charge des interruptions SNMP. Pour obtenir des informations sur la configuration des interruptions SNMP avec le module complémentaire Red Hat High Availability, reportez-vous au Chapitre 10, *Configuration SNMP avec le module complémentaire Red Hat High Availability*.
- À partir de Red Hat Enterprise Linux 6.1 et ses versions plus récentes, le module complémentaire Red Hat High Availability offre maintenant la prise en charge de la commande de configuration du cluster **ccs**. Pour obtenir des informations sur la commande **ccs**, reportez-vous au Chapitre 5, *Configurer le module complémentaire Red Hat High Availability avec la commande **ccs*** et au Chapitre 6, *Gérer le module complémentaire Red Hat High Availability avec **ccs***.
- La documentation sur la configuration et la gestion du logiciel du module complémentaire Red Hat High Availability à l'aide de Conga a été mise à jour afin de refléter la mise à jour des écrans Conga et la prise en charge des fonctionnalités.

- Pour Red Hat Enterprise Linux 6.1 et ses versions plus récentes, l'utilisation de **ricci** requiert un mot de passe la première fois que vous propagerez une configuration de cluster mise à jour depuis n'importe quel nœud en particulier. Pour obtenir des informations sur **ricci** reportez-vous à la Section 2.13, « Considérations pour **ricci** ».

- Vous pouvez maintenant spécifier une politique d'échec *Restart-Disable* (Redémarrer-désactiver) pour un service, indiquant au système de tenter de redémarrer le service à sa place s'il devait échouer, et de désactiver le service si le redémarrage échouait aussi au lieu de le déplacer sur un autre hôte dans le cluster. Cette fonctionnalité est documentée dans la Section 3.10, « Ajouter un service cluster à un cluster » l'Annexe B, *Paramètres des ressources HA*.

- Vous pouvez maintenant configurer un sous-arbre indépendant comme non-critique, indiquant que si la ressource échoue alors seule cette ressource est désactivée. Pour obtenir des informations sur cette fonctionnalité, voir la Section 3.10, « Ajouter un service cluster à un cluster » et la Section C.4, « Récupération de défaillance et sous-arbres indépendants ».

- Ce document inclut maintenant le nouveau Chapitre 9, *Diagnostiquer et corriger des problèmes dans un cluster*.

En outre, de petites corrections et clarifications ont été effectuées sur le document.

1.1.2. Nouvelles fonctionnalités et fonctionnalités modifiées de Red Hat Enterprise Linux 6.2

Red Hat Enterprise Linux 6.2 inclut la documentation et les mises à jour et modifications des fonctionnalités suivantes.

- Red Hat Enterprise Linux fournit maintenant du support pour exécuter « Clustered Samba » sous une configuration active/active. Pour obtenir des informations sur les configurations de Samba clusterisé, reportez-vous au Chapitre 11, *Configuration de Samba en cluster*.

- Même si tout utilisateur en mesure de s'authentifier sur le système hébergeant **luci** peut se connecter à **luci**, à partir de Red Hat Enterprise Linux 6.2, seul l'utilisateur root du système exécutant **luci** peut accéder à tous les composants **luci** jusqu'à ce qu'un administrateur (l'utilisateur root, ou un utilisateur avec des permissions d'administrateur) définisse les permissions pour cet utilisateur. Pour obtenir des informations sur la définition des permissions **luci** pour les utilisateurs, reportez-vous à la Section 3.3, « Contrôler l'accès à luci ».

- Les nœuds d'un cluster peuvent communiquer entre eux en utilisant le mécanisme de transport de monodiffusion UDP. Pour obtenir des informations sur la configuration de la monodiffusion UDP, veuillez vous reporter à la Section 2.12, « Trafic de monodiffusion UDP ».

- Vous pouvez maintenant configurer certains aspects du comportement de **luci** par le biais du fichier **/etc/sysconfig/luci**. Par exemple, vous pouvez configurer spécifiquement l'unique adresse IP à laquelle **luci** est servi. Pour obtenir des informations sur la configuration de l'unique adresse IP à laquelle **luci** est servi, reportez-vous au Tableau 2.2, « Port IP activé sur un ordinateur exécutant **luci** ». Pour obtenir des informations sur le fichier **/etc/sysconfig/luci** en général, reportez-vous à la Section 2.4, « Configurer **luci** avec **/etc/sysconfig/luci** ».

- La commande **ccs** inclut maintenant l'option **--lsfenceopts**, qui imprime une liste des périphériques fence disponibles, ainsi que l'option **--lsfenceopts** *fence_type*, qui imprime chaque type fence disponible. Pour obtenir des informations sur ces options, reportez-vous à la Section 5.6, « Répertorier les périphériques fence et les options de périphériques fence ».

- La commande **ccs** inclut maintenant l'option **--lsserviceopts**, qui imprime une liste des services cluster actuellement disponibles pour votre cluster, ainsi que l'option **--lsserviceopts** *service_type*, qui imprime une liste des options que vous pouvez spécifier pour un type de service particulier. Pour obtenir des informations sur ces options, reportez-vous à la Section 5.11, « Répertorier les services cluster disponibles ».

- Red Hat Enterprise Linux 6.2 fournit le support pour l'agent fence VMware (interface SOAP). Pour obtenir des informations sur les paramètres des périphériques fence, reportez-vous à l'Annexe A, *Paramètres des périphériques fence*.

- Red Hat Enterprise Linux 6.2 fournit le support pour l'agent fence RHEV-M REST API, avec RHEV 3.0 et versions plus récentes. Pour obtenir des informations sur les paramètres des périphériques fence, reportez-vous à l'Annexe A, *Paramètres des périphériques fence*.

- À partir de Red Hat Enterprise Linux 6.2, lorsque vous configurez une machine virtuelle dans un cluster avec la commande **ccs**, vous pourrez utiliser l'option **--addvm** (plutôt que l'option **addservice**). Ceci assure la définition correcte de la ressource **vm** directement sous le nœud de configuration **rm** dans le fichier de configuration du cluster. Pour obtenir des informations sur la configuration des ressources de machines virtuelles avec la commande **ccs**, reportez-vous à la Section 5.12, « Ressources de machine virtuelle ».

- Ce document inclut un nouvel annexe, Annexe D, *Vérification des ressources de service de cluster et délai de basculement*. Cet annexe décrit comment **rgmanager** surveille le statut des ressources de clusters et comment modifier l'intervalle des vérifications de statut. L'annexe décrit aussi le paramètre de service **__enforce_timeouts**, qui indique qu'un délai d'inactivité pour une opération causera à un service d'échouer.

- Ce document inclut une nouvelle section, la Section 2.3.3, « Configurer le pare-feu iptables pour autoriser des composants de clusters ». Cette section affiche le filtrage que vous pouvez utiliser pour autoriser le trafic de multidiffusion à travers le pare-feu **iptables** pour les divers composants du cluster.

En outre, de petites corrections et clarifications ont été effectuées sur le document.

1.1.3. Nouvelles fonctionnalités et fonctionnalités modifiées de Red Hat Enterprise Linux 6.3

Red Hat Enterprise Linux 6.3 inclut la documentation, les mises à jour et les modifications des fonctionnalités suivantes :

- Red Hat Enterprise Linux 6.3 offre le support de l'agent de ressources **condor**. Pour obtenir des informations sur les paramètres de ressources HA, reportez-vous à l'Annexe B, *Paramètres des ressources HA*.

- Ce document inclut maintenant un nouvel annexe, Annexe F, *LVM haute disponibilité (HA-LVM)*.

- Les informations présentes dans ce document clarifient quels sont les changements de configuration qui requièrent le redémarrage d'un cluster. Pour obtenir un résumé de ces changements, reportez-vous à la Section 9.1, « Les changements de configuration ne prennent pas effet ».

- La documentation souligne maintenant que **luci** possède délai d'inactivité qui vous déconnecte après 15 minutes d'inactivité. Pour obtenir des informations sur le démarrage de **luci**, reportez-vous à la Section 3.2, « Démarrage de **luci** ».

- Le périphérique fence **`fence_ipmilan`** prend en charge un paramètre de niveau de privilège. Pour obtenir des informations sur les paramètres de périphérique fence, reportez-vous à l'Annexe A, *Paramètres des périphériques fence*.

- Ce document inclut maintenant une nouvelle section, Section 2.14, « Configurer des machines virtuelles dans un environnement clusterisé ».

- Ce document inclut maintenant une nouvelle section, Section 4.6, « Effectuer une copie de sauvegarde et restaurer une configuration Luci ».

- Ce document inclut maintenant une nouvelle section, Section 9.4, « Échec du démon cluster ».

- Ce document fournit des informations sur le paramétrage d'options de débogage dans la Section 5.14.4, « Journalisation », Section 7.7, « Configurer les options de débogage », et la Section 9.13, « La journalisation du débogage pour le DLM (« Distributed Lock Manager », ou gestionnaire de verrous distribués) doit être activée ».

- À partir de Red Hat Enterprise Linux 6.3, l'utilisateur root ou un utilisateur possédant des permissions d'administrateur **luci** peut aussi utiliser l'interface **luci** pour ajouter des utilisateurs au système, comme le décrit la Section 3.3, « Contrôler l'accès à luci ».

- À partir de Red Hat Enterprise Linux 6.3, la commande **`ccs`** valide la configuration selon le schéma du cluster de **`/usr/share/cluster/cluster.rng`** sur le nœud que vous spécifiez avec l'option **`-h`**. Auparavant, la commande **`ccs`** utilisait toujours le schéma du cluster qui était empaqueté avec la commande **`ccs`** elle-même, **`/usr/share/ccs/cluster.rng`** sur le système local. Pour obtenir des informations sur la validation de configuration, reportez-vous à la Section 5.1.6, « Validation de la configuration ».

- Les tableaux décrivant les paramètres de périphérique fence dans l'Annexe A, *Paramètres des périphériques fence* ainsi que les tableaux décrivant les paramètres de ressources HA dans l'Annexe B, *Paramètres des ressources HA* incluent maintenant les noms de ces paramètres comme ils apparaissent dans le fichier **`cluster.conf`**.

En outre, de petites corrections et clarifications ont été effectuées sur le document.

1.1.4. Nouvelles fonctionnalités et fonctionnalités modifiées de Red Hat Enterprise Linux 6.4

Red Hat Enterprise Linux 6.4 inclut la documentation, les mises à jour et les modifications des fonctionnalités suivantes :

- Red Hat Enterprise Linux 6.4 fournit la prise en charge de l'agent fence du contrôleur d'alimentation réseau Eaton (Interface SNMP), de l'agent fence de HP BladeSystem et de l'agent fence d'IBM iPDU. Pour obtenir des informations sur les paramètres des périphériques fence, veuillez vous référer à l'Annexe A, *Paramètres des périphériques fence*.

- L'Annexe B, *Paramètres des ressources HA* fournit maintenant une description de l'agent de ressources du serveur NFS.

- À partir de Red Hat Enterprise Linux 6.4, l'utilisateur root ou un utilisateur possédant des permissions d'administrateur **luci** peut aussi utiliser l'interface **luci** pour supprimer des utilisateurs du système. Ceci est documenté dans la Section 3.3, « Contrôler l'accès à luci ».

- L'Annexe B, *Paramètres des ressources HA* fournit une description du nouveau paramètre **`nfsrestart`** pour les ressources HA GFS2 et le système de fichiers.

- Ce document inclut une nouvelle section, Section 5.1.5, « Commandes remplaçant les paramètres précédents ».
- Section 2.3, « Activation des ports IP » inclut maintenant des informations sur le filtrage de **igmp** sur le pare-feu **iptables**.
- L'agent fence IPMI LAN prend maintenant en charge un paramètre pour configurer le niveau de privilèges sur le périphérique IPMI, comme documenté dans l'Annexe A, *Paramètres des périphériques fence*.
- En outre du mode de liaison Ethernet 1, les modes de liaisons 0 et 2 sont maintenant pris en charge pour les communications inter-nœuds dans un cluster. Des conseils pour les résolutions de problèmes dans ce document qui vous suggèrent de vous assurer que vous utilisez bien uniquement les modes de liaisons pris en charge ont pris note de cette addition.
- Les périphériques balisés VLAN sont maintenant pris en charge pour les communications cardiaques de cluster. Les conseils des résolutions de problèmes qui indiquaient que ceci n'est pas pris en charge ont été supprimés de ce document.
- Le module Red Hat High Availability prend maintenant en charge la configuration du protocole d'anneau redondant. Pour obtenir des informations générales sur l'utilisation de cette fonctionnalité et sur la configuration du fichier de configuration **cluster.conf**, reportez-vous à la Section 7.6, « Configurer le protocole d'anneau redondant (« Redundant Ring ») ». Pour obtenir des informations sur la configuration du protocole d'anneau redondant avec **luci**, reportez-vous à la Section 3.5.4, « Configurer le protocole d'anneau redondant (« Redundant Ring ») ». Pour obtenir des informations sur la configuration du protocole d'anneau redondant avec la commande **ccs**, reportez-vous à la Section 5.14.5, « Configurer le protocole d'anneau redondant (« Redundant Ring ») ».

En outre, de petites corrections et clarifications ont été effectuées sur le document.

1.2. BASES DE CONFIGURATION

Pour paramétrer un cluster, vous devez connecter les nœuds à certains matériaux du cluster et configurer les nœuds dans l'environnement du cluster. Configurer et gérer le module complémentaire Red Hat High Availability consiste des étapes de base suivantes :

1. Installation du matériel. Reportez-vous à la Section 1.3, « Installation du matériel ».
2. Installation du logiciel du module complémentaire Red Hat High Availability. Reportez-vous à la Section 1.4, « Installer le logiciel du module complémentaire Red Hat High Availability ».
3. Configuration du module complémentaire Red Hat High Availability. Reportez-vous à la Section 1.5, « Configurer le logiciel du module complémentaire Red Hat High Availability ».

1.3. INSTALLATION DU MATÉRIEL

L'installation du matériel consiste en la connexion des nœuds du cluster au reste du matériel requis pour exécuter le module complémentaire Red Hat High Availability. La quantité et le type de matériel varie selon le but et les pré-requis disponibles du cluster. Typiquement, un cluster de niveau entreprise requiert le type de matériel suivant (reportez-vous à la Figure 1.1, « Aperçu du matériel du module complémentaire Red Hat High Availability »). Pour voir les considérations à prendre en compte sur le matériel et les autres sujets de préoccupation de configuration du cluster, reportez-vous au Chapitre 2, *Avant de configurer le module complémentaire Red Hat High Availability (Haute Disponibilité)* ou vérifiez avec un représentant autorisé de Red Hat.

- Nœuds de cluster — Ordinateurs capables d'exécuter le logiciel Red Hat Enterprise Linux 6, avec au moins 1Go de RAM.
- Commutateur ou concentrateur Ethernet pour réseau public — Ceci est requis pour que le client puisse accéder au cluster.
- Commutateur ou concentrateur Ethernet pour réseau privé — Ceci est requis pour la communication entre les nœuds du cluster et le reste du matériel du cluster, comme les commutateurs d'alimentation réseau et les interrupteurs Fibre Channel.
- Commutateur d'alimentation du réseau — Un commutateur d'alimentation du réseau est recommandé pour effectuer le fencing dans un cluster de niveau entreprise.
- Commutateur Fibre Channel — Un commutateur Fibre Channel fournit l'accès au stockage Fibre Channel. D'autres options sont disponibles pour le stockage selon le type d'interface de stockage, iSCSI par exemple. Un commutateur Fibre Channel peut être configuré de manière à effectuer le fencing.
- Stockage — Un certain type de stockage est requis pour un cluster. Le type requis dépend du but du cluster.

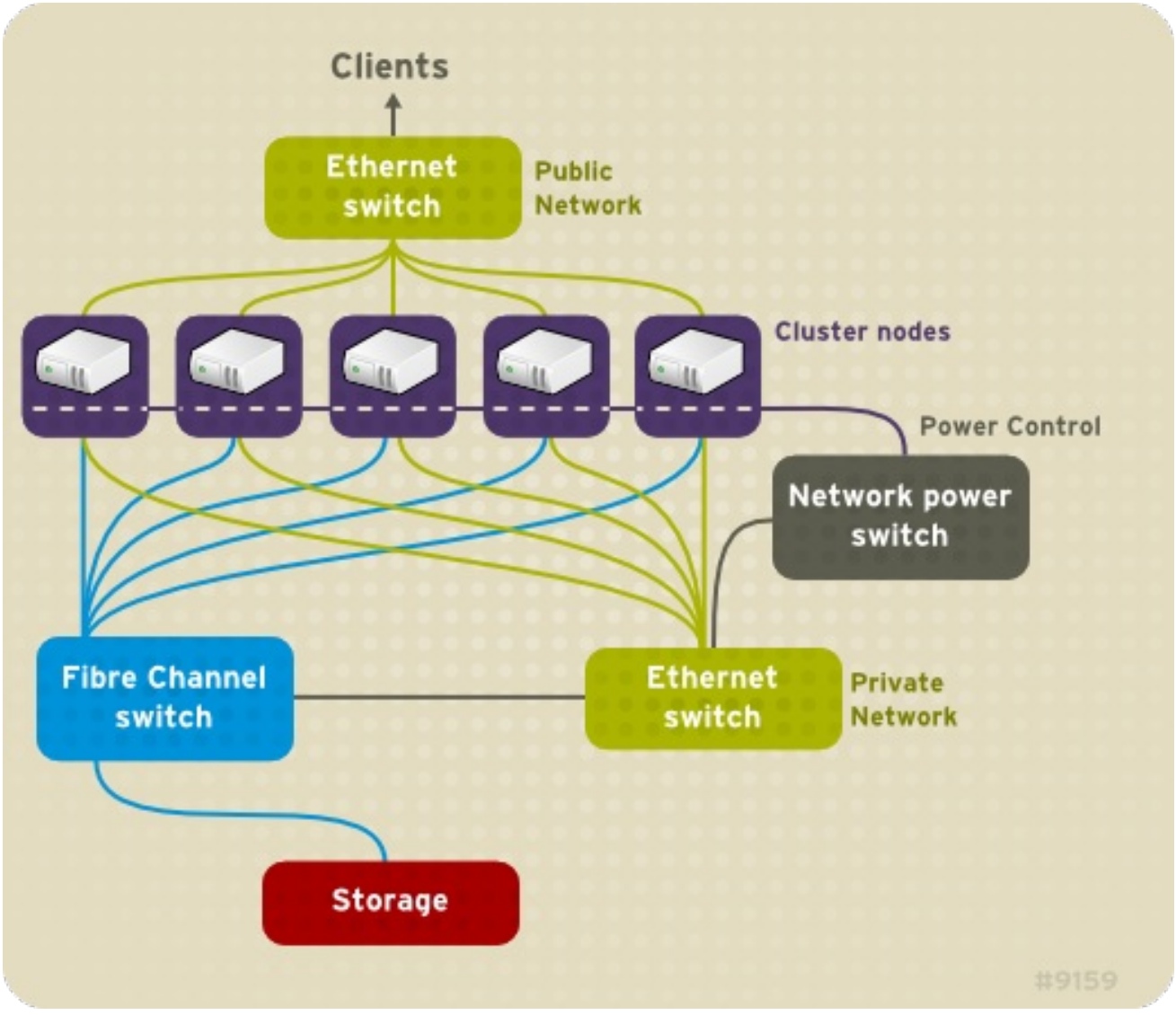


Figure 1.1. Aperçu du matériel du module complémentaire Red Hat High Availability

1.4. INSTALLER LE LOGICIEL DU MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

Pour installer le logiciel du module Red Hat High Availability, vous devez posséder des droits d'accès au logiciel. Si vous utilisez l'interface utilisateur graphique **luci**, vous pouvez la laisser installer logiciel du cluster. Si vous utilisez d'autres outils pour configurer le cluster, veuillez installer et sécuriser le logiciel comme vous le feriez avec d'autres logiciels Red Hat Enterprise Linux.

Vous pouvez utiliser la commande **`yum install`** pour installer les paquetages des logiciels du module complémentaire Red Hat High Availability :

```
# yum install rgmanager lvm2-cluster gfs2-utils
```

Remarque qu'installer **`rgmanager`** uniquement téléchargera toutes les dépendances nécessaires pour créer un cluster HA depuis le canal HighAvailability. Les paquetages **`lvm2-cluster`** et **`gfs2-utils`** font partie du canal ResilientStorage et pourraient ne pas être nécessaires sur votre site.

Mise à niveau du logiciel du module complémentaire Red Hat High Availability

Il est possible de mettre à niveau le logiciel du cluster sur une version majeure de Red Hat Enterprise Linux sans sortir le cluster de la production. Ce faire requiert de désactiver le logiciel du cluster sur un hôte à la fois, de mettre le logiciel à niveau, puis de le redémarrer sur cet hôte.

1. Éteignez tous les services du cluster sur un seul nœud de cluster. Pour obtenir des instructions sur l'arrêt du logiciel du cluster sur un nœud, reportez-vous à la Section 8.1.2, « Arrêter un logiciel de cluster ». Il peut être désirable de déplacer manuellement les services gérés par le cluster et les machines virtuelles hors de l'hôte avant d'arrêter **`rgmanager`**.

2. Veuillez exécuter la commande **`yum update`** pour mettre à jour les paquetages installés.

3. Redémarrez le nœud du cluster ou redémarrez les services du cluster manuellement. Pour obtenir des instructions sur le démarrage du logiciel du cluster sur un nœud, reportez-vous à la Section 8.1.1, « Démarrer un logiciel de cluster ».

1.5. CONFIGURER LE LOGICIEL DU MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

Configurer le logiciel du module complémentaire Red Hat High Availability consiste en l'utilisation d'outils de configuration pour spécifier les relations entre les composants du cluster. Les outils de configuration du cluster suivants sont disponibles avec le module complémentaire Red Hat High Availability :

- **Conga** — Interface utilisateur complète pour l'installation, la configuration et la gestion du module complémentaire Red Hat High Availability. Reportez-vous au Chapitre 3, *Configurer le module complémentaire Red Hat High Availability avec **Conga*** et au Chapitre 4, *Gérer le module complémentaire Red Hat High Availability avec **Conga*** pour obtenir des informations sur la configuration et la gestion du module complémentaire High Availability avec **Conga**.

- La commande **`ccs`** — Cette commande configure et gère le module complémentaire Red Hat High Availability. Reportez-vous au Chapitre 5, *Configurer le module complémentaire Red Hat High Availability avec la commande **ccs*** et au Chapitre 6, *Gérer le module complémentaire Red Hat High Availability avec **ccs*** pour obtenir des informations sur la configuration et la gestion du module complémentaire High Availability avec la commande **`ccs`**.

- Outils de ligne de commande — Ensemble d'outils de ligne de commande pour la configuration

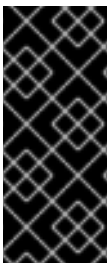
et la gestion du module complémentaire Red Hat High Availability. Reportez-vous au Chapitre 7, *Configurer le module complémentaire Red Hat High Availability avec des outils de ligne de commande* et au Chapitre 8, *Gérer le module complémentaire Red Hat High Availability avec des outils de ligne de commande* pour obtenir des informations sur la configuration et la gestion d'un cluster avec des outils de ligne de commande. Reportez-vous à l'Annexe E, *Résumé des outils de la ligne de commande* pour obtenir un résumé des outils de ligne de commande préférés.

NOTE

`system-config-cluster` n'est pas disponible dans Red Hat Enterprise Linux 6.

CHAPITRE 2. AVANT DE CONFIGURER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY (HAUTE DISPONIBILITÉ)

Ce chapitre décrit les tâches à effectuer et les considérations à prendre en compte avant de procéder à l'installation et à la configuration du module complémentaire Red Hat High Availability. Ce chapitre est composé des sections suivantes.



IMPORTANT

Assurez-vous que le déploiement du module complémentaire Red Hat High Availability correspond bien à vos besoins et peut être pris en charge. Consultez un représentant autorisé de Red Hat pour vérifier votre configuration avant de la déployer. En outre, prévoyez suffisamment de temps pour une période de rodage de la configuration afin de tester les différents modes d'échec.

2.1. CONSIDÉRATIONS POUR UNE CONFIGURATION GÉNÉRALE

Vous pouvez configurer le module complémentaire Red Hat High Availability de différentes manières afin de mieux correspondre à vos besoins. Prenez en compte les considérations générales suivantes lorsque vous planifiez, configurez et implémentez votre déploiement.

Le nombre de nœuds de cluster pris en charge

Le nombre maximum de nœuds de cluster pris en charge par le module complémentaire High Availability (Haute Disponibilité) est 16.

Clusters de site unique

Seuls les clusters pour site unique sont entièrement pris en charge à l'heure actuelle. Les clusters s'étalant sur de multiples emplacements physiques ne sont pas officiellement pris en charge. Pour obtenir plus de détails et discuter des clusters sur de multiples sites, veuillez consulter votre représentant du support ou des ventes Red Hat.

GFS2

Même si un système de fichiers GFS2 peut être implémenté sur un système autonome ou en tant que partie d'une configuration de cluster, Red Hat ne prend pas en charge GFS2 en tant que système de fichiers à nœud unique. Red Hat prend en charge un certain nombre de systèmes de fichiers à nœud unique de haute performance qui sont optimisés pour un nœud unique et possèdent ainsi un plafond plus bas qu'un système de fichiers de cluster. Red Hat recommande l'utilisation de ces systèmes de fichiers plutôt que GFS2 dans le cas où un nœud unique doit monter le système de fichiers. Red Hat continuera à prendre en charge les systèmes de fichiers GFS2 à nœud unique pour ses clients existants.

Lorsque vous configurer un système de fichiers GFS2 en tant que système de fichiers de cluster, vous devez vous assurer que tous les nœuds du cluster ont accès au système de fichiers partagé. Les configurations de clusters asymétriques dans lesquelles certains nœuds ont accès au système de fichiers et pas d'autres ne sont pas prises en charge. Ceci ne requiert pas que tous les nœuds montent le système de fichiers GFS2.

Configuration du matériel sans point de défaillance unique (No-single-point-of-failure hardware configuration)

Les clusters peuvent inclure une matrice RAID à double contrôleur, de multiples canaux réseau liés, de multiples chemins d'accès entre les membres du cluster et le stockage, et des systèmes onduleurs (UPS, de l'anglais « un-interruptible power supply ») afin de s'assurer qu'aucune défaillance unique ne résulte en temps d'inactivité ou en perte de données.

Alternativement, un cluster de bas coût peut être installé pour offrir moins de disponibilité qu'un cluster sans point de défaillance unique. Par exemple, vous pouvez paramétrer un cluster avec une matrice RAID à contrôleur unique et un seul canal Ethernet.

Certaines alternatives à bas coût, comme les contrôleurs RAID hôtes, les RAID logiciels sans prise en charge de clusters, et les configurations parallèles SCSI avec multi-initiateur ne sont pas compatibles, ou ne sont pas appropriées pour une utilisation en tant que stockage partagé de cluster.

Assurance d'intégrité des données

Pour s'assurer de l'intégrité des données, seul un nœud peut exécuter un service de cluster et accéder aux données du cluster-service à la fois. L'utilisation d'interrupteurs d'alimentation dans la configuration du matériel du cluster active un nœud pour alimenter le cycle d'alimentation d'un autre nœud avant de redémarrer les services HA de ce nœud pendant le processus de basculement. Ceci empêche les deux nœuds d'accéder simultanément aux données et de les corrompres. Des *périphériques fence* (des solutions matérielles ou logicielles pouvant allumer, éteindre et redémarrer des nœuds de clusters à distance) sont utilisés pour garantir l'intégrité des données sous toutes conditions d'échec.

Liaison de canal Ethernet

Le quorum du cluster et la santé du nœud sont déterminés par la communication de messages parmi les nœuds du cluster via Ethernet. En outre, les nœuds de clusters utilisent Ethernet pour tout un éventail d'autres fonctions critiques de clusters (pour le fencing par exemple). Avec la liaison de canaux Ethernet, de multiples interfaces Ethernet sont configurées de manière à se comporter

comme une seule interface, réduisant ainsi le risque de défaillance d'un point unique dans la connexion Ethernet commutée habituelle parmi les nœuds de clusters et le reste du matériel du cluster.

À partir de Red Hat Enterprise Linux 6.4, les modes de liaisons 0, 1 et 2 sont pris en charge.

IPv4 et IPv6

Le module complémentaire High Availability prend en charge les protocoles Internet IPv4 et IPv6. La prise en charge de IPv6 par le module complémentaire High Availability est une nouveauté de Red Hat Enterprise Linux 6.

2.2. MATÉRIEL COMPATIBLE

Avant de configurer le logiciel du module complémentaire Red Hat High Availability, assurez-vous que votre cluster utilise le matériel approprié (avec la prise en charge des périphériques fence, des périphériques de stockage et des interrupteurs Fibre Channel par exemple). Reportez-vous aux instructions de configuration du matériel sur http://www.redhat.com/cluster_suite/hardware/ pour obtenir les informations les plus récentes sur la compatibilité du matériel.

2.3. ACTIVATION DES PORTS IP

Avant de déployer le module complémentaire Red Hat High Availability, vous devez activer certains ports IP sur les nœuds de clusters et ordinateurs qui exécutent **luci** (le serveur de l'interface utilisateur **Conga**). Les sections suivantes identifient les ports IP à activer :

- Section 2.3.1, « Activation des ports IP sur des nœuds de clusters »
- Section 2.3.2, « Activer le port IP pour **luci** »

La section suivante fournit les règles **`iptables`** pour activer les ports IP nécessaires au module complémentaire Red Hat High Availability :

- Section 2.3.3, « Configurer le pare-feu iptables pour autoriser des composants de clusters »

2.3.1. Activation des ports IP sur des nœuds de clusters

Pour permettre aux nœuds dans un cluster de communiquer entre eux, vous devez activer les ports IP assignés à certains composants du module complémentaire Red Hat High Availability. Tableau 2.1, « Ports IP activés sur les nœuds du module complémentaire Red Hat High Availability » répertorie les numéros des ports IP, leurs protocoles respectifs, ainsi que les composants auxquels les numéros de ports sont assignés. À chaque nœud de cluster, activez les ports IP selon Tableau 2.1, « Ports IP activés sur les nœuds du module complémentaire Red Hat High Availability ». Vous pouvez utiliser **`system-config-firewall`** pour activer les ports IP.

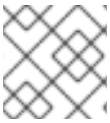
Tableau 2.1. Ports IP activés sur les nœuds du module complémentaire Red Hat High Availability

Numéro de port IP	Protocole	Composant
5404, 5405	UDP	**`corosync/cman`** (Gestionnaire du cluster)
11111	TCP	**`ricci`** (propage les informations mises à jour du cluster)

Numéro de port IP	Protocole	Composant
21064	TCP	**dlm** (Gestionnaire de verrous distribués)
16851	TCP	**modclusterd**

2.3.2. Activer le port IP pour luci

Pour permettre aux ordinateurs clients de communiquer avec un ordinateur qui exécute **luci** (le serveur de l'interface utilisateur **Conga**), vous devez activer le port IP assigné à **luci**. Sur chaque ordinateur qui exécute **luci**, activez le port IP comme indiqué dans le Tableau 2.2, « Port IP activé sur un ordinateur exécutant **luci** ».



NOTE

Si un nœud de cluster exécute **luci**, le port 11111 devrait déjà être activé.

Tableau 2.2. Port IP activé sur un ordinateur exécutant luci

Numéro de port IP	Protocole	Composant
8084	TCP	**luci** (serveur de l'interface utilisateur **Conga**)

À partir de Red Hat Enterprise Linux 6.1, qui permet l'activation de la configuration par le biais du fichier **/etc/sysconfig/luci**, vous pouvez spécifiquement configurer l'unique adresse IP à laquelle **luci** est servi. Vous pouvez utiliser cette capacité sir l'infrastructure de votre serveur incorpore plus d'un réseau et que vous souhaitez accéder à **luci** depuis le réseau interne uniquement. Pour ce faire, veuillez « décommenter » et modifier la ligne dans le fichier spécifiant **host**. Par exemple, pour modifier le paramètre **host** dans le fichier sur 10.10.10.10, modifiez la ligne **host** comme suit :

```
host = 10.10.10.10
```

Pour obtenir plus d'informations sur le fichier **/etc/sysconfig/luci**, reportez-vous à la Section 2.4, « Configurer **luci** avec **/etc/sysconfig/luci** ».

2.3.3. Configurer le pare-feu iptables pour autoriser des composants de clusters

Ci-dessous figure une liste des règles iptables pour activer les ports IP nécessaires à Red Hat Enterprise Linux 6 (avec le module High Availability). Remarquez que ces exemples utilisent 192.168.1.0/24 comme sous-réseau, mais vous devrez remplacer 192.168.1.0/24 par le sous-réseau approprié si vous utilisez ces règles.

Pour **cman** (Gestionnaire de clusters), veuillez utiliser le filtrage suivant.

```
$ iptables -I INPUT -m state --state NEW -m multiport -p udp -s
192.168.1.0/24 -d 192.168.1.0/24 --dports 5404,5405 -j ACCEPT
$ iptables -I INPUT -m addrtype --dst-type MULTICAST -m state --state NEW
-m multiport -p udp -s 192.168.1.0/24 --dports 5404,5405 -j ACCEPT
```

-

Pour **dlm** (Gestionnaire de verrous distribués, « Distributed Lock Manager ») :

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 21064 -j ACCEPT
```

Pour **ricci** (qui fait partie de l'agent distant de Conga) :

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 11111 -j ACCEPT
```

Pour **modclusterd** (qui fait partie de l'agent distant de Conga) :

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 16851 -j ACCEPT
```

Pour **luci** (serveur de l'interface utilisateur Conga) :

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 16851 -j ACCEPT
```

Pour **igmp** (protocole de gestion de groupes internet « Internet Group Management Protocol ») :

```
$ iptables -I INPUT -p igmp -j ACCEPT
```

Après avoir exécuté ces commandes, veuillez exécuter la commande suivante pour enregistrer la configuration actuelle afin que les changements soient persistants lors des redémarrages.

```
$ service iptables save ; service iptables restart
```

2.4. CONFIGURER LUCI AVEC /ETC/SYSCONFIG/LUCI

À partir de Red Hat Enterprise Linux 6.1, vous pouvez configurer certains aspects du comportement de **luci** par le biais du fichier **/etc/sysconfig/luci**. Les paramètres que vous pouvez modifier avec ce fichier incluent les paramètres auxiliaires de l'environnement d'exécution utilisés par le script init ainsi que la configuration du serveur. En outre, vous pouvez modifier ce fichier afin de changer certains paramètres de configuration de l'application. Des instructions sont fournies dans le fichier, celles-ci décrivent les paramètres de configuration pouvant être changés en modifiant ce fichier.

Afin de protéger le format destiné, vous ne devriez pas modifier les lignes de non-configuration du fichier **/etc/sysconfig/luci** lorsque vous modifiez le fichier. En outre, vous devez prendre soin de bien suivre la syntaxe requise pour ce fichier, particulièrement dans la section **INITSCRIPT**, qui n'autorise pas d'espaces blancs autour du signe égal et qui requiert que vous utilisiez des guillemets pour enfermer les chaînes contenant des espaces.

L'exemple suivant indique comment modifier le port par lequel **luci** est servi en modifiant le fichier **/etc/sysconfig/luci**.

1. Décommentez la ligne suivante dans le fichier **/etc/sysconfig/luci** :

   ```
   #port = 4443
   ```

2. Remplacez 4443 par le numéro de port souhaité, qui peut être plus grand que ou égal à 1024 (qui n'est pas un port privilégié). Par exemple, vous pouvez modifier cette ligne du fichier comme suit pour définir le port par lequel **luci** est servi sur 8084.

```
port = 8084
```

3. Redémarrez le service **luci** pour que les modifications prennent effet.

IMPORTANT

Lorsque vous modifiez un paramètre de configuration dans le fichier **/etc/sysconfig/luci** pour redéfinir une valeur par défaut, vous devriez prendre soin de bien utiliser la valeur à la place de la valeur documentée par défaut. Par exemple, lorsque vous modifiez le port sur lequel **luci** est servi, assurez-vous de bien spécifier la valeur modifiée lors de l'activation d'un port IP pour **luci**, comme le décrit la Section 2.3.2, « Activer le port IP pour **luci** ».

Les paramètres du port et de l'hôte modifiés seront automatiquement reflétés dans l'URL affiché lorsque le service **luci** démarre, comme le décrit la Section 3.2, « Démarrage de **luci** ». Vous devriez utiliser cet URL pour accéder à **luci**.

Pour obtenir plus d'informations sur les paramètres que vous pouvez configurer avec le fichier **/etc/sysconfig/luci**, reportez-vous à la documentation dans le fichier même.

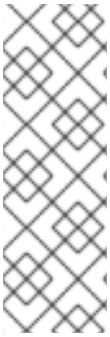
2.5. CONFIGURER L'ACPI POUR UNE UTILISATION AVEC DES PÉRIPHÉRIQUES FENCE INTÉGRÉS

Si votre cluster utilise des périphériques fence intégrés, vous devez configurer l'ACPI (de l'anglais, « Advanced Configuration and Power Interface ») afin que la clôture s'effectue de manière complète et immédiate.

NOTE

Pour obtenir les informations les plus récentes sur les périphériques fence intégrés pris en charge par le module complémentaire Red Hat High Availability, reportez-vous à http://www.redhat.com/cluster_suite/hardware/.

Si un nœud de cluster est configuré pour être clos par un périphérique fence intégré, désactivez l'ACPI Soft-Off sur ce nœud. La désactivation de l'ACPI Soft-Off permet à un périphérique fence intégré d'arrêter un nœud complètement et immédiatement plutôt que de tenter d'effectuer un arrêt normal (par exemple avec **shutdown -h now**). Si l'ACPI Soft-Off est activé, un périphérique fence intégré peut prendre quatre secondes ou plus pour arrêter un nœud (voir la remarque suivante). En outre, si l'ACPI Soft-Off est activé et qu'un nœud panique ou se fige lors de l'arrêt, un périphérique fence intégré pourrait ne pas réussir à arrêter le nœud. Dans ces circonstances, la clôture est retardée ou mise en échec. Ainsi, lorsqu'un nœud est clos avec un périphérique fence intégré et qu'ACPI Soft-Off est activé, un cluster devra être récupéré lentement ou nécessitera une intervention administrative.



NOTE

Le temps requis pour clore un nœud dépend du périphérique fence intégré utilisé. Certains périphériques fence intégrés effectuent l'équivalent de lorsque le bouton d'alimentation est pressé et maintenu ; ainsi, le périphérique fence éteint le nœud en quatre à cinq secondes. D'autres périphériques fence intégrés effectuent l'équivalent de lorsque le bouton d'alimentation est pressé momentanément, se fiant au système d'exploitation pour arrêter le nœud, dans ce cas, le laps de temps pris par le périphérique fence pour éteindre le nœud est bien plus long que quatre à cinq secondes.

Pour désactiver l'ACPI Soft-Off, utilisez la gestion **chkconfig** et vérifiez que le nœud s'arrête immédiatement lorsqu'il est « fenced ». La manière préférée de désactiver l'ACPI Soft-Off est avec la gestion **chkconfig**. Cependant, si cette méthode n'est pas satisfaisante pour votre cluster, vous pouvez désactiver ACPI Soft-Off à l'aide de l'une des méthodes suivantes :

- Modifiez le paramètre BIOS sur "instant-off" ou sur un autre paramètre équivalent qui arrêtera le nœud sans délai

  NOTE

  Désactiver l'ACPI Soft-Off avec le BIOS peut ne pas être possible sur certains ordinateurs.

- Ajouter **acpi=off** à la ligne de commande de démarrage du noyau du fichier **/boot/grub/grub.conf**.

  IMPORTANT

  Cette méthode désactive complètement l'ACPI ; certains ordinateurs ne démarrent pas correctement si l'ACPI est complètement désactivé. Utilisez cette méthode *uniquement* si les autres méthodes ne sont pas effectives sur votre cluster.

Les sections suivantes fournissent des procédures pour la méthode préférée et les méthodes alternatives de désactivation de l'ACPI Soft-Off :

- Section 2.5.1, « Désactivation de l'ACPI Soft-Off avec la gestion **chkconfig** » — Méthode préférée
- Section 2.5.2, « Désactivation de l'ACPI Soft-Off avec le BIOS » — Première méthode alternative
- Section 2.5.3, « Complètement désactiver ACPI dans le fichier **grub.conf** » — Seconde méthode alternative

2.5.1. Désactivation de l'ACPI Soft-Off avec la gestion chkconfig

Vous pouvez utiliser la gestion **chkconfig** pour désactiver l'ACPI Soft-Off soit en supprimant le démon ACPI (**acpid**) de la gestion **chkconfig** ou en éteignant **acpid**.

NOTE

Ceci est la méthode préférée pour désactiver l'ACPI Soft-Off.

Désactivez l'ACPI Soft-Off avec la gestion **chkconfig** sur chaque nœud du cluster comme suit :

1. Exécutez l'une des commandes suivantes :
   - **chkconfig --del acpid** — Cette commande supprime **acpid** de la gestion **chkconfig**.

     — OU —

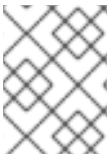
   - **chkconfig --level 2345 acpid off** — Cette commande éteint **acpid**.
2. Redémarrez le nœud.
3. Lorsque le cluster est configuré et en cours d'exécution, vérifiez que le nœud s'éteint immédiatement lorsqu'il est « fenced ».

NOTE

Vous pouvez clore le nœud avec la commande **fence_node** ou **Conga**.

2.5.2. Désactivation de l'ACPI Soft-Off avec le BIOS

Méthode préférée de désactivation de l'ACPI Soft-Off avec la gestion **chkconfig** (Section 2.5.1, « Désactivation de l'ACPI Soft-Off avec la gestion **chkconfig** »). Cependant, si la méthode préférée ne fonctionne pas sur votre cluster, suivez la procédure décrite dans cette section.

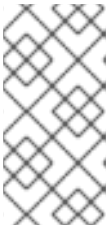


NOTE

Désactiver l'ACPI Soft-Off avec le BIOS peut ne pas être possible sur certains ordinateurs.

Vous pouvez désactiver l'ACPI Soft-Off en configurant le BIOS de chaque nœud de cluster comme suit :

1. Redémarrez le nœud et lancez le programme **BIOS CMOS Setup Utility**.
2. Accédez au menu **Power** (ou à un autre menu de gestion de l'alimentation).
3. Dans le menu **Power**, ajustez la fonction **Soft-Off by PWR-BTTN** (ou son équivalent) sur **Instant-Off** (ou sur le paramètre équivalent qui arrête le nœud sans délai via le bouton d'alimentation). L'Exemple 2.1, « **BIOS CMOS Setup Utility** : **Soft-Off by PWR-BTTN** paramétré sur **Instant-Off** » montre un menu **Power** avec la fonction **ACPI Function** paramétrée sur **Enabled** (activé) et **Soft-Off by PWR-BTTN** paramétrée sur **Instant-Off**.



NOTE

Les équivalents de **ACPI Function**, **Soft-Off by PWR-BTTN**, et **Instant-Off** peuvent varier grandement selon les ordinateurs. Cependant, l'objectif de cette procédure est de configurer le BIOS de manière à ce que l'ordinateur puisse être éteint via le bouton de l'alimentation sans délais.

4. Quittez le programme **BIOS CMOS Setup Utility** en enregistrant la configuration BIOS.

5. Lorsque le cluster est configuré et en cours d'exécution, vérifiez que le nœud s'éteint immédiatement lorsqu'il est « fenced ».

NOTE

Vous pouvez clore le nœud avec la commande **fence_node** ou **Conga**.

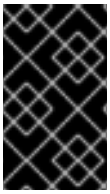
Exemple 2.1. BIOS CMOS Setup Utility : Soft-Off by PWR-BTTN paramétré sur Instant-Off

```
+---------------------------------------------|-------------------+
|    ACPI Function             [Enabled]      |    Item Help      |
|    ACPI Suspend Type         [S1(POS)]      |-------------------|
|  x Run VGABIOS if S3 Resume   Auto          |   Menu Level   *  |
|    Suspend Mode              [Disabled]     |                   |
|    HDD Power Down            [Disabled]     |                   |
|    Soft-Off by PWR-BTTN      [Instant-Off   |                   |
|    CPU THRM-Throttling       [50.0%]        |                   |
|    Wake-Up by PCI card       [Enabled]      |                   |
|    Power On by Ring          [Enabled]      |                   |
|    Wake Up On LAN            [Enabled]      |                   |
|  x USB KB Wake-Up From S3     Disabled      |                   |
|    Resume by Alarm           [Disabled]     |                   |
|  x  Date(of Month) Alarm       0            |                   |
|  x  Time(hh:mm:ss) Alarm       0 :  0 :     |                   |
|    POWER ON Function         [BUTTON ONLY   |                   |
|  x KB Power ON Password       Enter         |                   |
|  x Hot Key Power ON           Ctrl-F1       |                   |
|                                             |                   |
|                                             |                   |
+---------------------------------------------|-------------------+
```

Cet exemple montre la fonction **ACPI Function** paramétrée sur **Enabled** (activé) et **Soft-Off by PWR-BTTN** paramétré sur **Instant-Off**.

2.5.3. Complètement désactiver ACPI dans le fichier `grub.conf`

La méthode préférée pour désactiver l'ACPI Soft-Off est avec la gestion **chkconfig** (Section 2.5.1, « Désactivation de l'ACPI Soft-Off avec la gestion **chkconfig** »). Si la méthode préférée n'est pas effective sur votre cluster, vous pouvez désactiver l'ACPI Soft-Off avec la gestion de l'alimentation BIOS (Section 2.5.2, « Désactivation de l'ACPI Soft-Off avec le BIOS »). Si aucune de ces méthodes ne fonctionne sur votre cluster, vous pouvez aussi complètement désactiver l'ACPI en ajoutant **acpi=off** à la ligne de commande du démarrage du noyau dans le fichier **grub.conf**.

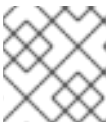


IMPORTANT

Cette méthode désactive complètement l'ACPI ; certains ordinateurs ne démarrent pas correctement si l'ACPI est complètement désactivé. Utilisez cette méthode *uniquement* si les autres méthodes ne sont pas effectives sur votre cluster.

Vous pouvez complètement désactiver l'ACPI en modifiant le fichier **grub.conf** de chaque nœud du cluster comme suit :

1. Ouvrez **/boot/grub/grub.conf** à l'aide d'un éditeur de texte.
2. Ajoutez **acpi=off** à la ligne de commande du démarrage du noyau dans **/boot/grub/grub.conf** (reportez-vous à l'Exemple 2.2, « Ligne de commande du démarrage du noyau avec **acpi=off** ajouté »).
3. Redémarrez le nœud.
4. Lorsque le cluster est configuré et en cours d'exécution, vérifiez que le nœud s'éteint immédiatement lorsqu'il est « fenced ».



NOTE

Vous pouvez clore le nœud avec la commande **fence_node** ou **Conga**.

Exemple 2.2. Ligne de commande du démarrage du noyau avec acpi=off ajouté

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this
file
# NOTICE:  You have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/mapper/vg_doc01-lv_root
#          initrd /initrd-[generic-]version.img
#boot=/dev/hda
default=0
timeout=5
serial --unit=0 --speed=115200
terminal --timeout=5 serial console
title Red Hat Enterprise Linux Server (2.6.32-193.el6.x86_64)
        root (hd0,0)
        kernel /vmlinuz-2.6.32-193.el6.x86_64 ro
root=/dev/mapper/vg_doc01-lv_root console=ttyS0,115200n8 acpi=off
        initrd /initramrs-2.6.32-131.0.15.el6.x86_64.img
```

Dans cet exemple, **acpi=off** a été ajouté à la ligne de commande du démarrage du noyau — la ligne commençant par "kernel /vmlinuz-2.6.32-193.el6.x86_64.img".

2.6. CONSIDÉRATIONS POUR LA CONFIGURATION DES SERVICES HA

Vous pouvez créer un cluster convenant à vos besoins pour la haute disponibilité (HA, ou High Availability) en configurant les services HA. Le composant-clé pour la gestion des services HA dans le module complémentaire Red Hat High Availability, **rgmanager**, implémente un basculement à froid pour des applications prises sur étagère (COTS). Dans le module complémentaire Red Hat High Availability, une application est configurée avec d'autres ressources de cluster pour former un service HA qui peut basculer d'un nœud de cluster à un autre sans interruption apparente aux clients du cluster. Le

basculement HA-service peut se produire si un nœud de cluster est en échec ou si un administrateur système de clusters déplace le service d'un nœud de cluster à un autre par exemple, lors du temps d'indisponibilité planifié d'un nœud de cluster).

Pour créer un service HA, vous devez le configurer dans le fichier de configuration du cluster. Un service HA est composé de *ressources* du cluster. Les ressources du cluster sont des blocs de construction que vous créez et gérez dans le fichier de configuration du cluster — par exemple, une adresse IP, un script d'initialisation d'applications, ou une partition partagée Red Hat GFS2.

Pour maintenir l'intégrité des données, un service HA peut être exécuté sur un seul nœud de cluster à la fois. Vous pouvez spécifier les priorités des basculements dans un domaine de basculement. La spécification des priorités de basculements revient à déterminer le niveau de priorité de chaque nœud dans un domaine de basculement. Le niveau de priorité détermine l'ordre de basculement — déterminant ainsi sur quel nœud un service HA devrait basculer. Si vous ne spécifiez pas de priorités de basculement, un service HA peut alors basculer sur n'importe quel nœud dans son domaine de basculement. Vous pouvez aussi spécifier si un service HA est restreint de manière à uniquement s'exécuter sur les nœuds du domaine de basculement qui lui est associé. (Lorsque associé à un domaine de basculement non-restreint, un service HA peut démarrer sur n'importe quel nœud si aucun des membres du domaine de basculement n'est disponible.)

La Figure 2.1, « Exemple de service de cluster de serveur web » montre un exemple d'un service HA qui est un serveur web nommé "content-webserver". Celui-ci est exécuté dans le nœud B du cluster et se trouve dans un domaine de basculement consistant des nœuds A, B et D. En outre, le domaine de basculement est configuré avec une priorité de basculements vers le nœud D avant le nœud A et avec une restriction de basculements vers les nœuds de ce domaine de basculements uniquement. Le service HA comprend ces ressources de cluster :

- Ressource adresse IP — IP address 10.10.10.201.
- Ressource d'application nommée "httpd-content" — script d'initialisation d'application de serveur web **`/etc/init.d/httpd`** (spécifiant **`httpd`**).
- Ressource de système de fichiers — Red Hat GFS2 nommé "gfs2-content-webserver".

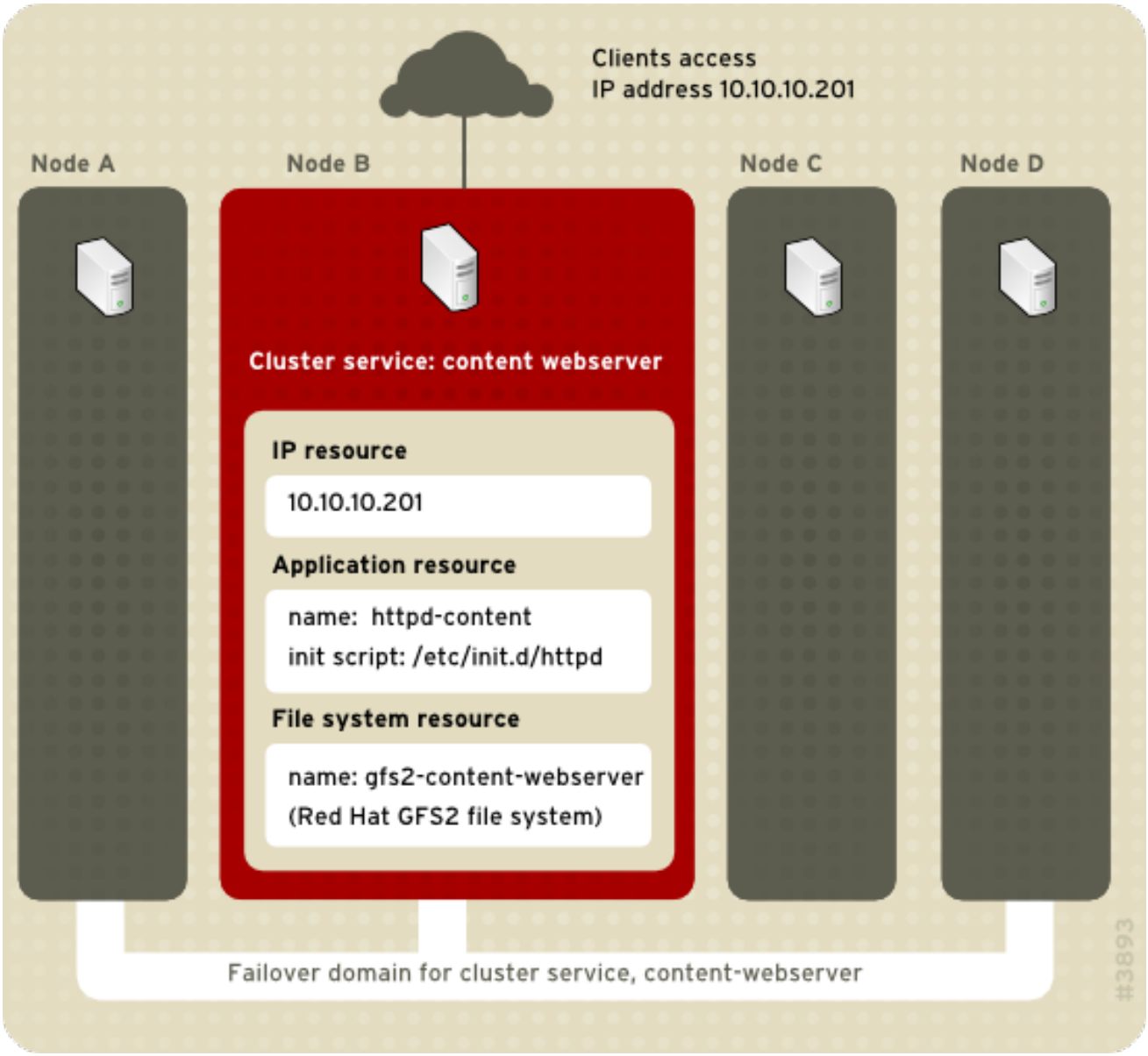
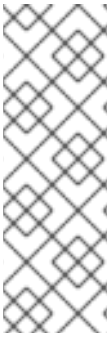


Figure 2.1. Exemple de service de cluster de serveur web

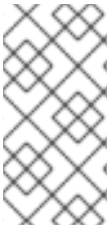
Les clients accèdent au service HA via l'adresse IP 10.10.10.201, activant l'interaction avec l'application du serveur web httpd-content. L'application httpd-content utilise le système de fichiers gfs2-content-webserver. Si le nœud B devait échouer, le service HA content-webserver basculera sur le nœud D. Si le nœud D n'est pas disponible ou s'il échouait aussi, le service basculera sur le nœud A. Le basculement se produira avec une interruption minimale du service des clients du cluster. Par exemple, dans un service HTTP, certaines informations sur l'état peuvent être perdues (comme avec les données de session). Le service HA sera accessible depuis un autre nœud du cluster via la même adresse IP que celle précédant le basculement.



NOTE

Pour obtenir plus d'informations sur les services HA et sur les domaines de basculements, reportez-vous à l'*Aperçu du module complémentaire High Availability*. Pour obtenir des informations sur la configuration des domaines de basculement, reportez-vous au Chapitre 3, *Configurer le module complémentaire Red Hat High Availability avec* ***Conga*** (avec **Conga**) où au Chapitre 7, *Configurer le module complémentaire Red Hat High Availability avec des outils de ligne de commande* (avec des utilitaires en ligne de commande).

Un service HA est un groupe de ressources de cluster configurées en une entité cohérente fournissant des services spécialisés aux clients. Un service HA est représenté comme une arborescence de ressources dans le fichier de configuration du cluster **/etc/cluster/cluster.conf** (dans chaque nœud du cluster). Dans le fichier de configuration du cluster, chaque arborescence de ressources est une représentation XML spécifiant chaque ressource, ses attributs, et ses relations aux autres ressources dans l'arborescence des ressources (parents, enfants et de même parenté).



NOTE

Comme un service HA est composé de ressources organisées en une arborescence hiérarchique, on peut parfois faire référence à un service en tant qu'*arborescence de ressources* ou que *groupe de ressources*. Les deux termes sont synonymes de *service HA*.

À la racine de chaque arborescence de ressources se trouve un type de ressources spécial — une *ressource de service*. Les autres types de ressources comprennent le reste d'un service, déterminant ainsi ses caractéristiques. Configurer un service HA revient à créer une ressource de service, créer des ressources de cluster subordonnées et les organiser en une entité cohérente conforme aux restrictions hiérarchiques du service.

Deux considérations majeures sont à prendre en compte lors de la configuration d'un service HA :

- Le type de ressources nécessaires à la création du service
- Les relations entre les parents, les enfants et les enfants de mêmes parents dans les ressources

Le type de ressources et la hiérarchie de celles-ci dépendent du type de service que vous configurez.

Les types de ressources de clusters sont répertoriés dans l'Annexe B, *Paramètres des ressources HA*. Des informations sur les relations entre les parents, les enfants, les enfants de même parents et les ressources sont décrites dans l'Annexe C, *Comportement des ressources HA*.

2.7. VALIDATION DE LA CONFIGURATION

La configuration du cluster est automatiquement validée selon le schéma du cluster sur **/usr/share/cluster/cluster.rng** au moment du démarrage et lorsqu'une configuration est rechargée. Vous pouvez aussi valider une configuration de cluster à tout moment en utilisant la commande **ccs_config_validate**. Pour obtenir des informations sur la validation de configuration lors de l'utilisation de la commande **ccs**, voir la Section 5.1.6, « Validation de la configuration ».

Un schéma annoté est disponible sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

La validation de la configuration vérifie les erreurs de base suivantes :

- Validité XML — Vérifie que le fichier de configuration est un fichier XML valide.
- Options de configuration — Vérifie que les options (éléments XML et attributs) sont valides.
- Valeurs des options — Vérifie que les options contiennent des données valides (limité).

Les exemples suivants montrent une configuration valide et des configurations invalides qui illustrent les vérifications de validation :

- Configuration valide — Exemple 2.3, « **cluster.conf** Exemple de configuration : Fichier valide »
- XML invalide — Exemple 2.4, « **cluster.conf** Exemple de configuration : XML invalide »
- Option invalide — Exemple 2.5, « **cluster.conf** Exemple de configuration : Option invalide »
- Valeur de l'option invalide — Exemple 2.6, « **cluster.conf** Exemple de configuration : Valeur de l'option invalide »

Exemple 2.3. `cluster.conf` Exemple de configuration : Fichier valide

```
<cluster name="mycluster" config_version="1">
  <logging debug="off"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Exemple 2.4. `cluster.conf` Exemple de configuration : XML invalide

```
<cluster name="mycluster" config_version="1">
  <logging debug="off"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
```

```
      </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
<cluster>         <----------------INVALID
```

Dans cet exemple, il manque une barre oblique à la dernière ligne de la configuration (annotée comme "INVALID") — il s'agit de **<cluster>** au lieu de **</cluster>**.

Exemple 2.5. `cluster.conf` Exemple de configuration : Option invalide

```
<cluster name="mycluster" config_version="1">
  <loging debug="off"/>         <----------------INVALID
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
<cluster>
```

Dans cet exemple, la seconde ligne de la configuration (annotée comme "INVALID") contient un élément XML invalide — il s'agit de **loging** au lieu de **logging**.

Exemple 2.6. `cluster.conf` Exemple de configuration : Valeur de l'option invalide

```
<cluster name="mycluster" config_version="1">
  <loging debug="off"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="-1">  <--------
INVALID
         <fence>
```

```
            </fence>
        </clusternode>
        <clusternode name="node-02.example.com" nodeid="2">
            <fence>
            </fence>
        </clusternode>
        <clusternode name="node-03.example.com" nodeid="3">
            <fence>
            </fence>
        </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
<cluster>
```

Dans cet exemple, la quatrième ligne de la configuration (annotée comme "INVALID") contient une valeur invalide pour l'attribut XML **nodeid** dans la ligne **clusternode** de **node-01.example.com**. La valeur est une valeur négative ("-1") au lieu d'une valeur positive ("1"). Pour l'attribut **nodeid**, la valeur doit être une valeur positive.

2.8. CONSIDÉRATIONS POUR NETWORKMANAGER

L'utilisation de **NetworkManager** n'est pas prise en charge sur les nœuds de clusters. Si vous n'avez pas installé **NetworkManager** sur vos nœuds de clusters, vous devriez le supprimer ou le désactiver.

NOTE

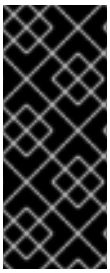
Le service **cman** ne démarrera pas si **NetworkManager** est exécuté ou a été configuré de manière à s'exécuter avec la commande **chkconfig**.

2.9. CONSIDÉRATIONS POUR UTILISER LE DISQUE QUORUM

Le disque Quorum est un démon de quorum basé sur disque, **qdiskd**, qui fournit des heuristiques supplémentaires pour déterminer la santé des nœuds. Avec les heuristiques, vous pouvez déterminer des facteurs importants à l'opération des nœuds dans le cas d'une partition de réseau. Par exemple, dans un cluster à quatre nœuds avec un partage 3:1 ; habituellement, les trois nœuds "gagnent" automatiquement grâce à leur majorité. Sous ces circonstances, le nœud "seul" est clos. Cependant, avec **qdiskd**, vous pouvez paramétrer des heuristiques qui permettent à ce nœud de gagner en se basant sur l'accès à une ressource critique (par exemple, un chemin d'accès de réseau critique). Si votre cluster nécessite des méthodes supplémentaires pour déterminer la santé des nœuds, vous devriez configurer **qdiskd** afin de répondre à ces besoins.

NOTE

La configuration de **qdiskd** n'est pas requise à moins que vous n'ayez des besoins spéciaux pour la santé des nœuds. La configuration "all-but-one" (tous sauf un) est exemple de besoin spécifique. Dans ce type de configuration, **qdiskd** est configuré afin de fournir suffisamment de votes de quorum pour maintenir le quorum même si un seul nœud travaille.



IMPORTANT

En général, les heuristiques et autres paramètres **qdiskd** de votre déploiement dépendent de l'environnement du site et des besoins spécifiques nécessités. Pour comprendre l'utilisation des heuristiques et des autres paramètres **qdiskd**, reportez-vous à la page man qdisk(5). Si vous nécessitez de l'aide pour comprendre et utiliser **qdiskd** pour votre site, veuillez contacter un représentant du support Red Hat autorisé.

Si vous devez utiliser **qdiskd**, vous devriez prendre en compte les considérations suivantes :

Votes de nœuds de clusters

Lors de l'utilisation du disque Quorum, chaque nœud de cluster doit avoir un vote.

Valeur du délai d'expiration de l'appartenance à CMAN

La valeur du délai d'expiration de l'appartenance à CMAN (la durée sans réponse du nœud avant que CMAN considère le nœud comme étant 'mort', donc n'étant plus membre) devrait être au moins deux fois plus élevé que la valeur du délai d'expiration de l'appartenance à **qdiskd**. La raison pour ceci est que le démon du quorum doit pouvoir détecter les nœuds en échec par lui-même et celui-ci peut prendre bien plus longtemps pour ce faire que le CMAN. La valeur par défaut du délai d'expiration de l'appartenance à CMAN est de 10 secondes. D'autres conditions spécifiques au site peuvent affecter la relation entre les valeurs des délais d'expiration des adhésions à CMAN et à **qdiskd**. Pour obtenir de l'aide avec l'ajustement de la valeur du délai d'expiration de l'appartenance à CMAN, veuillez contacter un représentant du support Red Hat autorisé.

Clôtures (Fencing)

Pour garantir une clôture fiable lors de l'utilisation de **qdiskd**, utiliser le power fencing. Alors que les autres types de fencing peuvent être fiables avec les clusters qui ne sont pas configurés avec **qdiskd**, ceux-ci ne sont pas fiables pour un cluster configuré avec **qdiskd**.

Nombre maximum de nœuds

Un cluster configuré avec **qdiskd** prend en charge un maximum de 16 nœuds. La raison pour cette limite est liée à l'évolutivité, l'augmentation du compte des nœuds augmente les conflits d'E/S synchrones sur le périphérique du disque quorum.

Périphérique de disque quorum

Un périphérique de disque quorum doit être un périphérique bloc partagé avec accès lecture/écriture simultané par tous les nœuds d'un cluster. La taille minimum du périphérique bloc est de 10 méga-octets. Une matrice RAID SCSI multiports, un SAN RAID Fibre Channel, une cible iSCSI configuré RAID sont des exemples de périphériques blocs partagés pouvant être utilisés par **qdiskd**. Vous pouvez créer un périphérique de disque quorum avec **mkqdisk**, l'utilitaire du disque Quorum de clusters. Pour obtenir des informations sur l'utilisation de l'utilitaire, reportez-vous à la page man mkqdisk(8).

NOTE

Utiliser JBOD comme disque quorum n'est pas recommandé. Un JBOD ne peut pas fournir une performance fiable et ne pourrait ainsi ne pas permettre à un nœud d'écrire dessus assez rapidement. Si un nœud n'est pas en mesure d'écrire sur un périphérique de disque quorum assez rapidement, le nœud sera alors incorrectement expulsé du cluster.

2.10. MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY ET SELINUX

Le module complémentaire High Availability pour Red Hat Enterprise Linux 6 prend en charge SELinux dans l'état **`enforcing`** avec le type de politique SELinux défini sur **`targeted`**.

Pour obtenir plus d'informations sur SELinux, reportez-vous au *Guide de déploiement* de Red Hat Enterprise Linux 6.

2.11. ADRESSES DE MULTIDIFFUSION

Les nœuds dans une grappe communiquent entre eux à l'aide d'adresses de multi-diffusion. Ainsi, chaque commutateur réseau et équipement de réseau associé dans le module Red Hat High Availability doit être configuré de manière à activer les adresses de multidiffusion et à prendre en charge le protocole IGMP (« Internet Group Management Protocol »). Assurez-vous que chaque commutateur réseau et équipement de réseau associé dans le module Red Hat High Availability soit bien en mesure de prendre en charge les adresses de multidiffusion ert IGMP. Si c'est le cas, assurez-vous que l'adressage de multidiffusion et IGMP soient bien activé. Sans la multidiffusion et IGMP, tous les nœuds ne pourront pas participer dans la grappe, causant l'échec de celle-ci. Veuillez utiliser la monodiffusion UDP dans ces environnements, comme décrit dans la Section 2.12, « Trafic de monodiffusion UDP ».

NOTE

Les procédures de configuration des commutateurs réseau et des équipements réseau associés varient selon le produit. Reportez-vous à la documentation appropriée du fournisseur ou à d'autres informations sur la configuration des commutateurs réseau et des équipements réseau associés pour activer les adresses de multidiffusion et IGMP.

2.12. TRAFIC DE MONODIFFUSION UDP

À partir de Red Hat Enterprise Linux 6.2, les nœuds dans une grappe peuvent communiquer entre eux à l'aide du mécanisme de transport de monodiffusion UDP. Il est recommandé d'utiliser la multidiffusion IP pour le réseau de grappes. La monodiffusion UDP est une alternative pouvant être utilisée lorsque la multidiffusion IP est indisponible.

Vous pouvez configurer le module complémentaire Red Hat High-Availability afin d'utiliser la monodiffusion UDP en définissant le paramètre **`cman transport="udpu"`** dans le fichier de configuration **`cluster.conf`**. Vous pouvez aussi spécifier la monodiffusion à partir de la page **Configuration réseau** de l'interface utilisateur **Conga**, comme le décrit la Section 3.5.3, « Configuration du réseau ».

2.13. CONSIDÉRATIONS POUR `RICCI`

Dans Red Hat Enterprise Linux 6, **ricci** remplace **ccsd**. Ainsi, il est nécessaire que **ricci** soit exécuté dans chaque nœud de cluster pour pouvoir propager la configuration du cluster mis à jour, que ce soit via la commande **cman_tool version -r**, via la commande **ccs**, ou via le serveur de l'interface utilisateur **luci**. Vous pouvez démarrer **ricci** en utilisant **service ricci start** ou en l'autorisant à s'exécuter lors du démarrage via **chkconfig**. Pour obtenir des informations sur l'activation des ports IP pour **ricci**, reportez-vous à la Section 2.3.1, « Activation des ports IP sur des nœuds de clusters ».

Dans Red Hat Enterprise Linux 6.1 et ses versions plus récentes, l'utilisation de **ricci** requiert un mot de passe la première fois que vous propagez une configuration mise à jour d'un cluster depuis n'importe quel nœud en particulier. Définissez le mot de passe de **ricci** après avoir installé **ricci** sur votre système avec la commande **passwd ricci** pour l'utilisateur **ricci**.

2.14. CONFIGURER DES MACHINES VIRTUELLES DANS UN ENVIRONNEMENT CLUSTERISÉ

Lorsque vous configurez votre cluster avec des ressources de machine virtuelle, vous devriez utiliser les outils **rgmanager** pour démarrer et arrêter les machines virtuelles. L'utilisation de **virsh** pour démarrer une machine peut faire que celle-ci soit exécutée dans plusieurs emplacements, ce qui peut corrompre les données de la machine virtuelle.

Pour réduire les chances qu'un administrateur effectue un « double-démarrage » accidentel de machines virtuelles, en utilisant des outils cluster et des outils non-cluster dans un environnement en grappe (clusterisé), vous pouvez configurer votre système en stockant les fichiers de configuration de la machine virtuelle dans un emplacement qui n'est pas l'emplacement par défaut. Stocker les fichiers de configuration ailleurs que sur l'emplacement par défaut fait qu'il est plus difficile de lancer une machine virtuelle par erreur avec **virsh** car l'emplacement du fichier de configuration sera inconnu à**virsh**.

L'emplacement non par défaut des fichiers de configuration de la machine virtuelle peut être n'importe où. L'avantage apporté par l'utilisation d'un partage NFS ou par un système de fichiers GFS2 partagé réside dans le fait que l'administrateur ne doit pas conserver les fichiers de configuration synchronisés à travers les membres du cluster. Cependant, il est aussi permissible d'utiliser un répertoire local tant que l'administrateur conserve le contenu synchronisé à travers la totalité du cluster.

Dans la configuration du cluster, les machines virtuelles peuvent faire référence à cet emplacement qui n'est pas celui par défaut en utilisant l'attribut **path** d'une ressource de machine virtuelle. Remarquez que l'attribut **path** est un répertoire ou un ensemble de répertoires séparés par le caractère des deux-points « : », il ne s'agit pas du chemin vers un fichier spécifique.

AVERTISSEMENT

Le service **libvirt-guests** devrait être désactivé sur tous les nœuds qui exécutent **rgmanager**. Si une machine virtuelle démarre automatiquement (« autostart ») ou reprend, ceci peut résulter en la machine virtuelle étant exécutée dans plusieurs emplacements, ce qui peut corrompre les données de la machine virtuelle.

Pour obtenir des informations sur les attributs des ressources d'une machine virtuelle, reportez-vous au Tableau B.24, « Virtual Machine ».

CHAPITRE 3. CONFIGURER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY AVEC CONGA

Ce chapitre décrit comment configurer le logiciel du module complémentaire Red Hat High Availability à l'aide de **Conga**. Pour obtenir des informations sur l'utilisation de **Conga** pour gérer un cluster en cours d'exécution, voir le Chapitre 4, *Gérer le module complémentaire Red Hat High Availability avec **Conga***.

NOTE

Conga est l'interface utilisateur graphique pouvant être utilisée pour administrer le module complémentaire Red Hat High Availability. Remarquez cependant que pour utiliser cette interface efficacement, vous devez avoir une bonne compréhension des concepts sous-jacents. L'apprentissage de la configuration des clusters en explorant les fonctionnalités disponibles dans l'interface utilisateur n'est pas recommandé, car ceci pourrait résulter en un système qui n'est pas suffisamment robuste pour que tous les services puisse continuer à s'exécuter lorsqu'un composant échoue.

Ce chapitre est composé des sections suivantes :

- Section 3.1, « Tâches de configuration »
- Section 3.2, « Démarrage de **luci** »
- Section 3.3, « Contrôler l'accès à luci »
- Section 3.4, « Créer un cluster »
- Section 3.5, « Propriétés globales du cluster »
- Section 3.6, « Configurer des périphériques fence »
- Section 3.7, « Configurer le fencing pour les membres du cluster »
- Section 3.8, « Configurer un domaine de basculement »
- Section 3.9, « Configurer les ressources globales du cluster »
- Section 3.10, « Ajouter un service cluster à un cluster »

3.1. TÂCHES DE CONFIGURATION

La configuration du logiciel du module complémentaire Red Hat High Availability avec **Conga** consiste des étapes suivantes :

1. Configuration et exécution de l'interface utilisateur de configuration **Conga** — le serveur **luci**. Reportez-vous à la Section 3.2, « Démarrage de **luci** ».
2. Création d'un cluster. Reportez-vous à la Section 3.4, « Créer un cluster ».
3. Configuration des propriétés globales du cluster. Reportez-vous à la Section 3.5, « Propriétés globales du cluster ».
4. Configuration des périphériques fence. Reportez-vous à la Section 3.6, « Configurer des périphériques fence ».

5. Configuration du fencing pour les membres du cluster. Reportez-vous à la Section 3.7, « Configurer le fencing pour les membres du cluster ».

6. Création de domaines de basculement. Reportez-vous à la Section 3.8, « Configurer un domaine de basculement ».

7. Création des ressources. Reportez-vous à la Section 3.9, « Configurer les ressources globales du cluster ».

8. Création des services du cluster. Reportez-vous à la Section 3.10, « Ajouter un service cluster à un cluster ».

3.2. DÉMARRAGE DE LUCI

NOTE

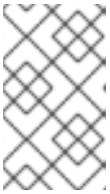
L'utilisation de **luci** pour configurer un cluster requiert que **ricci** soit installé et exécuté sur les nœuds du cluster, comme décrit dans la Section 2.13, « Considérations pour **ricci** ». Comme noté dans cette section, l'utilisation de **ricci** requiert un mot de passe que **luci** vous demande de saisir pour chaque nœud de cluster lorsque vous créez un cluster, comme décrit dans la Section 3.4, « Créer un cluster ».

Avant de lancer **luci**, assurez-vous que les ports IP sur vos nœuds de clusters permettent les connexions au port 11111 depuis le serveur **luci** vers tous les nœuds avec lesquels **luci** communiquera. Pour obtenir des informations sur l'activation des ports IP sur les nœuds de clusters, voir la Section 2.3.1, « Activation des ports IP sur des nœuds de clusters ».

Pour administrer le module complémentaire Red Hat High Availability avec **Conga**, installez et exécutez **luci** comme suit :

1. Sélectionnez un ordinateur pour héberger **luci** et installez le logiciel **luci** sur cet ordinateur. Par exemple :

```
# yum install luci
```



NOTE

Typiquement, un ordinateur dans une cage de serveur ou dans un centre de données héberge **luci** ; cependant, un ordinateur cluster peut aussi héberger **luci**.

2. Démarrez **luci** à l'aide de **service luci start**. Par exemple :

```
# service luci start
Starting luci: generating https SSL certificates...  done
                                                           [  OK
]

Please, point your web browser to https://nano-01:8084 to access
luci
```

NOTE

À partir de Red Hat Enterprise Linux 6.1, vous pouvez configurer certains aspects du comportement de **luci** par le biais du fichier **/etc/sysconfig/luci**, y compris les paramètres du port et de l'hôte, comme le décrit la Section 2.4, « Configurer **luci** avec **/etc/sysconfig/luci** ». Les paramètres du port et de l'hôte seront automatiquement reflétés dans l'URL affiché lorsque le service **luci** est lancé.

3. Sur un navigateur web, placez l'URL du serveur **luci** dans la boîte de l'adresse URL et cliquez sur **Aller à** (ou équivalent). La syntaxe de l'URL du serveur **luci** est **https://*luci_server_hostname*:*luci_server_port***. La valeur par défaut de *luci_server_port* est **8084**.

   La première fois que vous accéderez à **luci**, une invite spécifique au navigateur web concernant le certificat SSL auto-signé (du serveur **luci**) s'affiche. Après confirmation de la boîte (ou des boîtes) de dialogue, votre navigateur web affichera la page de connexion de **luci**.

4. Même si tout utilisateur en mesure de s'authentifier sur le système qui héberge **luci** peut se connecter à **luci**, à partir de la version 6.2 de Red Hat Enterprise Linux, seul l'utilisateur root du système qui exécute **luci** pourra accéder à tous les composants **luci**, jusqu'à ce qu'un administrateur (l'utilisateur root, ou un utilisateur avec des permissions d'administrateur) définisse les permissions pour cet utilisateur. Pour obtenir des informations sur la définition des permissions **luci** pour les utilisateurs, reportez-vous à la Section 3.3, « Contrôler l'accès à luci ».

   Se connecter à **luci** affiche la page **Homebase** (page d'accueil) **luci**, comme indiqué dans la Figure 3.1, « Page Homebase luci ».

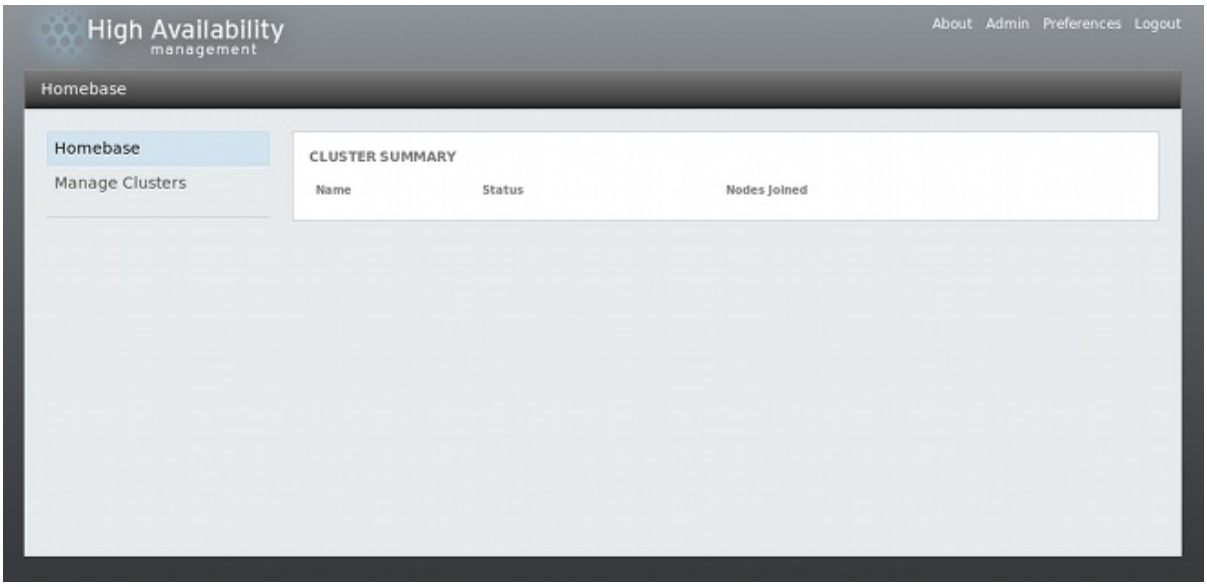


Figure 3.1. Page Homebase luci

NOTE

luci possède délai d'inactivité qui vous déconnecte après 15 minutes d'inactivité.

3.3. CONTRÔLER L'ACCÈS À LUCI

Depuis la sortie initiale de Red Hat Enterprise Linux 6, les fonctionnalités suivantes ont été ajoutées à la page **Utilisateurs et permissions**.

- À partir de Red Hat Enterprise Linux 6.2, l'utilisateur root ou un utilisateur qui s'est vu offrir des permissions d'administrateur **luci** sur un système exécutant **luci** peut contrôler l'accès aux divers composants **luci** en paramétrant des permissions pour les utilisateurs individuels sur un système.
- À partir de Red Hat Enterprise Linux 6.3, l'utilisateur root ou un utilisateur qui s'est vu offrir des permissions d'administrateur **luci** peut aussi utiliser l'interface **luci** pour ajouter des utilisateurs au système.
- À partir de Red Hat Enterprise Linux 6.4, l'utilisateur root ou un utilisateur qui s'est vu offrir des permissions d'administrateur **luci** peut aussi utiliser l'interface **luci** pour supprimer des utilisateurs du système.

Pour ajouter des utilisateurs, pour les supprimer, ou pour paramétrer leurs permissions, connectez-vous à **luci** en tant que **root** ou en tant qu'utilisateur qui possède déjà des permissions d'administrateur et cliquez sur la sélection **Admin** dans le coin en haut à droite de l'écran **luci**. Ceci ouvre la page **Utilisateurs et permissions**, qui affiche les utilisateurs existants.

Pour supprimer des utilisateurs, sélectionnez le (ou les) utilisateur(s), puis cliquez sur **Supprimer la sélection**.

Pour ajouter un utilisateur, cliquez sur **Ajouter un utilisateur** puis saisissez le nom de l'utilisateur à ajouter.

Pour définir ou modifier les permissions d'un utilisateur, sélectionnez l'utilisateur dans le menu déroulant sous **Permissions de l'utilisateur**. Ceci vous permet de définir les permissions suivantes :

Administrateur Luci

Offre à l'utilisateur les mêmes permissions que l'utilisateur root, avec des permissions complètes sur tous les clusters et la possibilité de définir ou supprimer des permissions pour tous les utilisateurs, à l'exception de l'utilisateur root, dont les permissions ne peuvent pas être restreintes.

Créer des clusters

Permet à l'utilisateur de créer des clusters, comme décrit dans la Section 3.4, « Créer un cluster ».

Importer des clusters existants

Permet à l'utilisateur d'ajouter un cluster existant à l'interface **luci**, comme le décrit la Section 4.1, « Ajouter un cluster existante à l'interface luci ».

Pour chaque cluster qui a été créé ou importé sur **luci**, les permissions suivantes peuvent être définies pour l'utilisateur indiqué :

Voir ce cluster

Autorise l'utilisateur à voir le cluster spécifié.

Changer la configuration du cluster

Permet à l'utilisateur de modifier la configuration du cluster spécifié, à l'exception de l'ajout et de la suppression de nœuds du cluster.

`Activer, désactiver, déplacer et migrer des groupes de services`

Permet à l'utilisateur de gérer les services de haute disponibilité (« High Availability »), comme le décrit la Section 4.5, « Gérer les services High-Availability ».

`Arrêter, démarrer et redémarrer des nœuds de cluster`

Permet à l'utilisateur de gérer les nœuds individuels d'un cluster, comme le décrit la Section 4.3, « Gérer les nœuds de clusters ».

`Ajouter et supprimer des nœuds`

Permet à l'utilisateur d'ajouter et de supprimer des nœuds d'un cluster, comme le décrit la Section 3.4, « Créer un cluster ».

`Supprimer ce cluster de Luci`

Permet à l'utilisateur de supprimer un cluster de l'interface **luci**, comme le décrit la Section 4.4, « Démarrer, arrêter, redémarrer et supprimer des clusters ».

Cliquez sur **`Soumettre`** pour que les permissions prennent effet, ou sur **`Réinitialiser`** pour retourner aux valeurs initiales.

3.4. CRÉER UN CLUSTER

La création d'un cluster avec **luci** consiste en la dénomination d'un cluster, l'ajout de nœuds de cluster au cluster, la saisie d'un mot de passe **ricci** pour chaque nœud et la soumission d'une requête pour créer un cluster. Si les informations et les mots de passe des nœuds sont corrects, **Conga** installera automatiquement un logiciel dans les nœuds du cluster (si les paquetages logiciels appropriés ne sont pas déjà installés) et démarre le cluster. Créez le cluster comme suit :

1. Cliquez sur **Gérer les clusters** dans le menu sur le côté gauche de la page **Homebase** de **luci**. L'écran **Clusters** apparaît, comme décrit dans la Figure 3.2, « page de gestion de cluster luci ».

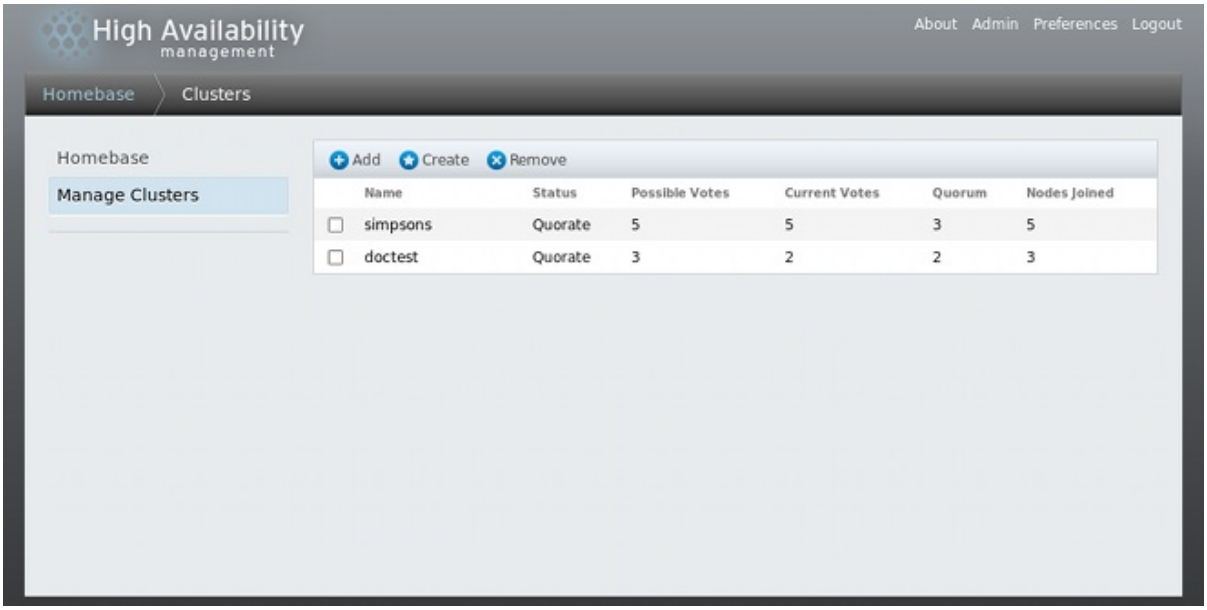


Figure 3.2. page de gestion de cluster luci

2. Cliquez sur **Créer**. La boîte de dialogue **Créer un nouveau cluster**, comme décrit dans la Figure 3.3, « boîte de dialogue luci de création de cluster ».

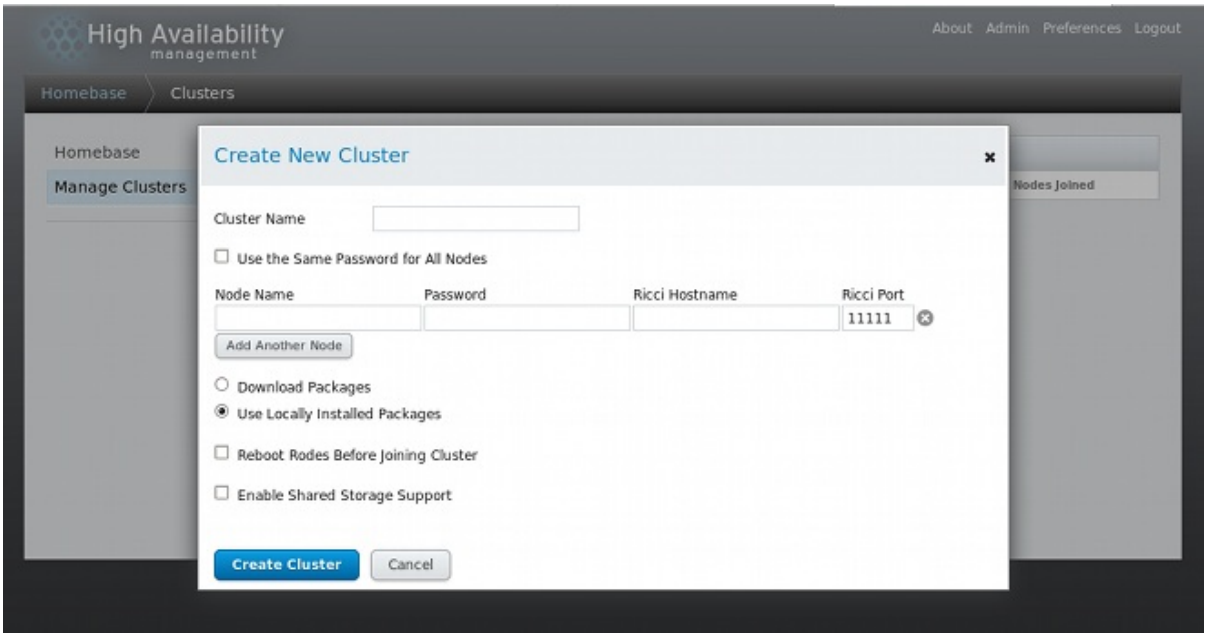


Figure 3.3. boîte de dialogue luci de création de cluster

3. Saisissez les paramètres suivants dans la boîte de dialogue **Créer un nouveau cluster** comme nécessaire :

   - Dans la boîte de texte **Nom du cluster**, saisissez un nom de cluster. Le nom du cluster ne doit pas excéder 15 caractères.

   - Si chaque nœud du cluster possède le même mot de passe **ricci**, vous pouvez cocher **Use the same password for all nodes** (utiliser le même mot de passe pour tous les nœuds) afin de remplir automatiquement le champ **password** (mot de passe) lorsque vous ajoutez des nœuds.

   - Saisissez le nom du nœud pour un nœud dans le cluster dans la colonne **Node Name** (nom du nœud) puis saisissez le mot de passe **ricci** du nœud dans la colonne **Password** (mot de passe).

   - Si votre système est configuré avec un réseau privé dédié au trafic des clusters, vous devriez configurer **luci** de manière à communiquer avec **ricci** sur une différente adresse de celle résolue par le nom du nœud du cluster. Ceci peut être accompli en saisissant cette adresse comme le **Ricci Hostname** (nom d'hôte Ricci).

   - Si vous utilisez un autre port pour l'agent **ricci** que le port par défaut 11111, vous pouvez modifier ce paramètre.

   - Cliquez sur **Add Another Node** (ajouter un autre nœud) puis saisissez le nom du nœud et le mot de passe **ricci** pour chaque nœud supplémentaire du cluster.

   - Si vous ne souhaitez pas mettre à niveau les paquetages logiciels déjà installés sur les nœuds lorsque vous créez le cluster, laissez l'option **Use locally installed packages** (Utiliser les paquetages installés localement) sélectionnée. Si vous ne souhaitez pas mettre à niveau tous les paquetages logiciels du cluster, sélectionnez l'option **Download Packages** (Télécharger les paquetages).

NOTE

Que vous sélectionniez l'option **Use locally installed packages** (Utiliser les paquetages installés localement) ou l'option **Download Packages** (Télécharger les paquetages), si des composants de base du cluster ne sont pas présents (**`cman`**, **`rgmanager`**, **`modcluster`** et leurs dépendances), il seront installés. S'ils ne peuvent pas être installés, la création du nœud échouera.

- Sélectionnez **Reboot nodes before joining cluster** (Redémarrer les nœuds avant de joindre le cluster) si nécessaire.
- Sélectionnez **Enable shared storage support** (activer le support du stockage partagé) si un stockage clusterisé est requis ; ceci télécharge les paquetages qui prennent en charge le stockage clusterisé et active LVM sur les clusters. Vous devriez sélectionner ceci uniquement lorsque vous avez accès au module complémentaire Resilient Storage (Stockage résilient) ou au module complémentaire Scalable File System (Système de fichiers scalable).

4. Cliquez sur **`Créer un cluster`**. Cliquer sur **`Créer un cluster`** provoque les actions suivantes :

   1. Si vous avez sélectionné **Download Packages** (Télécharger les paquetages), les paquetages logiciels du cluster sont téléchargés sur les nœuds.
   2. Les logiciels du cluster sont installés sur les nœuds (sinon, il est vérifié que les bons paquetages logiciels sont installés).
   3. Le fichier de configuration du cluster est mis à jour et propagé sur chaque nœud dans le cluster.
   4. Les nœuds ajoutés rejoignent le cluster.

   Un message indiquant que le cluster est en train d'être créé est affiché. Lorsque le cluster est prêt, l'écran affiche l'état du cluster nouvellement créé, comme indiqué dans la Figure 3.4, « Affichage du nœud du cluster ». Remarquez que si **ricci** n'est pas en cours d'exécution sur l'un des nœuds, la création du cluster échouera.

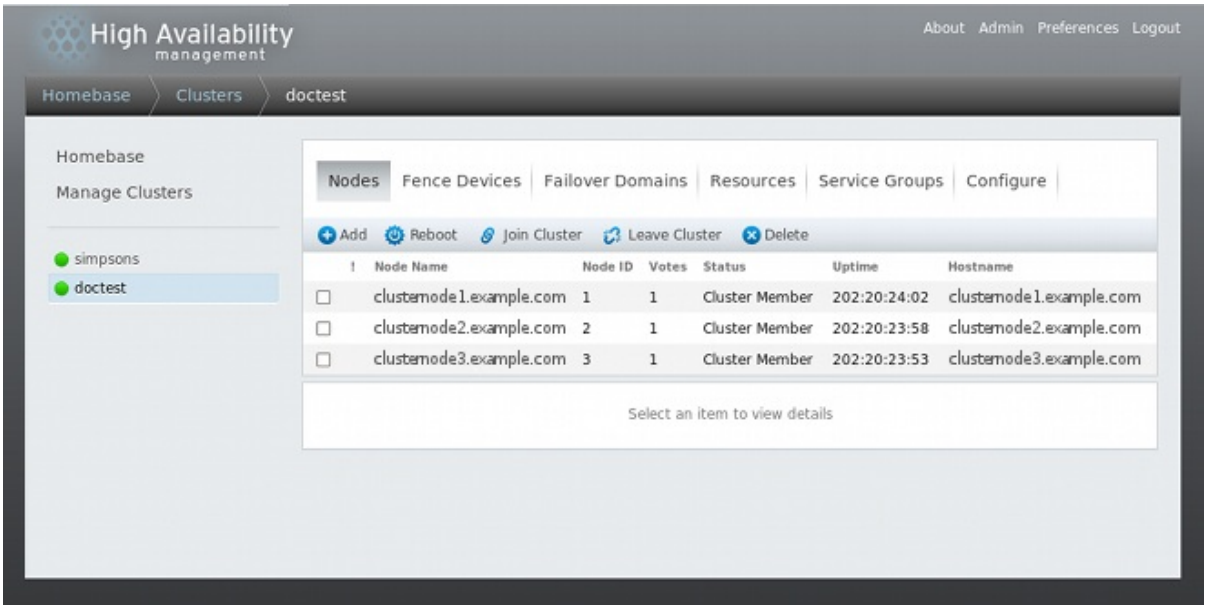


Figure 3.4. Affichage du nœud du cluster

5. Après avoir cliqué sur **`Créer un cluster`** pour créer le cluster, vous pouvez ajouter ou supprimer des nœuds du cluster en cliquant sur la fonction **Ajouter** ou **Supprimer** sur le menu en haut de la page d'affichage des nœuds du cluster. À moins que vous ne tentiez de supprimer un cluster entier, les nœuds doivent être arrêtés avant d'être supprimés. Pour obtenir des informations sur la suppression d'un nœud d'un cluster existant qui est actuellement en cours de fonctionnement, voir Section 4.3.4, « Supprimer un membre d'un cluster ».

NOTE

La suppression d'un nœud du cluster est une opération destructive qui est irréversible.

3.5. PROPRIÉTÉS GLOBALES DU CLUSTER

Lorsque vous sélectionnez un cluster à configurer, une page spécifique aux clusters s'affiche. La page fournit une interface pour configurer les propriétés de la totalité du cluster. Vous pouvez configurer ces propriétés en cliquant sur **Configurer** en haut de l'affichage du cluster. Ceci affiche une interface contenant les onglets suivants : **Général**, **Démon Fence**, **Réseau**, **Anneau redondant**, **QDisk** et **Connexion**. Pour configurer les paramètres de ces onglets, procédez aux étapes décrites dans les sections suivantes. Si vous n'éprouvez pas le besoin de configurer les paramètres dans un onglet, ignorez la section de cet onglet.

3.5.1. Configurer les propriétés générales

Cliquer sur l'onglet **General** (Général) affiche la page **General Properties** (Propriétés générales), qui fournit une interface pour modifier la version de la configuration.

- La boîte de texte **Cluster Name** (Nom du cluster) affiche le nom du cluster ; elle n'accepte pas de modification du nom du cluster. La seule manière de changer le nom d'un cluster est de créer une nouvelle configuration de cluster avec le nouveau nom.

- La valeur de la **Version de la configuration** est définie sur **`1`** au moment de la création du cluster et est automatiquement incrémentée à chaque fois que vous modifiez la configuration de votre cluster. Cependant, si vous devez changer la valeur, vous pouvez la spécifier dans la boîte de texte **Version de la configuration**.

Si vous avez modifié la valeur de la **Version de la configuration**, cliquez sur **Appliquer** pour que la modification prenne effet.

3.5.2. Configurer les propriétés du démon fence

Cliquer sur l'onglet **Fence Daemon** (Démon fence) affiche la page **Fence Daemon Properties** (Propriétés du démon fence), qui fournit une interface pour configurer **Post Fail Delay** et **Post Join Delay**. Les valeurs que vous devrez configurer pour ces paramètres sont des propriétés générales de fencing pour le cluster. Pour configurer des périphériques fence spécifiques pour les nœuds du cluster, utilisez l'élément du menu **Fence Devices** (Périphériques fence) de l'affichage du cluster, comme décrit dans la Section 3.6, « Configurer des périphériques fence ».

- Le paramètre **Post Fail Delay** est le nombre de secondes que le démon fence (**fenced**) doit attendre avant de clôturer un nœud (un membre du domaine fence) une fois que le nœud a échoué. La valeur par défaut de **Post Fail Delay** est **0**. Sa valeur peut être modifiée pour mieux correspondre à performance du cluster et du réseau.

- Le paramètre **Post Join Delay** correspond au nombre de secondes que le démon Fence (**fenced**) attend avant de clôturer un nœud après que le nœud a rejoint le domaine fence. La valeur par défaut du délai **Post Join Delay** est **6**. Typiquement, le paramètre de délai **Post Join Delay** se situe entre 20 et 30 seconds, mais celui-ci peut varier en fonction de la performance du cluster et du réseau.

Saisissez les valeurs requises et cliquez sur **Appliquer** pour que les modifications prennent effet.

NOTE

Pour obtenir plus d'informations sur **Post Join Delay** et **Post Fail Delay**, reportez-vous à la page man fenced(8).

3.5.3. Configuration du réseau

Cliquer sur l'onglet **Network** (Réseau) affiche la page **Network Configuration** (Configuration du réseau), qui fournit une interface pour configurer le type de transport réseau.

Vous pouvez utiliser cet onglet pour sélectionner l'une des options suivantes :

- **Multidiffusion UDP et laisser le cluster choisir l'adresse de multidiffusion**

  Ceci est le paramètre par défaut. Avec cette option sélectionnée, le module complémentaire Red Hat High Availability crée une adresse de multidiffusion basée sur l'ID du cluster. Il génère les 16 bits les plus bas de l'adresse et les ajoute à la portion supérieure de l'adresse selon que le protocole IP est IPv4 ou IPv6 :

  - Pour IPv4 — L'adresse formée est 239.192. plus les 16 bits les plus bas générés par le logiciel du module complémentaire Red Hat High Availability.

  - Pour IPv6 — L'adresse formée est FF15:: plus les 16 bits les plus bas générés par le logiciel du module complémentaire Red Hat High Availability.

NOTE

L'ID du cluster est un identifiant unique que **cman** génère pour chaque cluster. Pour voir l'ID du cluster, exécutez la commande **cman_tool status** sur un nœud de cluster.

- **Multidiffusion UDP et spécifier l'adresse de multidiffusion manuellement**

  Si vous devez utiliser une adresse de multidiffusion spécifique, sélectionnez cette option et saisissez une adresse de multidiffusion dans la boîte de texte **Adresse de multidiffusion**.

  Si vous spécifiez une adresse de multidiffusion, vous devriez utiliser les séries 239.192.x.x (ou FF15:: pour IPv6) que **cman** utilise. L'utilisation d'une adresse de multidiffusion hors de cette plage peut causer des résultats imprévisibles. Par exemple, l'utilisation de 224.0.0.x, qui est "All hosts on the network" (Tous les hôtes sur le réseau) pourrait ne pas être routé correctement, ou même ne pas être routé du tout par le matériel.

  Si vous spécifiez ou modifiez une adresse de multidiffusion, vous devrez redémarrer le cluster pour que celle-ci prenne effet. Pour obtenir des informations sur le démarrage et l'arrêt d'un cluster avec **Conga**, reportez-vous à la Section 4.4, « Démarrer, arrêter, redémarrer et supprimer des clusters ».

  NOTE

  Si vous spécifiez une adresse de multidiffusion, assurez-vous de bien vérifier la configuration des routeurs par lesquels les paquets des clusters passent. Certains routeurs prennent longtemps pour apprendre les adresses, affectant ainsi sévèrement la performance du cluster.

- **Monodiffusion UDP (UDPU)**

  À partir de Red Hat Enterprise Linux 6.2, les nœuds dans une grappe peuvent communiquer entre eux à l'aide du mécanisme de transport de monodiffusion UDP. Il est recommandé d'utiliser la multidiffusion IP pour le réseau de grappes. La monodiffusion UDP est une alternative pouvant être utilisée lorsque la multidiffusion IP est indisponible. La monodiffusion UDP n'est pas recommandée pour les déploiements GFS2.

Cliquez sur **Appliquer**. Lors de la modification du type de transport, un redémarrage du cluster est nécessaire pour que les changements prennent effet.

3.5.4. Configurer le protocole d'anneau redondant (« Redundant Ring »)

À partir de Red Hat Enterprise Linux 6.4, le module complémentaire Red Hat High Availability prend en charge la configuration du protocole d'anneau redondant. Lors de l'utilisation du protocole d'anneau redondant, un certain nombre de considérations sont à prendre en compte, comme le décrit la Section 7.6, « Configurer le protocole d'anneau redondant (« Redundant Ring ») ».

Cliquer sur l'onglet **Anneau redondant** affiche la page **Configuration du protocole d'anneau redondant**. Cette page affiche tous les nœuds actuellement configurés pour le cluster. Si vous configurez un système pour qu'il utilise le protocole d'anneau redondant, vous devrez spécifier le **Nom alterne** de chaque nœud pour le second anneau.

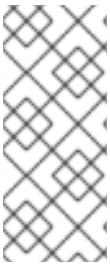
La page **Configuration du protocole d'anneau redondant** vous permet de spécifier optionnellement l'**Adresse de multidiffusion de l'anneau alterne**, le **Port CMAN de l'anneau alterne** et le **TTL de**

paquet de multidiffusion de l'anneau alterne (de l'anglais, « Alternate Ring Multicast Packet TTL ») du second anneau.

Si vous spécifiez une adresse de multidiffusion pour le deuxième anneau, l'adresse de multidiffusion alterne ou le port alterne doit être différent de l'adresse de multidiffusion du premier anneau. Si vous spécifiez un port alterne, les numéros des ports du premier et du second anneau doivent être différents d'au moins deux car le système utilise port et port-1pour effectuer des opérations. Si vous ne spécifiez pas d'adresse de multidiffusion alterne, le système utilisera automatiquement une adresse de multidiffusion différente pour le second anneau.

3.5.5. Configuration du disque quorum

Cliquer sur l'onglet **QDisk**affiche la page **Configuration du disque quorum**, qui fournit une interface pour la configuration des paramètres du disque quorum si vous devez utiliser un disque quorum.



NOTE

Les paramètres et heuristiques du disque quorum dépendent de l'environnement du site et des pré-requis spéciaux nécessaires. Pour comprendre l'utilisation des paramètres et heuristiques du disque quorum, reportez-vous à la page man qdisk(5). Si vous avez besoin d'aide pour la compréhension et l'utilisation du disque quorum, veuillez contacter un représentant autorisé du support Red Hat.

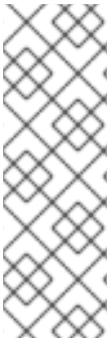
Le paramètre **Ne pas utiliser un disque quorum** est activé par défaut. Si vous devez utiliser un disque quorum, cliquez sur **Utiliser un disque quorum**, saisissez les paramètres du disque quorum, cliquez sur **`Appliquer`**, puis redémarrez le cluster pour que les changements prennent effet.

Le Tableau 3.1, « Paramètres du disque quorum » décrit les paramètres du disque quorum.

Tableau 3.1. Paramètres du disque quorum

Paramètre	Description
Spécifier un périphérique physique : par étiquette de périphérique	Spécifie l'étiquette du disque quorum créée par l'utilitaire **`mkqdisk`**. Si ce champ est utilisé, le démon quorum lit le fichier **`/proc/partitions`** et vérifiera les signatures qdisk sur chaque périphérique bloc trouvé, en comparant l'étiquette avec l'étiquette spécifiée. Ceci est utile dans des configurations où le nom du périphérique quorum diffère selon les nœuds.
Heuristiques	**Path to Program** — Programme utilisé pour déterminer si cette heuristique est disponible. Ceci peut être n'importe quoi qui est exécutable par **`/bin/sh -c`**. Une valeur retournée de 0 indique un succès ; toute autre chose indique un échec. Ce champ est requis. **Interval** — Fréquence (en secondes) à laquelle l'heuristique est analysée. L'intervalle par défaut pour chaque heuristique est de 2 secondes. **Score** — Poids de l'heuristique. Soyez prudent lors de la détermination des scores des heuristiques. Le score par défaut pour chaque heuristique est 1. **TKO** — Nombre d'échecs consécutifs requis avant que cette heuristique ne soit déclarée indisponible.

Paramètre	Description
Score total minimum	Score minimum qu'un nœud doit effectuer pour être considéré comme « vivant ». Si oublié, ou si ajusté sur 0, **`floor((n+1)/2)`**, est utilisé, où *n* est la somme des scores heuristiques. La valeur **Score total minimum** ne doit jamais excéder la somme des scores heuristiques, sinon le disque quorum ne pourra pas être disponible.



NOTE

Cliquer sur **Appliquer** dans l'onglet **Configuration QDisk** propage les modifications apportées au fichier de configuration (**`/etc/cluster/cluster.conf`**) dans chaque nœud du cluster. Cependant, pour que le disque quorum puisse opérer ou pour que toute modification apportée aux paramètres du disque quorum puisse prendre effet, vous devez redémarrer le cluster (reportez-vous à la Section 4.4, « Démarrer, arrêter, redémarrer et supprimer des clusters ») et vous assurer que le démon **`qdiskd`** est redémarré sur chaque nœud.

3.5.6. Configuration de la journalisation

Cliquer sur l'onglet **Logging** (Journalisation) affiche la page **Logging Configuration** (Configuration de la journalisation), qui fournit une interface pour la configuration des paramètres de journalisation.

Vous pouvez configurer les paramètres suivants pour la configuration globale de la journalisation :

- Cocher **Log debugging messages** (Journaliser les messages de débogage) active les messages de débogage dans le fichier de journalisation.
- Cocher **Log messages to syslog** (Journaliser les messages sur syslog) active les messages sur **`syslog`**. Vous pouvez sélectionner **Syslog message facility** et **Syslog message priority**. Le paramètre **Syslog message priority** indique que les messages au niveau sélectionné et aux niveaux supérieurs sont envoyés sur **`syslog`**.
- Cocher **Log messages to log file** (Journaliser les messages sur le fichier de journalisation) active les messages sur le fichier de journalisation. Vous pouvez spécifier le nom de **Log File Path** (chemin d'accès du fichier de journalisation). Le paramètre **logfile message priority**

indique que les messages au niveau sélectionné et aux niveaux supérieurs sont écrits sur le fichier de journalisation.

Vous pouvez remplacer les paramètres globaux de journalisation pour des démons spécifiques en sélectionnant l'un des démons au bas de l'en-tête **Remplacement de la journalisation spécifique au démon** de la page **Configuration de la journalisation**. Après avoir sélectionné le démon, vous pouvez vérifier si vous souhaitez journaliser les messages de débogage pour ce démon en particulier. Vous pouvez aussi spécifier **`syslog`** et les paramètres du fichier de journalisation de ce démon.

Cliquez sur **`Appliquer`** pour que les modifications de la configuration de la journalisation que vous avez spécifiées prennent effet.

3.6. CONFIGURER DES PÉRIPHÉRIQUES FENCE

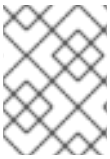
La configuration des périphériques fence consiste en la création, la mise à jour, et la suppression des périphériques fence pour le cluster. Vous devez configurer les périphériques fence dans un cluster avant de pouvoir configurer le fencing pour les nœuds dans le cluster.

La création d'un périphérique fence consiste de la sélection d'un type de périphérique fence et de la saisie de paramètres pour celui-ci (par exemple, le nom, l'adresse IP, l'identifiant et le mot de passe). La mise à jour d'un périphérique fence se compose de la sélection d'un périphérique fence existant et de la modification de ses paramètres. La suppression d'un périphérique fence est composée de la sélection d'un périphérique fence et de sa suppression.

Cette section fournit des procédures pour les tâches suivantes :

- Creating fence devices — Reportez-vous à la Section 3.6.1, « Créer un périphérique fence ». Une fois que vous avez créé et nommé un périphérique fence, vous pourrez configurer les périphériques fence pour chaque nœud dans le cluster, comme décrit dans la Section 3.7, « Configurer le fencing pour les membres du cluster ».
- Mise à jour des périphériques fence — Reportez-vous à la Section 3.6.2, « Modifier un périphérique fence ».
- Suppression de périphériques fence — Reportez-vous à la Section 3.6.3, « Supprimer un périphérique fence ».

À partir de la page spécifique aux clusters, vous pouvez configurer des périphériques fence pour ce cluster en cliquant sur **Périphériques fence** en haut de l'affichage du cluster. Ceci affiche les périphériques fence du cluster et les éléments du menu de la configuration de périphériques fence : **Ajouter** et **Supprimer**. Ceci est le point de départ de chaque procédure décrite dans les sections suivantes.



NOTE

S'il s'agit d'une configuration initiale du cluster, aucun périphérique fence n'a été créé, ce qui explique pourquoi aucun d'entre eux n'est affiché.

La Figure 3.5, « Page luci de la configuration des périphériques fence » montre l'écran de configuration des périphériques fence avant que tout périphérique fence ne soit créé.

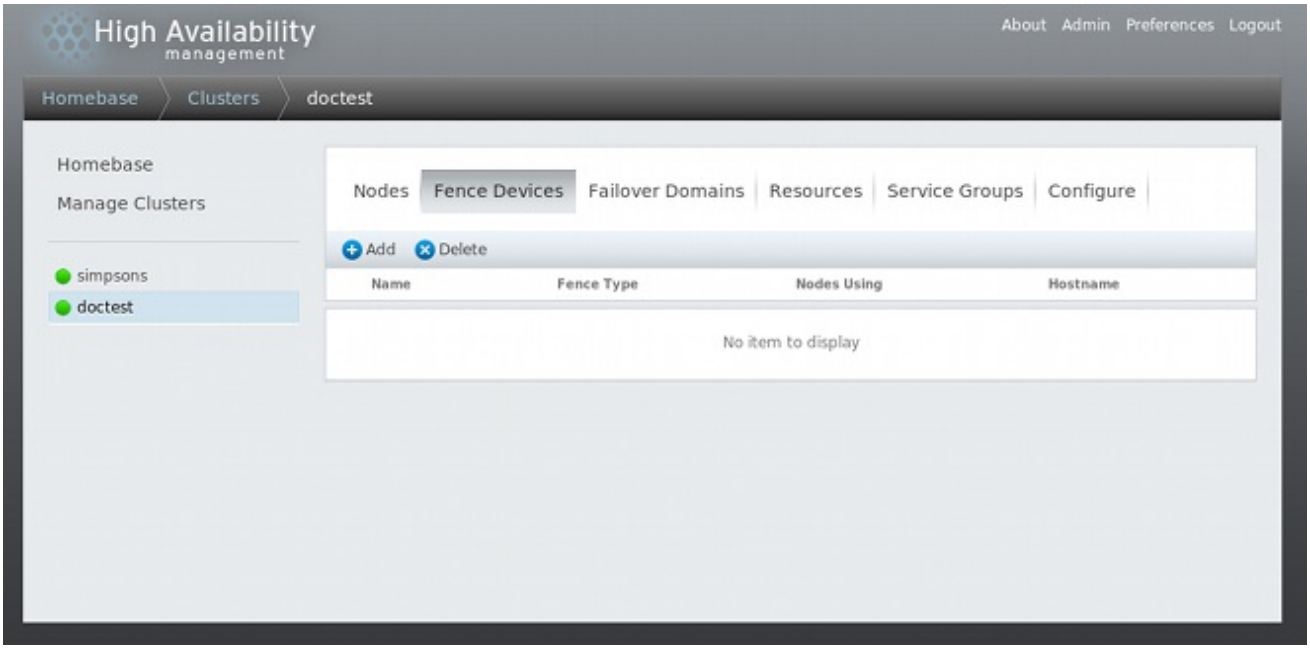


Figure 3.5. Page luci de la configuration des périphériques fence

3.6.1. Créer un périphérique fence

Pour créer un périphérique fence, suivez ces étapes :

1. À partir de la page de configuration **Périphériques fence**, cliquez sur **Ajouter**. Cliquer sur **Ajouter** affiche la boîte de dialogue **`Ajouter un périphérique fence (instance)`** . À partir de cette boîte de dialogue, sélectionnez le type de périphérique fence à configurer.

2. Spécifiez les informations dans la boîte de dialogue **`Ajouter un périphérique fence (instance)`** en fonction du type de périphérique fence. Reportez-vous à l'Annexe A, *Paramètres des périphériques fence* afin d'obtenir plus d'informations sur les paramètres des périphériques fence. Dans certains cas, vous devrez spécifier des paramètres spécifiques au nœud pour le périphérique fence lorsque vous configurer le fencing pour des nœuds individuels, comme décrit dans la Section 3.7, « Configurer le fencing pour les membres du cluster ».

3. Cliquez sur **`Submit`** (Soumettre).

Une fois que le périphérique fence a été ajouté, il apparaît sur la page de configuration **Périphériques fence**.

3.6.2. Modifier un périphérique fence

Pour modifier un périphérique fence, suivez ces étapes :

1. À partir de la page **Périphériques fence**, cliquez sur le nom du périphérique fence à modifier. Ceci affiche la boîte de dialogue de ce périphérique fence, avec les valeurs qui ont été configurées pour ce périphérique.

2. Pour modifier le périphérique fence, saisissez les modifications aux paramètres affichés. Reportez-vous à l'Annexe A, *Paramètres des périphériques fence* pour obtenir plus d'informations.

3. Cliquez sur **`Appliquer`** et attendez que la configuration soir mise à jour.

3.6.3. Supprimer un périphérique fence

NOTE

Les périphériques fence qui sont en cours d'utilisation ne peuvent pas être supprimés. Pour supprimer un périphérique fence qu'un nœud est en train d'utiliser, mettez tout d'abord la configuration fence du nœud à jour pour tout nœud utilisant le périphérique puis supprimez le périphérique.

Pour supprimer un périphérique fence, suivez ces étapes :

1. À partir de la page de configuration **Périphériques fence**, cochez la case à gauche du (ou des) périphérique(s) fence afin de sélectionner les périphériques à supprimer.

2. Cliquez sur **`Supprimer`** et attendez que la configuration soit mise à jour. Un message apparaît indiquant quels périphériques sont en cours de suppression.

Lorsque la configuration a été mise à jour, le périphérique fence supprimé n'apparaît plus dans l'affichage.

3.7. CONFIGURER LE FENCING POUR LES MEMBRES DU CLUSTER

Une fois les étapes initiales de création du cluster et des périphériques fence terminées, vous devrez configurer le fencing pour les nœuds du cluster. Pour configurer le fencing pour les nœuds après la création d'un nouveau cluster et la configuration des périphériques fence du cluster, suivez les étapes de cette section. Remarquez que vous devez configurer le fencing pour chaque nœud du cluster.

Les sections suivantes proposent des procédures pour la configuration d'un périphérique fence unique pour un nœud, pour la configuration d'un nœud avec un périphérique fence de sauvegarde et pour la configuration d'un nœud avec un système d'alimentation redondant :

- Section 3.7.1, « Configurer un périphérique fence unique pour un nœud »
- Section 3.7.2, « Configurer un périphérique fence de sauvegarde »
- Section 3.7.3, « Configurer un nœud avec une alimentation redondante »

3.7.1. Configurer un périphérique fence unique pour un nœud

Utilisez la procédure suivante pour configurer un nœud avec un périphérique fence unique.

1. À partir de la page spécifique aux clusters, vous pouvez configurer le fencing pour les nœuds dans le cluster en cliquant sur **nœuds** en haut de l'affichage du cluster. Ceci affiche les nœuds constituant le cluster. Cette page est aussi la page par défaut apparaissant lorsque vous cliquez sur le nom du cluster sous **Gérer les clusters** dans le menu, sur le côté gauche de la page **Homebase luci**.

2. Cliquez sur un nom de nœud. Cliquer sur un lien vers un nœud fait qu'une page pour ce lien sera affichée, montrant comment ce nœud est configuré.

   La page spécifique aux nœuds affiche tous les services actuellement en cours d'exécution sur le nœud, ainsi que tous les domaines de basculement dont ce nœud est un membre. Vous pouvez modifier un domaine de basculement existant en cliquant sur son nom. Pour obtenir des informations sur la configuration des domaines de basculement, voir la Section 3.8, « Configurer un domaine de basculement ».

3. Sur la page spécifique aux nœuds, sous **Périphériques fence**, cliquez sur **`Ajouter une méthode fence`**. Ceci affiche la boîte de dialogue **`Ajouter une méthode fence au nœud`**.

4. Saisissez un **Nom de méthode** pour la méthode de fencing que vous configurez pour ce nœud. Ceci est un nom arbitraire qui sera utilisé par le module complémentaire Red Hat High Availability ; il ne s'agit pas de la même chose que le nom DNS du périphérique.

5. Cliquez sur **`Soumettre`**. Cela ouvre l'écran spécifique aux nœuds qui affiche la méthode que vous venez d'ajouter sous **Périphériques fence**.

6. Configurez une instance fence pour cette méthode en cliquant sur le bouton **`Ajouter une instance fence`** qui apparaît sous la méthode fence. Ceci affiche un menu déroulant **Ajouter un périphérique fence (Instance)** à partir duquel vous pouvez sélectionner un périphérique fence que vous avez précédemment configuré, comme décrit dans la Section 3.6.1, « Créer un périphérique fence ».

7. Sélectionnez un périphérique fence pour cette méthode. Si ce périphérique fence requiert que vous configuriez des paramètres spécifiques au nœud, l'affichage montrera les paramètres à configurer. Pour obtenir des informations sur les paramètres du fencing, reportez-vous à l'Annexe A, *Paramètres des périphériques fence*.

NOTE

Pour les méthodes fence qui ne sont pas basées sur l'alimentation (comme le fencing SAN ou de stockage), **Unfencing** est sélectionné par défaut sur l'affichage des paramètres spécifiques au(x) nœud(s). Ceci assure que l'accès d'un nœud clôturé (fenced) au stockage n'est pas ré-activé jusqu'à ce que le nœud ne soit redémarré. Pour obtenir des informations sur l'unfencing d'un nœud, reportez-vous à la page man **`fence_node`**(8).

8. Cliquez sur **`Soumettre`**. Cela vous ramène à l'écran spécifique aux nœuds avec la méthode et l'instance fence affichées.

3.7.2. Configurer un périphérique fence de sauvegarde

Vous pouvez définir de multiples méthodes fence pour un nœud. Si le fencing échoue avec la première méthode, le système tentera de clôturer le nœud à l'aide de la seconde méthode, puis par toute autre méthode que vous aurez configurée.

Utilisez la procédure suivante pour configurer un périphérique fence de sauvegarde pour un nœud.

1. Utilisez la procédure fournie dans la Section 3.7.1, « Configurer un périphérique fence unique pour un nœud » pour configurer la méthode de fencing primaire pour un nœud.

2. Sous l'affichage de la méthode primaire que vous avez défini, cliquez sur **`Ajouter une méthode fence`**.

3. Saisissez un nom pour la méthode de fencing de sauvegarde que vous avez configuré pour ce nœud et cliquez sur **`Soumettre`**. Ceci affiche l'écran spécifique aux nœuds, qui montre la méthode que vous avez ajoutée en dessous de la méthode fence primaire.

4. Configurez une instance fence pour cette méthode en cliquant sur **`Ajouter une instance fence`**. Ceci affiche un menu déroulant à partir duquel vous pouvez sélectionner un périphérique fence que vous aurez précédemment configuré, comme décrit dans la Section 3.6.1, « Créer un

périphérique fence ».

5. Sélectionnez un périphérique fence pour cette méthode. Si ce périphérique fence requiert que vous configuriez des paramètres spécifiques au nœud, l'affichage montrera les paramètres à configurer. Pour obtenir des informations sur les paramètres du fencing, reportez-vous à l'Annexe A, *Paramètres des périphériques fence*.

6. Cliquez sur **Soumettre**. Cela vous ramène à l'écran spécifique aux nœuds avec la méthode et l'instance fence affichées.

Vous pouvez continuer à ajouter des méthodes de fencing comme nécessaire. Vous pouvez réarranger l'ordre des méthodes fence qui seront utilisées pour ce nœud en cliquant sur **Move Up** (Haut) et **Move Down** (Bas).

3.7.3. Configurer un nœud avec une alimentation redondante

Si votre cluster est configuré avec une alimentation redondante pour vos nœuds, vous devez vous assurer de configurer le fencing de manière à ce que vos nœuds s'éteignent complètement lorsqu'ils ont besoin d'être clôturés (fenced). Si vous configurez chaque source d'alimentation en tant que méthode fence séparée, chaque alimentation devra être clôturée (fenced) séparément ; la seconde alimentation permettra au système de continuer à s'exécuter lorsque la première est fenced et le système ne sera pas fenced. Pour configurer un système avec un système d'alimentation duel, vous devrez configurer vos périphériques fence de manière à ce que les deux sources d'alimentation soient éteintes et que le système soit totalement éteint. Lors de la configuration de votre système à l'aide de **Conga**, il vous faudra configurer deux instances dans une seule méthode de fencing.

Pour configurer le fencing pour un nœud à système d'alimentation électrique duel, suivez les étapes de cette section.

1. Avant de pouvoir configurer le fencing pour un nœud avec une alimentation redondante, vous devez configurer chaque interrupteur d'alimentation en tant que périphérique fence pour le cluster. Pour obtenir des informations sur la configuration des périphériques fence, voir la Section 3.6, « Configurer des périphériques fence ».

2. À partir de la page spécifique aux clusters, cliquez sur **nœuds** en haut de l'affichage des clusters. Ceci affiche les nœuds constituant le cluster. Ceci est aussi la page par défaut apparaissant lorsque vous cliquez sur le nom du cluster en dessous de **Gérer les clusters** dans le menu sur le côté gauche de la page **Homebase luci**.

3. Cliquez sur un nom de nœud. Cliquer sur un lien vers un nœud fait qu'une page pour ce lien sera affichée, montrant comment ce nœud est configuré.

4. Sur la page spécifique aux nœuds, cliquez sur **Ajouter une méthode fence**.

5. Saisissez un nom pour la méthode de fencing que vous configurez pour ce nœud.

6. Cliquez sur **Soumettre**. Cela ouvre l'écran spécifique aux nœuds qui affiche la méthode que vous venez d'ajouter sous **Périphériques fence**.

7. Configurez la première source d'alimentation en tant qu'instance fence pour cette méthode en cliquant sur **Ajouter une instance fence**. Ceci affiche un menu déroulant à partir duquel vous pouvez sélectionner un des périphériques de fencing alimenté que vous aurez précédemment configuré, comme décrit dans la Section 3.6.1, « Créer un périphérique fence ».

8. Sélectionnez l'un des périphériques de fencing alimenté pour cette méthode et saisissez les paramètres appropriés pour celui-ci.

9. Cliquez sur **Soumettre**. Cela vous ramène à l'écran spécifique aux nœuds avec la méthode et l'instance fence affichées.

10. Sous la même méthode fence pour laquelle vous avez configuré le premier périphérique de fencing alimenté, cliquez sur **Ajouter une instance fence**. Ceci affiche un menu déroulant à partir duquel vous pouvez sélectionner le second périphérique de fencing que vous avez précédemment configuré, comme décrit dans la Section 3.6.1, « Créer un périphérique fence ».

11. Sélectionnez le second périphérique de fencing alimenté pour cette méthode et saisissez les paramètres appropriés pour celui-ci.

12. Cliquez sur **Submit**. Ceci vous ramène à l'écran spécifique aux nœuds où les méthodes et instances fence sont affichées, montrant que chaque périphérique éteindra et allumera le système en séquence. Ceci vous est montré dans la Figure 3.6, « Configuration du fencing à alimentation duelle ».

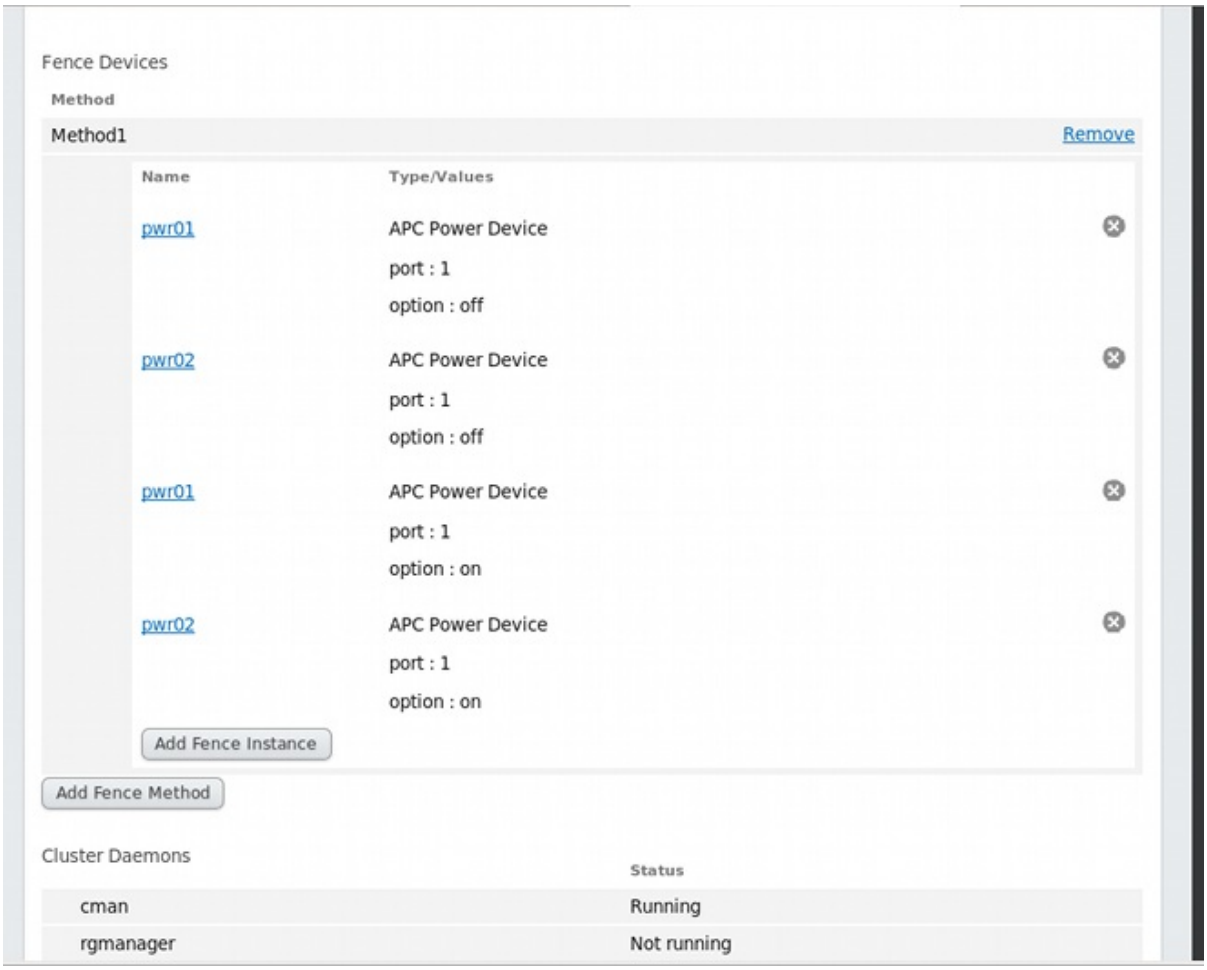


Figure 3.6. Configuration du fencing à alimentation duelle

3.8. CONFIGURER UN DOMAINE DE BASCULEMENT

Un domaine de basculement est un sous-ensemble nommé de nœuds d'un cluster qui sont capables d'exécuter un service cluster dans le cas d'un échec de nœud. Un domaine de basculement peut posséder les caractéristiques suivantes :

- Unrestricted — Ceci vous permet de spécifier qu'un sous-ensemble de membres est préféré, mais qu'un service cluster assigné à ce domaine peut s'exécuter sur n'importe quel membre disponible.

- Restricted — Ceci vous permet de restreindre les membres pouvant exécuter un service cluster en particulier. Si aucun des membres dans un domaine de basculement restricted n'est disponible, le service cluster ne pourra pas être lancé (manuellement ou par le logiciel du cluster).

- Unordered — Lorsqu'un service cluster est assigné à un domaine de basculement unordered, le membre sur lequel le service cluster est exécuté est choisi parmi les membres disponibles du domaine de basculement sans ordre de priorité.

- Ordered — Ceci vous permet de spécifier un ordre de préférence parmi les membres d'un domaine de basculement. Le membre le plus haut dans la liste est le préféré, suivi par le second membre dans la liste, et ainsi de suite.

- Failback — Ceci vous permet de spécifier si un service dans le domaine de basculement devrait être restauré sur le nœud sur lequel il était initialement exécuté avant que ce nœud tombe en panne. La configuration de cette caractéristique est utile dans des circonstances où un nœud tombe en panne de manière répétitive et fait partie d'un domaine de basculement ordered. Dans ces circonstances, si un nœud est le nœud préféré dans un domaine de basculement, il est possible qu'un service tombe en panne puis se restaure de manière répétitive entre le nœud préféré et un autre nœud, affectant sévèrement la performance.

NOTE

La caractéristique failback est uniquement applicable si le basculement ordered est configuré.

NOTE

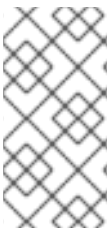
Modifier la configuration d'un domaine de basculement n'a aucun effet sur les services en cours d'exécution.

NOTE

Les domaines de basculement ne sont *pas* requis pour les opérations.

Par défaut, les domaines de basculement sont unrestricted et unordered.

Dans un cluster possédant plusieurs membres, l'utilisation d'un domaine de basculement restricted peut minimiser le travail de paramétrage du cluster pour qu'il exécute un service cluster (comme **`httpd`**), qui requiert que vous paramétriez la configuration de manière identique sur tous les membres exécutant le service cluster. Au lieu de paramétrer le cluster entier afin qu'il exécute le service cluster, il vous est possible de paramétrer uniquement les membres dans le domaine de basculement restricted que vous associerez au service cluster.



NOTE

Pour configurer un membre préféré, vous pouvez créer un domaine de basculement unrestricted comprenant uniquement un membre du cluster. Faire ceci cause au service cluster de s'exécuter sur ce membre du cluster en premier (le membre préféré), mais permet au service cluster de basculer sur tout autre membre.

Les sections suivantes décrivent l'addition, la modification et la suppression d'un domaine de basculement :

- Section 3.8.1, « Ajouter un domaine de basculement »
- Section 3.8.2, « Modifier un domaine de basculement »
- Section 3.8.3, « Supprimer un domaine de basculement »

3.8.1. Ajouter un domaine de basculement

Pour ajouter un domaine de basculement, suivez les étapes de cette section.

1. À partir de la page spécifique aux clusters, vous pouvez configurer des domaines de basculement pour ce cluster en cliquant sur **Domaines de basculement** en haut de l'affichage des clusters. Ceci affiche les domaines de basculement qui ont été configurés pour ce cluster.

2. Cliquez sur **Ajouter**. Cliquer sur **Ajouter** affichera la boîte de dialogue **`Ajouter le domaine de basculement au cluster`**, comme décrit dans la Figure 3.7, « Boîte de dialogue luci de la configuration du domaine de basculement ».

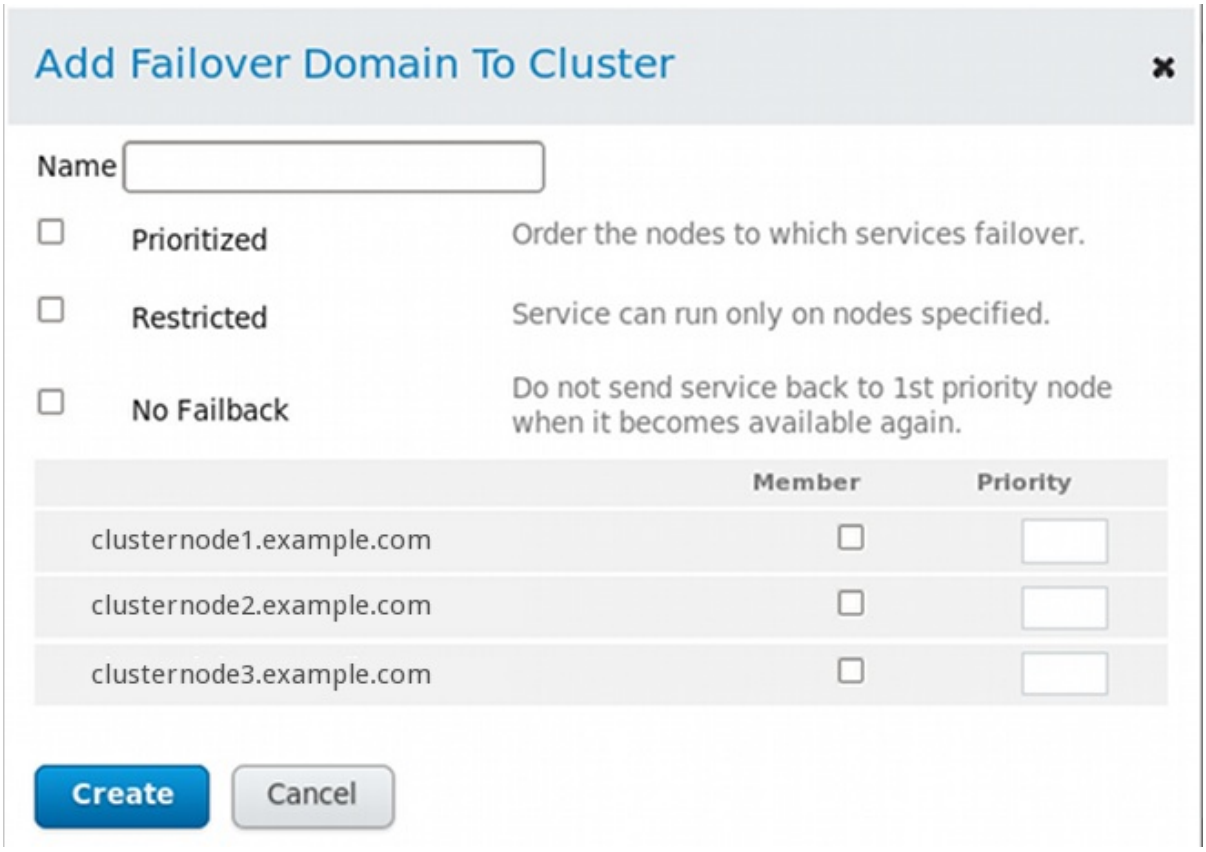


Figure 3.7. Boîte de dialogue luci de la configuration du domaine de basculement

3. Dans la boîte de dialogue **`Ajouter un domaine de basculement au cluster`**, spécifiez un nom de domaine de basculement dans la boîte de texte **Nom**.

NOTE

Le nom doit être suffisamment descriptif pour distinguer son but par rapport aux autres noms utilisés dans votre cluster.

4. Pour activer le paramétrage de la priorité des basculements des membres dans le domaine de basculement, cliquez sur la case à cocher **Prioritized** (Priorisés). Lorsque **Prioritized** est coché,

vous pouvez paramétrer la valeur de priorité **Priority** (Priorité) pour chaque nœud sélectionné en tant que membre du domaine de basculement.

5. Pour restreindre le basculement aux membres dans ce domaine de basculement, cliquez sur la case à cocher **Restricted** (Restreint). Lorsque **Restricted** est coché, les services assignés à ce domaine de basculement ne basculent que sur les nœuds dans ce domaine de basculement.

6. Pour spécifier qu'un nœud ne bascule pas dans ce domaine de basculement, cliquez sur la case à cocher **No Failback** (Pas de basculement). Lorsque **No Failback** est coché, si un service bascule depuis un nœud préféré, ce service ne basculera pas vers le nœud d'origine une fois que celui-ci est restauré.

7. Configurez les membres de ce domaine de basculement. Cliquez sur la case à cocher **Membre** de chaque nœud devant être un membre du domaine de basculement. Si **Prioritized** est coché, paramétrez la priorité dans la boîte de texte **Priority** pour chaque membre du domaine de basculement.

8. Cliquez sur **`Créer`**. Ceci affiche la page **`Domaines de basculement`** en affichant le domaine de basculement nouvellement créé. Un message indique que le nouveau domaine est en cours de création. Réactualisez la page pour mettre à jour l'état.

3.8.2. Modifier un domaine de basculement

Pour modifier un domaine de basculement, suivez les étapes de cette section.

1. À partir de la page spécifique aux clusters, vous pouvez configurer les domaines de basculement pour ce cluster en cliquant sur **Domaines de basculement** en haut de l'affichage des clusters. Ceci affiche les domaines de basculement qui ont été configurés pour ce cluster.

2. Cliquez sur le nom d'un domaine de basculement. Ceci affiche la page de configuration de ce domaine de basculement.

3. Pour modifier les propriétés **Prioritized**, **Restricted** ou **No Failback** du domaine de basculement, cochez ou décochez la case à cocher à côté de la propriété puis cliquez sur **`Update Properties`** (Mettre à jour les propriétés).

4. Pour modifier l'adhésion au domaine de basculement, cochez ou décochez la case à cocher à côté du membre du cluster. Si le domaine de basculement est priorisé, vous pouvez aussi modifier le paramètre de la priorité du membre du cluster. Cliquez ensuite sur **`Update Settings`** (Mettre à jour les paramètres).

3.8.3. Supprimer un domaine de basculement

Pour supprimer un domaine de basculement, suivez les étapes de cette section.

1. À partir de la page spécifique aux clusters, vous pouvez configurer les domaines de basculement pour ce cluster en cliquant sur **Domaines de basculement** en haut de l'affichage des clusters. Ceci affiche les domaines de basculement qui ont été configurés pour ce cluster.

2. Sélectionnez la case à cocher du domaine de basculement à supprimer.

3. Cliquez sur **Delete** (supprimer).

3.9. CONFIGURER LES RESSOURCES GLOBALES DU CLUSTER

Vous pouvez configurer les ressources globales pouvant être utilisées par tout service exécuté dans le cluster. Vous pouvez aussi configurer des ressources qui ne sont disponibles qu'à un service spécifique.

Pour ajouter une ressource globale de cluster, suivez les étapes de cette section. Vous pouvez ajouter une ressource qui est locale à un service en particulier lorsque vous configurez le service, comme décrit dans la Section 3.10, « Ajouter un service cluster à un cluster ».

1. Sur la page spécifique aux clusters, vous pouvez ajouter des ressources à ce cluster en cliquant sur **Ressources** en haut de l'affichage des clusters. Ceci affiche les ressources qui ont été configurées pour ce cluster.

2. Cliquez sur **Ajouter**. Ceci affiche le menu déroulant **Ajouter une ressource au cluster**.

3. Cliquez sur la boîte déroulante sous **Ajouter une ressource au cluster** et sélectionnez le type de ressource à configurer.

4. Saisissez les paramètres de ressources de la ressource que vous ajoutez. L'Annexe B, *Paramètres des ressources HA* décrit les paramètres de ressources.

5. Cliquez sur **`Soumettre`**. Cliquer sur **`Soumettre`** vous ramène à la page des ressources qui affiche l'écran **`Ressources`**, montrant la ressource ajoutée (ainsi que d'autres ressources).

Pour modifier une ressource existante, procédez aux étapes suivantes.

1. À partir de la page **luci Ressources**, cliquez sur le nom de la ressource à modifier. Ceci affiche les paramètres de cette ressource.

2. Modifiez les paramètres de la ressource.

3. Cliquez sur **`Appliquer`**.

Pour supprimer une ressource existante, procédez aux étapes suivantes.

1. À partir de la page **luci Ressources**, cliquez sur la case à cocher pour supprimer toute ressource.

2. Cliquez sur **Delete** (supprimer).

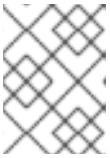
3.10. AJOUTER UN SERVICE CLUSTER À UN CLUSTER

Pour ajouter un service cluster au cluster, suivez les étapes de cette section.

1. Sur la page spécifique aux clusters, vous pouvez ajouter des services à ce cluster en cliquant sur **Groupes de services** en haut de l'affichage des clusters. Ceci affiche les services qui ont été configurés pour ce cluster. (Depuis la page **Groupes de services**, vous pouvez aussi démarrer, redémarrer et désactiver un service, comme décrit dans la Section 4.5, « Gérer les services High-Availability ».)

2. Cliquez sur **Ajouter**. Ceci affiche la boîte de dialogue **Ajouter un groupe de services au cluster**.

3. Dans la boîte de texte **Nom du service** se trouvant dans la boîte de dialogue **`Ajouter un groupe de services au cluster`**, saisissez le nom du service.



NOTE

Utilisez un nom descriptif qui distingue clairement le service des autres services dans le cluster.

4. Cochez la case **Démarrer ce service automatiquement** si vous souhaitez que ce service démarre automatiquement lorsqu'un cluster est lancé et fonctionne. Si la case n'est *pas* cochée, le service devra être lancé manuellement à chaque fois que le cluster sortira de l'état arrêté.

5. Cochez la case **Run exclusive** pour définir une stratégie avec laquelle le service ne s'exécute que sur des nœuds sur lesquels aucun autre service ne s'exécute.

6. Si vous avez configuré des domaines de basculement pour le cluster, vous pouvez utiliser le menu déroulant du paramètre **Domaine de basculement** pour sélectionner un domaine de basculement pour ce service. Pour obtenir des informations sur la configuration de domaines de basculement, voir la Section 3.8, « Configurer un domaine de basculement ».

7. Utilisez la boîte déroulante **Politique de récupération** pour sélectionner une politique de récupération pour le service. Les options pour le service sont **Relocate** (Déplacer), **Restart** (Redémarrer), **Restart-Disable** (Redémarrer-désactiver), ou **Disable** (Désactiver).

   Sélectionner l'option **Restart** (redémarrer) indique que le système devrait tenter de redémarrer le service en échec avant de le déplacer. Sélectionner l'option **Relocate** (déplacer) indique que le système devrait tenter de redémarrer le service dans un autre nœud. Sélectionner l'option **Disable** (désactiver) indique que le système devrait désactiver le groupe de ressources si l'un des composants échoue. Sélectionner l'option **Restart-Disable** (redémarrer-désactiver) indique que le système devrait tenter de redémarrer le service à sa place s'il échoue, mais si le redémarrage échoue, alors le service sera désactivé au lieu d'être déplacé vers un autre hôte dans le cluster.

   Si vous sélectionnez **Restart** ou **Restart-Disable** en tant que politique de récupération pour le service, vous pourrez spécifier le nombre maximum d'échecs de redémarrage avant le déplacement ou la désactivation du service. Vous pouvez aussi spécifier (en secondes) à partir de combien de temps il ne faudra plus effectuer de redémarrages.

8. Pour ajouter une ressource au service, cliquez sur **`Ajouter une ressource`**. Cliquer sur **`Ajouter une ressource`** affiche la boîte déroulante de l'écran **`Ajouter une ressource au service`** qui vous permet d'ajouter une ressource globale existante ou d'ajouter une nouvelle ressource qui est *uniquement* disponible à ce service.

   - Pour ajouter une ressource globale existante, cliquez sur le nom de la ressource existante dans la boîte déroulante **`Ajouter une ressource au service`**. Ceci affiche la ressource et ses paramètres sur la page **Groupes de services** pour le service que vous configurez. Pour obtenir des informations sur l'ajout et sur la modification des ressources globales, voir la Section 3.9, « Configurer les ressources globales du cluster ».

   - Pour ajouter un nouvelle ressource uniquement disponible à ce service, sélectionnez le type de ressource à configurer à partir de la boîte déroulante **`Ajouter une ressource au service`** et saisissez les paramètres de ressources pour la ressource que vous ajoutez. L'Annexe B, *Paramètres des ressources HA* décrit les paramètres de ressources.

   - Lors de l'ajout d'une ressource à un service, qu'il s'agisse d'une ressource globale existante ou d'une ressource uniquement disponible à ce service, vous pouvez spécifier si la ressource est une **sous-arborescence indépendante** ou une **Ressource non-critique**.

     Si vous spécifiez la ressource en tant qu'arborescence indépendante et que celle-ci échoue,

elle seule sera redémarrée (plutôt que le service entier) avant que le système ne tente d'effectuer une récupération normale. Vous pouvez spécifier le nombre maximum de redémarrages que cette ressource devra tenter sur un nœud avant d'implémenter une politique de récupération pour ce service. Vous pouvez aussi spécifier la durée, en secondes, après laquelle le système implémentera la politique de récupération pour ce service.

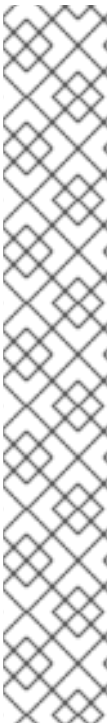
Si vous spécifiez la ressource en tant que ressource non-critique et que celle-ci échoue, elle seule sera redémarrée. Si elle continue à échouer, alors plutôt que le service entier, elle seule sera désactivée. Vous pouvez spécifier le nombre maximum de redémarrages que cette ressource devra tenter sur un nœud avant de la désactiver. Vous pouvez aussi spécifier la durée, en secondes, après laquelle le système désactivera cette ressource.

9. Si vous souhaitez ajouter des ressources enfant à la ressource que êtes en train de définir, cliquez sur **`Ajouter une ressource enfant`**. Cliquer sur **`Ajouter une ressource enfant`** affiche la boite déroulante **Ajouter une ressource au service**, à partir de laquelle vous pouvez ajouter une ressource globale existante ou une nouvelle ressource uniquement disponible à ce service. Vous pouvez continuer d'ajouter des ressources enfant à la ressource selon vos besoins.

NOTE

Si vous êtes en train d'ajouter une ressource du service Samba, ajoutez-la directement au service, et *non pas* en tant qu'enfant d'une autre ressource.

10. Lorsque vous aurez fini d'ajouter des ressources au service et des ressources enfant aux ressources, cliquez sur **`Soumettre`**. Cliquer sur **`Soumettre`** vous ramène à la page **Groupes de services**, qui affiche le service ajouté (et les autres services).



NOTE

Pour vérifier l'existence de la ressource du service IP utilisée dans un service cluster, vous pouvez utiliser la commande **`/sbin/ip addr show`** sur un nœud de cluster (plutôt que la commande obsolète **`ifconfig`**). La sortie suivante montre la commande **`/sbin/ip addr show`** exécutée sur un nœud qui exécute un service cluster :

```
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1356 qdisc pfifo_fast
qlen 1000
    link/ether 00:05:5d:9a:d8:91 brd ff:ff:ff:ff:ff:ff
    inet 10.11.4.31/22 brd 10.11.7.255 scope global eth0
    inet6 fe80::205:5dff:fe9a:d891/64 scope link
    inet 10.11.4.240/22 scope global secondary eth0
       valid_lft forever preferred_lft forever
```

Pour modifier un service existant, procédez aux étapes suivantes.

1. À partir de la boîte de dialogue **Groupes de services**, cliquez sur le nom du service à modifier. Ceci affiche les paramètres et les ressources qui ont été configurés pour ce service.

2. Modifiez les paramètres de service.

3. Cliquez sur **`Submit`** (Soumettre).

Pour supprimer un ou plusieurs service(s) existant(s), procédez aux étapes suivantes.

1. À partir de la page **luci Groupes de services**, cliquez sur la case à cocher pour supprimer tout service.

2. Cliquez sur **Delete** (supprimer).

3. À partir de Red Hat Enterprise Linux 6.3, avant que **luci** ne supprime un (ou plusieurs) service(s), un message s'affiche vous demandant de confirmer si vous souhaitez bien supprimer le ou les groupe(s) de services, ce qui arrête les ressources qui le ou qui les contiennent. Cliquez sur **Annuler** pour fermer la boîte de dialogue sans supprimer de services, ou sur **Procéder** pour supprimer le ou les service(s) sélectionné(s).

CHAPITRE 4. GÉRER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY AVEC CONGA

Ce chapitre décrit les diverses tâches administratives pour la gestion du module complémentaire Red Hat High Availability et comporte les sections suivantes :

- Section 4.1, « Ajouter un cluster existante à l'interface luci »
- Section 4.2, « Supprimer un cluster de l'interface luci »
- Section 4.3, « Gérer les nœuds de clusters »
- Section 4.4, « Démarrer, arrêter, redémarrer et supprimer des clusters »
- Section 4.5, « Gérer les services High-Availability »
- Section 4.6, « Effectuer une copie de sauvegarde et restaurer une configuration Luci »

4.1. AJOUTER UN CLUSTER EXISTANTE À L'INTERFACE LUCI

Si vous avez déjà créé un cluster avec module complémentaire High Availability, vous pourrez facilement ajouter le cluster à l'interface **luci**, de manière à gérer le cluster avec **Conga**.

Pour ajouter un cluster existant à l'interface **luci**, procédez aux étapes suivantes :

1. Cliquez sur **Gérer les clusters** dans le menu à gauche de la page **luci Homebase**. L'écran **Clusters** s'affiche.
2. Cliquez sur **Ajouter**. L'écran **Ajouter un cluster existant** s'affiche.
3. Saisissez le nom d'hôte du nœud et un mot de passe **ricci** pour n'importe quel nœud dans le cluster existant. Chaque nœud dans le cluster contient toutes les informations de configuration du cluster, celles-ci devraient être suffisantes pour pouvoir ajouter le cluster à l'interface **luci**.
4. Cliquez sur **`Connecter`**. L'écran **Ajouter un cluster existant** affiche alors le nom du cluster et les nœuds restants dans celui-ci.
5. Saisissez les mots de passe **ricci** individuels pour chaque nœuds dans le cluster, ou saisissez un mot de passe et sélectionnez **Utiliser le même mot de passe pour tous les nœuds**.
6. Cliquez sur **`Ajoute un cluster`**. Le cluster précédemment configuré s'affiche maintenant sur l'écran **Gérer les clusters**.

4.2. SUPPRIMER UN CLUSTER DE L'INTERFACE LUCI

Vous pouvez supprimer un cluster de l'interface utilisateur graphique de gestion **luci** sans affecter les services ou l'abonnement du cluster. Si vous supprimez un cluster, vous pouvez le rajouter plus tard ou l'ajouter à une autre instance **luci**, comme le décrit la Section 4.1, « Ajouter un cluster existante à l'interface luci ».

Pour supprimer un cluster de l'interface utilisateur graphique de gestion **luci** sans affecter les services ou l'abonnement du cluster, veuillez suivre les étapes suivantes :

1. Cliquez sur **Gérer les clusters** dans le menu à gauche de la page **luci Homebase**. L'écran **Clusters** s'affiche.

2. Sélectionnez le ou les cluster(s) que vous souhaitez supprimer.

3. Cliquez sur **Delete** (supprimer).

Pour obtenir des informations sur la suppression complète d'un cluster, l'arrêt de tous les services du cluster et la suppression des informations de configuration du cluster depuis les nœuds, reportez-vous à la Section 4.4, « Démarrer, arrêter, redémarrer et supprimer des clusters ».

4.3. GÉRER LES NŒUDS DE CLUSTERS

Cette section document comment effectuer les fonctions suivantes de gestion de nœuds avec le composant serveur de **Conga**, **luci** :

- Section 4.3.1, « Redémarrer un nœud de cluster »
- Section 4.3.2, « Causer à un nœud de joindre ou quitter un cluster »
- Section 4.3.3, « Ajouter un membre à un cluster en cours d'exécution »
- Section 4.3.4, « Supprimer un membre d'un cluster »

4.3.1. Redémarrer un nœud de cluster

Pour redémarrer un nœud dans un cluster, procédez aux étapes suivantes :

1. À partir de la page spécifique aux clusters, cliquez sur **nœuds** en haut de l'affichage des clusters. Ceci affiche les nœuds constituant le cluster. Ceci est aussi la page par défaut apparaissant lorsque vous cliquez sur le nom du cluster en dessous de **Gérer les clusters** dans le menu sur le côté gauche de la page **Homebase luci**.

2. Sélectionnez le nœud à redémarrer en cliquant sur la case à cocher pour ce nœud.

3. Sélectionnez la fonction **Reboot** (Redémarrer) dans le menu en haut de la page. Ceci cause au nœud sélectionné de redémarrer et un message s'affiche en haut de la page indiquant que le nœud est en train de redémarrer.

4. Réactualise la page pour voir l'état du nœud mis à jour.

Il est aussi possible de redémarrer plus d'un nœud à la fois en sélectionnant tous les nœuds que vous souhaitez redémarrer avant de cliquer sur **Reboot**.

4.3.2. Causer à un nœud de joindre ou quitter un cluster

Vous pouvez utiliser le composant serveur de **Conga**, **luci** pour faire en sorte qu'un nœud quitte un cluster actif en arrêtant tous les services cluster sur ce nœud. Vous pouvez aussi utiliser le composant serveur de **Conga**, **luci** pour causer à un nœud ayant quitté le cluster de rejoindre le cluster.

Causer à un nœud de quitter un cluster ne supprime pas les informations de configuration du cluster de ce nœud, et celui-ci apparaît toujours dans l'affichage du nœud du cluster avec le statut **`Not a cluster member`**(N'est pas un membre du cluster). Pour obtenir des informations sur la suppression complète du nœud de la configuration du cluster, voir la Section 4.3.4, « Supprimer un membre d'un cluster ».

Pour faire qu'un nœud quitte un cluster, effectuez les étapes suivantes. Ceci éteint le logiciel du cluster dans le nœud. Faire en sorte qu'un nœud quitte un cluster empêche le nœud de rejoindre le cluster automatiquement lorsqu'il est redémarré.

1. À partir de la page spécifique aux clusters, cliquez sur **nœuds** en haut de l'affichage des clusters. Ceci affiche les nœuds constituant le cluster. Ceci est aussi la page par défaut apparaissant lorsque vous cliquez sur le nom du cluster en dessous de **Gérer les clusters** dans le menu sur le côté gauche de la page **Homebase luci**.

2. Sélectionnez le nœud que vous souhaitez faire quitter le cluster en cliquant sur la case à cocher de ce nœud.

3. Sélectionnez la fonction **Quitter le cluster** (en anglais, « Leave Cluster ») dans le menu en haut de la page. Ceci fait apparaître un message en haut de la page indiquant que le nœud est en train d'être arrêté.

4. Réactualise la page pour voir l'état du nœud mis à jour.

Il est aussi possible de faire en sorte que plus d'un nœud quitte le cluster à la fois en sélectionnant tous les nœuds devant quitter le cluster avant de cliquer sur **Quitter le cluster**.

Pour causer à un nœud de rejoindre un cluster, sélectionnez tous les nœuds que vous souhaitez faire rejoindre le cluster en cliquant sur la case à cocher de ceux-ci et en sélectionnant **Rejoindre le cluster**. Ceci cause aux nœuds sélectionnés de rejoindre le cluster, et permet aux nœuds sélectionnés de le rejoindre lorsqu'ils sont redémarrés.

4.3.3. Ajouter un membre à un cluster en cours d'exécution

Pour ajouter un membre à un cluster en cours d'exécution, suivez les étapes de cette section.

1. À partir de la page spécifique aux clusters, cliquez sur **nœuds** en haut de l'affichage du cluster. Ceci affiche les nœuds constituant le cluster. Ceci est aussi la page par défaut apparaissant lorsque vous cliquez sur le nom du cluster en-dessous de **Gérer les clusters** sur le côté gauche de la page **luci**, **Homebase**.

2. Cliquez sur **Ajouter**. Cliquer sur **Ajouter** provoque l'affichage de la boîte de dialogue **`Ajouter des nœuds au cluster`**.

3. Saisissez le nom du nœud dans la boîte de texte **Nom d'hôte du nœud** ; saisissez le mot de passe **ricci** dans la boîte de texte **Mot de passe**. Si vous utilisez un port différent pour l'agent **ricci** autre que celui par défaut, 11111, modifiez ce paramètre sur le port que vous utilisez.

4. Cochez la case **Activer le support du stockage partagé** (de l'anglais, « Enable Shared Storage Support ») si le stockage clusterisé est requis pour télécharger les paquetages qui prennent en charge le stockage clusterisé et activez LVM sous clusters ; vous devriez sélectionner ceci uniquement lorsque vous avez accès au module complémentaire Resilient Storage ou au module complémentaire Scalable File System.

5. Si vous souhaitez ajouter plus de nœuds, cliquez sur **`Ajouter un autre nœud`** et saisissez le nom du nœud et le mot de passe pour chaque nœud supplémentaire.

6. Cliquez sur **`Ajouter des nœuds`**. Cliquer sur **`Ajouter des nœuds`** provoque les actions suivantes :

    1. Si vous avez sélectionné **Download Packages** (Télécharger les paquetages), les paquetages logiciels du cluster sont téléchargés sur les nœuds.

    2. Les logiciels du cluster sont installés sur les nœuds (sinon, il est vérifié que les bons paquetages logiciels sont installés).

3. Le fichier de configuration est mis à jour et propagé vers chaque nœud dans le cluster — y compris vers le nœud ajouté.

4. Le nœud ajouté rejoint le cluster.

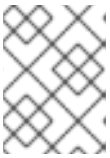
La page **nœuds** s'affiche avec un message indiquant que le nœud est en train d'être ajouté au cluster. Réactualisez la page pour mettre le statut à jour.

7. Lorsque le processus d'ajout du nœud est terminé, cliquez sur le nom du nœud pour que le nœud nouvellement ajouté configure le fencing pour le nœud, comme le décrit la Section 3.6, « Configurer des périphériques fence ».

4.3.4. Supprimer un membre d'un cluster

Pour supprimer un membre d'un cluster existant qui est en cours d'opération, suivez les étapes de cette section. Remarquez que les nœuds doivent être arrêtés avec d'être supprimés à moins que vous ne supprimiez tous les nœuds du cluster à la fois.

1. À partir de la page spécifique aux clusters, cliquez sur **nœuds** en haut de l'affichage du cluster. Ceci affiche les nœuds constituant le cluster. Ceci est aussi la page par défaut apparaissant lorsque vous cliquez sur le nom du cluster en-dessous de **Gérer les clusters** sur le côté gauche de la page **luci**, **Homebase**.



NOTE

Pour permettre aux services exécutés sur nœud de basculer lorsque le nœud est supprimé, ignorez l'étape suivante.

2. Désactivez ou déplacez chaque service en cours d'exécution sur le nœud à supprimer. Pour obtenir des informations sur la désactivation et le déplacement des services, voir la Section 4.5, « Gérer les services High-Availability ».

3. Sélectionnez le ou les nœud(s) à supprimer.

4. Cliquez sur **Supprimer**. La page **nœuds** indique que le nœud est en cours de suppression. Réactualisez la page pour voir le statut actuel.

IMPORTANT

La suppression d'un nœud de cluster

4.4. DÉMARRER, ARRÊTER, REDÉMARRER ET SUPPRIMER DES CLUSTERS

Vous pouvez démarrer, arrêter, redémarrer et supprimer un cluster en effectuant ces actions sur les nœuds individuels dans le cluster. À partir de la page spécifique aux clusters, cliquez sur **nœuds** en haut de l'affichage du cluster. Ceci affiche les nœuds constituant le cluster.

Les opérations de démarrage et de redémarrage de nœuds de clusters ou de clusters entiers vous permettent de créer de courtes pannes des services du cluster si un service doit être déplacé sur un autre membre du cluster parce qu'il est exécuté sur un nœud devant être arrêté ou redémarré.

Pour arrêter un cluster, effectuez les étapes suivantes. Celles-ci ferment le logiciel du cluster dans les nœuds, mais cela ne supprime pas les informations de la configuration du cluster des nœuds et les

nœuds apparaissent toujours sur l'affichage des nœuds du cluster avec le statut **`Not a cluster member`**.

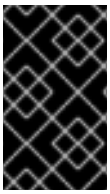
1. Sélectionnez tous les nœuds dans le cluster en cliquant sur la case à cocher à côté de chaque nœud.
2. Sélectionnez la fonction **Quitter le cluster** dans le menu en haut de la page. Ceci fait apparaître un message en haut de la page indiquant que chaque nœud est en train d'être arrêté.
3. Réactualisez la page pour voir le statut mis à jour des nœuds.

Pour démarrer un cluster, effectuez les étapes suivantes :

1. Sélectionnez tous les nœuds dans le cluster en cliquant sur la case à cocher à côté de chaque nœud.
2. Sélectionnez la fonction **Rejoindre un cluster** dans le menu en haut de la page.
3. Réactualisez la page pour voir le statut mis à jour des nœuds.

Pour redémarrer un cluster en cours d'exécution, commencez par arrêter tous les nœuds dans le cluster, puis démarrez tous les nœuds dans le cluster, comme décrit ci-dessus.

Pour supprimer un cluster entier, effectuez les étapes suivantes. Ceci cause à tous les services du cluster de s'arrêter et supprime les informations de configuration des nœuds et de l'affichage du cluster. Si vous décidez d'ajouter un cluster existant ultérieurement à l'aide de l'un des nœuds que vous avez supprimé, **luci** indiquera que le nœud n'est membre d'aucun cluster.



IMPORTANT

La suppression d'un cluster est une opération destructive qui ne peut pas être annulée. Restaurer un cluster après l'avoir supprimé nécessite de recréer et redéfinir le cluster depuis le début.

1. Sélectionnez tous les nœuds dans le cluster en cliquant sur la case à cocher à côté de chaque nœud.
2. Sélectionnez la fonction **Supprimer** du menu en haut de la page.

Si vous souhaitez supprimer un cluster de l'interface **luci** sans arrêter le moindre service cluster ou sans modifier l'appartenance du cluster, vous pouvez utiliser l'option **Supprimer** sur la page **Gérer les clusters**, comme le décrit la Section 4.2, « Supprimer un cluster de l'interface luci ».

4.5. GÉRER LES SERVICES HIGH-AVAILABILITY

En plus d'ajouter et de modifier un service, comme décrit dans la Section 3.10, « Ajouter un service cluster à un cluster », il vous est aussi possible d'utiliser les fonctions de gestion pour les services haute disponibilité (high-availability) via la composante serveur de **Conga**, **luci** :

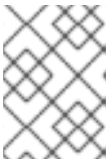
- Démarrer un service
- Redémarrer un service
- Désactiver un service
- Supprimer un service

- Déplacer un service

À partir de la page spécifique aux clusters, vous pouvez gérer les services pour ce cluster en cliquant sur **Groupes de services** en haut de l'affichage des clusters. Ceci affiche les services qui ont été configurés pour ce cluster.

- **Démarrer un service** — Pour démarrer tout service qui n'est pas en cours d'exécution, sélectionnez le service que vous souhaitez démarrer en cliquant sur sa case à cocher, puis cliquez sur **Démarrer**.
- **Redémarrer un service** — Pour redémarrer tout service en cours d'exécution, sélectionnez le service que vous souhaitez redémarrer en cliquant sur sa case à cocher, puis cliquez sur **Redémarrer**.
- **Désactiver un service** — Pour désactiver tout service en cours d'exécution, sélectionnez le service que vous souhaitez désactiver en cliquant sur sa case à cocher, puis cliquez sur **Désactiver**.
- **Supprimer un service** — Pour supprimer tout service qui n'est pas en cours d'exécution, sélectionnez le service que vous souhaitez supprimer en cliquant sur sa case à cocher, puis cliquez sur **Supprimer**.
- **Déplacer un service** — Pour déplacer un service en cours d'exécution, cliquez sur le nom du service dans l'écran affichant les services. Ceci affiche la page de configuration des services, indiquant sur quel nœud le service est actuellement en cours d'exécution.

  Dans la boîte déroulante **`Start on node...`** (Démarrer sur le nœud...), sélectionnez le nœud sur lequel vous souhaitez déplacer le service, puis cliquez sur l'icône **Démarrer**. Un message s'affichera en haut de l'écran, indiquant que le service est en train de démarrer. Vous pourriez devoir réactualiser l'écran pour voir si le service est exécuté sur le nœud que vous avez sélectionné.



NOTE

Si le service que vous avez sélectionné est un service **`vm`**, la boîte déroulante affichera l'option **`migrer`** au lieu de l'option **`déplacer`**.

NOTE

Vous pouvez aussi démarrer, redémarrer, désactiver ou supprimer un service individuel en cliquant sur le nom du service sur la page **Services**. Ceci affiche la page de configuration du service. En haut à droite de la page de configuration du service se trouvent les mêmes icônes pour **Démarrer**, **Redémarrer**, **Désactiver** et **Supprimer**.

4.6. EFFECTUER UNE COPIE DE SAUVEGARDE ET RESTAURER UNE CONFIGURATION LUCI

À partir de Red Hat Enterprise Linux 6.2, vous pouvez utiliser la procédure suivante pour effectuer une copie de sauvegarde de la base de données **luci**, qui est stockée dans le fichier **`/var/lib/luci/data/luci.db`**. Il ne s'agit pas de la configuration du cluster, qui est stocké dans le fichier **`cluster.conf`**. Au contraire, ce fichier contient la liste des utilisateurs, des clusters et des propriétés liées que **luci** maintient. Par défaut, la sauvegarde que cette procédure crée sera écrite sur le même répertoire que le fichier **`luci.db`**.

1. Exécutez **service luci stop**.
2. Exécutez **service luci backup-db**.

   Optionnellement, vous pouvez spécifier un nom de fichier en tant que paramètre pour la commande **backup-db**, qui écrira la base de données **luci** sur ce fichier. Par exemple, pour écrire la base de données **luci** sur le fichier **/root/luci.db.backup**, vous pouvez exécuter la commande **service luci backup-db /root/luci.db.backup**. Remarquez cependant que les fichiers de sauvegarde qui sont écrits sur des emplacements autres que **/var/lib/luci/data/** (pour les sauvegardes dont les noms de fichiers sont spécifiés lors de l'utilisation de **service luci backup-db**) n'apparaîtront pas dans la sortie de la commande **list-backups**.
3. Exécutez **service luci start**.

Utilisez la procédure suivante pour restaurer une base de données **luci**.

1. Exécutez **service luci stop**.
2. Exécutez **service luci list-backups** et notez le nom du fichier à restaurer.
3. Exécutez **service luci restore-db /var/lib/luci/data/*lucibackupfile***, où *lucibackupfile* est le fichier de sauvegarde à restaurer.

   Par exemple, la commande suivante restaure les informations de configuration **luci** qui étaient stockées dans le fichier **luci-backup20110923062526.db** :

   ```
   service luci restore-db /var/lib/luci/data/luci-
   backup20110923062526.db
   ```
4. Exécutez **service luci start**.

Si vous devez restaurer une base de données **luci** mais que vous avez perdu le fichier **host.pem** de la machine sur laquelle vous avez créé la sauvegarde, par exemple à cause d'une réinstallation complète, vous devrez ajouter vos clusters sur **luci** manuellement afin de ré-authentifier les nœuds du cluster.

Utilisez la procédure suivante pour restaurer une base de données **luci** sur une machine autre que celle sur laquelle la sauvegarde a été créée. Remarquez qu'en plus de restaurer la base de données, vous devez aussi copier le fichier certificat SSL afin de vous assurer que **luci** a bien été authentifié sur les nœuds **ricci**. Dans cet exemple, la sauvegarde est créée sur la machine **luci1** et la sauvegarde est restaurée sur la machine **luci2**.

1. Exécutez la séquence de commandes suivante pour créer une copie de sauvegarde de **luci** sur **luci1** et copiez le fichier certificat SSL et la sauvegarde **luci** sur **luci2**.

   ```
   [root@luci1 ~]# service luci stop
   [root@luci1 ~]# service luci backup-db
   [root@luci1 ~]# service luci list-backups
   /var/lib/luci/data/luci-backup20120504134051.db
   [root@luci1 ~]# scp /var/lib/luci/certs/host.pem
   /var/lib/luci/data/luci-backup20120504134051.db root@luci2:
   ```
2. Sur la machine **luci2**, assurez-vous que **luci** a été installé et n'est pas en cours d'exécution. Installez le paquet s'il ne l'a pas déjà été.

3. Exécutez la séquence de commandes suivante afin de vous assurer que les authentifications sont effectuées et pour restaurer la base de données **luci** de **`luci1`** sur **`luci2`**.

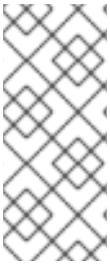
```
[root@luci2 ~]# cp host.pem /var/lib/luci/certs/
[root@luci2 ~]# chown luci: /var/lib/luci/certs/host.pem
[root@luci2 ~]# /etc/init.d/luci restore-db ~/luci-
backup20120504134051.db
[root@luci2 ~]# shred -u ~/host.pem ~/luci-backup20120504134051.db
[root@luci2 ~]# service luci start
```

CHAPITRE 5. CONFIGURER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY AVEC LA COMMANDE CCS

À partir de la version 6.1 de Red Hat Enterprise Linux, le module complémentaire Red Hat High Availability fournit la prise en charge de la commande de configuration du cluster **ccs**. La commande **ccs** permet à un administrateur de créer, de modifier et d'afficher le fichier de configuration du cluster **cluster.conf**. Vous pouvez utiliser la commande **ccs** pour configurer un fichier de configuration de cluster sur un système de fichiers local ou sur un nœud distant. Avec la commande **ccs**, un administrateur peut aussi démarrer et arrêter les services du cluster sur un ou tous les nœuds d'un cluster configuré.

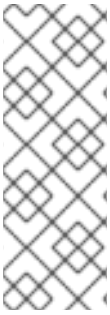
Ce chapitre décrit comment configurer le fichier de configuration du cluster du module complémentaire Red Hat High Availability avec la commande **ccs**. Pour obtenir des informations sur l'utilisation de la commande **ccs** pour gérer un cluster en cours d'exécution, voir Chapitre 6, *Gérer le module complémentaire Red Hat High Availability avec **ccs***.

Ce chapitre est composé des sections suivantes :



NOTE

Assurez-vous que le déploiement du module complémentaire High Availability répond bien à vos besoins et qu'il est pris en charge. Consultez un représentant Red Hat autorisé afin de vérifier votre configuration avant le déploiement. En outre, prévoyez suffisamment de temps pour une période de rodage de la configuration afin de tester les différents modes d'échec.



NOTE

Ce chapitre fait référence aux éléments et attributs de **cluster.conf** communément utilisés. Pour obtenir la liste et la description complète des éléments et attributs **cluster.conf**, reportez-vous au schéma des clusters sur **/usr/share/cluster/cluster.rng**, et au schéma annoté sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

5.1. APERÇU OPÉRATIONNEL

Cette section décrit les aspects opérationnels généraux d'utilisation de la commande **ccs** pour configurer un cluster :

- Section 5.1.1, « Créer le fichier de configuration du cluster sur un système local »
- Section 5.1.2, « Afficher la configuration actuelle du cluster »
- Section 5.1.3, « Spécifier les mots de passe ricci avec la commande css »
- Section 5.1.4, « Modifier les composants de la configuration du cluster »

5.1.1. Créer le fichier de configuration du cluster sur un système local

À l'aide de la commande **ccs**, vous pouvez créer un fichier de configuration de cluster sur un nœud de cluster, ou un fichier de configuration de cluster sur un système de fichiers local, puis envoyer ce fichier sur un hôte dans un cluster. Ceci vous permet de travailler sur un fichier à partir d'une machine locale, où vous pourrez le maintenir sous contrôle de version, ou étiqueter le fichier selon vos besoins. L'utilisation de la commande **ccs** ne requiert pas le privilège root.

Lorsque vous créez et modifiez un fichier de configuration sur un nœud de cluster avec la commande **ccs**, vous utilisez l'option **-h** pour spécifier le nom de l'hôte. Ceci crée et modifie le fichier **cluster.conf** sur l'hôte :

```
ccs -h host [options]
```

Pour créer et modifier un fichier de configuration de cluster sur un système local, utilisez l'option **-f** de la commande **ccs** pour spécifier le nom du fichier de configuration lorsque vous effectuez une opération de cluster. Vous pouvez nommer ce fichier comme bon vous semble.

```
ccs -f file [options]
```

Après avoir créé le fichier localement, vous pouvez l'envoyer à un nœud de cluster à l'aide de l'option **--setconf** de la commande **ccs**. Sur une machine hôte dans un cluster, le fichier envoyé sera nommé **cluster.conf** et sera placé dans le répertoire **/etc/cluster**.

```
ccs -h host -f file --setconf
```

Pour obtenir des informations sur l'utilisation de l'option **--setconf** de la commande **ccs**, voir la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.1.2. Afficher la configuration actuelle du cluster

Si à tout moment pendant la création d'un fichier de configuration de cluster, vous souhaitez imprimer le fichier actuel, veuillez utiliser la commande suivante, en spécifiant un nœud dans le cluster en tant qu'hôte :

```
ccs -h host --getconf
```

Si vous créez le fichier de configuration de votre cluster sur un système local, vous pouvez spécifier l'option **-f** au lieu de l'option **-h**, comme décrit dans la Section 5.1.1, « Créer le fichier de configuration du cluster sur un système local ».

5.1.3. Spécifier les mots de passe ricci avec la commande css

L'exécution de commandes **ccs** qui distribuent des copies du fichier **cluster.conf** aux nœuds d'un cluster requiert que **ricci** soit installé et exécuté sur les nœuds du cluster, comme décrit dans la Section 2.13, « Considérations pour **ricci** ». L'utilisation de **ricci** requiert un mot de passe la première fois que vous aurez une interaction avec **ricci**, et ce, depuis n'importe quelle machine spécifique.

Si vous n'avez pas saisi de mot de passe pour une instance de **ricci** sur une machine en particulier à partir de la machine que vous utilisez, il vous sera demandé ce mot de passe lorsque la commande **ccs** le requerra. Alternativement, vous pouvez utiliser l'option **-p** pour spécifier un mot de passe **ricci** sur la ligne de commande.

```
ccs -h host -p password --sync --activate
```

Lorsque vous propagez le fichier **cluster.conf** vers tous les nœuds du cluster avec l'option **--sync** de la commande **ccs** et que vous spécifiez un mot de passe **ricci** pour la commande, la commande **ccs** utilisera ce mot de passe pour chaque nœud du cluster. Si vous devez définir différents mots de passe pour **ricci** sur des nœuds individuels, vous pouvez utiliser l'option **--setconf** avec l'option **-p** pour distribuer le fichier de configuration sur un nœud à la fois.

5.1.4. Modifier les composants de la configuration du cluster

Utilisez la commande **ccs** pour configurer les composants du cluster et leurs attributs dans le fichier de configuration du cluster. Après avoir ajouté un composant de cluster au fichier dans le but de modifier les attributs de ce composant, vous devrez supprimer le composant quvous avez défini puis ajouter ce composant à nouveau, avec les attributs modifiés. Des informations sur la manière d'effectuer cela avec chaque composant sont fournies dans des sections individuelles de ce chapitre.

Les attributs du composant de cluster **cman** fournissent une exception à cette procédure pour modifier les composants de clusters. Pour modifier ces attributs, exécutez l'option **--setcman** de la commande **ccs**, en spécifiant les nouveaux attributs. Remarquez que spécifier cette option ré-initialisera toutes les valeurs que vous n'aurez pas explicitement spécifié comme étant des valeurs par défaut, comme le décrit la Section 5.1.5, « Commandes remplaçant les paramètres précédents ».

5.1.5. Commandes remplaçant les paramètres précédents

Il existe plusieurs options de la commande **ccs** qui implémentent des sémantiques de remplacement lors de la définition de propriétés. Cela signifie que vous pouvez exécuter la commande **ccs** avec l'une de ces options sans spécifier de paramètres et tous les paramètres seront ré-initialisés à leurs valeurs par défaut. Ces options sont comme suit :

- **--settotem**

- **--setdlm**
- **--setrm**
- **--setcman**
- **--setmulticast**
- **--setaltmulticast**
- **--setfencedaemon**
- **--setlogging**
- **--setquorumd**

Par exemple, pour réinitialiser toutes les propriétés du démon fence, vous pouvez exécuter la commande suivante :

```
# ccs -h hostname --setfencedaemon
```

Remarquez cependant que si vous utilisez l'une de ces commandes pour réinitialiser une propriété, alors les autres propriétés de la commande seront réinitialisées à leurs valeurs par défaut. Par exemple, vous pouvez utiliser la commande suivante pour définir la propriété **post_fail_delay** sur 5 :

```
# ccs -h hostname --setfencedaemon post_fail_delay=5
```

Si, après avoir exécuté cette commande, vous exécutez la commande suivante pour réinitialiser la propriété **post_join_delay** sur 10, la propriété **post_fail_delay** sera restaurée à sa valeur par défaut :

```
# ccs -h hostname --setfencedaemon post_join_delay=10
```

Pour réinitialiser les propriétés **post_fail_delay** et **post_join_delay**, indiquez-les toutes les deux sur la même commande, comme dans l'exemple suivant :

```
# ccs -h hostname --setfencedaemon post_fail_delay=5 post_join_delay=10
```

Pour obtenir des informations supplémentaires sur la configuration de périphériques fence, reportez-vous à la Section 5.5, « Configuration des périphériques fence ».

5.1.6. Validation de la configuration

Lorsque vous utilisez la commande **ccs** pour créer et modifier le fichier de configuration du cluster, la configuration est automatiquement validée selon le schéma du cluster. À partir de Red Hat Enterprise Linux 6.3, la commande **ccs** valide la configuration selon le schéma du cluster de **/usr/share/cluster/cluster.rng** sur le nœud que spécifierez avec l'option **-h**. Auparavant, la commande **ccs** utilisait toujours le schéma du cluster empaqueté avec la commande **ccs**-même, **/usr/share/ccs/cluster.rng** sur le système local. Lorsque vous utilisez l'option **-f** pour spécifier le système local, la commande **ccs** utilise toujours le schéma du cluster **/usr/share/ccs/cluster.rng** qui était empaqueté avec la commande **ccs**-même sur ce système.

5.2. TÂCHES DE CONFIGURATION

La configuration du logiciel du module complémentaire Red Hat High Availability avec **ccs** comprend les étapes suivantes :

1. S'assurer que **ricci** est en cours d'exécution sur tous les nœuds du cluster. Reportez-vous à la Section 5.3, « Démarrage de **ricci** ».
2. Création d'un cluster. Reportez-vous à la Section 5.4, « Créer un cluster ».
3. Configuration des périphériques fence. Reportez-vous à la Section 5.5, « Configuration des périphériques fence ».
4. Configuration du fencing pour les membres du cluster. Reportez-vous à la Section 5.7, « Configuration du fencing pour les membres du cluster ».
5. Création de domaines de basculements. Reportez-vous à la Section 5.8, « Configurer un domaine de basculement ».
6. Création de ressources. Reportez-vous à la Section 5.9, « Configurer les ressources globales du cluster ».
7. Création de services de clusters. Reportez-vous à la Section 5.10, « Ajouter un service cluster à un cluster ».
8. Configuration d'un disque quorum. Reportez-vous à la Section 5.13, « Configurer un disque Quorum : ».
9. Configuration des propriétés globales du cluster. Reportez-vous à la Section 5.14, « Diverses configurations de clusters ».
10. Propagation du fichier de configuration du cluster à tous les nœuds du cluster. Reportez-vous à la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.3. DÉMARRAGE DE RICCI

Pour pouvoir créer et distribuer des fichiers de configuration de clusters sur les nœuds du cluster, le service **ricci** doit être en cours d'exécution sur chaque nœud. Avant de lancer **ricci**, vous devriez vous assurer que vous avez bien configuré votre système comme suit :

1. Les ports IP des nœuds de votre cluster doivent être activés pour **ricci**. Pour obtenir des informations sur l'activation des ports IP sur les nœuds de clusters, voir la Section 2.3.1, « Activation des ports IP sur des nœuds de clusters ».
2. Le service **ricci** est installé sur tous les nœuds du cluster et possède un mot de passe **ricci**, comme décrit dans la Section 2.13, « Considérations pour **ricci** ».

Une fois que **ricci** a bien été installé et configuré sur chaque nœud, lancez le service **ricci** sur chaque nœud :

```
# service ricci start
Starting ricci:                                            [  OK  ]
```

5.4. CRÉER UN CLUSTER

Cette section décrit comment créer, modifier et supprimer une configuration squelette d'un cluster avec la commande **ccs** sans utiliser de fencing, de domaines de basculement et de services HA. Les sections suivantes décrivent comment configurer ces parties de la configuration.

Pour créer un fichier de configuration squelette d'un cluster, commencez par créer et nommer le cluster, puis ajoutez les nœuds à celui-ci comme le décrit la procédure suivante :

1. Créez un fichier de configuration de cluster sur l'un des nœuds du cluster en exécutant la commande **ccs** et en utilisant le paramètre **-h** pour spécifier le nœud sur lequel créer le fichier ainsi que l'option **createcluster** pour spécifier un nom pour le cluster :

   ```
   ccs -h host --createcluster clustername
   ```

   Par exemple, la commande suivante crée un fichier de configuration sur **node-01.example.com** nommé **mycluster** :

   ```
   ccs -h node-01.example.com --createcluster mycluster
   ```

   Le nom du cluster ne doit pas excéder 15 caractères.

   Si un fichier **cluster.conf** existe déjà sur l'hôte spécifié, l'exécution de cette commande remplacera le fichier existant.

   Si vous souhaitez créer un fichier de configuration de cluster sur votre système local, vous pouvez spécifier l'option **-f** au lieu de l'option **-h**. Pour obtenir des informations sur la création locale du fichier, reportez-vous à la Section 5.1.1, « Créer le fichier de configuration du cluster sur un système local ».

2. Pour configurer les nœuds contenus par le cluster, exécutez la commande suivante sur chaque nœud du cluster :

   ```
   ccs -h host --addnode node
   ```

   Par exemple, les trois commandes suivantes ajoutent les nœuds **node-01.example.com**, **node-02.example.com**, et **node-03.example.com** au fichier de configuration sur **node-01.example.com** :

   ```
   ccs -h node-01.example.com --addnode node-01.example.com
   ccs -h node-01.example.com --addnode node-02.example.com
   ccs -h node-01.example.com --addnode node-03.example.com
   ```

   Pour afficher une liste des nœuds qui ont été configurés pour un cluster, exécutez la commande suivante :

   ```
   ccs -h host --lsnodes
   ```

   L'Exemple 5.1, « Fichier **cluster.conf** après l'ajout de trois nœuds » affiche un fichier de configuration **cluster.conf** une fois que vous avez créé le cluster **mycluster**, celui-ci contient les nœuds **node-01.example.com**, **node-02.example.com** et **node-03.example.com**.

   Exemple 5.1. Fichier cluster.conf après l'ajout de trois nœuds

```
<cluster name="mycluster" config_version="2">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Lorsque vous ajoutez un nœud au cluster, vous pouvez spécifier le nombre de votes auquel le nœud contribue afin de déterminer si le quorum est atteint. Pour ajuster le nombre de vote d'un nœud de cluster, veuillez utiliser la commande suivante :

```
ccs -h host --addnode host --votes votes
```

Lorsque vous ajoutez un nœud, **ccs** assigne à celui-ci un entier unique qui est utilisé en tant qu'identifiant de nœud. Sivous souhaitez spécifier l'identifiant du nœud manuellement lorsque vous créez un nœud, utilisez la commande suivante :

```
ccs -h host --addnode host --nodeid nodeid
```

Pour supprimer un nœud d'un cluster, exécutez la commande suivante :

```
ccs -h host --rmnode node
```

Une fois que vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster avec tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.5. CONFIGURATION DES PÉRIPHÉRIQUES FENCE

La configuration de périphériques fence consiste en la création, la mise à jour et la suppression de périphériques fence du cluster. Vous devez créer et nommer les périphériques fence dans un cluster avant de pouvoir configurer le fencing pour les nœuds dans le cluster. Pour obtenir des informations sur la configuration du fencing pour les nœuds individuels dans le cluster, reportez-vous à la Section 5.7, « Configuration du fencing pour les membres du cluster ».

Avant de configurer vos périphériques fence, vous devriez modifier certaines propriétés du démon fence

sur votre système. Les valeurs que vous configurez pour le démon fence sont généralement des valeurs pour le cluster. Les propriétés générales du fencing du cluster que vous souhaitez modifier sont résumées comme suit :

- L'attribut **post_fail_delay** correspond au nombre de secondes que le démon fence (**fenced**) attend avant de fencer » n n«oeud (un membre du domaine fence) une fois que celui-ci a échoué.
- L'attribut **post-join_delay** correspond au nombre de secondes que le démon Fence (**fenced**) attend avant de clôturer un nœud après que le nœud a rejoint le domaine fence. La valeur par défaut de **post_join_delay** est **6**. Typiquement, le paramètre de délai de **post_join_delay** se situe entre 20 et 30 secondes, mais celui-ci peut varier en fonction de la performance du cluster et du réseau.

Réinitialiser les valeurs des attributs **post_fail_delay** et **post_join_delay** avec l'option **--setfencedaemon** de la commande **ccs**. Remarquez cependant que l'exécution de la commande **ccs --setfencedaemon** remplace toutes les propriétés du démon fence existant ayant été explicitement paramétrées et restaurera leurs valeurs par défaut.

Par exemple, pour configurer une valeur pour l'attribut **post_fail_delay**, exécutez la commande suivante. Cette commande remplacera les valeurs de toutes les autres propriétés existantes du démon fence que vous aurez paramétré avec cette commande et restaurera leurs valeurs par défaut.

```
ccs -h host --setfencedaemon post_fail_delay=value
```

Pour configurer une valeur pour l'attribut **post_join_delay**, exécutez la commande suivante. Cette commande remplacera les valeurs de toutes les autres propriétés existantes du démon fence que vous aurez paramétré avec cette commande et restaurera leurs valeurs par défaut.

```
ccs -h host --setfencedaemon post_join_delay=value
```

Pour configurer une valeur pour l'attribut **post_join_delay** et **post_fail_delay**, veuillez exécuter la commande suivante :

```
ccs -h host --setfencedaemon post_fail_delay=value post_join_delay=value
```

NOTE

Pour obtenir plus d'informations sur les attributs **post_join_delay** et **post_fail_delay** ainsi que sur les propriétés supplémentaires du démon fence que vous pouvez modifier, reportez-vous à la page man fenced(8), au schéma des clusters sur **/usr/share/cluster/cluster.rng** et au schéma annoté sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html**.

Pour configurer un périphérique fence pour un cluster, exécutez la commande suivante :

```
ccs -h host --addfencedev devicename [fencedeviceoptions]
```

Par exemple, pour configurer un périphérique fence APC dans le fichier de configuration sur le nœud du cluster **node1** nommé **myfence** avec l'adresse IP **apc_ip_example**, l'identifiant de connexion **login_example**, et le mot de passe **password_example**, exécutez la commande suivante :

```
ccs -h node1 --addfencedev myfence agent=fence_apc ipaddr=apc_ip_example
login=login_example passwd=password_example
```

L'exemple suivant montre la section **fencedevices** du fichier de configuration **cluster.conf** une fois ce périphérique fence APC ajouté :

```
<fencedevices>
      <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="myfence" passwd="password_example"/>
</fencedevices>
```

Lors de la configuration de périphériques fence pour un cluster, vous trouverez utile de pouvoir afficher une liste des périphériques disponibles pour votre cluster et les options qui leurs sont disponibles. Vous trouverez tout aussi utile la présence d'une liste des périphériques fence actuellement configurés pour votre cluster. Pour obtenir des informations sur l'utilisation de la commande **ccs** pour imprimer une liste des périphériques fence et options disponibles ou pour imprimer une liste des périphériques fence actuellement configurés pour votre cluster, reportez-vous à la Section 5.6, « Répertorier les périphériques fence et les options de périphériques fence ».

Pour supprimer un périphérique fence de la configuration de votre cluster, exécutez la commande suivante :

```
ccs -h host --rmfencedev fence_device_name
```

Par exemple, pour supprimer un périphérique fence que vous auriez nommé **myfence** depuis le fichier de configuration du cluster du nœud de cluster **node1**, exécutez la commande suivante :

```
ccs -h node1 --rmfencedev myfence
```

Si vous devez modifier les attributs d'un périphérique fence que vous avez déjà configuré, vous devrez d'abord supprimer ce périphérique fence puis l'ajouter à nouveau avec les attributs modifiés.

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.6. RÉPERTORIER LES PÉRIPHÉRIQUES FENCE ET LES OPTIONS DE PÉRIPHÉRIQUES FENCE

Vous pouvez utiliser la commande **ccs** pour imprimer une liste des périphériques fence disponibles et pour imprimer une liste des options pour chaque type fence disponible. Vous pouvez aussi utiliser la commande **ccs** pour imprimer une liste des périphériques fence actuellement configurés pour votre cluster.

Pour imprimer une liste des périphériques fence actuellement disponibles pour votre cluster, exécutez la commande suivante :

```
ccs -h host --lsfenceopts
```

Par exemple, la commande suivante répertorie les périphériques fence disponibles sur le nœud **node1** du cluster, affichant un exemple de sortie.

```
[root@ask-03 ~]# ccs -h node1 --lsfenceopts
fence_rps10 - RPS10 Serial Switch
fence_vixel - No description available
fence_egenera - No description available
fence_xcat - No description available
fence_na - Node Assassin
fence_apc - Fence agent for APC over telnet/ssh
fence_apc_snmp - Fence agent for APC over SNMP
fence_bladecenter - Fence agent for IBM BladeCenter
fence_bladecenter_snmp - Fence agent for IBM BladeCenter over SNMP
fence_cisco_mds - Fence agent for Cisco MDS
fence_cisco_ucs - Fence agent for Cisco UCS
fence_drac5 - Fence agent for Dell DRAC CMC/5
fence_eps - Fence agent for ePowerSwitch
fence_ibmblade - Fence agent for IBM BladeCenter over SNMP
fence_ifmib - Fence agent for IF MIB
fence_ilo - Fence agent for HP iLO
fence_ilo_mp - Fence agent for HP iLO MP
fence_intelmodular - Fence agent for Intel Modular
fence_ipmilan - Fence agent for IPMI over LAN
fence_kdump - Fence agent for use with kdump
fence_rhevm - Fence agent for RHEV-M REST API
fence_rsa - Fence agent for IBM RSA
fence_sanbox2 - Fence agent for QLogic SANBox2 FC switches
fence_scsi - fence agent for SCSI-3 persistent reservations
fence_virsh - Fence agent for virsh
fence_virt - Fence agent for virtual machines
fence_vmware - Fence agent for VMware
fence_vmware_soap - Fence agent for VMware over SOAP API
fence_wti - Fence agent for WTI
fence_xvm - Fence agent for virtual machines
```

Pour imprimer une liste des options que vous pouvez spécifier pour un type fence particulier, exécutez la commande suivante :

```
ccs -h host --lsfenceopts fence_type
```

Par exemple, la commande suivante répertorie les options fence pour l'agent fence **`fence_wti`**.

```
[root@ask-03 ~]# ccs -h node1 --lsfenceopts fence_wti
fence_wti - Fence agent for WTI
  Required Options:
  Optional Options:
    option: No description available
    action: Fencing Action
    ipaddr: IP Address or Hostname
    login: Login Name
    passwd: Login password or passphrase
    passwd_script: Script to retrieve password
    cmd_prompt: Force command prompt
    secure: SSH connection
    identity_file: Identity file for ssh
    port: Physical plug number or name of virtual machine
    inet4_only: Forces agent to use IPv4 addresses only
    inet6_only: Forces agent to use IPv6 addresses only
```

```
ipport: TCP port to use for connection with device
verbose: Verbose mode
debug: Write debug information to given file
version: Display version information and exit
help: Display help and exit
separator: Separator for CSV created by operation list
power_timeout: Test X seconds for status change after ON/OFF
shell_timeout: Wait X seconds for cmd prompt after issuing command
login_timeout: Wait X seconds for cmd prompt after login
power_wait: Wait X seconds after issuing ON/OFF
delay: Wait X seconds before fencing is started
retry_on: Count of attempts to retry power on
```

Pour imprimer une liste des périphériques fence actuellement configurés pour votre cluster, exécutez la commande suivante :

```
ccs -h host --lsfencedev
```

5.7. CONFIGURATION DU FENCING POUR LES MEMBRES DU CLUSTER

Une fois les étapes initiales de création du cluster et des périphériques fence terminées, vous devrez configurer le fencing pour les nœuds du cluster. Pour configurer le fencing pour les nœuds après la création d'un nouveau cluster et la configuration des périphériques fence du cluster, suivez les étapes de cette section. Remarquez que vous devez configurer le fencing pour chaque nœud du cluster.

Cette section documente les procédures suivantes :

- Section 5.7.1, « Configurer un périphérique fence unique basé sur l'alimentation pour un nœud »
- Section 5.7.2, « Configurer un périphérique fence unique basé sur stockage pour un nœud »
- Section 5.7.3, « Configurer un périphérique fence de sauvegarde »
- Section 5.7.4, « Configurer un nœud avec une alimentation redondante »
- Section 5.7.5, « Supprimer les méthodes et instances fence »

5.7.1. Configurer un périphérique fence unique basé sur l'alimentation pour un nœud

Uilisez la procédure suivante pour configurer un nœud avec un périphérique fence unique basé sur l'alimentation qui utilise un périphérique fence nommé **apc** utilisant l'agent de fencing **fence_apc**.

1. Ajoutez une méthode fence pour le nœud, en fournissant un nom pour la méthode fence.

   ```
   ccs -h host --addmethod method node
   ```

   Par exemple, pour configurer une méthode fence nommée **APC** pour le nœud **node-01.example.com** dans le fichier de configuration sur le nœud du cluster **node-01.example.com**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addmethod APC node01.example.com
   ```

2. Ajoutez une instance fence à la méthode. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode, et toute autre option de cette méthode qui serait spécifique à ce nœud :

```
ccs -h host --addfenceinst fencedevicename node method [options]
```

Par exemple, pour configurer une instance fence dans le fichier de configuration sur le nœud du cluster **node-01.example.com** qui utilise le port d'alimentation 1 de l'interrupteur APC sur le périphérique fence nommé **apc** pour clore le nœud du cluster **node-01.example.com** à l'aide de la méthode nommée **APC**, exécutez la commande suivante :

```
ccs -h node01.example.com --addfenceinst apc node01.example.com APC
port=1
```

Vous devrez ajouter une méthode fence pour chaque nœud du cluster. Les commande suivantes configurent une méthode fence pour chaque nœud avec la méthode nommée **APC**. Le périphérique pour la méthode fence spécifie **apc** comme nom de périphérique, qui est un périphérique précédemment configuré avec l'option **--addfencedev**, comme le décrit la Section 5.5, « Configuration des périphériques fence ». Chaque nœud est configuré avec un numéro de port d'alimentation de l'interrupteur APC unique : le numéro de port de **node-01.example.com** est **1**, le numéro de port de **node-02.example.com** est **2**, et le numéro de port de **node-03.example.com** est **3**.

```
ccs -h node01.example.com --addmethod APC node01.example.com
ccs -h node01.example.com --addmethod APC node02.example.com
ccs -h node01.example.com --addmethod APC node03.example.com
ccs -h node01.example.com --addfenceinst apc node01.example.com APC port=1
ccs -h node01.example.com --addfenceinst apc node02.example.com APC port=2
ccs -h node01.example.com --addfenceinst apc node03.example.com APC port=3
```

L'Exemple 5.2, « **cluster.conf** après avoir ajouté des méthodes fence basées sur l'alimentation » montrera un fichier de configuration **cluster.conf** une fois que vous aurez ajouté ces méthodes et instances de fencing à chaque nœud du cluster.

Exemple 5.2. cluster.conf après avoir ajouté des méthodes fence basées sur l'alimentation

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
```

```
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
          <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.7.2. Configurer un périphérique fence unique basé sur stockage pour un nœud

Lors de l'utilisation de méthodes fence sans alimentation (de l'anglais, « non-power fencing methods ». Par exemple, le fencing de réseaux SAN ou de stockage) pour clôturer un nœud, vous devez configurer *unfencing* pour le périphérique fence. Cela vous permet de vous assurer qu'un nœud clôturé ne sera pas ré-activé avant que le nœud ne soit redémarré. Lorsque vous configurez unfencing pour un nœud, vous spécifiez un périphérique qui met en miroir le périphérique fence correspondant que vous avez configuré pour le nœud avec l'addition notable de l'action explicite de **on** ou de **enable**.

Pour obtenir plus d'informations sur le processus pour unfence un nœud, reportez-vous à la page man **fence_node**(8).

Utilisez la procédure suivante pour configurer un nœud avec un périphérique fence unique basé sur stockage qui utilise un périphérique fence nommé **sanswitch1** utilisant l'agent de fencing **fence_sanbox2**.

1. Ajoutez une méthode fence pour le nœud, en fournissant un nom pour la méthode fence.

   ```
   ccs -h host --addmethod method node
   ```

   Par exemple, pour configurer une méthode fence nommée **SAN** pour le nœud **node-01.example.com** dans le fichier de configuration du nœud du cluster **node-01.example.com**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addmethod SAN  node01.example.com
   ```

2. Ajoutez une instance fence à la méthode. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode, et toute autre option de cette méthode qui serait spécifique à ce nœud :

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   ```

   Par exemple, pour configurer une instance fence dans le fichier de configuration du nœud du cluster **node-01.example.com** qui utilise le port d'alimentation 11 de l'interrupteur SAN sur le

périphérique fence nommé **sanswitch1** afin qu'il clôture le nœud du cluster **node-01.example.com** à l'aide de la méthode nommée **SAN**, exécutez la commande suivante :

```
ccs -h node01.example.com --addfenceinst sanswitch1
node01.example.com SAN port=11
```

3. Pour configurer unfencing pour le périphérique fence basé sur stockage de ce nœud, exécutez la commande suivante :

```
ccs -h host --addunfence fencedevicename node action=on|off
```

Vous devrez ajouter une méthode fence pour chaque nœud dans le cluster. Les commandes suivantes configurent une méthode fence pour chaque nœud avec la méthode nommée **SAN**. Le périphérique de la méthode fence spécifie **sanswitch** comme nom de périphérique, qui est un périphérique précédemment configuré avec l'option --addfencedev, comme le décrit la Section 5.5, « Configuration des périphériques fence ». Chaque nœud est configuré avec un numéro de port physique SAN unique : le numéro de port de **node-01.example.com** est **11**, le numéro de port de **node-02.example.com** est **12**, et le numéro de port de **node-03.example.com** est **13**.

```
ccs -h node01.example.com --addmethod SAN node01.example.com
ccs -h node01.example.com --addmethod SAN node02.example.com
ccs -h node01.example.com --addmethod SAN node03.example.com
ccs -h node01.example.com --addfenceinst sanswitch1 node01.example.com SAN
port=11
ccs -h node01.example.com --addfenceinst sanswitch1 node02.example.com SAN
port=12
ccs -h node01.example.com --addfenceinst sanswitch1 node03.example.com SAN
port=13
ccs -h node01.example.com --addunfence sanswitch1 node01.example.com
port=11 action=on
ccs -h node01.example.com --addunfence sanswitch1 node02.example.com
port=12 action=on
ccs -h node01.example.com --addunfence sanswitch1 node03.example.com
port=13 action=on
```

L'Exemple 5.3, « **cluster.conf** après avoir ajouté des méthodes fence basé sur stockage » montre un fichier de configuration **cluster.conf** après avoir ajouté des méthodes de fencing, des instances de fencing et « l'unfencing » à chaque nœud du cluster.

Exemple 5.3. cluster.conf après avoir ajouté des méthodes fence basé sur stockage

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="SAN">
       <device name="sanswitch1" port="11"/>
            </method>
         </fence>
         <unfence>
            <device name="sanswitch1" port="11" action="on"/>
         </unfence>
     </clusternode>
```

```
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
             <method name="SAN">
        <device name="sanswitch1" port="12"/>
              </method>
          </fence>
          <unfence>
              <device name="sanswitch1" port="12" action="on"/>
          </unfence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
             <method name="SAN">
        <device name="sanswitch1" port="13"/>
              </method>
          </fence>
          <unfence>
              <device name="sanswitch1" port="13" action="on"/>
          </unfence>
      </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.7.3. Configurer un périphérique fence de sauvegarde

Vous pouvez définir de multiples méthodes de fencing pour un nœud. Si le fencing échoue lors de l'utilisation de la première méthode, le système tentera de clôturer le nœud à l'aide de la seconde méthode, puis par toute méthode supplémentaire configurée. Pour configurer une méthode de fencing de sauvegarde pour un nœud, configurez deux méthodes pour un nœud tout en configurant une instance fence pour chaque méthode.

NOTE

L'ordre dans lequel le système utilisera les méthodes de fencing que vous avez configuré suit l'ordre dans le fichier de configuration du cluster. La première méthode configurée avec la commande **ccs** est la méthode de fencing primaire et la seconde méthode configurée est la méthode de fencing de sauvegarde. Pour changer l'ordre, vous pouvez supprimer la méthode de fencing primaire du fichier de configuration, puis ajoutez cette méthode à nouveau.

Remarquez qu'à tout moment, il vous est possible d'imprimer une liste des méthodes et instances fence actuellement configurées pour un nœud en exécutant la commande suivante. Si vous ne spécifiez pas

de nœud, cette commande répertoriera les méthodes et instances fence actuellement configurées pour tous les nœuds.

```
ccs -h host --lsfenceinst [node]
```

Utilisez la procédure suivante pour configurer un nœud avec une méthode de fencing primaire qui utilise un périphérique fence nommé **apc** qui utilise l'agent de fencing **fence_apc** et un périphérique de fencing de sauvegarde utilisant un périphérique fence nommé **sanswitch1** qui utilise l'agent de fencing **fence_sanbox2**. Comme le périphérique **sanswitch1** est un agent de fencing basé sur stockage, vous devrez aussi configurer « l'unfencing » pour ce périphérique.

1. Ajouter une méthode fence primaire pour le nœud, en fournissant un nom pour la méthode fence.

   ```
   ccs -h host --addmethod method node
   ```

   Par exemple, pour configurer une méthode fence nommée **APC** comme méthode primaire pour le nœud **node-01.example.com** dans le fichier de configuration sur le nœud du cluster **node-01.example.com**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addmethod APC node01.example.com
   ```

2. Ajoutez une instance fence pour la méthode primaire. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode et toutes les options de cette méthode qui sont spécifiques à ce nœud :

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   ```

   Par exemple, pour configurer une instance fence dans le fichier de configuration sur le nœud du cluster **node-01.example.com** qui utilise le port d'alimentation 1 de l'interrupteur APC sur le périphérique fence nommé **apc** pour clore le nœud du cluster **node-01.example.com** à l'aide de la méthode nommée **APC**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addfenceinst apc node01.example.com APC
   port=1
   ```

3. Ajoutez une méthode fence de sauvegarde pour ce nœud, tout en fournissant un nom pour la méthode fence.

   ```
   ccs -h host --addmethod method node
   ```

   Par exemple, pour configurer une méthode fence de sauvegarde nommée **SAN** pour le nœud **node-01.example.com** dans le fichier de configuration sur le nœud du cluster **node-01.example.com**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addmethod SAN  node01.example.com
   ```

4. Ajoutez une instance fence pour la méthode de sauvegarde. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode et toutes les options de cette méthode qui sont spécifiques à ce nœud :

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   ```

Par exemple, pour configurer une instance fence dans le fichier de configuration du nœud du cluster **node-01.example.com** qui utilise le port d'alimentation 11 de l'interrupteur SAN sur le périphérique fence nommé **sanswitch1** afin qu'il clôture le nœud du cluster **node-01.example.com** à l'aide de la méthode nommée **SAN**, exécutez la commande suivante :

```
ccs -h node01.example.com --addfenceinst sanswitch1
node01.example.com SAN port=11
```

5. Comme le périphérique **sanswitch1** est un périphérique basé sur stockage, vous devez configurer unfencing pour celui-ci.

```
ccs -h node01.example.com --addunfence sanswitch1 node01.example.com
port=11 action=on
```

Vous pouvez continuer à ajouter des méthodes de fencing selon vos besoins.

Cette procédure configure un périphérique fence et un périphérique fence de sauvegarde pour un nœud dans le cluster. Vous devrez aussi configurer le fencing pour les autres nœuds.

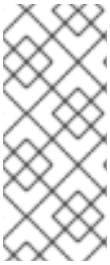
L'Exemple 5.4, « **cluster.conf** après avoir ajouté des méthodes fence de sauvegarde » montre un fichier de configuration **cluster.conf** après avoir ajouté une méthode de fencing primaire basé sur l'alimentation et une méthode de fencing de sauvegarde basé sur stockage à chaque nœud du cluster.

Exemple 5.4. cluster.conf après avoir ajouté des méthodes fence de sauvegarde

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
            <method name="SAN">
       <device name="sanswitch1" port="11"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="11" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
            <method name="SAN">
       <device name="sanswitch1" port="12"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="12" action="on"/>
         </unfence
```

```
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
             <method name="APC">
               <device name="apc" port="3"/>
             </method>
             <method name="SAN">
        <device name="sanswitch1" port="13"/>
              </method>
          </fence>
          <unfence>
              <device name="sanswitch1" port="13" action="on"/>
          </unfence
      </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».



NOTE

L'ordre dans lequel le système utilisera les méthodes de fencing que vous avez configuré suit l'ordre défini dans le fichier de configuration. La première méthode configurée est la méthode de fencing primaire, la seconde méthode configurée est la méthode de fencing de sauvegarde. Pour modifier cet ordre, vous pouvez supprimer la méthode de fencing primaire du fichier de configuration, puis ajoutez-la à nouveau.

5.7.4. Configurer un nœud avec une alimentation redondante

Si votre cluster est configuré avec une alimentation redondante pour vos nœuds, vous devez vous assurer de configurer le fencing de manière à ce que vos nœuds s'éteignent complètement lorsqu'ils doivent être clôturés. Si vous configurez chaque alimentation électrique comme une méthode fence séparée, alors chacune de ces alimentations sera clôturée séparément ; la seconde alimentation électrique permettra au système de continuer à s'exécuter lorsque la première alimentation est clôturée et le système ne sera donc pas complètement clôturé. Pour configurer un système avec un système d'alimentation électrique duel, vous devrez configurer vos périphériques fence de manière à ce que les deux sources d'alimentation soient éteintes et le système complètement arrêté. Ceci requiert que vous configuriez chaque périphérique avec un attribut **`action`** sur **on**.

Pour configurer le fencing pour un nœud à système d'alimentation électrique duel, suivez les étapes de cette section.

1. Avant de pouvoir configurer le fencing pour un nœud avec une alimentation redondante, vous devez configurer chaque interrupteur de l'alimentation en tant que périphérique fence pour le cluster. Pour obtenir des informations sur la configuration des périphériques fence, voir la Section 5.5, « Configuration des périphériques fence ».

   Pour imprimer une liste des périphériques fence actuellement configurés pour votre cluster, exécutez la commande suivante :

   ```
   ccs -h host --lsfencedev
   ```

2. Ajoutez une méthode fence pour le nœud, en fournissant un nom pour la méthode fence.

   ```
   ccs -h host --addmethod method node
   ```

   Par exemple, pour configurer une méthode nommée **APC-dual** pour le nœud **node-01.example.com** dans le fichier de configuration du nœud du cluster **node-01.example.com**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addmethod APC-dual node01.example.com
   ```

3. Ajoutez une instance fence pour la première alimentation électrique à la méthode fence. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel cette instance s'applique, le nom de la méthode, et toutes les options de cette méthode qui sont spécifiques à ce nœud. À ce moment, configurez l'attribut **action** sur **off**.

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   action=off
   ```

   Par exemple, pour configurer une instance fence dans le fichier de configuration du nœud du cluster **node-01.example.com**, qui utilise le port d'alimentation 1 de l'interrupteur APC du périphérique fence nommé **apc1**, pour clôturer le nœud du cluster **node-01.example.com**, qui utilise la méthode nommée **APC-dual**, et pour paramétrer l'attribut **action** sur **off**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addfenceinst apc1 node01.example.com
   APC-dual port=1 action=off
   ```

4. Ajoutez une instance fence pour la seconde alimentation à la méthode fence. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode et toutes les options de cette méthode qui sont spécifiques à ce nœud. À ce moment, configurez aussi l'attribut **action** sur **off** pour cette instance :

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   action=off
   ```

   Par exemple, pour configurer une seconde instance fence dans le fichier de configuration du nœud du cluster **node-01.example.com**, qui utilise le port d'alimentation 1 de l'interrupteur APC du périphérique fence nommé **apc2**, pour clôturer le nœud du cluster **node-01.example.com**, qui utilise la même méthode que pour la première instance nommée **APC-dual**, et pour paramétrer l'attribut **action** sur **off**, exécutez la commande suivante :

```
ccs -h node01.example.com --addfenceinst apc2 node01.example.com
APC-dual port=1 action=off
```

5. À ce moment, ajoutez une autre instance fence pour la première alimentation à la méthode fence, tout en configurant l'attribut **action** sur **on**. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode et toutes les options de cette méthode qui sont spécifiques à ce nœud, puis spécifiez l'attribut **action** comme étant **on** :

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   action=on
   ```

   Par exemple, pour configurer une instance fence dans le fichier de configuration du nœud du cluster **node-01.example.com**, qui utilise le port d'alimentation 1 de l'interrupteur APC du périphérique fence nommé **apc1**, pour clôturer le nœud du cluster **node-01.example.com**, qui utilise la méthode nommée **APC-dual**, et pour paramétrer l'attribut **action** sur **on**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addfenceinst apc1 node01.example.com
   APC-dual port=1 action=on
   ```

6. Ajoutez une autre instance fence pour la seconde alimentation à la méthode fence, tout en spécifiant l'attribut **action** de cette instance sur **on**. Vous devez spécifier le périphérique fence à utiliser pour le nœud, le nœud auquel s'applique cette instance, le nom de la méthode et toutes les options de cette méthode qui sont spécifiques à ce nœud, ainsi que spécifier l'attribut **action** sur **on** :

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   action=on
   ```

   Par exemple, pour configurer une seconde instance fence dans le fichier de configuration du nœud du cluster **node-01.example.com**, qui utilise le port d'alimentation 1 de l'interrupteur APC du périphérique fence nommé **apc2**, pour clôturer le nœud du cluster **node-01.example.com**, qui utilise la même méthode que pour la première instance nommée **APC-dual**, et pour paramétrer l'attribut **action** sur **on**, exécutez la commande suivante :

   ```
   ccs -h node01.example.com --addfenceinst apc2 node01.example.com
   APC-dual port=1 action=on
   ```

L'Exemple 5.5, « **cluster.conf** après avoir ajouté le fencing à double alimentation » montre un fichier de configuration **cluster.conf** après avoir ajouté le fencing sur deux alimentations électriques pour chaque nœud dans un cluster :

Exemple 5.5. cluster.conf après avoir ajouté le fencing à double alimentation

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="1"action="off"/>
```

```
                <device name="apc2" port="1"action="off"/>
                <device name="apc1" port="1"action="on"/>
                <device name="apc2" port="1"action="on"/>
               </method>
            </fence>
        </clusternode>
        <clusternode name="node-02.example.com" nodeid="2">
            <fence>
               <method name="APC-dual">
                 <device name="apc1" port="2"action="off"/>
                 <device name="apc2" port="2"action="off"/>
                 <device name="apc1" port="2"action="on"/>
                 <device name="apc2" port="2"action="on"/>
               </method>
            </fence>
        </clusternode>
        <clusternode name="node-03.example.com" nodeid="3">
            <fence>
               <method name="APC-dual">
                 <device name="apc1" port="3"action="off"/>
                 <device name="apc2" port="3"action="off"/>
                 <device name="apc1" port="3"action="on"/>
                 <device name="apc2" port="3"action="on"/>
               </method>
            </fence>
        </clusternode>
    </clusternodes>
    <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc1" passwd="password_example"/>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc2" passwd="password_example"/>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.7.5. Supprimer les méthodes et instances fence

Pour supprimer une méthode fence de la configuration de votre cluster, exécutez la commande suivante :

```
ccs -h host --rmmethod method node
```

Par exemple, pour supprimer une méthode fence nommée **APC** que vous avez configuré pour **node01.example.com** depuis le fichier de configuration sur le nœud du cluster **node01.example.com**, exécutez la commande suivante :

```
ccs -h node01.example.com  --rmmethod APC node01.example.com
```

Pour supprimer toutes les instances fence d'un périphérique fence d'une méthode fence, exécutez la commande suivante :

```
ccs -h host --rmfenceinst fencedevicename node method
```

Par exemple, pour supprimer toutes les instances du périphérique fence nommé **apc1** de la méthode nommée **APC-dual** configurée pour **node01.example.com** du fichier de configuration du cluster du nœud du cluster **node01.example.com**, exécutez la commande suivante :

```
ccs -h node01.example.com --rmfenceinst apc1 node01.example.com APC-dual
```

5.8. CONFIGURER UN DOMAINE DE BASCULEMENT

Un domaine de basculement est un sous-ensemble de nœuds d'un cluster nommé qui sont capables d'exécuter un service cluster dans le cas d'un échec de nœud. Un domaine de basculement peut posséder les caractéristiques suivantes :

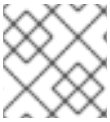
- Unrestricted — Ceci vous permet de spécifier qu'un sous-ensemble de membres est préféré, mais qu'un service cluster assigné à ce domaine peut s'exécuter sur n'importe quel membre disponible.
- Restricted — Ceci vous permet de restreindre les membres pouvant exécuter un service cluster en particulier. Si aucun des membres dans un domaine de basculement restricted n'est disponible, le service cluster ne pourra pas être lancé (manuellement ou par le logiciel du cluster).
- Unordered — Lorsqu'un service cluster est assigné à un domaine de basculement unordered, le membre sur lequel le service cluster est exécuté est choisi parmi les membres disponibles du domaine de basculement sans ordre de priorité.
- Ordered — Ceci vous permet de spécifier un ordre de préférence parmi les membres d'un domaine de basculement. Le membre le plus haut dans la liste est le préféré, suivi par le second membre dans la liste, et ainsi de suite.
- Failback — Ceci vous permet de spécifier si un service dans le domaine de basculement devrait être restauré sur le nœud sur lequel il était initialement exécuté avant que ce nœud tombe en panne. La configuration de cette caractéristique est utile dans des circonstances où un nœud tombe en panne de manière répétitive et fait partie d'un domaine de basculement ordered. Dans ces circonstances, si un nœud est le nœud préféré dans un domaine de basculement, il est possible qu'un service tombe en panne puis se restaure de manière répétitive entre le nœud préféré et un autre nœud, affectant sévèrement la performance.

NOTE

La caractéristique failback est uniquement applicable si le basculement ordered est configuré.

NOTE

Modifier la configuration d'un domaine de basculement n'a aucun effet sur les services en cours d'exécution.

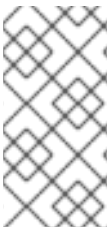


NOTE

Les domaines de basculement ne sont *pas* requis pour les opérations.

Par défaut, les domaines de basculement sont unrestricted et unordered.

Dans un cluster possédant plusieurs membres, l'utilisation d'un domaine de basculement restricted peut minimiser le travail de paramétrage du cluster pour qu'il exécute un service cluster (comme **`httpd`**), qui requiert que vous paramétriez la configuration de manière identique sur tous les membres exécutant le service cluster. Au lieu de paramétrer le cluster entier afin qu'il exécute le service cluster, il vous est possible de paramétrer uniquement les membres dans le domaine de basculement restricted que vous associerez au service cluster.



NOTE

Pour configurer un membre préféré, vous pouvez créer un domaine de basculement unrestricted comprenant uniquement un membre du cluster. Faire ceci cause au service cluster de s'exécuter sur ce membre du cluster en premier (le membre préféré), mais permet au service cluster de basculer sur tout autre membre.

Pour configurer un domaine de basculement, effectuez la procédure suivante :

1. Pour ajouter un domaine de basculement, exécutez la commande suivante :

```
ccs -h host --addfailoverdomain name [restricted] [ordered]
[nofailback]
```

NOTE

Le nom doit être suffisamment descriptif pour distinguer son but par rapport aux autres noms utilisés dans votre cluster.

Par exemple, la commande suivante configure un domaine de basculement nommé **`example_pri`** sur **`node-01.example.com`**, qui est unrestricted, ordered, et permet le failback :

```
ccs -h node-01.example.com --addfailoverdomain example_pri ordered
```

2. Pour ajouter un nœud au domaine de basculement, exécutez la commande suivante :

```
ccs -h host --addfailoverdomainnode failoverdomain node priority
```

Par exemple, pour configurer le domaine de basculement **`example_pri`** du fichier de configuration sur **`node-01.example.com`** afin qu'il contienne **`node-01.example.com`** avec une priorité de 1, **`node-02.example.com`** avec une priorité de 2 et **`node-03.example.com`** avec une priorité de 3, exécutez les commandes suivantes :

```
ccs -h node-01.example.com --addfailoverdomainnode example_pri node-
01.example.com 1
ccs -h node-01.example.com --addfailoverdomainnode example_pri node-
02.example.com 2
ccs -h node-01.example.com --addfailoverdomainnode example_pri node-
```

```
03.example.com 3
```

Vous pouvez répertorier tous les domaines de basculement et les nœuds de domaines de basculement configurés dans un cluster avec la commande suivante :

```
ccs -h host --lsfailoverdomain
```

Pour supprimer un domaine de basculement, exécutez la commande suivante :

```
ccs -h host --rmfailoverdomain name
```

Pour supprimer un nœud d'un domaine de basculement, exécutez la commande suivante :

```
ccs -h host --rmfailoverdomainnode failoverdomain node
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.9. CONFIGURER LES RESSOURCES GLOBALES DU CLUSTER

Vous povuez configurer deux types de ressources :

- Global — Les ressources disponibles à tous les services dans le cluster.
- Service-specific — Les ressources disponibles à un seul service.

Pour afficher une liste des ressources et services actuellement configurés dans le cluster, exécutez la commande suivante :

```
ccs -h host --lsservices
```

Pour ajouter une ressource globale du cluster, exécutez la commande suivante. Vous pouvez ajouter une ressource locale à un service en particulier lorsque vous configurez ce service, comme le décrit la Section 5.10, « Ajouter un service cluster à un cluster ».

```
ccs -h host --addresource resourcetype [resource options]
```

Par exemple, la commande suivante ajoute une ressource de système de fichier global au fichier de configuration du cluster sur **node01.example.com**. Le nom de la ressource est **web_fs**, le périphérique du système de fichier est **/dev/sdd2**, le point de montage du système de fichiers est **/var/www**, et le type de système de fichiers est **ext3**.

```
ccs -h node01.example.com --addresource fs name=web_fs device=/dev/sdd2
mountpoint=/var/www fstype=ext3
```

Pour obtenir des informations sur les options et les types de ressources, voir l'Annexe B, *Paramètres des ressources HA*.

Pour supprimer une ressource globale, exécutez la commande suivante :

```
ccs -h host --rmresource resourcetype [resource options]
```

Si vous devez modifier les paramètres d'une ressource globale existante, vous pouvez supprimer la ressource et la configurer à nouveau.

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.10. AJOUTER UN SERVICE CLUSTER À UN CLUSTER

Pour configurer un service cluster dans un cluster, procédez aux étapes suivantes :

1. Ajoutez un service au cluster avec la commande suivante :

```
ccs -h host --addservice servicename [service options]
```

NOTE

Utilisez un nom descriptif qui distingue clairement le service des autres services dans le cluster.

Lorsque vous ajoutez un service à la configuration du cluster, vous devez configurer les attributs suivants :

- **autostart** — Spécifie s'il faut démarrer le service automatiquement lorsque le cluster démarre. Veuillez utiliser « 1 » pour activer et « 0 » pour désactiver, le service est activé par défaut.
- **domain** — Spécifie un domaine de basculement (s'il est requis).
- **exclusive** — Spécifie une politique où le service s'exécute uniquement sur des nœuds sur lesquels aucun autre service ne s'exécute.
- **recovery** — Spécifie une stratégie de récupération pour le service. Les options pour le service sont « relocate » (déplacer), « restart » (redémarrer), « disable » (désactiver), ou « restart-disable » (redémarrer-désactiver). La stratégie de récupération « restart » (redémarrer) indique que le système devrait tenter de redémarrer le service en échec avant de tenter de déplacer le service vers un autre nœud. La stratégie « relocate » indique que le système devrait tenter de redémarrer le service sur un autre nœud. La stratégie « disable » (désactiver) indique que le système devrait désactiver le groupe de ressources si un composant échoue. La stratégie « restart-disable » (redémarrer-désactiver) devrait tenter de redémarrer le service au même endroit s'il échoue, mais que si le redémarrage du service échoue, le service sera désactivé au lieu d'être déplacé vers un autre hôte dans le cluster.

  Si vous sélectionnez **Restart** ou **Restart-Disable** en tant que politique de récupération pour le service, vous pourrez spécifier le nombre maximum d'échecs de redémarrage avant le déplacement ou la désactivation du service. Vous pouvez aussi spécifier (en secondes) à partir de combien de temps il ne faudra plus effectuer de redémarrages.

Par exemple, pour ajouter un service au fichier de configuration sur le nœud du cluster **node-01.example.com** nommé **example_apache** qui utilise le domaine de basculement **example_pri**, et possède la politique **relocate**, exécutez la commande suivante :

```
ccs -h node-01.example.com --addservice example_apache
domain=example_pri recovery=relocate
```

Lors de la configuration de services pour un cluster, vous trouverez utile de pouvoir afficher une liste des services disponibles pour votre cluster ainsi que les options qui leurs sont disponibles. Pour obtenir des informations sur l'utilisation de la commande **ccs** pour imprimer une liste des services et options disponibles, reportez-vous à la Section 5.11, « Répertorier les services cluster disponibles ».

2. Ajoutez des ressources au service avec la commande suivante :

```
ccs -h host --addsubservice servicename subservice [service options]
```

Selon le type de ressources que vous souhaitez utiliser, remplissez le service avec des resources globales ou spécifiques au service. Pour ajouter une ressource globale, utilisez l'option **--addsubservice** de **ccs**. Par exemple, pour ajouter la ressource globale d'un système de fichiers nommée **web_fs** au service nommé **example_apache** du fichier de configuration du cluster sur **node-01.example.com**, exécutez la commande suivante :

```
ccs -h node01.example.com --addsubservice example_apache fs
ref=web_fs
```

Pour ajouter une ressource spécifique au service, vous devez spécifier toutes les options du service. Par exemple, si vous n'avez pas défini **web_fs** en tant que service global au préalable, vous pourriez l'ajouter en tant que ressource spécifique au service avec la commande suivante :

```
ccs -h node01.example.com --addsubservice example_apache fs
name=web_fs device=/dev/sdd2 mountpoint=/var/www fstype=ext3
```

3. Pour ajouter un service enfant au service, vous pouvez aussi utiliser l'option **--addsubservice** à la commande **ccs** tout en spécifiant les options du service.

Si vous devez ajouter des services dans une structure arborescente de dépendances, utilisez le caractère des deux-points (":") pour séparer les éléments et des parenthèses pour identifier les sous-services du même type. L'exemple suivant ajoute un troisième service **nfsclient** en tant que sous-service d'un service **nfsclient**, qui lui-même est un sous-service d'un service **nfsclient**, qui est un sous-service du service nommé **service_a** :

```
ccs -h node01.example.com --addsubservice service_a
nfsclient[1]:nfsclient[2]:nfsclient
```

NOTE

Si vous êtes en train d'ajouter une ressource du service Samba, ajoutez-la directement au service, et *non pas* en tant qu'enfant d'une autre ressource.

NOTE

Pour vérifier l'existence de la ressource du service IP utilisée dans un service cluster, vous pouvez utiliser la commande **/sbin/ip addr show** sur un nœud de cluster (plutôt que la commande obsolète **ifconfig**). La sortie suivante montre la commande **/sbin/ip addr show** exécutée sur un nœud qui exécute un service cluster :

```
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1356 qdisc pfifo_fast
qlen 1000
    link/ether 00:05:5d:9a:d8:91 brd ff:ff:ff:ff:ff:ff
    inet 10.11.4.31/22 brd 10.11.7.255 scope global eth0
    inet6 fe80::205:5dff:fe9a:d891/64 scope link
    inet 10.11.4.240/22 scope global secondary eth0
       valid_lft forever preferred_lft forever
```

Pour supprimer un service et tous ses sous-services, exécutez la commande suivante :

```
ccs -h host --rmservice servicename
```

Pour supprimer un sous-service, exécutez la commande suivante :

```
ccs -h host --rmsubservice servicename subservice [service options]
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.11. RÉPERTORIER LES SERVICES CLUSTER DISPONIBLES

vous pouvez utiliser la commande **ccs** pour imprimer une liste des services disponibles à un cluster. Vous pouvez aussi utiliser la commande **ccs** pour imprimer une liste des options pouvant être spécifiées pour un type de service particulier.

Pour imprimer une liste des services cluster actuellement disponibles pour votre cluster, exécutez la commande suivante :

```
ccs -h host --lsserviceopts
```

Par exemple, la commande suivante répertorie les services cluster disponibles sur le nœud **node1** du cluster, affichant un exemple de sortie.

```
[root@ask-03 ~]# ccs -h node1 --lsserviceopts
service - Defines a service (resource group).
ASEHAagent - Sybase ASE Failover Instance
SAPDatabase - SAP database resource agent
SAPInstance - SAP instance resource agent
apache - Defines an Apache web server
clusterfs - Defines a cluster file system mount.
```

```
fs - Defines a file system mount.
ip - This is an IP address.
lvm - LVM Failover script
mysql - Defines a MySQL database server
named - Defines an instance of named server
netfs - Defines an NFS/CIFS file system mount.
nfsclient - Defines an NFS client.
nfsexport - This defines an NFS export.
nfsserver - This defines an NFS server resource.
openldap - Defines an Open LDAP server
oracledb - Oracle 10g Failover Instance
orainstance - Oracle 10g Failover Instance
oralistener - Oracle 10g Listener Instance
postgres-8 - Defines a PostgreSQL server
samba - Dynamic smbd/nmbd resource agent
script - LSB-compliant init script as a clustered resource.
tomcat-6 - Defines a Tomcat server
vm - Defines a Virtual Machine
action - Overrides resource action timings for a resource instance.
```

Pour imprimer une liste des options que vous pouvez spécifier pour un type de service particulier, exécutez la commande suivante :

```
ccs -h host --lsserviceopts service_type
```

Par exemple, la commande suivante répertorie les options de service pour le service **vm**.

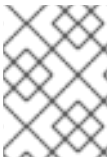
```
[root@ask-03 ~]# ccs -f node1 --lsserviceopts vm
vm - Defines a Virtual Machine
  Required Options:
    name: Name
  Optional Options:
    domain: Cluster failover Domain
    autostart: Automatic start after quorum formation
    exclusive: Exclusive resource group
    recovery: Failure recovery policy
    migration_mapping: memberhost:targethost,memberhost:targethost ..
    use_virsh: If set to 1, vm.sh will use the virsh command to manage
virtual machines instead of xm. This is required when using non-Xen
virtual machines (e.g. qemu / KVM).
    xmlfile: Full path to libvirt XML file describing the domain.
    migrate: Migration type (live or pause, default = live).
    path: Path to virtual machine configuration files.
    snapshot: Path to the snapshot directory where the virtual machine
image will be stored.
    depend: Top-level service this depends on, in service:name format.
    depend_mode: Service dependency mode (soft or hard).
    max_restarts: Maximum restarts for this service.
    restart_expire_time: Restart expiration time; amount of time before a
restart is forgotten.
    status_program: Additional status check program
    hypervisor: Hypervisor
    hypervisor_uri: Hypervisor URI (normally automatic).
    migration_uri: Migration URI (normally automatic).
    __independent_subtree: Treat this and all children as an independent
```

```
subtree.
    __enforce_timeouts: Consider a timeout for operations as fatal.
    __max_failures: Maximum number of failures before returning a failure
to a status check.
    __failure_expire_time: Amount of time before a failure is forgotten.
    __max_restarts: Maximum number restarts for an independent subtree
before giving up.
    __restart_expire_time: Amount of time before a failure is forgotten
for an independent subtree.
```

5.12. RESSOURCES DE MACHINE VIRTUELLE

Les ressources de machine virtuelle sont configurées différemment des autres ressources de cluster. En particulier, elles ne sont pas regroupées en définitions de services. À partir de la version 6.2 de Red Hat Enterprise Linux, lorsque vous configurez une machine virtuelle dans un cluster avec la commande **ccs**, vous pouvez utiliser **--addvm** (plutôt que l'option **addservice**). Ceci assure que la ressource **vm** est directement définie sous le nœud de configuration **rm** dans le fichier de configuration du cluster.

Une ressource de machine virtuelle requiert au minimum les attributs **name** (nom) et **path** (chemin). L'attribut **name** doit correspondre au nom du domaine **libvirt** et l'attribut **path** doit spécifier le répertoire où les définitions partagées de la machine virtuelle sont stockées.



NOTE

L'attribut **path** dans le fichier de configuration du cluster est une spécification de chemin ou un nom de répertoire, pas un chemin vers un fichier individuel.

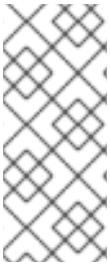
Si les définitions de machines virtuelles sont stockées sur un répertoire partagé nommé **/mnt/vm_defs**, la commande suivante définira une machine virtuelle nommée **guest1** :

```
# ccs -h node1.example.com --addvm guest1 path=/mnt/vm_defs
```

L'exécution de cette commande ajoute la ligne suivante au nœud de configuration dans le fichier **cluster.conf** :

```
<vm name="guest1" path="/mnt/vm_defs"/>
```

5.13. CONFIGURER UN DISQUE QUORUM :



NOTE

Les paramètres et heuristiques d'un disque quorum dépendent de l'environnement du site et des prérequis spéciaux nécessaires. Pour comprendre l'utilisation des paramètres et heuristiques du disque quorum, reportez-vous à la page man qdisk(5). Si vous avez besoin d'aide pour la compréhension et l'utilisation d'un disque quorum, veuillez contacter un représentant autorisé du support Red Hat.

Pour configurer votre système pour l'utilisation d'un disque quorum, utilisez la commande suivante :

```
ccs -h host --setquorumd [quorumd options]
```

Remarquez que cette commande réinitialise toutes les autres propriétés que vous pouvez paramétrer avec l'option **--setquorumd** avec leurs valeurs par défaut, comme le décrit la Section 5.1.5, « Commandes remplaçant les paramètres précédents ».

Le Tableau 5.1, « Options du disque quorum » résume la signification des options du disque quorum que vous pourriez devoir paramétrer. Pour obtenir la liste complète des paramètres du disque quorum, reportez-vous au schéma du cluster sur **/usr/share/cluster/cluster.rng** et au schéma annoté sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html**.

Tableau 5.1. Options du disque quorum

Paramètre	Description
interval	Fréquence, en secondes, des cycles de lecture/écriture.
votes	Nombre de votes que le démon quorum annonce à **cman** lorsqu'il obtient un score assez élevé.
tko	Nombre de cycles qu'un nœud doit rater pour être déclaré comme étant mort.
min_score	Score minimum qu'un nœud doit effectuer pour être considéré comme « vivant ». Si oublié, ou si ajusté sur 0, **floor((*n*+1)/2)**, est utilisé, où *n* est la somme des scores heuristiques. La valeur **Minimum Score** ne doit jamais excéder la somme des scores heuristiques, sinon le disque quorum ne pourra pas être disponible.
device	Périphérique de stockage que le démon quorum utilise. Le périphérique doit être le même sur tous les nœuds.
label	Spécifie l'étiquette du disque quorum créé par l'utilitaire **mkqdisk**. Si ce champ contient une entrée, l'étiquette remplace le champ **Device**. Si ce champ est utilisé, le démon quorum lit **/proc/partitions** et vérifie les signatures qdisk sur chaque périphérique bloc trouvé, comparant l'étiquette à l'étiquette spécifiée. Ceci est utile pour les configurations dans lesquelles le nom du périphérique quorum diffère selon les nœuds.

Utilisez la commande suivante pour configurer les heuristiques pour un disque quorum :

```
ccs -h host --addheuristic [heuristic options]
```

Le Tableau 5.2, « Heuristiques du disque quorum » résume la signification des heuristiques du disque quorum que vous pourriez devoir paramétrer.

Tableau 5.2. Heuristiques du disque quorum

Paramètre	Description
program	Chemin vers le programme utilisé pour déterminer si cette heuristique est disponible. Ceci peut être n'importe quoi qui est exécutable par **/bin/sh -c**. Une valeur retournée de 0 indique un succès ; toute autre chose indique un échec. Ce paramètre est requis pour utiliser un disque quorum.

Paramètre	Description
interval	Fréquence (en secondes) à laquelle l'heuristique est analysée. L'intervalle par défaut pour toute heuristique est de 2 secondes.
score	Poids de l'heuristique. Soyez prudent lorsque vous déterminez les scores des heuristiques. Le score par défaut de chaque heuristique est de 1.
tko	Nombre d'échecs consécutifs requis avant que cette heuristique ne soit déclarée indisponible.

Pour afficher une liste des options du disque quorum et des heuristique configurées sur un système, vous pouvez exécuter la commande suivante :

```
ccs -h host --lsquorum
```

Pour supprimer une heuristique spécifiée par une option d'heuristique, vous pouvez exécuter la commande suivante :

```
ccs -h host rmheuristic [heuristic options]
```

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

NOTE

La synchronisation et l'activation propage et active le fichier de configuration du cluster mis à jour. Cependant, pour que le disque quorum puisse opérer, vous devez redémarrer le cluster (reportez-vous à la Section 6.2, « Démarrer et arrêter un cluster »), vous assurant ainsi que vous avez bien redémarré le démon **qdiskd** sur chaque nœud.

5.14. DIVERSES CONFIGURATIONS DE CLUSTERS

Cette section décrit l'utilisation de la commande **ccs** pour configurer ce qui suit :

- Section 5.14.1, « Version de la configuration du cluster »
- Section 5.14.2, « Configuration de la multidiffusion »
- Section 5.14.3, « Configurer un cluster à deux nœuds »
- Section 5.14.4, « Journalisation »
- Section 5.14.5, « Configurer le protocole d'anneau redondant (« Redundant Ring ») »

Vous pouvez aussi utiliser la commande **ccs** pour définir les paramètres de configuration avancés du cluster, y compris les options **totem**, **dlm**, **rm** et **cman**. Pour obtenir des informations sur la définition de ces paramètres, voir la page man **ccs**(8) et le schéma annoté du fichier de configuration du cluster sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html**.

Pour afficher une liste des divers attributs du cluster qui ont été configurés pour un cluster, exécutez la commande suivante :

```
ccs -h host --lsmisc
```

5.14.1. Version de la configuration du cluster

Un fichier de configuration de cluster inclut une valeur de version de configuration d'un cluster. La valeur de la version de la configuration est définie sur **1** par défaut lorsque vous créez un fichier de configuration de cluster, celle-ci est automatiquement incrémentée chaque fois que vous modifiez la configuration de votre cluster. Cependant, si vous devez la définir sur une autre valeur, vous pouvez la spécifier à l'aide de la commande suivante :

```
ccs -h host --setversion n
```

Vous pouvez obtenir la valeur de la version de la configuration actuelle sur un fichier de configuration de cluster existant à l'aide de la commande suivante :

```
ccs -h host --getversion
```

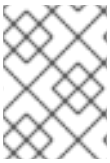
Pour incrémenter la valeur de la version de la configuration de 1 dans le fichier de configuration du cluster sur tous les nœuds du cluster, exécutez la commande suivante :

```
ccs -h host --incversion
```

5.14.2. Configuration de la multidiffusion

Si vous ne spécifiez pas d'adresse de multidiffusion dans le fichier de configuration du cluster, le logiciel du module complémentaire Red Hat High Availability va en créer une basée sur l'ID du cluster. Le logiciel générera les 16 bits les plus bas de l'adresse et les ajoutera à la portion la plus haute de l'adresse selon que le protocole IP est IPv4 ou IPv6 :

- Pour IPv4 — L'adresse formée est 239.192. plus les 16 bits les plus bas générés par le logiciel du module complémentaire Red Hat High Availability.
- Pour IPv6 — L'adresse formée est FF15:: plus les 16 bits les plus bas générés par le logiciel du module complémentaire Red Hat High Availability.



NOTE

L'ID du cluster est un identifiant unique que **cman** génère pour chaque cluster. Pour voir l'ID du cluster, exécutez la commande **cman_tool status** sur un nœud de cluster.

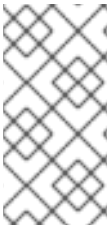
Vous pouvez manuellement spécifier une adresse de multidiffusion dans le fichier de configuration du cluster avec la commande suivante :

```
ccs -h host --setmulticast multicastaddress
```

Remarquez que cette commande réinitialise toutes les autres propriétés que vous pouvez paramétrer avec l'option **--setmulticast** avec leurs valeurs par défaut, comme le décrit la Section 5.1.5, « Commandes remplaçant les paramètres précédents ».

Si vous spécifiez une adresse de multidiffusion, vous devriez utiliser les séries 239.192.x.x (ou FF15:: pour IPv6) utilisées par **cman**. L'utilisation d'une adresse de multidiffusion hors de cette plage peut provoquer des résultats imprévisibles. Par exemple, utiliser 224.0.0.x (qui équivaut à "All hosts on the network") peut ne pas être acheminé correctement, certains matériaux pourraient même ne pas du tout l'acheminer.

Si vous spécifiez ou modifiez une adresse de multidiffusion, vous devrez redémarrer le cluster pour que celle-ci prenne effet. Pour obtenir des informations sur le démarrage et l'arrêt d'un cluster avec la commande **ccs**, reportez-vous à la Section 6.2, « Démarrer et arrêter un cluster ».



NOTE

Si vous spécifiez une adresse de multidiffusion, assurez-vous de bien vérifier la configuration des routeurs par lesquels les paquets des clusters passent. Certains routeurs prennent longtemps pour apprendre les adresses, affectant ainsi sévèrement la performance du cluster.

Pour supprimer une adresse de multidiffusion d'un fichier de configuration, utilisez l'option **--setmulticast** de **ccs** mais ne spécifiez pas d'adresse de multidiffusion :

```
ccs -h host --setmulticast
```

5.14.3. Configurer un cluster à deux nœuds

Si vous êtes en train de configurer un cluster à deux nœuds, vous pouvez exécuter la commande suivante afin de permettre à un nœud unique de maintenir le quorum (si un nœud échoue par exemple) :

```
ccs -h host --setcman two_node=1 expected_votes=1
```

Remarquez que cette commande réinitialise toutes les autres propriétés que vous pouvez paramétrer avec l'option **--setcman** avec leurs valeurs par défaut, comme le décrit la Section 5.1.5, « Commandes remplaçant les paramètres précédents ».

Lorsque vous utilisez la commande **ccs --setcman** pour ajouter, supprimer, ou pour modifier l'option **two_node**, vous devez redémarrer le cluster pour que ce changement prenne effet. Pour obtenir des informations sur le démarrage et l'arrêt d'un cluster avec la commande **ccs**, reportez-vous à la Section 6.2, « Démarrer et arrêter un cluster ».

5.14.4. Journalisation

Vous pouvez activer le débogage de tous les démons dans un cluster ou activer la journalisation pour le traitement de cluster spécifique.

Pour activer le débogage de tous les démons, exécutez la commande suivante. Par défaut, la journalisation est dirigée vers le fichier **/var/log/cluster/*démon*.log**.

```
ccs -h host --setlogging [logging options]
```

Par exemple, la commande suivante active le débogage de tous les démons.

```
# ccs -h node1.example.com --setlogging debug=on
```

Remarquez que cette commande réinitialise toutes les autres propriétés que vous pouvez paramétrer avec l'option **--setlogging** avec leurs valeurs par défaut, comme le décrit la Section 5.1.5, « Commandes remplaçant les paramètres précédents ».

Pour activer le débogage d'un processus individuel, exécutez la commande suivante. La configuration de la journalisation « par démon » remplace les paramètres généraux.

```
ccs -h host --addlogging [logging daemon options]
```

Par exemple, les commandes suivantes activent le débogage des démons **corosync** et **fenced**.

```
# ccs -h node1.example.com --addlogging name=corosync debug=on
# ccs -h node1.example.com --addlogging name=fenced debug=on
```

Pour supprimer les paramètres de journalisation des démons individuels, utilisez la commande suivante.

```
ccs -h host --rmlogging name=clusterprocess
```

Par exemple, la commande suivante supprime les paramètres de journalisation spécifiques au démon **fenced**

```
ccs -h host --rmlogging name=fenced
```

Pour la liste des démons pour lesquels vous pouvez activer la journalisation ainsi que pour les options de journalisation supplémentaires que vous pouvez configurer pour la journalisation globale ou « par démon », veuillez vous reporter à la page man **cluster.conf**(5).

Remarquez que lorsque vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster à tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.14.5. Configurer le protocole d'anneau redondant (« Redundant Ring »)

À partir de Red Hat Enterprise Linux 6.4, le module complémentaire Red Hat High Availability prend en charge la configuration du protocole d'anneau redondant. Lors de l'utilisation du protocole d'anneau redondant, un certain nombre de considérations sont à prendre en compte, comme le décrit la Section 7.6, « Configurer le protocole d'anneau redondant (« Redundant Ring ») ».

Pour spécifier une seconde interface réseau à utiliser pour le protocole d'anneau redondant, ajoutez un nom alterne pour le nœud en utilisant l'option **--addalt** de la commande **ccs** :

```
ccs -h host --addalt node_name alt_name
```

Par exemple, la commande suivante configure le nom alternatif **clusternet-node1-eth2** du nœud de cluster **clusternet-node1-eth1** :

```
# ccs -h clusternet-node1-eth1 --addalt clusternet-node1-eth1 clusternet-
node1-eth2
```

Optionnellement, vous pouvez manuellement spécifier une adresse de multidiffusion, un port et un TTL pour le second anneau. Si vous spécifiez une adresse de multidiffusion pour le second anneau, soit l'adresse de multidiffusion alterne, soit le port alterne doit être différent de l'adresse de multidiffusion du premier anneau. Si vous spécifiez un port alterne, les numéros de port du premier anneau et du second

anneau doivent être différents d'au moins deux car le système utilise port et port-1 pour effectuer des opérations. Si vous ne spécifiez pas d'adresse de multidiffusion, le système utilisera automatiquement une adresse de multidiffusion différente pour le second anneau.

Pour spécifier un adresse de multidiffusion alterne, ou un port ou un TTL alterne pour le second anneau, vous devez utiliser l'option **--setaltmulticast** de la commande **ccs** :

```
ccs -h host --setaltmulticast [alt_multicast_address]
[alt_multicast_options].
```

Par exemple, la commande suivante définit une adresse de multidiffusion de 239.192.99.88, le port 888 et un TTL de 3 pour le cluster défini dans le fichier **cluster.conf** sur le nœud **clusternet-node1-eth1** :

```
ccs -h clusternet-node1-eth1 --setaltmulticast 239.192.99.88 port=888
ttl=3
```

Pour supprimer une adresse de multidiffusion alterne, spécifiez l'option **--setaltmulticast** de la commande **ccs**, mais ne spécifiez pas d'adresse de multidiffusion. Remarquez que l'exécution de cette commande réinitialise toutes les autres propriétés que vous pouvez définir avec l'option **--setaltmulticast** avec leurs valeurs par défaut, comme décrit dans la Section 5.1.5, « Commandes remplaçant les paramètres précédents ».

Une fois que vous aurez terminé de configurer tous les composants de votre cluster, vous devrez synchroniser le fichier de configuration du cluster avec tous les nœuds, comme le décrit la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

5.15. PROPAGER LE FICHIER DE CONFIGURATION SUR LES NŒUDS DU CLUSTER

Après avoir créé ou modifié un fichier de configuration de cluster sur un des nœuds du cluster, vous devrez propager ce même fichier sur tous les nœuds du cluster et activer la configuration.

Utilisez la commande suivante pour propager et activer un fichier de configuration de cluster :

```
ccs -h host --sync --activate
```

Pour vérifier que tous les nœuds spécifiés dans le fichier de configuration du cluster hôte possèdent un fichier de configuration identique, exécutez la commande suivante :

```
ccs -h host --checkconf
```

Si vous avez créé ou modifié un fichier de configuration sur un nœud local, utilisez la commande suivante pour envoyer ce fichier sur un des nœuds du cluster :

```
ccs -f file -h host --setconf
```

Pour vérifier que tous les nœuds spécifiés dans le fichier local possèdent le même fichier de configuration de cluster, exécutez la commande suivante :

```
ccs -f file --checkconf
```

CHAPITRE 6. GÉRER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY AVEC CCS

Ce chapitre décrit les diverses tâches adminstratives pour la gestion du module complémentaire Red Hat High Availability au moyen de la commande **ccs**, qui est prise en charge à partir de la version 6.1 de Red Hat Enterprise Linux et de ses versions plus récentes. Ce chapitre est composé des sections suivantes :

- Section 6.1, « Gérer les nœuds de clusters »
- Section 6.2, « Démarrer et arrêter un cluster »
- Section 6.3, « Diagnostiquer et corriger des problèmes dans un cluster »

6.1. GÉRER LES NŒUDS DE CLUSTERS

Cette section documente comment effectuer les fonctions de gestion de nœuds suivantes avec la commande **ccs** :

- Section 6.1.1, « Causer à un nœud de joindre ou quitter un cluster »
- Section 6.1.2, « Ajouter un membre à un cluster en cours d'exécution »

6.1.1. Causer à un nœud de joindre ou quitter un cluster

Vous pouvez utiliser la commande **ccs** pour faire qu'un nœud quitte un cluster en arrêtant les services cluster sur ce nœud. Causer le départ d'un nœud d'un cluster ne supprime pas les informations de configuration de ce nœud. Faire qu'un nœud quitte un cluster empêche le nœud de joindre le cluster automatiquement lorsqu'il est redémarré.

Pour qu'un nœud quitte un cluster, exécutez la commande suivante, celle-ci stoppe les services cluster sur le nœud spécifié avec l'option **-h** :

```
ccs -h host --stop
```

Lorsque vous arrêtez les services cluster sur un nœud, tout service exécuté sur ce nœud basculera :

Pour complètement supprimer un nœud de la configuration du cluster, utilisez l'option **--rmnode** de la commande **ccs**, comme décrit dans la Section 5.4, « Créer un cluster ».

Pour faire en sorte qu'un nœud rejoigne un cluster, exécutez la commande suivante, celle-ci démarre les services cluster sur le nœud spécifié avec l'option **-h** :

```
ccs -h host --start
```

6.1.2. Ajouter un membre à un cluster en cours d'exécution

Pour ajouter un membre à un cluster en cours d'exécution, ajoutez un nœud au cluster comme décrit dans la Section 5.4, « Créer un cluster ». Après avoir mis à jour le fichier de configuration, propagez le fichier sur tous les nœuds dans le cluster et assurez-vous de bien activer le nouveau fichier de configuration du cluster, comme décrit dans la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».

6.2. DÉMARRER ET ARRÊTER UN CLUSTER

Vous pouvez utiliser la commande **ccs** pour arrêter un cluster à l'aide de la commande suivante, celle-ci stoppe les services cluster sur tous les nœuds dans le cluster :

```
ccs -h host --stopall
```

Vous pouvez utiliser la commande **ccs** pour démarrer un cluster qui n'est pas en cours d'exécution à l'aide de la commande suivante, celle-ci lance les services cluster sur tous les nœuds dans le cluster :

```
ccs -h host --startall
```

6.3. DIAGNOSTIQUER ET CORRIGER DES PROBLÈMES DANS UN CLUSTER

Pour obtenir des informations sur le diagnostic et la correction de problèmes dans un cluster, voir le Chapitre 9, *Diagnostiquer et corriger des problèmes dans un cluster*. Il existe de simples vérifications que vous pouvez mener à l'aide de la commande **ccs**.

Pour vérifier que tous les nœuds spécifiés dans le fichier de configuration du cluster hôte possèdent des fichiers de configuration de cluster identiques, exécutez la commande suivante :

```
ccs -h host --checkconf
```

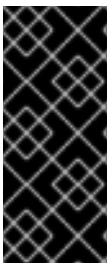
Si vous avez créé ou modifié un fichier de configuration sur un nœud local, vous pouvez vérifier que tous les nœuds spécifiés dans le fichier local possèdent des fichiers de configuration du cluster identiques à l'aide de la commande suivante :

```
ccs -f file --checkconf
```

CHAPITRE 7. CONFIGURER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY AVEC DES OUTILS DE LIGNE DE COMMANDE

Ce chapitre décrit comment configurer le logiciel du module complémentaire Red Hat High Availability en modifiant directement le fichier de configuration du cluster (**/etc/cluster/cluster.conf**) et en utilisant des outils de ligne de commande. Ce chapitre fournit des procédures sur la construction d'un fichier de configuration étape par étape, en commençant par un fichier exemple fournit dans le chapitre. Un fichier de configuration squelette peut être copié depuis la page man **cluster.conf** et servir d'alternative au commencement avec l'exemple de fichier ci-joint. Cependant, faire ainsi ne s'aligne pas forcément sur les informations fournies dans les procédures ultérieures de ce chapitre. Il existe d'autres manières de créer et de configurer un fichier de configuration de cluster ; ce chapitre propose des procédures pour la construction une section à la fois. Aussi, n'oubliez pas qu'il ne s'agit que du point de départ pour le développement d'un fichier de configuration adapté à vos besoins de mise en cluster.

Ce chapitre est composé des sections suivantes :

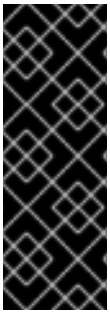


IMPORTANT

Assurez-vous que le déploiement du module complémentaire High Availability répond bien à vos besoins et qu'il est pris en charge. Consultez un représentant Red Hat autorisé afin de vérifier votre configuration avant le déploiement. En outre, prévoyez suffisamment de temps pour une période de rodage de la configuration afin de tester les différents modes d'échec.

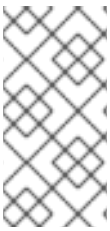
IMPORTANT

Ce chapitre fait référence aux éléments et attributs de **cluster.conf** communément utilisés. Pour obtenir la liste et la description complète des éléments et attributs **cluster.conf**, reportez-vous au schéma des clusters sur **/usr/share/cluster/cluster.rng**, et au schéma annoté sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).



IMPORTANT

Certaines procédures dans ce chapitre appellent à utiliser la commande **`cman_tool version -r`** pour propager une configuration de cluster à travers un cluster. L'utilisation de cette commande requiert que **`ricci`** soit en cours d'exécution. L'utilisation de **ricci** requerra un mot de passe la première fois que vous aurez une interaction avec **ricci**, et ce depuis n'importe quelle machine. Pour obtenir des informations sur le service **`ricci`**, reportez-vous à la Section 2.13, « Considérations pour **`ricci`** ».



NOTE

Les procédures dans ce chapitre peuvent inclure des commandes spécifiques pour certains outils en ligne de commande répertoriés dans l'Annexe E, *Résumé des outils de la ligne de commande*. Pour obtenir plus d'informations sur les commandes et les variables, reportez-vous à la page man de chaque outil de ligne de commande.

7.1. TÂCHES DE CONFIGURATION

La configuration du logiciel du module complémentaire Red Hat High Availability avec des outils de ligne de commande est composé des étapes suivantes :

1. Création d'un cluster. Reportez-vous à la Section 7.2, « Création d'un fichier de configuration de cluster de base ».
2. Configuration du fencing. Reportez-vous à la Section 7.3, « Configurer le fencing ».
3. Configuration des domaines de basculement. Reportez-vous à la Section 7.4, « Configurer les domaines de basculement ».
4. Configuration des services HA. Reportez-vous à la Section 7.5, « Configurer les services HA ».
5. Vérification d'une configuration. Reportez-vous à la Section 7.8, « Vérifier une configuration ».

7.2. CRÉATION D'UN FICHIER DE CONFIGURATION DE CLUSTER DE BASE

Pourvu que le matériel du cluster, Red Hat Enterprise Linux, et le logiciel du module complémentaire High Availability soient installés, vous pourrez créer un fichier de configuration de cluster (**`/etc/cluster/cluster.conf`**) et commencer à exécuter le module complémentaire High Availability. En tant que point de démarrage seulement, cette section décrit comment créer un squelette de fichier de configuration de cluster sans utiliser le fencing, de domaines de basculement, ou de services HA. Les sections utlérieures décrivent comment configurer ces parties du fichier de configuration.

IMPORTANT

Ceci n'est qu'une étape intermédiaire pour créer un fichier de configuration de cluster, le fichier en résultant n'est pas clôturé et n'est pas considéré comme une configuration prise en charge.

Les étapes suivantes décrivent comment créer et configurer un squelette de fichier de configuration de cluster. Finalement, le fichier de configuration de votre cluster variera selon le nombre de nœuds, le type de fencing, le type et le nombre de services HA et selon d'autres exigences spécifiques au site.

1. Sur n'importe quel nœud du cluster, créez **/etc/cluster/cluster.conf** à l'aide du modèle de l'exemple dans l'Exemple 7.1, « Exemple de **cluster.conf** : configuration de base ».

2. **(Optional)** Si vous configurez un cluster à deux nœuds, vous pouvez ajouter la ligne suivante au fichier de configuration afin de permettre à un nœud unique de maintenir le quorum (si un nœud échoue par exemple) :

   ```
   <cman two_node="1" expected_votes="1"/>
   ```

   Lorsque vous ajoutez ou supprimez l'option **two_node** du fichier **cluster.conf**, vous devez redémarrer le cluster pour que cette modification prenne effet lors de la mise à jour de la configuration. Pour des informations sur la mise à jour d'une configuration de cluster, reportez-vous à la Section 8.4, « Mettre à jour une configuration ». Pour un exemple de spécification de l'option **two_node**, reportez-vous à l'Exemple 7.2, « Exemple de **cluster.conf** : configuration à deux nœuds de base ».\n\n\t\n

3. Spécifiez le nom du cluster ainsi que son numéro de version de configuration à l'aide des attributs **cluster** : **name** et **config_version** (reportez-vous à l'Exemple 7.1, « Exemple de **cluster.conf** : configuration de base » ou à l'Exemple 7.2, « Exemple de **cluster.conf** : configuration à deux nœuds de base »).

4. Dans la section **clusternodes**, spécifiez le nom du nœud et l'ID du nœud de chaque nœud utilisant les attributs **clusternode** : **name** et **nodeid**.

5. Enregistrez **/etc/cluster/cluster.conf**.

6. Validez le fichier avec le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

   ```
   [root@example-01 ~]# ccs_config_validate
   Configuration validates
   ```

7. Propagez le fichier de configuration sur **/etc/cluster/** dans chaque nœud du cluster. Par exemple, vous pourriez propager le fichier vers d'autres nœuds de cluster à l'aide de la commande **scp**.

NOTE

La propagation d'un fichier de configuration de cluster de cette manière est nécessaire la première fois qu'un cluster est créé. Une fois que le cluster est installé et en cours d'exécution, le fichier de configuration du cluster peut être propagé à l'aide de **cman_tool version -r**. Il est possible d'utiliser la commande **scp** pour propager un fichier de configuration mis à jour. Cependant, le logiciel du cluster doit être arrêté sur tous les nœuds pendant l'utilisation de la commande **scp**. En outre, vous devriez exécuter **ccs_config_validate** si vous propagez un fichier de configuration mis à jour via la commande **scp**.

NOTE

Tandis que d'autres éléments et attributs sont présents dans l'exemple du fichier de configuration (par exemple, **fence** et **fencedevices**), il n'est pas nécessaire de les remplir maintenant. Des procédures expliquées ultérieurement dans ce chapitre fournissent des informations sur la spécification d'autres éléments et attributs.

8. Démarrez le cluster. Exécutez la commande suivante sur chaque nœud de cluster :

   service cman start

   Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [  OK
]
   Global setup...                                         [  OK
]
   Loading kernel modules...                               [  OK
]
   Mounting configfs...                                    [  OK
]
   Starting cman...                                        [  OK
]
   Waiting for quorum...                                   [  OK
]
   Starting fenced...                                      [  OK
]
   Starting dlm_controld...                                [  OK
]
   Starting gfs_controld...                                [  OK
]
   Unfencing self...                                       [  OK
]
   Joining fence domain...                                 [  OK
]
```

9. Sur n'importe quel nœud de cluster, exécutez **cman_tool nodes** pour vérifier que les nœuds fonctionnent en tant que membres dans le cluster (décrit comme « M » dans la colonne du statut « Sts »). Par exemple :

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

10. Si le cluster est en cours d'exécution, procédez à Section 7.3, « Configurer le fencing ».

Exemples de configurations de base

L'Exemple 7.1, « Exemple de **cluster.conf** : configuration de base » et l'Exemple 7.2, « Exemple de **cluster.conf** : configuration à deux nœuds de base » (pour un cluster à deux nœuds) fournissent tous deux un exemple très basique de fichier de configuration de cluster comme point de départ. Les procédures suivantes dans ce chapitre fournissent des informations sur la configuration du fencing et des services HA.

Exemple 7.1. Exemple de `cluster.conf` : configuration de base

```
<cluster name="mycluster" config_version="2">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Exemple 7.2. Exemple de `cluster.conf` : configuration à deux nœuds de base

```
<cluster name="mycluster" config_version="2">
   <cman two_node="1" expected_votes="1"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

La valeur du `consensus` pour `totem` dans un cluster à deux nœuds

Lorsque vous créez un cluster à deux nœuds et que vous ne souhaitez pas ajouter de nœud supplémentaire au cluster ultérieurement, vous devriez alors omettre la valeur du **consensus** dans la balise **`totem`** du fichier **`cluster.conf`**, ainsi la valeur **consensus** sera calculée automatiquement. Lorsque la valeur **consensus** est calculée automatiquement, les règles suivantes sont utilisées :

- S'il y a deux nœuds ou moins, la valeur **consensus** sera (token * 0.2), avec un plafond de 2000 msec et un plancher de 200 msec.
- S'il y a trois nœuds ou plus, la valeur **consensus** sera (token + 2000 msec)

Si vous laissez l'utilitaire **cman** configurer votre délai d'expiration de consensus de cette manière, alors le déplacement ultérieur de deux à trois nœuds (ou plus) requerra le redémarrage du cluster, puisque le délai d'expiration du consensus devra changer vers cette valeur plus importante, basée sur le délai d'expiration du token.

Si vous configurez un cluster à deux nœuds et souhaitez le mettre à jour dans le futur à plus de deux nœuds, vous pouvez remplacer le délai d'expiration du consensus de manière à ce qu'un redémarrage du cluster ne soit pas requis lors du déplacement de deux à trois nœuds (ou plus). Ceci peut être effectué dans le fichier **cluster.conf** comme suit :

```
<totem token="X" consensus="X + 2000" />
```

Remarquez que l'analyse de configuration (de l'anglais, « configuration parser ») ne calcule pas X + 2000 automatiquement. Une valeur entière doit être utilisée plutôt qu'une équation.

L'avantage offert par l'utilisation du délai d'expiration optimisé du consensus pour des clusters à deux nœuds est que le temps pris par le basculement est réduit pour les cas à deux nœuds puisque le consensus n'est pas une fonction du délai d'expiration du token.

Remarquez que pour l'autodétection de deux nœuds dans **cman**, le nombre de neouds physiques est le plus importants, et non la présence de la directive **two_node=1** dans le fichier **cluster.conf**.

7.3. CONFIGURER LE FENCING

La configuration du fencing consiste en (a) la spécification d'un (ou plusieurs) périphérique(s) fence dans un cluster et en (b) la spécification d'une (ou plusieurs) méthode(s) fence pour chaque nœud (à l'aide du ou des périphériques spécifiés).

Configurez **cluster.conf** comme suit en vous basant sur le type des périphériques et des méthodes fence requis pour votre configuration :

1. Dans la section **fencedevices**, spécifiez chaque périphérique fence à l'aide d'un élément **fencedevice** et d'attributs dépendants au(x) périphérique(s) fence. L'Exemple 7.3, « Périphérique fence APC ajouté à **cluster.conf** » montre un exemple de fichier de configuration avec un périphérique fence APC qui lui est ajouté.
2. Sur la section **clusternodes**, dans l'élément **fence** de chaque section de **clusternode**, spécifiez chaque méthode fence du nœud. Spécifiez le nom de la méthode fence à l'aide de l'attribut **name** de **method**. Spécifiez le périphérique fence pour chaque méthode fence à l'aide de l'élément **device** et de ses attributs, **name** et des paramètres spécifiques au périphérique fence. L'Exemple 7.4, « Méthodes fence ajoutées à **cluster.conf** » montre un exemple de méthode fence avec un périphérique fence pour chaque nœud dans le cluster.
3. Pour des méthodes fence non-alimentées (c'est-à-dire le fencing SAN/stockage) dans la section **clusternodes**, ajoutez une section **unfence**. Ceci vous assure qu'un nœud fenced n'est pas ré-activé tant que le nœud n'a pas été redémarré. Pour plus d'informations sur l'unfencing d'un nœud, reportez-vous à la page man **fence_node**(8).

La section **unfence** ne contient pas de sections **method** comme la section **fence**. Elle contient des références **device**, qui mettent en miroir les sections des périphériques correspondants pour **fence**, avec l'addition notable de l'action explicite (**action**) sur "on" ou sur "enable". La même section **fencedevice** est référencée par les lignes **fence** et **unfence device** et les mêmes arguments par nœud devraient être répétés.

La spécification de l'attribut **action** sur "on" ou sur "enable" active le nœud lorsque redémarré. L'Exemple 7.4, « Méthodes fence ajoutées à **cluster.conf** » et l'Exemple 7.5, « **cluster.conf** : Multiples méthodes fence par nœud » incluent des exemple des éléments et attributs **unfence**.

Pour obtenir plus d'informations sur **unfence**, reportez-vous à la page man **fence_node**.

4. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en la modifiant de **config_version="2"** à **config_version="3">**).

5. Enregistrez **/etc/cluster/cluster.conf**.

6. **(Optional)** Validez le fichier mis à jour sur le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Exécutez la commande **cman_tool version -r** pour propager la configuration au reste des nœuds du cluster. Ceci exécutera aussi une validation supplémentaire. Il est nécessaire que **ricci** soit exécuté dans chaque nœud de cluster afin de pouvoir propager les informations mises à jour sur la configuration du cluster.

8. Vérifiez que le fichier de configuration mis à jour a été propagé.

9. Procédez à la Section 7.4, « Configurer les domaines de basculement ».

Si nécessaire, vous pouvez configurer des configurations complexes avec de multiples méthodes fence par nœud et de multiples périphériques fence par méthode fence. Lors de la spécification de multiples méthodes fence par nœud, si le fencing échoue avec la première méthode, le démon fence, **fenced**, tente la méthode suivante et continuera d'essayer les méthodes successivement jusqu'à ce qu'une d'entre elles fonctionne.

De temps en temps, le fencing d'un nœud requiert la désactivation de deux chemins d'E/S ou de deux ports d'alimentation. Ceci est effectué en spécifiant deux périphériques ou plus dans la méthode fence. **fenced** exécute l'agent fence une fois par ligne de périphérique fence, chaque ligne doit fonctionner pour que le fencing soit considéré comme réussi.

Des configurations plus complexes sont affichées dans la section intitulée « Exemples de configurations du fencing ».

Vous pouvez trouver plus d'informations sur la configuration de périphériques fence spécifiques sur la page man de l'agent des périphériques fence (par exemple, la page man **fence_apc**). En outre, vous pourrez trouver des informations supplémentaires sur les paramètres du fencing dans l'Annexe A, *Paramètres des périphériques fence*, sur les agents fence dans **/usr/sbin/**, sur le schéma du cluster dans **/usr/share/cluster/cluster.rng** et sur le schéma annoté sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

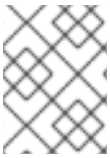
Exemples de configurations du fencing

Les exemple suivants montrent une simple configuration avec une méthode par nœud et un périphérique fence par méthode fence :

- Exemple 7.3, « Périphérique fence APC ajouté à **cluster.conf** »
- Exemple 7.4, « Méthodes fence ajoutées à **cluster.conf** »

Les exemples suivants proposent des configurations plus complexes :

- Exemple 7.5, « **cluster.conf** : Multiples méthodes fence par nœud »
- Exemple 7.6, « **cluster.conf** : Fencing, multiples ports multipath »
- Exemple 7.7, « **cluster.conf** : Effectuer le fencing de nœuds avec des alimentations duelles »



NOTE

Les exemples dans cette section ne sont pas exhaustifs, il peut y avoir d'autres manières de configurer le fencing en fonction de vos besoins.

Exemple 7.3. Périphérique fence APC ajouté à `cluster.conf`

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Dans cet exemple, un périphérique fence (**fencedevice**) a été ajouté à l'élément **fencedevices** en spécifiant l'agent fence (**agent**) en tant que **fence_apc**, l'adresse IP (**ipaddr**) en tant que **apc_ip_example**, l'identifiant de connexion (**login**) en tant que **login_example**, le nom du

périphérique fence (**name**) en tant que **apc** et le mot de passe (**passwd**) en tant que **password_example**.

Exemple 7.4. Méthodes fence ajoutées à `cluster.conf`

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Dans cet exemple, une méthode fence (**method**) a été ajoutée à chaque nœud. Le nom de la méthode fence (**name**) de chaque nœud est **APC**. Le périphérique (**device**) pour la méthode fence dans chaque nœud spécifie le nom (**name**) comme **apc** et un unique numéro de port d'alimentation de l'interrupteur APC (**port**) pour chaque nœud. Par exemple, le numéro de port de node-01.example.com est **1** (**port="1"**). Le nom de périphérique de chaque nœud (**device name="apc"**) pointe vers le périphérique fence au nom (**name**) **apc** sur cette ligne de l'élément **fencedevices** : **fencedevice agent="fence_apc" ipaddr="apc_ip_example" login="login_example" name="apc" passwd="password_example"**.

Exemple 7.5. `cluster.conf` : Multiples méthodes fence par nœud

```
<cluster name="mycluster" config_version="3">
```

```
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
            <method name="SAN">
       <device name="sanswitch1" port="11"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="11" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
            <method name="SAN">
       <device name="sanswitch1" port="12"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="12" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
            <method name="SAN">
       <device name="sanswitch1" port="13"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="13" action="on"/>
         </unfence
     </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Exemple 7.6. `cluster.conf` : Fencing, multiples ports multipath

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="SAN-multi">
       <device name="sanswitch1" port="11"/>
       <device name="sanswitch2" port="11"/>
     </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="11" action="on"/>
             <device name="sanswitch2" port="11" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="SAN-multi">
       <device name="sanswitch1" port="12"/>
       <device name="sanswitch2" port="12"/>
            </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="12" action="on"/>
             <device name="sanswitch2" port="12" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="SAN-multi">
       <device name="sanswitch1" port="13"/>
       <device name="sanswitch2" port="13"/>
            </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="13" action="on"/>
             <device name="sanswitch2" port="13" action="on"/>
         </unfence
     </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch2" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Exemple 7.7. `cluster.conf` : Effectuer le fencing de nœuds avec des alimentations duelles

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="1"action="off"/>
              <device name="apc2" port="1"action="off"/>
              <device name="apc1" port="1"action="on"/>
              <device name="apc2" port="1"action="on"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="2"action="off"/>
              <device name="apc2" port="2"action="off"/>
              <device name="apc1" port="2"action="on"/>
              <device name="apc2" port="2"action="on"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="3"action="off"/>
              <device name="apc2" port="3"action="off"/>
              <device name="apc1" port="3"action="on"/>
              <device name="apc2" port="3"action="on"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
       <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc1" passwd="password_example"/>
       <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc2" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Lors de l'utilisation d'interrupteurs d'alimentation pour clôturer des nœuds avec des alimentations duelles, les agents doivent fermer les deux ports d'alimentation avant de restaurer l'alimentation sur l'un ou l'autre. Le comportement off-on par défaut de l'agent pourrait faire que l'alimentation n'est jamais complètement désactivée sur le nœud.

7.4. CONFIGURER LES DOMAINES DE BASCULEMENT

Un domaine de basculement est un sous-ensemble nommé de nœuds d'un cluster qui sont capables d'exécuter un service cluster dans le cas d'un échec de nœud. Un domaine de basculement peut posséder les caractéristiques suivantes :

- Unrestricted — Ceci vous permet de spécifier qu'un sous-ensemble de membres est préféré, mais qu'un service cluster assigné à ce domaine peut s'exécuter sur n'importe quel membre disponible.
- Restricted — Ceci vous permet de restreindre les membres pouvant exécuter un service cluster en particulier. Si aucun des membres dans un domaine de basculement restricted n'est disponible, le service cluster ne pourra pas être lancé (manuellement ou par le logiciel du cluster).
- Unordered — Lorsqu'un service cluster est assigné à un domaine de basculement unordered, le membre sur lequel le service cluster est exécuté est choisi parmi les membres disponibles du domaine de basculement sans ordre de priorité.
- Ordered — Ceci vous permet de spécifier un ordre de préférence parmi les membres d'un domaine de basculement. Les domaines de basculement ordered sélectionnent le nœud avec le numéro de priorité le plus bas en premier. Autrement dit, le nœud dans un domaine de basculement avec un numéro de priorité de "1" spécifie la priorité la plus haute, il est ainsi le nœud préféré dans le domaine de basculement. Après ce nœud, le nœud préféré suivant sera le nœud avec le numéro de priorité le plus haut qui suit, et ainsi de suite.
- Failback — Ceci vous permet de spécifier si un service dans le domaine de basculement devrait être restauré sur le nœud sur lequel il était initialement exécuté avant que ce nœud tombe en panne. La configuration de cette caractéristique est utile dans des circonstances où un nœud tombe en panne de manière répétitive et fait partie d'un domaine de basculement ordered. Dans ces circonstances, si un nœud est le nœud préféré dans un domaine de basculement, il est possible qu'un service tombe en panne puis se restaure de manière répétitive entre le nœud préféré et un autre nœud, affectant sévèrement la performance.

NOTE

La caractéristique failback est uniquement applicable si le basculement ordered est configuré.

NOTE

Modifier la configuration d'un domaine de basculement n'a aucun effet sur les services en cours d'exécution.

NOTE

Les domaines de basculement ne sont *pas* requis pour les opérations.

Par défaut, les domaines de basculement sont unrestricted et unordered.

Dans un cluster possédant plusieurs membres, l'utilisation d'un domaine de basculement restricted peut minimiser le travail de paramétrage du cluster pour qu'il exécute un service cluster (comme **`httpd`**), qui requiert que vous paramétriez la configuration de manière identique sur tous les membres exécutant le service cluster. Au lieu de paramétrer le cluster entier afin qu'il exécute le service cluster, il vous est possible de paramétrer uniquement les membres dans le domaine de basculement restricted que vous associerez au service cluster.

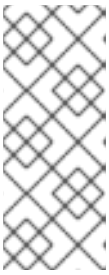
NOTE

Pour configurer un membre préféré, vous pouvez créer un domaine de basculement unrestricted comprenant uniquement un membre du cluster. Faire ceci cause au service cluster de s'exécuter sur ce membre du cluster en premier (le membre préféré), mais permet au service cluster de basculer sur tout autre membre.

Pour configurer un domaine de basculement, utilisez les procédures suivantes :

1. Ouvrez **/etc/cluster/cluster.conf** sur n'importe quel nœud dans le cluster.
2. Ajoutez la section squelette suivante dans l'élément **rm** pour chaque domaine de basculement à utiliser :

```
        <failoverdomains>
            <failoverdomain name="" nofailback="" ordered=""
restricted="">
                <failoverdomainnode name="" priority=""/>
                <failoverdomainnode name="" priority=""/>
                <failoverdomainnode name="" priority=""/>
            </failoverdomain>
        </failoverdomains>
```



NOTE

Le nombre d'attributs **failoverdomainnode** dépend du nombre de nœuds dans le domaine de basculement. La section squelette **failoverdomain** dans le texte précédent affiche trois éléments **failoverdomainnode** (sans aucun nom de nœud spécifié), signifiant ainsi qu'il y a trois nœuds dans le domaine de basculement.

3. Dans la section **failoverdomain**, fournissez les valeurs des éléments et attributs. Pour des descriptions des éléments et attributs, reportez-vous à la section *failoverdomain* du schéma de clusters annoté. Le schéma de clusters annoté est disponible sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**) dans n'importe quel nœud de cluster. Pour voir un exemple de section **failoverdomains**, reportez-vous à l'Exemple 7.8, « Domaine de basculement ajouté à **cluster.conf** ».
4. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en la modifiant de **config_version="2"** à **config_version="3">**).
5. Enregistrez **/etc/cluster/cluster.conf**.
6. **(Optional)** Validez le fichier sur le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Exécutez la commande **`cman_tool version -r`** pour propager la configuration au reste des nœuds de cluster.

8. Procédez à la Section 7.5, « Configurer les services HA ».

L'Exemple 7.8, « Domaine de basculement ajouté à **`cluster.conf`** » propose un exemple de configuration avec un domaine de basculement ordered et unrestricted.

Exemple 7.8. Domaine de basculement ajouté à `cluster.conf`

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
               <failoverdomainnode name="node-03.example.com"
priority="3"/>
           </failoverdomain>
       </failoverdomains>
   </rm>
</cluster>
```

La section **`failoverdomains`** contient une section **`failoverdomain`** pour chaque domaine de

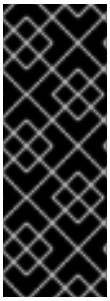
basculement dans le cluster. Cet exemple possède un domaine de basculement. Sur la ligne **failoverdomain**, le nom (**name**) est spécifié en tant que **example_pri**. En outre, il ne spécifie aucune restauration « failback » (**failback="0"**), il spécifie que le basculement est ordered (**ordered="1"**) et que le domaine de basculement est unrestricted (**restricted="0"**).

7.5. CONFIGURER LES SERVICES HA

La configuration des services HA (haute disponibilité, ou « High Availability ») consiste en la configuration des ressources et leur assignement à des services.

Les sections qui suivent décrivent comment modifier **/etc/cluster/cluster.conf** afin d'ajouter des ressources et des services.

- Section 7.5.1, « Ajouter des ressources cluster »
- Section 7.5.2, « Ajouter un service cluster à un cluster »



IMPORTANT

Il peut y avoir un grand éventail de configurations possible avec les ressources et services High Availability. Pour une meilleure compréhension des paramètres et du comportement des ressources, reportez-vous à l'Annexe B, *Paramètres des ressources HA* et à l'Annexe C, *Comportement des ressources HA*. Pour une performance optimale et pour vous assurer que votre configuration peut être prise en charge, contactez un représentant approuvé du support Red Hat.

7.5.1. Ajouter des ressources cluster

Vous povuez configurer deux types de ressources :

- Global — Ressources disponibles à tous les services dans le cluster. Celles-ci sont configurées dans la section **resources** du fichier de configuration (dans l'élément **rm**).
- Service-specific — Ressources disponibles à un seul service. Celles-ci sont configurées dans chaque section **service** du fichier de configuration (dans l'élément **rm**).

Cette section décrit comment ajouter une ressource globale. Pour voir les procédures sur la configuration des ressources spécifiques au service (« service-specific »), reportez-vous à la Section 7.5.2, « Ajouter un service cluster à un cluster ».

Suivez les étapes dans cette section pour ajouter une ressource cluster globale.

1. Ouvrez **/etc/cluster/cluster.conf** sur n'importe quel nœud dans le cluster.
2. Ajoutez une section **resources** dans l'élément **rm**. Par exemple :

```
<rm>
    <resources>

    </resources>
</rm>
```

3. Remplissez-la avec les ressources correspondantes aux services que vous souhaitez créer. Par exemple, voici des ressources à utiliser dans le service Apache. Celles-ci sont composées d'une ressource de système de fichiers (**fs**), d'une ressource IP (**ip**) et d'une ressource Apache (**apache**).

```
    <rm>
        <resources>
           <fs name="web_fs" device="/dev/sdd2"
mountpoint="/var/www" fstype="ext3"/>
           <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf"
name="example_server" server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
    </rm>
```

   L'Exemple 7.9, « Fichier **cluster.conf** avec des ressources ajoutées » montre un exemple du fichier **cluster.conf** avec la section **resources** ajoutée.

4. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en modifiant **config_version="2"** en **config_version="3"**).

5. Enregistrez **/etc/cluster/cluster.conf**.

6. **(Optional)** Validez le fichier sur le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Exécutez la commande **cman_tool version -r** pour propager la configuration au reste des nœuds de cluster.

8. Vérifiez que le fichier de configuration mis à jour a été propagé.

9. Procédez à la Section 7.5.2, « Ajouter un service cluster à un cluster ».

Exemple 7.9. Fichier `cluster.conf` avec des ressources ajoutées

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
```

```
              <device name="apc" port="2"/>
            </method>
          </fence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
             <method name="APC">
               <device name="apc" port="3"/>
             </method>
          </fence>
      </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
               <failoverdomainnode name="node-03.example.com"
priority="3"/>
           </failoverdomain>
       </failoverdomains>
       <resources>
           <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
           <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>

   </rm>
</cluster>
```

7.5.2. Ajouter un service cluster à un cluster

Pour ajouter un service cluster au cluster, suivez les étapes de cette section.

1. Ouvrez **`/etc/cluster/cluster.conf`** sur n'importe quel nœud dans le cluster.

2. Ajoutez une section **`service`** avec l'élément **`rm`** pour chaque service. Par exemple :

```
    <rm>
        <service autostart="1" domain="" exclusive="0" name=""
recovery="restart">
```

```
        </service>
    </rm>
```

3. Configurez les paramètres suivants (attributs) dans l'élément **service** :

   - **autostart** — Spécifie s'il faut démarrer le service automatiquement lorsque le cluster démarre. Veuillez utiliser « 1 » pour activer et « 0 » pour désactiver, le service est activé par défaut.
   - **domain** — Spécifie un domaine de basculement (s'il est requis).
   - **exclusive** — Spécifie une politique où le service s'exécute uniquement sur des nœuds sur lesquels aucun autre service ne s'exécute.
   - **recovery** — Spécifie une stratégie de récupération pour le service. Les options disponibles du service sont déplacer, redémarrer, désactiver, ou redémarrer-désactiver.

4. Selon le type de ressources que vous souhaitez utiliser, remplissez le service avec des ressources globales ou spécifiques au service

   Par exemple, voici un service Apache qui utilise des ressources globales :

```
    <rm>
        <resources>
                <fs name="web_fs" device="/dev/sdd2"
mountpoint="/var/www" fstype="ext3"/>
                <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
                <apache config_file="conf/httpd.conf"
name="example_server" server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
        <service autostart="1" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
                <fs ref="web_fs"/>
                <ip ref="127.143.131.100"/>
                <apache ref="example_server"/>
        </service>
    </rm>
```

   Par exemple, voici un service Apache qui utilise des ressources spécifiques au service :

```
    <rm>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
                <fs name="web_fs2" device="/dev/sdd3"
mountpoint="/var/www2" fstype="ext3"/>
                <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
                <apache config_file="conf/httpd.conf"
name="example_server2" server_root="/etc/httpd" shutdown_wait="0"/>
```

```
        </service>
    </rm>
```

L'Exemple 7.10, « **cluster.conf** avec services ajoutés : l'un utilisant des ressources globales et l'autre utilisant des ressources spécifiques au service » montre un exemple de fichier **cluster.conf** avec deux services :

- **example_apache** — Ce service utilise les ressources globales **web_fs**, **127.143.131.100** et **example_server**.
- **example_apache2** — Ce service utilise les ressources spécifiques au service **web_fs2**, **127.143.131.101** et **example_server2**.

5. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en la modifiant de **config_version="2"** à **config_version="3">**).
6. Enregistrez **/etc/cluster/cluster.conf**.
7. **(Optional)** Validez le fichier mis à jour sur le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

8. Exécutez la commande **cman_tool version -r** pour propager la configuration au reste des nœuds de cluster.
9. Vérifiez que le fichier de configuration mis à jour a été propagé.
10. Procédez à la Section 7.8, « Vérifier une configuration ».

Exemple 7.10. cluster.conf avec services ajoutés : l'un utilisant des ressources globales et l'autre utilisant des ressources spécifiques au service

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
```

```
              <device name="apc" port="3"/>
            </method>
          </fence>
      </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
               <failoverdomainnode name="node-03.example.com"
priority="3"/>
           </failoverdomain>
       </failoverdomains>
       <resources>
           <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
           <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
       </resources>
       <service autostart="1" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
           <fs ref="web_fs"/>
           <ip ref="127.143.131.100"/>
           <apache ref="example_server"/>
       </service>
       <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
           <fs name="web_fs2" device="/dev/sdd3" mountpoint="/var/www2"
fstype="ext3"/>
           <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server2"
server_root="/etc/httpd" shutdown_wait="0"/>
       </service>
   </rm>
</cluster>
```

7.6. CONFIGURER LE PROTOCOLE D'ANNEAU REDONDANT (« REDUNDANT RING »)

À partir de Red Hat Enterprise Linux 6.4, le module complémentaire Red Hat High Availability prend en charge la configuration du protocole d'anneau redondant.

Lorsque vous configurez un système pour qu'il utilise le protocole d'anneau redondant, vous devez prendre en compte les considérations suivantes :

- Ne spécifiez pas plus de deux anneaux.
- Chaque anneau doit utiliser le même protocole, ne mélangez pas IPv4 et IPv6.
- Si nécessaire, vous pouvez spécifier une adresse de multidiffusion pour le deuxième anneau. Si vous spécifiez une adresse de multidiffusion pour le deuxième anneau, l'adresse de multidiffusion alterne ou le port alterne doit être différent de l'adresse de multidiffusion du premier anneau. Si vous ne spécifiez pas d'adresse de multidiffusion alterne, le système utilisera automatiquement une adresse de multidiffusion différente pour le deuxième anneau.

  Si vous spécifiez un port alterne, les numéros de port du premier et second anneau doivent différer d'au moins deux, car le système utilise port et port-1 pour effectuer des opérations.
- N'utilisez pas deux différentes interfaces sur le même sous-réseau.
- En général, il est recommandé de configurer le protocole d'anneau redondant sur deux différents NIC et deux différents commutateurs, au cas où un NIC ou un commutateur échouerait.
- N'utilisez pas la commande **`ifdown`** ou **`service network stop`** pour simuler une panne réseau. Cela détruit le cluster et requiert que vous redmarriez tous les nœuds du cluster pour restaurer.
- N'utilisez pas **`NetworkManager`**, car il exécutera la commande **`ifdown`** si le cable est débranché.
- Lorsqu'un nœud d'un NIC échoue, l'anneau entier est marqué comme étant en échec.
- Aucune intervention manuelle n'est requise pour restaurer après un anneau en échec. Pour effectuer une restauration, vous devrez uniquement corriger la raison originale de l'échec, telle que l'échec d'un NIC ou d'un commutateur.

Pour spécifier une seconde interface réseau à utiliser pour le protocole d'anneau redondant, ajoutez un composant **`altname`** à la section **`clusternode`** du fichier de configuration **`cluster.conf`**. Lorsque vous spécifiez **`altname`**, vous devrez spécifier un attribut **`name`** pour indiquer un second nom d'hôte ou adresse IP pour le nœud.

L'exemple suivant spécifie **`clusternet-node1-eth2`** comme nom alternatif pour le nœud du cluster **`clusternet-node1-eth1`**.

```
<cluster name="mycluster" config_version="3" >
  <logging debug="on"/>
  <clusternodes>
    <clusternode name="clusternet-node1-eth1" votes="1" nodeid="1">
      <fence>
        <method name="single">
          <device name="xvm" domain="clusternet-node1"/>
        </method>
      </fence>
      <altname name="clusternet-node1-eth2"/>
    </clusternode>
```

La section **altname** dans le bloc **clusternode** n'est pas dépendant de sa position. Elle peut se trouver avant ou après la section **fence**. Ne spécifiez pas plus d'un composant **altname** pour un nœud de cluster ou le système échouera au démarrage.

Optionnellement, vouspouvez spécifier une adresse de multidiffusion, un port et un TTL pour le second anneau en incluant un composant **altmulticast** dans la section **cman** du fichier de configuration **cluster.conf**. Le composant **altmulticast** accepte **addr**, **port** et le paramètre **ttl**.

L'exemple suivant affiche la section **cman** d'un fichier de configuration de cluster qui définit une adresse de multidiffusion, un port et un TTL pour le second anneau.

```
<cman>
   <multicast addr="239.192.99.73" port="666" ttl="2"/>
   <altmulticast addr="239.192.99.88" port="888" ttl="3"/>
</cman>
```

7.7. CONFIGURER LES OPTIONS DE DÉBOGAGE

Vous pouvez activer le débogage de tous les démons dans un cluster ou activer la journalisation pour le traitement de cluster spécifique.

Pour activer le débogage de tous les démons, ajoutez ce qui suit au fichier **/etc/cluster/cluster.conf**. Par défaut, la journalisation est dirigée vers le fichier **/var/log/cluster/*démon*.log**.

```
<cluster config_version="7" name="rh6cluster">
  <logging debug="on"/>
   ...
</cluster>
```

Pour activer le débogage des processus individuels, ajoutez les lignes suivantes au fichier **/etc/cluster/cluster.conf**. La configuration de la journalisation « par démon » remplace les paramètres généraux.

```
<cluster config_version="7" name="rh6cluster">
   ...
  <logging>
       <!-- turning on per-subsystem debug logging -->
       <logging_daemon name="corosync" debug="on" />
       <logging_daemon name="fenced" debug="on" />
       <logging_daemon name="qdiskd" debug="on" />
       <logging_daemon name="rgmanager" debug="on" />
       <logging_daemon name="dlm_controld" debug="on" />
       <logging_daemon name="gfs_controld" debug="on" />
  </logging>
   ...
</cluster>
```

Pour la liste des démons pour lesquels vous pouvez activer la journalisation ainsi que pour les options de journalisation supplémentaires que vous pouvez configurer pour la journalisation globale ou « par démon », veuillez vous reporter à la page man **`cluster.conf`**(5).

7.8. VÉRIFIER UNE CONFIGURATION

Une fois que vous avez créé votre fichier de configuration de cluster, vérifiez qu'il fonctionne correctement en effectuant les étapes suivantes :

1. Sur chaque nœud, redémarrez le logiciel du cluster. Cette action vous assurera que toute addition de configuration qui n'est vérifiée qu'au moment du démarrage sera bien incluse dans la configuration en cours d'exécution. Vous pouvez redémarrer le logiciel du cluster en exécutant **`service cman restart`**. Par exemple :

```
[root@example-01 ~]# service cman restart
Stopping cluster:
   Leaving fence domain...                                 [  OK
]
   Stopping gfs_controld...                                [  OK
]
   Stopping dlm_controld...                                [  OK
]
   Stopping fenced...                                      [  OK
]
   Stopping cman...                                        [  OK
]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK
]
   Unmounting configfs...                                  [  OK
]
Starting cluster:
   Checking Network Manager...                             [  OK
]
   Global setup...                                         [  OK
]
   Loading kernel modules...                               [  OK
]
   Mounting configfs...                                    [  OK
]
   Starting cman...                                        [  OK
]
   Waiting for quorum...                                   [  OK
]
   Starting fenced...                                      [  OK
]
   Starting dlm_controld...                                [  OK
]
   Starting gfs_controld...                                [  OK
]
   Unfencing self...                                       [  OK
]
   Joining fence domain...                                 [  OK
]
```

2. Exécutez **`service clvmd start`** si CLVM est utilisé pour créer des volumes clusterisés. Par exemple :

```
[root@example-01 ~]# service clvmd start
Activating VGs:                                            [  OK
]
```

3. Exécutez **`service gfs2 start`** si vous utilisez Red Hat GFS2. Par exemple :

```
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK  ]
```

4. Exécutez **`service rgmanager start`** si vous utilisez des services HA (haute disponibilité, de l'anglais « high-availability »). Par exemple :

```
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK  ]
```

5. Sur n'importe quel nœud de cluster, exécutez **`cman_tool nodes`** pour vérifier que les nœuds fonctionnent en tant que membres dans le cluster (décrit comme « M » dans la colonne du statut « Sts »). Par exemple :

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

6. Sur tout nœud, vérifiez que les services HA fonctionnent bien comme prévu à l'aide de l'utilitaire **`clustat`**. En outre, **`clustat`** affiche le statut des nœuds du cluster. Par exemple :

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)
State
 ------- ----                   ----- ------                    --
---
 service:example_apache         node-01.example.com
started
 service:example_apache2        (none)
disabled
```

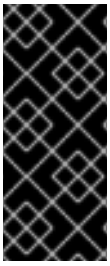
7. Si le cluster fonctionne comme prévu, alors la création du fichier de configuration est terminée.

Vous pouvez gérer le cluster avec les outils de ligne de commande décrits dans le Chapitre 8, *Gérer le module complémentaire Red Hat High Availability avec des outils de ligne de commande*.

CHAPITRE 8. GÉRER LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY AVEC DES OUTILS DE LIGNE DE COMMANDE

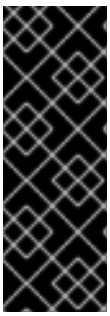
Ce chapitre décrit les diverses tâches administratives pour la gestion du module complémentaire Red Hat High Availability et comporte les sections suivantes :

- Section 8.1, « Démarrer et arrêter le logiciel du cluster »
- Section 8.2, « Ajouter ou supprimer un nœud »
- Section 8.3, « Gérer les services High-Availability »
- Section 8.4, « Mettre à jour une configuration »



IMPORTANT

Assurez-vous que le déploiement du module complémentaire Red Hat High Availability correspond bien à vos besoins et peut être pris en charge. Consultez un représentant autorisé de Red Hat pour vérifier votre configuration avant de la déployer. En outre, prévoyez suffisamment de temps pour une période de rodage de la configuration afin de tester les différents modes d'échec.



IMPORTANT

Ce chapitre fait référence aux éléments et attributs de **`cluster.conf`** communément utilisés. Pour obtenir la liste et la description complète des éléments et attributs **`cluster.conf`**, reportez-vous au schéma des clusters sur **`/usr/share/cluster/cluster.rng`**, et au schéma annoté sur **`/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html`** (par exemple, **`/usr/share/doc/cman-3.0.12/cluster_conf.html`**).

IMPORTANT

Certaines procédures dans ce chapitre appellent à utiliser la commande **`cman_tool version -r`** pour propager une configuration de cluster à travers un cluster. L'utilisation de cette commande requiert que **`ricci`** soit en cours d'exécution.

NOTE

Les procédures dans ce chapitre peuvent inclure des commandes spécifiques pour certains outils en ligne de commande répertoriés dans l'Annexe E, *Résumé des outils de la ligne de commande*. Pour obtenir plus d'informations sur les commandes et les variables, reportez-vous à la page man de chaque outil de ligne de commande.

8.1. DÉMARRER ET ARRÊTER LE LOGICIEL DU CLUSTER

Vous pouvez démarrer ou arrêter un logiciel de cluster sur un nœud selon la Section 8.1.1, « Démarrer un logiciel de cluster » et la Section 8.1.2, « Arrêter un logiciel de cluster ». Démarrer le logiciel de cluster sur un nœud le fait rejoindre le cluster ; arrêter le logiciel de cluster sur un nœud le fait quitter le cluster.

8.1.1. Démarrer un logiciel de cluster

Pour démarrer le logiciel du cluster sur un nœud, saisissez les commandes suivantes dans cet ordre :

1. **`service cman start`**
2. **`service clvmd start`**, si CLVM a été utilisé pour créer des volumes clusterisés
3. **`service gfs2 start`**, si vous utilisez Red Hat GFS2
4. **`service rgmanager start`**, si vous utilisez les services high-availability (HA) (**`rgmanager`**).

Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [  OK  ]
   Global setup...                                         [  OK  ]
   Loading kernel modules...                               [  OK  ]
   Mounting configfs...                                    [  OK  ]
   Starting cman...                                        [  OK  ]
   Waiting for quorum...                                   [  OK  ]
   Starting fenced...                                      [  OK  ]
   Starting dlm_controld...                                [  OK  ]
   Starting gfs_controld...                                [  OK  ]
   Unfencing self...                                       [  OK  ]
   Joining fence domain...                                 [  OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                            [  OK  ]
Activating VG(s):   2 logical volume(s) in volume group "vg_example" now
active
                                                           [  OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK  ]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]#
```

8.1.2. Arrêter un logiciel de cluster

Pour arrêter le logiciel du cluster sur un nœud, saisissez les commandes suivantes dans cet ordre :

1. **`service rgmanager stop`**, si vous utilisez les services high-availability (HA) (**`rgmanager`**).
2. **`service gfs2 stop`**, si vous utilisez Red Hat GFS2
3. **`umount -at gfs2`**, si vous utilisez Red Hat GFS2 en conjonction avec **`rgmanager`**, pour vous assurer que tous les fichiers GFS2 montés pendant le lancement de **`rgmanager`** (mais pas démontés lors de la fermeture) ont bien été démontés.
4. **`service clvmd stop`**, si CLVM a été utilisé pour créer des volumes clusterisés
5. **`service cman stop`**

Par exemple :

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK  ]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK  ]
[root@example-01 ~]# umount -at gfs2
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [  OK  ]
clvmd terminated                                           [  OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [  OK  ]
   Stopping gfs_controld...                                [  OK  ]
   Stopping dlm_controld...                                [  OK  ]
   Stopping fenced...                                      [  OK  ]
   Stopping cman...                                        [  OK  ]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK  ]
   Unmounting configfs...                                  [  OK  ]
[root@example-01 ~]#
```

NOTE

Arrêter un logiciel de cluster sur un nœud cause aux services HA de basculer sur un autre nœud. Comme alternative, prenez en considération la possibilité de déplacer ou de migrer les services HA vers un autre nœud avant d'arrêter le logiciel du cluster. Pour obtenir des informations sur la gestion des services HA, reportez-vous à la Section 8.3, « Gérer les services High-Availability ».

8.2. AJOUTER OU SUPPRIMER UN NŒUD

Cette section décrit comment supprimer un nœud d'un cluster et comment ajouter un nœud à un cluster. Vous pouvez supprimer un nœud d'un cluster selon la Section 8.2.1, « Supprimer un nœud d'un cluster ». Vous pouvez ajouter un nœud à un cluster selon la Section 8.2.2, « Ajouter un nœud à un cluster ».

8.2.1. Supprimer un nœud d'un cluster

Supprimer un nœud d'un cluster consiste en la fermeture du logiciel du cluster sur le nœud à supprimer et en la mise à jour de la configuration du cluster pour refléter la modification.

IMPORTANT

Si la suppression d'un nœud du cluster cause une transition de plus de deux nœuds à deux nœuds, vous devrez redémarrer le logiciel du cluster sur chaque nœud après avoir mis à jour le fichier de configuration du cluster.

Pour supprimer un nœud d'un cluster, procédez aux étapes suivantes :

1. Sur n'importe quel nœud, utilisez l'utilitaire **`clusvcadm`** pour déplacer, migrer, ou arrêter chaque service HA en cours de suppression du cluster qui est exécuté sur le nœud. Pour obtenir plus

d'informations sur l'utilisation de **clusvcadm**, reportez-vous à la Section 8.3, « Gérer les services High-Availability ».

2. Sur le nœud à supprimer du cluster, arrêtez le logiciel du cluster selon la Section 8.1.2, « Arrêter un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK  ]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK  ]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [  OK
]
clvmd terminated                                           [  OK
]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [  OK
]
   Stopping gfs_controld...                                [  OK
]
   Stopping dlm_controld...                                [  OK
]
   Stopping fenced...                                      [  OK
]
   Stopping cman...                                        [  OK
]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK
]
   Unmounting configfs...                                  [  OK
]
[root@example-01 ~]#
```

3. Sur n'importe quel nœud dans le cluster, modifiez **/etc/cluster/cluster.conf** pour supprimer la section **clusternode** du nœud à supprimer. Par exemple, dans l'Exemple 8.1, « Configuration d'un cluster à trois nœuds », si node-03.example.com est censé être supprimé, alors supprimez la section **clusternode** pour ce nœud. Si la suppression d'un (ou plusieurs) nœud(s) cause au cluster de devenir un cluster à deux nœuds, vous pouvez ajouter la ligne suivante au fichier de configuration afin de permettre à un nœud unique de maintenir le quorum (au cas où un nœud échoue) :

   <cman two_node="1" expected_votes="1"/>

   Reportez-vous à la Section 8.2.3, « Exemples de configurations à deux nœuds et à trois nœuds » pour une comparaison entre une configuration à deux nœuds et une configuration à trois nœuds.

4. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en la modifiant de **config_version="2"** à **config_version="3">**).

5. Enregistrez **/etc/cluster/cluster.conf**.

6. **(Optional)** Validez le fichier mis à jour sur le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Exécutez la commande **cman_tool version -r** pour propager la configuration au reste des nœuds de cluster.

8. Vérifiez que le fichier de configuration mis à jour a été propagé.

9. Si le décompte des nœuds du cluster est passé de plus de deux nœuds à deux nœuds, vous devrez redémarrer le logiciel du cluster comme suit :

   1. Sur chaque nœud, arrêtez le logiciel du cluster selon la Section 8.1.2, « Arrêter un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK
]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK
]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK
]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [
OK  ]
clvmd terminated                                           [
OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [
OK  ]
   Stopping gfs_controld...                                [
OK  ]
   Stopping dlm_controld...                                [
OK  ]
   Stopping fenced...                                      [
OK  ]
   Stopping cman...                                        [
OK  ]
   Waiting for corosync to shutdown:                       [  OK
]
   Unloading kernel modules...                             [
OK  ]
   Unmounting configfs...                                  [
OK  ]
[root@example-01 ~]#
```

   2. Sur chaque nœud, démarrez le logiciel du cluster selon la Section 8.1.1, « Démarrer un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
```

```
   Checking Network Manager...                                [
OK  ]
   Global setup...                                            [
OK  ]
   Loading kernel modules...                                  [
OK  ]
   Mounting configfs...                                       [
OK  ]
   Starting cman...                                           [
OK  ]
   Waiting for quorum...                                      [
OK  ]
   Starting fenced...                                         [
OK  ]
   Starting dlm_controld...                                   [
OK  ]
   Starting gfs_controld...                                   [
OK  ]
   Unfencing self...                                          [
OK  ]
   Joining fence domain...                                    [
OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                               [
OK  ]
Activating VG(s):   2 logical volume(s) in volume group
"vg_example" now active
                                                              [
OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                         [  OK
]
Mounting GFS2 filesystem (/mnt/gfsB):                         [  OK
]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                             [  OK
]
[root@example-01 ~]#
```

3. Sur n'importe quel nœud de cluster, exécutez **cman_tool nodes** pour vérifier que les nœuds fonctionnent en tant que membres dans le cluster (décrit comme « M » dans la colonne du statut « Sts »). Par exemple :

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
```

4. Sur tout nœud, vérifiez que les services HA fonctionnent bien comme prévu à l'aide de l'utilitaire **clustat**. En outre, **clustat** affiche le statut des nœuds du cluster. Par exemple :

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate
```

```
 Member Name                              ID   Status
 ------ ----                              ---- ------
 node-02.example.com                          2 Online, rgmanager
 node-01.example.com                          1 Online, Local,
rgmanager

 Service Name                    Owner (Last)
State
 ------- ----                    ----- ------
-----
 service:example_apache          node-01.example.com
started
 service:example_apache2         (none)
disabled
```

8.2.2. Ajouter un nœud à un cluster

Ajouter un nœud à un cluster consiste en la mise à jour de la configuration du cluster, la propagation de la configuration mise à jour vers les nœuds à ajouter et le démarrage du logiciel du cluster sur ce nœud. Pour ajouter un nœud à un cluster, procédez aux étapes suivantes :

1. Sur n'importe quel nœud dans le cluster, modifiez **`/etc/cluster/cluster.conf`** pour ajouter la section **`clusternode`** du nœud à ajouter. Par exemple, dans l'Exemple 8.2, « Configuration d'un cluster à deux nœuds », si node-03.example.com est censé être ajouté, alors ajoutez la section **`clusternode`** pour ce nœud. Si l'ajout d'un (ou plusieurs) nœud(s) cause au cluster de passer d'un cluster à deux nœuds à un cluster à trois nœuds ou plus, supprimez les attributs **`cman`** de **`/etc/cluster/cluster.conf`** comme suit :

   - **`cman two_node="1"`**
   - **`expected_votes="1"`**

   Reportez-vous à la Section 8.2.3, « Exemples de configurations à deux nœuds et à trois nœuds » pour une comparaison entre une configuration à deux nœuds et une configuration à trois nœuds.

2. Mettez à jour l'attribut **`config_version`** en incrémentant sa valeur (par exemple, en la modifiant de **`config_version="2"`** à **`config_version="3">`**).

3. Enregistrez **`/etc/cluster/cluster.conf`**.

4. **`(Optional)`** Validez le fichier mis à jour sur le schéma du cluster (**`cluster.rng`**) en exécutant la commande **`ccs_config_validate`**. Par exemple :

   ```
   [root@example-01 ~]# ccs_config_validate
   Configuration validates
   ```

5. Exécutez la commande **`cman_tool version -r`** pour propager la configuration au reste des nœuds de cluster.

6. Vérifiez que le fichier de configuration mis à jour a été propagé.

7. Propagez le fichier de configuration mis à jour vers **`/etc/cluster/`** dans chaque nœud à ajouter au cluster. Par exemple, utilisez la commande **`scp`** pour envoyer le fichier de configuration mis à jour sur chaque nœud à ajouter au cluster.

8. Si le décompte des nœuds du cluster est passé de deux nœuds à plus de deux nœuds, vous devrez redémarrer le logiciel du cluster dans les nœuds du cluster comme suit :

   1. Sur chaque nœud, arrêtez le logiciel du cluster selon la Section 8.1.2, « Arrêter un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK
]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK
]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK
]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [
OK  ]
clvmd terminated                                           [
OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [
OK  ]
   Stopping gfs_controld...                                [
OK  ]
   Stopping dlm_controld...                                [
OK  ]
   Stopping fenced...                                      [
OK  ]
   Stopping cman...                                        [
OK  ]
   Waiting for corosync to shutdown:                       [  OK
]
   Unloading kernel modules...                             [
OK  ]
   Unmounting configfs...                                  [
OK  ]
[root@example-01 ~]#
```

   2. Sur chaque nœud, démarrez le logiciel du cluster selon la Section 8.1.1, « Démarrer un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [
OK  ]
   Global setup...                                         [
OK  ]
   Loading kernel modules...                               [
OK  ]
   Mounting configfs...                                    [
OK  ]
   Starting cman...                                        [
OK  ]
   Waiting for quorum...                                   [
OK  ]
```

```
   Starting fenced...                                         [
OK  ]
   Starting dlm_controld...                                   [
OK  ]
   Starting gfs_controld...                                   [
OK  ]
   Unfencing self...                                          [
OK  ]
   Joining fence domain...                                    [
OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                               [
OK  ]
Activating VG(s):   2 logical volume(s) in volume group
"vg_example" now active
                                                              [
OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                         [  OK
]
Mounting GFS2 filesystem (/mnt/gfsB):                         [  OK
]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                             [  OK
]
[root@example-01 ~]#
```

9. Sur chaque nœud à ajouter au cluster, démarrez le logiciel du cluster selon la Section 8.1.1, « Démarrer un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                            [  OK
]
   Global setup...                                        [  OK
]
   Loading kernel modules...                              [  OK
]
   Mounting configfs...                                   [  OK
]
   Starting cman...                                       [  OK
]
   Waiting for quorum...                                  [  OK
]
   Starting fenced...                                     [  OK
]
   Starting dlm_controld...                               [  OK
]
   Starting gfs_controld...                               [  OK
]
   Unfencing self...                                      [  OK
]
   Joining fence domain...                                [  OK
]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                           [  OK
```

```
]
Activating VG(s):   2 logical volume(s) in volume group "vg_example"
now active
                                                           [  OK
]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK  ]

[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]#
```

10. Sur n'importe quel nœud et à l'aide de l'utilitaire **clustat**, vérifiez que chaque nœud ajouté est en cours d'exécution et fait partie du cluster. Par exemple :

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                              ID   Status
 ------ ----                              ---- ------
 node-03.example.com                          3 Online, rgmanager
 node-02.example.com                          2 Online, rgmanager
 node-01.example.com                          1 Online, Local,
rgmanager

 Service Name                    Owner (Last)
State
 ------- ----                    ----- ------                    --
---
 service:example_apache          node-01.example.com
started
 service:example_apache2         (none)
disabled
```

Pour obtenir des informations sur l'utilisation de **clustat**, reportez-vous à la Section 8.3, « Gérer les services High-Availability ».

En outre, vous pouvez utiliser **cman_tool status** pour vérifier les votes de nœuds, le compte des nœuds, et le compte quorum. Par exemple :

```
[root@example-01 ~]#cman_tool status
Version: 6.2.0
Config Version: 19
Cluster Name: mycluster
Cluster Id: 3794
Cluster Member: Yes
Cluster Generation: 548
Membership state: Cluster-Member
Nodes: 3
Expected votes: 3
Total votes: 3
Node votes: 1
Quorum: 2
```

```
Active subsystems: 9
Flags:
Ports Bound: 0 11 177
Node name: node-01.example.com
Node ID: 3
Multicast addresses: 239.192.14.224
Node addresses: 10.15.90.58
```

11. Sur n'importe quel nœud, vous pouvez vous servir de l'utilitaire **clusvcadm** pour migrer ou déplacer un service en cours d'exécution sur le nouveau nœud du cluster. Vous pouvez aussi activer tout service désactivé. Pour obtenir des informations sur l'utilisation de **clusvcadm**, reportez-vous à la Section 8.3, « Gérer les services High-Availability ».

8.2.3. Exemples de configurations à deux nœuds et à trois nœuds

Reportez-vous aux exemples suivants pour des comparaisons entre les configurations à deux nœuds et les configurations à trois nœuds.

Exemple 8.1. Configuration d'un cluster à trois nœuds

```
<cluster name="mycluster" config_version="3">
   <cman/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
              <failoverdomainnode name="node-01.example.com"
```

```
priority="1"/>
                <failoverdomainnode name="node-02.example.com"
priority="2"/>
                <failoverdomainnode name="node-03.example.com"
priority="3"/>
            </failoverdomain>
        </failoverdomains>
        <resources>
            <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
            <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
            <fs ref="web_fs"/>
            <ip ref="127.143.131.100"/>
            <apache ref="example_server"/>
        </service>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
            <fs name="web_fs2" device="/dev/sdd3" mountpoint="/var/www"
fstype="ext3"/>
            <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server2"
server_root="/etc/httpd" shutdown_wait="0"/>
        </service>
    </rm>
</cluster>
```

Exemple 8.2. Configuration d'un cluster à deux nœuds

```
<cluster name="mycluster" config_version="3">
    <cman two_node="1" expected_votes="1"/>
    <clusternodes>
      <clusternode name="node-01.example.com" nodeid="1">
          <fence>
             <method name="APC">
               <device name="apc" port="1"/>
              </method>
          </fence>
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
             <method name="APC">
               <device name="apc" port="2"/>
             </method>
          </fence>
    </clusternodes>
```

```
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
           </failoverdomain>
       </failoverdomains>
       <resources>
           <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
           <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
       </resources>
       <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
           <fs ref="web_fs"/>
           <ip ref="127.143.131.100"/>
           <apache ref="example_server"/>
       </service>
       <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
           <fs name="web_fs2" device="/dev/sdd3" mountpoint="/var/www"
fstype="ext3"/>
           <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server2"
server_root="/etc/httpd" shutdown_wait="0"/>
       </service>
   </rm>
</cluster>
```

8.3. GÉRER LES SERVICES HIGH-AVAILABILITY

Vous pouvez gérer les services high-availability en utilisant **Cluster Status Utility**, **`clustat`**, et **Cluster User Service Administration Utility**, **`clusvcadm`**. **`clustat`** affiche l'état d'un cluster et **`clusvcadm`** fournit possibilité de gérer les services high-availability.

Cette section fournit des informations de base sur la gestion des services HA à l'aide des commandes **`clustat`** et **`clusvcadm`**. Celle-ci comporte les sous-sections suivantes :

- Section 8.3.1, « Afficher l'état du service HA avec **`clustat`** »
- Section 8.3.2, « Gérer les services HA avec **`clusvcadm`** »

8.3.1. Afficher l'état du service HA avec `clustat`

`clustat` affiche l'état global du cluster. Il est ainsi possible de voir les informations sur l'adhésion, le quorum, l'état de tous les services high-availability (haute disponibilité), **`clustat`** indique aussi le nœud sur lequel la commande **`clustat`** est exécutée (Local). Le Tableau 8.1, « État des services » décrit les états dans lesquels les services peuvent se trouver, ceux-ci s'affichent lors de l'exécution de **`clustat`**. L'Exemple 8.3, « Écran **`clustat`** » montre un exemple de l'écran de **`clustat`**. Pour obtenir de plus amples informations sur l'exécution de la commande **`clustat`**, reportez-vous à la page man **`clustat`**.

Tableau 8.1. État des services

État des services	Description
Started	Les ressources d'un service sont configurées et disponibles sur le système du cluster propriétaire du service.
Recovering	Le service est en attente de démarrage sur un autre nœud.
Disabled	Le service a été désactivé et n'a pas de propriétaire qui lui est assigné. Un service désactivé n'est jamais redémarré automatiquement par le cluster.
Stopped	Dans l'état arrêté, le service sera évalué pour démarrer après le service suivant ou la transition de nœud. Ceci est un état temporaire. Vous pouvez activer ou désactiver le service de cet état.
Failed	Le service est présumé mort. Un service est placé dans cet état lorsque l'opération *stop* d'une ressource échoue. Une fois que le service se trouve dans cet état, vous devez vérifier qu'aucune ressource n'est allouée (par exemple, des systèmes de fichiers montés) avant d'effectuer une requête **`disable`**. La seule opération pouvant s'effectuer lorsqu'un service est entré dans cet état est **`disable`**..
Uninitialized	Cet état peut apparaître dans certains cas lors du démarrage et de l'exécution de **`clustat -f`**.

Exemple 8.3. Écran `clustat`

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:15 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)                   State
 ------- ----                   ----- ------                   -----
```

```
 service:example_apache         node-01.example.com           started
 service:example_apache2        (none)
disabled
```

8.3.2. Gérer les services HA avec `clusvcadm`

Vous pouvez gérer les services HA en utilisant la commande **`clusvcadm`**. Avec celle-ci, vous pouvez effectuer les opérations suivantes :

- Activer et lancer un service.
- Désactiver un service.
- Arrêter un service.
- Geler un service
- Dégeler un service
- Migrer un service (uniquement pour les services de machines virtuelles)
- Déplacer un service.
- Redémarrer un service.

Le Tableau 8.2, « Opérations des services » décrit les opérations avec plus de détails. Pour une description complète de la démarche à suivre pour effectuer ces opérations, reportez-vous à la page man de l'utilitaire **`clusvcadm`**.

Tableau 8.2. Opérations des services

Opération du service	Description	Syntaxe de la commande
Enable	Lance le service, optionnellement sur une cible préférée et optionnellement selon les règles du domaine de basculement. En l'absence d'une cible préférée ou de règles de domaine de basculement, l'hôte local où **`clusvcadm`** est exécuté lancera le service. Si l'opération d'origine *start* échoue, le service se comportera comme si l'opération *relocate* avait été requise (reportez-vous à **Relocate** dans ce tableau). Si l'opération fonctionne, le service sera placé dans l'état « started ».	**`clusvcadm -e <service_name>`** ou **`clusvcadm -e <service_name> -m <member>`** (L'utilisation de l'option -m option spécifie le membre-cible préféré sur lequel lancer le service.)
Disable	Arrête le service et le place dans un état déasctivé. Ceci est l'unique opération permise lorsqu'un service est dans l'état *failed*.	**`clusvcadm -d <service_name>`**

Opération du service	Description	Syntaxe de la commande
Relocate	Déplace le service sur un autre nœud. Optionnellement, vous pouvez spécifier un nœud préféré pour recevoir le service, mais l'incapacité du service à s'exécuter sur cet hôte (par exemple, si le service ne parvient pas à démarrer ou si l'hôte est hors-ligne) n'empêche pas le déplacement, et un autre nœud est choisi. **`rgmanager`** tente de démarrer le service sur n'importe quel nœud permis dans le cluster. Si aucun nœud-cible permis dans le cluster ne démarre le service, le déplacement échoue et le service tente d'être redémarré sur le propriétaire d'origine. Si le propriétaire d'origine ne peut pas redémarrer le service, alors le service est placé dans l'état *stopped*.	**`clusvcadm -r <service_name>`** ou **`clusvcadm -r <service_name> -m <member>`** (L'utilisation de l'option -m spécifie le membre-cible préféré sur lequel lancer le service.)
Stop	Arrête le service et le place dans l'état *stopped*.	**`clusvcadm -s <service_name>`**
Freeze	Gèle un service sur le nœud sur lequel il est en cours d'exécution. Ceci empêche les vérifications d'état du service ainsi que les basculements au cas où le nœud échouerait ou si rgmanager était arrêté. Ceci peut être utilisé pour suspendre un service afin de permettre la maintenance des ressources sous-jacentes. Reportez-vous à la section intitulée « Considérations pour l'utilisation des opérations Freeze et Unfreeze » pour obtenir des informations importantes sur l'utilisation des opérations *freeze* et *unfreeze*.	**`clusvcadm -Z <service_name>`**
Unfreeze	Unfreeze retire un service de l'état *freeze*. ceci ré-active les vérifications d'état. Reportez-vous à la section intitulée « Considérations pour l'utilisation des opérations Freeze et Unfreeze » pour obtenir d'importantes informations sur l'utilisation des opérations *freeze* et *unfreeze*.	**`clusvcadm -U <service_name>`**

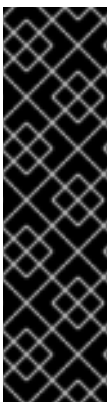
Opération du service	Description	Syntaxe de la commande
Migrate	Migre une machine virtuelle sur un autre nœud. Vous devez spécifier un nœud-cible. Selon l'échec, un échec de migration peut résulter en la machine virtuelle se trouvant dans l'état *failed* ou dans l'état « started » dans le propriétaire d'origine.	`clusvcadm -M <service_name> -m <member>`  **IMPORTANT** Pour l'opération *migrate*, vous *devez spécifier* un nœud-cible à l'aide de l'option `-m <member>`.
Restart	Redémarre un service sur le nœud sur lequel il est actuellement en cours d'exécution.	`clusvcadm -R <service_name>`

Considérations pour l'utilisation des opérations Freeze et Unfreeze

L'utilisation de l'opération *freeze* permet la maintenance de certaines parties des services `rgmanager`. Par exemple, si vous possédez une base de données et un serveur web dans un service `rgmanager`, vous pouvez geler le service `rgmanager`, arrêter la base de données, effectuer la maintenance, redémarrer la base de données, puis dégeler le service.

Lorsqu'un service est gelé, il se comporte comme suit :

- Les vérifications d'*État* sont désactivées.
- Les opérations *Start* sont désactivées.
- Les opérations *Stop* sont désactivées.
- Le basculement ne se produira pas (même si vous éteignez le propriétaire du service).



IMPORTANT

Ne pas suivre ces directives peut faire que les ressources soient allouées sur plusieurs hôtes :

- Vous *ne devriez pas* arrêter toutes les instances de rgmanager lorsqu'un service est gelé, à moins que vous ne planifiez de redémarrer les hôtes avant de relancer rgmanager.
- Vous *ne devriez pas* dégeler un service avant que le propriétaire du service ne rejoigne le cluster et qu'il ne redémarre rgmanager.

8.4. METTRE À JOUR UNE CONFIGURATION

Mettre à jour la configuration d'un cluster consiste en la modification du fichier de configuration du cluster (`/etc/cluster/cluster.conf`) et en sa propagation vers chaque nœud dans le cluster. Vous pouvez mettre à jour la configuration en utilisant l'une des procédures suivantes :

- Section 8.4.1, « Mettre à jour une configuration à l'aide de`cman_tool version -r` »

- Section 8.4.2, « Mettre à jour une configuration à l'aide de **scp** »

8.4.1. Mettre à jour une configuration à l'aide de `cman_tool version -r`

Pour mettre à jour la configuration à l'aide de la commande **cman_tool version -r**, procédez aux étapes suivantes :

1. Sur tout nœud dans le cluster, modifiez le fichier **/etc/cluster/cluster.conf**.
2. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en la modifiant de **config_version="2"** à **config_version="3">**).
3. Enregistrez **/etc/cluster/cluster.conf**.
4. Exécutez la commande **cman_tool version -r** pour propager la configuration au reste des nœuds du cluster. Il est nécessaire que **ricci** soit en cours d'exécution dans chaque nœud de cluster afin de pouvoir propager les informations mises à jour de la configuration du cluster.
5. Vérifiez que le fichier de configuration mis à jour a été propagé.
6. Vous pouvez ignorer cette étape (redémarrer le logiciel du cluster) si vous avez uniquement effectué les changements de configuration suivants :
   - Supprimer un nœud de la configuration du cluster — *Sauf* si le décompte des nœuds passe de plus de deux nœuds à deux nœuds. Pour obtenir des informations sur la suppression d'un nœud d'un cluster et sur la transition de plus de deux nœuds à deux nœuds, reportez-vous à la Section 8.2, « Ajouter ou supprimer un nœud ».
   - Ajouter un nœud à la configuration du cluster — *Sauf* si le décompte des nœuds passe de deux nœuds à plus de deux nœuds. Pour obtenir des informations sur l'ajout d'un nœud à un cluster et sur la transition de deux nœuds à plus de deux nœuds, reportez-vous à la Section 8.2.2, « Ajouter un nœud à un cluster ».
   - Modifier la manière dont les démons journalisent les informations.
   - Ajout, modification, ou suppression d'un service HA ou de la maintenance VM.
   - Ajout, modification, ou suppression de la maintenance des ressources.
   - Ajout, modification, ou suppression de la maintenance du domaine de basculement.

   Sinon, vous devrez redémarrer le logiciel du cluster comme suit :

   1. Sur chaque nœud, arrêtez le logiciel du cluster selon la Section 8.1.2, « Arrêter un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK
]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK
]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK
]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [
```

```
OK  ]
clvmd terminated                                           [
OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [
OK  ]
   Stopping gfs_controld...                                [
OK  ]
   Stopping dlm_controld...                                [
OK  ]
   Stopping fenced...                                      [
OK  ]
   Stopping cman...                                        [
OK  ]
   Waiting for corosync to shutdown:                       [  OK
]
   Unloading kernel modules...                             [
OK  ]
   Unmounting configfs...                                  [
OK  ]
[root@example-01 ~]#
```

2. Sur chaque nœud, démarrez le logiciel du cluster selon la Section 8.1.1, « Démarrer un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [
OK  ]
   Global setup...                                         [
OK  ]
   Loading kernel modules...                               [
OK  ]
   Mounting configfs...                                    [
OK  ]
   Starting cman...                                        [
OK  ]
   Waiting for quorum...                                   [
OK  ]
   Starting fenced...                                      [
OK  ]
   Starting dlm_controld...                                [
OK  ]
   Starting gfs_controld...                                [
OK  ]
   Unfencing self...                                       [
OK  ]
   Joining fence domain...                                 [
OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                            [
OK  ]
Activating VG(s):   2 logical volume(s) in volume group
"vg_example" now active
                                                           [
```

```
OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK
]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK
]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK
]
[root@example-01 ~]#
```

Arrêter et démarrer le logiciel du cluster assure que toutes les modifications de la configuration, qui ne sont vérifiées qu'au démarrage, sont bien incluses dans la configuration en cours d'exécution.

7. Sur n'importe quel nœud de cluster, exécutez **cman_tool nodes** pour vérifier que les nœuds fonctionnent en tant que membres dans le cluster (décrit comme « M » dans la colonne du statut « Sts »). Par exemple :

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

8. Sur tout nœud, vérifiez que les services HA fonctionnent bien comme prévu à l'aide de l'utilitaire **clustat**. En outre, **clustat** affiche le statut des nœuds du cluster. Par exemple :

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)
State
 ------- ----                   ----- ------                    --
---
 service:example_apache         node-01.example.com
started
 service:example_apache2        (none)
disabled
```

9. Si le cluster s'exécute comme prévu, vous avez terminé de mettre à jour la configuration.

8.4.2. Mettre à jour une configuration à l'aide de scp

Pour mettre à jour la configuration à l'aide de la commande **scp**, procédez aux étapes suivantes :

1. Sur chaque nœud, arrêtez le logiciel du cluster selon la Section 8.1.2, « Arrêter un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK  ]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK  ]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [  OK
]
clvmd terminated                                           [  OK
]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [  OK
]
   Stopping gfs_controld...                                [  OK
]
   Stopping dlm_controld...                                [  OK
]
   Stopping fenced...                                      [  OK
]
   Stopping cman...                                        [  OK
]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK
]
   Unmounting configfs...                                  [  OK
]
[root@example-01 ~]#
```

2. Sur tout nœud dans le cluster, modifiez le fichier **/etc/cluster/cluster.conf**.

3. Mettez à jour l'attribut **config_version** en incrémentant sa valeur (par exemple, en la modifiant de **config_version="2"** à **config_version="3">**).

4. Enregistrez **/etc/cluster/cluster.conf**.

5. Validez le fichier mis à jour avec le schéma du cluster (**cluster.rng**) en exécutant la commande **ccs_config_validate**. Par exemple :

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

6. Si le fichier mis à jour est valide, utilisez la commande **scp** pour le propager sur **/etc/cluster/** dans chaque nœud du cluster.:

7. Vérifiez que le fichier de configuration mis à jour a été propagé.

8. Sur chaque nœud, démarrez le logiciel du cluster selon la Section 8.1.1, « Démarrer un logiciel de cluster ». Par exemple :

```
[root@example-01 ~]# service cman start
Starting cluster:
```

```
   Checking Network Manager...                                [  OK
]
   Global setup...                                            [  OK
]
   Loading kernel modules...                                  [  OK
]
   Mounting configfs...                                       [  OK
]
   Starting cman...                                           [  OK
]
   Waiting for quorum...                                      [  OK
]
   Starting fenced...                                         [  OK
]
   Starting dlm_controld...                                   [  OK
]
   Starting gfs_controld...                                   [  OK
]
   Unfencing self...                                          [  OK
]
   Joining fence domain...                                    [  OK
]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                               [  OK
]
Activating VG(s):   2 logical volume(s) in volume group "vg_example"
now active
                                                              [  OK
]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                         [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                         [  OK  ]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                             [  OK  ]
[root@example-01 ~]#
```

9. Sur n'importe quel nœud de cluster, exécutez **cman_tool nodes** pour vérifier que les nœuds fonctionnent en tant que membres dans le cluster (décrit comme « M » dans la colonne du statut « Sts »). Par exemple :

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

10. Sur tout nœud, vérifiez que les services HA fonctionnent bien comme prévu à l'aide de l'utilitaire **clustat**. En outre, **clustat** affiche le statut des nœuds du cluster. Par exemple :

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
```

```
 node-03.example.com                          3 Online, rgmanager
 node-02.example.com                          2 Online, rgmanager
 node-01.example.com                          1 Online, Local,
rgmanager

 Service Name                    Owner (Last)
State
 ------- ----                    ----- ------                    --
---
 service:example_apache          node-01.example.com
started
 service:example_apache2         (none)
disabled
```

11. Si le cluster s'exécute comme prévu, vous avez terminé de mettre à jour la configuration.

CHAPITRE 9. DIAGNOSTIQUER ET CORRIGER DES PROBLÈMES DANS UN CLUSTER

Les problèmes de clusters, par nature, peuvent être difficiles à résoudre. Cela est dû à la complexité présentée par un cluster de systèmes comparé au diagnostic de problèmes sur un système unique. Toutefois, il existe des problèmes communs que les administrateurs système pourraient plus fréquemment rencontrer lors du déploiement ou de l'administration d'un cluster. Comprendre comment aborder ces problèmes communs peut grandement faciliter le déploiement et l'administration d'un cluster.

Ce chapitre fournit des informations sur certains problèmes communs de clusters et sur la manière de les résoudre. Davantage d'aide peut être trouvée dans la base de connaissances et en contactant un représentant autorisé Red Hat. Si le problème est plus particulièrement lié au système de fichiers GFS2, vous pourrez trouver des informations à ce sujet dans le document *Global File System 2*.

9.1. LES CHANGEMENTS DE CONFIGURATION NE PRENNENT PAS EFFET

Lorsque vous effectuez des changements à la configuration du cluster, vous devez propager ceux-ci sur chaque nœud dans le cluster.

- Lorsque vous configurez un cluster à l'aide de **Conga**, **Conga** propage les modifications automatiquement lorsque vous les appliquez.
- Pour obtenir des informations sur la propagation des modifications apportées à la configuration du cluster avec la commande **ccs**, reportez-vous à la Section 5.15, « Propager le fichier de configuration sur les nœuds du cluster ».
- Pour obtenir des informations sur la propagation des modifications apportées à la configuration du cluster avec des outils de ligne de commande, reportez-vous à la Section 8.4, « Mettre à jour une configuration ».

Si vous apportez l'une des modifications suivantes à votre cluster, il ne sera pas nécessaire de redémarrer le cluster après les avoir propagé afin qu'elles prennent effet.

- Supprimer un nœud de la configuration du cluster — *Sauf* si le décompte des nœuds passe de plus de deux nœuds à deux nœuds.
- Ajouter un nœud à la configuration du cluster — *Sauf* si le décompte des nœuds passe de deux nœuds à plus de deux nœuds.
- Modifier les paramètres de connexion.
- Ajouter, modifier ou supprimer des services HA ou des composants VM.
- Ajouter, modifier ou supprimer des ressources cluster.
- Ajouter, modifier ou supprimer des domaines de basculement.

Si vous procédez à toute autre modification de configuration sur votre cluster, vous devrez redémarrer le cluster pour que ces modifications soient implémentées. Les modifications de configuration de cluster qui suivent requièrent un redémarrage du cluster pour prendre effet :

- L'ajout ou la suppression de l'option **two_node** du fichier de configuration du cluster.

- Renommer le cluster.
- La modification de l'une des horloges de **corosync** ou **openais**.
- Ajouter, modifier ou supprimer des heuristiques du disque quorum, la modification de n'importe quelle horloge du disque quorum, ou la modification du périphérique disque du quorum. Un redémarrage complet du démon **qdiskd** est nécessaire pour que ces modifications prennent effet.
- La modification du mode **central_processing** pour **rgmanager**. Un redémarrage complet de **rgmanager** est nécessaire pour que ces changements prennent effet.
- La modification de l'adresse de multidiffusion.
- Le basculement du mode de transport de la multiffusion UDP à la monodiffusion UDP, ou de la monodiffusion UDP à la multiffusion UDP.

Vous pouvez redémarrer le cluster à l'aide de **Conga**, de la commande **ccs** ou des outils de ligne de commande,

- Pour obtenir des informations sur le redémarrage d'un cluster avec **Conga**, reportez-vous à la Section 4.4, « Démarrer, arrêter, redémarrer et supprimer des clusters ».
- Pour obtenir des informations sur le redémarrage d'un cluster avec la commande **ccs**, reportez-vous à la Section 6.2, « Démarrer et arrêter un cluster ».
- Pour obtenir des informations sur le redémarrage d'un cluster avec des outils de ligne de commande, reportez-vous à la Section 8.1, « Démarrer et arrêter le logiciel du cluster ».

9.2. LE CLUSTER NE SE FORME PAS

Si vous ne parvenez pas à former un nouveau cluster, vérifiez ce qui suit :

- Assurez-vous que la résolution de nom est correctement paramétrée. Le nom du nœud du cluster dans le fichier **cluster.conf** devrait correspondre au nom utilisé pour résoudre l'adresse de ce cluster sur le réseau que le cluster utilisera pour communiquer. Par exemple, si les noms des nœuds du cluster sont **nodea** et **nodeb**, assurez-vous que les deux nœuds possèdent bien des entrées dans les fichiers **/etc/cluster/cluster.conf** et **/etc/hosts** qui correspondent à ces noms.
- Si le cluster utilise la multidiffusion pour les communications entre nœuds, assurez-vous que le trafic de multidiffusion n'est pas bloqué, retardé ou que rien ne soit en train d'interférer avec sur le réseau que le cluster utilise pour communiquer. Notez que certains interrupteurs Cisco possèdent des fonctionnalités pouvant provoquer des délais au trafic de multidiffusion.
- Utilisez **telnet** ou **SSH** pour vérifier si vous pouvez atteindre des nœuds distants.
- Exécutez la commande **ethtool eth1 | grep link** pour vérifier si le lien ethernet fonctionne.
- Utilisez la commande **tcpdump** sur chaque nœud pour vérifier le trafic du réseau.
- Assurez-vous qu'aucune règle de pare-feu ne bloque les communications entre les nœuds.

- Assurez-vous que les interfaces utilisées par le cluster pour les communications inter-nœuds ne soient utilisées par aucun autre mode de liaison que les modes 0, 1, ou 2. (Les modes de liaison 0 et 2 sont pris en charge à partir de Red Hat Enterprise Linux 6.4.)

9.3. NŒUDS INCAPABLES DE REJOINDRE LE CLUSTER APRÈS UN CLÔTURAGE (FENCING) OU UN REDÉMARRAGE

Si les nœuds ne rejoignent pas le cluster après un clôturage (fencing) ou un redémarrage, vérifiez ce qui suit :

- Les clusters dont le trafic passe par un interrupteur Cisco Catalyst peuvent rencontrer ce problème.
- Assurez-vous que tous les nœuds du cluster possèdent la même version du fichier **cluster.conf**. Si le fichier **cluster.conf** est différent sur n'importe quel nœud, alors ces nœuds pourraient ne pas être en mesure de rejoindre le cluster après le fencing.

  À partir de Red Hat Enterprise Linux 6.1, vous pouvez utiliser la commande suivante pour vérifier que tous les nœuds spécifiés dans le fichier de configuration du cluster de l'hôte possèdent bien un fichier de configuration du cluster identique :

  ```
  ccs -h host --checkconf
  ```

  Pour obtenir des informations sur la commande **ccs**, voir le Chapitre 5, *Configurer le module complémentaire Red Hat High Availability avec la commande **ccs*** et le Chapitre 6, *Gérer le module complémentaire Red Hat High Availability avec **ccs***.
- Assurez-vous de bien avoir configuré **chkconfig on** pour les services clusters dans le nœud qui tente de rejoindre le cluster.
- Assurez-vous qu'aucune règle de pare-feu ne bloque les communications entre le nœud et les autres nœuds dans le cluster.

9.4. ÉCHEC DU DÉMON CLUSTER

RGManager possède un processus de surveillance qui redémarre l'hôte au cas où le processus **rgmanager** principal échouerait de manière inattendue. Ceci entraîne le fencing du nœud du cluster et la récupération du service par **rgmanager** sur un autre hôte. Lorsque le démon de surveillance détecte que le processus **rgmanager** principal est en panne, il redémarrera le nœud du cluster, puis les nœuds actifs du cluster détecteront que le nœud est parti et l'expulseront du cluster.

Le numéro de PID (*process ID*) le plus bas est le processus de surveillance qui effectuera une action si son processus enfant (processus avec un numéro de PID plus élevé) échoue. Capturer le cœur du processus avec le numéro de PID le plus haut en utilisant **gcore** peut vous aider à résoudre un démon en échec.

Installez les paquetages requis pour capturer et afficher le « core » (cœur), puis assurez-vous que les versions de **rgmanager** et **rgmanager-debuginfo** sont bien les mêmes, sinon le « core » (cœur) de l'application capturée pourrait se révéler inutilisable.

```
$ yum -y --enablerepo=rhel-debuginfo install gdb rgmanager-debuginfo
```

9.4.1. Capturer le « core » (cœur) de rgmanager lors du runtime.

Il existe deux processus **rgmanager** exécutés lors du démarrage. Vous devez capturer le « core » (cœur) du processus **rgmanager** avec le numéro PID le plus élevé.

Ci-dessous figure un exemple de sortie de la commande **ps**affichant deux processus de **rgmanager**.

```
$ ps aux | grep rgmanager | grep -v grep

root    22482  0.0  0.5  23544  5136 ?        S<Ls Dec01   0:00 rgmanager
root    22483  0.0  0.2  78372  2060 ?        S<l  Dec01   0:47 rgmanager
```

Dans l'exemple suivant, le programme **pidof** est utilisé pour déterminer automatiquement le numéro PID le plus élevé, qui est le PID correct pour créer le « core » (cœur). La commande complète capture le « core » de l'application du processus 22483, qui possède le PID le plus élevé.

```
$ gcore -o /tmp/rgmanager-$(date '+%F_%s').core $(pidof -s rgmanager)
```

9.4.2. Capturer le « core » (cœur) lorsque le démon échoue

Par défaut, le script **/etc/init.d/functions** bloque les fichiers principaux des démons appelés par **/etc/init.d/rgmanager**. Pour que le démon crée des « core » (cœurs) d'applications, vous devez activer cette option. Cette procédure doit être effectuée sur tous les nœuds des clusters dont le « core » d'application doit être capturé.

Pour créer un fichier cœur lorsque le démon rgmanager tombe en panne, veuillez modifier le fichier **/etc/sysconfig/cluster**. Le paramètre ***DAEMONCOREFILELIMIT*** permet au démon de créer des fichiers cœurs si le processus tombe en panne. Il existe une option, **-w**, qui empêche le processus de surveillance de s'exécuter. Le démon de surveillance est responsable du redémarrage du nœud du cluster si **rgmanager** tombe en panne et dans certains cas, si le démon est en cours d'exécution, alors le fichier cœur ne sera pas généré, il doit ainsi être désactivé afin de capturer les fichiers cœurs.

```
DAEMONCOREFILELIMIT="unlimited"
RGMGR_OPTS="-w"
```

Redémarrez rgmanager pour activer les nouvelles options de configuration :

```
service rgmanager restart
```

NOTE

Si les services du cluster sont exécutés sur ce nœud de cluster, alors les services exécutés pourraient se retrouver en mauvais état.

Le fichier « core » sera écrit lorsqu'il est généré par l'échec du processus **rgmanager**

```
ls /core*
```

La sortie devrait être similaire à ce qui suit :

```
/core.11926
```

Déplacez ou supprimez tous les anciens fichiers « core » sous le répertoire / avant de redémarrer **rgmanager** pour capturer le « core » de l'application. Le nœud du cluster qui a expérimenté l'échec de **rgmanager** devra être redémarré ou clôturé (« fenced ») une fois que le « core » est capturé afin d'être sûr que le processus de surveillance n'était pas en cours d'exécution.

9.4.3. Enregistrement d'une session de backtrace gdb

Une fois que vous avez capturé le fichier « core », vous pouvez voir son contenu en utilisant **gdb**, le débogueur GNU. Pour enregistrer une session script de **gdb** sur le fichier « core » du système affecté, veuillez exécuter ce qui suit :

```
$ script /tmp/gdb-rgmanager.txt
$ gdb /usr/sbin/rgmanager /tmp/rgmanager-.core.
```

Ceci lancera une session **gdb**, tandis que **script** l'enregistrera sur le fichier texte correspondant. Lorsque vous êtes dans une session **gdb**, exécutez les commandes suivantes :

```
(gdb) thread apply all bt full
(gdb) quit
```

Pressez sur **ctrl-D** pour arrêter la session script et l'enregistrer sur le fichier texte.

9.5. SUSPENSION DES SERVICES DU CLUSTER

Lorsque les services du cluster tentent de clôturer un nœud, les services du cluster s'arrêtent jusqu'à ce que l'opération fence se termine. Ainsi, si le stockage ou les services contrôlés par le cluster restent suspendus et que les nœuds du cluster affichent différentes vues de l'adhésion au cluster, ou si le cluster est suspendu lorsque vous tentez de clôturer un nœud et que vous devez redémarrer des nœuds pour la récupération, vérifiez les conditions suivantes :

- Le cluster a peut-être tenté de clôturer un nœud et l'opération fence a peut-être échouée.
- Observez le fichier **/var/log/messages** sur tous les nœuds et voyez s'il y a des messages d'échec du fencing. S'il y en a, alors redémarrez les nœuds dans le cluster et configurez le fencing correctement.
- Vérifiez qu'une partition du réseau ne s'est pas produite, comme décrit dans la Section 9.8, « Chaque nœud d'un cluster à deux nœuds rapporte que le second nœud est en panne ». Vérifiez aussi si les communications entre nœuds sont toujours possibles et si le réseau fonctionne.
- Si des nœuds quittent le cluster, les nœuds restants peuvent ne pas atteindre le quorum. Le quorum doit être atteint pour que le cluster puisse fonctionner. Si des nœuds sont supprimés et que le cluster n'atteint pas le quorum, alors les services et le stockage seront suspendus. Dans ce cas, ajustez les votes attendus ou restituez la quantité requise de nœuds au cluster.

NOTE

Vous pouvez clôturer un nœud manuellement avec la commande **fence_node** ou avec **Conga**. Pour obtenir des informations, voir la page man **fence_node** et la Section 4.3.2, « Causer à un nœud de joindre ou quitter un cluster ».

9.6. LE SERVICE CLUSTER NE DÉMARRE PAS

Si un service contrôlé par un cluster ne démarre pas, vérifiez les conditions suivantes.

- Il peut y avoir une erreur de syntaxe dans la configuration du service dans le fichier **cluster.conf**. Vous pouvez utiliser la commande **rg_test** pour valider la syntaxe de la configuration. S'il y a un faute dans la configuration ou la syntaxe, **rg_test** vous informera sur le problème.

  ```
  $ rg_test test /etc/cluster/cluster.conf start service servicename
  ```

  Pour obtenir plus d'informations sur la commande **rg_test**, voir la Section C.5, « Débogage et testage des services et de l'ordre des ressources ».

  Si la configuration est valide, augmentez alors la journalisation du gestionnaire du groupe de ressource puis lisez les journaux des messages pour déterminer ce qui cause l'échec du démarrage du service. Vous pouvez augmenter le niveau de journalisation en ajoutant le paramètre **loglevel="7"** au marqueur **rm** dans le fichier **cluster.conf**. Vous remarquerez alors une augmentation de la verbosité dans les journaux des messages concernant le démarrage, l'arrêt et la migration des services clusterisés.

9.7. ÉCHEC DE LA MIGRATION DES SERVICES CONTRÔLÉS PAR LE CLUSTER

Si un service contrôlé par un cluster ne parvient pas à migrer vers un autre nœud mais que le service démarre sur certains nœuds spécifiques, vérifiez les conditions suivantes :

- Assurez-vous que les ressources requises pour exécuter un service donné sont présentes sur tous les nœuds du cluster qui pourraient devoir exécuter ce service. Par exemple, si le service clusterisé suppose qu'un fichier script se trouve dans un emplacement spécifique ou dans un système de fichiers monté sur un point de montage spécifique, alors vous devez vous assurer que ces ressources sont disponibles aux emplacements prévus sur tous les nœuds du cluster.
- Assurez-vous que les domaines de basculement, les dépendances et l'exclusivité des services ne sont pas configurés d'une manière vous empêchant de migrer les services vers des nœuds.
- Si le service en question est une ressource de machine virtuelle, vérifiez la documentation pour vous assurer que tout le travail de configuration réalisé est correct.
- Augmentez la journalisation du gestionnaire des groupes de services, comme décrit dans la Section 9.6, « Le service cluster ne démarre pas », puis lisez les journaux de messages pour déterminer ce qui cause l'échec de la migration du démarrage du service.

9.8. CHAQUE NŒUD D'UN CLUSTER À DEUX NŒUDS RAPPORTE QUE LE SECOND NŒUD EST EN PANNE

Si votre cluster est un cluster à deux nœuds et que chaque nœud rapporte qu'il fonctionne et que l'autre nœud est en panne, ceci indique que les nœuds du cluster ne parviennent pas à communiquer via multidiffusion sur le réseau de pulsation du cluster, ce qui est aussi connu sous le nom de "split brain" ou de "partition du réseau". Pour répondre à ce problème, vérifiez les conditions décrites dans la Section 9.2, « Le cluster ne se forme pas ».

9.9. NŒUDS CLÔTURÉS SUR UN CHEMIN D'ACCÈS LUN EN ÉCHEC

Si un ou plusieurs nœuds du cluster sont clôturés lorsqu'il y a un échec du chemin d'accès LUN, cela peut résulter d'une utilisation d'un disque quorum sur un stockage à multiples chemins d'accès. Si vous utilisez un disque quorum et que celui-ci se trouve sur un stockage à multiple chemins d'accès, assurez-vous que les délais paramétrés sont corrects afin de tolérer les échecs de chemins d'accès.

9.10. LE DISQUE QUORUM N'APPARAÎT PAS EN TANT QUE MEMBRE DU CLUSTER

Si vous avez configuré le système pour qu'il utilise un disque quorum mais que le disque quorum n'apparaît pas en tant que membre du cluster, vérifiez les conditions suivantes.

- Assurez-vous de bien avoir ajusté **`chkconfig on`** pour le service **`qdisk`**.
- Assurez-vous de bien avoir démarré le service **`qdisk`**.
- Remarquez que l'enregistrement du disque quorum sur le cluster peut prendre quelques minutes. Ce comportement est normal et prévu.

9.11. COMPORTEMENT INHABITUEL DES BASCULEMENTS

L'un des problèmes communs des serveurs de clusters est le comportement inhabituel des basculements. Les services s'arrêteront lorsque d'autres services démarrent ou des services refuseront de démarrer lors d'un basculement. Ceci peut résulter de la nature de systèmes de basculement complexes consistant des domaines de basculement, des dépendances des services, et des exclusivités des services. Essayez d'échelonner vers un service ou une configuration de domaine de basculement plus simple pour voir si le problème persiste. Évitez les fonctionnalités comme l'exclusivité et la dépendance de service à moins de bien comprendre de quelle manière celles-ci peuvent affecter le basculement quelles que soient les conditions.

9.12. LE FENCING SE PRODUIT AU HASARD

Si vous remarquez qu'un nœud est clôturé au hasard, vérifiez les conditions suivantes.

- La cause profonde des fences est *toujours* un nœud perdant un jeton, cela signifie que celui-ci a perdu la faculté de communiquer avec le reste du cluster et arrêté de retourner la pulsation.
- Toute situation résultant en un système ne retournant pas la pulsation dans l'intervalle spécifiée du jeton peut mener à une opération de fencing. Par défaut, l'intervalle du jeton est de 10 secondes. Cet intervalle peut être spécifié en ajoutant la valeur souhaitée (en millisecondes) au paramètre du jeton de la balise totem dans le fichier **`cluster.conf`** (par exemple, en paramétrant **`totem token="30000"`** pour 30 secondes).
- Assurez-vous que le réseau est solide et fonctionne comme prévu.
- Assurez-vous que les interfaces utilisées par le cluster pour les communications inter-nœuds ne soient utilisées par aucun autre mode de liaison que les modes 0, 1, ou 2. (Les modes de liaison 0 et 2 sont pris en charge à partir de Red Hat Enterprise Linux 6.4.)
- Prenez des mesures pour déterminer si le système est gelé ou s'il y a une panique du noyau. Paramétrez l'utilitaire **`kdump`** et voyez si vous trouvez un cœur lors de l'une de ces clôtures.
- Assurez-vous qu'il ne s'agisse pas d'un problème attribué par erreur au fencing. Par exemple, lorsque le disque quorum éjecte un nœud dû à un échec du stockage ou à un produit de tierce partie comme Oracle RAC redémarrant un nœud à cause d'une condition externe quelconque.

Les journaux des messages sont souvent très utiles pour déterminer de tels problèmes. Lorsque des redémarrages de nœuds se produisent ou lorsque des fences se mettent en place, l'inspection des journaux des messages de tous les nœuds dans le cluster à partir du moment auquel le redémarrage ou le fencing s'est produit devrait être une pratique standard.

- Inspectez minutieusement le système pour trouver des défauts de matériel pouvant mener le système à ne plus répondre à la pulsation lorsqu'il le devrait.

9.13. LA JOURNALISATION DU DÉBOGAGE POUR LE DLM (« DISTRIBUTED LOCK MANAGER », OU GESTIONNAIRE DE VERROUS DISTRIBUÉS) DOIT ÊTRE ACTIVÉE

Il existe deux options de débogage pour le DLM (gestionnaire de verrous distribués) pouvant être activées si nécessaire : Le débogage du noyau DLM et le débogage de verrous POSIX.

Pour activer le débogage DLM, modifiez le fichier **/etc/cluster/cluster.conf** pour ajouter des options de configuration à la balise **dlm**. L'option **log_debug** active les messages de débogage du noyau DLM et l'option **plock_debug** active les messages de débogage de verrous POSIX.

L'exemple de section du fichier **/etc/cluster/cluster.conf** qui suit affiche la balise **dlm** qui active les options de débogage DLM :

```
<cluster config_version="42" name="cluster1">
  ...
  <dlm log_debug="1" plock_debug="1"/>
  ...
</cluster>
```

Après avoir modifié le fichier **/etc/cluster/cluster.conf**, veuillez exécutez la commande **cman_tool version -r** pour propager la configuration au reste des nœuds du cluster.

CHAPITRE 10. CONFIGURATION SNMP AVEC LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

À partir de Red Hat Enterprise Linux 6.1 et de ses versions plus récentes, le module complémentaire Red Hat High Availability fournit la prise en charge des interruptions SNMP. Ce chapitre décrit comment configurer votre système pour SNMP et est suivi d'un résumé des interruptions que le module complémentaire Red Hat High Availability émet pour des événements spécifiques de clusters.

10.1. SNMP ET LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

foghorn, le sous-agent SNMP du module complémentaire Red Hat High Availability, émet les interruptions SNMP. Le sous-agent **foghorn**, communique avec le démon **snmpd** au moyen du protocole AgentX. Le sous-agent **foghorn** peut uniquement créer les interruptions SNMP, il ne prend pas en charge d'autres opérations SNMP comme **get** ou **set**.

Il n'y a pas d'options **config** actuellement pour le sous-agent **foghorn**. Il ne peut pas être configuré pour utiliser un socket spécifique. Seul le socket AgentX est actuellement pris en charge.

10.2. CONFIGURER SNMP AVEC LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

Pour configurer SNMP avec le module complémentaire Red Hat High Availability, effectuez les étapes suivantes sur chaque nœud dans le cluster afin de vous assurer que les services nécessaires sont bien activés et en cours d'exécution.

1. Pour utiliser des interruptions SNMP avec le module complémentaire Red Hat High Availability, le service **snmpd** est requis et agit en tant qu'agent maître. Comme le service **foghorn** est le sous-agent et utilise le protocole AgentX, vous devez ajouter la ligne suivante au fichier **/etc/snmp/snmpd.conf** pour activer la prise en charge d'AgentX :

   ```
   master agentx
   ```

2. Pour spécifier l'hôte vers lequel les notifications des interruptions SNMP devraient être envoyées, ajoutez la ligne suivante au fichier **/etc/snmp/snmpd.conf** :

   ```
   trap2sink host
   ```

   Pour obtenir des informations sur la gestion des notifications, voir la page man **snmpd.conf**.

3. Assurez-vous que le démon **snmpd** est bien activé et en cours d'exécution en exécutant les commandes suivantes :

   ```
   # chkconfig snmpd on
   # service snmpd start
   ```

4. Si le démon **messagebus** n'est pas déjà activé et en cours d'exécution, exécutez les commandes suivantes :

   ```
   # chkconfig messagebus on
   # service messagebus start
   ```

5. Assurez-vous que le démon **foghorn** est bien activé et en cours d'exécution en exécutant les commandes suivantes :

```
# chkconfig foghorn on
# service foghorn start
```

6. Exécutez la commande suivante pour configurer votre système de manière à ce que **COROSYNC-MIB** génère des interruptions SNMP et pour vous assurer que le démon **corosync-notifyd** est bien activé et en cours d'exécution :

```
# echo "OPTIONS=\"-d\" " > /etc/sysconfig/corosync-notifyd
# chkconfig corosync-notifyd on
# service corosync-notifyd start
```

Après avoir configuré chaquen nœud dans le cluster pour SNMP et vous être assuré que les services nécessaires sont en cours d'exécution, des signaux D-bus seront reçus par le service **foghorn** et traduits en interruptions SNMPv2. Ces interruptions sont ensuite passées à l'hôte que vous avez défini avec l'entrée **trapsink** pour recevoir les interruptions SNMPv2.

10.3. TRANSFÉRER LES INTERRUPTIONS SNMP

Il est possible de transférer des interruptions SNMP sur une machine qui ne fait pas partie du cluster, et sur laquelle vous pouvez utiliser le démon **snmptrapd** et personnaliser la manière par laquelle répondre aux notifications.

Effectuez les étapes suivantes pour transférer des interruptions SNMP dans un cluster vers une machine qui n'est pas l'un des nœuds du cluster :

1. Pour chaque nœud dans le cluster, suivez la procédure décrite dans la Section 10.2, « Configurer SNMP avec le module complémentaire Red Hat High Availability », en paramétrant l'entrée **trap2sink *host*** dans le fichier **/etc/snmp/snmpd.conf** pour spécifier l'hôte externe qui exécutera le démon **snmptrapd**.

2. Sur l'hôte externe qui recevra les interruptions, modifiez le fichier de configuration **/etc/snmp/snmptrapd.conf** pour spécifier vos chaînes de communauté. Par exemple, vous pouvez utiliser l'entrée suivante pour permettre au démon **snmptrapd** de traiter les notifications à l'aide de la chaîne de communauté **public**.

```
authCommunity log,execute,net public
```

3. Sur l'hôte externe qui recevra les interruptions, assurez-vous que le démon **snmptrapd** est activé et en cours d'exécution en saisissant les commandes suivantes :

```
# chkconfig snmptrapd on
# service snmptrapd start
```

Pour obtenir plus d'informations sur le traitement des notifications SNMP, voir la page man **snmptrapd.conf**.

10.4. INTERRUPTIONS SNMP PRODUITES PAR LE MODULE COMPLÉMENTAIRE RED HAT HIGH AVAILABILITY

Le démon **foghorn** génère les interruptions suivantes :

- **fenceNotifyFenceNode**

  Cette interruption se produit lorsqu'un nœud clôturé (fenced) tente de clôturer un autre nœud. Remarquez que cette interruption est uniquement générée sur un nœud - le nœud qui a tenté d'effectuer l'opération de fencing. La notification inclut les champs suivants :

  - **fenceNodeName** - nom du nœud clôturé
  - **fenceNodeID** - id de nœud du nœud clôturé
  - **fenceResult** - résultat de l'opération fence (0 pour une réussite, -1 lorsqu'un problème s'est produit, -2 si aucune méthode fence n'est définie)

- **rgmanagerServiceStateChange**

  Cette interruption se produit lorsque l'état d'un service cluster change. La notification inclut les champs suivants :

  - **rgmanagerServiceName** - nom du service, qui inclut le type de service (par exemple, **service:foo** ou **vm:foo**).
  - **rgmanagerServiceState** - état du service. Ceci exclut les états transitionnels tels que **starting** (démarrage) et **stopping** (arrêt) pour réduire l'encombrement dans les interruptions.
  - **rgmanagerServiceFlags** - indicateurs de service. Actuellement, deux indicateurs sont pris en charge : **frozen**, indiquant un service qui a été gelé à l'aide de **clusvcadm -Z** et **partial**, indiquant un service dans lequel une ressource en échec a été marquée comme **non-critique** pour que celle-ci puisse échouer et que ses composants puissent être redémarrés sans que le service entier ne soit affecté.
  - **rgmanagerServiceCurrentOwner** - propriétaire du service. Si le service n'est pas en cours d'exécution, celui-ci affichera **(none)** (aucun).
  - **rgmanagerServicePreviousOwner** - dernier propriétaire du service, s'il est connu. S'il n'est pas connu, celui-ci peut afficher **(none)** (aucun).

Le démon **corosync-nodifyd** génère les interruptions suivantes :

- **corosyncNoticesNodeStatus**

  Cette interruption se produit lorsqu'un nœud rejoint ou quitte le cluster. La notification inclut les champs suivants :

  - **corosyncObjectsNodeName** - nom du nœud
  - **corosyncObjectsNodeID** - id du nœud
  - **corosyncObjectsNodeAddress** - adresse IP du nœud
  - **corosyncObjectsNodeStatus** - état du nœud (**joined** ou **left**)

- **corosyncNoticesQuorumStatus**

Cette interruption se produit lorsque l'état du quorum change. La notification inclut les champs suivants :

- **corosyncObjectsNodeName** - nom du nœud
- **corosyncObjectsNodeID** - id du nœud
- **corosyncObjectsQuorumStatus** - nouvel état du quorum (**quorate** ou **NOT quorate**)

- **corosyncNoticesAppStatus**

  Cette interruption se produit lorsqu'une application client se connecte ou se déconnecte de Corosync.

  - **corosyncObjectsNodeName** - nom du nœud
  - **corosyncObjectsNodeID** - id du nœud
  - **corosyncObjectsAppName** - nom de l'application
  - **corosyncObjectsAppStatus** - nouvel état de l'application (**connected** ou **disconnected**)

CHAPITRE 11. CONFIGURATION DE SAMBA EN CLUSTER

À partir de Red Hat Enterprise Linux 6.2, le module complémentaire Red Hat High Availability a fourni la prise en charge de l'exécution de Samba en cluster sous une configuration active/active. Ceci requiert que vous installiez et configuriez CTDB sur tous les nœuds dans un cluster, ce que vous utilisez en conjonction avec les systèmes de fichiers en cluster GFS2.

NOTE

Red Hat Enterprise Linux 6 prend en charge un maximum de quatre nœuds exécutant Samba clusterisé.

Ce chapitre décrit la procédure pour configurer CTDB en configurant un exemple de système. Pour obtenir des informations sur la configuration des systèmes de fichier GFS2, reportez-vous à *Global File System 2*. Pour obtenir des informations sur la configuration de volumes logiques, reportez-vous à *Administration LVM*.

11.1. VUE D'ENSEMBLE DE CTDB

CTDB est une implémentation cluster de la base de données TDB utilisée par Samba. Pour utiliser CTDB, un système de fichiers clusterisé doit être disponible et partagé sur tous les nœuds dans le cluster. CTDB fournit des fonctionnalités pour cluster sur le haut de ce système de fichiers clusterisé. À partir de Red Hat Enterprise Linux 6.2, CTDB exécute aussi une pile de cluster en parallèle à celle fournie par le clustering Red Hat Enterprise Linux. CTDB gère les appartenances des nœuds, les récupérations/basculements, les relocations IP et les services Samba.

11.2. PAQUETAGES REQUIS

En plus des paquetages standards requis pour exécuter les modules complémentaires Red Hat High Availability et Red Hat Resilient Storage, l'exécution de Samba avec le clustering Red Hat Enterprise Linux requiert les paquetages suivants :

- **`ctdb`**
- **`samba`**
- **`samba-common`**
- **`samba-winbind-clients`**

11.3. CONFIGURATION GFS2

Configurer Samba avec le clustering Red Hat Enterprise Linux requiert deux systèmes de fichiers GFS2 : un petit système de fichiers pour CTDB et un second pour le partage Samba. Cet exemple montre comment créer les deux systèmes de fichiers GFS2.

Avant de créer les systèmes de fichiers GFS2, créez un volume logique LVM pour chacun des systèmes de fichiers. Pour obtenir des informations sur la création de volumes logiques, reportez-vous à l'ouvrage *Administration LVM*. Cet exemple utiliser les volumes logiques suivants :

- **`/dev/csmb_vg/csmb_lv`**, qui contient les données utilisateur qui seront exportées via le partage Samba et doit donc être dimensionné en conséquence. Cet exemple crée un volume logique d'une taille de 100 Go.

- **/dev/csmb_vg/ctdb_lv**, qui contiendra les informations sur l'état du partage CTDB et doit faire une taille de 1 Go.

Créez des groupes de volumes et volumes logiques clusterisés sur un nœud du cluster uniquement.

Pour créer un système de fichiers GFS2 sur un volume logique, exécutez la commande **mkfs.gfs2**. Exécutez cette commande sur un nœud du cluster uniquement.

Pour créer le système de fichiers devant héberger le partage Samba sur le volume logique **/dev/csmb_vg/csmb_lv**, veuillez exécuter la commande suivante :

```
[root@clusmb-01 ~]# mkfs.gfs2 -j3 -p lock_dlm -t csmb:gfs2
/dev/csmb_vg/csmb_lv
```

La signification des paramètres est comme suit :

-j

Spécifie le nombre de journaux à créer dans le système de fichiers. Cet exemple utilise un cluster à trois nœuds, nous créons donc un journal par nœud.

-p

Spécifie le protocole de verrouillage. **lock_dlm** est le protocole de verrouillage utilisé par GFS2 pour les communications entre les nœuds.

-t

Spécifie le nom du tableau de verrouillage et se trouve sous le format *cluster_name:fs_name*. Dans cet exemple, le nom du cluster spécifié dans le fichier **cluster.conf** est **csmb** et nous utilisons **gfs2** en tant que nom du système de fichiers.

La sortie de cette commande apparaît comme suit :

```
This will destroy any data on /dev/csmb_vg/csmb_lv.
  It appears to contain a gfs2 filesystem.

Are you sure you want to proceed? [y/n] y

Device:
/dev/csmb_vg/csmb_lv
Blocksize:  4096
Device Size  100.00 GB (26214400 blocks)
Filesystem Size: 100.00 GB (26214398 blocks)
Journals:  3
Resource Groups:  400
Locking Protocol:   "lock_dlm"
Lock Table:   "csmb:gfs2"
UUID:
  94297529-ABG3-7285-4B19-182F4F2DF2D7
```

Dans cet exemple, le système de fichiers **/dev/csmb_vg/csmb_lv** sera monté sur l'emplacement **/mnt/gfs2** sur tous les nœuds. Ce point de montage doit correspondre à la valeur que vous spécifiez comme étant l'emplacement du répertoire **share** (répertoire de partage) avec l'option **path =** dans le fichier **/etc/samba/smb.conf**, comme le décrit la Section 11.5, « Configuration de Samba ».

Pour créer le système de fichiers devant héberger les informations d'état de CTDB sur le volume logique **/dev/csmb_vg/ctdb_lv**, veuillez exécuter la commande suivante :

```
[root@clusmb-01 ~]# mkfs.gfs2 -j3 -p lock_dlm -t csmb:ctdb_state
/dev/csmb_vg/ctdb_lv
```

Remarquez que cette commande spécifie un nom de tableau de verrouillage différent de celui dans l'exemple qui crée le système de fichier sur **/dev/csmb_vg/csmb_lv**. Ceci permet de distinguer les noms des tableaux de verrouillage des différents périphériques utilisés pour les systèmes de fichiers.

La sortie de **mkfs.gfs2** apparaît comme suit :

```
This will destroy any data on /dev/csmb_vg/ctdb_lv.
  It appears to contain a gfs2 filesystem.

Are you sure you want to proceed? [y/n] y

Device:
/dev/csmb_vg/ctdb_lv
Blocksize:    4096
Device Size   1.00 GB (262144 blocks)
Filesystem Size:  1.00 GB (262142 blocks)
Journals:  3
Resource Groups:  4
Locking Protocol:  "lock_dlm"
Lock Table:   "csmb:ctdb_state"
UUID:
  BCDA8025-CAF3-85BB-B062-CC0AB8849A03
```

Dans cet exemple, le système de fichiers **/dev/csmb_vg/ctdb_lv** sera monté sur l'emplacement **/mnt/ctdb** sur tous les nœuds. Ce point de montage doit correspondre à la valeur que vous spécifiez comme étant l'emplacement du fichier **.ctdb.lock** avec l'option **CTDB_RECOVERY_LOCK** dans le fichier **/etc/sysconfig/ctdb**, comme le décrit la Section 11.4, « Configuration de CTDB ».

11.4. CONFIGURATION DE CTDB

Le fichier de configuration de CTDB se trouve dans **/etc/sysconfig/ctdb**. Les champs devant être obligatoirement configurés pour opérer CTDB sont les suivants :

- **CTDB_NODES**
- **CTDB_PUBLIC_ADDRESSES**
- **CTDB_RECOVERY_LOCK**
- **CTDB_MANAGES_SAMBA** (doit être activé)
- **CTDB_MANAGES_WINBIND** (doit être activé si exécuté sur un serveur membre)

L'exemple suivant montre un fichier de configuration avec les champs obligatoires pour opérer CTDB définis avec des exemples de paramètres :

```
CTDB_NODES=/etc/ctdb/nodes
CTDB_PUBLIC_ADDRESSES=/etc/ctdb/public_addresses
```

```
CTDB_RECOVERY_LOCK="/mnt/ctdb/.ctdb.lock"
CTDB_MANAGES_SAMBA=yes
CTDB_MANAGES_WINBIND=yes
```

La signification de ces paramètres est comme suit.

CTDB_NODES

Spécifie l'emplacement du fichier contenant la liste des nœuds du cluster.

Le fichier **/etc/ctdb/nodes** que **CTDB_NODES** référence répertorie simplement les adresses IP des nœuds du cluster, comme dans l'exemple suivant :

```
192.168.1.151
192.168.1.152
192.168.1.153
```

Dans cet exemple, il n'y a qu'une seule interface/adresse IP sur chaque nœud utilisé pour les communications cluster/CTDB et pour servir les clients. Cependant, il est fortement recommandé que chaque nœud de cluster possède deux interfaces réseau, ainsi un ensemble d'interfaces pourra être dédié aux communications cluster/CTDB et un autre ensemble pourra être dédié à l'accès public du client. Veuillez utiliser les adresses IP correctes du réseau du cluster et vous assurer que les nom d'hôtes/adresses IP utilisés dans le fichier **cluster.conf** sont bien les mêmes. De la même manière, veuillez utiliser les interfaces correctes du réseau public pour l'accès client dans le fichier **public_addresses**.

Il est critique que le fichier **/etc/ctdb/nodes** soit identique sur tous les nœuds car l'ordre est important et CTDB échouera si différentes informations se trouvent sur différents nœuds.

CTDB_PUBLIC_ADDRESSES

Spécifie l'emplacement du fichier qui répertorie les adresses IP pouvant être utilisées pour accéder aux partages Samba exportés par ce cluster. Ce sont les adresses IP que vous devriez configurer dans DNS pour le nom du serveur Samba clusterisé et les adresses auxquelles les clients CIFS se connecteront. Configurez le nom du serveur Samba clusterisé comme étant un enregistrement DNS de type A avec de multiples adresses IP et laissez le DNS Round-Robin distribuer les clients à travers les nœuds du cluster.

Pour cet exemple, nous avons configuré une entrée DNS Round-Robin **csmb-server** avec toutes les adresses répertoriées dans le fichier **/etc/ctdb/public_addresses**. Le DNS distribuera les clients utilisant cette entrée sur le cluster à l'aide de la technique du DNS Round-Robin.

Le contenu du fichier **/etc/ctdb/public_addresses** sur chaque nœud est comme suit :

```
192.168.1.201/0 eth0
192.168.1.202/0 eth0
192.168.1.203/0 eth0
```

Cet exemple utilise trois adresses qui sont actuellement inutilisées sur le réseau. Dans votre propre configuration, choisissez les adresses pouvant être accédées par les clients voulus.

Alternativement, cet exemple affiche le contenu des fichiers **/etc/ctdb/public_addresses** dans un cluster dans lequel se trouvent trois nœuds, mais un total de quatre adresses publiques. Dans cet exemple, l'adresse IP 198.162.2.1 peut être hébergée par le nœud 0 ou le nœud 1 et sera disponible aux clients aussi longtemps que l'un de ces nœuds sera disponible. Cette adresse publique sera

indisponible aux clients uniquement si les nœuds 0 et 1 échouent. Toutes les autres adresses publiques peuvent uniquement être servies par un seul nœud respectivement et seront donc seulement disponibles si le nœud respectif est aussi disponible.

Le fichier **/etc/ctdb/public_addresses** sur le nœud 0 inclut le contenu suivant :

```
198.162.1.1/24 eth0
198.162.2.1/24 eth1
```

Le fichier **/etc/ctdb/public_addresses** sur le nœud 1 inclut le contenu suivant :

```
198.162.2.1/24 eth1
198.162.3.1/24 eth2
```

Le fichier **/etc/ctdb/public_addresses** sur le nœud 2 inclut le contenu suivant :

```
198.162.3.2/24 eth2
```

CTDB_RECOVERY_LOCK

Spécifie un fichier verrou que CTDB utilise de manière interne pour la récupération. Ce fichier doit être sur un stockage partagé afin que tous les nœuds du cluster puissent y accéder. L'exemple de cette section utilise le système de fichiers GFS2 qui sera monté sur **/mnt/ctdb** sur tous les nœuds. Ceci est différent du système de fichiers GFS2 qui hébergera le partage Samba devant être exporté. Ce fichier verrou de récupération est utilisé afin de prévenir les scénarios de type « split-brain ». Dans les versions plus récentes de CTDB (à partir de la version 1.0.112), la spécification de ce fichier est optionnelle à partir du moment où celle-ci est remplacée par un autre mécanisme de prévention de « split-brain ».

CTDB_MANAGES_SAMBA

Lorsqu'activé, en paramétrant sur **yes**, cette valeur spécifie que CTDB est autorisé à démarrer et arrêter le service Samba comme nécessaire, afin de fournir un basculement ou une migration du service.

Lorsque **CTDB_MANAGES_SAMBA** est activé, vous devriez désactiver le démarrage automatique **init** des démons **smb** et **nmb** en exécutant les commandes suivantes :

```
[root@clusmb-01 ~]# chkconfig snb off
[root@clusmb-01 ~]# chkconfig nmb off
```

CTDB_MANAGES_WINBIND

Lorsqu'activé, en paramétrant sur **yes**, cette valeur spécifie que CTDB est autorisé à démarrer et arrêter le démon **winbind** comme requis. Celui-ci devrait être activé lorsque CTDB est utilisé dans un domaine Windows ou en mode de sécurité de répertoire actif.

Lorsque **CTDB_MANAGES_WINBIND** est activé, vous devriez désactiver le démarrage automatique **init** du démon **winbind** en exécutant la commande suivante :

```
[root@clusmb-01 ~]# chkconfig windinbd off
```

11.5. CONFIGURATION DE SAMBA

Dans cet exemple, le fichier de configuration de Samba **smb.conf** est placé sur **/etc/samba/smb.conf**. Il contient les paramètres suivants :

```
[global]
 guest ok = yes
 clustering = yes
 netbios name = csmb-server
[csmb]
 comment = Clustered Samba
  public = yes
 path = /mnt/gfs2/share
 writeable = yes
 ea support = yes
```

Cet exemple exporte un partage avec le nom **csmb**, se trouvant sur **/mnt/gfs2/share**. Ceci est différent du système de fichiers partagé GFS2 sur **/mnt/ctdb/.ctdb.lock** que nous avons spécifié comme étant le paramètre **CTDB_RECOVERY_LOCK** dans le fichier de configuration CTDB sur **/etc/sysconfig/ctdb**.

Dans cet exemple, nous créerons le répertoire **share** sur **/mnt/gfs2** lorsque nous le monterons pour la première fois. L'entrée **clustering = yes** ordonne à Samba d'utiliser CTDB. L'entrée **netbios name = csmb-server** paramètre explicitement tous les nœuds de manière à ce qu'ils aient un nom NetBIOS commun. Le paramètre **ea support** est requis si vous planifiez d'utiliser des attributs étendus.

Le fichier de configuration **smb.conf** doit être identique sur tous les nœuds du cluster.

Samba propose aussi une configuration basée sur le registre avec la commande **net conf** pour que la configuration reste automatiquement synchronisée entre les différents membres du cluster sans avoir à copier manuellement les fichiers de configuration des nœuds du cluster. Pour obtenir des informations sur la commande **net conf**, veuillez vous reporter à la page man **net**(8).

11.6. LANCER CTDB ET LES SERVICES SAMBA

Après avoir démarré le cluster, vous devez monter les systèmes de fichiers GFS2 créés, comme le décrit la Section 11.3, « Configuration GFS2 ». Les permissions sur le répertoire Samba **share** et les comptes d'utilisateurs sur les nœuds du cluster doivent être paramétrés pour l'accès client.

Veuillez exécuter la commande suivante sur tous les nœuds pour lancer le démon **ctdbd**. Comme cet exemple configure CTDB avec **CTDB_MANAGES_SAMBA=yes**, CTDB lancera aussi le service Samba sur tous les nœuds et exportera tous les partages Samba configurés.

```
[root@clusmb-01 ~]# service ctdb start
```

CTDB peut prendre quelques minutes pour lancer Samba, exporter les partages et se stabiliser. Exécuter **ctdb status** affiche le statut de CTDB, comme l'exemple suivant le montre :

```
[root@clusmb-01 ~]# ctdb status
Number of nodes:3
pnn:0 192.168.1.151     OK (THIS NODE)
pnn:1 192.168.1.152     OK
pnn:2 192.168.1.153     OK
```

```
Generation:1410259202
Size:3
hash:0 lmaster:0
hash:1 lmaster:1
hash:2 lmaster:2
Recovery mode:NORMAL (0)
Recovery master:0
```

Lorsque tous les nœuds sont « Ok », vous pourrez utiliser le serveur Samba clusterisé en toute sécurité, comme le décrit la Section 11.7, « Utiliser le serveur Samba clusterisé » .

11.7. UTILISER LE SERVEUR SAMBA CLUSTERISÉ

Les clients peuvent se connecter au partage Samba qui a été exporté en se connectant à l'une des adresses IP spécifiées dans le fichier **/etc/ctdb/public_addresses**, ou en utilisant l'entrée DNS **csmb-server** configurée au préalable, comme suit :

```
[root@clusmb-01 ~]# mount -t cifs //csmb-server/csmb /mnt/sambashare -o
user=testmonkey
```

ou

```
[user@clusmb-01 ~]$ smbclient //csmb-server/csmb
```

ANNEXE A. PARAMÈTRES DES PÉRIPHÉRIQUES FENCE

Cet annexe fournit des tableaux avec les descriptions des paramètres des périphériques fence. Vous pouvez configurer les paramètres avec **luci**, en utilisant la commande **ccs**, ou en modifiant le fichier **etc/cluster/cluster.conf**. Pour obtenir une liste et description complète des paramètres du périphérique fence pour chaque agent fence, veuillez vous reporter à la page man de cet agent.

NOTE

Le paramètre **Name** pour un périphérique fence spécifie un nom arbitraire pour le périphérique qui sera utilisé par le module complémentaire Red Hat High Availability. Il ne s'agit pas de la même chose que le nom DNS du périphérique.

NOTE

Certains périphériques fence possèdent un paramètre optionnel **Password Script**. Le paramètre **Password Script** vous permet de spécifier qu'un mot de passe de périphérique fence soit fournit par un script plutôt que par le paramètre **Password**. L'utilisation du paramètre **Password Script** supplante le paramètre **Password**, permettant aux mots de passe de ne pas être visibles dans le fichier de configuration du cluster (**/etc/cluster/cluster.conf**).

Tableau A.1, « Résumé des périphériques fence » répertorie les périphériques fence, les agents des périphériques fence associés aux périphériques fence, et fournit une référence au tableau documentant les paramètres des périphériques fence.

Tableau A.1. Résumé des périphériques fence

Périphérique fence	Agent fence	Références aux descriptions des paramètres
Interrupteur d'alimentation APC (telnet/SSH)	fence_apc	Tableau A.2, « Interrupteur d'alimentation APC (telnet/SSH) »
Interrupteur Brocade Fabric	fence_brocade	Tableau A.4, « Interrupteur Brocade Fabric »
MDS Cisco	fence_cisco_mds	Tableau A.5, « MDS Cisco »
UCS Cisco	fence_cisco_ucs	Tableau A.6, « UCS Cisco »
DRAC 5 de Dell	fence_drac5	Tableau A.7, « DRAC 5 de Dell »

Périphérique fence	Agent fence	Références aux descriptions des paramètres
Commutateur d'alimentation réseau Eaton « Eaton Network Power Switch » (Interface SNMP)	fence_eaton_snmp	Tableau A.8, « Contrôleur d'alimentation réseau Eaton (Interface SNMP) (Red Hat Enterprise Linux 6.4 et versions supérieures) »
Contrôleur SAN Egenera	fence_egenera	Tableau A.9, « Contrôleur SAN Egenera »
ePowerSwitch	fence_eps	Tableau A.10, « ePowerSwitch »
Fence virt	fence_virt	Tableau A.11, « Fence virt »
RSB Fujitsu Siemens (Remoteview Service Board)	fence_rsb	Tableau A.12, « RSB Fujitsu Siemens (Remoteview Service Board) »
HP BladeSystem	fence_hpblade	Tableau A.13, « HP BladeSystem (Red Hat Enterprise Linux 6.4 et versions supérieures) »
HP iLO/iLO2 (Integrated Lights Out)	fence_ilo	Tableau A.14, « HP iLO/iLO2 (Integrated Lights Out) »
HP iLO (Integrated Lights Out) MP	fence_ilo_mp	Tableau A.15, « HP iLO (Integrated Lights Out) MP »
IBM BladeCenter	fence_bladecenter	Tableau A.16, « IBM BladeCenter »
IBM BladeCenter SNMP	fence_ibmblade	Tableau A.17, « IBM BladeCenter SNMP »
IBM iPDU	fence_ipdu	Tableau A.18, « IBM iPDU (Red Hat Enterprise Linux 6.4 et versions supérieures) »
IF MIB	fence_ifmib	Tableau A.19, « IF MIB »
Modular Intel	fence_intelmodular	Tableau A.20, « Modular Intel »

Périphérique fence	Agent fence	Références aux descriptions des paramètres
IPMI (Interface de gestion de plateforme intelligente, en anglais « Intelligent Platform Management Interface ») LAN	fence_ipmilan	Tableau A.21, « IPMI (Interface de gestion de plateforme intelligente, en anglais « Intelligent Platform Management Interface ») LAN »
API REST RHEV-M	fence_rhevm	Tableau A.22, « RHEV-M REST API (RHEL 6.2 et versions plus récentes avec RHEV 3.0 et versions plus récentes) »
Fencing SCSI	fence_scsi	Tableau B.19, « SAP Instance »
Fencing VMware (Interface SOAP)	fence_vmware_soap	Tableau A.24, « Fencing VMware (interface SOAP) (Red Hat Enterprise Linux 6.2 et versions plus récentes) »
Interrupteur d'alimentation WTI	fence_wti	Tableau A.25, « Interrupteur d'alimentation WTI »

Le Tableau A.2, « Interrupteur d'alimentation APC (telnet/SSH) » répertorie les paramètres de périphériques fence utilisés par **`fence_apc`**, l'agent fence pour APC sur telnet/SSH.

Tableau A.2. Interrupteur d'alimentation APC (telnet/SSH)

Champ luci	Attribut `cluster.conf`	Description
Nom	**`name`**	Nom du périphérique APC connecté au cluster auquel le démon fence se connecte via telnet/ssh.
Adresse IP ou nom d'hôte	**`ipaddr`**	Adresse IP ou nom d'hôte assigné au périphérique.
Port IP (optionnel)	**`ipport`**	Port TCP à utiliser pour se connecter au périphérique.
Connexion	**`login`**	Nom de connexion utilisé pour accéder au périphérique.

Champ luci	Attribut cluster.conf	Description
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Port	**port**	Le port.
Interrupteur (optionnel)	**switch**	Numéro d'interrupteur de l'interrupteur APC qui se connecte au nœud lorsque vous avez de multiples interrupteurs connectés en chaîne.
Utiliser SSH	**secure**	Indique que le système utilisera SSH pour accéder au périphérique.
Chemin vers le fichier d'identité SSH	**identity_file**	Fichier d'identité de SSH.

Le Tableau A.3, « Interrupteur d'alimentation sur SNMP » répertorie les paramètres de périphérique fence utilisés par **fence_apc_snmp**, qui est l'agent fence pour APC qui se connecte au périphérique SNP via le protocole SNMP.

Tableau A.3. Interrupteur d'alimentation sur SNMP

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique APC connecté au cluster auquel le démon fence se connecte via le protocole SNMP.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
UDP/TCP port	**udpport**	Port UDP/TCP à utiliser pour la connexion avec le périphérique, la valeur par défaut est 161.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.

Champ luci	Attribut cluster.conf	Description
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Version de SNMP	**snmp_version**	Version SNMP à utiliser (1, 2c, 3), la valeur par défaut est 1.
Communauté SNMP	**community**	Chaîne « SNMP community », la valeur par défaut est **private**.
Niveau de sécurité SNMP	**snmp_sec_level**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**snmp_auth_prot**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**snmp_priv_prot**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**snmp_priv_passwd**	Mot de passe du protocole de confidentialité SNMP.
Script du protocole de confidentialité SNMP	**snmp_priv_passwd_script**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Numéro (de la prise) du port	**port**	Le port.

Le Tableau A.4, « Interrupteur Brocade Fabric » répertorie les paramètres de périphérique fence utilisés par **fence_brocade**, l'agent fence des interrupteurs Brocade FC.

Tableau A.4. Interrupteur Brocade Fabric

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique Brocade connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP assignée au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Port	**port**	Numéro de la prise de l'interrupteur.

Le Tableau A.5, « MDS Cisco » répertorie les paramètres du périphérique fence utilisés par **fence_cisco_mds**, l'agent fence pour Cisco MDS.

Tableau A.5. MDS Cisco

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique MDS 9000 series de Cisco avec SNMP activé.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
UDP/TCP port	**udpport**	Port UDP/TCP à utiliser pour la connexion avec le périphérique, la valeur par défaut est 161.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Numéro (de la prise) du port	**port**	Le port.
Version de SNMP	**snmp_version**	Version SNMP à utiliser (1, 2c, 3).

Champ luci	Attribut cluster.conf	Description
Communauté SNMP	**community**	Chaîne SNMP Community.
Niveau de sécurité SNMP	**snmp_sec_level**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**snmp_auth_prot**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**snmp_priv_prot**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**snmp_priv_passwd**	Mot de passe du protocole de confidentialité SNMP.
Script du protocole de confidentialité SNMP	**snmp_priv_passwd_script**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.

Le Tableau A.6, « UCS Cisco » répertorie les paramètres du périphérique fence utilisés par **fence_cisco_ucs**, l'agent fence pour Cisco UCS.

Tableau A.6. UCS Cisco

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique UCS Cisco.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
IP port (optional)	**ipport**	Port TCP à utiliser pour se connecter au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.

Champ luci	Attribut cluster.conf	Description
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Utiliser SSH	**ssl**	Utilisez les connexions SSL pour communiquer avec le périphérique.
Sous-organisation	**suborg**	Chemin supplémentaire nécessaire pour accéder à la sous-organisation.
Numéro (de la prise) du port	**port**	Nom de la machine virtuelle.
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.

Le Tableau A.7, « DRAC 5 de Dell » répertorie les paramètres du périphérique fence utilisés par **fence_drac5**, **fence_cisco_ucs**, l'agent fence pour Dell DRAC 5.

Tableau A.7. DRAC 5 de Dell

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom assigné au DRAC.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au DRAC.
Port IP (optionnel)	**ipport**	Port TCP à utiliser pour se connecter au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au DRAC.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au DRAC.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Utiliser SSH	**secure**	Indique que le système utilisera SSH pour accéder au périphérique.

Champ luci	Attribut cluster.conf	Description
Chemin vers le fichier d'identité SSH	identity_file	Fichier d'identité de SSH.
Nom du module	module_name	(Optionnel) Nom du module pour le DRAC lorsque vous possédez de multiples modules DRAC.
Forcer l'invite de commande	cmd_prompt	Invite de commande à utiliser. La valeur par défaut est '\$'.
Délai de l'alimentation	power_wait	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.

Tableau A.8, « Contrôleur d'alimentation réseau Eaton (Interface SNMP) (Red Hat Enterprise Linux 6.4 et versions supérieures) » répertorie les paramètres de périphérique réseau utilisés par **fence_eaton_snmp**, l'agent fence du commutateur d'alimentation réseau Eaton sur SNMP.

Tableau A.8. Contrôleur d'alimentation réseau Eaton (Interface SNMP) (Red Hat Enterprise Linux 6.4 et versions supérieures)

Champ luci	Attribut cluster.conf	Description
Nom	name	Nom du commutateur d'alimentation réseau Eaton connecté au cluster.
Adresse IP ou nom d'hôte	ipaddr	Adresse IP ou nom d'hôte assigné au périphérique.
Port UDP/TCP (optionnel)	udpport	Port UDP/TCP à utiliser pour la connexion avec le périphérique, la valeur par défaut est 161.
Connexion	login	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	passwd	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	passwd_script	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Version de SNMP	snmp_version	Version SNMP à utiliser (1, 2c, 3), la valeur par défaut est 1.
Communauté SNMP	community	Chaîne « SNMP community », la valeur par défaut est **private**.

Champ luci	Attribut cluster.conf	Description
Niveau de sécurité SNMP	**snmp_sec_level**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**snmp_auth_prot**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**snmp_priv_prot**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**snmp_priv_passwd**	Mot de passe du protocole de confidentialité SNMP.
Script du protocole de confidentialité SNMP	**snmp_priv_passwd_script**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Attente démarrage (secondes)	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Numéro (de la prise) du port	**port**	Numéro de la prise physique ou nom de la machine virtuelle. Ce paramètre est toujours requis.

Le Tableau A.9, « Contrôleur SAN Egenera » répertorie les paramètres du périphérique fence utilisés par **fence_egenera**, l'agent fence pour Egenera BladeFrame.

Tableau A.9. Contrôleur SAN Egenera

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique BladeFrame Egenera connecté au cluster.
CServer	**cserver**	Nom d'hôte (optionnellement nom d'utilisateur sous la forme **username@hostname**) assigné au périphérique. Reportez-vous à la page man fence_egenera(8) pour obtenir plus d'informations.
ESH Path (optional)	**esh**	Chemin d'accès de la commande esh sur le cserver (par défaut /opt/panmgr/bin/esh)

Champ luci	Attribut `cluster.conf`	Description
Nom d'utilisateur :	**`user`**	Nom de la connexion, la valeur par défaut est **`root`**.
lpan	**`lpan`**	LPAN (de l'anglais, « Logical Process Area Network », réseau de la zone du processus logique) du périphérique.
pserver	**`pserver`**	Nom du « processing blade » (pserver) du périphérique.

Le Tableau A.10, « ePowerSwitch » répertorie les paramètres du périphérique fence utilisés par **`fence_eps`**, l'agent fence pour ePowerSwitch.

Tableau A.10. ePowerSwitch

Champ luci	Attribut `cluster.conf`	Description
Nom	**`name`**	Nom du périphérique ePowerSwitch connecté au cluster.
Adresse IP ou nom d'hôte	**`ipaddr`**	Adresse IP ou nom d'hôte assigné au périphérique.
Connexion	**`login`**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**`passwd`**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**`passwd_script`**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Nom de la page cachée	**`hidden_page`**	Nom de la page cachée du périphérique.
Numéro (de la prise) du port	**`port`**	Numéro de la prise physique ou nom de machine virtuelle.

Le Tableau A.11, « Fence virt » répertorie les paramètres du périphérique fence utilisés par **`fence_virt`**, l'agent fence pour un périphérique fence « Fence virt ».

Tableau A.11. Fence virt

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique fence « Fence virt ».
Périphérique série	**serial_device**	Sur l'hôte, le périphérique série doit être mappé dans le fichier de configuration de chaque domaine. Pour obtenir plus d'informations, voir la page man **fence_virt.conf**. Si ce champ est spécifié, il cause à l'agent du fencing **fence_virt** d'opérer en mode série. Ne pas spécifier de valeur cause à l'agent du fencing **fence_virt** d'opérer en mode canal VM.
Paramètres de série	**serial_params**	Paramètres de série. Les valeurs par défaut sont 115200, 8N1.
Adresse IP du canal VM	**channel_address**	Adresse IP du canal. La valeur par défaut est 10.0.2.179.
Port ou domaine (déprécié)	**port**	Machine virtuelle (Nom ou UUID de domaine) à clôturer.
	ipport	Port du canal. La valeur par défaut est 1229, qui est la valeur utilisée lors de la configuration de ce périphérique fence avec **luci**.

Le Tableau A.12, « RSB Fujitsu Siemens (Remoteview Service Board) » répertorie les paramètres du périphérique fence utilisés par **fence_rsb**, l'agent fence pour le RSB Fujitsu-Siemens.

Tableau A.12. RSB Fujitsu Siemens (Remoteview Service Board)

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du RSB à utiliser en tant que périphérique fence.
Adresse IP ou nom d'hôte	**ipaddr**	Nom d'hôte assigné au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.

Champ luci	Attribut cluster.conf	Description
Port TCP	**ipport**	Numéro de port écouté par le service telnet. La valeur par défaut est 3172.

Tableau A.13, « HP BladeSystem (Red Hat Enterprise Linux 6.4 et versions supérieures) » répertorie les paramètres du périphérique fence utilisés par **fence_hpblade**, l'agent fence de HP BladeSystem.

Tableau A.13. HP BladeSystem (Red Hat Enterprise Linux 6.4 et versions supérieures)

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique HP BladeSystem connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique HP BladeSystem.
Port IP (optionnel)	**ipport**	Port TCP à utiliser pour se connecter au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique HP BladeSystem. Ce paramètre est requis.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion sur le périphérique fence.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Forcer l'invite de commande	**cmd_prompt**	Invite de commande à utiliser. La valeur par défaut est '\$'.
Le port manquant retourne OFF au lieu d'un échec	**missing_as_off**	Le port manquant retourne OFF au lieu d'un échec.
Attente démarrage (secondes)	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Utiliser SSH	**secure**	Indique que le système utilisera SSH pour accéder au périphérique.

Champ luci	Attribut cluster.conf	Description
Chemin vers le fichier d'identité SSH	identity_file	Fichier d'identité de SSH.

Le Tableau A.14, « HP iLO/iLO2 (Integrated Lights Out) » répertorie les paramètres du périphérique fence utilisés par **fence_ilo**, l'agent fence pour les périphériques HP iLO.

Tableau A.14. HP iLO/iLO2 (Integrated Lights Out)

Champ luci	Attribut cluster.conf	Description
Nom	name	Nom du serveur avec le support HP iLO.
Adresse IP ou nom d'hôte	ipaddr	Adresse IP ou nom d'hôte assigné au périphérique.
Port IP (optionnel)	ipport	Port TCP à utiliser pour une connexion avec le périphérique.
Connexion	login	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	passwd	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	passwd_script	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Délai de l'alimentation	power_wait	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.

Le Tableau A.15, « HP iLO (Integrated Lights Out) MP » répertorie les paramètres du périphérique fence utilisés par **fence_ilo_mp**, l'agent fence pour les périphériques HP iLO MP.

Tableau A.15. HP iLO (Integrated Lights Out) MP

Champ luci	Attribut cluster.conf	Description
Nom	name	Nom du serveur avec le support HP iLO.
Adresse IP ou nom d'hôte	ipaddr	Adresse IP ou nom d'hôte assigné au périphérique.

Champ luci	Attribut cluster.conf	Description
Port IP (optionnel)	**ipport**	Port TCP à utiliser pour une connexion avec le périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Utiliser SSH	**secure**	Indique que le système utilisera SSH pour accéder au périphérique.
Chemin vers le fichier d'identité SSH	**identity_file**	Fichier d'identité de SSH.
Forcer l'invite de commande	**cmd_prompt**	Invite de commande à utiliser. La valeur par défaut est 'MP>', 'hpiLO->'.
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.

Le Tableau A.16, « IBM BladeCenter » répertorie les paramètres du périphérique fence utilisés par **fence_bladecenter**, l'agent fence pour IBM BladeCenter.

Tableau A.16. IBM BladeCenter

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique IBM BladeCenter connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
IP port (optional)	**ipport**	Port TCP à utiliser pour une connexion avec le périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.

Champ luci	Attribut cluster.conf	Description
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Utiliser SSH	**secure**	Indique que le système utilisera SSH pour accéder au périphérique.
Chemin vers le fichier d'identité SSH	**identity_file**	Fichier d'identité de SSH.

Le Tableau A.17, « IBM BladeCenter SNMP » répertorie les paramètres du périphérique fence utilisés par **fence_ibmblade**, l'agent fence pour IBM BladeCenter sur SNMP.

Tableau A.17. IBM BladeCenter SNMP

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique IBM BladeCenter SNMP connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
Port UDP/TCP (optionnel)	**udpport**	Port UDP/TCP à utiliser pour les connexions avec le périphérique, la valeur par défaut est 161.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Version de SNMP	**snmp_version**	Version SNMP à utiliser (1, 2c, 3), la valeur par défaut est 1.
Communauté SNMP	**community**	Chaîne SNMP Community.

Champ luci	Attribut cluster.conf	Description
Niveau de sécurité SNMP	**snmp_sec_level**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**snmp_auth_prot**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**snmp_priv_prot**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**snmp_priv_passwd**	Mot de passe du protocole de confidentialité SNMP.
Script du protocole de confidentialité SNMP	**snmp_priv_passwd_script**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Port	**port**	Numéro de la prise physique ou nom de machine virtuelle.

Tableau A.18, « IBM iPDU (Red Hat Enterprise Linux 6.4 et versions supérieures) » répertorie les paramètres du périphérique fence utilisés par **fence_ipdu**, l'agent fence pour iPDU sur périphériques SNMP.

Tableau A.18. IBM iPDU (Red Hat Enterprise Linux 6.4 et versions supérieures)

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique IBM iPDU connecté au cluster auquel le démon fence se connecte via le protocole SNMP.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
Port UDP/TCP	**udpport**	Port UDP/TCP à utiliser pour la connexion avec le périphérique, la valeur par défaut est 161.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.

Champ luci	Attribut `cluster.conf`	Description
Mot de passe	**`passwd`**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**`passwd_script`**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Version de SNMP	**`snmp_version`**	Version SNMP à utiliser (1, 2c, 3), la valeur par défaut est 1.
Communauté SNMP	**`community`**	Chaîne « SNMP community », la valeur par défaut est **`private`**.
Niveau de sécurité SNMP	**`snmp_sec_level`**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**`snmp_auth_prot`**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**`snmp_priv_prot`**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**`snmp_priv_passwd`**	Mot de passe du protocole de confidentialité SNMP.
Script du protocole de confidentialité SNMP	**`snmp_priv_passwd_script`**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Délai de l'alimentation	**`power_wait`**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Port	**`port`**	Le port.

Le Tableau A.19, « IF MIB » répertorie les paramètres du périphérique fence utilisés par **`fence_ifmib`**, l'agent fence pour les périphériques IF-MIB.

Tableau A.19. IF MIB

Champ luci	Attribut `cluster.conf`	Description
Nom	**`name`**	Nom du périphérique IF MIB connecté au cluster.
Adresse IP ou nom d'hôte	**`ipaddr`**	Adresse IP ou nom d'hôte assigné au périphérique.
Port UDP/TCP (optionnel)	**`udpport`**	Port UDP/TCP à utiliser pour la connexion avec le périphérique, la valeur par défaut est 161.
Connexion	**`login`**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**`passwd`**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**`passwd_script`**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Version de SNMP	**`snmp_version`**	Version SNMP à utiliser (1, 2c, 3), la valeur par défaut est 1.
Communauté SNMP	**`community`**	Chaîne SNMP Community.
Niveau de sécurité SNMP	**`snmp_sec_level`**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**`snmp_auth_prot`**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**`snmp_priv_prot`**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**`snmp_priv_passwd`**	Mot de passe du protocole de confidentialité SNMP.
Script du protocole de confidentialité SNMP	**`snmp_priv_passwd_script`**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Délai de l'alimentation	**`power_wait`**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.

Champ luci	Attribut cluster.conf	Description
Port	**port**	Numéro de la prise physique ou nom de machine virtuelle.

Le Tableau A.20, « Modular Intel » répertorie les paramètres du périphérique fence utilisés par **fence_intelmodular**, l'agent fence pour Intel Modular.

Tableau A.20. Modular Intel

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique Intel Modular connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Version de SNMP	**snmp_version**	Version SNMP à utiliser (1, 2c, 3), la valeur par défaut est 1.
Communauté SNMP	**community**	Chaîne « SNMP community », la valeur par défaut est **private**.
Niveau de sécurité SNMP	**snmp_sec_level**	Niveau de sécurité SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocole d'authentification SNMP	**snmp_auth_prot**	Protocole d'authentification SNMP (MD5, SHA).
Protocole de confidentialité SNMP	**snmp_priv_prot**	Protocole de confidentialité SNMP (DES, AES).
Mot de passe du protocole de confidentialité SNMP	**snmp_priv_passwd**	Mot de passe du protocole de confidentialité SNMP.

Champ luci	Attribut cluster.conf	Description
Script du protocole de confidentialité SNMP	**snmp_priv_passwd_script**	Script fournissant un mot de passe pour le protocole de confidentialité SNMP. Son utilisation supplante le paramètre **SNMP privacy protocol password** (mot de passe du protocole de confidentialité SNMP).
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Port	**port**	Numéro de la prise physique ou nom de machine virtuelle.

Le Tableau A.21, « IPMI (Interface de gestion de plateforme intelligente, en anglais « Intelligent Platform Management Interface ») LAN » répertorie les paramètres de périphériques fence utilisés par **fence_ipmilan**, l'agent fence pour IPMI sur LAN.

Tableau A.21. IPMI (Interface de gestion de plateforme intelligente, en anglais « Intelligent Platform Management Interface ») LAN

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique IPMI LAN connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
Connexion	**login**	Nom de connexion d'un utilisateur en mesure d'effectuer des commandes de mise sous/hors tension sur un port IPMI donné.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion sur le port IPMI.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Type d'authentification	**auth**	Type d'authentification IPMI LAN : **none** (aucun), **password** (mot de passe), ou **md5**.
Utiliser Lanplus	**lanplus**	**True** ou **1**. Si vide, alors la valeur est **False**.
Ciphersuite to use	**cipher**	Serveur distant d'authentification, et algoritmes d'intégrité et de chiffrement à utiliser pour les connexions lanplus IPMIv2.

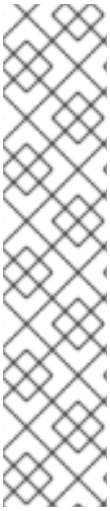
Champ luci	Attribut cluster.conf	Description
Niveau de privilèges	**privlvl**	Niveau de privilèges du périphérique IPMI.

Le Tableau A.22, « RHEV-M REST API (RHEL 6.2 et versions plus récentes avec RHEV 3.0 et versions plus récentes) » répertorie les paramètres du périphérique fence utilisés par **fence_rhevm**, l'agent fence pour RHEV-M REST API.

Tableau A.22. RHEV-M REST API (RHEL 6.2 et versions plus récentes avec RHEV 3.0 et versions plus récentes)

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom du périphérique de fencing RHEV-M REST API.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou nom d'hôte assigné au périphérique.
Port IP (optionnel)	**ipport**	Port TCP à utiliser pour une connexion avec le périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Utiliser SSH	**ssl**	Utilisez les connexions SSL pour communiquer avec le périphérique.
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Port	**port**	Numéro de la prise physique ou nom de machine virtuelle.

Le Tableau A.23, « Fencing SCSI » répertorie les paramètres du périphérique fence utilisés par **fence_scsi**, l'agent fence pour les réservations persistantes SCSI.



NOTE

L'utilisation des réservations persistantes SCSI en tant que méthode fence est prise en charge avec les limitations suivantes :

- Lors de l'utilisation du fencing SCSI, tous les nœuds dans le cluster doivent s'enregistrer avec les mêmes périphériques afin que chaque nœud puisse supprimer la clé d'enregistrement d'un autre nœud de tous les périphériques auprès desquels elle est enregistrée.
- Les périphériques utilisés pour les volumes de clusters devraient être un LUN complet et non des partitions. Les réservations persistantes SCSI fonctionnent sur un LUN entier, ce qui signifie que l'accès est contrôlé sur chaque LUN, pas sur les partitions individuelles.

Tableau A.23. Fencing SCSI

Champ luci	Attribut `cluster.conf`	Description
Nom	**`name`**	Nom du périphérique fence SCSI.
Node name		
Clé pour l'action actuelle		(remplace le nom du nœud)

Le Tableau A.24, « Fencing VMware (interface SOAP) (Red Hat Enterprise Linux 6.2 et versions plus récentes) » répertorie les paramètres du périphérique fence utilisés par **`fence_vmware_soap`**, l'agent fence pour VMWare sur SOAP API.

Tableau A.24. Fencing VMware (interface SOAP) (Red Hat Enterprise Linux 6.2 et versions plus récentes)

Champ luci	Attribut `cluster.conf`	Description
Nom	**`name`**	Nom du périphérique de fencing de la machine virtuelle.
Adresse IP ou nom d'hôte	**`ipaddr`**	Adresse IP ou nom d'hôte assigné au périphérique.
Port IP (optionnel)	**`ipport`**	Port TCP à utiliser pour une connexion avec le périphérique.
Connexion	**`login`**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**`passwd`**	Mot de passe utilisé pour authentifier la connexion au périphérique.

Champ luci	Attribut cluster.conf	Description
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Séparateur	**separator**	Séparateur pour CSV créé par la liste des opérations. La valeur par défaut est une virgule (« , »).
Délai de l'alimentation	**power_wait**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Nom de la VM	**port**	Nom de la machine virtuelle sous le format de chemin d'inventaire (par exemple, /datacenter/vm/Discovered_virtual_machine/myMachine).
UUID de la VM	**uuid**	UUID de la machine virtuelle sur laquelle effectuer le fencing.
Utiliser SSH	**ssl**	Utilisez les connexions SSL pour communiquer avec le périphérique.

Le Tableau A.25, « Interrupteur d'alimentation WTI » répertorie les paramètres du périphérique fence utilisés par **fence_wti**, l'agent fence pour l'interrupteur d'alimentation réseau WTI.

Tableau A.25. Interrupteur d'alimentation WTI

Champ luci	Attribut cluster.conf	Description
Nom	**name**	Nom de l'interrupteur d'alimentation WTI connecté au cluster.
Adresse IP ou nom d'hôte	**ipaddr**	Adresse IP ou adresse du nom d'hôte assignée au périphérique.
Port IP (optionnel)	**ipport**	Port TCP à utiliser pour se connecter au périphérique.
Connexion	**login**	Nom de connexion utilisé pour accéder au périphérique.
Mot de passe	**passwd**	Mot de passe utilisé pour authentifier la connexion au périphérique.
Script de mot de passe (optionnel)	**passwd_script**	Script fournissant un mot de passe pour accéder au périphérique fence. Son utilisation supplante le paramètre **Password**.
Port	**port**	Numéro de la prise physique ou nom de machine virtuelle.

Champ luci	Attribut `cluster.conf`	Description
Force command prompt	**`cmd_prompt`**	Invite de commande à utiliser. La valeur par défaut est [’RSM>’, ’>MPC’, ’IPS>’, ’TPS>’, ’NBB>’, ’NPS>’, ’VMR>’]
Délai de l'alimentation	**`power_wait`**	Nombre de secondes d'attente après avoir effectué une commande de mise hors tension ou de mise sous tension.
Utiliser SSH	**`secure`**	Indique que le système utilisera SSH pour accéder au périphérique.
Chemin vers le fichier d'identité SSH	**`identity_file`**	Fichier d'identité de SSH.

ANNEXE B. PARAMÈTRES DES RESSOURCES HA

Cet annexe fournit les descriptions des paramètres des ressources HA. Vous pouvez configurer les paramètres avec **luci**, en utilisant la commande **ccs**, ou en modifiant le fichier **etc/cluster/cluster.conf**. Le Tableau B.1, « Sommaire des ressources HA » répertorie les ressources, leurs agents de ressources correspondants, et les références aux autres tableaux contenant des descriptions de paramètres. Pour mieux comprendre les agents de ressources, vous pouvez les voir dans le fichier **/usr/share/cluster** de chaque nœud du cluster.

En outre des agents de ressources décrits dans cet annexe, le répertoire **/usr/share/cluster** inclut un script OCF factice pour un groupe de ressources, **service.sh**. Pour obtenir des informations supplémentaires sur les paramètres inclus dans ce script, reportez-vous au script **service.sh**.

Pour obtenir la liste et les descriptions complètes des éléments et attributs de **cluster.conf**, reportez-vous au schéma du cluster sous **/usr/share/cluster/cluster.rng**, et au schéma sous **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

Tableau B.1. Sommaire des ressources HA

Ressource	Agent de ressources	Références aux descriptions des paramètres
Apache	apache.sh	Tableau B.2, « Serveur Apache »
Instance Condor	condor.sh	Tableau B.3, « Instance Condor »
Système de fichiers	fs.sh	Tableau B.4, « Système de fichiers »
Système de fichiers GFS2	clusterfs.sh	Tableau B.5, « GFS2 »
Adresse IP	ip.sh	Tableau B.6, « Adresse IP »
LVM HA	lvm.sh	Tableau B.7, « LVM HA »
MySQL	mysql.sh	Tableau B.8, « MySQL »
Client NFS	nfsclient.sh	Tableau B.9, « Client NFS »
Export NFS	nfsexport.sh	Tableau B.10, « Export NFS »
Serveur NFS	nfsserver.sh	Tableau B.11, « Serveur NFS »
Montage NFS/CIFS	netfs.sh	Tableau B.12, « Montage NFS/CIFS »
Open LDAP	openldap.sh	Tableau B.13, « Open LDAP »
Instance de basculement Oracle 10g/11g	oracledb.sh	Tableau B.14, « Instance de basculement Oracle 10g/11g »

Ressource	Agent de ressources	Références aux descriptions des paramètres
Instance de basculement Oracle 10g	orainstance.sh	Tableau B.15, « Instance de basculement Oracle 10g »
Listener Oracle 10g	oralistener.sh	Tableau B.16, « Listener Oracle 10g »
PostgreSQL 8	postgres-8.sh	Tableau B.17, « PostgreSQL 8 »
SAP Database	SAPDatabase	Tableau B.18, « SAP Database »
SAP Instance	SAPInstance	Tableau B.19, « SAP Instance »
Samba	samba.sh	Tableau B.20, « Serveur Samba »
Script	script.sh	Tableau B.21, « Script »
Sybase ASE	ASEHAagent.sh	Tableau B.22, « Instance de basculement ASE Sybase »
Tomcat 6	tomcat-6.sh	Tableau B.23, « Tomcat 6 »
Virtual Machine	vm.sh	Tableau B.24, « Virtual Machine » REMARQUE : **luci** affiche ceci en tant que service virtuel si le cluster hôte peut prendre en charge les machines virtuelles.

Tableau B.2. Serveur Apache

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Nom du service Apache.
Serveur root	**`server_root`**	La valeur pas défaut est **`/etc/httpd`**.
Fichier de configuration	**`config_file`**	Spécifie le fichier de configuration. La valeur par défaut est **`/etc/httpd/conf`**.
Options httpd	**`httpd_options`**	Autres options en ligne de commande pour **`httpd`**.
Attente fermeture (en secondes)	**`shutdown_wait`**	Spécifie le nombre de secondes à attendre pour la fermeture correcte de la fin du service.

Tableau B.3. Instance Condor

Champ	Champ luci	Attribut de `cluster.conf`
Nom d'instance	**`name`**	Spécifie un nom unique pour l'instance Condor.
Type de sous-système Condor	**`type`**	Spécifie le type de sous-système Condor pour cette instance : **`schedd`**, **`job_server`**, ou **`query_server`**.

Tableau B.4. Système de fichiers

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie un nom pour la ressources du système de fichiers.
Type de système de fichiers	**`fstype`**	Si non spécifié, **`mount`** essaie de déterminer le type de système de fichiers.
Point de montage	**`mountpoint`**	Chemin dans la hiérarchie du système de fichiers pour monter ce système de fichiers.
Device (périphérique), FS Label (étiquette FS), ou UUID	**`device`**	Spécifie le périphérique associé aux ressources du système de fichiers. Ceci peut être un périphérique bloc, une étiquette de système de fichiers, ou l'UUID d'un système de fichiers.
Options de montage	**`options`**	Options de montage ; options utilisées lorsque le système de fichiers est monté. Celles-ci peuvent être spécifique au système de fichiers. Reportez-vous à la page man ***`mount`***`(8)` pour voir les options de montage prises en charge.
ID du système de fichiers (optionnel)	**`fsid`**	 **NOTE** ***`ID du système de fichiers`*** est uniquement utilisé par les services NFS. Lors de la création d'une nouvelle ressource de système de fichiers, vous pouvez laisser ce champ vide. Laisser ce champ vide fait que l'ID du système de fichiers sera assigné automatiquement après avoir committé le paramètre pendant la configuration. Si vous devez assigner un ID de système de fichiers excplicitement, spécifiez-le dans ce champ.

Champ luci	Attribut de `cluster.conf`	Description
Forcer le démontage	**`force_unmount`**	Si activé, il force le système de fichier à se démonter. Le paramètre par défaut est ***`disabled`***. Lorsqu'il essaie d'effectuer le démontage, ***`Force Unmount`*** supprime tous les processus utilisant le point de montage afin de le libérer.
Forcer fsck	**`force_fsck`**	Si activé, **`fsck`** sera exécuté sur le système de ficheirs avant qu'il ne soit monté. Le paramètre par défaut est ***`disabled`***.
Activez le démon NFS et la solution de contournement lockd (Red Hat Enterprise Linux 6.4 et versions supérieures)	**`nfsrestart`**	Si votre système de fichiers est exporté via NFS et qu'il échoue occasionnellement à se démonter (lors d'une fermeture ou du transfert d'un service), le paramétrage de cette option effacera toute référence au système de fichiers avant l'opération de démontage. Le paramétrage de cette option requiert que vous activiez l'option **`Forcer le démontage`** et ne doit pas être utilisée avec la ressource **`NFS Server`**. Comme il s'agit d'une tentative forcée de démontage d'un système de fichiers, veuillez paramétrer cette option en dernier recours uniquement.
Utiliser les vérifications rapides de statut	**`quick_status`**	Si activé, effectue des vérifications rapides du statut.
Redémarrer le nœud hôte si le démontage échoue	**`self_fence`**	Si activé, redémarre le nœud le démontage du système de fichiers échoue. L'agent de ressources **`filesystem`** (système de fichiers) accepte les valeurs 1, **`yes`**, **`on`**, ou **`true`** pour activer ce paramètre et les valeurs 0, **`no`**, **`off`**, ou **`false`** pour le désactiver. Le paramètre par défaut est ***`disabled`*** (désactivé).

Tableau B.5. GFS2

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Nom de la ressource du système de fichiers.
Point de montage	**`mountpoint`**	Cheminselon lequel la ressource du système de fichiers est montée.
Device (périphérique), FS Label (étiquette FS), ou UUID	**`device`**	Fichier du périphérique associé à la ressource du système de fichiers.

Champ luci	Attribut de `cluster.conf`	Description
Type de système de fichiers	**`fstype`**	Paramétrer sur GFS2 sur **luci**
Options de montage	**`options`**	Options de montage.
ID du système de fichiers (optionnel)	**`fsid`**	 **NOTE** ***`ID du système de fichiers`*** est uniquement utilisé par les services NFS. Lors de la création d'une nouvelle ressource GFS2, vous pouvez laisser ce champ vide. Laisser ce champ vide fera que l'ID du système de fichiers sera assigné automatiquement après avoir committé le paramètre pendant la configuration. Si vous devez assigner un ID de système de fichiers explicitement, spécifiez-le dans ce champ.
Forcer le démontage	**`force_unmount`**	Si activé, il force le système de fichiers à se démonter. Le paramètre par défaut est ***`disabled`***. Lorsqu'il essaie d'effectuer le démontage, ***`Force Unmount`*** supprime tous les processus utilisant le point de montage afin de libérer celui-ci. Avec les ressources GFS2, le point de montage n'est *pas* démonté lors du service démontage à moins que ***`Force Unmount`*** (Forcer le démontage) ne soit *enabled* (activé).
Activez le démon NFS et la solution de contournement lockd (Red Hat Enterprise Linux 6.4 et versions supérieures)	**`nfsrestart`**	Si votre système de fichiers est exporté via NFS et qu'il échoue occasionnellement à se démonter (lors d'une fermeture ou du transfert d'un service), le paramétrage de cette option effacera toute référence au système de fichiers avant l'opération de démontage. Le paramétrage de cette option requiert que vous activiez l'option **`Forcer le démontage`** et ne doit pas être utilisée avec la ressource **`NFS Server`**. Comme il s'agit d'une tentative forcée de démontage d'un système de fichiers, veuillez paramétrer cette option en dernier recours uniquement.
Redémarrer le nœud hôte si le démontage échoue	**`self_fence`**	Si activer et démonter le système de fichiers échoue, le nœud redémarrera immédiatement. En général, ceci est utilisé en conjonction avec la prise en charge de force-unmount (forcer le démontage), mais n'est pas requis. L'agent de ressources **GFS2** accepte les valeurs 1, **`yes`**, **`on`**, ou **`true`** pour activer ce paramètre et les valeurs 0, **`no`**, **`off`**, ou **`false`** pour le désactiver.

Tableau B.6. Adresse IP

Champ luci	Attribut de `cluster.conf`	Description
IP Address (adresse IP), Netmask Bits (bits de masque réseau)	**`address`**	L'adresse IP (et optionnellement les bits du masque réseau) pour la ressource. Les bits du masque réseau, ou la longueur du préfixe réseau, peut se situer après l'adresse avec une barre oblique utilisée comme séparateur, en conformité avec la notation CIDR (par exemple, 10.1.1.1/8). Ceci est une adresse IP virtuelle. Les adresses IPv4 et IPv6 sont prises en charge, tout comme le contrôle du lien NIC pour chaque adresse IP.
Monitor Link	**`monitor_link`**	Activer ceci cause à la vérification du statut d'échouer si le lien sur le NIC vers lequel cette adresse IP se dirige n'est pas présent.
Désactiver les mises à jour des routes statiques	**`disable_rdisc`**	Désactiver les mises à jour du routage à l'aide du protocole RDISC.
Nombre de secondes de veille après la suppression d'une adresse IP	**`sleeptime`**	Spécifie le temps (en secondes) de veille.

Tableau B.7. LVM HA

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Nom unique pour cette ressource LVM.
Nom du groupe de volumes	**`vg_name`**	Nom descriptif du groupe de volumes géré.
Nom du volume logique (optionnel)	**`lv_name`**	Nom du volume logique géré. Ce paramètre est optionnel s'il y a plus d'un volume logique dans le groupe de volumes géré.
Fencing du nœud s'il est incapable de nettoyer les balises LVM	**`self_fence`**	Effectuez le fencing du nœud s'il est incapable de supprimer les balises LVM. L'agent de ressources LVM accepte les valeurs 1 ou **yes** pour activer ce paramètre et les valeurs 0 ou **no** pour le désactiver.

Tableau B.8. MySQL

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie un nom pour la ressource de MySQL server.
Fichier de configuration	**`config_file`**	Spécifie le fichier de configuration. La valeur par défaut est **`/etc/my.cnf`**.
Listen Address	**`listen_address`**	Spécifie une adresse IP pour MySQL server. Si aucune adresse IP n'est fournie, la première adresse IP du service sera utilisée.
Options mysqld	**`mysqld_options`**	Autres options en ligne de commande pour **`httpd`**.
Attente démarrage (en secondes)	**`startup_wait`**	Spécifie le nombre de secondes à attendre pour la fin correcte du démarrage du service.
Attente fermeture (en secondes)	**`shutdown_wait`**	Spécifie le nombre de secondes à attendre pour la fermeture correcte de la fin du service.

Tableau B.9. Client NFS

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Ceci est un nom symbolique d'un client habitué à y faire référence dans l'arborescence des ressources. Il ne s'agit *pas* de la même chose que l'option ***`Target`*** (Cible).
Nom d'hôte, caractère générique, ou netgroup de la cible	**`target`**	Serveur à partir duquel vous effectuez le montage. Il peut être spécifié à l'aide d'un nom d'hôte, d'un caractère de remplacement (basé sur adresse IP ou nom d'hôte), ou d'un netgroup définissant un hôte ou des hôtes vers lequel (ou lesquels) exporter.
Autoriser la récupération de ce client NFS	**`allow_recover`**	Autorise la récupération.
Options	**`options`**	Définit une liste d'options pour ce client — par exemple, des droits d'accès client supplémentaires. Pour plus d'informations, reportez-vous aux *Options générales* de la page man ***`exports`*** *(5)*.

Tableau B.10. Export NFS

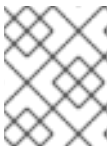
Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Nom descriptif de la ressource. La ressource NFS Export s'assure que les démons NFS sont en cours d'exécution. Elle est entièrement réutilisable ; habituellement, seule une ressource NFS Export est nécessaire.  **NOTE** Nom de la ressource NFS Export, afin qu'elle soit clairement distinguable des autres ressources NFS.

Tableau B.11. Serveur NFS

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Nom descriptif de la ressource du serveur NFS. La ressource du serveur NFS est utile pour exporter des systèmes de fichiers NFSv4 sur des clients. À cause de la manière dont NFSv4 fonctionne, seule une ressource NFSv4 peut exister sur un serveur à la fois. En outre, il n'est pas possible d'utiliser la ressource de serveur NFS lorsque des instances locales de NFS sont aussi utilisées sur chaque nœud de cluster.

Tableau B.12. Montage NFS/CIFS

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Nom symbolique du montage NFS ou CIFS.  **NOTE** Cette ressource est requise lorsqu'un service cluster est configuré de manière à être un client NFS.
Point de montage	**`mountpoint`**	Chemin sur lequel la ressource du système de fichiers est montée.
Hôte	**`host`**	Adresse IP ou nom d'hôte du serveur NFS/CIFS.

Champ luci	Attribut de `cluster.conf`	Description
Nom du répertoire NFS Export ou nom du partage CIFS	**`export`**	Nom du répertoire NFS Export ou nom du partage CIFS.
Type de système de fichiers	**`fstype`**	Type de système de fichiers : • ***`NFS`*** — Spécifie l'utilisation de la version par défaut de NFS. Ceci est le paramètre par défaut. • ***`NFS v4`*** — Spécifie l'utilisation du protocole NFSv4. • ***`CIFS`*** — Spécifie l'utilisation du protocole CIFS.
Forcer le démontage	**`force_unmount`**	Si ***`Force Unmount`*** (Forcer le démontage) est activé, le cluster supprime tous les processus à l'aide de ce système de fichiers lorsque le service est arrêté. La suppression de tous les processus à l'aide du système de fichiers libère l'espace du système de fichiers. Autrement, le démontage échouera et le service sera redémarré.
Ne pas démonter le système de fichiers pendant une opération d'arrêt ou de déplacement.	**`no_unmount`**	Si activé, cela spécifie que le système de fichiers ne doit pas être démonté pendant une opération d'arrêt ou de déplacement.
Options	**`options`**	Options de montage. Spécifie une liste des options de montage. Si aucune n'est spécifiée, le système de fichiers est monté **`-o sync`**.

Tableau B.13. Open LDAP

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie un nom de service pour la connexion et pour d'autres raisons.
Fichier de configuration	**`config_file`**	Spécifie un chemin absolu vers un fichier de configuration. La valeur par défaut est **`/etc/openldap/slapd.conf`**.
Liste des URL	**`url_list`**	La valeur par défaut est **`ldap:///`**.

Champ luci	Attribut de `cluster.conf`	Description
Options `slapd`	`slapd_options`	Autres options en ligne de commande de `slapd`.
Attente fermeture (en secondes)	`shutdown_wait`	Spécifie le nombre de secondes à attendre pour la fermeture correcte de la fin du service.

Tableau B.14. Instance de basculement Oracle 10g/11g

Champ luci	Attribut de `cluster.conf`	Description
Nom d'instance (SID) de l'instance Oracle	`name`	Nom d'instance.
Nom d'utilisateur Oracle	`user`	Ceci est le nom d'utilisateur de l'utilisateur Oracle sous lequel l'instance AS d'Oracle est exécutée.
Répertoire de base de l'application Oracle	`home`	Ceci est le répertoire de base d'Oracle (l'application, et non l'utilisateur). Il est configuré lorsque vous installez Oracle.
Type d'installation Oracle	`type`	Type d'installation Oracle. Par défaut : `10g`, Database Instance and Listener Only `base`, Database, Listener, Enterprise Manager et ISQL*Plus : `base-em` (ou `10g`), ou Internet Application Server (infrastructure) : `ias` (ou `10g-ias`).
Nom d'hôte virtuel (optionnel)	`vhost`	Nom d'hôte virtuel correspondant au nom d'hôte de l'installation d'Oracle 10g. Remarquez que pendant le démarrage/arrêt d'une ressource oracledb, votre nom d'hôte est temporairement modifié sous ce nom d'hôte. Ainsi, vous devriez configurer une ressource oracledb faisant partie d'un service exclusif uniquement.

Tableau B.15. Instance de basculement Oracle 10g

Champ luci	Attribut de `cluster.conf`	Description
Nom d'instance (SID) de l'instance Oracle	**`name`**	Nom d'instance.
Nom d'utilisateur Oracle	**`user`**	Ceci est le nom d'utilisateur de l'utilisateur Oracle sous lequel l'instance d'Oracle est exécutée.
Répertoire de base de l'application Oracle	**`home`**	Ceci est le répertoire de base d'Oracle (l'application, et non l'utilisateur). Il est configuré lorsque vous installez Oracle.
Liste des listeners Oracle (optionnels, séparés par des espaces)	**`listeners`**	Liste des listeners Oracle qui seront lancés avec l'instance de la base de données. Les noms de listeners sont séparés par des espaces vides. Vide par défaut, ce qui désactive les listeners.
Chemin vers le fichier verrou (optionnel)	**`lockfile`**	Emplacement du fichier verrou qui sera utilisé pour vérifier si Oracle devrait être exécuté ou non. Se met par défaut sous l'emplacement **`/tmp`**.

Tableau B.16. Listener Oracle 10g

Champ luci	Attribut de `cluster.conf`	Description
Nom du listener	**`name`**	Nom du listener.
Nom d'utilisateur Oracle	**`user`**	Ceci est le nom d'utilisateur de l'utilisateur Oracle sous lequel l'instance d'Oracle est exécutée.
Répertoire de base de l'application Oracle	**`home`**	Ceci est le répertoire de base d'Oracle (l'application, et non l'utilisateur). Il est configuré lorsque vous installez Oracle.

Tableau B.17. PostgreSQL 8

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie un nom de service pour la connexion et pour d'autres raisons.
Fichier de configuration	**`config_file`**	Définit un chemin absolu vers le fichier de configuration. La valeur par défaut est **`/var/lib/pgsql/data/postgresql.conf`**.
Postmaster User	**`postmaster_user`**	Utilisateur qui exécute le serveur de la base de données car elle ne peut être exécutée par root. La valeur par défaut est postgres.
Options Postmaster	**`postmaster_options`**	Autres options en ligne de commande de postmaster.
Attente fermeture (en secondes)	**`shutdown_wait`**	Spécifie le nombre de secondes à attendre pour la fermeture correcte de la fin du service.

Tableau B.18. SAP Database

Champ luci	Attribut de `cluster.conf`	Description
Nom de base de données SAP	**`SID`**	Spécifie un identifiant de système SAP unique. Par exemple, P01.
Répertoire exécutable SAP	**`DIR_EXECUTABLE`**	Spécifie le chemin d'accès complet vers **`sapstartsrv`** et **`sapcontrol`**.
Type de base de données	**`DBTYPE`**	Spécifie un type des bases de données suivantes : Oracle, DB6, ou ADA.
Nom du listener Oracle	**`NETSERVICENAME`**	Spécifie le nom de l'écouteur (listener) TNS d'Oracle.
La pile ABAP n'est pas installée, seule la pile Java est installée	**`DBJ2EE_ONLY`**	Si aucune pile ABAP n'est installée dans la base de données SAP, activez ce paramètre.
Surveillance du niveau des applications	**`STRICT_MONITORING`**	Active la surveillance du niveau des applications.

Champ luci	Attribut de `cluster.conf`	Description
Récupération du démarrage automatique (« Automatic Startup Recovery »)	**`AUTOMATIC_RECOVER`**	Activer ou désactiver la récupération du démarrage automatique.
Chemin vers Java SDK	**`JAVE_HOME`**	Chemin vers Java SDK.
Nom du fichier du pilote JDBC	**`DB_JARS`**	Nom de fichier du pilote JDBC.
Chemin vers un script pré-démarrage	**`PRE_START_USEREXIT`**	Chemin vers un script pré-démarrage.
Chemin vers un script post-démarrage	**`POST_START_USEREXIT`**	Chemin vers un script post-démarrage.
Chemin vers un script pré-arrêt	**`PRE_STOP_USEREXIT`**	Chemin vers un script pré-arrêt
Chemin vers un script post-arrêt	**`POST_STOP_USEREXIT`**	Chemin vers un script post-arrêt
Répertoire « Bootstrap » de l'instance J2EE	**`DIR_BOOTSTRAP`**	Chemin d'accès complet du répertoire de démarrage de l'instance J2EE. Par exemple, **`/usr/sap/P01/J00/j2ee/cluster/bootstrap`**.
Chemin du stockage de sécurité J2EE	**`DIR_SECSTORE`**	Chemin d'accès complet du répertoire de stockage de sécurité J2EE. Par exemple, **`/usr/sap/P01/SYS/global/security/lib/tools`**.

Tableau B.19. SAP Instance

Champ luci	Attribut de `cluster.conf`	Description
Nom d'instance SAP	**`InstanceName`**	Nom complet de l'instance SAP. Par exemple, P01_DVEBMGS00_sapp01ci.

Champ luci	Attribut de `cluster.conf`	Description
Répertoire exécutable SAP	**DIR_EXECUTABLE**	Chemin d'accès complet vers **sapstartsrv** et **sapcontrol**.
Répertoire contenant le profil START SAP.	**DIR_PROFILE**	Chemin d'accès complet vers le profil START SAP.
Nom du profil START SAP	**START_PROFILE**	Spécifie le nom du profil START SAP.
Nombre de secondes d'attente avant la vérification du statut du démarrage	**START_WAITTIME**	Spécifie le nombre de secondes à attendre avant de vérifier le statut du démarrage (ne pas attendre J2EE-Addin).
Activer la récupération du démarrage automatique (« Automatic Startup Recovery »)	**AUTOMATIC_RECOVER**	Activer ou désactiver la récupération du démarrage automatique.
Chemin vers un script pré-démarrage	**PRE_START_USEREXIT**	Chemin vers un script pré-démarrage.
Chemin vers un script post-démarrage	**POST_START_USEREXIT**	Chemin vers un script post-démarrage.
Chemin vers un script pré-arrêt	**PRE_STOP_USEREXIT**	Chemin vers un script pré-arrêt
Chemin vers un script post-arrêt	**POST_STOP_USEREXIT**	Chemin vers un script post-arrêt

NOTE

Concernant le Tableau B.20, « Serveur Samba », lors de la création ou de la modification d'un service cluster, connectez une ressource du service Samba directement au service, et *non* à une ressource faisant partie d'un service.

Tableau B.20. Serveur Samba

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie le nom du serveur Samba.
Fichier de configuration	**`config_file`**	Fichier de configuration Samba
Autres options en ligne de commande de smbd	**`smbd_options`**	Autres options en ligne de commande de smbd.
Autres options en ligne de commande de nmbd	**`nmbd_options`**	Autres options en ligne de commande de nmbd.
Attente fermeture (en secondes)	**`shutdown_wait`**	Spécifie le nombre de secondes à attendre pour la fermeture correcte de la fin du service.

Tableau B.21. Script

Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie un nom pour le script personnalisé de l'utilisateur. La ressource script permet à un script init compatible avec LSB d'être utilisé pour démarrer un service clusterisé.
Chemin complet vers le fichier script	**`file`**	Saisir le chemin d'accès de ce script personnalisé (par exemple, **`/etc/init.d/`*`userscript`***).

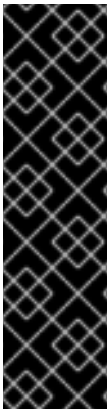
Tableau B.22. Instance de basculement ASE Sybase

Champ luci	Attribut de `cluster.conf`	Description
Nom d'instance	**`name`**	Spécifie le nom d'instance de la ressource ASE Sybase.
Nom du serveur ASE	**`server_name`**	Nom du serveur ASE configuré pour le service HA.
Répertoire de base SYBASE	**`sybase_home`**	Répertoire de base des produits Sybase.
Fichier de connexion	**`login_file`**	Chemin d'accès complet du fichier de connexion qui contient la paire identifiant-mot de passe.
Fichier des interfaces	**`interfaces_file`**	Chemin d'accès complet du fichiers des interfaces utilisé pour démarrer/accéder au serveur ASE.
Nom du répertoire SYBASE_ASE	**`sybase_ase`**	Nom du répertoire sour sybase_home où les produits ASE sont installés.
Nom du répertoire SYBASE_OCS	**`sybase_ocs`**	Nom du répertoire sous sybase_home où les produits OCS sont installés. Par exemple, ASE-15_0.
Utilisateur Sybase	**`sybase_user`**	Utilisateur pouvant exécuter le serveur ASE.
Attente démarrage (en secondes)	**`start_timeout`**	Valeur du délai du démarrage.
Attente fermeture (en secondes)	**`shutdown_timeout`**	Valeur du délai de fermeture.
Délai d'expiration Deep Probe	**`deep_probe_timeout`**	Le nombre maximum de secondes de l'attente pour une réponse du serveur ASE avant de déterminer que le serveur n'a pas de réponse pendant l'exécution de Deep Probe.

Tableau B.23. Tomcat 6

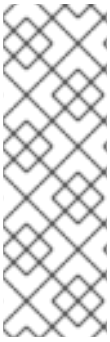
Champ luci	Attribut de `cluster.conf`	Description
Nom	**`name`**	Spécifie un nom de service pour la connexion et pour d'autres raisons.

Champ luci	Attribut de `cluster.conf`	Description
Fichier de configuration	**`config_file`**	Spécifie le chemin d'accès absolu du fichier de configuration. La valeur par défaut est **`/etc/tomcat6/tomcat6.conf`**.
Attente fermeture (en secondes)	**`shutdown_wait`**	Spécifie le nombre de secondes d'attendre de la fin correcte de la fermeture du service. La valeur par défaut est 30.



IMPORTANT

Concernant le Tableau B.24, « Virtual Machine », lorsque vous configurez votre cluster avec les ressources d'une machine virtuelle, vous devriez utiliser les outils **`rgmanager`** pour démarrer et arrêter les machines virtuelles. L'utilisation de **`virsh`** pour démarrer une machine peut entraîner l'exécution de la machine virtuelle dans plusieurs emplacements, ce qui peut provoquer une corruption de données dans celle-ci. Pour obtenir des informations sur la configuration de votre système pour réduire la possibilité qu'un administrateur effectue un « double-démarrage » accidentel en utilisant les outils du cluster et des outils non-clusterisés, reportez-vous à la Section 2.14, « Configurer des machines virtuelles dans un environnement clusterisé ».



NOTE

Les ressources de machines virtuelles sont configurées différemment des autres ressources de cluster. Pour configurer une machine virtuelle avec **luci**, veuillez ajouter un groupe de services au cluster, puis ajoutez une ressource au service, en sélectionnant **`Virtual Machine`** en tant que type de ressource virtuelle et en saisissant les paramètres des ressource de la machine virtuelle. Pour obtenir des informations sur la configuration d'une machine virtuelle avec la commande **`ccs`**, reportez-vous à la Section 5.12, « Ressources de machine virtuelle ».

Tableau B.24. Virtual Machine

Champ luci	Attribut de `cluster.conf`	Description
Nom du service	**`name`**	Spécifie le nom de la machine virtuelle. Lors de l'utilisation de l'interface **luci**, vous pouvez spécifier ceci en tant que nom de service.
Démarrer ce service automatiquement	**`autostart`**	Si activé, cette machine virtuelle est démarrée automatiquement une fois que le cluster atteint le quorum. Si ce paramètre est *désactivé*, cette machine virtuelle ne sera *pas* démarrée automatiquement une fois que le cluster aura atteint le quorum, et elle entrera alors dans l'état ***`disabled`*** (déactivé).

Champ luci	Attribut de `cluster.conf`	Description
Exécuter de manière exclusive	**`exclusive`**	Si activé, cette machine virtuelle peut uniquement être déplacée pour être exécutée sur un autre nœud de manière exclusive. C'est-à-dire de s'exécuter sur un nœud sur lequel aucune autre machine virtuelle n'est exécutée. S'il n'y a pas d'autre nœud disponible pour qu'une machine virtuelle puisse être exécutée de manière exclusive, alors la machine virtuelle ne sera pas redémarrée après cet échec. En outre, les autres machines virtuelles ne se déplacent pas automatiquement sur un nœud qui exécute cette machine virtuelle en tant que ***`Run exclusive`*** (Exécuter de manière exclusive). Vous pouvez outrepasser cette option avec un démarrage manuel ou des opérations de déplacement.
Domaine de basculement	**`domain`**	Définit une liste des membres du cluster à essayer au cas où une machine virtuelle échouerait.
Stratégie de récupération	**`recovery`**	La ***`Recovery policy`*** (stratégie de récupération) offre les options suivantes : • ***`Disable`*** — Désactive la machine virtuelle si elle échoue. • ***`Relocate`*** — Tente de redémarrer la machine virtuelle dans un autre nœud (ne tente pas de redémarrer avec le nœud actuel). • ***`Restart`*** — Tente de redémarrer la machine virtuelle localement (dans le nœud actuel) avant de tenter de la déplacer (par défaut) vers un autre nœud. • ***`Restart-Disable`*** — Le service sera redémarré s'il échoue. Cependant, si le redémarrage du service échoue, celui-ci sera désactivé, au lieu d'être déplacé sur un autre hôte dans le cluster.
Options de redémarrage	**`max_restarts`**, **`restart_expire_time`**	Avec **Restart** ou **Restart-Disable** sélectionné comme politique de récupération pour un service, ceci spécifie le nombre maximum d'échec du redémarrage avant que le déplacement ou la désactivation du service ne soit effectué. Spécifie aussi le temps en secondes au bout duquel il faut arrêter de redémarrer.
Type de migration	**`migrate`**	Spécifie un type de migration ***`live`*** ou ***`pause`***. Le paramètre par défaut est ***`live`***.

Champ luci	Attribut de `cluster.conf`	Description
Mappage de migration	**`migration_mapping`**	Spécifie une interface de migration alternative. Vous pouvez spécifier ceci par exemple lorsque l'adresse réseau utilisée pour la migration de la machine virtuelle sur un nœud est différente de l'adresse du nœud utilisée pour les communications du cluster. La spécification de ce qui suit indique que lorsque vous migrer un machine virtuelle de **`member`** à **`member2`**, vous effectuez en fait une migration vers **`target2`**. De la même manière, lorsque vous effectuez une migration de **`member2`** à **`member`**, la migration est effectuée avec **`target`**. **`member:target,member2:target2`**
Status Program	**`status_program`**	État du programme à exécuter en plus de la vérification standard de la présence d'une machine virtuelle. Si spécifié, l'état du programme est exécuté une fois par minute. Ceci vous permet de déterminer l'état des services critiques dans une machine virtuelle. Par exemple, si une machine virtuelle exécute un serveur web, votre état du programme peut vérifier si un serveur web fonctionne correctement ; si la vérification de l'état échoue, (qui est signifiée par le retour d'une valeur inégale à zéro), la machine virtuelle est récupérée. Une fois qu'une machine virtuelle est démarrée, l'agent de ressources de la machine virtuelle appellera l'état du programme et attendra le retour d'un code de réussite (zéro) avant de revenir. Le délai d'expiration est fixé à 5 minutes.
Chemin d'accès xmlfile utilisé pour créer la VM	**`xmlfile`**	Chemin d'accès complet du fichier XML **`libvirt`** contenant la définition du domaine **`libvirt`**.
Chemin d'accès du fichier de configuration de la VM	**`path`**	Spécification du chemin d'accès délimitée par le caractère deux-points (:) indiquant que l'agent des ressources de la machine virtuelle (**`vm.sh`**) recherche le fichier de configuration de la machine virtuelle. Par exemple : **`/mnt/guests/config:/etc/libvirt/qemu`**.  **IMPORTANT** Le chemin d'accès ne doit *jamais* directement pointer vers le fichier de configuration d'une machine virtuelle.
Chemin d'accès du répertoire VM snapshot	**`snapshot`**	Chemin d'accès du répertoire d'instantanés où l'image de la machine virtuelle sera stockée.
URI de l'hyperviseur	**`hypervisor_uri`**	URI de l'hyperviseur (habituellement automatique).

Champ luci	Attribut de `cluster.conf`	Description
URI de la migration	**`migration_uri`**	URI de la migration (habituellement automatique).
Données du tunnel sur ssh pendant la migration	**`tunnelled`**	Données du tunnel sur ssh pendant la migration.

ANNEXE C. COMPORTEMENT DES RESSOURCES HA

Cet annexe décrit le comportement habituel des ressources HA. Il est conçu dans le but de fournir des informations accessoires pouvant être utiles lors de la configuration des services HA. Vous pouvez configurer les paramètres avec **Luci** ou en modifiant **`/etc/cluster/cluster.conf`**. Pour obtenir des descriptions des paramètres de ressources HA, reportez-vous à l'Annexe B, *Paramètres des ressources HA*. Pour mieux comprendre les agents de ressources, vous pouvez les voir dans le fichier **`/usr/share/cluster`** de chaque nœud du cluster.

NOTE

Pour bien comprendre les informations présentes dans cet annexe, vous devrez posséder une compréhension détaillée des agents de ressources et du fichier de configuration du cluster, **`/etc/cluster/cluster.conf`**.

Un service HA est un groupe de ressources de cluster configurées en une entité cohérente fournissant des services spécialisés aux clients. Un service HA est représenté comme une arborescence de ressources dans le fichier de configuration du cluster **`/etc/cluster/cluster.conf`** (dans chaque nœud du cluster). Dans le fichier de configuration du lcuster, chaque arborescence de ressources est une représentation XML spécifiant chaque ressource, ses attributs, et ses relations aux autres ressources dans l'arborescence des ressources (parents, enfants et de même parenté).

NOTE

Comme un service HA est composé de ressources organisées en une arborescence hiérarchique, on peut parfois faire référence à un service en tant qu'*arborescence de ressources* ou que *groupe de ressources*. Les deux termes sont synonymes de *service HA*.

À la racine de chaque arborescence de ressources se trouve un type de ressources spécial — une *ressource de service*. Les autres types de ressources comprennent le reste d'un service, déterminant ainsi ses caractéristiques. Configurer un service HA revient à créer une ressource de service, créer des ressources de cluster subordinées et les organiser en une entité cohérente conforme aux restrictions hiérarchiques du service.

Cet annexe est composé des sections suivantes :

- Section C.1, « Relations entre parents, enfants, et enfants de mêmes parents parmi les ressources »
- Section C.2, « Ordre de démarrage des relations de même parenté et ordre des enfants de ressources »
- Section C.3, « Héritage, le bloc <ressources>, et la réutilisation des ressources »
- Section C.4, « Récupération de défaillance et sous-arbres indépendants »
- Section C.5, « Débogage et testage des services et de l'ordre des ressources »

NOTE

Les sections qui suivent les exemples présents du fichier de configuration du cluster **`/etc/cluster/cluster.conf`**, sont à des fins d'illustration uniquement.

C.1. RELATIONS ENTRE PARENTS, ENFANTS, ET ENFANTS DE MÊMES PARENTS PARMI LES RESSOURCES

Un service cluster est une entité intégrée qui est exécutée sous le contrôle de **rgmanager**. Toutes les ressources d'un service sont exécutées sur le même nœud. Du point de vue de **rgmanager**, un service cluster est une entité qui peut être démarrée, arrêtée, ou déplacée. Cependant, à l'intérieur d'un service clusterm la hiérarchie des ressources détermine l'ordredans lequel chaque ressource est démarrée puis arrêtée. Les niveaux de hiérarchie sont : parent, enfant, et de même parenté.

Exemple C.1, « Hiérarchie des ressources du service foo » affiche un exemple d'arborescence de ressources du service *foo*. Dans l'exemple, les relations entre les ressources sont comme suit :

- **fs:myfs** (<fs name="myfs" ...>) et **ip:10.1.1.2** (<ip address="10.1.1.2 .../>) sont de même parenté.
- **fs:myfs** (<fs name="myfs" ...>) is the parent of **script:script_child** (<script name="script_child"/>).
- **script:script_child** (<script name="script_child"/>) est l'enfant de **fs:myfs** (<fs name="myfs" ...>).

Exemple C.1. Hiérarchie des ressources du service foo

```
<service name="foo" ...>
    <fs name="myfs" ...>
        <script name="script_child"/>
    </fs>
    <ip address="10.1.1.2" .../>
</service>
```

Les règles suivantes s'appliquent aux relations parents/enfants dans une arborescence de ressources :

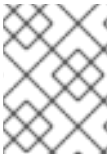
- Les parents sont démarrés avant les enfants.
- Les enfants doivent tous s'arrêter correctement avant qu'un parent puisse être arrêté.
- Pour qu'une ressource soit considérée comme étant en bonne santé, tous ses enfants doivent être en bonne santé.

C.2. ORDRE DE DÉMARRAGE DES RELATIONS DE MÊME PARENTÉ ET ORDRE DES ENFANTS DE RESSOURCES

La ressource Service détermine l'ordre de démarrage et l'ordre d'arrêt des ressources enfants si elle désigne un attribut de type enfant pour une ressource enfant comme suit :

- Désigne un attribut de type enfant (ressource enfant *typée*) — Si la ressource Service désigne un attribut de type enfant pour une ressource enfant, la ressource enfant est *typée*. L'attribut de type enfant détermine de manière explicite l'ordre de début et de fin de la ressource enfant.
- *Ne désigne pas* d'attributs de type enfant (ressource enfant *non-typée*) — Si la ressource Service *ne désigne pas* un attribut de type enfant pour une ressource enfant, la ressource enfant est *non-typée*. La ressource Service ne contrôle pas explicitement les ordres de démarrage et d'arrêt d'une ressource enfant non-typée. Cependant, une ressource enfant non-

typée est démarrée et arrêtée en fonction de son ordre dans **`/etc/cluster/cluster.conf`**. En outre, les ressources enfant non-typées sont démarrées une fois que toutes les ressources enfants typées sont démarrées et elles sont arrêtées avant que toute ressource enfant typée ne soit arrêtée.



NOTE

La ressource Service est la seule ressource implémentant un ordre *de type de ressource enfant* défini.

Pour obtenir plus d'informations sur l'ordre de démarrage et d'arrêt des ressources enfants typées, reportez-vous à la Section C.2.1, « Ordre de démarrage et d'arrêt des ressources enfant typées ». Pour obtenir plus d'information sur l'ordre de démarrage et d'arrêt des ressources enfants non-typées, reportez-vous à la Section C.2.2, « Ordre de démarrage et d'arrêt de ressources enfant non-typées ».

C.2.1. Ordre de démarrage et d'arrêt des ressources enfant typées

Pour une ressource enfant typée, l'attribut du type de la ressource enfant définit l'ordre de démarrage et d'arrêt de chaque type de ressource avec un numéro pouvant aller de 1 à 100 ; une valeur pour le démarrage et une valeur pour l'arrêt. Plus le numéro est bas, plus le type de ressource démarrera ou s'arrêtera tôt. Par exemple, le Tableau C.1, « Ordre de démarrage et d'arrêt des ressources enfants » affiche les valeurs de démarrage et d'arrêt pour chaque type de ressource ; l'Exemple C.2, « Valeurs de démarrage et d'arrêt de la ressource : extraites de l'agent de la ressource Service **`service.sh`** » affiche les valeurs de démarrage et d'arrêt telles qu'elles apparaissent sur l'agent de la ressource Service **`service.sh`**. Pour la ressource Service, tous les enfants LVM sont démarrés en premier, suivis par tous les enfant systèmes de fichiers, suivis par tous les enfants scripts, et ainsi de suite.

Tableau C.1. Ordre de démarrage et d'arrêt des ressources enfants

Ressource	Type d'enfant	Valeur de l'ordre de démarrage	Valeur de l'ordre d'arrêt
LVM	lvm	1	9
Système de fichiers	fs	2	8
Système de fichiers GFS2	clusterfs	3	7
Montage NFS	netfs	4	6
Export NFS	nfsexport	5	5
Client NFS	nfsclient	6	4
IP Address	ip	7	2
Samba	smb	8	3
Script	script	9	1

Exemple C.2. Valeurs de démarrage et d'arrêt de la ressource : extraites de l'agent de la ressource Service `service.sh`

```
<special tag="rgmanager">
    <attributes root="1" maxinstances="1"/>
    <child type="lvm" start="1" stop="9"/>
    <child type="fs" start="2" stop="8"/>
    <child type="clusterfs" start="3" stop="7"/>
    <child type="netfs" start="4" stop="6"/>
    <child type="nfsexport" start="5" stop="5"/>
    <child type="nfsclient" start="6" stop="4"/>
    <child type="ip" start="7" stop="2"/>
    <child type="smb" start="8" stop="3"/>
    <child type="script" start="9" stop="1"/>
</special>
```

Le classement dans un type de ressource est préservé lorsqu'il est fermé dans le fichier de configuration du cluster **`/etc/cluster/cluster.conf`**. Par exemple, prenez en considération l'ordre de démarrage et d'arrêt des ressources enfants typées dans l'Exemple C.3, « Classement dans un type de ressource ».

Exemple C.3. Classement dans un type de ressource

```
<service name="foo">
  <script name="1" .../>
  <lvm name="1" .../>
  <ip address="10.1.1.1" .../>
  <fs name="1" .../>
  <lvm name="2" .../>
</service>
```

Ordre de démarrage de ressource enfant typée

Dans l'Exemple C.3, « Classement dans un type de ressource », les ressources sont démarrées dans l'ordre suivant :

1. **`lvm:1`** — Ceci est une ressource LVM, Toutes les ressources LVM sont démarrées en premier. **`lvm:1`** (**`<lvm name="1" .../>`**) est la première ressource LVM démarrée dar il s'agit de la première ressource LVM répertoriée dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

2. **`lvm:2`** — Ceci est une ressource LVM. Toutes les ressources LVM sont démarrées en premier. **`lvm:2`** (**`<lvm name="2" .../>`**) est démarré après **`lvm:1`** car **`lvm:2`** est répertorié après **`lvm:1`** dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

3. **`fs:1`** — Ceci est une ressource de système de fichiers. S'il y avait d'autres ressources de systèmes de fichiers dans Service *foo*, elles seraient démarrées dans l'ordre définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

4. **`ip:10.1.1.1`** — Ceci est une ressource d'adresse IP. S'il y avait d'autres ressources d'adresses IP dans Service *foo*, elles seraient démarrées dans l'ordre définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

5. **`script:1`** — Ceci est une ressource de script. S'il y avait d'autres ressources de scripts dans Service *foo*, elles seraient démarrées dans l'ordre définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

Ordre d'arrêt des ressources enfants typées

Dans l'Exemple C.3, « Classement dans un type de ressource », les ressources sont arrêtées dans selon l'ordre suivant :

1. **`script:1`** — Ceci est une ressource Script. S'il y avait d'autres ressources Script dans Service *foo*, elles seraient arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

2. **`ip:10.1.1.1`** — Ceci est une ressource adresse IP. S'il y avait d'autres ressources adresse IP dans Service *foo*, elles seraient arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

3. **`fs:1`** — Ceci est une ressource Système de fichiers. S'il y avait d'autres ressources Système de fichiers dans Service *foo*, elles seraient arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

4. **`lvm:2`** — Ceci est une ressource LVM. Toutes les ressources LVM sont arrêtées en dernier. **`lvm:2`** (**`<lvm name="2" .../>`**) est arrêté avant **`lvm:1`** ; les ressources à l'intérieur du groupe d'un type de ressources sont arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

5. **`lvm:1`** — Ceci est une ressource LVM. Toutes les ressources LVM sont arrêtées en dernier. **`lvm:1`** (**`<lvm name="1" .../>`**) est arrêté après **`lvm:2`** ; les ressources à l'intérieur du groupe d'un type de ressources sont arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

C.2.2. Ordre de démarrage et d'arrêt de ressources enfant non-typées

Des considérations supplémentaires sont requises pour les ressources enfants sans type. Pour une ressource enfant sans type, l'ordre de démarrage et d'arrêt ne sont pas explicitement spécifiés par la ressource Service. Au lieu de cela, l'ordre de démarrage et d'arrêt sont déterminés en fonction de l'ordre de la ressources enfant dans **`/etc/cluster/cluster.conf`**. En outre, les ressources enfant sans type sont démarrées après toutes les ressources enfants avec type et arrêtées avant toute ressource enfant avec type.

Par exemple, prenez en considération l'ordre de démarrage et d'arrêt des ressources enfant non-typées dans l'Exemple C.4, « Ressources enfant typées et non-typées dans un service ».

Exemple C.4. Ressources enfant typées et non-typées dans un service

```
<service name="foo">
  <script name="1" .../>
  <nontypedresource name="foo"/>
  <lvm name="1" .../>
  <nontypedresourcetwo name="bar"/>
  <ip address="10.1.1.1" .../>
```

```
    <fs name="1" .../>
    <lvm name="2" .../>
  </service>
```

Ordre de démarrage de ressources enfant non-typées

Dans l'Exemple C.4, « Ressources enfant typées et non-typées dans un service », les ressources enfant sont démarrées dans l'ordre suivant :

1. **`lvm:1`** — Ceci est une ressource LVM, Toutes les ressources LVM sont démarrées en premier. **`lvm:1`** (**`<lvm name="1" .../>`**) est la première ressource LVM démarrée dar il s'agit de la première ressource LVM répertoriée dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

2. **`lvm:2`** — Ceci est une ressource LVM. Toutes les ressources LVM sont démarrées en premier. **`lvm:2`** (**`<lvm name="2" .../>`**) est démarré après **`lvm:1`** car **`lvm:2`** est répertorié après **`lvm:1`** dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

3. **`fs:1`** — Ceci est une ressource de système de fichiers. S'il y avait d'autres ressources de systèmes de fichiers dans Service *foo*, elles seraient démarrées dans l'ordre définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

4. **`ip:10.1.1.1`** — Ceci est une ressource d'adresse IP. S'il y avait d'autres ressources d'adresses IP dans Service *foo*, elles seraient démarrées dans l'ordre définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

5. **`script:1`** — Ceci est une ressource de script. S'il y avait d'autres ressources de scripts dans Service *foo*, elles seraient démarrées dans l'ordre définit dans la portion Service *foo* de **`/etc/cluster/cluster.conf`**.

6. **`nontypedresource:foo`** — Ressource non-typée. Comme il s'agit d'une ressource non-typée, celle-ci est lancée après le démarrage des ressources typées. En outre, son ordre dans la ressource Service est avant les autres ressources non-typées **`nontypedresourcetwo:bar`** ; elle est ainsi lancée avant **`nontypedresourcetwo:bar`**. (Les ressources non-typées sont lancées dans l'ordre dans lequel elles apparaissent dans la ressource Service.)

7. **`nontypedresourcetwo:bar`** — Ressource non-typée. Comme il s'agit d'une ressource non-typée, celle-ci est lancée après le démarrage des ressources typées. En outre, son ordre dans la ressource Service est après les autres ressources non-typées **`nontypedresource:foo`** ; elle est ainsi lancée après **`nontypedresource:foo`**. (Les ressources non-typées sont lancées dans l'ordre dans lequel elles apparaissent dans la ressource Service.)

Ordre d'arrêt des ressources enfant non-typées

Dans l'Exemple C.4, « Ressources enfant typées et non-typées dans un service », les ressources enfant sont arrêtées dans l'ordre suivant :

1. **`nontypedresourcetwo:bar`** — Ressource non-typée. Comme il s'agit d'une ressource non-typée, celle-ci est arrêtée avant l'arrêt des ressources typées. En outre, son ordre dans la ressource Service est après les autres ressources non-typées **`nontypedresource:foo`** ; elle

est ainsi arrêtée avant **nontypedresource:foo**. (Les ressources non-typées sont arrêtées dans l'ordre inverse par rapport à l'ordre dans lequel elles apparaissent dans la ressource Service.)

2. **nontypedresource:foo** — Ceci est une ressource non-typée. Comme il s'agit d'une ressource non-typée, celle-ci est arrêtée avant l'arrêt des ressources typées. En outre, son ordre dans la ressource Service est avant les autres ressources non-typées **nontypedresourcetwo:bar** ; elle est ainsi arrêtée après **nontypedresourcetwo:bar**. (Les ressources non-typées sont arrêtées l'ordre inverse par rapport à l'ordre dans lequel elles apparaissent dans la ressource Service.)

3. **script:1** — Ceci est une ressource Script. S'il y avait d'autres ressources Script dans Service *foo*, elles seraient arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **/etc/cluster/cluster.conf**.

4. **ip:10.1.1.1** — Ceci est une ressource adresse IP. S'il y avait d'autres ressources adresse IP dans Service *foo*, elles seraient arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **/etc/cluster/cluster.conf**.

5. **fs:1** — Ceci est une ressource Système de fichiers. S'il y avait d'autres ressources Système de fichiers dans Service *foo*, elles seraient arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **/etc/cluster/cluster.conf**.

6. **lvm:2** — Ceci est une ressource LVM. Toutes les ressources LVM sont arrêtées en dernier. **lvm:2** (**<lvm name="2" .../>**) est arrêté avant **lvm:1** ; les ressources à l'intérieur du groupe d'un type de ressources sont arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **/etc/cluster/cluster.conf**.

7. **lvm:1** — Ceci est une ressource LVM. Toutes les ressources LVM sont arrêtées en dernier. **lvm:1** (**<lvm name="1" .../>**) est arrêté après **lvm:2** ; les ressources à l'intérieur du groupe d'un type de ressources sont arrêtées dans l'ordre inverse définit dans la portion Service *foo* de **/etc/cluster/cluster.conf**.

C.3. HÉRITAGE, LE BLOC <RESSOURCES>, ET LA RÉUTILISATION DES RESSOURCES

Certaines ressources bénéficient de l'héritage des valeurs depuis une ressource parente. Ceci est communément le case pour un service NFS. L'Exemple C.5, « Paramétrage du service NFS pour une réutilisation des ressources et un héritage » montre une configuration de service NFS typique, paramétrée pour une réutilisation des ressources et un héritage.

Exemple C.5. Paramétrage du service NFS pour une réutilisation des ressources et un héritage

```
    <resources>
        <nfsclient name="bob" target="bob.example.com"
options="rw,no_root_squash"/>
        <nfsclient name="jim" target="jim.example.com"
options="rw,no_root_squash"/>
        <nfsexport name="exports"/>
    </resources>
    <service name="foo">
        <fs name="1" mountpoint="/mnt/foo" device="/dev/sdb1"
```

```
fsid="12344">
            <nfsexport ref="exports">  <!-- nfsexport's path and fsid
attributes
                                                are inherited from the
mountpoint &
                                                fsid attribute of the
parent fs
                                                resource -->
                <nfsclient ref="bob"/> <!-- nfsclient's path is
inherited from the
                                                mountpoint and the fsid
is added to the
                                                options string during
export -->
                <nfsclient ref="jim"/>
            </nfsexport>
        </fs>
        <fs name="2" mountpoint="/mnt/bar" device="/dev/sdb2"
fsid="12345">
            <nfsexport ref="exports">
                <nfsclient ref="bob"/> <!-- Because all of the critical
data for this
                                                resource is either
defined in the
                                                resources block or
inherited, we can
                                                reference it again! -->
                <nfsclient ref="jim"/>
            </nfsexport>
        </fs>
        <ip address="10.2.13.20"/>
    </service>
```

Si le service était plat (c'est-à-dire sans relations de type parent/enfant), il devrait alors être configuré comme suit :

- Le service nécessiterait quatre ressources nfsclient — une par fichier (soit un total de deux pour les systèmes de fichiers), et une par machine-cible (soit un total de deux pour les machines-cibles).
- Le service devrait spécifier le chemin d'exportation et l'ID du système de fichiers à chaque nfsclient, ce qui induit la possibilité d'erreurs dans la configuration.

Cependant, dans l'Exemple C.5, « Paramétrage du service NFS pour une réutilisation des ressources et un héritage » , les ressources client NFS *nfsclient:bob* et *nfsclient:jim* ne sont définies qu'une seule fois. De même, la ressource d'exportation NFS *nfsexport:exports* n'est définie qu'une seule fois. Tous les attributs nécessités par les ressources sont hérités de ressources parentes. Comme les attributs hérités sont dynamiques (et ne rentrent pas en conflit les uns avec les autres), il est possible de réutiliser ces ressources — c'est pourquoi ils sont définit dans le bloc des ressources, même si ce n'est pas pratique pour la configuration de certaines ressources se trouvant dans de multiples emplacements. Par exemple, la configuration d'une ressource de système de fichiers dans de multiples emplacements peut résulter en le montage d'un système de fichiers sur deux nœuds, et ainsi provoquer des problèmes.

C.4. RÉCUPÉRATION DE DÉFAILLANCE ET SOUS-ARBRES INDÉPENDANTS

Dans la plupart des environnements d'entreprise, le déroulement habituel de la récupération d'un service après une défaillance est d'effectuer un redémarrage complet du service si l'un des composants du service échoue. Par exemple, dans Exemple C.6, « Récupération normale après défaillance du service *foo* », si l'un des scripts défini dans ce service échoue, le déroulement habituel est de redémarrer (ou de transférer ou désactiver, selon la politique de restauration du service) le service. Cependant, sous certaines circonstances, des parties du services peuvent être considérées comme non-critiques ; il peut se révéler nécessaire de ne redémarrer qu'une partie du service en place avant de tenter une récupération normale. Pour effectuer ceci, vous pouvez utiliser l'attribut ***__independent_subtree***. Par exemple, dans l'Exemple C.7, « Récupération après défaillance du service *foo* avec l'attribut ***__independent_subtree*** », l'attribut ***__independent_subtree*** est utilisé afin d'effectuer les actions suivantes :

- Si script:script_one échoue, redémarrez script:script_one, script:script_two, et script:script_three.
- Si script:script_two échoue, redémarrez script:script_two uniquement.
- Si script:script_three échoue, redémarrez restart script:script_one, script:script_two, et script:script_three.
- Si script:script_four échoue, redémarrez la totalité du service.

Exemple C.6. Récupération normale après défaillance du service *foo*

```
<service name="foo">
      <script name="script_one" ...>
          <script name="script_two" .../>
      </script>
      <script name="script_three" .../>
</service>
```

Exemple C.7. Récupération après défaillance du service *foo* avec l'attribut *__independent_subtree*

```
<service name="foo">
      <script name="script_one" __independent_subtree="1" ...>
          <script name="script_two" __independent_subtree="1" .../>
          <script name="script_three" .../>
      </script>
      <script name="script_four" .../>
</service>
```

Dans certaines circonstances, si le composant d'un service échoue, vous devriez désactiver ce composant uniquement, sans désactiver le service entier afin d'éviter que d'autres services utilisant d'autres composants de ce service soient affectés. À partir de Red Hat Enterprise Linux 6.1, ceci peut être accompli en utilisant l'attribut ***__independent_subtree="2"***, qui désigne le sous-arbre indépendant comme étant non-critique.

NOTE

Vous pouvez utiliser l'indicateur non-critique sur les ressources à référence unique uniquement. L'indicateur non-critique fonctionne avec toutes les ressources à tous les niveaux de l'arborescence des ressources, mais ne devrait pas être utilisé au niveau le plus haut lors de la définition des services ou des machines virtuelles.

À partir de la version 6.1 de Red Hat Enterprise Linux, vous pouvez définir maximum restart et les expirations restart sur une base par nœud dans l'arborescence des ressources des sous-arbres indépendants. Pour définir ces limites, vous pouvez utiliser les attributs suivants :

- **`__max_restarts`** configure le nombre maximum de redémarrages tolérés avant d'abandonner.
- **`__restart_expire_time`** configure le temps, en secondes, à partir duquel un redémarrage n'est plus tenté.

C.5. DÉBOGAGE ET TESTAGE DES SERVICES ET DE L'ORDRE DES RESSOURCES

Vous pouvez déboguer et tester l'ordre des services et des ressources avec l'utilitaire **`rg_test`**. **`rg_test`** est un utilitaire en ligne de commande fourni par le paquetage **`rgmanager`** qui est exécuté depuis un shell ou un terminal (il n'est pas disponible sous **Conga**). Le Tableau C.2, « Résumé de l'utilitaire **`rg_test`** » résume les actions et la syntaxe de l'utilitaire **`rg_test`**.

Tableau C.2. Résumé de l'utilitaire `rg_test`

Action	Syntaxe
Afficher les règles des ressources que **`rg_test`** comprend.	**`rg_test rules`**
Tester une configuration (et /usr/share/cluster) pour des erreurs ou des agents de ressources redondants.	**`rg_test test /etc/cluster/cluster.conf`**
Afficher l'ordre de démarrage et d'arrêt d'un service.	Afficher l'ordre de démarrage : **`rg_test noop /etc/cluster/cluster.conf start service servicename`** Afficher l'ordre d'arrêt : **`rg_test noop /etc/cluster/cluster.conf stop service servicename`**

Action	Syntaxe
Démarrer ou arrêter un service de manière explicite.	 **IMPORTANT** Effectuez cela sur un seul nœud, et désactivez le service dans rgmanager en premier à chaque fois. Démarrer un service : **rg_test test /etc/cluster/cluster.conf start service *servicename*** Arrêter un service : **rg_test test /etc/cluster/cluster.conf stop service *servicename***
Calculer et afficher le delta de l'arborescence des ressources entre deux fichiers cluster.conf.	**rg_test delta *cluster.conf file 1* *cluster.conf file 2*** Par exemple : **rg_test delta /etc/cluster/cluster.conf.bak /etc/cluster/cluster.conf**

ANNEXE D. VÉRIFICATION DES RESSOURCES DE SERVICE DE CLUSTER ET DÉLAI DE BASCULEMENT

Cet annexe décrit comment **rgmanager** surveille le statut des ressources de cluster et comment modifier l'intervalle de vérification de statut. L'annexe décrit aussi le paramètre du service **__enforce_timeouts**, qui indique si un délai pour une opération provoque l'échec d'un service.

NOTE

Pour mieux comprendre les informations présentes dans cet annexe, vous aurez besoin d'une bonne compréhension des agents de ressources et du fichier de configuration du cluster, **/etc/cluster/cluster.conf**. Pour obtenir la liste et la description complète des éléments et attributs **cluster.conf**, reportez-vous au schéma des clusters sur **/usr/share/cluster/cluster.rng**, et au schéma annoté sur **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (par exemple, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

D.1. MODIFIER L'INTERVALLE DE VÉRIFICATION DU STATUT DES RESSOURCES

rgmanager vérifie le statut des ressources individuelles, pas des services entiers. Toutes les 10 secondes, rgmanager scanne l'arborescence des ressources, cherchant des ressources ayant passé leur intervalle de « vérification de statut ».

Chaque agent de ressources spécifie l'intervalle de temps entre les vérifications périodiques de statut. Chaque ressource utilise ces valeurs d'intervalles sauf si explicitement supprimées dans le fichier **cluster.conf** avec la balise spéciale **<action>** :\t

```
<cman two_node="1" expected_votes="1"/>
```

Cette balise est un enfant spécial de la ressource elle-même dans le fichier **cluster.conf**. Par exemple, si vous possédez une ressource de système de fichier sur laquelle vous souhaiteriez supprimer l'intervalle de vérification de statut, vous pouvez spécifier la ressource du système de fichier dans le fichier **cluster.conf** comme suit :

```
<fs name="test" device="/dev/sdb3">
  <action name="status" depth="*" interval="10" />
  <nfsexport...>
  </nfsexport>
</fs>
```

Certains agents fournissent de multiples « profondeurs » de vérification. Par exemple, une vérification de statut de système de fichiers normal (profondeur 0) vérifie si le système de fichiers est monté sur le bon emplacement. Une vérification plus intensive aura une profondeur de 10, et vérifiera si vous pouvez lire un fichier depuis le système de fichiers. Une vérification de profondeur 20 vérifiera si vous pouvez écrire sur le système de fichiers. Dans l'exemple donné ici, la **profondeur** est définie sur *****, indiquant ainsi que ces valeurs devraient être utilisées pour toutes les profondeurs. Le résultat est que le système de fichiers **test** est vérifié à la plus grande profondeur offerte par l'agent de ressources (dans ce cas, 20) toutes les 10 secondes.

D.2. APPLIQUER LES DÉLAIS DES RESSOURCES

Il n'y a pas de délai pour démarrer, arrêter, ou faire basculer des ressources. Certaines ressources prennent un temps de durée indéterminée pour démarrer ou pour s'arrêter. Malheureusement, l'échec d'un arrêt (y compris un délai) rend le service inopérable (état d'échec). Vous pouvez, si vous le souhaitez, activer l'application de délais sur chaque ressource dans un service de manière individuelle en ajoutant **__enforce_timeouts="1"** à la référence dans le fichier **cluster.conf**.

L'exemple suivant affiche un service de cluster ayant été configuré avec l'attribut **__enforce_timeouts** définit pour la ressource **netfs**. Avec cet attribut défini, si plus de 30 secondes sont nécessaires pour démonter le système de fichiers NFS pendant un processus de récupération, l'opération expirera, mettant par la même occasion le service en état d'échec.

```
</screen>
<rm>
  <failoverdomains/>
  <resources>
    <netfs export="/nfstest" force_unmount="1" fstype="nfs"
host="10.65.48.65"
           mountpoint="/data/nfstest" name="nfstest_data"
options="rw,sync,soft"/>
  </resources>
  <service autostart="1" exclusive="0" name="nfs_client_test"
recovery="relocate">
    <netfs ref="nfstest_data" __enforce_timeouts="1"/>
  </service>
</rm>
```

ANNEXE E. RÉSUMÉ DES OUTILS DE LA LIGNE DE COMMANDE

Tableau E.1, « Résumé des outils de la ligne de commande » résume les outils en ligne de commande préférés pour la configuration et la gestion du composant additionnel High Availability. Pour obtenir plus d'informations sur les commandes et les variables, reportez-vous à la page man de chaque outil de ligne de commande.

Tableau E.1. Résumé des outils de la ligne de commande

Outil de la ligne de commande	Utilisé avec	But
`ccs_config_dump` — Outil de vidage de configuration de cluster	Infrastructure du cluster	**`ccs_config_dump`** génère une sortie XML de la configuration en cours d'exécution. La configuration en cours d'exécution est parfois différente de la configuration stockée sur fichier car certains sous-systèmes stockent ou paramètrent des informations par défaut dans la configuration. Ces valeurs ne sont généralement pas présentes sur la version sur disque de la configuration, mais sont requises lors de l'exécution pour que le cluster puisse fonctionner correctement. Pour plus d'informations sur cet outil, reportez-vous à la page man ccs_config_dump(8).
`ccs_config_validate` — Outil de validation de la configuration du cluster	Infrastructure du cluster	**`ccs_config_validate`** valide **`cluster.conf`** sur le schéma, **`cluster.rng`** (qui se trouve dans **`/usr/share/cluster/cluster.rng`**) sur chaque nœud. Pour plus d'informations sur cet outil, reportez-vous à la page man ccs_config_validate(8).
`clustat` — Utilitaire de statut du cluster	Composants de gestion du service High-availability	La commande **`clustat`** affiche le statut du cluster. Elle affiche les informations d'abonnement, une vue du quorum, ainsi que l'état de tous les services utilisateur configurés. Pour plus d'informations sur cet outil, reportez-vous à la page man clustat(8).
`clusvcadm` — Utilitaire d'administration du service utilisateur du cluster	Composants de gestion du service High-availability	La commande **`clusvcadm`** vous permet d'activer, de désactiver, de transférer, et de redémarrer les services high-availability dans un cluster. Pour plus d'informations sur cet outil, reportez-vous à la page man clusvcadm(8).

Outil de la ligne de commande	Utilisé avec	But
`cman_tool` — Outil de gestion du cluster	Infrastructure du cluster	**`cman_tool`** est un programme qui gère le gestionnaire de clusters CMAN. Il offre la possibilité de rejoindre un cluster, de le quitter, de tuer (kill) un nœud, ou de changer le nombre de votes d'un nœud pour atteindre le quorum dans un cluster. Pour plus d'informations sur cet outil, reportez-vous à la page man cman_tool(8).
`fence_tool` — Outil Fence	Infrastructure du cluster	**`fence_tool`** est un programme utilisé pour rejoindre et quitter le domaine Fence. Pour plus d'informations sur cet outil, reportez-vous à la page man fence_tool(8).

ANNEXE F. LVM HAUTE DISPONIBILITÉ (HA-LVM)

Le module complémentaire Red Hat High Availability fournit la prise en charge haute disponibilité des volumes LVM (HA-LVM) dans une configuration de basculement. Celle-ci est différente des configurations active/active qui sont activées par CLVM (gestionnaire de volumes logiques clusterisés), qui est un ensemble d'extensions mises en cluster de LVM permettant à un cluster d'ordinateurs de gérer leur stockage partagé.

L'utilisation de CLVM ou de HA-LVM doit être basée sur les besoins des applications ou services déployés.

- Si les applications sont conscientes de l'existence du cluster et ont été paramétrées pour être exécutées simultanément sur de multiples machines à la fois, alors CLVM devrait être utilisé. Plus particulièrement, si plus d'un nœud de votre cluster requiert accès à votre stockage, qui sera ensuite partagé à travers les différents nœuds actifs, alors vous devrez utiliser CLVM. CLVM permet à un utilisateur de configurer des volumes logiques sur un stockage partagé en verrouillant l'accès au stockage physique pendant qu'un volume est en cours de configuration et utilise les services de verrouillage clusterisés pour gérer le stockage partagé. Pour obtenir des informations sur CLVM et sur la configuration LVM en général, reportez-vous au document *Administration LVM*.

- Si les applications fonctionnent de manière optimale dans des configurations actives/passives (basculement) où seul un nœud unique accédant au stockage est actif à la fois, vous devriez utiliser des agents LVM de haute disponibilité (HA-LVM).

La plupart des applications fonctionneront mieux sous une configuration active/passive car elles ne sont pas conçues ou optimisées pour être exécutées simultanément avec d'autres instances. Choisir d'exécuter une application qui n'est pas consciente des clusters sur des volumes logiques clusterisés peut provoquer une dégradation de la performance si le volume logique est mis en miroir. Ceci est dû au fait qu'il y a une surcharge de communications du ou des cluster(s) pour les volumes logiques dans ces instances. Une application consciente du ou des cluster(s) doit être en mesure de réaliser des améliorations de la performance au-delà des pertes de performance offertes par les systèmes de fichiers du ou des cluster(s) et des volumes logiques reconnaissant le(s) cluster(s). Ceci est plus facilement faisable pour certaines applications et charges de travail que pour d'autres. Déterminer quels sont les pré-requis du cluster et si l'effort supplémentaire pour optimiser un cluster dans une configuration active/active offrira des dividendes est la meilleure manière de choisir entre deux variantes de LVM. La plupart des utilisateurs obtiendront les meilleurs résultats de haute disponibilité en utilisant HA-LVM.

HA-LVM et CLVM sont similaires dans le fait qu'ils empêchent la corruption des métadonnées LVM et de ses volumes logiques, qui pourraient autrement se produire si de multiples machines étaient autorisées à effectuer des changements superposés. HA-LVM impose une restriction faisant qu'un volume logique peut uniquement être activé de manière exclusive ; c'est-à-dire qu'il ne peut être actif que sur une seule machine à la fois. Ceci signifie que seules des implémentations locales (non-clusterisées) de pilotes de stockages sont utilisées. Éviter une surcharge de coordination de cluster de cette manière permet d'améliorer la performance. CLVM n'impose pas de telles restrictions - l'utilisateur est libre d'activer un volume logique sur toutes les machines d'un cluster, ce qui pousse à utiliser des pilotes de stockage reconnaissant le cluster, permettant ainsi d'installer des systèmes de fichiers et des applications reconnaissant le cluster au-dessus de celui-ci.

HA-LVM peut être paramétré afin d'utiliser l'une de deux méthodes pour réaliser son mandat d'activation de volume logique de manière exclusive.

- La méthode préférée utilise CLVM, mais celle-ci active uniquement les volumes logiques de manière exclusive. Cela présente les avantages d'une installation plus facile et permet une meilleure prévention des erreurs administratives (comme la suppression d'un volume logique en

cours d'utilisation). Pour utiliser CLVM, les logiciels des modules complémentaires High Availability et Resilient Storage, y compris le démon **clvmd**, doivent être en cours d'exécution.

La procédure pour configurer HA-LVM à l'aide de cette méthode est décrite dans la Section F.1, « Configurer le basculement HA-LVM avec CLVM (méthode préférée) ».

- La seconde méthode utilise le verrouillage de machine locale et des « balises » LVM. Cette méthode présente l'avantage de ne pas nécessiter de paquetages de cluster LVM ; elle requiert cependant des étapes supplémentaires lors de son installation et n'empêchera pas un administrateur de supprimer par erreur un volume logique d'un nœud du cluster lorsqu'il n'est pas actif. La procédure pour configurer HA-LVM à l'aide de cette méthode est décrite dans la Section F.2, « Configurer le basculement HA-LVM avec le Tagging (étiquetage) ».

F.1. CONFIGURER LE BASCULEMENT HA-LVM AVEC CLVM (MÉTHODE PRÉFÉRÉE)

Pour définir la basculement HA-LVM (à l'aide de la variante CLVM préférée), veuillez procéder aux étapes suivantes :

1. Assurez-vous que votre système est configuré pour prendre en charge CLVM, ce qui requiert :
   - Les modules complémentaires High Availability et Resilient Storage installés, y compris le paquetage **cmirror** si les volumes logiques CLVM doivent être mis en miroir.
   - Le paramètre **locking_type** dans la section globale du fichier **/etc/lvm/lvm.conf** doit être défini sur la valeur « 3 ».
   - Les logiciels des modules complémentaires High Availability et Resilient Storage, y compris le démon **clvmd**, doivent être en cours d'exécution. Pour la mise en miroir CLVM, le service **cmirrord** doit aussi être lancé.
2. Créez le volume logique et le système de fichiers à l'aide des commandes standard de LVM et des systèmes de fichiers, comme dans l'exemple suivant.

```
# pvcreate /dev/sd[cde]1

# vgcreate -cy shared_vg /dev/sd[cde]1

# lvcreate -L 10G -n ha_lv shared_vg

# mkfs.ext4 /dev/shared_vg/ha_lv

# lvchange -an shared_vg/ha_lv
```

   Pour obtenir des informations sur la création de volumes logiques LVM, reportez-vous au document *Administration LVM*.
3. Modifiez le fichier **/etc/cluster/cluster.conf** afin d'inclure le nouveau volume logique créé en tant que ressource dans l'un de vos services. Alternativement, vous pouvez utiliser **Conga** ou la commande **ccs** pour configurer LVM et les ressources du système de fichiers du cluster. Ci-dessous figure une section exemple du gestionnaire de ressources du fichier **/etc/cluster/cluster.conf**, qui configure un volume logique CLVM en tant que ressource de cluster :

```
<rm>
   <failoverdomains>
       <failoverdomain name="FD" ordered="1" restricted="0">
          <failoverdomainnode name="neo-01" priority="1"/>
          <failoverdomainnode name="neo-02" priority="2"/>
       </failoverdomain>
   </failoverdomains>
   <resources>
       <lvm name="lvm" vg_name="shared_vg" lv_name="ha-lv"/>
       <fs name="FS" device="/dev/shared_vg/ha-lv" force_fsck="0"
force_unmount="1" fsid="64050" fstype="ext4" mountpoint="/mnt"
options="" self_fence="0"/>
   </resources>
   <service autostart="1" domain="FD" name="serv"
recovery="relocate">
       <lvm ref="lvm"/>
       <fs ref="FS"/>
   </service>
</rm>
```

F.2. CONFIGURER LE BASCULEMENT HA-LVM AVEC LE TAGGING (ÉTIQUETAGE)

Pour configurer le basculement HA-LVM en utilisant des balises dans le fichier **/etc/lvm/lvm.conf**, veuillez procéder aux étapes suivantes :

1. Assurez-vous que le paramètre **locking_type** dans la section globale du fichier **/etc/lvm/lvm.conf** est bien défini sur la valeur « 1 ».

2. Créez le volume logique et le système de fichiers à l'aide des commandes standard de LVM et des systèmes de fichiers, comme dans l'exemple suivant.

   ```
   # pvcreate /dev/sd[cde]1

   # vgcreate shared_vg /dev/sd[cde]1

   # lvcreate -L 10G -n ha_lv shared_vg

   # mkfs.ext4 /dev/shared_vg/ha_lv
   ```

   Pour obtenir des informations sur la création de volumes logiques LVM, reportez-vous au document *Administration LVM*.

3. Modifiez le fichier **/etc/cluster/cluster.conf** afin d'inclure le nouveau volume logique créé en tant que ressource dans l'un de vos services. Alternativement, vous pouvez utiliser **Conga** ou la commande **ccs** pour configurer LVM et les ressources du système de fichiers du cluster. Ci-dessous figure une section exemple du gestionnaire de ressources du fichier **/etc/cluster/cluster.conf**, qui configure un volume logique CLVM en tant que ressource de cluster :

   ```
   <rm>
      <failoverdomains>
   ```

```
        <failoverdomain name="FD" ordered="1" restricted="0">
           <failoverdomainnode name="neo-01" priority="1"/>
           <failoverdomainnode name="neo-02" priority="2"/>
        </failoverdomain>
    </failoverdomains>
    <resources>
        <lvm name="lvm" vg_name="shared_vg" lv_name="ha_lv"/>
        <fs name="FS" device="/dev/shared_vg/ha_lv" force_fsck="0"
force_unmount="1" fsid="64050" fstype="ext4" mountpoint="/mnt"
options="" self_fence="0"/>
    </resources>
    <service autostart="1" domain="FD" name="serv"
recovery="relocate">
        <lvm ref="lvm"/>
        <fs ref="FS"/>
    </service>
</rm>
```

NOTE

Si de multiples volumes logiques se trouvent dans le groupe de volumes, alors le nom du volume logique (**lv_name**) dans la ressource **lvm** doit être laissé vide ou non-spécifié. Veuillez aussi remarquer que dans une configuration HA-LVM, un groupe de volumes peut uniquement être utilisé par un seul service.

4. Modifiez le champ **volume_list** dans le fichier **/etc/lvm/lvm.conf**. Veuillez inclure le nom de votre groupe de volumes root et votre nom d'hôte comme répertorié dans le fichier **/etc/cluster/cluster.conf** et précédé du caractère « @ ». Le nom d'hôte à inclure ici est la machine sur laquelle vous modifiez le fichier **lvm.conf**, et non un nom d'hôte distant. Remarquez que cette chaîne *DOIT* correspondre au nom du nœud spécifié dans le fichier **cluster.conf**. Ci-dessous figure un exemple d'entrée du fichier **/etc/lvm/lvm.conf** :

```
volume_list = [ "VolGroup00", "@neo-01" ]
```

Cette balise sera utilisée pour activer les VG (groupes de volumes) ou LV (volumes logiques) partagés. *N'INCLUEZ PAS* les noms des groupes de volumes devant être partagés à l'aide de HA-LVM.

5. Mettez à jour le périphérique **initrd** sur tous les nœuds de votre cluster :

```
# dracut -H -f /boot/initramfs-$(uname -r).img $(uname -r)
```

6. Redémarrez tous les nœuds afin de vous assurer que le périphérique **initrd** correct est en cours d'utilisation.

ANNEXE G. HISTORIQUE DES VERSIONS

Version 5.0-25.2.400 **2013-10-31** **Rüdiger Landmann**

Rebuild with publican 4.0.0

Version 5.0-25.2 **Wed May 1 2013** **Sam Friedmann**

Fichiers de traduction synchronisés avec les sources XML 5.0-25

Version 5.0-25.1 **Thu Apr 18 2013** **Chester Cheng**

Fichiers de traduction synchronisés avec les sources XML 5.0-25

Version 5.0-25 **Mon Feb 18 2013** **Steven Levine**

Version pour la distribution GA 6.4

Version 5.0-23 **Wed Jan 30 2013** **Steven Levine**

Résout : 901641

Corrige et clarifie les règles iptables.

Version 5.0-22 **Tue Jan 29 2013** **Steven Levine**

Résout : 788636

Documente la configuration RRP à travers la commande `ccs`.

Résout : 789010

Documente la configuration RRP dans le fichier `cluster.conf`.

Version 5.0-20 **Fri Jan 18 2013** **Steven Levine**

Résout : 894097

Supprime les conseils pour vous assurer que vous n'utilisez pas le balisage VLAN.

Résout : 845365

Indique que les modes de liaisons 0 et 2 sont maintenant pris en charge.

Version 5.0-19 **Thu Jan 17 2013** **Steven Levine**

Résout : 896234

Clarifie la terminologie des références aux nœuds de clusters.

Version 5.0-16 **Mon Nov 26 2012** **Steven Levine**

Version pour la distribution 6.4 Bêta

Version 5.0-15 **Wed Nov 20 2012** **Steven Levine**

Résout : 838988
Documente l'attribut nfsrestart pour les agents de ressources de systèmes de fichiers.

Résout : 843169
Documente l'agent fence IBM iPDU.

Résout : 846121
Documente l'agent fence du contrôleur d'alimentation réseau Eaton (Interface SNMP).

Résout : 856834
Documente l'agent fence HP Bladesystem.

Résout : 865313
Documente l'agent de ressources du serveur NFS.

Résout : 862281
Clarifie quelles commandes **`ccs`** remplacent les paramètres précédents.

Résout : 846205
Documente le filtrage du composant **`igmp`** du pare-feu **`iptables`**.

Résout : 857172
Documente la capacité de supprimer des utilisateurs de luci.

Résout : 857165
Documente le niveau de privilèges de l'agent fence IPMI.

Résout : 840912
Clarifie le problème du formatage avec le tableau des paramètres de ressources.

Résout : 849240, 870292
Clarifie la procédure d'installation.

Résout : 871165
Clarifie la description du paramètres d'adresse IP dans la description de l'agent de ressources de l'adresse IP.

Résout : 845333, 869039, 856681
Corrige des erreurs de typographie mineures et clarifie des ambiguïtés techniques mineures.

Version 5.0-12 **Thu Nov 1 2012** **Steven Levine**
Ajout d'agents fence maintenant pris en charge.

Version 5.0-7 **Thu Oct 25 2012** **Steven Levine**
Ajout d'une section sur les sémantiques de remplacement.

Version 5.0-6 **Tue Oct 23 2012** **Steven Levine**
Correction de la valeur par défaut de « Post Join Delay ».

Version 5.0-4 **Tue Oct 16 2012** **Steven Levine**
Ajout de la description de la ressource du serveur NFS.

Version 5.0-2 **Thu Oct 11 2012** **Steven Levine**
Mises à jour des descriptions de Conga.

Version 5.0-1 **Mon Oct 8 2012** **Steven Levine**
Clarifications des sémantiques de ccs

Version 4.0-5 **Fri Jun 15 2012** **Steven Levine**

Version pour la distribution GA 6.3

Version 4.0-4 **Tue Jun 12 2012** **Steven Levine**

Résout : 830148

Assure la consistance des exemples de numéros de port pour Luci.

Version 4.0-3 **Tue May 21 2012** **Steven Levine**

Résout : 696897

Ajoute des informations de paramètres cluster.conf aux tableaux de paramètres et de ressources de périphériques fence.

Résout : 811643

Ajoute la procédure pour restaurer une base de données **luci** sur une machine séparée.

Version 4.0-2 **Wed Apr 25 2012** **Steven Levine**

Résout : 815619

Supprime l'avertissement sur l'utilisation de UDP Unicast (monodiffusion UDP) avec les systèmes de fichiers GFS2.

Version 4.0-1 **Fri Mar 30 2012** **Steven Levine**

Résout : 771447, 800069, 800061

Mise à jour de la documentation de **luci** pour qu'elle soit consistante avec la version 6.3 de Red Hat Enterprise Linux.

Résout : 712393

Ajoute des informations sur la capture d'un cœur d'application pour RGManager.

Résout : 800074

Documente l'agent de ressources `condor`.

Résout : 757904

Documente la copie de sauvegarde et la restauration de la configuration de **luci**.

Résout : 772374

Ajotue une section sur la gestion des machines virtuelles dans un cluster.

Résout : 712378

Ajoute la documentation sur la configuration HA-LVM.

Résout : 712400

Documente les options de débogage.

Résout : 751156

Documente le nouveau paramètre `fence_ipmilan`.

Résout : 721373

Documente les changements de configuration qui nécessitent un redémarrage du cluster.

Version 3.0-5 **Thu Dec 1 2011** **Steven Levine**

Publication pour disponibilité générale de Red Hat Enterprise Linux 6.2

Résout : 755849

Corrige l'exemple du paramètre monitor_link.

Version 3.0-4 **Mon Nov 7 2011** **Steven Levine**

Résout : 749857

Ajoute la documentation pour le périphérique fence RHEV-M REST API.

Version 3.0-3 **Fri Oct 21 2011** **Steven Levine**

Résout : #747181, #747182, #747184, #747185, #747186, #747187, #747188, #747189, #747190, #747192
Corrige les erreurs typographiques et ambiguïtés trouvées pendant la révision QE de la documentation de Red Hat Enterprise Linux 6.2.

Version 3.0-2 **Fri Oct 7 2011** **Steven Levine**

Résout : #743757
Corrige la référence au mode de liaison pris en charge dans la section troubleshooting (résolution de problèmes).

Version 3.0-1 **Wed Sep 28 2011** **Steven Levine**

Révision initiale de Red Hat Enterprise Linux version 6.2 Beta

Résout : #739613
Documente la prise en charge des nouvelles options **`ccs`** pour afficher les périphériques fence et les services disponibles.

Résout : #707740
Documente les mises à jour de l'interface Conga et la prise en charge de la définition des permissions utilisateur pour administrer Conga.

Résout : #731856
Documente le support de la configuration de **luci** par le biais du fichier **`/etc/sysconfig/luci`**.

Résout : #736134
Documente la prise en charge du transport UDPU.

Résout : #736143
Documente la prise en charge de Samba clusterisé.

Résout : #617634
Documente comment configurer l'unique adresse IP sur laquelle **luci** est servi.

Résout : #713259
Documente la prise en charge de l'agent **`fence_vmware_soap`**.

Résout : #721009
Fournit un lien vers l'article « Support Essentials » (les essentiels du support)

Résout : #717006
Fournit des informations sur l'autorisation du trafic de multidiffusion via le pare-feu **`iptables`**.

Résout : #717008
Fournit des informations sur la vérification de statut de service cluster et sur le délai de basculement.

Résout : #711868
Clarifie la description d'autostart.

Résout : #728337
Documente la procédure pour ajouter des ressources **`vm`** avec la commande **`ccs`**.

Résout : #725315, #733011, #733074, #733689
Correction d'erreurs typographiques mineures.

Version 2.0-1 **Thu May 19 2011** **Steven Levine**

Révision initiale de Red Hat Enterprise Linux 6.1

Résout : #671250
Documente la prise en charge des interruptions SNMP.

Résout : #659753
Documente la commande **`ccs`**.

Résout : #665055
Met à jour la documentation Conga pour refléter l'affichage mis à jour et la prise en charge des fonctionnalités.

Résout : #680294
Documente le besoin de mot de passe d'accès pour l'agent **`ricci`**.

Résout : #687871
Ajoute un chapitre sur la résolution.

Résout : #673217
Correction d'erreurs typographiques.

Résout : #675805
Ajoute une référence sur le schéma de **`cluster.conf`** aux tableaux des paramètres des ressources HA.

Résout : #672697
Met à jour les tableaux des paramètres des périphériques fence pour inclure tous les périphériques de clôturage pris en charge.

Résout : #677994
Corrige les informations sur les paramètres de l'agent fence **`fence_ilo`**.

Résout : #629471
Ajoute une note technique sur le paramétrage d'une valeur de consensus dans un cluster à deux nœuds.

Résout : #579585
Met à jour une section sur la mise à niveau du logiciel du module complémentaire Red Hat High Availability.

Résout : #643216
Clarifie de petits problèmes à travers le document.

Résout : #643191
Fournit des améliorations et des corrections sur la documentation de **luci**.

Résout : #704539
Met à jour le tableau des paramètres des ressources Virtual Machine.

Version 1.0-1 **Wed Nov 10 2010** **Paul Kennedy**
Publication initiale de Red Hat Enterprise Linux 6

INDEX

A

D

F

G

I

Q

R

S

T

V