



Gérer des clusters BeeGFS

BeeGFS on NetApp with E-Series Storage

NetApp
January 27, 2026

Sommaire

Gérer des clusters BeeGFS	1
Présentation, concepts clés et terminologie	1
Présentation	1
Concepts clés	1
Terminologie commune	2
Quand utiliser Ansible contre l'outil pcs	2
Vérifiez l'état du cluster	3
Présentation	3
Présentation de la sortie de <code>pcs status</code>	3
Reconfigurer le cluster HA et BeeGFS	4
Présentation	4
Comment désactiver et activer la fonction de fencing	4
Mettez à jour les composants du cluster HA	5
Mise à niveau des services BeeGFS	5
Mise à jour vers BeeGFS v8	8
Mise à niveau des packages Pacemaker et Corosync dans un cluster haute disponibilité	19
Mettez à jour le micrologiciel de l'adaptateur de nœud de fichier	22
Mettez à niveau la baie de stockage E-Series	27
Entretien et maintenance	29
Services de basculement/rétablissement	29
Placer le cluster en mode maintenance	31
Arrêtez et démarrez le cluster	32
Remplacer les nœuds de fichiers	33
Développez ou réduisez le cluster	34
Résoudre les problèmes	36
Présentation	36
Guides de dépannage	36
Problèmes courants	40
Tâches courantes de dépannage	41

Gérer des clusters BeeGFS

Présentation, concepts clés et terminologie

Apprenez à gérer des clusters BeeGFS HA après leur déploiement.

Présentation

Cette section s'adresse aux administrateurs de cluster qui doivent gérer des clusters BeeGFS HA après leur déploiement. Même ceux qui connaissent les clusters haute disponibilité Linux doivent lire attentivement ce guide car il existe un certain nombre de différences dans la gestion du cluster, en particulier concernant la reconfiguration, grâce à l'utilisation d'Ansible.

Concepts clés

Certains de ces concepts sont présentés sur la "[termes et concepts](#)" page principale, mais il est utile de les réintroduire dans le contexte d'un cluster BeeGFS HA :

Cluster Node: Un serveur exécutant les services Pacemaker et Corosync et participant au cluster HA.

Nœud de fichiers : Nœud de cluster utilisé pour exécuter un ou plusieurs services de gestion, de métadonnées ou de stockage BeeGFS.

Nœud de bloc : Un système de stockage NetApp E-Series qui fournit un stockage bloc aux nœuds de fichiers. Ces nœuds ne participent pas au cluster BeeGFS HA car ils fournissent leurs propres capacités HA autonomes. Chaque nœud est constitué de deux contrôleurs de stockage qui assurent une haute disponibilité au niveau de la couche bloc.

Service BeeGFS: Un service de gestion, de métadonnées ou de stockage BeeGFS. Chaque nœud de fichiers exécute un ou plusieurs services qui utilisent les volumes du nœud de bloc pour stocker leurs données.

Building Block : Un déploiement standardisé de nœuds de fichiers BeeGFS, de nœuds de blocs E-Series et des services BeeGFS s'exécutent sur eux qui simplifient l'évolution d'un cluster/système de fichiers BeeGFS HA grâce à une architecture vérifiée NetApp. Les clusters haute disponibilité personnalisés sont également pris en charge, mais leur approche consiste souvent à adopter des éléments de base similaires pour simplifier l'évolutivité.

BeeGFS HA Cluster: Nombre évolutif de nœuds de fichiers utilisés pour exécuter les services BeeGFS sauvegardés par des nœuds de blocs pour stocker des données BeeGFS de façon hautement disponible. Repose sur des composants open source éprouvés Pacemaker et Corosync avec Ansible pour le packaging et le déploiement.

Cluster services: désigne les services Pacemaker et Corosync exécutés sur chaque nœud participant au cluster. Notez qu'un nœud n'exécute pas de services BeeGFS et qu'il participe uniquement au cluster comme un nœud « Tiebreaker » s'il n'y a que besoin de deux nœuds de fichiers.

Cluster Resources: pour chaque service BeeGFS s'exécutant dans le cluster, vous verrez une ressource de moniteur BeeGFS et un groupe de ressources contenant des ressources pour les cibles BeeGFS, les adresses IP (IP flottantes) et le service BeeGFS.

Ansible: Un outil de provisionnement logiciel, de gestion de la configuration et de déploiement des applications, permettant ainsi une infrastructure comme du code. Tout cela est possible grâce au package de clusters BeeGFS HA pour simplifier le processus de déploiement, de reconfiguration et de mise à jour de

BeeGFS sur NetApp.

Pcs: Une interface de ligne de commande disponible à partir de n'importe quel nœud de fichiers du cluster utilisé pour interroger et contrôler l'état des nœuds et des ressources du cluster.

Terminologie commune

Basculement: chaque service BeeGFS a un noeud de fichier préféré qu'il fonctionne à moins que ce noeud ne tombe en panne. Lorsqu'un service BeeGFS s'exécute sur un nœud de fichier non préféré/secondaire, il est dit qu'il est en cours de basculement.

Retour arrière: le fait de déplacer les services BeeGFS d'un noeud de fichier non préféré vers leur noeud préféré.

Paire HA : deux nœuds de fichiers qui accèdent au même ensemble de nœuds de bloc sont parfois appelés paire HA. Ce terme est utilisé dans l'ensemble de NetApp pour désigner deux contrôleurs ou nœuds de stockage qui peuvent « prendre le relais » les uns les autres.

Mode Maintenance : désactive la surveillance de toutes les ressources et empêche Pacemaker de déplacer ou de gérer les ressources dans le cluster (voir également la section "[mode maintenance](#)").

Cluster HA: un ou plusieurs nœuds de fichiers exécutant les services BeeGFS qui peuvent basculer entre plusieurs nœuds du cluster pour créer un système de fichiers BeeGFS haute disponibilité. Ils sont souvent configurés en paires HA et qui peuvent exécuter un sous-ensemble des services BeeGFS dans le cluster.

Quand utiliser Ansible contre l'outil pcs

Quand devriez-vous utiliser Ansible par rapport à l'outil de ligne de commande pcs pour gérer le cluster HA ?

Toutes les tâches de déploiement et de reconfiguration du cluster doivent être effectuées à l'aide d'Ansible à partir d'un nœud de contrôle Ansible externe. Les modifications temporaires de l'état du cluster (par exemple, placement des nœuds en veille ou en dehors) sont généralement effectuées en se connectant à un nœud du cluster (de préférence un nœud qui n'est pas dégradé ou sur le point de subir des opérations de maintenance) et en utilisant l'outil de ligne de commande pcs.

Tout changement de configuration de cluster, y compris les ressources, les contraintes, les propriétés et les services BeeGFS, doit toujours être effectué à l'aide d'Ansible. Maintenir une copie à jour de l'inventaire et du manuel de vente Ansible (idéalement en contrôle source pour suivre les modifications) fait partie de la maintenance du cluster. Si vous devez modifier la configuration, mettez à jour l'inventaire et exécutez à nouveau le PlayBook Ansible qui importe le rôle BeeGFS HA.

Le rôle HA gère le placement du cluster en mode maintenance, puis les modifications nécessaires avant de redémarrer BeeGFS ou les services du cluster pour appliquer la nouvelle configuration. Le redémarrage complet de nœud n'est généralement pas nécessaire en dehors du déploiement initial. Toutefois, le redémarrage d'Ansible est généralement considéré comme une procédure « sûre », mais toujours recommandé pendant les fenêtres de maintenance ou hors heures si les services BeeGFS doivent redémarrer. Ces redémarrages ne doivent généralement pas provoquer d'erreurs d'application, mais peuvent nuire aux performances (que certaines applications peuvent traiter mieux que d'autres).

Le réexécution Ansible est également une option pour rétablir l'état optimal de l'ensemble du cluster. Dans certains cas, il peut récupérer l'état du cluster plus facilement qu'avec les pièces. Notamment en cas d'urgence où le cluster est hors service, une fois que tous les nœuds sont sauvegardées, Ansible peut récupérer le cluster plus rapidement et de façon plus fiable que toute tentative d'utilisation de pcs.

Vérifiez l'état du cluster

Utilisez les pièces pour voir l'état du bloc d'instruments.

Présentation

Exécution `pcs status` À partir de n'importe quel nœud de cluster est le moyen le plus simple de voir l'état global du cluster et l'état de chaque ressource (par exemple, les services BeeGFS et leurs dépendances). Cette section présente ce que vous trouverez dans les résultats du `pcs status` commande.

Présentation de la sortie de `pcs status`

Exécutez `pcs status` Sur n'importe quel nœud de cluster où les services de cluster (Pacemaker et Corosync) sont démarrés. Le haut de la sortie affiche un récapitulatif du cluster :

```
[root@beegfs_01 ~]# pcs status
Cluster name: hacluster
Cluster Summary:
  * Stack: corosync
  * Current DC: beegfs_01 (version 2.0.5-9.el8_4.3-ba59be7122) - partition
with quorum
  * Last updated: Fri Jul 1 13:37:18 2022
  * Last change: Fri Jul 1 13:23:34 2022 by root via cibadmin on
beegfs_01
  * 6 nodes configured
  * 235 resource instances configured
```

La section ci-dessous liste les nœuds du cluster :

```
Node List:
  * Node beegfs_06: standby
  * Online: [ beegfs_01 beegfs_02 beegfs_04 beegfs_05 ]
  * OFFLINE: [ beegfs_03 ]
```

Cela indique notamment tous les nœuds en veille ou hors ligne. Les nœuds en veille font toujours partie du cluster, mais sont marqués comme non éligibles pour l'exécution des ressources. Les nœuds hors ligne indiquent que les services du cluster ne s'exécutent pas sur ce nœud, soit en raison d'un arrêt manuel, soit en raison du redémarrage ou de l'arrêt du nœud.



Lorsque les nœuds démarrent initialement, les services de cluster sont arrêtés et doivent être démarrés manuellement pour éviter de basculer accidentellement des ressources sur un nœud défaillant.

Si les nœuds sont en attente ou hors ligne en raison d'une raison non administrative (par exemple, une panne), un texte supplémentaire s'affiche à côté de l'état du nœud entre parenthèses. Par exemple, si l'escrime est désactivé et qu'une ressource rencontre une défaillance, vous verrez Node <HOSTNAME> :

standby (on-fail). Un autre état possible est Node <HOSTNAME>: UNCLEAN (offline), qui sera brièvement vu comme un nœud est clôturé, mais persistera si l'escrime a échoué indiquant que le cluster ne peut pas confirmer l'état du nœud (cela peut bloquer les ressources de démarrer sur d'autres noeuds).

La section suivante affiche la liste de toutes les ressources du cluster et leur état :

Full List of Resources:

```
* mgmt-monitor    (ocf::eseries:beegfs-monitor):    Started beegfs_01
* Resource Group: mgmt-group:
  * mgmt-FS1    (ocf::eseries:beegfs-target):    Started beegfs_01
  * mgmt-IP1    (ocf::eseries:beegfs-ipaddr2):    Started beegfs_01
  * mgmt-IP2    (ocf::eseries:beegfs-ipaddr2):    Started beegfs_01
  * mgmt-service (systemd:beegfs-mgtd):    Started beegfs_01
[....]
```

Tout comme les nœuds, un texte supplémentaire s'affiche en regard de l'état de la ressource entre parenthèses s'il y a des problèmes avec la ressource. Par exemple, si Pacemaker demande un arrêt de ressource et qu'il ne s'effectue pas dans le temps alloué, Pacemaker tente de verrouiller le nœud. Si l'escrime est désactivé ou que l'opération d'escrime échoue, l'état de la ressource sera FAILED <HOSTNAME> (blocked). Et Pacemaker ne pourra pas démarrer sur un autre nœud.

Il est utile de noter que les clusters BeeGFS HA utilisent un certain nombre d'agents de ressources personnalisées de BeeGFS optimisés pour les OCF. En particulier, le moniteur BeeGFS est responsable du déclenchement d'un basculement lorsque les ressources BeeGFS sur un nœud donné ne sont pas disponibles.

Reconfigurer le cluster HA et BeeGFS

Utilisez Ansible pour reconfigurer le cluster.

Présentation

En général, la reconfiguration d'un aspect du cluster BeeGFS haute disponibilité doit être effectuée en mettant à jour votre inventaire Ansible et en réexécutant `ansible-playbook` la commande. Cela inclut la mise à jour des alertes, la modification de la configuration de l'escrime permanent ou l'ajustement de la configuration du service BeeGFS. Ils sont ajustés à l'aide du `group_vars/ha_cluster.yml` fichier et une liste complète des options se trouve dans la ["Spécifiez la configuration de nœud de fichier commun"](#) section.

Pour plus d'informations sur les options de configuration que les administrateurs doivent connaître lors des opérations de maintenance ou de maintenance du cluster, consultez ci-dessous.

Comment désactiver et activer la fonction de fencing

Par défaut, l'escrime est activé/requis lors de la configuration du cluster. Dans certains cas, il peut être souhaitable de désactiver temporairement l'escrime pour s'assurer que les nœuds ne s'arrêtent pas accidentellement lors de certaines opérations de maintenance (par exemple, la mise à niveau du système d'exploitation). Bien que cette fonction puisse être désactivée manuellement, les administrateurs doivent en être conscients des compromis.

OPTION 1 : désactivez l'escrime avec Ansible (recommandé).

Lorsque l'escrime est désactivé à l'aide d'Ansible, l'action en cas d'échec du moniteur BeeGFS passe de « clôture » à « veille ». Cela signifie que si le moniteur BeeGFS détecte une défaillance, il tente de placer le nœud en veille et de basculer tous les services BeeGFS. En dehors du dépannage/test actif, ceci est généralement plus souhaitable que l'option 2. L'inconvénient est que si une ressource ne s'arrête pas sur le nœud d'origine, elle sera bloquée pour commencer ailleurs (c'est pourquoi une clôture est généralement nécessaire pour les grappes de production).

1. Dans votre inventaire Ansible à `groups_vars/ha_cluster.yml` ajoutez la configuration suivante :

```
beegfs_ha_cluster_crm_config_options:  
  stonith-enabled: False
```

2. Exécutez à nouveau le manuel de vente Ansible afin d'appliquer les modifications apportées au cluster.

OPTION 2 : désactivez manuellement l'escrime.

Dans certains cas, vous pouvez désactiver temporairement l'escrime sans qu'il soit nécessaire de réexécuter Ansible, afin de faciliter le dépannage ou le test du cluster.

 Dans cette configuration, si le moniteur BeeGFS détecte une défaillance, le cluster tente d'arrêter le groupe de ressources correspondant. Il NE déclenchera PAS un basculement complet ni ne tentera de redémarrer ou de déplacer le groupe de ressources affecté vers un autre hôte. Pour restaurer le système, traitez les problèmes avant de l'exécuter `pcs resource cleanup` ou placez manuellement le nœud en veille.

Étapes :

1. Pour déterminer si l'escrime (stonith) est globalement activé ou désactivé : `pcs property show stonith-enabled`
2. Pour désactiver la séquence d'escrime : `pcs property set stonith-enabled=false`
3. Pour activer la séquence d'escrime : `pcs property set stonith-enabled=true`

 Ce paramètre sera remplacé la prochaine fois que vous exécuterez le playbook Ansible.

Mettez à jour les composants du cluster HA

Mise à niveau des services BeeGFS

Utilisez Ansible pour mettre à jour la version de BeeGFS exécutée sur votre cluster HA.

Présentation

BeeGFS applique un `major.minor.patch` schéma de gestion des versions. Des rôles Ansible haute disponibilité BeeGFS sont fournis pour chaque `major.minor` version prise en charge (par exemple, `beegfs_ha_7_2` et `beegfs_ha_7_3`). Chaque rôle HA est épingle à la dernière version de correctif BeeGFS disponible au moment de la publication de la collection Ansible.

Ansible doit être utilisé pour toutes les mises à niveau de BeeGFS, y compris le passage entre les versions majeure, mineure et corrective de BeeGFS. Pour mettre à jour BeeGFS, vous devrez d'abord mettre à jour la collection Ansible BeeGFS, ce qui intégrera également les derniers correctifs et améliorations de l'automatisation du déploiement/gestion et du cluster HA sous-jacent. Même après la mise à jour vers la dernière version de la collection, BeeGFS ne sera pas mis à niveau tant que `ansible-playbook` n'aura pas été exécuté avec l'`-e "beegfs_ha_force_upgrade=true"` activé. Pour plus de détails sur chaque mise à niveau, consultez la ["Documentation de mise à niveau BeeGFS"](#) pour votre version actuelle.



Si vous effectuez une mise à niveau vers BeeGFS v8, consultez plutôt la ["Mise à jour vers BeeGFS v8"](#) procédure.

Chemins de mise à niveau testés

Les voies de mise à niveau suivantes ont été testées et vérifiées :

Version d'origine	Mettre à niveau la version	Multirail	Détails
7.2.6	7.3.2	Oui.	Mise à niveau de la collection beegfs de v3.0.1 à v3.1.0, multirail ajouté
7.2.6	7.2.8	Non	Mise à niveau de la collection beegfs de v3.0.1 à v3.1.0
7.2.8	7.3.1	Oui.	Mise à niveau avec beegfs collection v3.1.0, multirail ajouté
7.3.1	7.3.2	Oui.	Mise à niveau avec beegfs collection v3.1.0
7.3.2	7.4.1	Oui.	Mise à niveau avec beegfs collection v3.2.0
7.4.1	7.4.2	Oui.	Mise à niveau avec beegfs collection v3.2.0
7.4.2	7.4.6	Oui.	Mise à niveau avec beegfs collection v3.2.0
7.4.6	8,0	Oui.	Mettez à niveau en suivant les instructions dans la "Mise à jour vers BeeGFS v8" procédure.
7.4.6	8,1	Oui.	Mettez à niveau en suivant les instructions dans la "Mise à jour vers BeeGFS v8" procédure.
7.4.6	8,2	Oui.	Mettez à niveau en suivant les instructions dans la "Mise à jour vers BeeGFS v8" procédure.

Étapes de mise à niveau BeeGFS

Les sections suivantes expliquent comment mettre à jour la collection BeeGFS Ansible et BeeGFS. Portez une attention particulière à toute étape(s) supplémentaire(s) pour la mise à jour de BeeGFS version majeure ou mineure.

Étape 1 : mise à niveau de la collection BeeGFS

Pour les mises à niveau de collecte avec accès à ["Galaxy Ansible"](#), exécutez la commande suivante :

```
ansible-galaxy collection install netapp_eseries.beegfs --upgrade
```

Pour les mises à niveau hors ligne de la collection, téléchargez la collection à partir de ["Galaxy Ansible"](#) en

cliquant sur le bouton souhaité `Install Version` puis `Download tarball`. Transférez le tarball sur votre nœud de contrôle Ansible, puis exécutez la commande suivante.

```
ansible-galaxy collection install netapp_eseries-beegfs-<VERSION>.tar.gz  
--upgrade
```

Voir "[Installation de Collections](#)" pour en savoir plus.

Étape 2 : mise à jour de l'inventaire Ansible

Apportez toutes les mises à jour requises ou souhaitées aux fichiers d'inventaire Ansible de votre cluster. Voir la section [Notes de mise à niveau de la version](#) ci-dessous pour plus de détails sur vos exigences spécifiques de mise à niveau. Voir la section "[Présentation d'Ansible Inventory](#)" pour des informations générales sur la configuration de votre inventaire BeeGFS HA.

Étape 3 : mise à jour du PlayBook Ansible (uniquement en cas de mise à jour des versions principales ou secondaires)

Si vous passez d'une version majeure à une version mineure, dans le `playbook.yml` fichier utilisé pour déployer et gérer le cluster, mettez à jour le nom du `beegfs_ha_<VERSION>` rôle pour refléter la version souhaitée. Par exemple, si vous souhaitez déployer BeeGFS 7.4 `beegfs_ha_7_4`:

```
- hosts: all  
gather_facts: false  
any_errors_fatal: true  
collections:  
  - netapp_eseries.beegfs  
tasks:  
  - name: Ensure BeeGFS HA cluster is setup.  
    ansible.builtin.import_role: # import_role is required for tag  
    availability.  
    name: beegfs_ha_7_4
```

Pour plus de détails sur le contenu de ce fichier PlayBook "[Déployez le cluster BeeGFS HA](#)", reportez-vous à la section.

Étape 4 : exécutez la mise à niveau BeeGFS

Pour appliquer la mise à jour BeeGFS :

```
ansible-playbook -i inventory.yml beegfs_ha_playbook.yml -e  
"beegfs_ha_force_upgrade=true" --tags beegfs_ha
```

En coulisse, le rôle haute disponibilité BeeGFS gère :

- Assurez-vous que le cluster est dans un état optimal avec chaque service BeeGFS situé sur son nœud préféré.
- Mettre le cluster en mode maintenance.

- Mettre à jour les composants du cluster haute disponibilité (le cas échéant)
- Mettez à niveau chaque nœud de fichiers un par un en procédant comme suit :
 - Mettez le système en veille et basculez ses services vers le nœud secondaire.
 - Mise à jour des packs BeeGFS.
 - Proposer de nouveaux services.
- Déplacez le cluster hors du mode maintenance.

Notes de mise à niveau de la version

Mise à jour de BeeGFS version 7.2.6 ou 7.3.0

Modifications de l'authentification basée sur la connexion

BeeGFS version 7.3.2 et ultérieures nécessitent que l'authentification basée sur la connexion soit configurée. Les services ne démarreront pas sans l'une des options suivantes :

- Spécifier un connAuthFile, ou
- Paramétrage connDisableAuthentication=true dans le fichier de configuration du service.

Il est fortement recommandé d'activer l'authentification basée sur la connexion pour des raisons de sécurité. Voir "[Authentification basée sur la connexion BeeGFS](#)" pour plus d'informations.

Les beegfs_ha* rôles génèrent et distribuent automatiquement le fichier d'authentification à :

- Tous les nœuds de fichiers du cluster
- Le nœud de contrôle Ansible à
`<playbook_directory>/files/beegfs/<beegfs_mgmt_ip_address>_connAuthFile`

Le beegfs_client rôle détectera et appliquera automatiquement ce fichier aux clients lorsqu'il sera présent.

Si vous n'avez pas utilisé le beegfs_client rôle pour configurer les clients, vous devez distribuer manuellement le fichier d'authentification à chaque client et configurer le paramètre connAuthFile dans le fichier beegfs-client.conf. Lors d'une mise à niveau depuis une version de BeeGFS sans authentification basée sur la connexion, les clients perdront l'accès sauf si vous désactivez l'authentification basée sur la connexion pendant la mise à niveau en définissant beegfs_ha_conn_auth_enabled: false dans group_vars/ha_cluster.yml (non recommandé).

Pour plus de détails et d'options de configuration alternatives, consultez l'étape de configuration de l'authentification de connexion dans la section "[Spécifiez la configuration de nœud de fichier commun](#)".

Mise à jour vers BeeGFS v8

Suivez ces étapes pour mettre à niveau votre cluster BeeGFS HA de la version 7.4.6 à BeeGFS v8.

Présentation

BeeGFS v8 introduit plusieurs changements importants qui nécessitent une configuration supplémentaire

avant la mise à niveau depuis BeeGFS v7. Ce document vous guide dans la préparation de votre cluster aux nouvelles exigences de BeeGFS v8, puis dans la mise à niveau vers BeeGFS v8.

 Avant de procéder à la mise à niveau vers BeeGFS v8, assurez-vous que votre système exécute au moins BeeGFS 7.4.6. Tout cluster exécutant une version antérieure à BeeGFS 7.4.6 doit d'abord "["Mise à jour vers la version 7.4.6"](#)" avant de poursuivre cette procédure de mise à niveau vers BeeGFS v8.

Principaux changements dans BeeGFS v8

BeeGFS v8 introduit les changements majeurs suivants :

- **Application des licences** : BeeGFS v8 requiert une licence pour utiliser les fonctionnalités premium telles que les pools de stockage, les cibles de stockage distantes, BeeOND, et plus encore. Procurez-vous une licence valide pour votre cluster BeeGFS avant la mise à niveau. Si nécessaire, vous pouvez obtenir une licence d'évaluation temporaire de BeeGFS v8 auprès du "["Portail de licences BeeGFS"](#)".
- **Migration de la base de données du service de gestion** : Pour activer la configuration avec le nouveau format basé sur TOML dans BeeGFS v8, vous devez migrer manuellement votre base de données du service de gestion BeeGFS v7 vers le format BeeGFS v8 mis à jour.
- **Chiffrement TLS** : BeeGFS v8 introduit TLS pour sécuriser la communication entre les services. Vous devrez générer et distribuer des certificats TLS pour le service de gestion BeeGFS et l' `beegfs` utilitaire en ligne de commande dans le cadre de la mise à niveau.

Pour plus de détails et les modifications supplémentaires apportées à BeeGFS 8, consultez le "["Guide de mise à niveau BeeGFS v8.0.0"](#)".

 La mise à niveau vers BeeGFS v8 nécessite une interruption du cluster. De plus, les clients BeeGFS v7 ne peuvent pas se connecter aux clusters BeeGFS v8. Coordonnez soigneusement le calendrier de mise à niveau entre le cluster et les clients afin de minimiser l'impact sur les opérations.

Préparez votre cluster BeeGFS pour la mise à niveau

Avant de commencer la mise à niveau, préparez soigneusement votre environnement afin d'assurer une transition en douceur et de minimiser l'interruption.

1. Assurez-vous que votre cluster est dans un état sain, avec tous les services BeeGFS exécutés sur leurs nœuds préférés. À partir d'un nœud de fichiers exécutant les services BeeGFS, vérifiez que toutes les ressources Pacemaker sont exécutées sur leurs nœuds préférés :

```
pcs status
```

2. Enregistrez et sauvegardez la configuration de votre cluster.
 - a. Consultez le "["Documentation de sauvegarde BeeGFS"](#)" pour obtenir des instructions sur la sauvegarde de la configuration de votre cluster.
 - b. Sauvegardez le répertoire de données de gestion existant :

```
cp -r /mnt/mgmt_tgt_mgmt01/data  
/mnt/mgmt_tgt_mgmt01/data_beegfs_v7_backup_$(date +%Y%m%d)
```

c. Exécutez les commandes suivantes depuis un client beegfs et enregistrez leur sortie pour référence :

```
beegfs-ctl --getentryinfo --verbose /path/to/beegfs/mountpoint
```

d. Si vous utilisez la mise en miroir, recueillez des informations détaillées sur l'état :

```
beegfs-ctl --listtargets --longnodes --state --spaceinfo  
--mirrorgroups --nodetype=meta  
beegfs-ctl --listtargets --longnodes --state --spaceinfo  
--mirrorgroups --nodetype=storage
```

3. Préparez vos clients à l'interruption beegfs-client des services. Pour chaque client, exécutez :

```
systemctl stop beegfs-client
```

4. Pour chaque cluster Pacemaker, désactivez STONITH. Cela vous permettra de vérifier l'intégrité du cluster après la mise à niveau sans provoquer de redémarrages inutiles des nœuds.

```
pcs property set stonith-enabled=false
```

5. Pour tous les clusters Pacemaker dans l'espace de noms BeeGFS, utilisez PCS pour arrêter le cluster :

```
pcs cluster stop --all
```

Mettez à niveau les packages BeeGFS

Sur tous les nœuds de fichiers du cluster, ajoutez le dépôt de paquets BeeGFS v8 correspondant à votre distribution Linux. Des instructions pour utiliser les dépôts officiels BeeGFS sont disponibles à ["page de téléchargement BeeGFS"](#). Sinon, configurez votre dépôt miroir local BeeGFS en conséquence.

La procédure suivante décrit comment procéder à l'aide du dépôt officiel BeeGFS 8.2 sur des nœuds de fichiers RHEL 9. Effectuez les étapes suivantes sur tous les nœuds de fichiers du cluster :

1. Importez la clé GPG de BeeGFS :

```
rpm --import https://www.beegfs.io/release/beegfs_8.2/gpg/GPG-KEY-beegfs
```

2. Importez le dépôt BeeGFS :

```
curl -L -o /etc/yum.repos.d/beegfs-rhel9.repo  
https://www.beegfs.io/release/beegfs_8.2/dists/beegfs-rhel9.repo
```



Supprimez tous les dépôts BeeGFS précédemment configurés pour éviter les conflits avec le nouveau dépôt BeeGFS v8.

3. Videz le cache de votre gestionnaire de paquets :

```
dnf clean all
```

4. Sur tous les nœuds de fichiers, mettez à jour les paquets BeeGFS vers BeeGFS 8.2.

```
dnf update beegfs-mgtd beegfs-storage beegfs-meta libbeegfs-ib
```



Dans un cluster standard, le `beegfs-mgtd` package ne sera mis à jour que sur les deux premiers nœuds de fichiers.

Mettre à niveau la base de données de gestion

Sur l'un des nœuds de fichiers exécutant le service de gestion BeeGFS, effectuez les étapes suivantes pour migrer la base de données de BeeGFS v7 vers v8.

1. Lister tous les périphériques NVMe et filtrer selon la cible de gestion :

```
nvme netapp smdevices | grep mgmt_tgt
```

a. Notez le chemin d'accès au périphérique dans la sortie.

b. Montez le périphérique cible de gestion sur le point de montage cible de gestion existant (remplacez `/dev/nvmeXnY` par le chemin d'accès à votre périphérique) :

```
mount /dev/nvmeXnY /mnt/mgmt_tgt_mgmt01/
```

2. Importez vos données de gestion BeeGFS 7 dans le nouveau format de base de données en exécutant :

```
/opt/beegfs/sbin/beegfs-mgtd --import-from  
-v7=/mnt/mgmt_tgt_mgmt01/data/
```

Résultat attendu:

```
Created new database version 3 at "/var/lib/beegfs/mgmtd.sqlite".
Successfully imported v7 management data from
"/mnt/mgmt_tgt_mgmt01/data/".
```



L'importation automatique peut échouer dans certains cas en raison des exigences de validation plus strictes dans BeeGFS v8. Par exemple, si des cibles sont affectées à des pools de stockage inexistant, l'importation échouera. Si la migration de la base de données échoue, ne procédez pas à la mise à niveau. Contactez le support NetApp pour obtenir de l'aide concernant la résolution des problèmes de migration de la base de données. À titre de solution temporaire, vous pouvez rétrograder les packages BeeGFS v8 et continuer à utiliser BeeGFS v7 pendant que le problème est résolu.

3. Déplacez le fichier SQLite généré vers le point de montage du service de gestion :

```
mv /var/lib/beegfs/mgmtd.sqlite /mnt/mgmt_tgt_mgmt01/data/
```

4. Déplacez le fichier généré `beegfs-mgmtd.toml` vers le point de montage du service de gestion :

```
mv /etc/beegfs/beegfs-mgmtd.toml /mnt/mgmt_tgt_mgmt01/mgmt_config/
```

La préparation du fichier de configuration `beegfs-mgmtd.toml` sera effectuée après avoir terminé les étapes de configuration de la licence et du chiffrement TLS dans les sections suivantes.

Configurer les licences

1. Installez les packages de licence `beegfs` sur tous les nœuds qui exécutent le service de gestion `beegfs`. Il s'agit généralement des deux premiers nœuds du cluster :

```
dnf install libbeegfs-license
```

2. Téléchargez votre fichier de licence BeeGFS v8 sur les nœuds de gestion et placez-le à :

```
/etc/beegfs/license.pem
```

Configurer le chiffrement TLS

BeeGFS v8 requiert le chiffrement TLS pour sécuriser les communications entre les services de gestion et les clients. Il existe trois options pour configurer le chiffrement TLS sur les communications réseau entre les services de gestion et les services clients. La méthode recommandée et la plus sécurisée consiste à utiliser des certificats signés par une autorité de certification de confiance. Vous pouvez également créer votre propre autorité de certification locale pour signer les certificats de votre cluster BeeGFS. Pour les environnements où le chiffrement n'est pas requis ou pour le dépannage, TLS peut être entièrement désactivé, bien que cela soit déconseillé car cela expose des informations sensibles au réseau.

Avant de continuer, suivez les instructions du "["Configurer le chiffrement TLS pour BeeGFS 8"](#)" guide pour configurer le chiffrement pour votre environnement.

Configuration du service de gestion des mises à jour

Préparez le fichier de configuration du service de gestion BeeGFS v8 en transférant manuellement les paramètres de votre fichier de configuration BeeGFS v7 dans le fichier

/mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-mgmtd.toml.

1. Sur le nœud de gestion où la cible de gestion est montée, référez le /mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-mgmtd.conf fichier de service de gestion pour BeeGFS 7, puis transférez tous les paramètres dans le fichier /mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-mgmtd.toml. Pour une configuration de base, votre beegfs-mgmtd.toml pourrait ressembler à ce qui suit :

```
beemsg-port = 8008
grpc-port = 8010
log-level = "info"
node-offline-timeout = "900s"
quota-enable = false
auth-disable = false
auth-file = "/etc/beegfs/<mgmt_service_ip>_connAuthFile"
db-file = "/mnt/mgmt_tgt_mgmt01/data/mgmtd.sqlite"
license-disable = false
license-cert-file = "/etc/beegfs/license.pem"
tls-disable = false
tls-cert-file = "/etc/beegfs/mgmtd_tls_cert.pem"
tls-key-file = "/etc/beegfs/mgmtd_tls_key.pem"
interfaces = ['i1b:mgmt_1', 'i2b:mgmt_2']
```

Adaptez tous les chemins selon les besoins pour correspondre à votre environnement et à votre configuration TLS.

2. Sur chaque nœud de fichiers exécutant des services de gestion, modifiez votre fichier de service systemd pour qu'il pointe vers le nouvel emplacement du fichier de configuration.

```
sudo sed -i 's|ExecStart=.*|ExecStart=nice -n -3
/opt/beegfs/sbin/beegfs-mgmtd --config-file
/mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-mgmtd.toml|'
/etc/systemd/system/beegfs-mgmtd.service
```

- a. Recharger systemd :

```
systemctl daemon-reload
```

3. Pour chaque nœud de fichier exécutant des services de gestion, ouvrez le port 8010 pour la

communication gRPC du service de gestion.

- Ajoutez le port 8010/tcp à la zone beegfs :

```
sudo firewall-cmd --zone=beegfs --permanent --add-port=8010/tcp
```

- Rechargez le pare-feu pour appliquer la modification :

```
sudo firewall-cmd --reload
```

Mettre à jour le script de surveillance BeeGFS

Le script OCF de `beegfs-monitor` Pacemaker nécessite des mises à jour pour prendre en charge le nouveau format de configuration TOML et la gestion des services systemd. Mettez à jour le script sur un nœud du cluster, puis copiez le script mis à jour sur tous les autres nœuds.

- Créez une sauvegarde du script actuel :

```
cp /usr/lib/ocf/resource.d/eseries/beegfs-monitor  
/usr/lib/ocf/resource.d/eseries/beegfs-monitor.bak.$(date +%F)
```

- Mettez à jour le chemin du fichier de configuration de gestion de .conf à .toml :

```
sed -i 's|mgmt_config/beegfs-mgmtd\.conf|mgmt_config/beegfs-mgmtd.toml|'  
/usr/lib/ocf/resource.d/eseries/beegfs-monitor
```

Sinon, localisez manuellement le bloc suivant dans le script :

```
case $type in  
  management)  
    conf_path="${configuration_mount}/mgmt_config/beegfs-mgmtd.conf"  
  ;;
```

Et remplacez-le par :

```
case $type in  
  management)  
    conf_path="${configuration_mount}/mgmt_config/beegfs-mgmtd.toml"  
  ;;
```

- Mettez à jour les `get_interfaces()` et `get_subnet_ips()` fonctions pour prendre en charge la configuration TOML :

- a. Ouvrez le script dans un éditeur de texte :

```
vi /usr/lib/ocf/resource.d/eseries/beegfs-monitor
```

- b. Localisez les deux fonctions : `get_interfaces()` et `get_subnet_ips()`.
- c. Supprimez les deux fonctions complètes, en commençant à `get_interfaces()` jusqu'à la fin de `get_subnet_ips()`.
- d. Copiez et collez les fonctions mises à jour suivantes à leur place :

```

# Return network communication interface name(s) from the BeeGFS
resource's connInterfaceFile
get_interfaces() {
    # Determine BeeGFS service network IP interfaces.
    if [ "$type" = "management" ]; then
        interfaces_line=$(grep "interfaces =" "$conf_path")
        interfaces_list=$(echo "$interfaces_line" | sed "s/.*= \[\(\.\*"
        \)\]/\1/")
        interfaces=$(echo "$interfaces_list" | tr -d '"' | tr -d " " | tr
        ',' '\n')

        for entry in $interfaces; do
            echo "$entry" | cut -d ':' -f 1
        done
    else
        connInterfacesFile_path=$(grep "connInterfacesFile" "$conf_path"
        | tr -d "[[:space:]]" | cut -f 2 -d "=")

        if [ -f "$connInterfacesFile_path" ]; then
            while read -r entry; do
                echo "$entry" | cut -f 1 -d ':'
            done < "$connInterfacesFile_path"
        fi
    fi
}

# Return list containing all the BeeGFS resource's usable IP
addresses. *Note that these are filtered by the connNetFilterFile
entries.
get_subnet_ips() {
    # Determine all possible BeeGFS service network IP addresses.
    if [ "$type" != "management" ]; then
        connNetFilterFile_path=$(grep "connNetFilterFile" "$conf_path" |
        tr -d "[[:space:]]" | cut -f 2 -d "=")

        filter_ips=""
        if [ -n "$connNetFilterFile_path" ] && [ -e
$connNetFilterFile_path ]; then
            while read -r filter; do
                filter_ips="$filter_ips $(get_ipv4_subnet_addresses $filter)"
            done < $connNetFilterFile_path
        fi

        echo "$filter_ips"
    fi
}

```

- e. Enregistrez et quittez l'éditeur de texte.
- f. Exécutez la commande suivante pour vérifier le script pour des erreurs de syntaxe avant de poursuivre. L'absence de résultat indique que le script est syntaxiquement correct.

```
bash -n /usr/lib/ocf/resource.d/eseries/beegfs-monitor
```

4. Copiez le script OCF mis à jour `beegfs-monitor` sur tous les autres nœuds du cluster pour garantir la cohérence :

```
scp /usr/lib/ocf/resource.d/eseries/beegfs-monitor
user@node:/usr/lib/ocf/resource.d/eseries/beegfs-monitor
```

Remettre le cluster en ligne

1. Une fois toutes les étapes de mise à niveau précédentes terminées, remettez le cluster en ligne en démarrant les services BeeGFS sur tous les nœuds.

```
pcs cluster start --all
```

2. Vérifiez que le `beegfs-mgtd` service a démarré correctement :

```
journalctl -xeu beegfs-mgtd
```

Le résultat attendu comprend des lignes telles que :

```
Started Cluster Controlled beegfs-mgtd.
Loaded config file from "/mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-
mgtd.toml"
Successfully initialized certificate verification library.
Successfully loaded license certificate: TMP-113489268
Opened database at "/mnt/mgmt_tgt_mgmt01/data/mgtd.sqlite"
Listening for BeeGFS connections on [::]:8008
Serving gRPC requests on [::]:8010
```



Si des erreurs apparaissent dans les journaux, vérifiez les chemins d'accès au fichier de configuration de gestion et assurez-vous que toutes les valeurs ont été correctement transférées depuis le fichier de configuration BeeGFS 7.

3. Exécutez `pcs status` et vérifiez que le cluster est sain et que les services sont démarrés sur leurs nœuds préférés.
4. Une fois que le cluster est vérifié comme étant sain, réactivez STONITH :

```
pcs property set stonith-enabled=true
```

5. Passez à la section suivante pour mettre à niveau les clients BeeGFS dans le cluster et vérifier l'état de santé du cluster BeeGFS.

Mise à niveau des clients BeeGFS

Après avoir réussi la mise à niveau de votre cluster vers BeeGFS v8, vous devez également mettre à niveau tous les clients BeeGFS.

Les étapes suivantes décrivent le processus de mise à niveau des clients BeeGFS sur un système basé sur Ubuntu.

1. Si ce n'est pas déjà fait, arrêtez le service client BeeGFS :

```
systemctl stop beegfs-client
```

2. Ajoutez le dépôt de paquets BeeGFS v8 pour votre distribution Linux. Des instructions pour utiliser les dépôts officiels BeeGFS se trouvent à "[Page de téléchargement BeeGFS](#)". Sinon, configurez votre dépôt miroir BeeGFS local en conséquence.

Les étapes suivantes utilisent le dépôt officiel BeeGFS 8.2 sur un système basé sur Ubuntu :

3. Importez la clé GPG de BeeGFS :

```
wget https://www.beegfs.io/release/beegfs_8.2/gpg/GPG-KEY-beegfs -O  
/etc/apt/trusted.gpg.d/beegfs.asc
```

4. Téléchargez le fichier du dépôt :

```
wget https://www.beegfs.io/release/beegfs_8.2/dists/beegfs-noble.list -O  
/etc/apt/sources.list.d/beegfs.list
```



Supprimez tous les dépôts BeeGFS précédemment configurés pour éviter les conflits avec le nouveau dépôt BeeGFS v8.

5. Mettez à jour les packages clients BeeGFS :

```
apt-get update  
apt-get install --only-upgrade beegfs-client
```

6. Configurez TLS pour le client. TLS est requis pour utiliser la CLI BeeGFS. Consultez la "["Configurer le chiffrement TLS pour BeeGFS 8"](#)" procédure pour configurer TLS sur le client.
7. Démarrer le service client BeeGFS :

```
systemctl start beegfs-client
```

Vérifier la mise à jour

Après avoir terminé la mise à niveau vers BeeGFS v8, exécutez les commandes suivantes pour vérifier que la mise à niveau a réussi.

1. Vérifiez que l'inode racine appartient bien au même nœud de métadonnées qu'auparavant. Cela devrait se faire automatiquement si vous avez utilisé la `import-from-v7` fonctionnalité dans le service de gestion :

```
beegfs entry info /mnt/beegfs
```

2. Vérifiez que tous les nœuds et cibles sont en ligne et en bon état :

```
beegfs health check
```



Si la vérification de la « capacité disponible » signale que les cibles manquent d'espace libre, vous pouvez ajuster les seuils du « pool de capacité » définis dans le `beegfs-mgmtd.toml` fichier afin qu'ils soient mieux adaptés à votre environnement.

Mise à niveau des packages Pacemaker et Corosync dans un cluster haute disponibilité

Procédez comme suit pour mettre à niveau les packages Pacemaker et Corosync dans un cluster HA.

Présentation

La mise à niveau de Pacemaker et Corosync garantit que le cluster bénéficie de nouvelles fonctionnalités, de nouveaux correctifs de sécurité et d'améliorations des performances.

Approche de mise à niveau

Il existe deux approches recommandées pour la mise à niveau d'un cluster : une mise à niveau par déploiement ou un arrêt complet du cluster. Chaque approche a ses propres avantages et inconvénients. La procédure de mise à niveau peut varier en fonction de la version de votre Pacemaker. Reportez-vous à la documentation de ClusterLabs "[Mise à niveau d'un cluster Pacemaker](#)" pour déterminer quelle approche utiliser. Avant de suivre une approche de mise à niveau, vérifiez que :

- Les nouveaux packages Pacemaker et Corosync sont pris en charge dans la solution NetApp BeeGFS.
- Il existe des sauvegardes valides pour votre système de fichiers BeeGFS et la configuration de cluster Pacemaker.
- Le cluster est en bon état.

Mise à jour du déploiement

Cette méthode implique de supprimer chaque nœud du cluster, de le mettre à niveau, puis de le réintégrer dans le cluster jusqu'à ce que tous les nœuds exécutent la nouvelle version. Cette approche assure le fonctionnement continu du cluster, ce qui est idéal pour les clusters haute disponibilité de plus grande taille, mais comporte le risque d'exécuter des versions mixtes lors du processus. Cette approche doit être évitée dans un cluster à deux nœuds.

1. Vérifiez que le cluster est dans un état optimal, chaque service BeeGFS étant exécuté sur le nœud favori. Voir "[Vérifiez l'état du cluster](#)" pour plus de détails.
2. Pour la mise à niveau du nœud, mettez-le en mode veille afin d'exploiter (ou de déplacer) tous les services BeeGFS :

```
pcs node standby <HOSTNAME>
```

3. Vérifier que les services du nœud ont été vidangés en exécutant :

```
pcs status
```

Assurez-vous qu'aucun service n'est signalé comme **Started** sur le nœud en veille.



Selon la taille de votre cluster, le déplacement des services vers le nœud sœur peut prendre quelques secondes, voire quelques minutes. Si un service BeeGFS ne démarre pas sur le nœud Sister, reportez-vous au "[Guides de dépannage](#)".

4. Arrêter le cluster sur le nœud :

```
pcs cluster stop <HOSTNAME>
```

5. Mettez à niveau les packages Pacemaker, Corosync et pcs sur le nœud :



Les commandes du gestionnaire de paquets varient selon le système d'exploitation. Les commandes suivantes sont destinées aux systèmes exécutant RHEL 8 et versions ultérieures.

```
dnf update pacemaker-<version>
```

```
dnf update corosync-<version>
```

```
dnf update pcs-<version>
```

6. Démarrer les services de cluster Pacemaker sur le nœud :

```
pcs cluster start <HOSTNAME>
```

7. Si le `pcs` pack a été mis à jour, réauthentifier le nœud avec le cluster :

```
pcs host auth <HOSTNAME>
```

8. Vérifiez que la configuration du stimulateur est toujours valide avec l' `crm_verify` outil.



Cette vérification doit être effectuée une seule fois lors de la mise à niveau du cluster.

```
crm_verify -L -v
```

9. Mettre le nœud hors veille :

```
pcs node unstandby <HOSTNAME>
```

10. Retransférez tous les services BeeGFS vers le nœud de votre choix :

```
pcs resource relocate run
```

11. Répétez les étapes précédentes pour chaque nœud du cluster jusqu'à ce que tous les nœuds exécutent les versions Pacemaker, Corosync et `pcs` souhaitées.

12. Enfin, exédez `pcs status` et vérifiez que le cluster fonctionne correctement et `Current DC` indique la version du stimulateur souhaitée.



Si le `Current DC` indique « version limite », un nœud du cluster fonctionne toujours avec la version précédente de Pacemaker et doit être mis à niveau. Si un nœud mis à niveau ne parvient pas à rejoindre le cluster ou si les ressources ne démarrent pas, consultez les journaux du cluster et consultez les notes de mise à jour ou les guides de l'utilisateur Pacemaker pour connaître les problèmes de mise à niveau connus.

Arrêt complet du cluster

Dans cette approche, tous les nœuds et toutes les ressources du cluster sont arrêtés, les nœuds sont mis à niveau, puis le cluster est redémarré. Cette approche est nécessaire si les versions Pacemaker et Corosync ne prennent pas en charge une configuration en version mixte.

1. Vérifiez que le cluster est dans un état optimal, chaque service BeeGFS étant exécuté sur le nœud favori. Voir "[Vérifiez l'état du cluster](#)" pour plus de détails.
2. Arrêtez le logiciel de cluster (Pacemaker et Corosync) sur tous les nœuds.



Selon la taille du cluster, l'arrêt de tout le cluster peut prendre quelques secondes, voire quelques minutes.

```
pcs cluster stop --all
```

3. Une fois les services de cluster arrêtés sur tous les nœuds, mettez à niveau les packages Pacemaker, Corosync et pcs sur chaque nœud en fonction de vos besoins.



Les commandes du gestionnaire de paquets varient selon le système d'exploitation. Les commandes suivantes sont destinées aux systèmes exécutant RHEL 8 et versions ultérieures.

```
dnf update pacemaker-<version>
```

```
dnf update corosync-<version>
```

```
dnf update pcs-<version>
```

4. Une fois la mise à niveau de tous les nœuds effectuée, démarrez le logiciel du cluster sur tous les nœuds :

```
pcs cluster start --all
```

5. Si le `pcs` pack a été mis à jour, réauthentifier chaque nœud du cluster :

```
pcs host auth <HOSTNAME>
```

6. Enfin, exécutez `pcs status` et vérifiez que le cluster est en bon état et `Current DC` indique la version correcte du Pacemaker.



Si le `Current DC` indique « version limite », un nœud du cluster fonctionne toujours avec la version précédente de Pacemaker et doit être mis à niveau.

Mettez à jour le micrologiciel de l'adaptateur de nœud de fichier

Procédez comme suit pour mettre à jour les cartes ConnectX-7 du nœud de fichiers vers la dernière version du micrologiciel.

Présentation

La mise à jour du firmware de l'adaptateur ConnectX-7 peut être nécessaire pour prendre en charge un nouveau pilote MLNX_OFED, activer de nouvelles fonctionnalités ou corriger des bogues. Ce guide utilisera

l'utilitaire NVIDIA `mlxfwmanager` pour les mises à jour de la carte en raison de sa facilité d'utilisation et de son efficacité.

Mise à niveau

Ce guide présente deux approches de mise à jour du firmware de la carte ConnectX-7 : une mise à jour en continu et une mise à jour de cluster à deux nœuds. Choisissez l'approche de mise à jour appropriée en fonction de la taille de votre cluster. Avant d'effectuer les mises à jour du micrologiciel, vérifiez que :

- Un pilote MLNX_OFED pris en charge est installé, reportez-vous au "[exigences technologiques](#)".
- Il existe des sauvegardes valides pour votre système de fichiers BeeGFS et la configuration de cluster Pacemaker.
- Le cluster est en bon état.

Préparation de la mise à jour du micrologiciel

Il est recommandé d'utiliser l'utilitaire de NVIDIA `mlxfwmanager` pour mettre à jour le micrologiciel de l'adaptateur d'un nœud, qui est fourni avec le pilote MLNX_OFED de NVIDIA. Avant de commencer les mises à jour, téléchargez l'image du micrologiciel de la carte à partir de "[Site de support NVIDIA](#)" et stockez-la sur chaque nœud de fichier.



Pour les cartes Lenovo ConnectX-7, utilisez l'outil `mlxfwmanager_LES`, disponible sur la page NVIDIA "[Micrologiciel OEM](#)".

Approche de mise à jour par roulement

Cette approche est recommandée pour tout cluster haute disponibilité de plus de deux nœuds. Cette approche implique de mettre à jour le firmware des adaptateurs sur un nœud de fichiers à la fois afin que le cluster haute disponibilité puisse continuer à traiter les demandes, bien qu'il soit recommandé d'éviter de traiter les E/S pendant ce temps.

1. Vérifiez que le cluster est dans un état optimal, chaque service BeeGFS étant exécuté sur le nœud favori. Voir "[Vérifiez l'état du cluster](#)" pour plus de détails.
2. Choisissez un nœud de fichiers à mettre à jour et placez-le en mode veille pour drains (ou déplacer) tous les services BeeGFS de ce nœud :

```
pcs node standby <HOSTNAME>
```

3. Vérifier que les services du nœud ont été vidangés en exécutant :

```
pcs status
```

Vérifiez qu'aucun service ne signale le `Started` nœud en mode veille.



Selon la taille du cluster, le déplacement des services BeeGFS peut prendre quelques secondes, voire quelques minutes. Si un service BeeGFS ne démarre pas sur le nœud Sister, reportez-vous au "[Guides de dépannage](#)".

4. Mettez à jour le micrologiciel de l'adaptateur à l'aide de `mlxfwmanager`.

```
mlxfwmanager -i <path/to/firmware.bin> -u
```

Notez `PCI Device Name` que pour chaque adaptateur recevant des mises à jour de micrologiciel.

5. Réinitialisez chaque carte à l'aide de l'``mlxfwreset``utilitaire pour appliquer le nouveau micrologiciel.



Certaines mises à jour du micrologiciel peuvent nécessiter un redémarrage pour appliquer la mise à jour. Reportez-vous "[Limitations de mlxfwreset de NVIDIA](#)" à pour obtenir des conseils. Si un redémarrage est nécessaire, effectuez un redémarrage au lieu de réinitialiser les adaptateurs.

a. Arrêter le service `openhm` :

```
systemctl stop opensm
```

b. Exécutez la commande suivante pour chacune des `PCI Device Name` opérations précédemment notées.

```
mlxfwreset -d <pci_device_name> reset -y
```

c. Démarrer le service `openhm` :

```
systemctl start opensm
```

d. Redémarrez le `eseries_nvme_ib.service`.

```
systemctl restart eseries_nvme_ib.service
```

e. Vérifiez que les volumes de la baie de stockage de la série E sont présents.

```
multipath -ll
```

1. Exécutez `ibstat` et vérifiez que toutes les cartes fonctionnent avec la version de micrologiciel souhaitée :

```
ibstat
```

2. Démarrer les services de cluster Pacemaker sur le nœud :

```
pcs cluster start <HOSTNAME>
```

3. Mettre le nœud hors veille :

```
pcs node unstandby <HOSTNAME>
```

4. Retransférez tous les services BeeGFS vers le nœud de votre choix :

```
pcs resource relocate run
```

Répétez ces étapes pour chaque nœud de fichiers du cluster jusqu'à ce que tous les adaptateurs aient été mis à jour.

Approche de mise à jour des clusters à deux nœuds

Cette approche est recommandée pour les clusters haute disponibilité à deux nœuds uniquement. Cette approche est similaire à une mise à jour propagée, mais elle comprend des étapes supplémentaires pour éviter tout temps d'indisponibilité des services lorsque les services de cluster d'un nœud sont arrêtés.

1. Vérifiez que le cluster est dans un état optimal, chaque service BeeGFS étant exécuté sur le nœud favori. Voir "["Vérifiez l'état du cluster"](#) pour plus de détails.
2. Choisissez un nœud de fichiers à mettre à jour et placez le nœud en mode veille, ce qui draine (ou déplace) tous les services BeeGFS de ce nœud :

```
pcs node standby <HOSTNAME>
```

3. Vérifier que les ressources du nœud ont été vidées en exécutant :

```
pcs status
```

Vérifiez qu'aucun service ne signale le Started nœud en mode veille.



Selon la taille du cluster, le reporting par les services BeeGFS peut prendre quelques secondes, voire quelques minutes, comme Started sur le nœud jumeau. Si un service BeeGFS ne démarre pas, reportez-vous au "["Guides de dépannage"](#).

4. Placer le cluster en mode maintenance.

```
pcs property set maintenance-mode=true
```

5. Mettez à jour le micrologiciel de l'adaptateur à l'aide de `mlxfwmanager`.

```
mlxfwmanager -i <path/to/firmware.bin> -u
```

Notez PCI Device Name que pour chaque adaptateur recevant des mises à jour de micrologiciel.

6. Réinitialisez chaque carte à l'aide de l' `mlxfwreset` utilitaire pour appliquer le nouveau micrologiciel.



Certaines mises à jour du micrologiciel peuvent nécessiter un redémarrage pour appliquer la mise à jour. Reportez-vous "[Limitations de mlxfwreset de NVIDIA](#)" à pour obtenir des conseils. Si un redémarrage est nécessaire, effectuez un redémarrage au lieu de réinitialiser les adaptateurs.

a. Arrêter le service openhm :

```
systemctl stop opensm
```

b. Exécutez la commande suivante pour chacune des PCI Device Name opérations précédemment notées.

```
mlxfwreset -d <pci_device_name> reset -y
```

c. Démarrer le service openhm :

```
systemctl start opensm
```

7. Exécutez ibstat et vérifiez que toutes les cartes fonctionnent avec la version de micrologiciel souhaitée :

```
ibstat
```

8. Démarrer les services de cluster Pacemaker sur le nœud :

```
pcs cluster start <HOSTNAME>
```

9. Mettre le nœud hors veille :

```
pcs node unstandby <HOSTNAME>
```

10. Sortir le cluster du mode de maintenance.

```
pcs property set maintenance-mode=false
```

11. Retransférez tous les services BeeGFS vers le nœud de votre choix :

```
pcs resource relocate run
```

Répétez ces étapes pour chaque nœud de fichiers du cluster jusqu'à ce que tous les adaptateurs aient été mis à jour.

Mettez à niveau la baie de stockage E-Series

Suivez ces étapes pour mettre à niveau les composants de la baie de stockage E-Series du cluster HA.

Présentation

En conservant les baies de stockage NetApp E-Series de votre cluster de haute disponibilité à jour avec le dernier firmware, vous bénéficiez de performances optimales et d'une sécurité renforcée. Les mises à jour du micrologiciel de la baie de stockage sont appliquées via le système d'exploitation SANtricity, la NVSRAM et les fichiers de micrologiciel du lecteur.

 Bien que les baies de stockage puissent être mises à niveau avec le cluster haute disponibilité en ligne, il est recommandé de placer le cluster en mode de maintenance pour toutes les mises à niveau.

Étapes de mise à niveau du nœud de bloc

Les étapes suivantes expliquent comment mettre à jour le firmware des baies de stockage à l'aide de la Netapp_Eseries_Santricity collection Ansible. Avant de continuer, consultez le "[Mise à niveau](#)" pour la mise à jour des systèmes E-Series.

 La mise à niveau vers SANtricity OS 11.80 ou versions ultérieures est possible uniquement à partir de 11.70.5P1. La baie de stockage doit d'abord être mise à niveau vers 11.70.5P1 avant d'appliquer d'autres mises à niveau.

1. Vérifiez que votre nœud de contrôle Ansible utilise la dernière collection SANtricity Ansible.
 - Pour les mises à niveau de collecte avec accès à "[Galaxy Ansible](#)", exécutez la commande suivante :

```
ansible-galaxy collection install netapp_eseries.santricity --upgrade
```

- Pour les mises à niveau hors ligne, téléchargez le fichier tarball de "[Galaxy Ansible](#)" la collection à partir de , transférez-le vers votre nœud de contrôle et exécutez :

```
ansible-galaxy collection install netapp_eseries-santricity-  
<VERSION>.tar.gz --upgrade
```

Voir "[Installation de Collections](#)" pour en savoir plus.

2. Obtenez la dernière version du micrologiciel pour votre matrice de stockage et vos lecteurs.
 - a. Téléchargez les fichiers du micrologiciel.
 - **SANtricity OS et NVSRAM** : naviguez jusqu'au "[Site de support NetApp](#)" et téléchargez la dernière version de SANtricity OS et NVSRAM pour votre modèle de matrice de stockage.
 - **Microprogramme de lecteur** : naviguez jusqu'au "[Site du firmware du disque E-Series](#)" et téléchargez le dernier micrologiciel pour chacun des modèles de lecteur de votre matrice de stockage.
 - b. Stockez les fichiers du système d'exploitation SANtricity, de la NVSRAM et du firmware des disques dans le <inventory_directory>/packages répertoire du nœud de contrôle Ansible.
3. Si nécessaire, mettez à jour les fichiers d'inventaire Ansible de votre cluster afin d'inclure toutes les baies de stockage (nœuds de bloc) nécessitant des mises à jour. Pour obtenir des conseils, voir "[Présentation d'Ansible Inventory](#)" la section.
4. Assurez-vous que le cluster est optimal avec chaque service BeeGFS sur le nœud de votre choix. Voir "[Vérifiez l'état du cluster](#)" pour plus de détails.
5. Placez le cluster en mode maintenance en suivant les instructions de "[Placer le cluster en mode maintenance](#)" la section .
6. Créez un nouveau PlayBook Ansible nommé update_block_node_playbook.yml. Remplissez le manuel avec le contenu suivant en remplaçant le système d'exploitation SANtricity, la NVSRAM et les versions de firmware des disques par le chemin de mise à niveau souhaité :

```

- hosts: eseries_storage_systems
  gather_facts: false
  any_errors_fatal: true
  collections:
    - netapp_eseries.santricity
  vars:
    eseries_firmware_firmware: "packages/<SantricityOS>.dlp"
    eseries_firmware_nvram: "packages/<NVSRAM>.dlp"
    eseries_drive_firmware_firmware_list:
      - "packages/<drive_firmware>.dlp"
    eseries_drive_firmware_upgrade_drives_online: true

  tasks:
    - name: Configure NetApp E-Series block nodes.
      import_role:
        name: nar_santricity_management

```

7. Pour démarrer les mises à jour, exécutez la commande suivante à partir de votre nœud de contrôle Ansible :

```
ansible-playbook -i inventory.yml update_block_node_playbook.yml
```

8. Une fois le manuel de vente terminé, vérifiez que chaque baie de stockage est dans un état optimal.
9. Déplacez le cluster hors du mode de maintenance et vérifiez qu'il est dans un état optimal, chaque service

BeeGFS étant sur le nœud privilégié.

Entretien et maintenance

Services de basculement/rétablissement

Déplacement des services BeeGFS entre les nœuds du cluster.

Présentation

Les services BeeGFS peuvent basculer entre les nœuds du cluster pour s'assurer que les clients sont en mesure de continuer à accéder au système de fichiers en cas de défaillance d'un nœud ou si vous devez effectuer une maintenance planifiée. Cette section décrit différentes méthodes permettant aux administrateurs d'effectuer une réparation sur le cluster après une reprise d'activité ou de déplacer manuellement les services entre les nœuds.

Étapes

Basculement et rétablissement

Basculement (planifié)

De manière générale, lorsque vous devez mettre un nœud de fichier unique hors ligne pour les opérations de maintenance, vous devez déplacer (ou vidanger) tous les services BeeGFS depuis ce nœud. Pour ce faire, le nœud peut d'abord être en veille :

```
pcs node standby <HOSTNAME>
```

Après vérification de l'utilisation `pcs status` toutes les ressources ont été redémarrées sur le nœud de fichier secondaire, vous pouvez arrêter ou apporter d'autres modifications au nœud si nécessaire.

Restauration (après un basculement planifié)

Lorsque vous êtes prêt à restaurer les services BeeGFS sur le nœud préféré s'exécutent d'abord `pcs status` Et vérifiez dans la « liste de nœuds » que l'état est en veille. Si le nœud a été redémarré, il s'affiche hors ligne jusqu'à ce que vous mettent les services du cluster en ligne :

```
pcs cluster start <HOSTNAME>
```

Une fois le nœud mis en ligne hors veille, grâce à :

```
pcs node unstandby <HOSTNAME>
```

Enfin, transférez tous les services BeeGFS vers leurs nœuds préférés avec :

```
pcs resource relocate run
```

Retour arrière (après basculement non planifié)

Si un nœud présente un défaut matériel ou autre, le cluster haute disponibilité doit réagir automatiquement et déplacer ses services vers un nœud sain, ce qui permet aux administrateurs de prendre des mesures correctives. Avant de continuer, reportez-vous à ["dépannage"](#) la section pour déterminer la cause du basculement et résoudre tout problème en suspens. Une fois le nœud mis sous tension et en bon état, vous pouvez continuer à le restaurer.

Lorsqu'un nœud démarre après un redémarrage non planifié (ou planifié), les services de cluster ne sont pas configurés pour démarrer automatiquement. Vous devez donc mettre le nœud en ligne avec :

```
pcs cluster start <HOSTNAME>
```

Ensuite, nettoyez toute défaillance de ressource et réinitialisez l'historique d'escrime du nœud :

```
pcs resource cleanup node=<HOSTNAME>
pcs stonith history cleanup <HOSTNAME>
```

Vérifier dans `pcs status` le nœud est en ligne et fonctionne correctement. Par défaut, les services BeeGFS ne sont pas automatiquement rebasculer afin d'éviter tout déplacement accidentel des ressources vers un nœud malsain. Une fois que vous êtes prêt à renvoyer toutes les ressources du cluster à leurs nœuds préférés avec :

```
pcs resource relocate run
```

Déplacement de services BeeGFS individuels vers d'autres nœuds de fichiers

Déplacer définitivement un service BeeGFS vers un nouveau noeud de fichier

Si vous souhaitez modifier de manière permanente le nœud de fichier favori pour un service BeeGFS, ajustez l'inventaire Ansible de sorte que le nœud préféré soit répertorié en premier et exécutez à nouveau le PlayBook Ansible.

Par exemple, dans cet exemple de `inventory.yml` fichier, `beegfs_01` est le nœud de fichiers préféré pour exécuter le service de gestion BeeGFS :

```
mgmt:
  hosts:
    beegfs_01:
    beegfs_02:
```

Inverser l'ordre ferait que les services de gestion seraient préférés sur `beegfs_02`:

```
mgmt:  
  hosts:  
    beegfs_02:  
    beegfs_01:
```

Déplacer temporairement un service BeeGFS vers un autre nœud de fichier

De manière générale, si un nœud est en cours de maintenance, il convient d'utiliser [les étapes de basculement et de retour arrière](#le basculement et la restauration) pour déplacer tous les services hors de ce nœud.

Si vous devez déplacer un service individuel vers un autre nœud de fichiers :

```
pcs resource move <SERVICE>-monitor <HOSTNAME>
```

Ne spécifiez pas les ressources individuelles ou le groupe de ressources. Spécifiez toujours le nom du moniteur pour le service BeeGFS que vous souhaitez déplacer. Par exemple, pour déplacer le service de gestion BeeGFS vers beegfs_02, exécutez : `pcs resource move mgmt-monitor beegfs_02`. Ce processus peut être répété afin de déplacer un ou plusieurs services hors de leurs nœuds préférés. Vérifiez à l'aide des `pcs status services` qui ont été déplacés/démarrés sur le nouveau nœud.

Pour déplacer un service BeeGFS vers son nœud préféré, effacez d'abord les contraintes de ressources temporaires (en répétant cette étape comme nécessaire pour plusieurs services) :

```
pcs resource clear <SERVICE>-monitor
```

Ensuite, une fois prêt à rapatrier les services sur les nœuds de leur choix :

```
pcs resource relocate run
```

Notez que cette commande permet de transférer tous les services qui ne disposent plus de contraintes temporaires en termes de ressources, situés sur les nœuds de leur choix.

Placer le cluster en mode maintenance

Empêcher le cluster de haute disponibilité de réagir accidentellement aux changements prévus dans l'environnement.

Présentation

Le fait de mettre le cluster en mode maintenance désactive toute la surveillance des ressources et empêche Pacemaker de déplacer ou de gérer des ressources dans le cluster. Toutes les ressources restent exécutées sur leurs nœuds d'origine, peu importe la condition de panne temporaire qui empêcherait leur accès. Voici quelques scénarios recommandés/utiles :

- Maintenance du réseau pouvant interrompre temporairement les connexions entre les nœuds de fichiers et les services BeeGFS.
- Mises à niveau des nœuds de blocs.
- Mises à jour du système d'exploitation de nœud de fichiers, du noyau ou d'autres modules.

En général, la seule raison de placer manuellement le cluster en mode de maintenance est d'éviter que le système ne réagisse à des modifications externes de l'environnement. Si un nœud individuel du cluster nécessite une réparation physique, n'utilisez pas le mode de maintenance et placez simplement ce nœud en veille après la procédure ci-dessus. Notez que le changement d'Ansible place automatiquement le cluster en mode de maintenance pour faciliter la plupart des opérations de maintenance logicielle, y compris les mises à niveau et les modifications de configuration.

Étapes

Pour vérifier si le cluster est en mode maintenance, exécutez :

```
pcs property config
```

La `maintenance-mode` propriété n'apparaît pas si le cluster fonctionne normalement. Si le cluster est actuellement en mode maintenance, la propriété indique `true`. Pour activer le mode maintenance, exécutez :

```
pcs property set maintenance-mode=true
```

Vous pouvez vérifier en exécutant l'état `pcs` et en vous assurant que toutes les ressources affichent « (non géré) ». Pour mettre le cluster hors mode maintenance, exécutez :

```
pcs property set maintenance-mode=false
```

Arrêtez et démarrez le cluster

Arrêt et démarrage du cluster HA avec élégance.

Présentation

Cette section décrit comment arrêter et redémarrer le cluster BeeGFS. Par exemple, la maintenance électrique ou la migration d'un data Center à l'autre ou d'un rack peut être nécessaire.

Étapes

Si, pour une raison quelconque, vous devez arrêter tout le cluster BeeGFS et arrêter tous les services exécutent :

```
pcs cluster stop --all
```

Il est également possible d'arrêter le cluster sur des nœuds individuels (qui basculeront automatiquement les services vers un autre nœud), bien qu'il soit d'abord recommandé de mettre le nœud en veille (voir la

"basculement" section) :

```
pcs cluster stop <HOSTNAME>
```

Pour démarrer les ressources et les services du cluster sur tous les nœuds, exécutez :

```
pcs cluster start --all
```

Ou démarrer les services sur un nœud spécifique avec :

```
pcs cluster start <HOSTNAME>
```

À ce stade, exécutez `pcs status` Et vérifiez que le cluster et les services BeeGFS démarrent sur tous les nœuds et que les services sont exécutés sur les nœuds que vous attendez.

Selon la taille du cluster, l'arrêt de l'ensemble du cluster peut prendre des secondes ou des minutes, ou s'afficher comme démarré dans `pcs status`. Si `pcs cluster <COMMAND>` se bloque pendant plus de cinq minutes, avant d'exécuter « Ctrl+C » pour annuler la commande, connectez-vous à chaque nœud du cluster et utilisez `pcs status` pour voir si les services de cluster (Corosync/Pacemaker) sont toujours en cours d'exécution sur ce nœud. A partir de n'importe quel nœud où le cluster est toujours actif, vous pouvez vérifier les ressources qui bloquent le cluster. Résoudre manuellement le problème et la commande doit être terminée ou réexécutée pour arrêter les services restants.

Remplacer les nœuds de fichiers

Remplacement d'un noeud de fichier si le serveur d'origine est défectueux.

Présentation

Voici un aperçu des étapes nécessaires au remplacement d'un noeud de fichier dans le cluster. Ces étapes presupposent que le nœud de fichier a échoué en raison d'un problème matériel et a été remplacé par un nouveau nœud de fichier identique.

Étapes :

1. Remplacez physiquement le nœud de fichiers et restaurez tout le câblage vers le nœud de bloc et le réseau de stockage.
2. Réinstallez le système d'exploitation sur le nœud de fichier, y compris l'ajout d'abonnements Red Hat.
3. Configurez la mise en réseau BMC et la gestion sur le nœud de fichiers.
4. Mettez à jour l'inventaire Ansible si le nom d'hôte, l'IP, les mappages de l'interface PCIe vers l'interface logique ou tout autre élément modifié concernant le nouveau nœud de fichier. En général, cette opération n'est pas nécessaire si le nœud a été remplacé par le même matériel serveur et que vous utilisez la configuration réseau d'origine.
 - a. Par exemple, si le nom d'hôte a changé, créez (ou renommez) le fichier d'inventaire du nœud (`host_vars/<NEW_NODE>.yml`) Puis dans le fichier d'inventaire Ansible (`inventory.yml`),

remplacer le nom de l'ancien nœud par le nouveau nom de nœud :

```
all:
  ...
  children:
    ha_cluster:
      children:
        mgmt:
          hosts:
            node_h1_new:    # Replaced "node_h1" with "node_h1_new"
            node_h2:
```

5. Depuis un des autres nœuds du cluster, supprimer l'ancien nœud : `pcs cluster node remove <HOSTNAME>`.



NE PAS POURSUIVRE AVANT D'EXÉCUTER CETTE ÉTAPE.

6. Sur le nœud de contrôle Ansible :

- Supprimez l'ancienne clé SSH avec :

```
`ssh-keygen -R <HOSTNAME_OR_IP>`
```

- Configurez SSH sans mot de passe sur le nœud remplacer par :

```
ssh-copy-id <USER>@<HOSTNAME_OR_IP>
```

7. Exécutez à nouveau le PlayBook Ansible pour configurer le nœud et l'ajouter au cluster :

```
ansible-playbook -i <inventory>.yml <playbook>.yml
```

8. A ce stade, exécuter `pcs status` et vérifiez que le nœud remplacé est maintenant répertorié et que les services sont en cours d'exécution.

Développez ou réduisez le cluster

Ajouter ou supprimer des éléments de base du cluster.

Présentation

Cette section présente divers éléments à prendre en compte et diverses options pour ajuster la taille de votre cluster BeeGFS HA. La taille du cluster est généralement ajustée en ajoutant ou en supprimant des éléments de base, qui sont généralement deux nœuds de fichiers configurés comme une paire haute disponibilité. Il est également possible d'ajouter ou de supprimer des nœuds de fichiers individuels (ou d'autres types de nœuds de cluster) si nécessaire.

Ajout d'un module au cluster

Considérations

Le développement du cluster par l'ajout d'éléments de base supplémentaires est un processus simple. Avant de commencer, les restrictions s'imposent concernant le nombre minimal et maximal de nœuds de cluster dans chaque cluster haute disponibilité. Déterminer si vous devez ajouter des nœuds au cluster haute disponibilité existant ou créer un nouveau cluster haute disponibilité. En général, chaque élément de base se compose de deux nœuds de fichiers, mais trois nœuds représentent le nombre minimum de nœuds par cluster (pour établir le quorum) et dix est le nombre maximum recommandé (testé). Pour les scénarios avancés, il est possible d'ajouter un nœud « Tiebreaker » unique qui n'exécute aucun service BeeGFS lors du déploiement d'un cluster à deux nœuds. Si vous envisagez un tel déploiement, contactez le support NetApp.

Gardez à l'esprit ces restrictions et toute future croissance des clusters lorsque vous décidez d'étendre le cluster. Par exemple, si vous disposez d'un cluster à six nœuds et que vous devez en ajouter quatre autres, il est recommandé de simplement démarrer un nouveau cluster haute disponibilité.

 N'oubliez pas qu'un seul système de fichiers BeeGFS peut consister en plusieurs clusters HA indépendants. Les systèmes de fichiers peuvent ainsi continuer à évoluer au-delà des limites recommandées/strictes des composants de cluster haute disponibilité sous-jacents.

Étapes

Lorsque vous ajoutez un élément de base à votre cluster, vous devez créer les `host_vars` fichiers pour chacun des nouveaux nœuds de fichiers et nœuds de blocs (baies E-Series). Les noms de ces hôtes doivent être ajoutés à l'inventaire, ainsi que les nouvelles ressources à créer. Les `group_vars` fichiers correspondants devront être créés pour chaque nouvelle ressource. Voir la "[utilisez des architectures personnalisées](#)" section pour plus de détails.

Une fois les fichiers corrects créés, il suffit de relancer l'automatisation à l'aide de la commande :

```
ansible-playbook -i <inventory>.yml <playbook>.yml
```

Retrait d'un module du cluster

Il y a plusieurs considérations à garder à l'esprit lorsque vous devez retirer un élément de construction, par exemple :

- Quels sont les services BeeGFS exécutés dans cet élément de base ?
- Les nœuds de fichiers ne sont-ils que ceux qui sont mis hors service et ceux qui doivent être associés à de nouveaux nœuds de fichiers ?
- Si l'ensemble du bloc de construction est retiré, les données doivent-elles être déplacées vers un nouveau bloc de construction, dispersées vers les nœuds existants du cluster ou déplacées vers un nouveau système de fichiers BeeGFS ou un autre système de stockage ?
- Cela peut-il avoir lieu en cas de panne ou doit-il être effectué sans interruption ?
- L'élément de base est-il activement utilisé ou contient-il principalement des données qui ne sont plus actives ?

Étant donné la diversité des points de départ et des États de terminaison, veuillez contacter le support NetApp afin que nous puissions identifier et vous aider à mettre en œuvre la meilleure stratégie en fonction de votre environnement et de vos besoins.

Résoudre les problèmes

Dépannage d'un cluster BeeGFS HA

Présentation

Dans cette section, vous apprendrez à rechercher et à dépanner diverses défaillances et d'autres scénarios possibles liés à l'utilisation d'un cluster BeeGFS HA.

Guides de dépannage

Étude des basculements inattendus

Lorsqu'un nœud est fermé de façon inattendue et que ses services sont déplacés vers un autre nœud, la première étape doit s'assurer que le cluster indique des défaillances de ressource en bas du `pcs status`. En général, rien ne sera présent si l'escrime s'est terminé avec succès et que les ressources ont été redémarrées sur un autre noeud.

Généralement, l'étape suivante consiste à rechercher dans les journaux système à l'aide de `journalctl`. Sur l'un des nœuds de fichiers restants (les journaux Pacemaker sont synchronisés sur tous les nœuds). Si vous connaissez l'heure de l'échec, vous pouvez lancer la recherche juste avant l'échec (généralement au moins dix minutes avant l'apparition de l'échec est recommandée) :

```
journalctl --since "<YYYY-MM-DD HH:MM:SS>"
```

Les sections suivantes montrent un texte commun que vous pouvez gresser dans les journaux pour affiner davantage l'enquête.

Étapes à suivre pour rechercher/résoudre

Étape 1 : vérifier si le moniteur BeeGFS a détecté une défaillance :

Si le basculement a été déclenché par le moniteur BeeGFS, une erreur s'affiche (si ce n'est pas le cas, passez à l'étape suivante).

```
journalctl --since "<YYYY-MM-DD HH:MM:SS>" | grep -i unexpected
[...]
Jul 01 15:51:03 beegfs_01 pacemaker-schedulerd[9246]: warning: Unexpected
result (error: BeeGFS service is not active!) was recorded for monitor of
meta_08-monitor on beegfs_02 at Jul 1 15:51:03 2022
```

Dans cet exemple, BeeGFS service META_08 s'est arrêté pour une raison quelconque. Pour poursuivre le dépannage, nous devons démarrer `beegfs_02` et consulter les journaux du service à l'adresse `/var/log/beegfs-meta-meta_08_tgt_0801.log`. Par exemple, le service BeeGFS peut avoir rencontré une erreur d'application en raison d'un problème interne ou d'un problème sur le nœud.



Contrairement aux logs de Pacemaker, les logs des services BeeGFS ne sont pas distribués à tous les nœuds du cluster. Pour examiner ces types de défaillances, les journaux du nœud d'origine où la défaillance est requise.

Les problèmes possibles pouvant être signalés par le moniteur sont les suivants :

- Les cibles ne sont pas accessibles !
 - Description : indique que les volumes de bloc n'ont pas été accessibles.
 - Dépannage :
 - Si le service n'a pas non plus démarré sur le nœud de fichier secondaire, confirmez que le nœud de bloc fonctionne correctement.
 - Vérifiez si des problèmes physiques empêchent l'accès aux nœuds de blocs depuis ce nœud de fichiers, par exemple des adaptateurs ou des câbles InfiniBand défectueux.
- Le réseau est inaccessible !
 - Description : aucun des adaptateurs utilisés par les clients pour se connecter à ce service BeeGFS n'était en ligne.
 - Dépannage :
 - Si plusieurs ou tous les nœuds de fichiers sont affectés, vérifiez s'il y a une défaillance sur le réseau utilisée pour connecter les clients BeeGFS et le système de fichiers.
 - Recherchez les problèmes physiques susceptibles d'empêcher l'accès des clients à partir de ce nœud de fichiers, par exemple des adaptateurs ou des câbles InfiniBand défectueux.
- Le service BeeGFS n'est pas actif!
 - Description : un service BeeGFS s'est arrêté de façon inattendue.
 - Dépannage :
 - Sur le nœud de fichier qui signale l'erreur, vérifiez les journaux du service eGFS impacté pour voir s'il signale une panne. Dans ce cas, ouvrez un dossier auprès du support NetApp afin que le problème puisse être examiné.
 - Si aucune erreur n'est signalée dans le journal BeeGFS, vérifiez les journaux du journal pour voir si systemd a enregistré une raison pour laquelle le service a été arrêté. Dans certains scénarios, le service BeeGFS n'a peut-être pas été donné la possibilité de consigner tous les messages avant la fin du processus (par exemple si quelqu'un a exécuté `kill -9 <PID>`).

Étape 2 : vérifiez si le nœud a quitté le cluster de manière inattendue

Si le nœud a subi une défaillance matérielle catastrophique (par exemple, la carte système est morte) ou s'il y avait une panique du noyau ou un problème logiciel similaire, le moniteur BeeGFS ne signale pas d'erreur. Au lieu de cela, recherchez le nom d'hôte et les messages de Pacemaker indiquant que le nœud a été perdu de façon inattendue :

```
journalctl --since "<YYYY-MM-DD HH:MM:SS>" | grep -i <HOSTNAME>
[...]
Jul 01 16:18:01 beegfs_01 pacemaker-attrd[9245]: notice: Node beegfs_02
state is now lost
Jul 01 16:18:01 beegfs_01 pacemaker-controld[9247]: warning:
Stonith/shutdown of node beegfs_02 was not expected
```

Étape 3 : vérifier que Pacemaker a pu verrouiller le nœud

Dans tous les scénarios, Pacemaker tente de limiter le nœud pour vérifier qu'il est réellement hors ligne (les messages exacts peuvent varier en fonction de la cause de l'escrime) :

```
Jul 01 16:18:02 beegfs_01 pacemaker-schedulerd[9246]: warning: Cluster
node beegfs_02 will be fenced: peer is no longer part of the cluster
Jul 01 16:18:02 beegfs_01 pacemaker-schedulerd[9246]: warning: Node
beegfs_02 is unclean
Jul 01 16:18:02 beegfs_01 pacemaker-schedulerd[9246]: warning: Scheduling
Node beegfs_02 for STONITH
```

Si l'action de clôture s'effectue correctement, des messages comme :

```
Jul 01 16:18:14 beegfs_01 pacemaker-fenced[9243]: notice: Operation 'off'
[2214070] (call 27 from pacemaker-controld.9247) for host 'beegfs_02' with
device 'fence_redfish_2' returned: 0 (OK)
Jul 01 16:18:14 beegfs_01 pacemaker-fenced[9243]: notice: Operation 'off'
targeting beegfs_02 on beegfs_01 for pacemaker-
controld.9247@beegfs_01.786df3a1: OK
Jul 01 16:18:14 beegfs_01 pacemaker-controld[9247]: notice: Peer
beegfs_02 was terminated (off) by beegfs_01 on behalf of pacemaker-
controld.9247: OK
```

Si l'action d'escrime a échoué pour une raison quelconque, les services BeeGFS ne pourront pas redémarrer sur un autre nœud pour éviter la corruption des données. Ce serait un problème à étudier séparément si, par exemple, le dispositif d'escrime (PDU ou BMC) était inaccessible ou mal configuré.

Echec des actions de ressource de l'adresse (en bas de l'état pcs)

Si une ressource requise pour exécuter un service BeeGFS échoue, un basculement est déclenché par le moniteur BeeGFS. Si cela se produit, il est probable qu'aucune « action de ressource ayant échoué » ne soit répertoriée `pcs status` en bas de et vous devez vous reporter aux étapes à suivre "[retour arrière après un basculement non planifié](#)" pour savoir comment .

Dans le cas contraire, il ne devrait y avoir que deux scénarios où vous verrez des « actions de ressource échouées ».

Étapes à suivre pour rechercher/résoudre

Scénario 1 : un problème temporaire ou permanent a été détecté avec un agent d'escrime et il a été redémarré ou déplacé vers un autre nœud.

Certains agents d'escrime sont plus fiables que d'autres et chacun mettra en œuvre sa propre méthode de surveillance pour s'assurer que le dispositif d'escrime est prêt. En particulier, l'agent d'escrime de Redfish a été vu pour signaler des actions de ressources échouées comme les suivantes, même s'il se présente toujours commencé :

```
* fence_redfish_2_monitor_60000 on beegfs_01 'not running' (7):  
call=2248, status='complete', exitreason='', last-rc-change='2022-07-26  
08:12:59 -05:00', queued=0ms, exec=0ms
```

Un agent d'escrime signalant l'échec des actions de ressources sur un nœud particulier ne devrait pas déclencher un basculement des services BeeGFS s'exécutant sur ce nœud. Il devrait simplement être redémarré automatiquement sur le même nœud ou sur un autre nœud.

Étapes à suivre pour résoudre :

1. Si l'agent d'escrime refuse systématiquement de s'exécuter sur tout ou sous-ensemble de nœuds, vérifiez si ces nœuds peuvent se connecter à l'agent d'escrime et vérifiez que l'agent d'escrime est configuré correctement dans l'inventaire Ansible.
 - a. Par exemple, si un agent d'escrime Redfish (BMC) s'exécute sur le même nœud qu'il est responsable de l'escrime, et que la gestion du système d'exploitation et les adresses IP BMC sont sur la même interface physique, certaines configurations de commutateurs réseau ne permettent pas la communication entre les deux interfaces (pour éviter les boucles réseau). Par défaut, le cluster HA tente d'éviter de placer des agents d'escrime sur le nœud qu'ils sont responsables de l'escrime, mais cela peut se produire dans certains scénarios/configurations.
2. Une fois tous les problèmes résolus (ou si le problème semble éphémère), exécutez `pcs resource cleanup` pour réinitialiser les actions de ressources ayant échoué.

Scénario 2 : le moniteur BeeGFS a détecté un problème et déclenché un basculement, mais pour une raison quelconque, les ressources ne peuvent pas démarrer sur un nœud secondaire.

Si l'escrime est activé et que la ressource n'a pas été bloquée pour s'arrêter sur le nœud d'origine (voir la section de dépannage pour « attente (en cas d'échec) »), les raisons les plus probables incluent des problèmes de démarrage de la ressource sur un nœud secondaire car :

- Le nœud secondaire était déjà hors ligne.
- Un problème de configuration physique ou logique a empêché le système secondaire d'accéder aux volumes de bloc utilisés comme cibles BeeGFS.

Étapes à suivre pour résoudre :

1. Pour chaque entrée des actions de ressources ayant échoué :
 - a. Confirmez que l'action de ressource échouée était une opération de démarrage.
 - b. En fonction de la ressource indiquée et du nœud spécifié dans les actions de ressources ayant échoué :
 - i. Recherchez et corrigez tout problème externe qui empêche le nœud de démarrer la ressource

spécifiée. Par exemple, si l'adresse IP BeeGFS (IP flottante) n'a pas démarré, vérifiez qu'au moins une des interfaces requises est connectée/en ligne et câblée au commutateur réseau approprié. Si une cible BeeGFS (périphérique de bloc/volume E-Series) est défectueuse, vérifiez que les connexions physiques vers le(s) nœud(s) du bloc principal sont connectées comme prévu, et vérifiez que les nœuds du bloc sont en bon état.

- c. Si aucun problème externe n'est évident et que vous souhaitez en savoir plus sur la cause première, nous vous recommandons d'ouvrir un dossier auprès des services de support de NetApp avant de poursuivre, car les étapes suivantes peuvent compliquer ou empêcher l'analyse des causes profondes (RCA).

2. Après la résolution de tout problème externe :

- a. Commentez tous les nœuds non fonctionnels à partir du fichier Ansible Inventory.yml et exécutez à nouveau le PlayBook Ansible complet pour vous assurer que toute la configuration logique est correctement configurée sur le ou les nœuds secondaires.
 - i. Remarque : n'oubliez pas d'annuler la commentaire de ces nœuds et d'exécuter à nouveau le manuel de vente une fois les nœuds sains et vous êtes prêt à revenir en arrière.
- b. Vous pouvez également tenter de restaurer manuellement le cluster :
 - i. Remettre en ligne tous les nœuds en utilisant : `pcs cluster start <HOSTNAME>`
 - ii. Effacer toutes les actions de ressources ayant échoué à l'aide de : `pcs resource cleanup`
 - iii. Exécutez l'état `pcs` et vérifiez que tous les services commencent comme prévu.
 - iv. Si nécessaire, exécutez `pcs resource relocate run` pour renvoyer les ressources vers le nœud de votre choix (s'il est disponible).

Problèmes courants

Les services BeeGFS ne sont pas de basculement ou de retour arrière sur demande

Question probable: le `pcs resource relocate` la commande d'exécution a été exécutée mais n'a jamais réussi.

Comment vérifier : Exécuter `pcs constraint --full` Et recherchez les contraintes d'emplacement avec un ID de `pcs-relocate-<RESOURCE>`.

Comment résoudre : Exécuter `pcs resource relocate clear` puis repassage `pcs constraint --full` pour vérifier que les contraintes supplémentaires sont supprimées.

Un nœud dans l'état `pcs` affiche "attente (on-fail)" lorsque l'escrime est désactivé

Problème probable : Pacemaker n'a pas pu confirmer avec succès que toutes les ressources ont été arrêtées sur le nœud qui a échoué.

Comment résoudre:

1. Courez `pcs status` enfin, recherchez les ressources qui ne sont pas « démarrées » et affichez les erreurs en bas de la page et résolvez les problèmes.
2. Pour rétablir l'exécution en ligne du nœud `pcs resource cleanup --node=<HOSTNAME>`.

Après un basculement inattendu, les ressources indiquent « Started (on-fail) » (démarré (on-fail)) dans l'état pcs (pcs) lorsque l'escrime est activé

Problème probable : Un problème s'est produit qui a déclenché un basculement, mais Pacemaker n'a pas pu vérifier que le nœud était clôturé. Cela pourrait se produire parce que l'escrime était mal configuré ou qu'il y avait un problème avec l'agent d'escrime (par exemple : l'unité de distribution d'alimentation était déconnectée du réseau).

Comment résoudre:

1. Vérifiez que le nœud est réellement hors tension.



Si le nœud que vous spécifiez n'est pas réellement arrêté, mais que vous exécutez les services ou les ressources du cluster, une corruption des données ou une défaillance du cluster se produit.

2. Confirmer manuellement l'escrime avec : `pcs stonith confirm <NODE>`

À ce stade, les services devraient finir le basculement et être redémarrés sur un autre noeud en bon état.

Tâches courantes de dépannage

Redémarrez chaque service BeeGFS

Normalement, si un service BeeGFS doit être redémarré (par exemple pour faciliter une modification de la configuration), il doit être fait en mettant à jour l'inventaire Ansible et en exécutant de nouveau le manuel de vente. Dans certains cas, il peut être souhaitable de redémarrer des services individuels pour accélérer le dépannage, par exemple pour modifier le niveau de journalisation sans avoir à attendre l'exécution du manuel de vente dans son intégralité.



Sauf si des modifications manuelles sont également ajoutées à l'inventaire Ansible, elles seront rétablies au prochain exécution du PlayBook Ansible.

Option 1 : redémarrage contrôlé par le système

S'il y a un risque que le service BeeGFS ne redémarre pas correctement avec la nouvelle configuration, tout d'abord placer le cluster en mode maintenance pour empêcher le moniteur BeeGFS de détecter le service est arrêté et déclencher un basculement non souhaité :

```
pcs property set maintenance-mode=true
```

Si nécessaire, modifiez la configuration des services à l'adresse `/mnt/<SERVICE_ID>/_config/beegfs-.conf` (exemple : `/mnt/meta_01_tgt_0101/metadata_config/beegfs-meta.conf`) puis utilisez `systemd` pour le redémarrer :

```
systemctl restart beegfs-*@<SERVICE_ID>.service
```

Exemple : `systemctl restart beegfs-meta@meta_01_tgt_0101.service`

Option 2 : redémarrage contrôlé par le stimulateur cardiaque

Si vous n'êtes pas préoccupé par la nouvelle configuration, le service peut s'arrêter de façon inattendue (par exemple, en modifiant simplement le niveau de journalisation), ou vous êtes dans une fenêtre de maintenance et ne vous préoccupez pas des temps d'arrêt, il vous suffit de redémarrer le moniteur BeeGFS pour le service que vous voulez redémarrer :

```
pcs resource restart <SERVICE>-monitor
```

Par exemple, pour redémarrer le service de gestion BeeGFS : `pcs resource restart mgmt-monitor`

Informations sur le copyright

Copyright © 2026 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

Les logiciels dérivés des éléments NetApp protégés par copyright sont soumis à la licence et à l'avis de non-responsabilité suivants :

CE LOGICIEL EST FOURNI PAR NETAPP « EN L'ÉTAT » ET SANS GARANTIES EXPRESSES OU TACITES, Y COMPRIS LES GARANTIES TACITES DE QUALITÉ MARCHANDE ET D'ADÉQUATION À UN USAGE PARTICULIER, QUI SONT EXCLUES PAR LES PRÉSENTES. EN AUCUN CAS NETAPP NE SERA TENU POUR RESPONSABLE DE DOMMAGES DIRECTS, INDIRECTS, ACCESSOIRES, PARTICULIERS OU EXEMPLAIRES (Y COMPRIS L'ACHAT DE BIENS ET DE SERVICES DE SUBSTITUTION, LA PERTE DE JOUSSANCE, DE DONNÉES OU DE PROFITS, OU L'INTERRUPTION D'ACTIVITÉ), QUELLES QU'EN SOIENT LA CAUSE ET LA DOCTRINE DE RESPONSABILITÉ, QU'IL S'AGISSE DE RESPONSABILITÉ CONTRACTUELLE, STRICTE OU DÉLICTUELLE (Y COMPRIS LA NÉGLIGENCE OU AUTRE) DÉCOULANT DE L'UTILISATION DE CE LOGICIEL, MÊME SI LA SOCIÉTÉ A ÉTÉ INFORMÉE DE LA POSSIBILITÉ DE TELS DOMMAGES.

NetApp se réserve le droit de modifier les produits décrits dans le présent document à tout moment et sans préavis. NetApp décline toute responsabilité découlant de l'utilisation des produits décrits dans le présent document, sauf accord explicite écrit de NetApp. L'utilisation ou l'achat de ce produit ne concède pas de licence dans le cadre de droits de brevet, de droits de marque commerciale ou de tout autre droit de propriété intellectuelle de NetApp.

Le produit décrit dans ce manuel peut être protégé par un ou plusieurs brevets américains, étrangers ou par une demande en attente.

LÉGENDE DE RESTRICTION DES DROITS : L'utilisation, la duplication ou la divulgation par le gouvernement sont sujettes aux restrictions énoncées dans le sous-paragraphe (b)(3) de la clause Rights in Technical Data-Noncommercial Items du DFARS 252.227-7013 (février 2014) et du FAR 52.227-19 (décembre 2007).

Les données contenues dans les présentes se rapportent à un produit et/ou service commercial (tel que défini par la clause FAR 2.101). Il s'agit de données propriétaires de NetApp, Inc. Toutes les données techniques et tous les logiciels fournis par NetApp en vertu du présent Accord sont à caractère commercial et ont été exclusivement développés à l'aide de fonds privés. Le gouvernement des États-Unis dispose d'une licence limitée irrévocable, non exclusive, non cessible, non transférable et mondiale. Cette licence lui permet d'utiliser uniquement les données relatives au contrat du gouvernement des États-Unis d'après lequel les données lui ont été fournies ou celles qui sont nécessaires à son exécution. Sauf dispositions contraires énoncées dans les présentes, l'utilisation, la divulgation, la reproduction, la modification, l'exécution, l'affichage des données sont interdits sans avoir obtenu le consentement écrit préalable de NetApp, Inc. Les droits de licences du Département de la Défense du gouvernement des États-Unis se limitent aux droits identifiés par la clause 252.227-7015(b) du DFARS (février 2014).

Informations sur les marques commerciales

NETAPP, le logo NETAPP et les marques citées sur le site <http://www.netapp.com/TM> sont des marques déposées ou des marques commerciales de NetApp, Inc. Les autres noms de marques et de produits sont des marques commerciales de leurs propriétaires respectifs.