



# Mettre à niveau Trident

Trident

NetApp  
July 01, 2026

# Sommaire

- Mettre à niveau Trident . . . . . 1
  - Mettre à niveau Trident . . . . . 1
    - Considérations avant la mise à niveau . . . . . 1
    - Étape 1 : Sélectionnez une version . . . . . 1
    - Étape 2 : Déterminez la méthode d'installation d'origine . . . . . 2
    - Étape 3 : Sélectionnez une méthode de mise à niveau . . . . . 2
  - Mise à niveau avec l'opérateur . . . . . 2
    - Comprendre le workflow de mise à niveau de l'opérateur . . . . . 2
    - Mettez à niveau une installation Trident à l'aide de l'opérateur Trident ou de Helm . . . . . 3
  - Mise à niveau avec tridentctl . . . . . 7

# Mettre à niveau Trident

## Mettre à niveau Trident

À compter de la version 24.02, Trident adopte un cycle de publication de quatre mois, avec trois mises à jour majeures chaque année civile. Chaque nouvelle version s'appuie sur les précédentes et propose de nouvelles fonctionnalités, des améliorations de performances, des corrections de bogues et des améliorations. Nous vous encourageons à effectuer une mise à jour au moins une fois par an afin de profiter des nouvelles fonctionnalités de Trident.

### Considérations avant la mise à niveau

Lors de la mise à niveau vers la dernière version de Trident, tenez compte des points suivants :

- Une seule instance de Trident doit être installée dans tous les espaces de noms d'un cluster Kubernetes donné.
- Trident 23.07 et versions ultérieures nécessitent des instantanés de volume v1 et ne prennent plus en charge les instantanés alpha ou beta.
- Lors de la mise à niveau, il est important de fournir `parameter.fsType` dans `StorageClasses` utilisé par Trident. Vous pouvez supprimer et recréer `StorageClasses` sans interrompre les volumes existants.
  - Il s'agit d'une **exigence** pour appliquer "[contextes de sécurité](#)" aux volumes SAN.
  - Le répertoire [sample input](#) contient des exemples, tels que `storage-class-basic.yaml.templ` et `storage-class-bronze-default.yaml`.
  - Pour plus d'informations, consultez "[Problèmes connus](#)".

### Étape 1 : Sélectionnez une version

Les versions de Trident suivent une `YY.MM` convention de nommage basée sur la date, où « YY » sont les deux derniers chiffres de l'année et « MM » le mois. Les versions dot suivent une `YY.MM.X` convention, où « X » est le niveau de correctif. Vous sélectionnez la version vers laquelle effectuer la mise à niveau en fonction de la version à partir de laquelle vous effectuez la mise à niveau.

- Vous pouvez effectuer une mise à niveau directe vers n'importe quelle version cible située dans un intervalle de quatre versions maximum par rapport à votre version installée. Par exemple, vous pouvez effectuer une mise à niveau directe de 24.06 (ou de toute version dérivée de 24.06) vers 25.06.
- Si vous effectuez une mise à niveau depuis une version en dehors de la fenêtre de quatre versions, procédez à une mise à niveau en plusieurs étapes. Utilisez les instructions de mise à niveau pour la "[version antérieure](#)" version à partir de laquelle vous effectuez la mise à niveau afin de passer à la version la plus récente qui correspond à la fenêtre de quatre versions. Par exemple, si vous utilisez la version 23.07 et souhaitez passer à la version 25.06 :
  - a. Première mise à niveau de 23.07 à 24.06.
  - b. Ensuite, passez de 24.06 à 25.06.



Lors de la mise à niveau à l'aide de l'opérateur Trident sur OpenShift Container Platform, vous devez passer à Trident 21.01.1 ou une version ultérieure. L'opérateur Trident publié avec la version 21.01.0 contient un problème connu qui a été corrigé dans la version 21.01.1. Pour plus de détails, consultez le "[détails du problème sur GitHub](#)".

## Étape 2 : Déterminez la méthode d'installation d'origine

Pour déterminer la version que vous avez utilisée pour installer Trident :

1. Utilisez `kubectl get pods -n trident` pour examiner les pods.
  - S'il n'y a pas de module opérateur, Trident a été installé à l'aide de `tridentctl`.
  - S'il existe un pod opérateur, Trident a été installé à l'aide de l'opérateur Trident, soit manuellement, soit à l'aide de Helm.
2. S'il existe un pod opérateur, utilisez `kubectl describe torc` pour déterminer si Trident a été installé à l'aide de Helm.
  - S'il y a une étiquette Helm, Trident a été installé avec Helm.
  - S'il n'y a pas d'étiquette Helm, Trident a été installé manuellement à l'aide de l'opérateur Trident.

## Étape 3 : Sélectionnez une méthode de mise à niveau

En règle générale, vous devez effectuer la mise à niveau en utilisant la même méthode que celle employée lors de l'installation initiale, cependant vous pouvez "[passer d'une méthode d'installation à une autre](#)". Il existe deux options pour mettre à niveau Trident.

- "[Mise à niveau à l'aide de l'opérateur Trident](#)"



Nous vous suggérons de vérifier "[Comprendre le workflow de mise à niveau de l'opérateur](#)" avant de procéder à la mise à niveau avec l'opérateur.

\*

# Mise à niveau avec l'opérateur

## Comprendre le workflow de mise à niveau de l'opérateur

Avant d'utiliser l'opérateur Trident pour mettre à niveau Trident, vous devez comprendre les processus en arrière-plan qui se produisent lors de la mise à niveau. Cela inclut les modifications apportées au contrôleur Trident, au pod du contrôleur, aux pods du nœud et au DaemonSet du nœud qui permettent les mises à jour progressives.

### Gestion de la mise à niveau de l'opérateur Trident

L'un des nombreux "[avantages de l'utilisation de l'opérateur Trident](#)" moyens d'installer et de mettre à niveau Trident est la gestion automatique des objets Trident et Kubernetes sans perturber les volumes montés existants. De cette façon, Trident peut prendre en charge les mises à niveau sans interruption, ou "[mises à jour progressives](#)". En particulier, l'opérateur Trident communique avec le cluster Kubernetes pour :

- Supprimez et recréez le déploiement du contrôleur Trident et le DaemonSet du nœud.

- Remplacez le Trident Controller Pod et les Trident Node Pods par de nouvelles versions.
  - Si un nœud n'est pas mis à jour, cela n'empêche pas les autres nœuds d'être mis à jour.
  - Seuls les nœuds avec un Trident Node Pod en cours d'exécution peuvent monter des volumes.



Pour plus d'informations sur l'architecture Trident sur le cluster Kubernetes, veuillez vous référer à "[Architecture Trident](#)".

## Flux de travail de mise à niveau de l'opérateur

Lorsque vous lancez une mise à niveau à l'aide de l'opérateur Trident :

1. L'opérateur **Trident** :
  - a. Détecte la version actuellement installée de Trident (version  $n$ ).
  - b. Met à jour tous les objets Kubernetes, y compris les CRDs, RBAC et Trident SVC.
  - c. Supprime le déploiement du contrôleur Trident pour la version  $n$ .
  - d. Crée le déploiement du contrôleur Trident pour la version  $n+1$ .
2. **Kubernetes** crée le pod Trident Controller pour  $n+1$ .
3. L'opérateur **Trident** :
  - a. Supprime le DaemonSet Trident Node pour  $n$ . L'opérateur n'attend pas la fin du Pod Node.
  - b. Crée le Daemonset de nœud Trident pour  $n+1$ .
4. **Kubernetes** crée des pods de nœud Trident sur les nœuds qui n'exécutent pas le pod de nœud Trident  $n$ . Cela garantit qu'il n'y a jamais plus d'un pod de nœud Trident, de n'importe quelle version, sur un nœud.

## Mettez à niveau une installation Trident à l'aide de l'opérateur Trident ou de Helm

Vous pouvez mettre à niveau Trident à l'aide de l'opérateur Trident, soit manuellement, soit à l'aide de Helm. Vous pouvez mettre à niveau une installation de l'opérateur Trident vers une autre installation de l'opérateur Trident ou effectuer une mise à niveau d'une installation `tridentctl` vers une version opérateur Trident. Examinez "[Sélectionnez une méthode de mise à niveau](#)" avant de mettre à niveau une installation de l'opérateur Trident.

### Mettre à niveau une installation manuelle

Vous pouvez effectuer une mise à niveau d'une installation d'opérateur Trident à l'échelle d'un cluster vers une autre installation d'opérateur Trident à l'échelle d'un cluster. Toutes les versions de Trident utilisent un opérateur à l'échelle d'un cluster.



Pour mettre à niveau Trident installé à l'aide de l'opérateur à portée d'espace de noms (versions 20.07 à 20.10), utilisez les instructions de mise à niveau "[votre version installée](#)" de Trident.

### À propos de cette tâche

Trident fournit un fichier bundle que vous pouvez utiliser pour installer l'opérateur et créer les objets associés pour votre version de Kubernetes.

- Pour les clusters exécutant Kubernetes 1.25 ou une version ultérieure, utilisez "[bundle\\_post\\_1\\_25.yaml](#)".

## Avant de commencer

Assurez-vous d'utiliser un cluster Kubernetes exécutant "[une version Kubernetes prise en charge](#)".

### Étapes

1. Vérifiez votre version de Trident :

```
./tridentctl -n trident version
```

2. Mettez à jour le `operator.yaml`, `tridentorchestrator_cr.yaml`, et `post_1_25_bundle.yaml` avec le registre et les chemins d'image pour la version vers laquelle vous effectuez la mise à niveau (par exemple 25.06), ainsi que le secret correct.
3. Supprimez l'opérateur Trident qui a été utilisé pour installer l'instance Trident actuelle. Par exemple, si vous effectuez une mise à niveau depuis la version 25.02, exécutez la commande suivante :

```
kubectl delete -f 25.02.0/trident-installer/deploy/<bundle.yaml> -n trident
```

4. Si vous avez personnalisé votre installation initiale à l'aide des `TridentOrchestrator` attributs, vous pouvez modifier l'objet `TridentOrchestrator` pour modifier les paramètres d'installation. Cela peut inclure des modifications apportées pour spécifier des registres d'images Trident et CSI en miroir pour le mode hors ligne, activer les journaux de débogage ou spécifier des secrets d'extraction d'images.
5. Installez Trident en utilisant le fichier YAML de bundle approprié pour votre environnement, où `<bundle.yaml>` est `bundle_pre_1_25.yaml` ou `bundle_post_1_25.yaml` selon votre version de Kubernetes. Par exemple, si vous installez Trident 25.06.0, exécutez la commande suivante :

```
kubectl create -f 25.06.0/trident-installer/deploy/<bundle.yaml> -n trident
```

6. Modifiez le `trident torc` pour inclure l'image 25.06.0.

## Mettre à niveau une installation Helm

Vous pouvez mettre à niveau une installation Trident Helm.



Lors de la mise à niveau d'un cluster Kubernetes de la version 1.24 à la version 1.25 ou ultérieure sur lequel Trident est installé, vous devez mettre à jour `values.yaml` pour définir `excludePodSecurityPolicy` sur `true` ou ajouter `--set excludePodSecurityPolicy=true` à la commande `helm upgrade` avant de pouvoir mettre à niveau le cluster.

Si vous avez déjà mis à niveau votre cluster Kubernetes de la version 1.24 à la version 1.25 sans mettre à niveau le helm Trident, la mise à niveau du helm échouera. Pour que la mise à niveau du helm réussisse, effectuez les étapes suivantes comme prérequis :

1. Installez le plugin `helm-mapkubeapis` depuis <https://github.com/helm/helm-mapkubeapis>.

2. Effectuez un test à blanc pour la version Trident dans l'espace de noms où Trident est installé. Cela liste les ressources qui seront nettoyées.

```
helm mapkubeapis --dry-run trident --namespace trident
```

3. Effectuez une exécution complète avec helm pour effectuer le nettoyage.

```
helm mapkubeapis trident --namespace trident
```

## Étapes

1. Si vous "[installé Trident à l'aide de Helm](#)", vous pouvez utiliser `helm upgrade trident netapp-trident/trident-operator --version 100.2602.0` pour effectuer la mise à niveau en une seule étape. Si vous n'avez pas ajouté le dépôt Helm ou si vous ne pouvez pas l'utiliser pour effectuer la mise à niveau :
  - a. Téléchargez la dernière version de Trident depuis "[la section Assets sur GitHub](#)".
  - b. Utilisez la commande `helm upgrade` où `trident-operator-26.02.0.tgz` correspond à la version vers laquelle vous souhaitez effectuer la mise à niveau.

```
helm upgrade <name> trident-operator-26.02.0.tgz
```



Si vous définissez des options personnalisées lors de l'installation initiale (comme la spécification de registres privés et en miroir pour Trident et les images CSI), ajoutez la `helm upgrade` commande en utilisant `--set` pour garantir que ces options sont incluses dans la commande de mise à niveau, sinon les valeurs seront réinitialisées à la valeur par défaut.

2. Exécutez `helm list` pour vérifier que le graphique et la version de l'application ont tous deux été mis à niveau. Exécutez `tridentctl logs` pour examiner les messages de débogage.

## Mise à niveau d'une `tridentctl` installation vers l'opérateur Trident

Vous pouvez mettre à niveau vers la dernière version de l'opérateur Trident à partir d'une installation `tridentctl` existante. Les backends et les PVC existants seront automatiquement disponibles.



Avant de changer de méthode d'installation, consultez "[Passer d'une méthode d'installation à une autre](#)".

## Étapes

1. Téléchargez la dernière version de Trident.

```
# Download the release required [26.02.0]
mkdir 26.02.0
cd 26.02.0
wget
https://github.com/NetApp/trident/releases/download/v26.02.0/trident-
installer-26.02.0.tar.gz
tar -xf trident-installer-26.02.0.tar.gz
cd trident-installer
```

## 2. Créez la tridentorchestrator CRD à partir du manifeste.

```
kubectl create -f
deploy/crds/trident.netapp.io_tridentorchestrators_crd_post1.16.yaml
```

## 3. Déployez l'opérateur à portée de cluster dans le même espace de noms.

```
kubectl create -f deploy/<bundle-name.yaml>

serviceaccount/trident-operator created
clusterrole.rbac.authorization.k8s.io/trident-operator created
clusterrolebinding.rbac.authorization.k8s.io/trident-operator created
deployment.apps/trident-operator created
podsecuritypolicy.policy/tridentoperatorpods created

#Examine the pods in the Trident namespace
NAME                                READY   STATUS    RESTARTS   AGE
trident-controller-79df798bdc-m79dc 6/6     Running   0           150d
trident-node-linux-xrst8             2/2     Running   0           150d
trident-operator-5574dbbc68-nthjv    1/1     Running   0           1m30s
```

## 4. Créer une TridentOrchestrator CR pour installer Trident.

```

cat deploy/crds/tridentorchestrator_cr.yaml
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident

kubectl create -f deploy/crds/tridentorchestrator_cr.yaml

#Examine the pods in the Trident namespace
NAME                                READY   STATUS    RESTARTS   AGE
trident-csi-79df798bdc-m79dc        6/6     Running   0           1m
trident-csi-xrst8                    2/2     Running   0           1m
trident-operator-5574dbbc68-nthjv   1/1     Running   0           5m41s

```

5. Confirmez que Trident a été mis à niveau vers la version prévue.

```

kubectl describe torc trident | grep Message -A 3

Message:          Trident installed
Namespace:        trident
Status:           Installed
Version:          v26.02.0

```

## Mise à niveau avec tridentctl

Vous pouvez facilement mettre à niveau une installation Trident existante en utilisant `tridentctl`.

### À propos de cette tâche

La désinstallation et la réinstallation de Trident constituent une mise à niveau. Lorsque vous désinstallez Trident, la Persistent Volume Claim (PVC) et le Persistent Volume (PV) utilisés par le déploiement Trident ne sont pas supprimés. Les PV déjà provisionnés restent disponibles pendant que Trident est hors ligne, et Trident provisionnera des volumes pour toutes les PVC créées entre-temps après son retour en ligne.

### Avant de commencer

Vérifiez "[Sélectionnez une méthode de mise à niveau](#)" avant la mise à niveau en utilisant `tridentctl`.

### Étapes

1. Exécutez la commande de désinstallation dans `tridentctl` pour supprimer toutes les ressources associées à Trident à l'exception des CRD et des objets associés.

```
./tridentctl uninstall -n <namespace>
```

2. Réinstallez Trident. Consultez "[Installez Trident à l'aide de tridentctl](#)".



N'interrompez pas le processus de mise à niveau. Assurez-vous que l'installateur s'exécute jusqu'à la fin.

## 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 JOUISSANCE, 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 DÉLICTELLE (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.