Upgrade with the operator

Astra Trident

NetApp
January 14, 2026

This PDF was generated from https://docs.netapp.com/us-en/trident-2406/trident-managing-k8s/upgrade-operator-overview.html on January 14, 2026. Always check docs.netapp.com for the latest.

Table of Contents

Upgrade with the operator . . . 1
- Understand the operator upgrade workflow . . . 1
  - Trident operator upgrade handling . . . 1
  - Operator upgrade workflow . . . 1
- Upgrade an Astra Trident installation using Trident operator or Helm . . . 1
  - Upgrade a manual installation . . . 2
  - Upgrade a Helm installation . . . 3
  - Upgrade from a `tridentctl` installation to Trident operator . . . 3

Upgrade with the operator

Understand the operator upgrade workflow

Before using the Trident operator to upgrade Astra Trident, you should understand the background processes that occur during upgrade. This includes changes to the Trident controller, controller Pod and node Pods, and node DaemonSet that enable rolling updates.

Trident operator upgrade handling

One of the many benefits of using the Trident operator to install and upgrade Astra Trident is the automatic handling of Astra Trident and Kubernetes objects without disrupting existing mounted volumes. In this way, Astra Trident can support upgrades with zero downtime, or *rolling updates*. In particular, the Trident operator communicates with the Kubernetes cluster to:

- Delete and recreate the Trident Controller deployment and node DaemonSet.
- Replace the Trident Controller Pod and Trident Node Pods with new versions.
  - If a node is not updated, it does not prevent remaining nodes from being updated.
  - Only nodes with a running Trident Node Pod can mount volumes.

For more information about Astra Trident architecture on the Kubernetes cluster, refer to Astra Trident architecture.

Operator upgrade workflow

When you initiate an upgrade using the Trident operator:

1. The **Trident operator**:
   a. Detects the currently installed version of Astra Trident (version *n*).
   b. Updates all Kubernetes objects including CRDs, RBAC, and Trident SVC.
   c. Deletes the Trident Controller deployment for version *n*.
   d. Creates the Trident Controller deployment for version *n+1*.
2. **Kubernetes** creates Trident Controller Pod for *n+1*.
3. The **Trident operator**:
   a. Deletes the Trident Node DaemonSet for *n*. The operator does not wait for Node Pod termination.
   b. Creates the Trident Node Daemonset for *n+1*.
4. **Kubernetes** creates Trident Node Pods on nodes not running Trident Node Pod *n*. This ensures there is never more than one Trident Node Pod, of any version, on a node.

Upgrade an Astra Trident installation using Trident operator or Helm

You can upgrade Astra Trident using the Trident operator either manually or using Helm.

You can upgrade from a Trident operator installation to another Trident operator installation or upgrade from a `tridentctl` installation to a Trident operator version. Review Select an upgrade method before upgrading a Trident operator installation.

Upgrade a manual installation

You can upgrade from a cluster-scoped Trident operator installation to another cluster-scoped Trident operator installation. All Astra Trident versions 21.01 and above use a cluster-scoped operator.

> To upgrade from Astra Trident that was installed using the namespace-scoped operator (versions 20.07 through 20.10), use the upgrade instructions for your installed version of Astra Trident.

About this task

Trident provides a bundle file you can use to install the operator and create associated objects for your Kubernetes version.

- For clusters running Kubernetes 1.24, use bundle_pre_1_25.yaml.
- For clusters running Kubernetes 1.25 or later, use bundle_post_1_25.yaml.

Before you begin

Ensure you are using a Kubernetes cluster running a supported Kubernetes version.

Steps

1. Verify your Astra Trident version:

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

2. Delete the Trident operator that was used to install the current Astra Trident instance. For example, if you are upgrading from 23.07, run the following command:

```
kubectl delete -f 23.07.0/trident-installer/deploy/<bundle.yaml> -n
trident
```

3. If you customized your initial installation using `TridentOrchestrator` attributes, you can edit the `TridentOrchestrator` object to modify the installation parameters. This might include changes made to specify mirrored Trident and CSI image registries for offline mode, enable debug logs, or specify image pull secrets.
4. Install Astra Trident using the correct bundle YAML file for your environment, where *<bundle.yaml>* is `bundle_pre_1_25.yaml` or `bundle_post_1_25.yaml` based on your Kubernetes version. For example, if you are installing Astra Trident 24.06, run the following command:

```
kubectl create -f 24.06.0/trident-installer/deploy/<bundle.yaml> -n
trident
```

Upgrade a Helm installation

You can upgrade an Astra Trident Helm installation.

> When upgrading a Kubernetes cluster from 1.24 to 1.25 or later that has Astra Trident installed, you must update values.yaml to set `excludePodSecurityPolicy` to `true` or add `--set excludePodSecurityPolicy=true` to the `helm upgrade` command before you can upgrade the cluster.

Steps

1. If you installed Astra Trident using Helm, you can use `helm upgrade trident netapp-trident/trident-operator --version 100.2406.0` to upgrade in one step. If you did not add the Helm repo or cannot use it to upgrade:
   a. Download the latest Astra Trident release from the *Assets* section on GitHub.
   b. Use the `helm upgrade` command where `trident-operator-24.06.0.tgz` reflects the version that you want to upgrade to.

```
helm upgrade <name> trident-operator-24.06.0.tgz
```

> If you set custom options during the initial installation (such as specifying private, mirrored registries for Trident and CSI images), append the `helm upgrade` command using `--set` to ensure those options are included in the upgrade command, otherwise the values will reset to default.

2. Run `helm list` to verify that the chart and app version have both been upgraded. Run `tridentctl logs` to review any debug messages.

Upgrade from a `tridentctl` installation to Trident operator

You can upgrade to the latest release of the Trident operator from a `tridentctl` installation. The existing backends and PVCs will automatically be available.

> Before switching between installation methods, review Moving between installation methods.

Steps

1. Download the latest Astra Trident release.

```
# Download the release required [24.060.0]
mkdir 24.06.0
cd 24.06.0
wget
https://github.com/NetApp/trident/releases/download/v24.06.0/trident-
installer-24.06.0.tar.gz
tar -xf trident-installer-24.06.0.tar.gz
cd trident-installer
```

2. Create the `tridentorchestrator` CRD from the manifest.

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

3. Deploy the cluster-scoped operator in the same namespace.

```
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. Create a `TridentOrchestrator` CR for installing Astra 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. Confirm Trident was upgraded to the intended version.

```
kubectl describe torc trident | grep Message -A 3

Message:                Trident installed
Namespace:              trident
Status:                 Installed
Version:                v24.06.0
```

Copyright information

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.