



Prepare the worker node

Trident

NetApp
January 14, 2026

Table of Contents

Prepare the worker node	1
Selecting the right tools	1
Node service discovery	1
NFS volumes	2
iSCSI volumes	2
iSCSI self-healing capabilities	2
Install the iSCSI tools	3
Configure or disable iSCSI self healing	5
NVMe/TCP volumes	6
Verify installation	7
Install the FC tools	7
Fibre Channel (FC) support	9
Prerequisites	9
Create a backend configuration	11
Create a storage class	11

Prepare the worker node

All worker nodes in the Kubernetes cluster must be able to mount the volumes you have provisioned for your pods. To prepare the worker nodes, you must install NFS, iSCSI, NVMe/TCP, or FC tools based on your driver selection.

Selecting the right tools

If you are using a combination of drivers, you should install all required tools for your drivers. Recent versions of RedHat CoreOS have the tools installed by default.

NFS tools

[Install the NFS tools](#) if you are using: `ontap-nas`, `ontap-nas-economy`, `ontap-nas-flexgroup`, `azure-netapp-files`, `gcp-cvs`.

iSCSI tools

[Install the iSCSI tools](#) if you are using: `ontap-san`, `ontap-san-economy`, `solidfire-san`.

NVMe tools

[Install the NVMe tools](#) if you are using `ontap-san` for nonvolatile memory express (NVMe) over TCP (NVMe/TCP) protocol.



We recommend ONTAP 9.12 or later for NVMe/TCP.

SCSI over FC tools

SCSI over Fibre Channel (FC) is a tech preview feature in the Trident 24.10 release.

[Install the FC tools](#) if you are using `ontap-san` with `sanType fcp` (SCSI over FC).

Refer to [Ways to configure FC & FC-NVMe SAN hosts](#) for more information.

Node service discovery

Trident attempts to automatically detect if the node can run iSCSI or NFS services.



Node service discovery identifies discovered services but does not guarantee services are properly configured. Conversely, the absence of a discovered service does not guarantee the volume mount will fail.

Review events

Trident creates events for the node to identify the discovered services. To review these events, run:

```
kubectl get event -A --field-selector involvedObject.name=<Kubernetes node name>
```

Review discovered services

Trident identifies services enabled for each node on the Trident node CR. To view the discovered services, run:

```
tridentctl get node -o wide -n <Trident namespace>
```

NFS volumes

Install the NFS tools using the commands for your operating system. Ensure the NFS service is started up during boot time.

RHEL 8+

```
sudo yum install -y nfs-utils
```

Ubuntu

```
sudo apt-get install -y nfs-common
```



Reboot your worker nodes after installing the NFS tools to prevent failure when attaching volumes to containers.

iSCSI volumes

Trident can automatically establish an iSCSI session, scan LUNs, and discover multipath devices, format them, and mount them to a pod.

iSCSI self-healing capabilities

For ONTAP systems, Trident runs iSCSI self-healing every five minutes to:

1. **Identify** the desired iSCSI session state and the current iSCSI session state.
2. **Compare** the desired state to the current state to identify needed repairs. Trident determines repair priorities and when to preempt repairs.
3. **Perform repairs** required to return the current iSCSI session state to the desired iSCSI session state.



Logs of self-healing activity are located in the `trident-main` container on the respective Daemonset pod. To view logs, you must have set `debug` to "true" during Trident installation.

Trident iSCSI self-healing capabilities can help prevent:

- Stale or unhealthy iSCSI sessions that could occur after a network connectivity issue. In the case of a stale session, Trident waits seven minutes before logging out to reestablish the connection with a portal.



For example, if CHAP secrets were rotated on the storage controller and the network loses connectivity, the old (`stale`) CHAP secrets could persist. Self-healing can recognize this and automatically reestablish the session to apply the updated CHAP secrets.

- Missing iSCSI sessions
- Missing LUNs

Points to consider before upgrading Trident

- If only per-node igroups (introduced in 23.04+) are in use, iSCSI self-healing will initiate SCSI rescans for all devices in the SCSI bus.
- If only backend-scoped igroups (deprecated as of 23.04) are in use, iSCSI self-healing will initiate SCSI rescans for exact LUN IDs in the SCSI bus.
- If a mix of per-node igroups and backend-scoped igroups are in use, iSCSI self-healing will initiate SCSI rescans for exact LUN IDs in the SCSI bus.

Install the iSCSI tools

Install the iSCSI tools using the commands for your operating system.

Before you begin

- Each node in the Kubernetes cluster must have a unique IQN. **This is a necessary prerequisite.**
- If using RHCOS version 4.5 or later, or other RHEL-compatible Linux distribution, with the `solidfire-san` driver and Element OS 12.5 or earlier, ensure that the CHAP authentication algorithm is set to MD5 in `/etc/iscsi/iscsid.conf`. Secure FIPS-compliant CHAP algorithms SHA1, SHA-256, and SHA3-256 are available with Element 12.7.

```
sudo sed -i 's/^\\(node.session.auth.chap_algs\\) .*\\1 = MD5/'  
/etc/iscsi/iscsid.conf
```

- When using worker nodes that run RHEL/RedHat CoreOS with iSCSI PVs, specify the `discard` `mountOption` in the `StorageClass` to perform inline space reclamation. Refer to [RedHat documentation](#).

RHEL 8+

1. Install the following system packages:

```
sudo yum install -y lsscsi iscsi-initiator-utils device-mapper-multipath
```

2. Check that iscsi-initiator-utils version is 6.2.0.874-2.el7 or later:

```
rpm -q iscsi-initiator-utils
```

3. Enable multipathing:

```
sudo mpathconf --enable --with_multipathd y --find_multipaths n
```



Ensure etc/multipath.conf contains `find_multipaths n` under defaults.

4. Ensure that `iscsid` and `multipathd` are running:

```
sudo systemctl enable --now iscsid multipathd
```

5. Enable and start `iscsi`:

```
sudo systemctl enable --now iscsi
```

Ubuntu

1. Install the following system packages:

```
sudo apt-get install -y open-iscsi lsscsi sg3-utils multipath-tools scsitoools
```

2. Check that `open-iscsi` version is 2.0.874-5ubuntu2.10 or later (for bionic) or 2.0.874-7.1ubuntu6.1 or later (for focal):

```
dpkg -l open-iscsi
```

3. Set scanning to manual:

```
sudo sed -i 's/^(node.session.scan\).*$/\1 = manual/'  
/etc/iscsi/iscsid.conf
```

4. Enable multipathing:

```
sudo tee /etc/multipath.conf <<-EOF  
defaults {  
    user_friendly_names yes  
    find_multipaths no  
}  
EOF  
sudo systemctl enable --now multipath-tools.service  
sudo service multipath-tools restart
```



Ensure `etc/multipath.conf` contains `find_multipaths no` under `defaults`.

5. Ensure that `open-iscsi` and `multipath-tools` are enabled and running:

```
sudo systemctl status multipath-tools  
sudo systemctl enable --now open-iscsi.service  
sudo systemctl status open-iscsi
```



For Ubuntu 18.04, you must discover target ports with `iscsiadm` before starting `open-iscsi` for the iSCSI daemon to start. You can alternatively modify the `iscsi` service to start `iscsid` automatically.

Configure or disable iSCSI self healing

You can configure the following Trident iSCSI self-healing settings to fix stale sessions:

- **iSCSI self-healing interval:** Determines the frequency at which iSCSI self-healing is invoked (default: 5 minutes). You can configure it to run more frequently by setting a smaller number or less frequently by setting a larger number.



Setting the iSCSI self-healing interval to 0 stops iSCSI self-healing completely. We do not recommend disabling iSCSI Self-healing; it should only be disabled in certain scenarios when iSCSI self-healing is not working as intended or for debugging purposes.

- **iSCSI Self-Healing Wait Time:** Determines the duration iSCSI self-healing waits before logging out of an unhealthy session and trying to log in again (default: 7 minutes). You can configure it to a larger number so that sessions that are identified as unhealthy have to wait longer before being logged out and then an attempt is made to log back in, or a smaller number to log out and log in earlier.

Helm

To configure or change iSCSI self-healing settings, pass the `iscsiSelfHealingInterval` and `iscsiSelfHealingWaitTime` parameters during the helm installation or helm update.

The following example sets the iSCSI self-healing interval to 3 minutes and self-healing wait time to 6 minutes:

```
helm install trident trident-operator-100.2410.0.tgz --set  
iscsiSelfHealingInterval=3m0s --set iscsiSelfHealingWaitTime=6m0s -n  
trident
```

tridentctl

To configure or change iSCSI self-healing settings, pass the `iscsi-self-healing-interval` and `iscsi-self-healing-wait-time` parameters during the tridentctl installation or update.

The following example sets the iSCSI self-healing interval to 3 minutes and self-healing wait time to 6 minutes:

```
tridentctl install --iscsi-self-healing-interval=3m0s --iscsi-self  
-healing-wait-time=6m0s -n trident
```

NVMe/TCP volumes

Install the NVMe tools using the commands for your operating system.

- NVMe requires RHEL 9 or later.
- If the kernel version of your Kubernetes node is too old or if the NVMe package is not available for your kernel version, you might have to update the kernel version of your node to one with the NVMe package.

RHEL 9

```
sudo yum install nvme-cli  
sudo yum install linux-modules-extra-$ (uname -r)  
sudo modprobe nvme-tcp
```

Ubuntu

```
sudo apt install nvme-cli  
sudo apt -y install linux-modules-extra-$ (uname -r)  
sudo modprobe nvme-tcp
```

Verify installation

After installation, verify that each node in the Kubernetes cluster has a unique NQN using the command:

```
cat /etc/nvme/hostnqn
```



Trident modifies the `ctrl_device_tmo` value to ensure NVMe doesn't give up on the path if it goes down. Do not change this setting.

Install the FC tools

Install the FC tools using the commands for your operating system.

- When using worker nodes that run RHEL/RedHat CoreOS with FC PVs, specify the `discard` mountOption in the StorageClass to perform inline space reclamation. Refer to [RedHat documentation](#).

RHEL 8+

1. Install the following system packages:

```
sudo yum install -y lsscsi device-mapper-multipath
```

2. Enable multipathing:

```
sudo mpathconf --enable --with_multipathd y --find_multipaths n
```



Ensure `etc/multipath.conf` contains `find_multipaths no` under `defaults`.

3. Ensure that `multipathd` is running:

```
sudo systemctl enable --now multipathd
```

Ubuntu

1. Install the following system packages:

```
sudo apt-get install -y lsscsi sg3-utils multipath-tools scsitoools
```

2. Enable multipathing:

```
sudo tee /etc/multipath.conf <<-EOF
defaults {
    user_friendly_names yes
    find_multipaths no
}
EOF
sudo systemctl enable --now multipath-tools.service
sudo service multipath-tools restart
```



Ensure `etc/multipath.conf` contains `find_multipaths no` under `defaults`.

3. Ensure that `multipath-tools` is enabled and running:

```
sudo systemctl status multipath-tools
```

Fibre Channel (FC) support

You can now use the Fibre Channel (FC) protocol with Trident to provision and manage storage resources on ONTAP system.

SCSI over Fibre Channel (FC) is a tech preview feature in the Trident 24.10 release.

Fibre Channel is a widely adopted protocol in enterprise storage environments due to its high performance, reliability, and scalability. It provides a robust and efficient communication channel for storage devices, enabling fast and secure data transfers.

By using SCSI over Fibre Channel, you can leverage their existing SCSI-based storage infrastructure while benefiting from the high-performance and long-distance capabilities of Fibre Channel. It enables the consolidation of storage resources and the creation of scalable and efficient storage area networks (SANs) that can handle large amounts of data with low latency.

Using the FC feature with Trident, you can do the following:

- Dynamically provision PVCs using a deployment spec.
- Take volume snapshots and create a new volume from the snapshot.
- Clone an existing FC-PVC.
- Resize an already deployed volume.

Prerequisites

Configure the required network and node settings for FC.

Network settings

1. Get the WWPN of the target interfaces. Refer to [network interface show](#) for more information.
2. Get the WWPN for the interfaces on initiator (Host).

Refer to the corresponding host operating system utilities.

3. Configure zoning on the FC switch using WWPNs of the Host and target.

Refer to the respective switch vendor documentation for information.

Refer to the following ONTAP documentation for details:

- [Fibre Channel and FCoE zoning overview](#)
- [Ways to configure FC & FC-NVMe SAN hosts](#)

Prepare the worker node

All worker nodes in the Kubernetes cluster must be able to mount the volumes you have provisioned for your pods. To prepare the worker nodes for FC, you must install the required tools.

Install the FC tools

Install the FC tools using the commands for your operating system.

- When using worker nodes that run RHEL/RedHat CoreOS with FC PVs, specify the `discard` mountOption

in the StorageClass to perform inline space reclamation. Refer to [RedHat documentation](#).

RHEL 8+

1. Install the following system packages:

```
sudo yum install -y lsscsi device-mapper-multipath
```

2. Enable multipathing:

```
sudo mpathconf --enable --with_multipathd y --find_multipaths n
```



Ensure `etc/multipath.conf` contains `find_multipaths no` under `defaults`.

3. Ensure that `multipathd` is running:

```
sudo systemctl enable --now multipathd
```

Ubuntu

1. Install the following system packages:

```
sudo apt-get install -y lsscsi sg3-utils multipath-tools scsitoools
```

2. Enable multipathing:

```
sudo tee /etc/multipath.conf <<-EOF
defaults {
    user_friendly_names yes
    find_multipaths no
}
EOF
sudo systemctl enable --now multipath-tools.service
sudo service multipath-tools restart
```



Ensure `etc/multipath.conf` contains `find_multipaths no` under `defaults`.

3. Ensure that `multipath-tools` is enabled and running:

```
sudo systemctl status multipath-tools
```

Create a backend configuration

Create a Trident backend for `ontap-san` driver and `fcp` as the `sanType`.

Refer to:

- [Prepare to configure backend with ONTAP SAN drivers](#)
- [ONTAP SAN configuration options and examples](#)

Backend configuration example with FC

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-san
spec:
  version: 1
  backendName: ontap-san-backend
  storageDriverName: ontap-san
  managementLIF: 10.0.0.1
  sanType: fcp
  svm: trident_svm
  credentials:
    name: backend-tbc-ontap-san-secret
```

Create a storage class

For more information, refer to:

- [Storage configuration options](#)

Storage class example

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: fcp-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  storagePools: "ontap-san-backend:.*"
  fsType: "ext4"
allowVolumeExpansion: True
```

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.