



Amazon FSx for NetApp ONTAP

Trident

NetApp

January 14, 2026

Table of Contents

Amazon FSx for NetApp ONTAP	1
Use Trident with Amazon FSx for NetApp ONTAP	1
Requirements	1
Considerations	1
Authentication	2
Tested Amazon Machine Images (AMIs)	2
Find more information	3
Create an IAM role and AWS Secret	3
Create AWS Secret Manager secret	3
Create IAM Policy	3
Install Trident	6
Install Trident via helm	6
Install Trident via the EKS add-on	7
Configure the Storage Backend	12
ONTAP SAN and NAS driver integration	12
FSx for ONTAP driver details	14
Backend advanced configuration and examples	15
Backend configuration options for provisioning volumes	19
Prepare to provision SMB volumes	21
Configure a storage class and PVC	22
Create a storage class	22
Create the PV and PVC	23
Trident attributes	26
Deploy sample application	27
Configure the Trident EKS add-on on an EKS cluster	28
Prerequisites	29
Steps	29
Install/uninstall the Trident EKS add-on using CLI	32

Amazon FSx for NetApp ONTAP

Use Trident with Amazon FSx for NetApp ONTAP

[Amazon FSx for NetApp ONTAP](#) is a fully managed AWS service that enables customers to launch and run file systems powered by the NetApp ONTAP storage operating system. FSx for ONTAP enables you to leverage NetApp features, performance, and administrative capabilities you are familiar with, while taking advantage of the simplicity, agility, security, and scalability of storing data on AWS. FSx for ONTAP supports ONTAP file system features and administration APIs.

You can integrate your Amazon FSx for NetApp ONTAP file system with Trident to ensure Kubernetes clusters running in Amazon Elastic Kubernetes Service (EKS) can provision block and file persistent volumes backed by ONTAP.

A file system is the primary resource in Amazon FSx, analogous to an ONTAP cluster on premises. Within each SVM you can create one or multiple volumes, which are data containers that store the files and folders in your file system. With Amazon FSx for NetApp ONTAP, Data ONTAP will be provided as a managed file system in the cloud. The new file system type is called **NetApp ONTAP**.

Using Trident with Amazon FSx for NetApp ONTAP, you can ensure Kubernetes clusters running in Amazon Elastic Kubernetes Service (EKS) can provision block and file persistent volumes backed by ONTAP.

Requirements

In addition to [Trident requirements](#), to integrate FSx for ONTAP with Trident, you need:

- An existing Amazon EKS cluster or self-managed Kubernetes cluster with `kubectl` installed.
- An existing Amazon FSx for NetApp ONTAP file system and storage virtual machine (SVM) that is reachable from your cluster's worker nodes.
- Worker nodes that are prepared for [NFS or iSCSI](#).



Ensure you follow the node preparation steps required for Amazon Linux and Ubuntu [Amazon Machine Images](#) (AMIs) depending on your EKS AMI type.

Considerations

- SMB volumes:
 - SMB volumes are supported using the `ontap-nas` driver only.
 - SMB volumes are not supported with Trident EKS add-on.
 - Trident supports SMB volumes mounted to pods running on Windows nodes only. Refer to [Prepare to provision SMB volumes](#) for details.
- Prior to Trident 24.02, volumes created on Amazon FSx file systems that have automatic backups enabled, could not be deleted by Trident. To prevent this issue in Trident 24.02 or later, specify the `fsxFilesystemID`, `AWS apiRegion`, `AWS apikey`, and `AWS secretKey` in the backend configuration file for AWS FSx for ONTAP.



If you are specifying an IAM role to Trident, then you can omit specifying the `apiRegion`, `apiKey`, and `secretKey` fields to Trident explicitly. For more information, refer to [FSx for ONTAP configuration options and examples](#).

Authentication

Trident offers two modes of authentication.

- Credential-based(Recommended): Stores credentials securely in AWS Secrets Manager. You can use the `fsxadmin` user for your file system or the `vsadmin` user configured for your SVM.



Trident expects to be run as a `vsadmin` SVM user or as a user with a different name that has the same role. Amazon FSx for NetApp ONTAP has an `fsxadmin` user that is a limited replacement of the ONTAP `admin` cluster user. We strongly recommend using `vsadmin` with Trident.

- Certificate-based: Trident will communicate with the SVM on your FSx file system using a certificate installed on your SVM.

For details on enabling authentication, refer to the authentication for your driver type:

- [ONTAP NAS authentication](#)
- [ONTAP SAN authentication](#)

Tested Amazon Machine Images (AMIs)

EKS cluster supports various operating systems, but AWS has optimized certain Amazon Machine Images (AMIs) for containers and EKS. The following AMIs have been tested with Trident 24.10.

AMI	NAS	NAS-economy	SAN	SAN-economy
AL2023_x86_64_ST ANDARD	Yes	Yes	Yes	Yes
AL2_x86_64	Yes	Yes	Yes**	Yes**
BOTTLEROCKET_x 86_64	Yes*	Yes	N/A	N/A
AL2023_ARM_64_S TANDARD	Yes	Yes	Yes	Yes
AL2_ARM_64	Yes	Yes	Yes**	Yes**
BOTTLEROCKET_A RM_64	Yes*	Yes	N/A	N/A

- *Must use “nolock” in mount options.
- ** Unable to delete the PV without restarting the node



If your desired AMI is not listed here, it does not mean that it is not supported; it simply means it has not been tested. This list serves as a guide for AMIs known to work.

Tests performed with:

- EKS version: 1.30
- Installation Method: Helm and as an AWS add-On
- For NAS both NFSv3 and NFSv4.1 were tested.
- For SAN only iSCSI was tested, not NVMe-oF.

Tests performed:

- Create: Storage Class, pvc, pod
- Delete: pod, pvc (regular, qtree/lun – economy, NAS with AWS backup)

Find more information

- [Amazon FSx for NetApp ONTAP documentation](#)
- [Blog post on Amazon FSx for NetApp ONTAP](#)

Create an IAM role and AWS Secret

You can configure Kubernetes pods to access AWS resources by authenticating as an AWS IAM role instead of by providing explicit AWS credentials.



To authenticate using an AWS IAM role, you must have a Kubernetes cluster deployed using EKS.

Create AWS Secret Manager secret

This example creates an AWS Secret Manager secret to store Trident CSI credentials:

```
aws secretsmanager create-secret --name trident-secret --description "Trident CSI credentials"\n  --secret-string\n  "{\"username\":\"vsadmin\",\"password\":\"<svmpassword>\"}"
```

Create IAM Policy

The following examples creates an IAM policy using the AWS CLI:

```
aws iam create-policy --policy-name AmazonFSxNCSIReaderPolicy --policy\n  -document file://policy.json\n  --description "This policy grants access to Trident CSI to FSxN and\n  Secret manager"
```

Policy JSON file:

```
policy.json:
{
  "Statement": [
    {
      "Action": [
        "fsx:DescribeFileSystems",
        "fsx:DescribeVolumes",
        "fsx>CreateVolume",
        "fsx:RestoreVolumeFromSnapshot",
        "fsx:DescribeStorageVirtualMachines",
        "fsx:UntagResource",
        "fsx:UpdateVolume",
        "fsx:TagResource",
        "fsx:DeleteVolume"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": "secretsmanager:GetSecretValue",
      "Effect": "Allow",
      "Resource": "arn:aws:secretsmanager:<aws-region>:<aws-account-id>:secret:<aws-secret-manager-name>*"
    }
  ],
  "Version": "2012-10-17"
}
```

Create an IAM role for the service account

AWS CLI

```
aws iam create-role --role-name trident-controller \
--assume-role-policy-document file://trust-relationship.json
```

trust-relationship.json file:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::<account_id>:oidc-provider/<oidc_provider>"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "<oidc_provider>:aud": "sts.amazonaws.com",
          "<oidc_provider>:sub": "system:serviceaccount:trident:trident-controller"
        }
      }
    }
  ]
}
```

Update the following values in the `trust-relationship.json` file:

- `<account_id>` - Your AWS account ID
- `<oidc_provider>` - The OIDC of your EKS cluster. You can obtain the `oidc_provider` by running:

```
aws eks describe-cluster --name my-cluster --query
"cluster.identity.oidc.issuer"\n
--output text | sed -e "s/^https://\//\//\//\//"
```

Attach the IAM role with the IAM policy:

Once the role has been created, attach the policy (that was created in the step above) to the role using this command:

```
aws iam attach-role-policy --role-name my-role --policy-arn <IAM policy ARN>
```

Verify OIDC provider is associated:

Verify that your OIDC provider is associated with your cluster. You can verify it using this command:

```
aws iam list-open-id-connect-providers | grep $oidc_id | cut -d "/" -f4
```

Use the following command to associate IAM OIDC to your cluster:

```
eksctl utils associate-iam-oidc-provider --cluster $cluster_name  
--approve
```

eksctl

The following example creates an IAM role for service account in EKS:

```
eksctl create iamserviceaccount --name trident-controller --namespace  
trident \  
--cluster <my-cluster> --role-name <AmazonEKS_FSn_CSI_DriverRole>  
--role-only \  
--attach-policy-arn <IAM-Policy ARN> --approve
```

Install Trident

Trident streamlines Amazon FSx for NetApp ONTAP storage management in Kubernetes to enable your developers and administrators focus on application deployment.

You can install Trident using one of the following methods:

- Helm
- EKS add-on

If you want to make use of the snapshot functionality, install the CSI snapshot controller add-on. Refer to [Enable snapshot functionality for CSI volumes](#) for more information.

Install Trident via helm

1. Download the Trident installer package

The Trident installer package contains everything you need to deploy the Trident operator and install Trident. Download and extract the latest version of the Trident installer from the Assets section on GitHub.

```
wget https://github.com/NetApp/trident/releases/download/v24.10.0/trident-installer-24.10.0.tar.gz  
tar -xf trident-installer-24.10.0.tar.gz  
cd trident-installer/helm
```

2. Set the values for **cloud provider** and **cloud identity** flags using the following environment variables:

The following example installs Trident and sets the `cloud-provider` flag to `$CP`, and `cloud-identity` to `$SCI`:

```
helm install trident trident-operator-100.2410.0.tgz --set
cloudProvider="AWS" \
--set cloudIdentity="'eks.amazonaws.com/role-arn:
arn:aws:iam::<accountID>:role/<AmazonEKS_FSxN_CSI_DriverRole>'" \
--namespace trident --create-namespace
```

You can use the `helm list` command to review installation details such as name, namespace, chart, status, app version, and revision number.

```
helm list -n trident
```

NAME	NAMESPACE	REVISION	UPDATED
STATUS	CHART		APP VERSION
trident-operator	trident	1	2024-10-14 14:31:22.463122
+0300 IDT	deployed	trident-operator-100.2410.0	24.10.0

Install Trident via the EKS add-on

The Trident EKS add-on includes the latest security patches, bug fixes, and is validated by AWS to work with Amazon EKS. The EKS add-on enables you to consistently ensure that your Amazon EKS clusters are secure and stable and reduce the amount of work that you need to do in order to install, configure, and update add-ons.

Prerequisites

Ensure that you have the following before configuring the Trident add-on for AWS EKS:

- An Amazon EKS cluster account with add-on subscription
- AWS permissions to the AWS marketplace:
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",
"aws-marketplace:Unsubscribe"
- AMI type: Amazon Linux 2 (AL2_x86_64) or Amazon Linux 2 Arm(AL2_ARM_64)
- Node type: AMD or ARM
- An existing Amazon FSx for NetApp ONTAP file system

Enable the Trident add-on for AWS

eksctl

The following example commands install the Trident EKS add-on:

```
eksctl create addon --name netapp_trident-operator --cluster
<cluster_name> \
    --service-account-role-arn
arn:aws:iam::<account_id>:role/<role_name> --force
```

Management console

1. Open the Amazon EKS console at <https://console.aws.amazon.com/eks/home#/clusters>.
2. On the left navigation pane, click **Clusters**.
3. Click the name of the cluster that you want to configure the NetApp Trident CSI add-on for.
4. Click **Add-ons** and then click **Get more add-ons**.
5. On the **Select add-ons** page, do the following:
 - a. In the AWS Marketplace EKS-addons section, select the **Trident by NetApp** check box.
 - b. Click **Next**.
6. On the **Configure selected add-ons** settings page, do the following:
 - a. Select the **Version** you would like to use.
 - b. For **Select IAM role**, leave at **Not set**.
 - c. Expand the **Optional configuration settings**, follow the **Add-on configuration schema** and set the configurationValues parameter on the **Configuration values** section to the role-arn you created on the previous step (value should be in the following format:
eks.amazonaws.com/role-arn:
arn:aws:iam::464262061435:role/AmazonEKS_FSXN_CSI_DriverRole). If you select **Override** for the Conflict resolution method, one or more of the settings for the existing add-on can be overwritten with the Amazon EKS add-on settings. If you don't enable this option and there's a conflict with your existing settings, the operation fails. You can use the resulting error message to troubleshoot the conflict. Before selecting this option, make sure that the Amazon EKS add-on doesn't manage settings that you need to self-manage.
7. Choose **Next**.
8. On the **Review and add** page, choose **Create**.

After the add-on installation is complete, you see your installed add-on.

AWS CLI

1. Create the add-on.json file:

```
add-on.json
{
  "clusterName": "<eks-cluster>",
  "addonName": "netapp_trident-operator",
  "addonVersion": "v24.10.0-eksbuild.1",
  "serviceAccountRoleArn": "<arn:aws:iam::123456:role/astratrident-role>",
  "configurationValues": "{\"cloudIdentity\":\n    \"'eks.amazonaws.com/role-arn:\n    <arn:aws:iam::123456:role/astratrident-role>'\"},\n  \"cloudProvider\": \"AWS\"}"
}
```

2. Install the Trident EKS add-on

```
aws eks create-addon --cli-input-json file://add-on.json
```

Update the Trident EKS add-on

eksctl

- Check the current version of your FSxN Trident CSI add-on. Replace `my-cluster` with your cluster name.

```
eksctl get addon --name netapp_trident-operator --cluster my-cluster
```

Example output:

NAME	VERSION	STATUS	ISSUES
IAMROLE	UPDATE AVAILABLE CONFIGURATION VALUES		
netapp_trident-operator	v24.10.0-eksbuild.1	ACTIVE	0
	{"cloudIdentity": "'eks.amazonaws.com/role-arn': 'arn:aws:iam::139763910815:role/AmazonEKS_FSXN_CSI_DriverRole'"}		

- Update the add-on to the version returned under UPDATE AVAILABLE in the output of the previous step.

```
eksctl update addon --name netapp_trident-operator --version v24.10.0-eksbuild.1 --cluster my-cluster --force
```

If you remove the `--force` option and any of the Amazon EKS add-on settings conflict with your existing settings, then updating the Amazon EKS add-on fails; you receive an error message to help you resolve the conflict. Before specifying this option, make sure that the Amazon EKS add-on does not manage settings that you need to manage, because those settings are overwritten with this option.

For more information about other options for this setting, see [Addons](#).

For more information about Amazon EKS Kubernetes field management, see [Kubernetes field management](#).

Management console

1. Open the Amazon EKS console <https://console.aws.amazon.com/eks/home#/clusters>.
2. On the left navigation pane, click **Clusters**.
3. Click the name of the cluster that you want to update the NetApp Trident CSI add-on for.
4. Click the **Add-ons** tab.
5. Click **Trident by NetApp** and then click **Edit**.
6. On the **Configure Trident by NetApp** page, do the following:
 - a. Select the **Version** you would like to use.
 - b. Expand the **Optional configuration settings** and modify as needed.
 - c. Click **Save changes**.

AWS CLI

The following example updates the EKS add-on:

```
aws eks update-addon --cluster-name my-cluster netapp_trident-operator
vpc-cni --addon-version v24.6.1-eksbuild.1 \
--service-account-role-arn arn:aws:iam::111122223333:role/role-name
--configuration-values '{}' --resolve-conflicts --preserve
```

Uninstall/remove the Trident EKS add-on

You have two options for removing an Amazon EKS add-on:

- **Preserve add-on software on your cluster** – This option removes Amazon EKS management of any settings. It also removes the ability for Amazon EKS to notify you of updates and automatically update the Amazon EKS add-on after you initiate an update. However, it preserves the add-on software on your cluster. This option makes the add-on a self-managed installation, rather than an Amazon EKS add-on. With this option, there's no downtime for the add-on. Retain the `--preserve` option in the command to preserve the add-on.
- **Remove add-on software entirely from your cluster** – We recommend that you remove the Amazon EKS add-on from your cluster only if there are no resources on your cluster that are dependent on it. Remove the `--preserve` option from the `delete` command to remove the add-on.



If the add-on has an IAM account associated with it, the IAM account is not removed.

eksctl

The following command uninstalls the Trident EKS add-on:

```
eksctl delete addon --cluster K8s-arm --name netapp_trident-operator
```

Management console

1. Open the Amazon EKS console at <https://console.aws.amazon.com/eks/home#/clusters>.
2. In the left navigation pane, click **Clusters**.
3. Click the name of the cluster that you want to remove the NetApp Trident CSI add-on for.
4. Click the **Add-ons** tab and then click **Trident by NetApp**.*
5. Click **Remove**.
6. In the **Remove netapp_trident-operator confirmation** dialog, do the following:
 - a. If you want Amazon EKS to stop managing settings for the add-on, select **Preserve on cluster**. Do this if you want to retain the add-on software on your cluster so that you can manage all of the settings of the add-on on your own.
 - b. Enter **netapp_trident-operator**.
 - c. Click **Remove**.

AWS CLI

Replace `my-cluster` with the name of your cluster, and then run the following command.

```
aws eks delete-addon --cluster-name my-cluster --addon-name netapp_trident-
operator --preserve
```

Configure the Storage Backend

ONTAP SAN and NAS driver integration

To create a storage backend, you need to create a configuration file in either JSON or YAML format. The file needs to specify the type of storage you want (NAS or SAN), the file system, and SVM to get it from and how to authenticate with it. The following example shows how to define NAS-based storage and using an AWS secret to store the credentials to the SVM you want to use:

YAML

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nas
  namespace: trident
spec:
  version: 1
  storageDriverName: ontap-nas
  backendName: tbc-ontap-nas
  svm: svm-name
  aws:
    fsxFilesystemID: fs-xxxxxxxxxx
  credentials:
    name: "arn:aws:secretsmanager:us-west-2:xxxxxxxx:secret:secret-
name"
    type: awsarn
```

JSON

```
{
  "apiVersion": "trident.netapp.io/v1",
  "kind": "TridentBackendConfig",
  "metadata": {
    "name": "backend-tbc-ontap-nas",
    "namespace": "trident"
  },
  "spec": {
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "tbc-ontap-nas",
    "svm": "svm-name",
    "aws": {
      "fsxFilesystemID": "fs-xxxxxxxxxx"
    },
    "managementLIF": null,
    "credentials": {
      "name": "arn:aws:secretsmanager:us-west-2:xxxxxxxx:secret:secret-
name",
      "type": "awsarn"
    }
  }
}
```

Run the following commands to create and validate the Trident Backend Configuration (TBC):

- Create trident backend configuration (TBC) from yaml file and run the following command:

```
kubectl create -f backendconfig.yaml -n trident
```

```
tridentbackendconfig.trident.netapp.io/backend-tbc-ontap-nas created
```

- Validate the trident backend configuration (TBC) was created successfully:

```
Kubectl get tbc -n trident
```

NAME	PHASE	STATUS	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-nas	b9ff-f96d916ac5e9	Bound	tbc-ontap-nas	933e0071-66ce-4324-

FSx for ONTAP driver details

You can integrate Trident with Amazon FSx for NetApp ONTAP using the following drivers:

- `ontap-san`: Each PV provisioned is a LUN within its own Amazon FSx for NetApp ONTAP volume. Recommended for block storage.
- `ontap-nas`: Each PV provisioned is a full Amazon FSx for NetApp ONTAP volume. Recommended for NFS and SMB.
- `ontap-san-economy`: Each PV provisioned is a LUN with a configurable number of LUNs per Amazon FSx for NetApp ONTAP volume.
- `ontap-nas-economy`: Each PV provisioned is a qtree, with a configurable number of qtrees per Amazon FSx for NetApp ONTAP volume.
- `ontap-nas-flexgroup`: Each PV provisioned is a full Amazon FSx for NetApp ONTAP FlexGroup volume.

For driver details, refer to [NAS drivers](#) and [SAN drivers](#).

Once the configuration file has been created, run this command to create it within your EKS:

```
kubectl create -f configuration_file
```

To verify the status, run this command:

```
kubectl get tbc -n trident
```

NAME	BACKEND NAME	BACKEND UUID
PHASE	STATUS	
backend-fsx-ontap-nas	backend-fsx-ontap-nas	7a551921-997c-4c37-a1d1-
f2f4c87fa629	Bound	Success

Backend advanced configuration and examples

See the following table for the backend configuration options:

Parameter	Description	Example
version		Always 1
storageDriverName	Name of the storage driver	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, ontap-san-economy
backendName	Custom name or the storage backend	Driver name + “_” + dataLIF
managementLIF	<p>IP address of a cluster or SVM management LIF</p> <p>A fully-qualified domain name (FQDN) can be specified.</p> <p>Can be set to use IPv6 addresses if Trident was installed using the IPv6 flag. IPv6 addresses must be defined in square brackets, such as [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555].</p> <p>If you provide the <code>fsxFilesystemID</code> under the <code>aws</code> field, you need not to provide the <code>managementLIF</code> because Trident retrieves the SVM <code>managementLIF</code> information from AWS. So, you must provide credentials for a user under the SVM (For example: <code>vsadmin</code>) and the user must have the <code>vsadmin</code> role.</p>	“10.0.0.1”, “[2001:1234:abcd::fefe]”

Parameter	Description	Example
dataLIF	<p>IP address of protocol LIF.</p> <p>ONTAP NAS drivers: We recommend specifying dataLIF. If not provided, Trident fetches data LIFs from the SVM. You can specify a fully-qualified domain name (FQDN) to be used for the NFS mount operations, allowing you to create a round-robin DNS to load-balance across multiple data LIFs. Can be changed after initial setting. Refer to [Update dataLIF after initial configuration].</p> <p>ONTAP SAN drivers: Do not specify for iSCSI. Trident uses ONTAP Selective LUN Map to discover the iSCSI LIFs needed to establish a multi path session. A warning is generated if dataLIF is explicitly defined.</p> <p>Can be set to use IPv6 addresses if Trident was installed using the IPv6 flag. IPv6 addresses must be defined in square brackets, such as [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555].</p>	
autoExportPolicy	<p>Enable automatic export policy creation and updating [Boolean].</p> <p>Using the <code>autoExportPolicy</code> and <code>autoExportCIDRs</code> options, Trident can manage export policies automatically.</p>	false
autoExportCIDRs	<p>List of CIDRs to filter Kubernetes' node IPs against when <code>autoExportPolicy</code> is enabled.</p> <p>Using the <code>autoExportPolicy</code> and <code>autoExportCIDRs</code> options, Trident can manage export policies automatically.</p>	"["0.0.0.0/0", "::/0"]"
labels	Set of arbitrary JSON-formatted labels to apply on volumes	""
clientCertificate	Base64-encoded value of client certificate. Used for certificate-based auth	""

Parameter	Description	Example
clientPrivateKey	Base64-encoded value of client private key. Used for certificate-based auth	""
trustedCACertificate	Base64-encoded value of trusted CA certificate. Optional. Used for certificate-based authentication.	""
username	Username to connect to the cluster or SVM. Used for credential-based authentication. For example, vsadmin.	
password	Password to connect to the cluster or SVM. Used for credential-based authentication.	
svm	Storage virtual machine to use	Derived if an SVM managementLIF is specified.
storagePrefix	Prefix used when provisioning new volumes in the SVM. Cannot be modified after creation. To update this parameter, you will need to create a new backend.	trident
limitAggregateUsage	Do not specify for Amazon FSx for NetApp ONTAP. The provided fsxadmin and vsadmin do not contain the permissions required to retrieve aggregate usage and limit it using Trident.	Do not use.
limitVolumeSize	Fail provisioning if requested volume size is above this value. Also restricts the maximum size of the volumes it manages for qtrees and LUNs, and the qtreesPerFlexvol option allows customizing the maximum number of qtrees per FlexVol.	"" (not enforced by default)
lunsPerFlexvol	Maximum LUNs per Flexvol, must be in range [50, 200]. SAN only.	"100"

Parameter	Description	Example
debugTraceFlags	<p>Debug flags to use when troubleshooting. Example, <code>{"api":false, "method":true}</code></p> <p>Do not use debugTraceFlags unless you are troubleshooting and require a detailed log dump.</p>	null
nfsMountOptions	<p>Comma-separated list of NFS mount options.</p> <p>The mount options for Kubernetes-persistent volumes are normally specified in storage classes, but if no mount options are specified in a storage class, Trident will fall back to using the mount options specified in the storage backend's configuration file.</p> <p>If no mount options are specified in the storage class or the configuration file, Trident will not set any mount options on an associated persistent volume.</p>	""
nasType	<p>Configure NFS or SMB volumes creation.</p> <p>Options are <code>nfs</code>, <code>smb</code>, or <code>null</code>.</p> <p>Must set to smb for SMB volumes. Setting to <code>null</code> defaults to NFS volumes.</p>	<code>nfs</code>
qtreesPerFlexvol	Maximum Qtrees per FlexVol, must be in range [50, 300]	<code>"200"</code>
smbShare	<p>You can specify one of the following: the name of an SMB share created using the Microsoft Management Console or ONTAP CLI or a name to allow Trident to create the SMB share.</p> <p>This parameter is required for Amazon FSx for ONTAP backends.</p>	<code>smb-share</code>

Parameter	Description	Example
useREST	<p>Boolean parameter to use ONTAP REST APIs. Tech preview</p> <p>useREST is provided as a tech preview that is recommended for test environments and not for production workloads. When set to true, Trident will use ONTAP REST APIs to communicate with the backend.</p> <p>This feature requires ONTAP 9.11.1 and later. In addition, the ONTAP login role used must have access to the <code>ontap</code> application. This is satisfied by the pre-defined <code>vsadmin</code> and <code>cluster-admin</code> roles.</p>	false
aws	<p>You can specify the following in the configuration file for AWS FSx for ONTAP:</p> <ul style="list-style-type: none"> - <code>fsxFilesystemID</code>: Specify the ID of the AWS FSx file system. - <code>apiRegion</code>: AWS API region name. - <code>apikey</code>: AWS API key. - <code>secretKey</code>: AWS secret key. 	"" "" ""
credentials	<p>Specify the FSx SVM credentials to store in AWS Secret Manager.</p> <ul style="list-style-type: none"> - <code>name</code>: Amazon Resource Name (ARN) of the secret, which contains the credentials of SVM. - <code>type</code>: Set to <code>awsarn</code>. Refer to Create an AWS Secrets Manager secret for more information. 	

Backend configuration options for provisioning volumes

You can control default provisioning using these options in the `defaults` section of the configuration. For an example, see the configuration examples below.

Parameter	Description	Default
<code>spaceAllocation</code>	Space-allocation for LUNs	true
<code>spaceReserve</code>	Space reservation mode; “none” (thin) or “volume” (thick)	none
<code>snapshotPolicy</code>	Snapshot policy to use	none

Parameter	Description	Default
qosPolicy	<p>QoS policy group to assign for volumes created. Choose one of qosPolicy or adaptiveQosPolicy per storage pool or backend.</p> <p>Using QoS policy groups with Trident requires ONTAP 9.8 or later.</p> <p>You should use a non-shared QoS policy group and ensuring the policy group is applied to each constituent individually. A shared QoS policy group enforces the ceiling for the total throughput of all workloads.</p>	""
adaptiveQosPolicy	<p>Adaptive QoS policy group to assign for volumes created. Choose one of qosPolicy or adaptiveQosPolicy per storage pool or backend.</p> <p>Not supported by ontap-nas-economy.</p>	""
snapshotReserve	Percentage of volume reserved for snapshots "0"	If snapshotPolicy is none, else ""
splitOnClone	Split a clone from its parent upon creation	false
encryption	<p>Enable NetApp Volume Encryption (NVE) on the new volume; defaults to false. NVE must be licensed and enabled on the cluster to use this option.</p> <p>If NAE is enabled on the backend, any volume provisioned in Trident will be NAE enabled.</p> <p>For more information, refer to: How Trident works with NVE and NAE.</p>	false
luksEncryption	<p>Enable LUKS encryption. Refer to Use Linux Unified Key Setup (LUKS).</p> <p>SAN only.</p>	""
tieringPolicy	Tiering policy to use none	snapshot-only for pre-ONTAP 9.5 SVM-DR configuration

Parameter	Description	Default
unixPermissions	Mode for new volumes. Leave empty for SMB volumes.	""
securityStyle	Security style for new volumes. NFS supports <code>mixed</code> and <code>unix</code> security styles. SMB supports <code>mixed</code> and <code>ntfs</code> security styles.	NFS default is <code>unix</code> . SMB default is <code>ntfs</code> .

Prepare to provision SMB volumes

You can provision SMB volumes using the `ontap-nas` driver. Before you complete [ONTAP SAN and NAS driver integration](#) complete the following steps.

Before you begin

Before you can provision SMB volumes using the `ontap-nas` driver, you must have the following.

- A Kubernetes cluster with a Linux controller node and at least one Windows worker node running Windows Server 2019. Trident supports SMB volumes mounted to pods running on Windows nodes only.
- At least one Trident secret containing your Active Directory credentials. To generate secret `smbcreds`:

```
kubectl create secret generic smbcreds --from-literal username=user
--from-literal password='password'
```

- A CSI proxy configured as a Windows service. To configure a `csi-proxy`, refer to [GitHub: CSI Proxy](#) or [GitHub: CSI Proxy for Windows](#) for Kubernetes nodes running on Windows.

Steps

1. Create SMB shares. You can create the SMB admin shares in one of two ways either using the [Microsoft Management Console Shared Folders snap-in](#) or using the ONTAP CLI. To create the SMB shares using the ONTAP CLI:

- a. If necessary, create the directory path structure for the share.

The `vserver cifs share create` command checks the path specified in the `-path` option during share creation. If the specified path does not exist, the command fails.

- b. Create an SMB share associated with the specified SVM:

```
vserver cifs share create -vserver vserver_name -share-name
share_name -path path [-share-properties share_properties,...]
[other_attributes] [-comment text]
```

- c. Verify that the share was created:

```
vserver cifs share show -share-name share_name
```



Refer to [Create an SMB share](#) for full details.

2. When creating the backend, you must configure the following to specify SMB volumes. For all FSx for ONTAP backend configuration options, refer to [FSx for ONTAP configuration options and examples](#).

Parameter	Description	Example
smbShare	<p>You can specify one of the following: the name of an SMB share created using the Microsoft Management Console or ONTAP CLI or a name to allow Trident to create the SMB share.</p> <p>This parameter is required for Amazon FSx for ONTAP backends.</p>	smb-share
nasType	<p>Must set to smb. If null, defaults to nfs.</p>	smb
securityStyle	<p>Security style for new volumes.</p> <p>Must be set to ntfs or mixed for SMB volumes.</p>	ntfs or mixed for SMB volumes
unixPermissions	<p>Mode for new volumes. Must be left empty for SMB volumes.</p>	""

Configure a storage class and PVC

Configure a Kubernetes StorageClass object and create the storage class to instruct Trident how to provision volumes. Create a PersistentVolume (PV) and a PersistentVolumeClaim (PVC) that uses the configured Kubernetes StorageClass to request access to the PV. You can then mount the PV to a pod.

Create a storage class

Configure a Kubernetes StorageClass object

The [Kubernetes StorageClass object](#) identifies Trident as the provisioner that is used for that class instructs Trident how to provision a volume. For example:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-gold
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
```

Refer to [Kubernetes and Trident objects](#) for details on how storage classes interact with the `PersistentVolumeClaim` and parameters for controlling how Trident provisions volumes.

Create a storage class

Steps

1. This is a Kubernetes object, so use `kubectl` to create it in Kubernetes.

```
kubectl create -f storage-class-ontapnas.yaml
```

2. You should now see a **basic-csi** storage class in both Kubernetes and Trident, and Trident should have discovered the pools on the backend.

```
kubectl get sc basic-csi
NAME      PROVISIONER          AGE
basic-csi  csi.trident.netapp.io  15h
```

Create the PV and PVC

A [PersistentVolume](#) (PV) is a physical storage resource provisioned by the cluster administrator on a Kubernetes cluster. The [PersistentVolumeClaim](#) (PVC) is a request for access to the PersistentVolume on the cluster.

The PVC can be configured to request storage of a certain size or access mode. Using the associated StorageClass, the cluster administrator can control more than PersistentVolume size and access mode—such as performance or service level.

After you create the PV and PVC, you can mount the volume in a pod.

Sample manifests

PersistentVolume sample manifest

This sample manifest shows a basic PV of 10Gi that is associated with StorageClass `basic-csi`.

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: pv-storage
  labels:
    type: local
spec:
  storageClassName: basic-csi
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteMany
  hostPath:
    path: "/my/host/path"
```

PersistentVolumeClaim sample manifests

These examples show basic PVC configuration options.

PVC with RWX access

This example shows a basic PVC with RWX access that is associated with a StorageClass named basic-csi.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-storage
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 1Gi
  storageClassName: basic-csi
```

PVC with NVMe/TCP

This example shows a basic PVC for NVMe/TCP with RWO access that is associated with a StorageClass named protection-gold.

```
---
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-san-nvme
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 300Mi
  storageClassName: protection-gold
```

Create the PV and PVC

Steps

1. Create the PV.

```
kubectl create -f pv.yaml
```

2. Verify the PV status.

```
kubectl get pv
NAME          CAPACITY  ACCESS MODES  RECLAIM POLICY  STATUS     CLAIM
STORAGECLASS  REASON    AGE
pv-storage    4Gi       RWO          Retain        Available
7s
```

3. Create the PVC.

```
kubectl create -f pvc.yaml
```

4. Verify the PVC status.

```
kubectl get pvc
NAME          STATUS  VOLUME      CAPACITY  ACCESS MODES  STORAGECLASS  AGE
pvc-storage  Bound   pv-name  2Gi       RWO          5m
```

Refer to [Kubernetes and Trident objects](#) for details on how storage classes interact with the `PersistentVolumeClaim` and parameters for controlling how Trident provisions volumes.

Trident attributes

These parameters determine which Trident-managed storage pools should be utilized to provision volumes of a given type.

Attribute	Type	Values	Offer	Request	Supported by
media ¹	string	hdd, hybrid, ssd	Pool contains media of this type; hybrid means both	Media type specified	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, solidfire-san
provisioningType	string	thin, thick	Pool supports this provisioning method	Provisioning method specified	thick: all ontap; thin: all ontap & solidfire-san

Attribute	Type	Values	Offer	Request	Supported by
backendType	string	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, solidfire-san, gcp-cvs, azure-netapp-files, ontap-san-economy	Pool belongs to this type of backend	Backend specified	All drivers
snapshots	bool	true, false	Pool supports volumes with snapshots	Volume with snapshots enabled	ontap-nas, ontap-san, solidfire-san, gcp-cvs
clones	bool	true, false	Pool supports cloning volumes	Volume with clones enabled	ontap-nas, ontap-san, solidfire-san, gcp-cvs
encryption	bool	true, false	Pool supports encrypted volumes	Volume with encryption enabled	ontap-nas, ontap-nas-economy, ontap-nas-flexgroups, ontap-san
IOPS	int	positive integer	Pool is capable of guaranteeing IOPS in this range	Volume guaranteed these IOPS	solidfire-san

¹: Not supported by ONTAP Select systems

Deploy sample application

Deploy sample application.

Steps

1. Mount the volume in a pod.

```
kubectl create -f pv-pod.yaml
```

These examples show basic configurations to attach the PVC to a pod:

Basic configuration:

```

kind: Pod
apiVersion: v1
metadata:
  name: pv-pod
spec:
  volumes:
    - name: pv-storage
      persistentVolumeClaim:
        claimName: basic
  containers:
    - name: pv-container
      image: nginx
      ports:
        - containerPort: 80
          name: "http-server"
      volumeMounts:
        - mountPath: "/my/mount/path"
          name: pv-storage

```



You can monitor the progress using `kubectl get pod --watch`.

2. Verify that the volume is mounted on `/my/mount/path`.

```
kubectl exec -it pv-pod -- df -h /my/mount/path
```

Filesystem	Size
Used Avail Use% Mounted on	
192.168.188.78:/trident_pvc_ae45ed05_3ace_4e7c_9080_d2a83ae03d06	1.1G
320K 1.0G 1% /my/mount/path	

You can now delete the Pod. The Pod application will no longer exist, but the volume will remain.

```
kubectl delete pod pv-pod
```

Configure the Trident EKS add-on on an EKS cluster

NetApp Trident streamlines Amazon FSx for NetApp ONTAP storage management in Kubernetes to enable your developers and administrators focus on application deployment. The NetApp Trident EKS add-on includes the latest security patches, bug fixes, and is validated by AWS to work with Amazon EKS. The EKS add-on enables you

to consistently ensure that your Amazon EKS clusters are secure and stable and reduce the amount of work that you need to do in order to install, configure, and update add-ons.

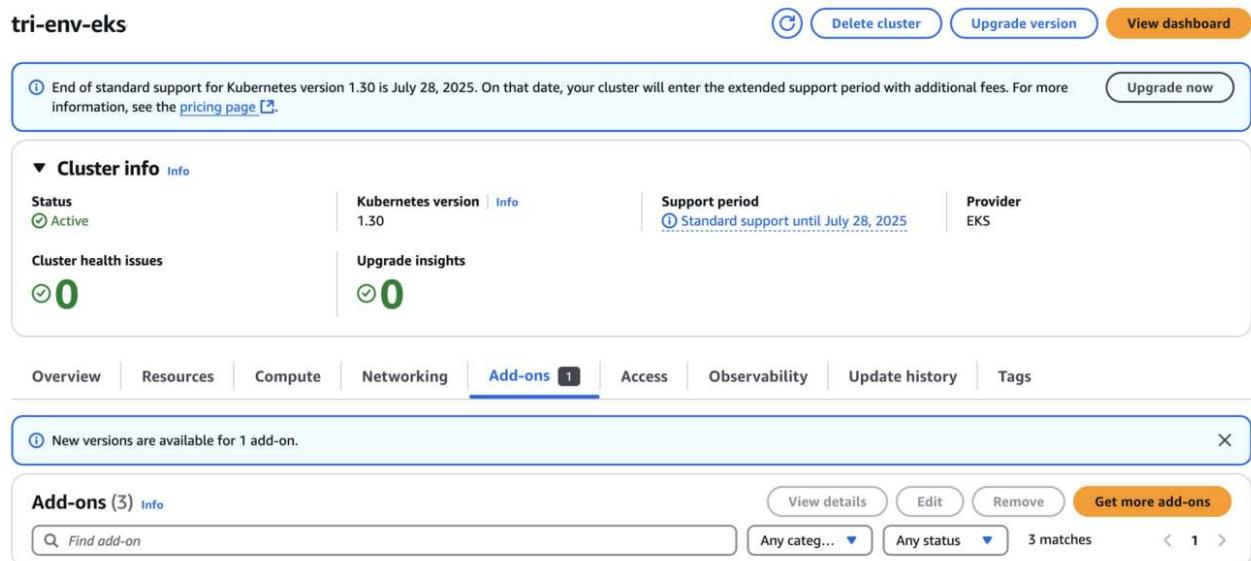
Prerequisites

Ensure that you have the following before configuring the Trident add-on for AWS EKS:

- An Amazon EKS cluster account with permissions to work with add-ons. Refer to [Amazon EKS add-ons](#).
- AWS permissions to the AWS marketplace:
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",
"aws-marketplace:Unsubscribe"
- AMI type: Amazon Linux 2 (AL2_x86_64) or Amazon Linux 2 Arm(AL2_ARM_64)
- Node type: AMD or ARM
- An existing Amazon FSx for NetApp ONTAP file system

Steps

1. Make sure to create IAM role and AWS secret to enable EKS pods to access AWS resources. For instructions, see [Create an IAM role and AWS Secret](#).
2. On your EKS Kubernetes cluster, navigate to the **Add-ons** tab.



tri-env-eks

Delete cluster Upgrade version View dashboard

End of standard support for Kubernetes version 1.30 is July 28, 2025. On that date, your cluster will enter the extended support period with additional fees. For more information, see the [pricing page](#). [Upgrade now](#)

Cluster info [Info](#)

Status Active	Kubernetes version Info 1.30	Support period Info Standard support until July 28, 2025	Provider EKS
Cluster health issues 0	Upgrade insights 0		

Overview Resources Compute Networking **Add-ons (1)** Access Observability Update history Tags

New versions are available for 1 add-on.

Add-ons (3) [Info](#)

	View details	Edit	Remove	Get more add-ons
<input type="text"/> Find add-on	Any categ...	Any status	3 matches	< 1 >

3. Go to **AWS Marketplace add-ons** and choose the **storage** category.

AWS Marketplace add-ons (1)

Discover, subscribe to and configure EKS add-ons to enhance your EKS clusters.

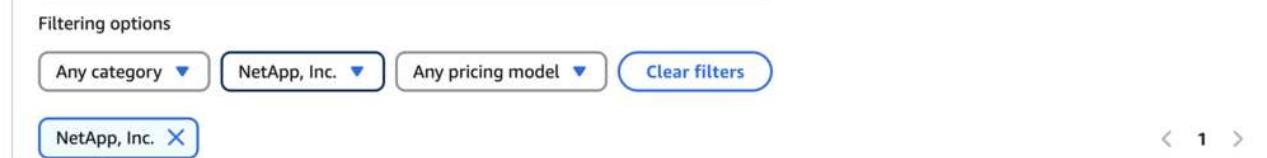
Find add-on

Filtering options

Any category NetApp, Inc. Any pricing model Clear filters

NetApp, Inc.

< 1 >



NetApp Trident

NetApp Trident streamlines Amazon FSx for NetApp ONTAP storage management in Kubernetes to let your developers and administrators focus on application deployment. FSx for ONTAP flexibility, scalability, and integration capabilities make it the ideal choice for organizations seeking efficient containerized storage workflows. [Product details](#)

Standard Contract

Category
storage

Listed by
[NetApp, Inc.](#)

Supported versions
1.31, 1.30, 1.29, 1.28,
1.27, 1.26, 1.25, 1.24,
1.23

Pricing starting at
[View pricing details](#)

[Cancel](#)

[Next](#)

4. Locate **NetApp Trident** and select the checkbox for the Trident add-on, and click **Next**.
5. Choose the desired version of the add-on.

NetApp Trident

[Remove add-on](#)

Listed by  NetApp	Category storage	Status  Ready to install
---	---------------------	---

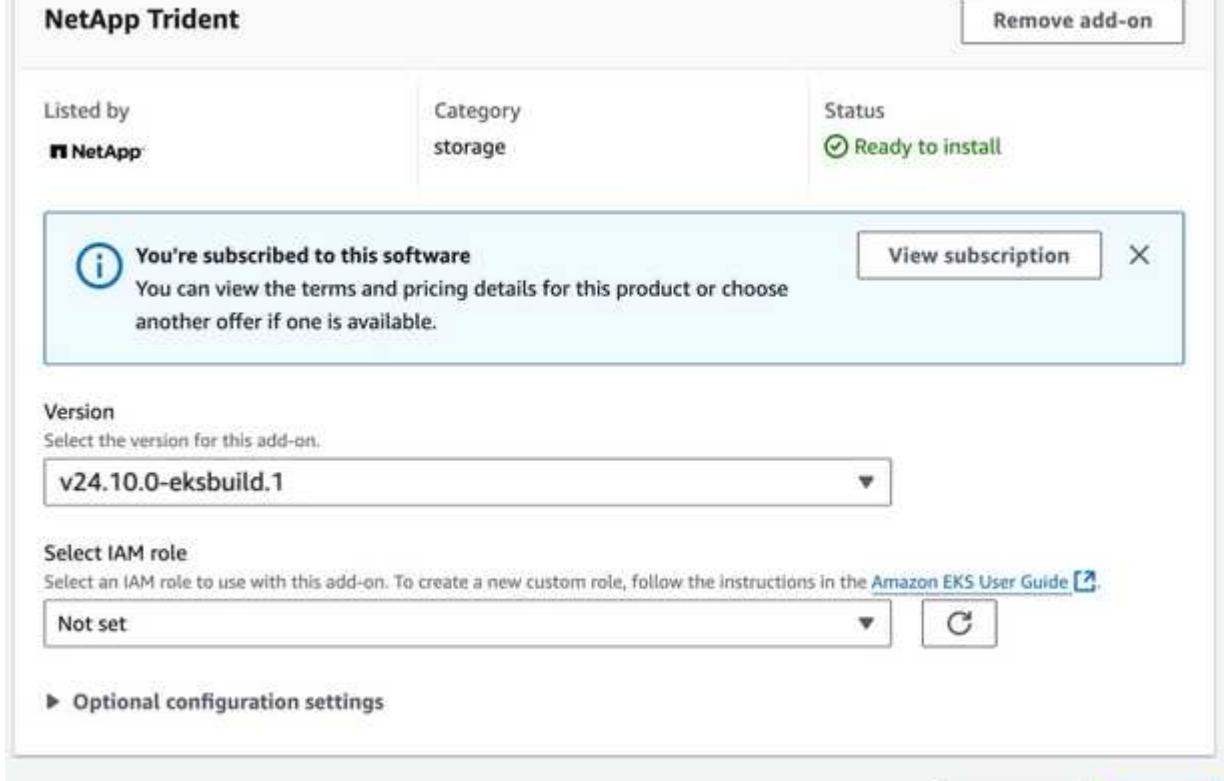
i You're subscribed to this software
You can view the terms and pricing details for this product or choose another offer if one is available. [View subscription](#) [X](#)

Version
Select the version for this add-on.

Select IAM role
Select an IAM role to use with this add-on. To create a new custom role, follow the instructions in the [Amazon EKS User Guide](#). [C](#)

Optional configuration settings

[Cancel](#) [Previous](#) [Next](#)



6. Select the IAM role option to inherit from the node.

Review and add

Step 1: Select add-ons

Edit

Selected add-ons (1)

Find add-on

< 1 >

Add-on name	Type	Status
netapp_trident-operator	storage	Ready to install

Step 2: Configure selected add-ons settings

Edit

Selected add-ons version (1)

< 1 >

Add-on name	Version	IAM role for service account (IRSA)
netapp_trident-operator	v24.10.0-eksbuild.1	Not set

EKS Pod Identity (0)

< 1 >

Add-on name	IAM role	Service account
No Pod Identity associations		

None of the selected add-on(s) have Pod Identity associations.

Cancel

Previous

Create

7. Configure any Optional configuration settings as required and select **Next**.

Follow the **Add-on configuration schema** and set the Configuration Values parameter on the **Configuration values** section to the role-arn you created on the previous step(Step 1) (value should be in the following format: `eks.amazonaws.com/role-arn`:

`arn:aws:iam::464262061435:role/AmazonEKS_FSXN_CSI_DriverRole`).

NOTE: If you select **Override** for the Conflict resolution method, one or more of the settings for the existing add-on can be overwritten with the Amazon EKS add-on settings. If you don't enable this option and there's a conflict with your existing settings, the operation fails. You can use the resulting error message to troubleshoot the conflict. Before selecting this option, make sure that the Amazon EKS add-on doesn't manage settings that you need to self-manage.

▼ Optional configuration settings

Add-on configuration schema

Refer to the JSON schema below. The configuration values entered in the code editor will be validated against this schema.

```
        "default": "",  
        "examples": [  
            {  
                "cloudIdentity": ""  
            }  
        ],  
        "properties": {  
            "cloudIdentity": {  
                "default": "",  
                "examples": [  
                    ""  
                ],  
                "title": "The cloudIdentity Schema",  
                "type": "string"  
            }  
        }  
    }  
}
```

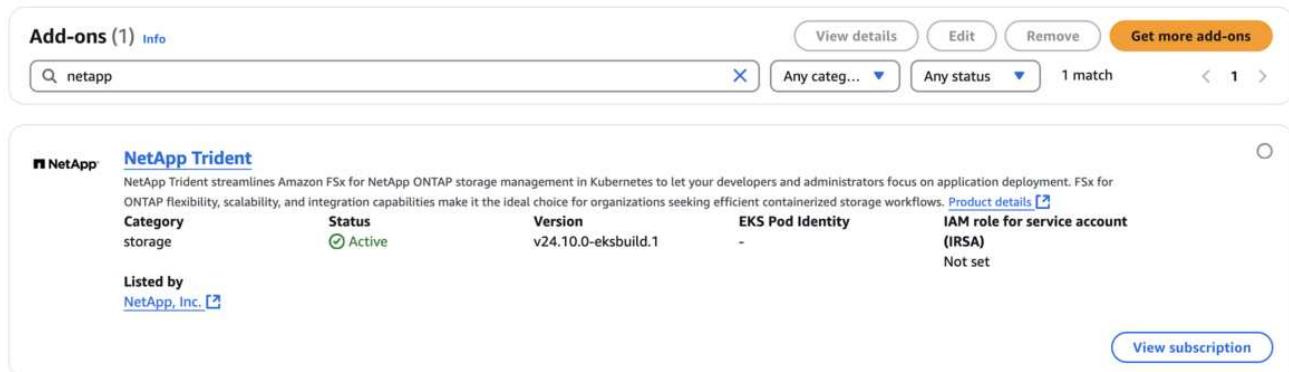
Configuration values | [Info](#)

Specify any additional JSON or YAML configurations that should be applied to the add-on.

```
1 ▾ {  
2     "cloudIdentity": "eks.amazonaws.com/role-arn: arn:aws:iam  
        ::186785786363:role/tri-env-eks-trident-controller-role"  
3 }
```

8. Select **Create**.

9. Verify that the status of the add-on is *Active*.



The screenshot shows the AWS EKS Add-ons console. At the top, there is a search bar with the text "netapp" and a "View details" button. Below the search bar, the "Add-ons (1)" section is shown. The single add-on listed is "NetApp Trident". The details for this add-on are as follows:

Category	Status	Version	EKS Pod Identity	IAM role for service account
storage	Active	v24.10.0-eksbuild.1	-	(IRSA) Not set

Below the table, it says "Listed by [NetApp, Inc.](#)". At the bottom right of the add-on card, there is a "View subscription" button.

10. Run the following command to verify that Trident is properly installed on the cluster:

```
kubectl get pods -n trident
```

11. Continue the setup and configure the storage backend. For information, see [Configure the Storage Backend](#).

Install/uninstall the Trident EKS add-on using CLI

Install the NetApp Trident EKS add-on using CLI:

The following example command installs the Trident EKS add-on:

```
eksctl create addon --name aws-ebs-csi-driver --cluster <cluster_name>
--service-account-role-arn arn:aws:iam::<account_id>:role/<role_name>
--force
```

Uninstall the NetApp Trident EKS add-on using CLI:

The following command uninstalls the Trident EKS add-on:

```
eksctl delete addon --cluster K8s-arm --name netapp_trident-operator
```

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.