Amazon FSx for NetApp ONTAP

Astra Trident

NetApp
December 03, 2024

Table of Contents

Amazon FSx for NetApp ONTAP

Use Astra 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 Astra 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 Astra 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 Astra Trident requirements, to integrate FSx for ONTAP with Astra 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 Astra Trident EKS add-on.
  - Astra Trident supports SMB volumes mounted to pods running on Windows nodes only. Refer to Prepare to provision SMB volumes for details.
- Prior to Astra 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 Astra 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 Astra Trident, then you can omit specifying the `apiRegion`, `apiKey`, and `secretKey` fields to Astra Trident explicitly. For more information, refer to FSx for ONTAP configuration options and examples.

Authentication

Astra 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.

Astra 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 Astra Trident.

- Certificate-based: Astra 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

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 Astra Trident CSI credentials:

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

Create IAM Policy

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

```
aws iam create-policy --policy-name AmazonFSxNCSIDriverPolicy --policy-document
```

file://policy.json --description "This policy grants access to Trident CSI to FSxN and 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 and IAM role for the service account

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_FSxN_CSI_DriverRole> --role-only
--attach-policy-arn arn:aws:iam::aws:policy/service-
role/AmazonFSxNCSIDriverPolicy --approve
```

Install Astra Trident

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

deployment.

You can install Astra 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
https://docs.aws.amazon.com/eks/latest/userguide/csi-snapshot-
controller.html.
```

Install Astra Trident via helm

1. Download the Astra Trident installer package

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

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

   ```
   export CP="AWS"
   export CI="'eks.amazonaws.com/role-arn:
   arn:aws:iam::<accountID>:role/<AmazonEKS_FSxN_CSI_DriverRole>'"
   ```

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

   ```
   helm install trident trident-operator-100.2406.0.tgz --set
   cloudProvider=$CP --set cloudIdentity=$CI --namespace trident
   ```

   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-100.2406.1    24.06.1
```

Install Astra Trident via the EKS add-on

The Astra 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 Astra 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 Astra Trident add-on for AWS

EKS cluster

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

```
eksctl create addon --cluster clusterName --name netapp_trident-operator
--version v24.6.1-eksbuild
eksctl create addon --cluster clusterName --name netapp_trident-operator
--version v24.6.1-eksbuild.1 (with a dedicated version)
```

When you configure the optional parameter `cloudIdentity`, ensure that you specify `cloudProvider` while installing Trident using the EKS add-on.

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 **S*elect add-ons** page, do the following:
   a. In the AWS Marketplace EKS-addons section, select the **Astra 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.

      When you configure the optional parameter `cloudIdentity`, ensure that you specify `cloudProvider` while installing Trident using the EKS add-on.

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": "v24.6.1-eksbuild.1",
   "serviceAccountRoleArn": "arn:aws:iam::123456:role/astratrident-
role",
   "configurationValues": "{"cloudIdentity":
"'eks.amazonaws.com/role-arn: arn:aws:iam::123456:role/astratrident-
role'",
   "cloudProvider": "AWS"}"
}
```

When you configure the optional parameter `cloudIdentity`, ensure that you specify `AWS` as the `cloudProvider` while installing Trident using the EKS add-on.

2. Install the Astra Trident EKS add-on"

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

Update the Astra Trident EKS add-on

EKS cluster

- 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    v24.6.1-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 v24.6.1-
  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 **Astra Trident by NetApp** and then click **Edit**.
6. On the **Configure Astra Trident by NetApp** page, do the following:
   a. Select the **Version** you would like to use.
   b. (Optional) You can 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 Astra 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.

EKS cluster

The following command uninstalls the Astra 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 **Astra 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

You can create a backend file using the SVM credentials (username and password) stored in AWS Secret Manager as shown in this example:

YAML

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nas
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"
  },
  "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"
    }
  }
}
```

For information about creating backends, refer to these pages:

- Configure a backend with ONTAP NAS drivers
- Configure a backend with ONTAP SAN drivers

FSx for ONTAP driver details

You can integrate Astra 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.

Example configurations

Configuration for AWS FSx for ONTAP with secret manager

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

Configuration of storage class for SMB volumes

Using `nasType`, `node-stage-secret-name`, and `node-stage-secret-namespace`, you can specify an SMB volume and provide the required Active Directory credentials. SMB volumes are supported using the `ontap-nas` driver only.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: nas-smb-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: "default"
```

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

Parameter	Description	Example
`managementLIF`	IP address of a cluster or SVM management LIF A fully-qualified domain name (FQDN) can be specified. Can be set to use IPv6 addresses if Astra 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]. If you provide the `fsxFilesystemID` under the `aws` field, you need not to provide the `managementLIF` because Astra Trident retrieves the SVM `managementLIF` information from AWS. So, you must provide credentials for a user under the SVM (For example: vsadmin) and the user must have the `vsadmin` role.	“10.0.0.1”, “[2001:1234:abcd::fefe]”

Parameter	Description	Example
`dataLIF`	IP address of protocol LIF. **ONTAP NAS drivers**: We recommend specifying dataLIF. If not provided, Astra 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]. **ONTAP SAN drivers**: Do not specify for iSCSI. Astra Trident uses ONTAP Selective LUN Map to discover the iSCI LIFs needed to establish a multi path session. A warning is generated if dataLIF is explicitly defined. Can be set to use IPv6 addresses if Astra 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].	
`autoExportPolicy`	Enable automatic export policy creation and updating [Boolean]. Using the `autoExportPolicy` and `autoExportCIDRs` options, Astra Trident can manage export policies automatically.	`false`
`autoExportCIDRs`	List of CIDRs to filter Kubernetes' node IPs against when `autoExportPolicy` is enabled. Using the `autoExportPolicy` and `autoExportCIDRs` options, Astra Trident can manage export policies automatically.	"[“0.0.0.0/0”, “::/0”]"
`labels`	Set of arbitrary JSON-formatted labels to apply on volumes	""

Parameter	Description	Example
`clientCertificate`	Base64-encoded value of client certificate. Used for certificate-based auth	""
`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 Astra 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`	Debug flags to use when troubleshooting. Example, {“api”:false, “method”:true} Do not use `debugTraceFlags` unless you are troubleshooting and require a detailed log dump.	null
`nfsMountOptions`	Comma-separated list of NFS mount options. The mount options for Kubernetes-persistent volumes are normally specified in storage classes, but if no mount options are specified in a storage class, Astra Trident will fall back to using the mount options specified in the storage backend’s configuration file. If no mount options are specified in the storage class or the configuration file, Astra Trident will not set any mount options on an associated persistent volume.	""
`nasType`	Configure NFS or SMB volumes creation. Options are `nfs`, `smb`, or null. **Must set to `smb` for SMB volumes.** Setting to null defaults to NFS volumes.	`nfs`
`qtreesPerFlexvol`	Maximum Qtrees per FlexVol, must be in range [50, 300]	`"200"`
`smbShare`	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 Astra Trident to create the SMB share. This parameter is required for Amazon FSx for ONTAP backends.	`smb-share`

Parameter	Description	Example
useREST	Boolean parameter to use ONTAP REST APIs. **Tech preview** useREST is provided as a **tech preview** that is recommended for test environments and not for production workloads. When set to true, Astra Trident will use ONTAP REST APIs to communicate with the backend. This feature requires ONTAP 9.11.1 and later. In addition, the ONTAP login role used must have access to the ontap application. This is satisfied by the pre-defined vsadmin and cluster-admin roles.	false
aws	You can specify the following in the configuration file for AWS FSx for ONTAP: - fsxFilesystemID: Specify the ID of the AWS FSx file system. - apiRegion: AWS API region name. - apikey: AWS API key. - secretKey: AWS secret key.	"" "" ""
credentials	Specify the FSx SVM credentials to store in AWS Secret Manager. - name: Amazon Resource Name (ARN) of the secret, which contains the credentials of SVM. - type: Set to awsarn. 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
spaceAllocation	Space-allocation for LUNs	true
spaceReserve	Space reservation mode; “none” (thin) or “volume” (thick)	none
snapshotPolicy	Snapshot policy to use	none

Parameter	Description	Default
`qosPolicy`	QoS policy group to assign for volumes created. Choose one of qosPolicy or adaptiveQosPolicy per storage pool or backend. Using QoS policy groups with Astra Trident requires ONTAP 9.8 or later. We recommend using a non-shared QoS policy group and ensuring the policy group is applied to each constituent individually. A shared QoS policy group will enforce the ceiling for the total throughput of all workloads.	“”
`adaptiveQosPolicy`	Adaptive QoS policy group to assign for volumes created. Choose one of qosPolicy or adaptiveQosPolicy per storage pool or backend. Not supported by ontap-nas-economy.	“”
`snapshotReserve`	Percentage of volume reserved for snapshots “0”	If `snapshotPolicy` is `none`, `else` “”
`splitOnClone`	Split a clone from its parent upon creation	`false`
`encryption`	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. If NAE is enabled on the backend, any volume provisioned in Astra Trident will be NAE enabled. For more information, refer to: How Astra Trident works with NVE and NAE.	`false`
`luksEncryption`	Enable LUKS encryption. Refer to Use Linux Unified Key Setup (LUKS). SAN only.	""
`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 `mixed` and `unix` security styles. SMB supports `mixed` and `ntfs` security styles.	NFS default is `unix`. SMB default is `ntfs`.

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. Astra Trident supports SMB volumes mounted to pods running on Windows nodes only.
- At least one Astra 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`	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 Astra Trident to create the SMB share. This parameter is required for Amazon FSx for ONTAP backends.	`smb-share`
`nasType`	**Must set to `smb`.** If null, defaults to `nfs`.	`smb`
`securityStyle`	Security style for new volumes. **Must be set to `ntfs` or `mixed` for SMB volumes.**	`ntfs` or `mixed` for SMB volumes
`unixPermissions`	Mode for new volumes. **Must be left empty for SMB volumes.**	""

Configure a storage class and PVC

Configure a Kubernetes StorageClass object and create the storage class to instruct Astra 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 Astra Trident as the provisioner that is used for that class instructs Astra 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 Astra 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 Astra Trident, and Astra 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 RWO 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:
  - ReadWriteOnce
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 Astra Trident provisions volumes.

Astra Trident attributes

These parameters determine which Astra 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 task-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
```

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

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

Configure the Astra Trident EKS add-on on an EKS cluster

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

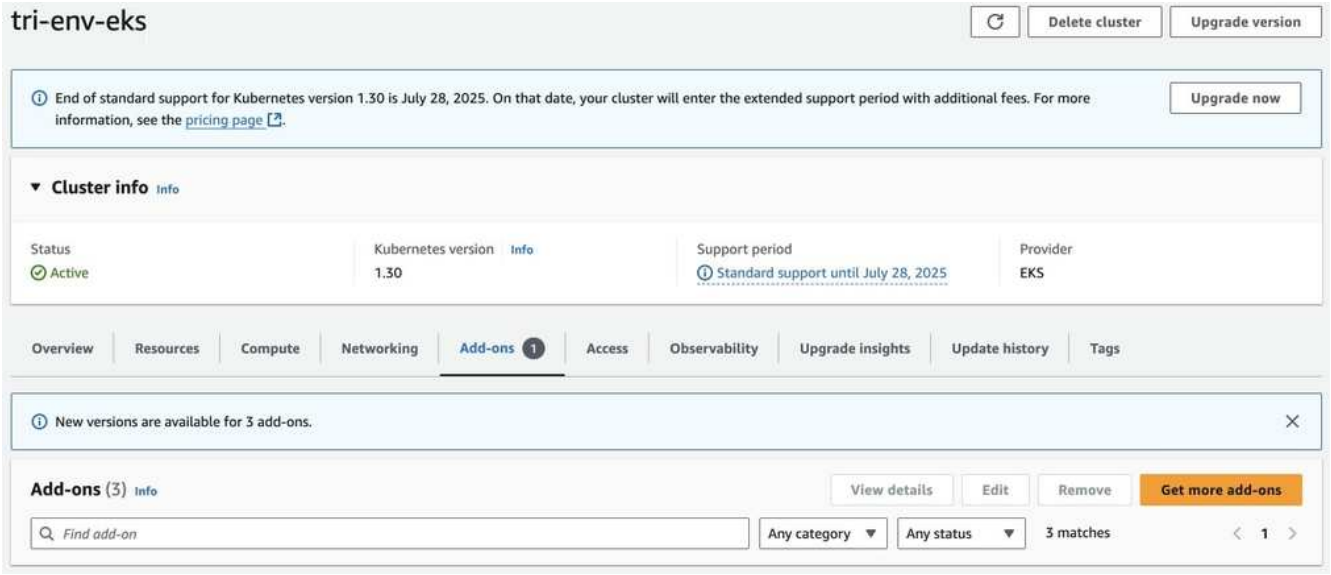
Prerequisites

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

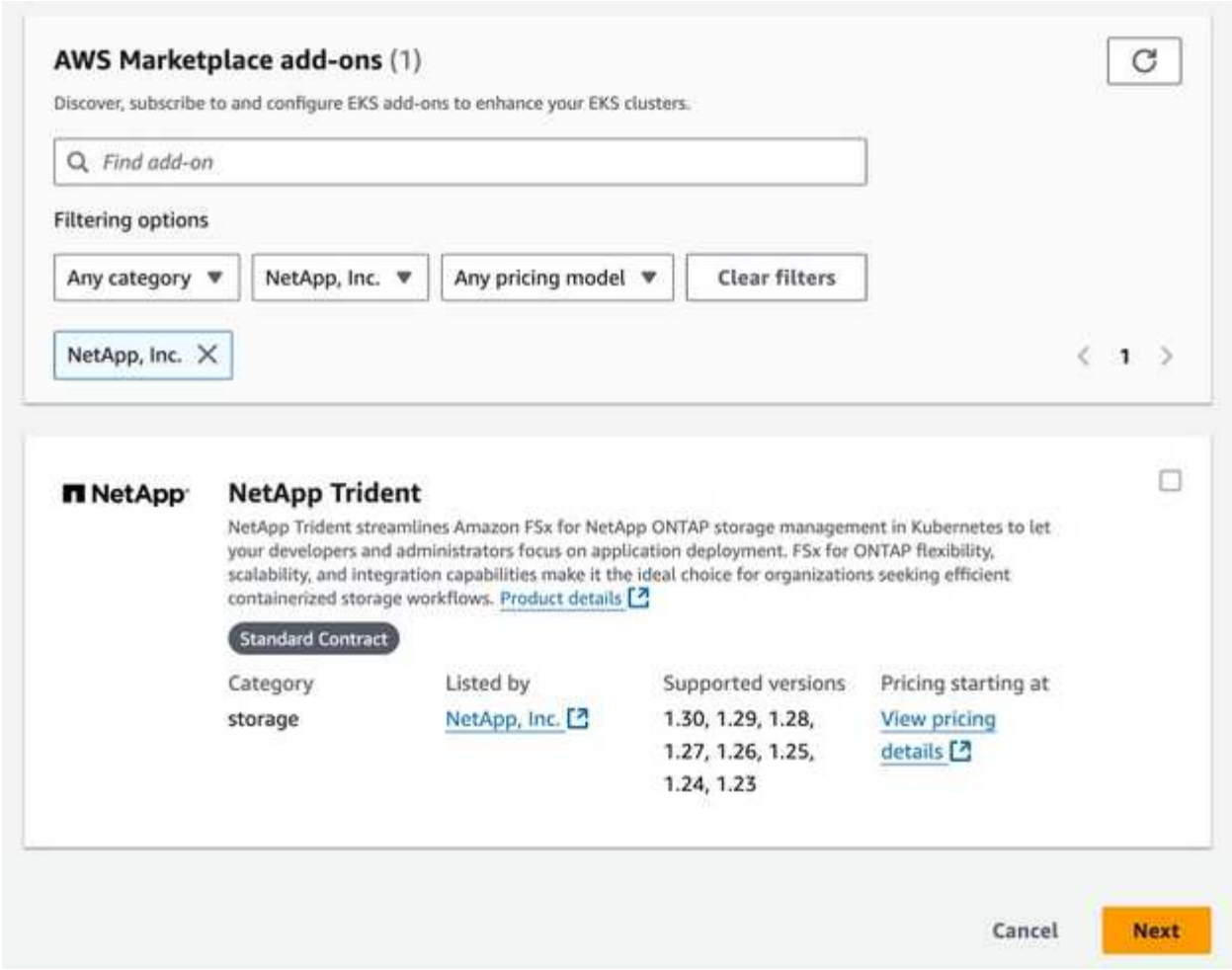
- An Amazon EKS cluster account with add-on subscription
- AWS permissions to the AWS marketplace:
  `"aws-marketplace:ViewSubscriptions",`
  `"aws-marketplace:Subscribe",`
  `"aws-marketplace:Unsubscribe`
- AMI type: Amazon Linux 2 (AL2_x86_64) or Amazon Linux 2 Arm(AL2_ARM_64)
- Node type: AMD or ARM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

Steps

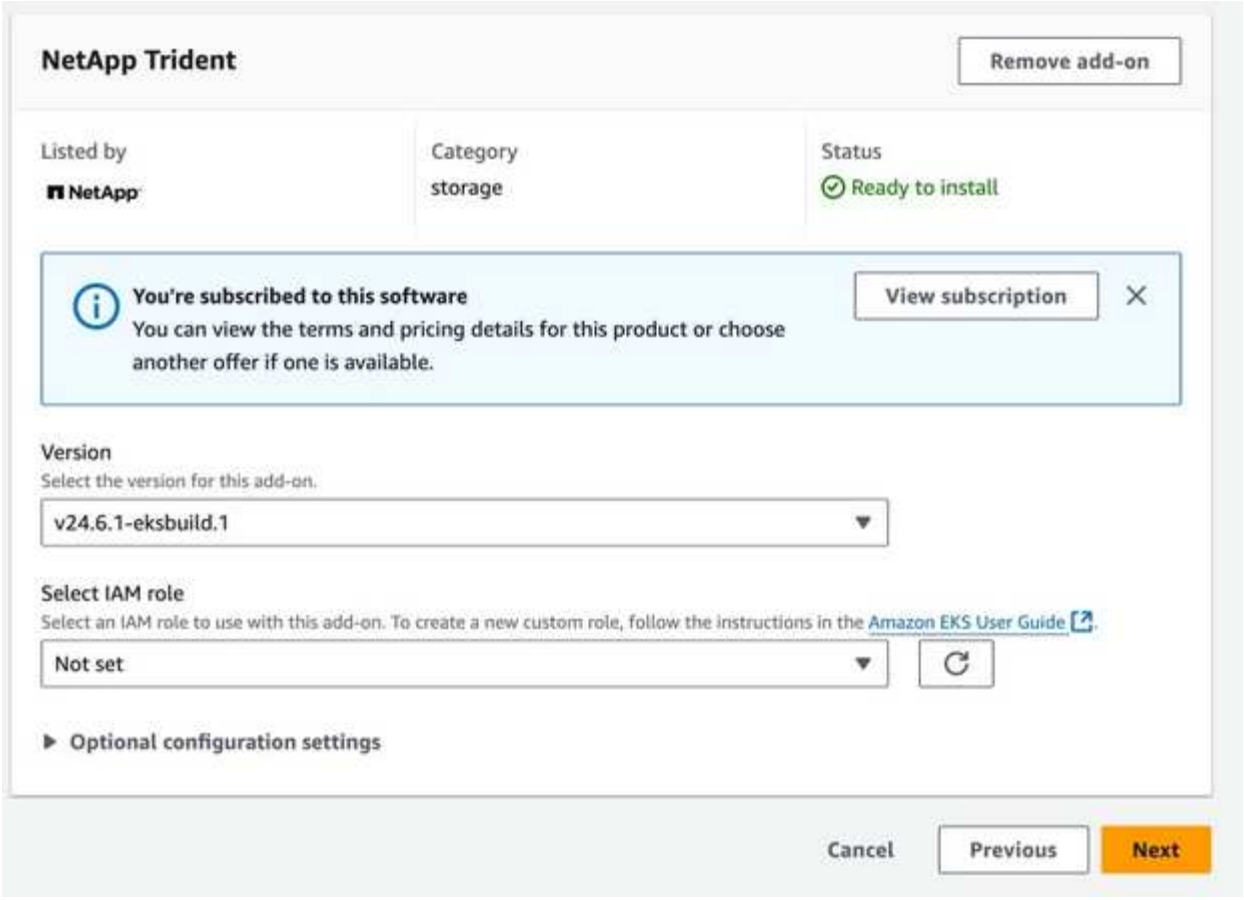
1. On your your EKS Kubernetes cluster, navigate to the **Add-ons** tab.



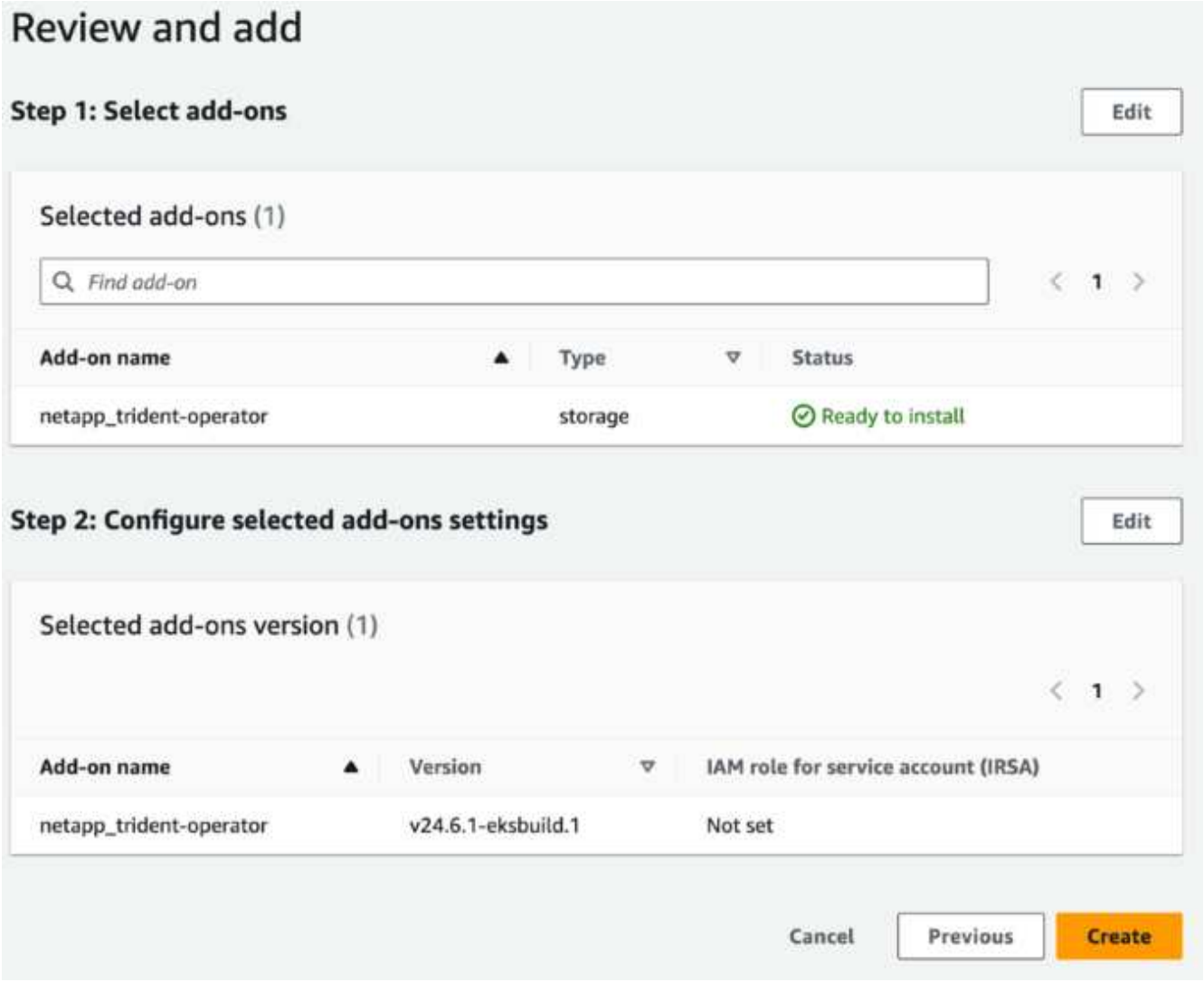
2. Go to **AWS Marketplace add-ons** and choose the *storage* category.



3. Locate **NetApp Trident** and select the checkbox for the Astra Trident add-on.
4. Choose the desired version of the add-on.



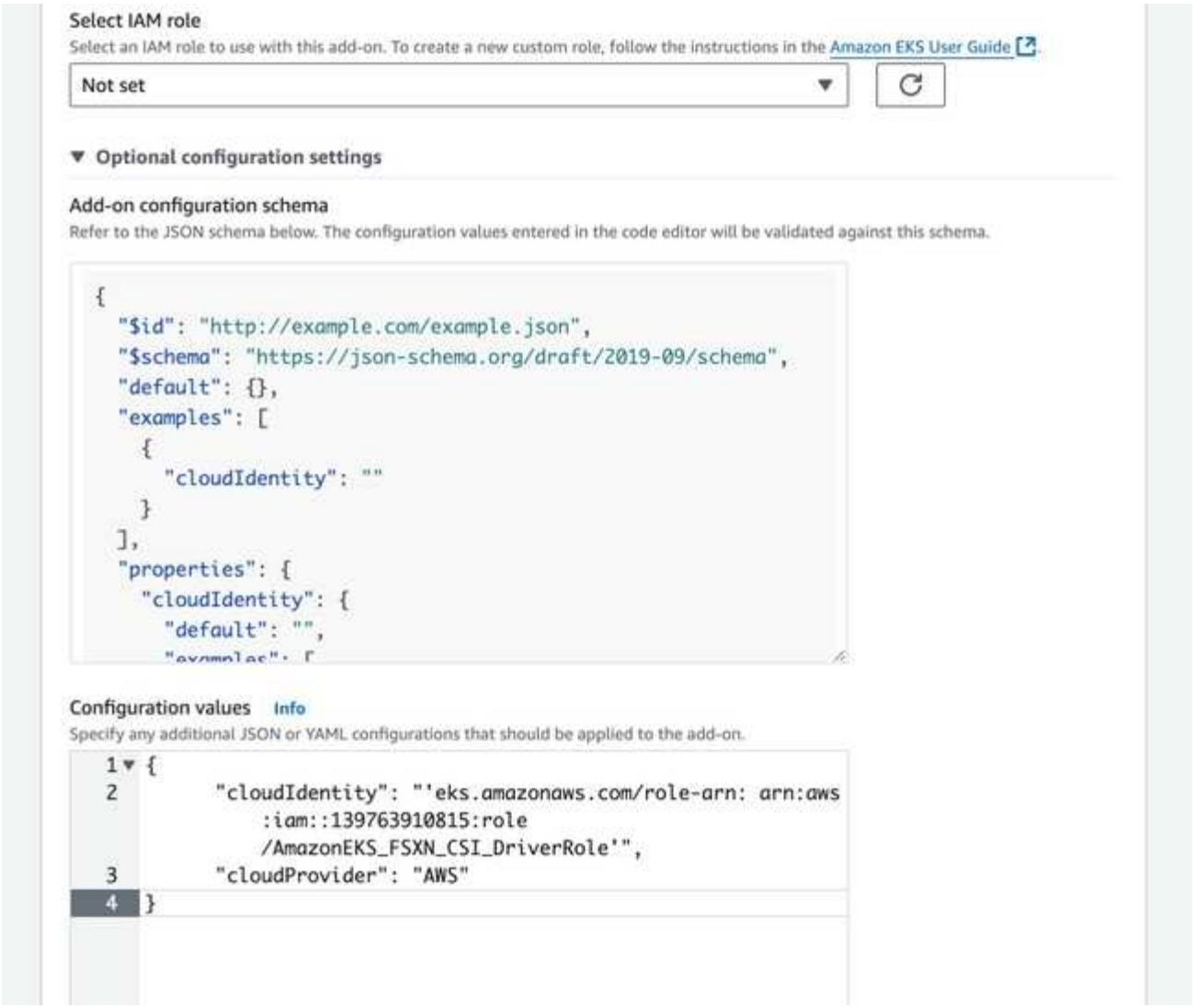
5. Select the IAM role option to inherit from the node.



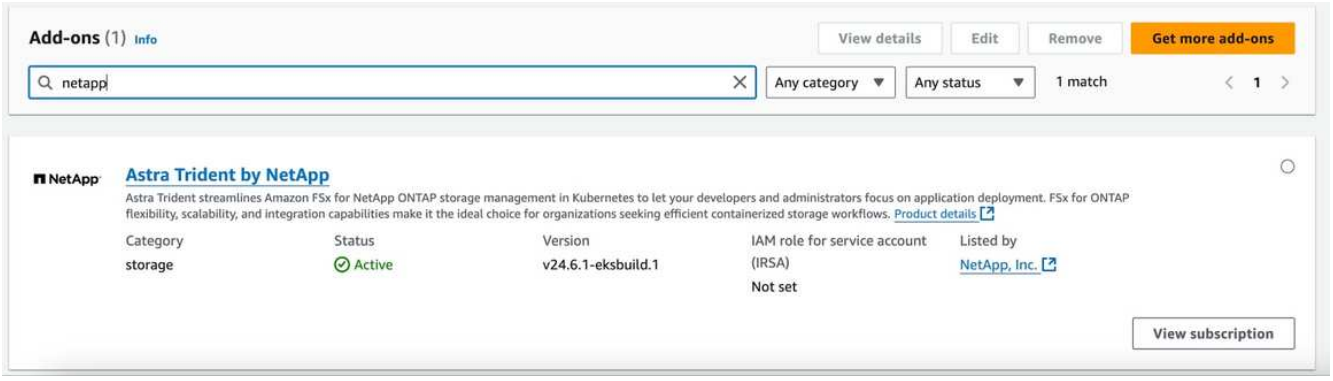
6. (Optional) Configure any Optional configuration settings as required and select **Next**.

   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.

   When you configure the optional parameter `cloudIdentity`, ensure that you specify `AWS` as the `cloudProvider` while installing Trident using the EKS add-on.



7. Select **Create**.
8. Verify that the status of the add-on is *Active*.



Install/uninstall the Astra Trident EKS add-on using CLI

Install the Astra Trident EKS add-on using CLI:

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

```
eksctl create addon --cluster K8s-arm --name netapp_trident-operator --version
```

`v24.6.1-eksbuild`
`eksctl create addon --cluster clusterName --name netapp_trident-operator --version v24.6.1-eksbuild.1` (with a dedicated version)

> When you configure the optional parameter `cloudIdentity`, ensure that you specify `cloudProvider` while installing Trident using the EKS add-on.

Uninstall the Astra Trident EKS add-on using CLI:

The following command uninstalls the Astra Trident EKS add-on:
`eksctl delete addon --cluster K8s-arm --name netapp_trident-operator`

Copyright information

Trademark information