



Amazon FSx for NetApp ONTAP

Trident

NetApp
October 29, 2026

目次

Amazon FSx for NetApp ONTAP	1
Amazon FSx for NetApp ONTAP で Trident を使用	1
要件	1
考慮事項	1
認証	2
テスト済みの Amazon マシンイメージ (AMI)	2
詳細情報の参照	3
IAMロールとAWSシークレットを作成する	4
AWS Secrets Manager シークレットを作成する	4
IAM ポリシーを作成する	4
Tridentをインストール	8
helm 経由で Trident をインストール	8
EKS アドオン経由で Trident をインストール	10
ストレージクラスを設定する	15
StorageClassオブジェクトを作成します	15
SMB ボリュームのプロビジョニング	23
バックエンドの高度な設定と例	23
ボリュームのプロビジョニング用のバックエンド設定オプション	27
PVCの設定	29
PVC を作成する	29
アプリケーションをデプロイする	30
サンプルアプリケーションを導入する	31
EKS クラスター上の Trident EKS アドオンを設定する	32
前提条件	32
手順	32
CLI を使用した Trident EKS アドオンのインストール / アンインストール	35

Amazon FSx for NetApp ONTAP

Amazon FSx for NetApp ONTAP で Trident を使用

"Amazon FSx for NetApp ONTAP"は、NetApp ONTAPストレージオペレーティングシステムを基盤とするファイルシステムを実行するフルマネージドAWSサービスです。AWSの拡張性と運用上の簡便性を備えたONTAPの機能、パフォーマンス、管理性を提供します。ファイルシステムはAmazon FSxの主要なリソースであり、オンプレミスのONTAPクラスタに相当します。各ファイルシステムには1つ以上のストレージ仮想マシン（SVM）が含まれており、各SVMにはファイルやディレクトリを格納する1つ以上のボリュームが含まれています。この統合により、Amazon Elastic Kubernetes Service（EKS）で実行されているKubernetesクラスターは、ブロックワークロードおよびファイルワークロード向けにONTAPをバックエンドとする永続ボリュームをプロビジョニングできるようになります。

要件

"Tridentの要件"に加えて、FSx for ONTAPをTridentと統合するには、次のものがが必要です：

- `kubectl`がインストールされている既存のAmazon EKSクラスタまたはセルフマネージドKubernetesクラスタ。
- クラスターのワーカーノードからアクセス可能な既存のAmazon FSx for NetApp ONTAPファイルシステムとStorage Virtual Machine（SVM）。
- "NFS または iSCSI"用に準備されたワーカーノード。



EKS AMI タイプに応じて、Amazon Linux および Ubuntu "Amazon Machine Images"（AMI）に必要なノード準備手順に従ってください。

考慮事項

- SMB ボリューム：
 - SMBボリュームは、`ontap-nas`ドライバーのみを使用してサポートされます。
 - SMB ボリュームは Trident EKS アドオンではサポートされていません。
 - Trident は、Windows ノード上で実行されているポッドにマウントされた SMB ボリュームのみをサポートします。詳細については、"[SMB ボリュームのプロビジョニングの準備](#)"を参照してください。
- Trident 24.02より前では、自動バックアップが有効になっているAmazon FSxファイルシステムで作成されたボリュームは、Tridentで削除できませんでした。Trident 24.02以降でこの問題を防ぐには、Amazon FSx for ONTAPのバックエンド設定ファイルで `fsxFilesystemID`、AWS `apiRegion`、AWS `apiKey`、およびAWS `secretKey`を指定します。



IAM ロールを Trident に指定する場合は、`apiRegion`、`apiKey`、および `secretKey` フィールドを Trident に明示的に指定することを省略できます。詳細については、"[FSx for ONTAP 設定オプションと例](#)"を参照してください。

Trident SAN/iSCSI と EBS-CSI ドライバの同時使用

AWS (EKS、ROSA、EC2、またはその他のインスタンス) でontap-sanドライバー (iSCSIなど) を使用する予定の場合、ノードに必要なマルチパス構成がAmazon Elastic Block Store (EBS) CSIドライバーと競合する可能性があります。同じノード上のEBSディスクに干渉することなくマルチパスが機能するようにするには、マルチパス設定でEBSを除外する必要があります。この例は、EBSディスクをマルチパスから除外しながら、必要なTrident設定を含む`multipath.conf`ファイルを示しています：

```
defaults {
    find_multipaths no
}
blacklist {
    device {
        vendor "NVME"
        product "Amazon Elastic Block Store"
    }
}
```

認証

Tridentには2つの認証モードがあります。

- 認証情報ベース (推奨)：認証情報を AWS Secrets Manager に安全に保存します。ファイルシステムの`fsxadmin`ユーザーまたは SVM 用に設定された`vsadmin`ユーザーを使用できます。



Trident は vsadmin SVM ユーザーとして、または同じロールを持つ別の名前のユーザーとして実行されることを想定しています。Amazon FSx for NetApp ONTAP には fsxadmin ユーザーがあり、これは ONTAP admin クラスター ユーザーの限定的な代替品です。vsadmin を Trident と併用することを強くお勧めします。

- 証明書ベース：Trident は、SVM にインストールされた証明書を使用して、FSx ファイル システム上の SVM と通信します。

認証を有効にする方法の詳細については、ドライバー タイプの認証を参照してください：

- ["ONTAP NAS 認証"](#)
- ["ONTAP SAN 認証"](#)

テスト済みの Amazon マシンイメージ (AMI)

EKS クラスターはさまざまなオペレーティングシステムをサポートしていますが、AWS は特定の Amazon マシンイメージ (AMI) をコンテナと EKS 用に最適化しています。以下の AMI は NetApp Trident 25.02 でテスト済みです。

AMI	NAS	NASEコノミー	iSCSI	iSCSIEコノミー
-----	-----	----------	-------	------------

AL2023_x86_64_STANDARDを使用したチャンクアップロード署名要求がサポートされるようになりました。	はい	はい	はい	はい
AL2_x86_64	はい	はい	○*	○*
BOTTLEROCKET_x86_64を使用したチャンクアップロード署名要求がサポートされるようになりました。	はい**	はい	該当なし	該当なし
AL2023_ARM_64_STANDARDを使用したチャンクアップロード署名要求がサポートされるようになりました。	はい	はい	はい	はい
AL2_ARM_64	はい	はい	○*	○*
BOTTLEROCKET_ARM_64	はい**	はい	該当なし	該当なし

- * ノードを再起動せずに PV を削除することはできません
- ** Trident バージョン 25.02 の NFSv3 では動作しません。



必要な AMI がここにリストされていない場合、それはサポートされていないという意味ではなく、単にテストされていないという意味です。このリストは、動作することがわかっている AMI のガイドとして機能します。

テストに使用したデバイス：

- EKS version: 1.32
- インストール方法：Helm 25.06 および AWS アドオン 25.06
- NAS については、NFSv3 と NFSv4.1 の両方がテストされました。
- SAN の場合、NVMe-oF ではなく iSCSI のみがテストされました。

実行されたテスト：

- 作成：Storage Class、pvc、pod
- 削除：ポッド、PVC（レギュラー、qtree/lun – エコノミー、AWS バックアップ付き NAS）

詳細情報の参照

- ["Amazon FSx for NetApp ONTAP ドキュメント"](#)
- ["Amazon FSx for NetApp ONTAP に関するブログ投稿"](#)

IAMロールとAWSシークレットを作成する

明示的な AWS 認証情報を提供する代わりに、AWS IAM ロールとして認証することで、Kubernetes ポッドが AWS リソースにアクセスするように設定できます。



AWS IAM ロールを使用して認証するには、EKS を使用して Kubernetes クラスタを導入する必要があります。

AWS Secrets Manager シークレットを作成する

Trident はストレージを管理するために FSx vserver に対して API を発行するため、そのためには資格情報が必要になります。これらの認証情報を渡す安全な方法は、AWS Secrets Manager シークレットを使用することです。したがって、まだお持ちでない場合は、vsadmin アカウントの認証情報を含む AWS Secrets Manager シークレットを作成する必要があります。

この例では、Trident CSIの資格情報を保存するためのAWS Secrets Managerシークレットを作成します：

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

IAM ポリシーを作成する

Tridentを正しく実行するには、AWS権限も必要です。したがって、Tridentに必要な権限を付与するポリシーを作成する必要があります。

次の例では、AWS CLI を使用して IAM ポリシーを作成します：

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

ポリシー **JSON** の例：

```
{
  "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"
}

```

Pod Identity またはサービス アカウントの関連付け（IRSA）用の IAM ロールを作成する

EKS Pod Identity を持つ AWS Identity and Access Management（IAM）ロール、またはサービスアカウントの関連付け（IRSA）の IAM ロールを引き受けるように Kubernetes サービスアカウントを設定できます。サービスアカウントを使用するように設定されたすべてのポッドは、ロールがアクセス権限を持つすべての AWS サービスにアクセスできるようになります。

Pod アイデンティティ

Amazon EKS Pod Identity associationsは、Amazon EC2インスタンスプロファイルがAmazon EC2インスタンスにクレデンシャルを提供するのと同様に、アプリケーションのクレデンシャルを管理する機能を提供します。

EKS クラスタに Pod Identity をインストールします：

AWS コンソールまたは次の AWS CLI コマンドを使用して、Pod identity を作成できます。

```
aws eks create-addon --cluster-name <EKS_CLUSTER_NAME> --addon-name
eks-pod-identity-agent
```

詳細については、"[Amazon EKS Pod Identity Agent を設定する](#)"を参照してください。

trust-relationship.json を作成：

EKS サービス プリンシパルが Pod Identity に対してこのロールを引き受けることができるように、trust-relationship.json を作成します。次に、この信頼ポリシーを持つロールを作成します：

```
aws iam create-role \  
  --role-name fsxn-csi-role --assume-role-policy-document file://trust-  
relationship.json \  
  --description "fsxn csi pod identity role"
```

trust-relationship.json ファイル：

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "pods.eks.amazonaws.com"  
      },  
      "Action": [  
        "sts:AssumeRole",  
        "sts:TagSession"  
      ]  
    }  
  ]  
}
```

IAM ロールにロールポリシーをアタッチします：

前の手順のロールポリシーを、作成した IAM ロールにアタッチします：

```
aws iam attach-role-policy \  
  --policy-arn arn:aws:iam::aws:111122223333:policy/fsxn-csi-policy \  
  --role-name fsxn-csi-role
```

ポッド **ID** の関連付けを作成します：

IAMロールとTridentサービスアカウント（trident-controller）の間にポッドIDの関連付けを作成する

```
aws eks create-pod-identity-association \  
  --cluster-name <EKS_CLUSTER_NAME> \  
  --role-arn arn:aws:iam::111122223333:role/fsxn-csi-role \  
  --namespace trident --service-account trident-controller
```

サービス アカウントの関連付け（**IRSA**）の **IAM** ロール

AWS CLI の使用：

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

trust-relationship.json ファイル：

```
{  
  "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"  
        }  
      }  
    }  
  ]  
}
```

`trust-relationship.json` ファイルで次の値を更新します：

- **<account_id>** - AWSアカウントID
- **<oidc_provider>** - EKS クラスタの OIDC。oidc_provider は次のコマンドを実行して取得できます：

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

IAM ロールを **IAM** ポリシーにアタッチします：

ロールが作成されたら、次のコマンドを使用して、（上記の手順で作成された）ポリシーをロールにアタッチします：

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

ARN>

OICD プロバイダーが関連付けられていることを確認します：

OIDC プロバイダーがクラスターに関連付けられていることを確認します。次のコマンドを使用して確認できます：

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

出力が空の場合は、次のコマンドを使用して IAM OIDC をクラスターに関連付けます：

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

eksctl を使用している場合、次の例を使用して EKS のサービス アカウントの IAM ロールを作成します：

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

Tridentをインストール

Trident は、Kubernetes における Amazon FSx for NetApp ONTAP のストレージ管理を合理化し、開発者と管理者がアプリケーションの展開に集中できるようにします。Tridentは次のいずれかの方法を使用してインストールできます：

- Helm
- EKS アドオン

スナップショット機能を利用する場合は、CSI スナップショット コントローラー アドオンをインストールします。詳細については、"[CSI ボリュームのスナップショット機能を有効にする](#)"を参照してください。

helm 経由で Trident をインストール

Pod アイデンティティ

1. Trident Helm リポジトリを追加します：

```
helm repo add netapp-trident https://netapp.github.io/trident-helm-
```

```
chart
```

2. 次の例を使用して Trident をインストールします：

```
helm install trident-operator netapp-trident/trident-operator
--version 100.2502.1 --namespace trident --create-namespace
```

``helm list`` コマンドを使用して、名前、ネームスペース、チャート、ステータス、アプリケーションバージョン、リビジョン番号などのインストールの詳細を確認できます。

```
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-
100.2502.0	25.02.0		

サービス アカウント アソシエーション (IRSA)

1. Trident Helm リポジトリを追加します：

```
helm repo add netapp-trident https://netapp.github.io/trident-helm-chart
```

2. **cloud provider** と **cloud identity** の値を設定します。

```
helm install trident-operator netapp-trident/trident-operator
--version 100.2502.1 \
--set cloudProvider="AWS" \
--set cloudIdentity="'eks.amazonaws.com/role-arn:
arn:aws:iam::<accountID>:role/<AmazonEKS_FSxN_CSI_DriverRole>' " \
--namespace trident \
--create-namespace
```

``helm list`` コマンドを使用して、名前、ネームスペース、チャート、ステータス、アプリケーション

ョンバージョン、リビジョン番号などのインストールの詳細を確認できます。

```
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-
100.2510.0	25.10.0		

iSCSI を使用する予定の場合は、クライアント マシンで iSCSI が有効になっていることを確認してください。AL2023 Worker node OS を使用している場合は、helm インストールで node prep パラメータを追加することで、iSCSI クライアントのインストールを自動化できます：



```
helm install trident-operator netapp-trident/trident-operator  
--version 100.2502.1 --namespace trident --create-namespace --  
set nodePrep={iscsi}
```

EKS アドオン経由で Trident をインストール

Trident EKS アドオンには最新のセキュリティパッチとバグ修正が含まれており、Amazon EKS で動作することが AWS によって検証されています。EKS アドオンを使用すると、Amazon EKS クラスターの安全性と安定性を常に確保し、アドオンのインストール、設定、更新に必要な作業量を削減できます。

前提条件

AWS EKS の Trident アドオンを設定する前に、以下のものを用意してください：

- アドオン サブスクリプション付きの Amazon EKS クラスタ アカウント
- AWS マーケットプレイスへの AWS 権限：
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",
"aws-marketplace:Unsubscribe"
- AMI タイプ：Amazon Linux 2 (AL2_x86_64) または Amazon Linux 2 Arm (AL2_ARM_64)
- ノードタイプ：AMDまたはARM
- 既存の Amazon FSx for NetApp ONTAP ファイルシステム

AWS の Trident アドオンを有効にする

管理コンソール

1. Amazon EKS コンソールを開きます <https://console.aws.amazon.com/eks/home#/clusters>。
2. 左側のナビゲーション ペインで、* Clusters * を選択します。
3. NetApp Trident CSI アドオンを設定するクラスタの名前を選択します。
4. *アドオン*を選択し、*さらにアドオンを取得*を選択します。
5. アドオンを選択するには、次の手順に従います。
 - a. **AWS Marketplace** アドオン セクションまでスクロールし、検索ボックスに "**Trident**" と入力します。
 - b. Trident by NetApp ボックスの右上隅にあるチェックボックスをオンにします。
 - c. 「次へ」を選択します。
6. *選択したアドオンの構成*設定ページで、次の操作を行います：



Pod Identity 関連付けを使用している場合は、これらの手順をスキップしてください。

- a. 使用する*バージョン*を選択します。
- b. IRSA 認証を使用している場合は、オプション構成設定で使用可能な構成値を必ず設定してください：
 - 使用する*バージョン*を選択します。
 - *アドオン構成スキーマ*に従って、*構成値*セクションの*configurationValues*パラメータを、前の手順で作成した role-arn に設定します（値は次の形式にする必要があります）：

```
{  
  
  "cloudIdentity": "'eks.amazonaws.com/role-arn: <role ARN>'",  
  "cloudProvider": "AWS"  
  
}
```

+

競合解決方法として [オーバーライド] を選択した場合、既存のアドオンの 1 つ以上の設定を Amazon EKS アドオン設定で上書きできます。このオプションを有効にせず、既存の設定と競合する場合、操作は失敗します。結果のエラーメッセージを使用して競合のトラブルシューティングを行うことができます。このオプションを選択する前に、Amazon EKS アドオンが自己管理する必要がある設定を管理していないことを確認してください。

7. **Next** を選択します。
8. *確認と追加*ページで、*作成*を選択します。

アドオンのインストールが完了すると、インストールされたアドオンが表示されます。

AWS CLI

1. `add-on.json` ファイルを作成する：

Pod Identity の場合は、次の形式を使用します：



ビジネス

```
{
  "clusterName": "<eks-cluster>",
  "addonName": "netapp_trident-operator",
  "addonVersion": "v25.6.0-eksbuild.1",
}
```

IRSA 認証の場合は、次の形式を使用します：

```
{
  "clusterName": "<eks-cluster>",
  "addonName": "netapp_trident-operator",
  "addonVersion": "v25.6.0-eksbuild.1",
  "serviceAccountRoleArn": "<role ARN>",
  "configurationValues": {
    "cloudIdentity": "'eks.amazonaws.com/role-arn: <role ARN>'",
    "cloudProvider": "AWS"
  }
}
```



`<role ARN>`を、前の手順で作成されたロールの ARN に置き換えます。

2.Trident EKS アドオンをインストールします。

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

eksctl

次のコマンド例では、Trident EKS アドオンをインストールします：

```
eksctl create addon --name netapp_trident-operator --cluster
<cluster_name> --force
```

Trident EKS アドオンを更新する

管理コンソール

1. Amazon EKS コンソールを開きます <https://console.aws.amazon.com/eks/home#/clusters>。

2. 左側のナビゲーション ペインで、* Clusters * を選択します。
3. NetApp Trident CSI アドオンを更新するクラスターの名前を選択します。
4. アドオン タブを選択します。
5. **Trident by NetApp** を選択し、*編集*を選択します。
6. *NetApp による Trident の設定*ページで、次の操作を行います：
 - a. 使用する*バージョン*を選択します。
 - b. * オプションの構成設定 * を展開し、必要に応じて変更します。
 - c. 「変更を保存」を選択します。

AWS CLI

次の例では、EKS add-on を更新します。

```
aws eks update-addon --cluster-name <eks_cluster_name> --addon-name
netapp_trident-operator --addon-version v25.6.0-eksbuild.1 \
  --service-account-role-arn <role-ARN> --resolve-conflict preserve \
  --configuration-values "{\"cloudIdentity\":
  \"'eks.amazonaws.com/role-arn: <role ARN>'\"}"
```

eksctl

- FSxN Trident CSI アドオンの現在のバージョンを確認してください。`my-cluster`をクラスター名に置き換えます。

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

出力例：

NAME	VERSION	STATUS	ISSUES
IAMROLE	UPDATE AVAILABLE	CONFIGURATION VALUES	
netapp_trident-operator	v25.6.0-eksbuild.1	ACTIVE	0
{\"cloudIdentity\": \"'eks.amazonaws.com/role-arn: arn:aws:iam::139763910815:role/AmazonEKS_FSXN_CSI_DriverRole'\"}			

- 前の手順の出力の UPDATE AVAILABLE で返されたバージョンにアドオン ソフトウェアを更新します。

```
eksctl update addon --name netapp_trident-operator --version
v25.6.0-eksbuild.1 --cluster my-cluster --force
```

`--force` オプションを削除し、Amazon EKS

アドオン設定のいずれかが既存の設定と競合する場合、Amazon EKS アドオンの更新は失敗し、競合を解決するためのエラーメッセージが表示されます。このオプションを指定する前に、Amazon EKS アドオンが管理する必要のある設定を管理していないことを確認してください。このオプションによってそれらの設定が上書きされるためです。この設定の他のオプションの詳細については、[link:https://eksctl.io/usage/addons/](https://eksctl.io/usage/addons/)["アドオン"]を参照してください。
。Amazon EKS Kubernetes
フィールド管理の詳細については、[link:https://docs.aws.amazon.com/eks/latest/userguide/kubernetes-field-management.html](https://docs.aws.amazon.com/eks/latest/userguide/kubernetes-field-management.html)["Kubernetes
フィールド管理"]を参照してください。

Trident EKS アドオンをアンインストール/削除します

Amazon EKS アドオンを削除するには、次の 2 つのオプションがあります：

- ・ クラスター上のアドオン ソフトウェアを保持する – このオプションを選択すると、Amazon EKS によるすべての設定の管理が削除されます。また、Amazon EKS が更新を通知し、更新を開始した後に Amazon EKS アドオンを自動的に更新する機能も削除されます。ただし、クラスター上のアドオン ソフトウェアは保持されます。このオプションを選択すると、アドオンは Amazon EKS アドオンではなく、自己管理型インストールになります。このオプションを使用すると、アドオンのダウンタイムは発生しません。アドオンを保持するには、コマンドで `--preserve` オプションを保持します。
- ・ アドオン ソフトウェアをクラスターから完全に削除する – NetApp は、クラスター上にそのアドオンに依存するリソースが存在しない場合のみ、Amazon EKS アドオンをクラスターから削除することを推奨しています。アドオンを削除するには、`--preserve` オプションを `delete` コマンドから削除してください。



アドオンに IAM アカウントが関連付けられている場合、IAM アカウントは削除されません。

管理コンソール

1. Amazon EKS コンソールを開きます <https://console.aws.amazon.com/eks/home#/clusters>。
2. 左側のナビゲーション ペインで、* Clusters * を選択します。
3. NetApp Trident CSI アドオンを削除するクラスターの名前を選択します。
4. アドオン*タブを選択し、*NetAppによるTrident*を選択します。
5. *削除*を選択します。
6. *netapp_trident-operator の削除確認*ダイアログで、次の操作を行います：
 - a. Amazon EKS によるアドオン設定の管理を停止する場合は、**Preserve on cluster** を選択します。クラスター上のアドオン ソフトウェアを保持し、アドオンのすべての設定を自分で管理する場合は、これを行ってください。
 - b. **netapp_trident-operator** と入力します。
 - c. *削除*を選択します。

AWS CLI

`my-cluster` をクラスタの名前に置き換えてから、次のコマンドを実行します。

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

eksctl

次のコマンドは、Trident EKS アドオンをアンインストールします：

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

ストレージクラスを設定する

<https://kubernetes.io/docs/concepts/storage/storage-classes/> ["Kubernetes StorageClass オブジェクト

"^"] はプロビジョナーを識別し、プロビジョナーにボリュームのプロビジョニング方法を指示します。このセクションでは、Trident をプロビジョナーとして指定する Kubernetes StorageClass オブジェクトの構成方法を説明します。

StorageClass オブジェクトを作成します

FSx for ONTAP 用の StorageClass を作成すると、Trident はバックエンド構成を自動的に作成します。



ストレージバックエンドを手動で構成する場合は、[自動バックエンド構成なしでKubernetes StorageClassを作成する](#) セクションを参照して、Trident バックエンドとストレージクラスを個別に作成してください。

必須の **StorageClass** パラメータを指定する

StorageClass の作成時に定義する必要がある次の3つのパラメータ：

パラメータ	必須	タイプ	概要
fsxFilesystemID	はい	string	FSx for NetApp ONTAP ファイルシステム ID
storageDriverName	はい	string	Trident ストレージドライバ（例：ontap-nas または `ontap-san`）
credentialsName	はい	string	ONTAP クレデンシャルを含む FSx for ONTAP の Kubernetes Secret の名前

オプションのパラメータを指定します

StorageClassを通じてオプションのバックエンドパラメータを渡すことができます。StorageClass `parameters` セクションで、すべてのオプション値を文字列として定義します。バックエンドパラメータの完全なリストについては、以下を参照してください：["FSx for NetApp ONTAP バックエンド構成"](#)。

StorageClass設定ファイルの例。

次の例は、バックエンドの自動設定をトリガーする StorageClass を示しています。

YAML

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-fsx-demo
  annotations:
    description: "Demo StorageClass for FSx for NetApp ONTAP"
provisioner: csi.trident.netapp.io
parameters:
  fsxFilesystemID: "fs-0abc123"
  storageDriverName: "ontap-nas"
  credentialsName: trident-fsx-credentials
allowVolumeExpansion: true
reclaimPolicy: Delete
volumeBindingMode: Immediate
```

JSON

```
{
  "apiVersion": "storage.k8s.io/v1",
  "kind": "StorageClass",
  "metadata": {
    "name": "ontap-fsx-demo",
    "annotations": {
      "description": "Demo StorageClass for FSx for NetApp ONTAP"
    }
  },
  "provisioner": "csi.trident.netapp.io",
  "parameters": {
    "fsxFilesystemID": "fs-0abc123",
    "storageDriverName": "ontap-nas",
    "credentialsName": "trident-fsx-credentials"
  },
  "allowVolumeExpansion": true,
  "reclaimPolicy": "Delete",
  "volumeBindingMode": "Immediate"
}
```

```
}
```

StorageClassを作成します

設定ファイルを作成したら、次のコマンドを実行してストレージクラスを作成します。

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

KubernetesとTridentの両方で*basic-csi*ストレージクラスが表示され、Tridentがバックエンドでプールを検出しているはずです。

```
kubectl get sc basic-csi
```

NAME	PROVISIONER	AGE
basic-csi	csi.trident.netapp.io	15h

StorageClassを適用すると、Tridentはバックエンドを自動的に作成します。その後、このStorageClassを参照するPersistentVolumeClaimsを作成できます。

バックエンド構成ステータスを確認する

Tridentは、バックエンド作成の結果をStorageClassアノテーションに記録します。

注釈	概要
trident.netapp.io/configuratorStatus	設定結果(Success`または`Failure)
trident.netapp.io/configuratorMessage	詳細なステータスまたはエラーメッセージ
trident.netapp.io/configuratorName	内部コンフィギュレーターリソースの名前
trident.netapp.io/managed	StorageClass が Trident によって管理されていることを示します
trident.netapp.io/additionalStoragePools	このバックエンド用に作成されたストレージプール

ステータスを確認するには、以下を実行してください：

```
kubectl get storageclass ontap-fsx-demo -o yaml
```

```
`trident.netapp.io/configuratorStatus`が  
`Success`に設定されていることを確認してください。値が  
`Failure`の場合は、エラーについて  
`trident.netapp.io/configuratorMessage`を確認してください。
```

追加のFSxNファイルシステムを追加する

同じ StorageClass を使い続けながら追加のストレージ容量が必要な場合は、追加の FSxN ファイルシステム ID を追加してください。

StorageClass を編集し、以下のアノテーションを追加します：

```
metadata:
  annotations:
    trident.netapp.io/additionalFsxnFileSystemID: '["fs-
xxxxxxxxxxxxxxxxxxxxx"]'
```

変更を適用すると、Trident はバックエンド構成を更新し、StorageClass アノテーションを更新します。

運用上の考慮事項と制限

- 自動バックエンド構成を持つStorageClassを削除すると、通常、関連するTridentバックエンドが削除されます。これにより、ストレージ接続が阻害され、実行中のワークロードが中断される可能性があります。管理対象の StorageClass を削除する前に、その影響を検証してください。
- 自動バックエンド構成は、AWS FSx for NetApp ONTAP でのみサポートされています。

自動バックエンド構成なしで**Kubernetes StorageClass**を作成する

TridentバックエンドとStorageClassを別々に作成する場合は、次の手順に従ってください。

自動バックエンド構成の仕組みを理解する

Trident は、StorageClass の定義からバックエンド構成を取得します。StorageClass を適用すると、Trident は必要なパラメータを検証し、バックエンドを作成して、StorageClass にステータスのアノテーションを付与します。

TridentはVolumeSnapshotClassを1回だけ作成します。Tridentは、後続のStorageClassesに対して同じVolumeSnapshotClassを再利用します。

Tridentバックエンドを作成する

Tridentバックエンドを作成するには、JSON形式またはYAML形式で設定ファイルを作成する必要があります。ファイルには、使用するストレージの種類（NASまたはSAN）、ファイルシステム、取得元のSVM、およびその認証方法を指定する必要があります。次の例は、NASベースのストレージを定義し、AWSシークレットを使用して使用するSVMのクレデンシャルを保存する方法を示しています：

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:xxxxxxx: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:xxxxxxx:secret:secret-
name",
      "type": "awsarn"
    }
  }
}
```

FSx for ONTAPドライバーの詳細

次のドライバを使用して、Trident を Amazon FSx for NetApp ONTAP と統合できます：

ドライバー名	概要
ontap-san	プロビジョニングされた各PVは、独自のAmazon FSx for NetApp ONTAPボリューム内のLUNです。ブロックストレージに推奨されます。

ドライバー名	概要
ontap-nas	プロビジョニングされた各PVは、完全なAmazon FSx for NetApp ONTAP ボリュームです。NFSおよびSMBに推奨されます。
ontap-san-economy	プロビジョニングされた各PVは、Amazon FSx for NetApp ONTAP ボリュームあたりの設定可能な数のLUNを持つLUNです。
ontap-nas-economy	プロビジョニングされた各PVはqtreeであり、Amazon FSx for NetApp ONTAPボリュームあたりのqtree数は設定可能です。
ontap-nas-flexgroup	プロビジョニングされた各PVは、完全なAmazon FSx for NetApp ONTAP FlexGroupボリュームです。

ドライバーの詳細については、"[NASドライバー](#)"および"[SANドライバー](#)"を参照してください。

バックエンドを作成する

設定ファイルを作成した後、以下のコマンドを実行して、Tridentバックエンド構成（TBC）を作成および検証します：

- yaml ファイルから Trident バックエンド構成（TBC）を作成し、次のコマンドを実行します：

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

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

- Trident バックエンド構成（TBC）が正常に作成されたことを確認します：

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

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

その他の構成オプションの詳細については、以下の[バックエンドの高度な設定と例](#)セクションを参照してください。

自動バックエンド構成*なし*でストレージクラスを構成する

以下は、TridentおよびFSx for ONTAPで使用するストレージクラス構成の例です。

NFS用ストレージクラス

この例を使用して、NFS を使用するボリュームの StorageClass を設定できます（属性の全リストについては、以下の Trident 属性セクションを参照してください）：

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

iSCSI用ストレージクラス

この例を使用して、iSCSIを使用するボリュームのStorageClassを設定します：

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

NFSv3とAWS Bottlerocketを使用したストレージクラス

AWS BottlerocketでNFSv3ボリュームをプロビジョニングするには、必要な `mountOptions` をストレージクラスに追加します：

```
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"
mountOptions:
```

- nfsvers=3
- nolock

Trident StorageClass属性

これらのパラメータは、特定のタイプのボリュームをプロビジョニングするためにどのTrident管理ストレージプールを使用するかを決定します。

属性	タイプ	値	オファー	要求	サポート対象
メディア ¹	string	HDD、ハイブリッド、SSD	プールにはこのタイプのメディアが含まれます。ハイブリッドとは両方を意味します	指定されたメディアタイプ	ontap-nas、ontap-nas-economy、ontap-nas-flexgroup、ontap-san、solidfire-san
provisioningType	string	薄い、厚い	プールはこのプロビジョニング方法をサポートしています	プロビジョニング方法が指定されました	thick：すべてのONTAP、thin：すべてのONTAPおよびsolidfire-san
backendType	string	ontap-nas、ontap-nas-economy、ontap-nas-flexgroup、ontap-san、solidfire-san、azure-netapp-files、ontap-san-economy	プールはこのタイプのバックエンドに属します	バックエンドが指定されました	すべてのドライバー
Snapshot	ブール値	true、false	プールはSnapshot 付きのボリュームをサポートします	スナップショットが有効になっているボリューム	ontap-nas、ontap-san、solidfire-san
クローン	ブール値	true、false	プールはボリュームのクローン作成をサポート	クローンが有効なボリューム	ontap-nas、ontap-san、solidfire-san
暗号化	ブール値	true、false	プールは暗号化されたボリュームをサポートします	暗号化が有効になっているボリューム	ontap-nas、ontap-nas-economy、ontap-nas-flexgroups、ontap-san
IOPS	int	正の整数	プールはこの範囲の IOPS を保証できる	ボリュームで保証されるIOPS	solidfire-san

¹：ONTAP SelectまたはFSx for ONTAPシステムではサポートされていません

"[KubernetesとTridentオブジェクト](#)"を参照して、ストレージクラスが`PersistentVolumeClaim`とどのように相互作用するか、および Trident がボリュームをプロビジョニングする方法を制御するパラメータの詳細を確認してください。

ストレージクラスを作成します

StorageClassを設定したら、Kubernetesで作成できます。

手順

1. これはKubernetesオブジェクトなので、`kubectl`を使用してKubernetesで作成します。

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

2. KubernetesとTridentの両方で`basic-csi`ストレージクラスが表示され、Tridentがバックエンドでプールを検出しているはずです。

```
kubectl get sc basic-csi
```

NAME	PROVISIONER	AGE
basic-csi	csi.trident.netapp.io	15h

SMB ボリュームのプロビジョニング

SMB ボリュームは、`ontap-nas`ドライバを使用してプロビジョニングできます。ただし、そのためには以下の手順を完了する必要があります：["SMB ボリュームのプロビジョニングの準備"](#)。

バックエンドの高度な設定と例

バックエンド構成オプションについては、次の表を参照してください：

パラメータ	概要	例
version		常に1
storageDriverName	ストレージドライバの名前	ontap-nas、ontap-nas-economy、ontap-nas-flexgroup、ontap-san、ontap-san-economy
backendName	カスタム名またはストレージバックエンド	ドライバ名 + "_" + dataLIF

パラメータ	概要	例
managementLIF	<p>クラスタまたはSVM管理LIFのIPアドレス（完全修飾ドメイン名（FQDN）も指定可能）TridentがIPv6 フラグを使用してインストールされている場合、IPv6 アドレスを使用するように設定できます。IPv6 アドレスは角括弧で囲んで定義する必要があります（例：[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]）。もし`fsxFilesystemID`を`aws`フィールドの下で指定した場合、`managementLIF`を指定する必要はありません。なぜならTridentがAWSからSVM `managementLIF`情報を取得するためです。したがって、SVM配下のユーザー（例：vsadmin）の認証情報を必ず指定し、そのユーザーが`vsadmin`ロールを持っている必要があります。</p>	"10.0.0.1"、"[2001:1234:abcd::fefe]"
dataLIF	<p>プロトコル LIF の IP アドレス。ONTAP NAS ドライバー：NetApp は dataLIF を指定することを推奨します。指定しない場合、Trident は SVM から dataLIF を取得します。NFS マウント操作に使用する完全修飾ドメイン名（FQDN）を指定することで、ラウンドロビン DNS を作成し、複数の dataLIF 間で負荷分散を行うことができます。初期設定後でも変更可能です。ONTAP SAN ドライバー：iSCSI には指定しないでください。Trident は ONTAP Selective LUN Map を使用して、マルチパスセッションの確立に必要な iSCSI LIF を検出します。dataLIF が明示的に定義されている場合、警告が生成されます。Trident が IPv6 フラグを使用してインストールされている場合、IPv6 アドレスを使用するように設定できます。IPv6 アドレスは角括弧で囲んで定義する必要があります（例：[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]）。</p>	

パラメータ	概要	例
autoExportPolicy	自動エクスポート ポリシーの作成と更新を有効にします [ブール値]。 `autoExportPolicy` および `autoExportCIDRs` オプションを使用すると、Tridentはエクスポートポリシーを自動的に管理できます。	false
autoExportCIDRs	`autoExportPolicy` が有効になっている場合にKubernetesのノードIPをフィルタリングするためのCIDRのリスト。`autoExportPolicy` および `autoExportCIDRs` オプションを使用すると、Tridentはエクスポート ポリシーを自動的に管理できます。	"["0.0.0.0/0", "::/0"]"
labels	ボリュームに適用する任意のJSON 形式のラベルのセット	""
clientCertificate	クライアント証明書の Base64 エンコードされた値。証明書ベースの認証に使用	""
clientPrivateKey	クライアント秘密キーの Base64 エンコードされた値。証明書ベースの認証に使用	""
trustedCACertificate	信頼された CA 証明書の Base64 エンコードされた値。任意。証明書ベースの認証に使用されます。	""
username	クラスタまたは SVM に接続するためのユーザー名。クレデンシャルベースの認証に使用されます。たとえば、vsadmin。	
password	クラスターまたは SVM に接続するためのパスワード。クレデンシャルベースの認証に使用されます。	
svm	使用するStorage Virtual Machine	SVM 管理 LIF が指定されている場合に派生されます。
storagePrefix	SVMで新しいボリュームをプロビジョニングする際に使用されるプレフィックス。作成後は変更できません。このパラメータを更新するには、新しいバックエンドを作成する必要があります。	trident

パラメータ	概要	例
limitAggregateUsage	*Amazon FSx for NetApp ONTAP には指定しないでください。*提供された `fsxadmin` と `vsadmin` には、Trident を使用してアグリゲートの使用状況を取得して制限するために必要な権限が含まれていません。	使用しないでください。
limitVolumeSize	要求されたボリューム サイズがこの値を超える場合、プロビジョニングは失敗します。また、qtree と LUN を管理するボリュームの最大サイズを制限し、`qtreesPerFlexvol` オプションにより、FlexVol volume あたりの qtree の最大数をカスタマイズできます	"" (デフォルトでは強制されません)
lunsPerFlexvol	FlexVol volume あたりの最大 LUN 数は、[50, 200] の範囲にする必要があります。SAN のみ。	"100"
debugTraceFlags	トラブルシューティング時に使用するデバッグ フラグ。例 : {"api":false, "method":true} `debugTraceFlags` を使用しないでください。ただし、トラブルシューティングを行っており、詳細なログ ダンプが必要な場合を除きます。	null
nfsMountOptions	NFS マウントオプションをカンマ区切りで指定します。Kubernetes 永続ボリュームのマウント オプションは通常ストレージ クラスで指定されますが、ストレージ クラスでマウント オプションが指定されていない場合、Trident はストレージ バックエンドの構成ファイルで指定されたマウント オプションを使用するようになります。ストレージ クラスまたは構成ファイルにマウント オプションが指定されていない場合、Trident は関連付けられている永続ボリュームにマウント オプションを設定しません。	""
nasType	NFS または SMB ボリュームの作成を設定します。オプションは nfs、smb、または null です。*SMB ボリュームの場合は `smb` に設定する必要があります。*null に設定すると、デフォルトで NFS ボリュームになります。	nfs

パラメータ	概要	例
qtreesPerFlexvol	FlexVol volume あたりの最大 qtree 数は、[50, 300] の範囲内である必要があります	"200"
smbShare	次のいずれかを指定できます ：Microsoft 管理コンソールまたは ONTAP CLI を使用して作成された SMB 共有の名前、または Trident が SMB 共有を作成できるようにするための名前。このパラメータは、Amazon FSx for NetApp ONTAP バックエンドに必要です。	smb-share
useREST	ONTAP REST APIを使用するためのブーリアン パラメータ。`true` に設定すると、TridentはONTAP REST APIを使用してバックエンドと通信します。この機能にはONTAP 9.11.1以降が必要です。さらに、使用するONTAPログインロールには、`ontap`アプリケーションへのアクセス権が必要です。これは、事前定義された `vsadmin` および `cluster-admin` ロールで満たされます。	false
aws	AWS FSx for ONTAP の設定ファイルでは以下を指定できます： fsxFilesystemID：AWS FSx ファイルシステムの ID を指定します。 apiRegion：AWS API リージョン名。 apikey：AWS API キー。 secretKey：AWS 秘密キー。	"" "" ""
credentials	AWS Secrets Manager に保存する FSx SVM 認証情報を指定します。 - name：SVM の認証情報が含まれるシークレットの Amazon リソースネーム (ARN) 。 - type：`awsarn` に設定します。詳細については、 "AWS Secrets Manager シークレットを作成する" を参照してください。	

ボリュームのプロビジョニング用のバックエンド設定オプション

デフォルトのプロビジョニングは、設定の `defaults` セクションにあるこれらのオプションを使用して制御できます。例については、以下の設定例を参照してください。

パラメータ	概要	デフォルト
spaceAllocation	LUNのスペース割り当て	true

パラメータ	概要	デフォルト
spaceReserve	スペース予約モード：「none」（シン）または「volume」（シック）	none
snapshotPolicy	使用するSnapshotポリシー	none
qosPolicy	作成されたボリュームに割り当てる QoS ポリシー グループ。ストレージ プールまたはバックエンドごとにqosPolicyまたはadaptiveQosPolicyのいずれかを選択してください。Trident で QoS ポリシー グループを使用するには、ONTAP 9.8 以降が必要です。共有されていない QoS ポリシー グループを使用し、ポリシー グループが各構成要素に個別に適用されるようにする必要があります。共有 QoS ポリシー グループは、すべてのワークロードの合計スループットの上限を適用します。	""
adaptiveQosPolicy	作成されたボリュームに割り当てるアダプティブ QoS ポリシー グループ。ストレージ プールまたはバックエンドごとにqosPolicyまたはadaptiveQosPolicyのいずれかを選択してください。ontap-nas-economy ではサポートされていません。	""
snapshotReserve	スナップショット用に予約されているボリュームの割合「0」	もし snapshotPolicy`が`none`の場合、`else`""
splitOnClone	作成時にクローンを親から分離する	false
encryption	新しいボリュームで NetApp Volume Encryption (NVE) を有効にします。デフォルトは`false`です。このオプションを使用するには、クラスター上で NVE のライセンスを取得し、有効化する必要があります。バックエンドでNAEが有効になっている場合、TridentでプロビジョニングされたボリュームはすべてNAEが有効になります。詳細については、次を参照してください： "Tridentと NVE および NAE の連携"	false
luksEncryption	LUKS暗号化を有効にします。 "Linux Unified Key Setup (LUKS) を使用する" を参照してください。SANのみ。	""

パラメータ	概要	デフォルト
tieringPolicy	使用する階層化ポリシー none	
unixPermissions	新しいボリュームのモード。 SMB ボリュームの場合は空白のままにします。	""
securityStyle	新しいボリュームのセキュリティスタイル。NFSは`mixed`および`unix`セキュリティスタイルをサポートします。SMBは`mixed`および`ntfs`セキュリティスタイルをサポートします。	NFSのデフォルトは`unix`です。SMBのデフォルトは`ntfs`です。

PVCの設定

このセクションでは、設定されたKubernetes StorageClassを使用してPVを要求するPersistentVolumeClaim（PVC）を作成する方法について説明します。成功すると、PVをpodにマウントできます。

PVC を作成する

A "[PersistentVolumeClaim](#)"（PVC）は、クラスタ上のPersistentVolumeへのアクセス要求です。PVCは、特定のサイズまたはアクセスモードのストレージを要求するように構成できます。関連するStorageClassを使用して、クラスタ管理者は、PersistentVolumeのサイズとアクセスモード以外にも、パフォーマンスやサービスレベルなどを制御できます。

TridentバックエンドとStorageClassを作成したら、PVCを作成できます。PVCを作成したら、そのボリュームをポッドにマウントできます。

サンプルマニフェスト

以下の例は、基本的なPVC構成オプションを示しています。

RWX アクセス付き PVC

この例では、RWXアクセスを持つ基本的なPVCが、`basic-csi`という名前のStorageClassに関連付けられています。

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

```
storageClassName: ontap-gold
```

iSCSI を使用した PVC の例

この例では、RWOアクセスを持つiSCSI用の基本PVCが、StorageClassという名前 `protection-gold` に関連付けられています。

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-san
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: protection-gold
```

PVCを作成する

手順

1. PVC を作成します。

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

2. PVC ステータスを確認します。

```
kubectl get pvc
```

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

"[KubernetesとTridentオブジェクト](#)"を参照して、ストレージクラスが `PersistentVolumeClaim` とどのように相互作用するか、および Trident がボリュームをプロビジョニングする方法を制御するパラメータの詳細を確認してください。

アプリケーションをデプロイする

ストレージクラスとPVCが作成されると、PVをポッドにマウントできます。このセクションでは、PVをポッドに接続するためのコマンドと構成の例を示します。

サンプルアプリケーションを導入する

手順

1. ボリュームをポッドにマウントします。

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

以下の例は、PVC をポッドに接続するための基本構成を示しています。基本構成：

```
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
```



進捗状況は `kubectl get pod --watch` で確認できます。

2. ボリュームが `/my/mount/path` にマウントされていることを確認します。

```
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	

これでPodを削除できます。Podアプリケーションは存在しなくなりますが、ボリュームは残ります。

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

EKS クラスター上の Trident EKS アドオンを設定する

NetApp Tridentは、Amazon FSx for NetApp ONTAPのKubernetesストレージ管理を合理化し、開発者と管理者がアプリケーションの導入に集中できるようにします。NetApp Trident EKSアドオンには、最新のセキュリティパッチとバグ修正が含まれており、Amazon EKSで動作することがAWSによって検証されています。EKS アドオンを使用すると、Amazon EKS クラスターの安全性と安定性を常に確保し、アドオンのインストール、設定、更新に必要な作業量を削減できます。

前提条件

AWS EKS の Trident アドオンを設定する前に、以下のものを用意してください：

- アドオンを操作する権限を持つ Amazon EKS クラスターアカウント。["Amazon EKS アドオン"](#)を参照してください。
- AWS マーケットプレイスへの AWS 権限：
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",
"aws-marketplace:Unsubscribe"
- AMI タイプ：Amazon Linux 2 (AL2_x86_64) または Amazon Linux 2 Arm (AL2_ARM_64)
- ノードタイプ：AMDまたはARM
- 既存の Amazon FSx for NetApp ONTAP ファイルシステム

手順

1. EKS ポッドがAWS リソースにアクセスできるようにするには、IAM ロールとAWS シークレットを作成してください。手順については、["IAMロールとAWSシークレットを作成する"](#)を参照してください。
2. EKS Kubernetes クラスターで、* アドオン * タブに移動します。



① 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

① 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

① New versions are available for 1 add-on.



Add-ons (3) Info

View details

Edit

Remove

Get more add-ons

Q Find add-on

Any categ...

Any status

3 matches

< 1 >

3. **AWS Marketplace** アドオン に移動し、*storage* カテゴリを選択します。

AWS Marketplace add-ons (1)



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

Q Find add-on

Filtering options

Any category

NetApp, Inc.

Any pricing model

Clear filters

NetApp, Inc. X

< 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
storageListed by
[NetApp, Inc.](#)Supported versions
1.31, 1.30, 1.29, 1.28,
1.27, 1.26, 1.25, 1.24,
1.23Pricing starting at
[View pricing details](#)

Cancel

Next

4. **NetApp Trident** を見つけて、Trident アドオンのチェックボックスをオンにし、次へ をクリックします。

5. アドオンの目的のバージョンを選択します。

Configure selected add-ons settings

Configure the add-ons for your cluster by selecting settings.

NetApp TridentRemove add-on

Listed by

Category

Status

NetApp

storage

Ready to install

You're subscribed to this software

View subscription

You can view the terms and pricing details for this product or choose another offer if one is available.

Version

Select the version for this add-on.

v25.6.0-eksbuild.1

Optional configuration settings

Cancel

Previous

Next

6. 必要なアドオン設定を構成します。

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 >

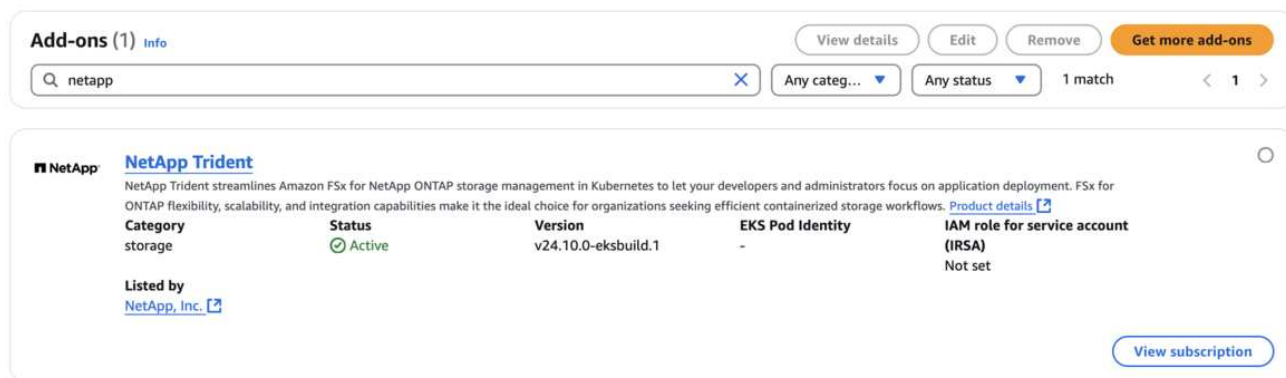
Add-on name	IAM role	Service account
No Pod Identity associations		
None of the selected add-on(s) have Pod Identity associations.		

[Cancel](#)[Previous](#)[Create](#)

7. IRSA（サービスアカウントのIAMロール）を使用している場合は、追加の構成手順を参照してください"[ここをクリックしてください。](#)"。

8. *作成*を選択します。

9. アドオンのステータスが **Active** であることを確認します。



10. 次のコマンドを実行して、Trident がクラスタに正しくインストールされていることを確認します：

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

11. セットアップを続行し、ストレージ バックエンドを構成します。詳細については、"[ストレージバックエンドを設定する](#)"を参照してください。

CLI を使用した Trident EKS アドオンのインストール / アンインストール

CLI を使用して **NetApp Trident EKS** アドオンをインストールします：

次のコマンド例では、Trident EKS アドオンをインストールします：

```
eksctl create addon --cluster clusterName --name netapp_trident-operator  
--version v25.6.0-eksbuild.1 (専用バージョンを使用)
```

以下のコマンド例は Trident EKS アドオンバージョン 25.6.1 をインストールします：

```
eksctl create addon --cluster clusterName --name netapp_trident-operator  
--version v25.6.1-eksbuild.1 (専用バージョンを使用)
```

以下のコマンド例は Trident EKS アドオンバージョン 25.6.2 をインストールします：

```
eksctl create addon --cluster clusterName --name netapp_trident-operator  
--version v25.6.2-eksbuild.1 (専用バージョンを使用)
```

CLI を使用して **NetApp Trident EKS** アドオンをアンインストールします：

次のコマンドは、Trident EKS アドオンをアンインストールします：

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

著作権に関する情報

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。