



# NetApp ONTAP 対応の Amazon FSX Trident

NetApp  
February 02, 2026

# 目次

NetApp ONTAP 対応の Amazon FSX .....	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 (AMIS) .....	3
詳細については、こちらをご覧ください .....	3
IAMロールとAWS Secretを作成する .....	4
AWS Secrets Managerシークレットの作成 .....	4
IAMポリシーの作成 .....	4
Trident をインストール .....	9
Helmを使用したTridentのインストール .....	9
EKSアドオンを使用してTridentをインストールする .....	11
ストレージバックエンドの設定 .....	17
ONTAP SANとNASドライバの統合 .....	17
FSx for ONTAPドライバの詳細 .....	19
バックエンドの高度な設定と例 .....	20
ボリュームのプロビジョニング用のバックエンド構成オプション .....	24
SMBボリュームのプロビジョニング .....	26
ストレージクラスとPVCを設定する .....	26
ストレージクラスを作成する。 .....	26
PVCの作成 .....	28
Trident属性 .....	30
サンプルアプリケーションのデプロイ .....	31
EKSクラスタでのTrident EKSアドオンの設定 .....	32
前提条件 .....	33
手順 .....	33
CLIを使用したTrident EKSアドオンのインストールとアンインストール .....	36

# NetApp ONTAP 対応の Amazon FSX

## Amazon FSx for NetApp ONTAPでTridentを使用

"[NetApp ONTAP 対応の Amazon FSX](#)" 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の起動や実行を可能にする、フルマネージドのAWSサービスです。FSX for ONTAP を使用すると、使い慣れたネットアップの機能、パフォーマンス、管理機能を活用しながら、AWSにデータを格納するためのシンプルさ、即応性、セキュリティ、拡張性を活用できます。FSX for ONTAP は、ONTAP ファイルシステムの機能と管理APIをサポートしています。

Amazon FSx for NetApp ONTAPファイルシステムをTridentと統合すると、Amazon Elastic Kubernetes Service (EKS) で実行されているKubernetesクラスタが、ONTAPを基盤とするブロックおよびファイルの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。

ファイルシステムは、オンプレミスの ONTAP クラスタに似た、Amazon FSX のプライマリリソースです。各 SVM 内には、ファイルとフォルダをファイルシステムに格納するデータコンテナである 1 つ以上のボリュームを作成できます。Amazon FSx for NetApp ONTAPは、クラウドのマネージドファイルシステムとして提供されます。新しいファイルシステムのタイプは \* NetApp ONTAP \* です。

TridentとAmazon FSx for NetApp ONTAPを使用すると、Amazon Elastic Kubernetes Service (EKS) で実行されているKubernetesクラスタが、ONTAPを基盤とするブロックおよびファイルの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](#)"FSx for ONTAPとTridentを統合するには、さらに次のものがが必要です。

- 既存の Amazon EKS クラスタまたは 'kubectl' がインストールされた自己管理型 Kubernetes クラスタ
- クラスタのワーカーノードから到達可能な既存のAmazon FSx for NetApp ONTAPファイルシステムおよびStorage Virtual Machine (SVM) 。
- 準備されているワーカーノード "[NFSまたはiSCSI](#)"。



Amazon LinuxおよびUbuntuで必要なノードの準備手順を実行します "[Amazon Machine Images の略](#)" (AMIS) EKS の AMI タイプに応じて異なります。

### 考慮事項

- SMBボリューム：
  - SMBボリュームは、を使用してサポートされます `ontap-nas` ドライバーのみ。
  - SMBボリュームは、Trident EKSアドオンではサポートされません。
  - Tridentでは、Windowsノードで実行されているポッドにマウントされたSMBボリュームのみがサポートされます。詳細については、を参照してください "[SMBボリュームをプロビジョニングする準備をします](#)"。
- Trident 24.02より前のバージョンでは、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

テム上に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以降でこの問題を回避するには、AWS FSx for ONTAPのバックエンド構成ファイルで、`apiRegion`AWS、AWS、およびAWS `apikey`を `secretKey`指定します `fsxFilesystemID`。



TridentにIAMロールを指定する場合は、`apiKey`、および `secretKey`の各フィールドをTridentに明示的に指定する必要はありません `apiRegion`。詳細については、を参照してください ["FSX \(ONTAP の構成オプションと例\)"](#)。

## Trident SAN/iSCSI と EBS-CSI ドライバーの同時使用

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 ディスクに干渉せずにマルチパスが機能することを保証するには、マルチパス設定で EBS を除外する必要があります。この例では、`multipath.conf` EBS ディスクをマルチパスから除外しながら必要なTrident設定を含むファイル:

```
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。

- クレデンシャルベース（推奨）：クレデンシャルをAWS Secrets Managerに安全に格納します。ファイルシステムのユーザ、またはSVM用に設定されているユーザを使用できます `fsxadmin` `vsadmin`。



Tridentは、SVMユーザ、または別の名前で同じロールのユーザとして実行することを想定しています `vsadmin`。Amazon FSx for NetApp ONTAPには、ONTAPクラスタユーザに代わる限定的なユーザが `admin` `い` `fsxadmin`ます。Tridentでの使用を強くお勧めします `vsadmin`。

- 証明書ベース：Tridentは、SVMにインストールされている証明書を使用して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、使用しているドライバ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 (AMIS)

EKSクラスタ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ではコンテナとEKS用に特定のAmazon Machine Images (AMIS) が最適化されています。次のAMI は NetApp Trident 25.02 でテストされています。

亜美	NAS	NASエコノミー	iSCSI	iSCSIエコノミー
AL2023_x86_64_STANDARD	はい。	はい。	はい。	はい。
AL2_x86_64	はい。	はい。	はい*	はい*
BOTTLEROCKET_x86_64	はい**	はい。	N/A	N/A
AL2023_ARM_64_STANDARD	はい。	はい。	はい。	はい。
AL2_ARM_64	はい。	はい。	はい*	はい*
BOTTLEROCKET_ARM_64	はい**	はい。	N/A	N/A

- \* 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。



目的のAMIがここにリストされていない場合、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ことを意味します。このリストは、AMI が動作することがわかっている場合のガイドとして機能します。

### テスト実施項目：

- 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の両方をテストしました。
- SANについてはiSCSIのみをテストし、NVMe-oFはテストしませんでした。

### 実行されたテスト：

- 作成：ストレージクラス、PVC、POD
- 削除：ポッド、PVC（通常、qtree / LUN-エコノミー、NASとAWSバックアップ）

### 詳細については、こちらをご覧ください

- ["Amazon FSX for NetApp ONTAP のドキュメント"](#)
- ["Amazon FSX for NetApp ONTAP に関するブログ記事です"](#)

# IAMロールとAWS Secretを作成する

KubernetesポッドがAWSリソースにアクセスするように設定するには、明示的なAWSクレデンシャルを指定する代わりに、AWS IAMロールとして認証します。



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シークレットの作成

TridentはFSx SVMに対してAPIを発行してストレージを管理するため、そのためにはクレデンシャルが必要になります。これらのクレデンシャルを安全に渡すには、AWS Secrets Managerシークレットを使用します。そのため、AWS Secrets Managerシークレットをまだ作成していない場合は、vsadminアカウントのクレデンシャルを含むシークレットを作成する必要があります。

次の例では、Trident CSIクレデンシャルを格納する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ポリシーの作成

Tridentを正しく実行するには、AWSの権限も必要です。そのため、必要な権限をTridentに付与するポリシーを作成する必要があります。

次の例は、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"
}

```

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

Kubernetes サービスアカウントを設定して、EKS ポッド ID またはサービスアカウントの関連付け (IRSA) 用の IAM ロールを持つ AWS Identity and Access Management (IAM) ロールを引き受けることができます。これにより、この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、その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。

## ポッドのアイデンティティ

Amazon EKS ポッドアイデンティティの関連付けは、Amazon EC2 インスタンスプロファイルが Amazon EC2 インスタンスに認証情報を提供するのと同様に、アプリケーションの認証情報を管理する機能を提供します。

**EKS** クラスターに **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 ID 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 ポッドアイデンティティエージェントを設定する](#)"。

**trust-relationship.json** を作成:

EKS サービスプリンシパルがポッド ID に対してこのロールを引き受けられるように、trust-relationship.json を作成します。次に、以下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 の関連付けを作成する:

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

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

ファイルの次の値を更新し `trust-relationship.json` ます。

- **<account\_id>**-お客様のAWSアカウントID
- **<oidc\_provider>**- EKSクラスタのOIDC。oidc\_providerを取得するには、次のコマンドを実行します。

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

- IAMポリシーにIAMロールを関連付ける\*：

ロールを作成したら、次のコマンドを使用して（上記の手順で作成した）ポリシーをロールに関連付けます。

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

- OICDプロバイダが関連付けられていることを確認します\*：

OIDCプロバイダがクラスタ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でAmazon FSx for NetApp ONTAPストレージ管理を合理化し、開発者や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。

次のいずれかの方法でTridentを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アドオンをインストールします。詳細については、[を参照してください "CSIボリュームのスナップショット機能を有効にする"](#)。

### Helmを使用したTridentのインストール

## ポッドのアイデンティティ

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
```

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

```
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. クラウド プロバイダー と クラウド ID 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
```

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

```
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。AL2023Worker node OS を使用している場合は、helm インストール時に `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アドオンを使用してTridentをインストールする

Trident EKSアドオンには、最新のセキュリティパッチ、バグ修正が含まれており、AWSによってAmazon EKSと連携することが検証されています。EKSアドオンを使用すると、Amazon EKSクラスタの安全性と安定性を一貫して確保し、アドオンのインストール、構成、更新に必要な作業量を削減できます。

### 前提条件

AWS EKS用のTridentアドオンを設定する前に、次の条件を満たしていることを確認してください。

- アドオンサブスクリプションがあるAmazon EKSクラスタアカウント
- AWS Marketplace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向けTridentアドオンを有効にする

## 管理コンソール

1. で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、\*[クラスタ]\*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、\*追加のアドオン\*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選択します。
  - c. 「\* 次へ \*」を選択します。
6. [Configure selected add-ons\* settings]ページで、次の手順を実行します。



**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 (値は次の形式である必要があります)。

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

+

[Conflict resolution method]で[Override]を選択すると、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しない場合、既存の設定と競合すると、操作は失敗します。表示された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に必要な設定を管理していないことを確認してください。

7. [次へ]\*を選択します。
8. [確認して追加]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、インストールされているアドオンが表示されます。

## AWS CLI

## 1.作成する `add-on.json` ファイル：

**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"
  }
}
```



を、前の手順で作成したロールのARNに置き換えます <role ARN>。

## 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アドオンの更新



## 管理コンソール

1. Amazon EKSコンソールを開き <https://console.aws.amazon.com/eks/home#/clusters> ます。
2. 左側のナビゲーションペインで、\*[クラスタ]\*を選択します。
3. NetApp Trident CSIアドオンを更新するクラスタ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、Edit \*を選択します。
6. [Configure Trident by NetApp \*]ページで、次の手順を実行します。
  - a. 使用する\*バージョン\*を選択します。
  - b. [Optional configuration settings]\*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アドオンを更新します。

```
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

- お使いの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。をクラスタ名に置き換え `my-cluster` 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'\"}			

- 前の手順の出力でupdate availableで返されたバージョンにアドオンを更新します。

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

オプションを削除し、いずれかのAmazon EKSアドオン設定が既存の設定と競合している場合 `--force`、Amazon EKSアドオンの更新は失敗します。競合の解決に役立つ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、管理する必要がある設定がAmazon EKSアドオンで管理されていないことを確認してください。これらの設定はこのオプションで上書きされます。この設定のその他のオプションの詳細については、を参照してください ["アドオン"](#)。Amazon EKS Kubernetesフィールド管理の詳細については、を参照してください ["Kubernetesフィールド管理"](#)。

## Trident EKSアドオンのアンインストール/削除

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。

- クラスタにアドオンソフトウェアを保持–このオプションを選択すると、Amazon EKSによる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、コマンドのオプションをそのまま使用し `--preserve` ます。
- クラスターからアドオンソフトウェアを完全に削除する–NetAppは、クラスターに依存するリソースがない場合にのみ、クラスターからAmazon EKSアドオンを削除することを推奨します。コマンドからオプションを削除してアドオンを削除し `--preserve delete` ます。



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. で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、\*[クラスタ]\*を選択します。
3. NetApp Trident CSIアドオンを削除するクラスタの名前を選択します。
4. アドオン\*タブを選択し、Trident by NetApp を選択します。
5. 「\* 削除」を選択します。
6. [Remove netapp\_trident-operator confirmation]\*ダイアログで、次の手順を実行します。
  - a. Amazon EKSでアドオンの設定を管理しないようにするには、\*[クラスタに保持]\*を選択します。クラスタにアドオンソフトウェアを残して、アドオンのすべての設定を自分で管理できるようにする場合は、この手順を実行します。
  - b. 「netapp\_trident -operator \*」と入力します。
  - c. 「\* 削除」を選択します。

## AWS CLI

をクラスタの名前に置き換え my-cluster 、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
```

# ストレージバックエンドの設定

## ONTAP SANとNASドライバの統合

ストレージバックエンドを作成するには、JSONまたはYAML形式の構成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タイプ（NASまたはSAN）、ファイルの取得元のファイルシステム、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にクレデンシャルを格納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"
    }
  }
}
```

次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	

## FSx for ONTAPドライバの詳細

次のドライバを使用して、TridentとAmazon FSx for NetApp ONTAPを統合できます。

- **ontap-san**：プロビジョニングされる各PVは、それぞれの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。
- **ontap-nas**：プロビジョニングされる各PVは、完全なAmazon FSx for NetApp ONTAPボリュームです。NFSとSMBで推奨されます。
- 「ONTAP と SAN の経済性」：プロビジョニングされた各 PV は、NetApp ONTAP ボリュームの Amazon FSX ごとに構成可能な数の LUN を持つ LUN です。
- 「ONTAP-NAS-エコノミー」：プロビジョニングされた各 PV は qtree であり、NetApp ONTAP ボリュームの Amazon FSX ごとに設定可能な数の qtree があります。
- 「ONTAP-NAS-flexgroup」：プロビジョニングされた各 PV は、NetApp ONTAP FlexGroup ボリューム用の完全な Amazon FSX です。

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

構成ファイルが作成されたら、次のコマンドを実行してEKS内に作成します。

```
kubectl create -f configuration_file
```

ステータスを確認するには、次のコマンドを実行します。

```
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	

## バックエンドの高度な設定と例

バックエンド設定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	説明	例
「バージョン」		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
「管理 LIF」	<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）を指定できます。IPv6フラグを使用してTridentがインストールされている場合は、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、[28e8：d9fb：a825：b7bf：69a8：d02f：9e7b：3555]などの角かっこで定義する必要があります。aws`フィールドでを指定する場合は`fsxFilesystemID、を指定する必要はありません。managementLIF`。TridentはAWSからSVM情報を取得するためです。</p> <p>`managementLIF`そのため、SVMの下ユーザ（vsadminなど）のクレデンシャルを指定し、そのユーザにロールが割り当てられている必要があります。`vsadmin`。</p>	"10.0.0.1 ","[2001：1234：abcd：：fe]"

パラメータ	説明	例
「重複排除	<p>プロトコル LIF の IP アドレス。* ONTAP NASドライバ*: NetAppではdataLIFの指定を推奨しています。指定しない場合、TridentはSVMからデータLIFをフェッチします。NFSのマウント処理に使用するFully Qualified Domain Name (FQDN; 完全修飾ドメイン名) を指定すると、ラウンドロビンDNSを作成して複数のデータLIF間で負荷を分散できます。初期設定後に変更できます。を参照してください。* ONTAP SANドライバ*: iSCSIには指定しないでくださいTridentは、ONTAP選択的LUNマップを使用して、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。データLIFが明示的に定義されている場合は警告が生成されます。IPv6フラグを使用してTridentがインストールされている場合は、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、[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]などの角かっこで定義する必要があります。</p>	
「autoExportPolicy」を参照してください	<p>エクスポートポリシーの自動作成と更新を有効にします[ブーリアン]。オプションと`autoExportCIDRs`オプションを使用する`autoExportPolicy`と、Tridentでエクスポートポリシーを自動的に管理できます。</p>	「偽」
「autoExportCI」	<p>が有効な場合にKubernetesのノードIPをフィルタリングするCIDRのリスト autoExportPolicy。オプションと`autoExportCIDRs`オプションを使用する`autoExportPolicy`と、Tridentでエクスポートポリシーを自動的に管理できます。</p>	"["0.0.0.0/0", ": : /0"]"
「ラベル」	<p>ボリウムに適用する任意のJSON形式のラベルのセット</p>	""
「clientCertificate」をクリックします	<p>クライアント証明書のBase64エンコード値。証明書ベースの認証に使用されます</p>	""
「clientPrivateKey」	<p>クライアント秘密鍵のBase64エンコード値。証明書ベースの認証に使用されます</p>	""

パラメータ	説明	例
「 trustedCacertifate 」	信頼された CA 証明書の Base64 エンコード値。任意。証明書ベースの認証に使用されます。	""
「ユーザ名」	クラスタまたはSVMに接続するためのユーザ名。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のように指定します。	
「 password 」と入力します	クラスタまたは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。	
'VM'	使用する Storage Virtual Machine	SVM管理LIFが指定されている場合に生成されます。
'toragePrefix'	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するプレフィックスを指定します。作成後に変更することはできません。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
「 AggreqlimitateUsage 」と入力します	* Amazon FSx for NetApp ONTAP には指定しないでください。*指定された vsadmin`には`fsxadmin、アグリゲートの使用量を取得してTridentを使用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 」と入力します	要求されたボリュームサイズがこの値を超えている場合、プロビジョニングが失敗します。また、qtreeおよびLUNに対して管理するボリュームの最大サイズを制限し、オプションを使用すると、FlexVol volumeあたりのqtreeの最大数をカスタマイズできます。 qtreesPerFlexvol	""（デフォルトでは適用されません）
'lunsPerFlexvol	FlexVol volumeあたりの最大LUN数は[50、200]の範囲で指定する必要があります。SANのみ。	"100"
「バグトレースフラグ」	<p>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 }</p> <p>使用しないでください debugTraceFlags トラブルシューティングを実行していて、詳細なログダンプが必要な場合を除きます。</p>	null



パラメータ	説明	例
「 <code>nfsvMountOptions</code> 」のように入力します	NFSマウントオプションをカンマで区切ったリスト。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、ストレージクラスに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に指定されているマウントオプションを使用してフォールバックします。ストレージクラスまたは構成ファイルでマウントオプションが指定されていない場合、Tridentは関連付けられた永続ボリュームにマウントオプションを設定しません。	""
<code>nasType</code>	NFSボリュームまたはSMBボリュームの作成を設定オプションは <code>nfs</code> 、 <code>smb</code> 、または <code>null</code> 。*をに設定する必要があります <code>smb</code> SMB ボリューム。*を <code>null</code> に設定すると、デフォルトでNFSボリュームが使用されます。	<code>nfs</code>
<code>qtreesPerFlexvol`</code>	FlexVol volumeあたりの最大qtree数は[50、300]の範囲で指定する必要があります。	"200"
<code>smbShare</code>	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にSMB共有の作成を許可する名前。このパラメータは、Amazon FSx for ONTAPバックエンドに必要です。	<code>smb-share</code>
「 <code>useREST`</code> 」	ONTAP REST API を使用するためのブーリアンパラメータ。に設定する <code>true`</code> 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。また、使用するONTAPログインロールには、アプリケーションへのアクセス権が必要です <code>`ontap</code> 。これは、事前に定義された役割と役割によって実現され <code>vsadmin cluster-admin</code> ます。	「偽」

パラメータ	説明	例
aws	<p>AWS FSx for ONTAPの構成ファイルでは、次の項目を指定できます。</p> <ul style="list-style-type: none"> <li>- fsxFilesystemID：AWS FSx ファイルシステムのIDを指定します。</li> <li>- apiRegion：AWS APIリージョン名。</li> <li>- apikey：AWS APIキー。</li> <li>- secretKey：AWSシークレットキー。</li> </ul>	<pre>"" "" ""</pre>
credentials	<p>AWS Secrets Managerに保存するFSx SVMのクレデンシャルを指定します。- name：シークレットのAmazonリソース名（ARN）。SVMのクレデンシャルが含まれています。- type：に設定します awsarn。詳細については、を参照してください <a href="#">"AWS Secrets Managerシークレットの作成"</a>。</p>	

## ボリュームのプロビジョニング用のバックエンド構成オプション

これらのオプションを使用して、のデフォルトプロビジョニングを制御できます defaults 設定のセクション。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	説明	デフォルト
「平和の配分」	space-allocation for LUN のコマンドを指定します	「真」
「平和のための準備」を参照してください	スペースリザーベーションモード： 「none」（シン）または「volume」（シック）	「NONE」
「ナプショットポリシー」	使用する Snapshot ポリシー	「NONE」

パラメータ	説明	デフォルト
「 QOSPolicy 」	作成し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ます。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コンスチチュエントに個別に適用されるようにします。QoSポリシーグループを共有すると、すべてのワークロードの合計スループットの上限が適用されます。	""
「 adaptiveQosPolicy 」を参照してください	アダプティブ QoS ポリシーグループ：作成したボリュームに割り当てます。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ます。経済性に影響する ONTAP - NAS ではサポートされません。	""
「スナップショット予約」	Snapshot 「0」 用にリザーブされているボリュームの割合	がの none `場合 `snapshotPolicy else、""
'plitOnClone	作成時にクローンを親からスプリットします	「偽」
「暗号化」	新しいボリュームでNetApp Volume Encryption (NVE) を有効にします。デフォルトはです。 `false`このオプションを使用するには、クラスタで NVE のライセンスが設定され、有効になっている必要があります。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すべてのボリュームでNAEが有効になります。詳細については、を参照してください" <a href="#">TridentとNVEおよびNAEとの連携</a> "。	「偽」
luksEncryption	LUKS暗号化を有効にします。を参照してください " <a href="#">Linux Unified Key Setup (LUKS；統合キーセットアップ)</a> を使用"。SANのみ。	""
階層ポリシー	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 ボリュームは空にしておきます。*	""

パラメータ	説明	デフォルト
'securityStyle'	新しいボリュームのセキュリティ形式。NFSのサポート mixed および unix セキュリティ形式SMBはをサポートします mixed および ntfs セキュリティ形式	NFSのデフォルトはです unix 。SMBのデフォルトはです ntfs。

## SMBボリュームのプロビジョニング

SMBボリュームをプロビジョニングするには、`ontap-nas`ドライバ。完了する前に [ONTAP SANとNASドライバの統合](#) 次の手順を実行します。"[SMBボリュームをプロビジョニングする準備をします](#)"。

## ストレージクラスとPVCを設定する

Kubernetes StorageClassオブジェクトを設定してストレージクラスを作成し、Tridentでボリュームのプロビジョニング方法を指定します。設定したKubernetes StorageClassを使用してPVへのアクセスを要求するPersistentVolumeClaim（PVC）を作成します。その後、PVをポッドにマウントできます。

ストレージクラスを作成する。

### Kubernetes StorageClassオブジェクトの設定

その "[Kubernetes StorageClassオブジェクト](#)"オブジェクトは、そのクラスに使用されるプロビジョナーとしてTridentを識別し、ボリュームをプロビジョニングする方法をTrident に指示します。NFS を使用するボリュームの Storageclass を設定するには、この例を使用します (属性の完全なリストについては、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 を使用するボリュームの Storageclass を設定するには、次の例を使用します。

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

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

```
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
```

ストレージクラスとパラメータおよびパラメータとの連携によるTridentによるボリュームのプロビジョニング方法の詳細については [PersistentVolumeClaim](#)、を参照してください"[Kubernetes オブジェクトと Trident オブジェクト](#)"。

ストレージクラスを作成する。

手順

1. これはKubernetesオブジェクトなので、`kubectl` をクリックして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ことを確認します。

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

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

## PVCの作成

<https://kubernetes.io/docs/concepts/storage/persistent-volumes>["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設定できます。クラスタ管理者は、関連付けられているStorageClassを使用して、PersistentVolumeのサイズとアクセスモード（パフォーマンスやサービスレベルなど）以上を制御できます。

PVCを作成したら、ボリュームをポッドにマウントできます。

マニフェストの例

## PersistentVolumeClaim サンプルマニフェスト

次に、基本的なPVC設定オプションの例を示します。

### RWXアクセスを備えたPVC

この例は、という名前のStorageClassに関連付けられたRWXアクセスを持つ基本的なPVCを示しています basic-csi。

```
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に関連付けられています。 protection-gold。

```
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

ストレージクラスとパラメータおよびパラメータとの連携によるTridentによるボリュームのプロビジョニング方法の詳細については PersistentVolumeClaim、を参照してください"[Kubernetes オブジェクトと Trident オブジェクト](#)"。

## Trident属性

これらのパラメータは、特定のタイプのボリュームのプロビジョニングに使用する Trident で管理されているストレージプールを決定します。

属性	を入力します	値	提供	リクエスト	でサポートされます
メディア ^1	文字列	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ています。ハイブリッドは両方を意味します	メディアタイプが指定されました	ONTAPNAS、ONTAPNAS エコノミー、ONTAP-NAS-flexgroup、ONTAPSAN、solidfire-san-SAN、solidfire-san-SAN のいずれかに対応しています
プロビジョニングタイプ	文字列	シン、シック	プールはこのプロビジョニング方法をサポートします	プロビジョニング方法が指定されました	シック：All ONTAP；thin：All ONTAP & solidfire-san-SAN
backendType	文字列	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ています	バックエンドが指定されて	すべてのドライバ



属性	を入力します	値	提供	リクエスト	でサポートされます
Snapshot	ブール値	true false	プールは、Snapshot を含むボリュームをサポートします	Snapshot が有効なボリューム	ontap-nas、ontapさん、solidfireさん
クローン	ブール値	true false	プールはボリュームのクローニングをサポートします	クローンが有効なボリューム	ontap-nas、ontapさん、solidfireさん
暗号化	ブール値	true false	プールでは暗号化されたボリュームをサポート	暗号化が有効なボリューム	ONTAP-NAS、ONTAP-NAS-エコノミー、ONTAP-NAS-FlexArray グループ、ONTAP-SAN
IOPS	整数	正の整数	プールは、この範囲内で IOPS を保証する機能を備えています	ボリュームで IOPS が保証されました	solidfire - SAN

^1 ^ : ONTAP Select システムではサポートされていません

## サンプルアプリケーションのデプロイ

ストレージクラスとPVCが作成されたら、そのPVをポッドにマウントできます。ここでは、PVをポッドに接続するためのコマンドと設定例を示します。

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
```

次に、PVCをポッドに接続するための基本的な設定例を示します。基本設定：

```

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

```



進捗状況は次を使用して監視できます。 `kubectl get pod --watch`。

2. ボリュームがマウントされていることを確認します。 `/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	

ポッドを削除できるようになりました。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クラスタでのTrident EKSアドオンの設定

NetApp Tridentは、KubernetesでAmazon FSx for NetApp ONTAPストレージ管理を合理化し、開発者や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、バグ修正が含まれており、AWS

によってAmazon EKSと連携することが検証されています。EKSアドオンを使用すると、Amazon EKSクラスタの安全性と安定性を一貫して確保し、アドオンのインストール、構成、更新に必要な作業量を削減できます。

## 前提条件

AWS EKS用のTridentアドオンを設定する前に、次の条件を満たしていることを確認してください。

- アドオンを使用する権限を持つAmazon EKSクラスタアカウント。を参照してください ["Amazon EKSアドオン"](#)。
- AWS Marketplace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ために、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、を参照してください ["IAMロールとAWS Secretを作成する"](#)。
2. EKS Kubernetesクラスタで、\*[アドオン]\*タブに移動します。

The screenshot shows the AWS EKS console interface for a cluster named 'tri-env-eks'. At the top, there are buttons for 'Delete cluster', 'Upgrade version', and 'View dashboard'. Below this is a notification bar stating: '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](#).' A 'Upgrade now' button is on the right. The main section is titled 'Cluster info' and includes details such as 'Status: Active', 'Kubernetes version: 1.30', 'Support period: Standard support until July 28, 2025', and 'Provider: EKS'. It also shows 'Cluster health issues: 0' and 'Upgrade insights: 0'. Below this is a navigation bar with tabs: 'Overview', 'Resources', 'Compute', 'Networking', 'Add-ons' (which is selected and has a '1' badge), 'Access', 'Observability', 'Update history', and 'Tags'. A notification bar below the tabs says: 'New versions are available for 1 add-on.' The 'Add-ons' section shows 'Add-ons (3)' with buttons for 'View details', 'Edit', 'Remove', and 'Get more add-ons'. There is a search bar with the placeholder 'Find add-on' and filters for 'Any categ...', 'Any status', and '3 matches'. A pagination control shows '< 1 >'.

3. [AWS Marketplace add-ons]\*にアクセスし、\_storage\_categoryを選択します。

### AWS Marketplace add-ons (1)

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

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**

<b>Category</b> storage	<b>Listed by</b> <a href="#">NetApp, Inc.</a>	<b>Supported versions</b> 1.31, 1.30, 1.29, 1.28, 1.27, 1.26, 1.25, 1.24, 1.23	<b>Pricing starting at</b> <a href="#">View pricing details</a>
----------------------------	--	---	--

Cancel
Next

- NetApp Trident を探し、**Trident**アドオンのチェックボックスを選択して Next \*をクリックします。
- 必要なアドオン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 Trident**
Remove add-on

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

**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](#) ✕

**Version**  
Select the version for this add-on.

v25.6.0-eksbuild.1 ▾

► Optional configuration settings

Cancel
Previous
Next

- 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)

&lt; 1 &gt;

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)

&lt; 1 &gt;

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

#### EKS Pod Identity (0)

&lt; 1 &gt;

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](#)

- 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。["こちらをご覧ください"](#)。
- 「\* Create \*」を選択します。
- アドオンのステータスが `_Active_` であることを確認します。

#### Add-ons (1) [Info](#)

[View details](#)[Edit](#)[Remove](#)[Get more add-ons](#)

Any categ...

Any status

1 match

&lt; 1 &gt;

#### 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](#)

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

Listed by  
[NetApp, Inc.](#)

[View subscription](#)

- 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設定します。詳細については、[を参照してください "ストレージバックエンドの設定"](#)。

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

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 (専用版あり)
```

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。