NetApp ONTAP 対応の Amazon FSX

Trident

NetApp
September 26, 2025

This PDF was generated from https://docs.netapp.com/ja-jp/trident-2410/trident-use/trident-fsx.html on September 26, 2025. Always check docs.netapp.com for the latest.

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） . . . 2
詳細情報 . . . 3
IAMロールとAWS Secretを作成する . . . 3
AWS Secret Managerシークレットの作成 . . . 3
IAMポリシーの作成 . . . 4
Trident をインストール . . . 6
Helmを使用したTridentのインストール . . . 6
EKSアドオンを使用してTridentをインストールする . . . 7
ストレージバックエンドの設定 . . . 13
ONTAP SANとNASドライバの統合 . . . 13
FSx for ONTAPドライバの詳細 . . . 15
バックエンドの高度な設定と例 . . . 16
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 . . . 20
SMBボリュームをプロビジョニングする準備をします . . . 21
ストレージクラスとPVCを設定する . . . 23
ストレージクラスを作成する。 . . . 23
PVおよびPVCの作成 . . . 24
Trident属性 . . . 27
サンプルアプリケーションのデプロイ . . . 28
EKSクラスタでのTrident EKSアドオンの設定 . . . 30
前提条件 . . . 30
手順 . . . 30
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 . . . 33

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を起動して実行できる、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 を使用すると、 Data ONTAP はクラウド内の管理対象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、さらに次のものが必要です。

- 既存のAmazon EKSクラスタまたはがインストールされた自己管理型Kubernetesクラスタ `kubectl`。
- 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）。
- 用に準備されたワーカーノード"NFSまたはiSCSI"。

> EKS AMIタイプに応じて、Amazon LinuxおよびUbuntu（AMIS）で必要なノードの準備手順に従って "Amazon Machine Images の略"ください。

考慮事項

- SMBボリューム：
  - SMBボリュームはドライバのみを使用してサポートされ `ontap-nas`ます。
  - 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）"。

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Trident 24.10でテストされています。

亜美	NAS	NASエコノミー	SAN	SAN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

- *マウントオプションでは「nolock」を使用する必要があります。
- **ノードを再起動せずにPVを削除できません

> 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、動作が確認されているAMISのガイドとして機能します。

テスト実施項目：

- EKS version: 1.30
- インストール方法：HelmとAWSアドオンとして
- 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）

詳細情報

- "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 Secret Managerシークレットの作成

次の例では、Trident CSIクレデンシャルを格納するAWSシークレットマネージャ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

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
Secret manager"
```

ポリシー**JSON**ファイル：

```
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"
}
```

サービスアカウント用の**IAM**ロールを作成する

AWS CLI

```
aws iam create-role --role-name trident-controller \
  --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
```

IAM OIDCをクラスタに関連付けるには、次のコマンドを使用します。

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

eksctl

次の例では、EKSでサービスアカウント用の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

1. Tridentインストーラパッケージのダウンロード

   Tridentインストーラパッケージには、Tridentオペレータの導入とTridentのインストールに必要なすべての

ものが含まれています。GitHubのAssetsセクションから最新バージョンのTridentインストーラをダウンロードして展開します。
wget https://github.com/NetApp/trident/releases/download/v24.10.0/trident-installer-24.10.0.tar.gz
tar -xf trident-installer-24.10.0.tar.gz
cd trident-installer/helm

2. 次の環境変数を使用して、* cloud provider フラグと cloud identity *フラグの値を設定します。

   次の例では、Tridentをインストールし、フラグをに設定し、 `cloud-identity`を `$CI`に `$CP`設定し `cloud-provider`ます。

```
helm install trident trident-operator-100.2410.0.tgz --set
cloudProvider="AWS" \

    --set cloudIdentity="'eks.amazonaws.com/role-arn:
arn:aws:iam::<accountID>:role/<AmazonEKS_FSxN_CSI_DriverRole>'" \
    --namespace trident --create-namespace
```

   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-100.2410.0    24.10.0
```

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_Linux_64 ARM）
- 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

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> \
    --service-account-role-arn
arn:aws:iam::<account_id>:role/<role_name> --force
```

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]*をクリックします。
3. NetApp Trident CSIアドオンを設定するクラスタの名前をクリックします。
4. をクリックし、[その他のアドオンの入手]*をクリックします。
5. [アドオンの選択]ページで、次の手順を実行します。
   a. [AWS Marketplace EKS-addons]セクションで、* Trident by NetApp *チェックボックスを選択します。
   b. 「* 次へ * 」をクリックします。
6. [Configure selected add-ons* settings]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. では、[Not set]*のままにします。
   c. *オプションの構成設定*を展開し、*アドオン構成スキーマ*に従って、*構成値*セクションのconfigurationValuesパラメーター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 `eks.amazonaws.com/role-arn: arn:aws:iam::464262061435:role/AmazonEKS_FSXN_CSI_DriverRole`）。[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. ファイルを作成し `add-on.json` ます。

```
add-on.json
{

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

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
```

Trident EKSアドオンの更新

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    v24.10.0-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 v24.10.0-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"。

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]*をクリックします。
3. NetApp Trident CSIアドオンを更新するクラスタの名前をクリックします。
4. [アドオン]タブをクリックします。
5. をクリックし、[編集（Edit）]*をクリックします。
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. [変更の保存 *] をクリックします。

AWS CLI

次の例では、EKSアドオンを更新します。

```
aws eks update-addon --cluster-name 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
```

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。
- クラスタからアドオンソフトウェアを完全に削除する–クラスターに依存するリソースがない場合にのみ、Amazon EKSアドオンをクラスターから削除することをお勧めします。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。

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
```

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]*をクリックします。
3. NetApp Trident CSIアドオンを削除するクラスタの名前をクリックします。
4. [アドオン]タブをクリックし、**[Trident by NetApp]***をクリックします。
5. [削除（ Remove ）] をクリックします。
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（ Remove ）] をクリックします。

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
```

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:xxxxxxxx:secret:secret-
name"
    type: awsarn
```

JSON

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

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-san-economy`：プロビジョニングされた各PVは、Amazon FSx for NetApp ONTAPボリュームごとに設定可能なLUN数を持つLUNです。
- `ontap-nas-economy`：プロビジョニングされる各PVはqtreeであり、Amazon FSx for NetApp ONTAPボリュームごとにqtree数を設定できます。
- `ontap-nas-flexgroup`：プロビジョニングされる各PVは、完全なAmazon FSx for NetApp ONTAP FlexGroupボリュームです。

ドライバの詳細については、およびを参照して"NASドライバ""SANドライバ"ください。

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	製品説明	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 + "_" + データ LIF
`managementLIF`	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。 `managementLIF`そのため、`SVM`の下のユーザ（`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`	「 10.0.0.1 」、「 [2001:1234:abcd::fefe] 」

パラメータ	製品説明	例
dataLIF	プロトコル LIF の IP アドレス。* ONTAP NASドライバ*：データLIFを指定することを推奨します。指定しない場合、TridentはSVMからデータLIFをフェッチします。NFS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）を指定して、ラウンドロビンDNSを作成して複数のデータLIF間で負荷を分散することができます。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iSCI 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。	
autoExportPolicy	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。	false
autoExportCIDRs	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。	「[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 エンコード値。証明書ベースの認証に使用されます	""
clientPrivateKey	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	""

パラメータ	製品説明	例
`trustedCACertificate`	信頼された CA 証明書の Base64 エンコード値。オプション。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`	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。	
`svm`	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。
`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`
`limitAggregateUsage`	* Amazon FSx for NetApp ONTAP には指定しないでください。*指定されたと `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`	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、`qtreesPerFlexvol`FlexVolあたりのqtreeの最大数をカスタマイズできます。	“”（デフォルトでは適用されません）
`lunsPerFlexvol`	FlexVol あたりの最大LUN数。有効な範囲は50、200です。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} トラブルシューティングを行って詳細なログダンプが必要な場合以外は使用しない `debugTraceFlags`でください。	null

パラメータ	製品説明	例
nfsMountOptions	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。	""
nasType	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 nfs、、 smb`またはnullです。* SMBボリュームの場合はに設定する必要があります `smb。*null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。	nfs
qtreesPerFlexvol	FlexVol あたりの最大 qtree 数。有効な範囲は [50 、 300] です。	"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。	smb-share
useREST	ONTAP REST API を使用するためのブーリアンパラメータ。技術プレビュー useREST は技術プレビューとして提供されており、本番環境のワークロードには推奨されません。に設定する true`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 `ontap 。これは、事前に定義された役割と役割によって実現され vsadmin cluster-admin ます。	false

パラメータ	製品説明	例
`aws`	AWS FSx for ONTAPの構成ファイルでは次のように指定できます。- ：AWS FSxファイルシステムのIDを指定します。 `fsxFilesystemID`- `apiRegion` ：AWS APIリージョン名。- `apikey`：AWS APIキー。- `secretKey`：AWSシークレットキー。	"" "" ""
`credentials`	AWS Secret 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 "AWS Secrets Managerシークレットの作成" 。	

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

設定のセクションで、これらのオプションを使用してデフォルトのプロビジョニングを制御できます `defaults`。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	製品説明	デフォルト
`spaceAllocation`	space-allocation for LUN のコマンドを指定します	`true`
`spaceReserve`	スペースリザベーションモード：「 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ティ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。	「」

パラメータ	製品説明	デフォルト
adaptiveQosPolicy	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。	「」
snapshotReserve	スナップショット "0" 用に予約されたボリュームの割合	がの none`場合 `snapshotPolicy else
splitOnClone	作成時にクローンを親からスプリットします	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"TridentとNVEおよびNAEとの連携"。	false
luksEncryption	LUKS暗号化を有効にします。を参照してください "Linux Unified Key Setup（LUKS；統合キーセットアップ）を使用"。SANのみ。	""
tieringPolicy	使用する階層化ポリシー none	`snapshot-only`ONTAP 9 .5より前のSVM-DR構成
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。*	「」
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`と `unix`セキュリティ形式SMBのサポート `mixed`と `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ボリュームをプロビジョニングする準備をします

ドライバを使用してSMBボリュームをプロビジョニングできます ontap-nas。完了する前に、次の手順を実行してONTAP SANとNASドライバの統合ください。

開始する前に

ドライバを使用してSMBボリュームをプロビジョニングする `ontap-nas`には、次の準備が必要です。

- Linuxコントローラノードと少なくとも1つのWindowsワーカーノードでWindows Server 2019を実行しているKubernetesクラスタ。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ボリュームのみがサポートされます。
- Active Directoryクレデンシャルを含む少なくとも1つのTridentシークレット。シークレットを生成するには `smbcreds`：

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

- Windowsサービスとして設定されたCSIプロキシ。を設定するには `csi-proxy`、Windowsで実行されているKubernetesノードについて、またはを"GitHub: Windows向けCSIプロキシ"参照してください"GitHub: CSIプロキシ"。

手順

1. SMB共有を作成SMB管理共有は、共有フォルダスナップインを使用するか、ONTAP CLIを使用して作成できます"Microsoft管理コンソール"。ONTAP CLIを使用してSMB共有を作成するには、次の手順を実行します

   a. 必要に応じて、共有のディレクトリパス構造を作成します。

      コマンドは `vserver cifs share create`、共有の作成時に-pathオプションで指定されたパスをチェックします。指定したパスが存在しない場合、コマンドは失敗します。

   b. 指定したSVMに関連付けられているSMB共有を作成します。

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

   c. 共有が作成されたことを確認します。

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

      詳細については、を参照して"SMB共有を作成する"ください。

2. バックエンドを作成する際に、SMBボリュームを指定するように次の項目を設定する必要があります。FSx for ONTAPのバックエンド構成オプションについては、を参照してください"FSX（ONTAP の構成オプションと例）"。

パラメータ	製品説明	例
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。	smb-share
nasType	*に設定する必要があります smb。*nullの場合、デフォルトはになります nfs。	smb
securityStyle	新しいボリュームのセキュリティ形式。* SMBボリュームの場合はまたは mixed`に設定する必要があります `ntfs。*	ntfs`SMBボリュームの場合はまたは `mixed
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く必要があります。*	""

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PersistentVolume（PV）と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

では、 "Kubernetes StorageClassオブジェクト"そのクラスで使用されるプロビジョニングツールとしてTridentが指定され、ボリュームのプロビジョニング方法がTridentに指示されます。例：

```
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"
```

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、を使用して `kubectl`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
```

PVおよびPVCの作成

"永続ボリューム_"（PV）は、Kubernetesクラスタ上のクラスタ管理者によってプロビジョニングされる物理ストレージリソースです。 "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。

PVとPVCを作成したら、ポッドにボリュームをマウントできます。

マニフェストの例

PersistentVolumeサンプルマニフェスト

このサンプルマニフェストは、StorageClassに関連付けられた10Giの基本PVを示しています `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: basic-csi
```

NVMe / TCP対応**PVC**

この例は、という名前のStorageClassに関連付けられたNVMe/TCPの基本的なPVCとRWOアクセスを示しています `protection-gold`。

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

PVおよび**PVC**の作成

手順

1. PVを作成

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

2. PVステータスを確認します。

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

3. PVCを作成

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

4. 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	タイプ	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

属性	タイプ	値	提供	リクエスト	でサポートされます
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	ONTAPNAS、ONTAPNASエコノミー、ONTAP-NAS-flexgroup、ONTAPSAN、solidfire-san-SAN、solidfire-san-SAN、GCP-cvs、azure-NetApp-files、ONTAP-SAN-bエコノミー	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
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-SAN, solidfire-san-, gcvs
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-SAN, solidfire-san-, gcvs
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

サンプルアプリケーションをデプロイします。

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
```

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

2. ボリュームがに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。

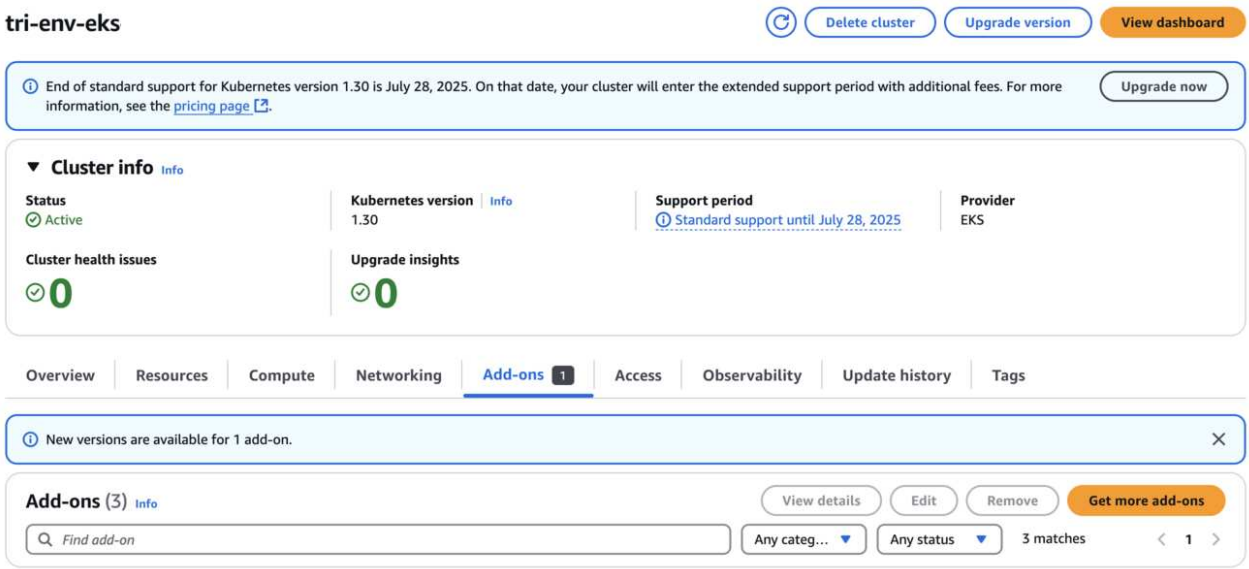
前提条件

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。

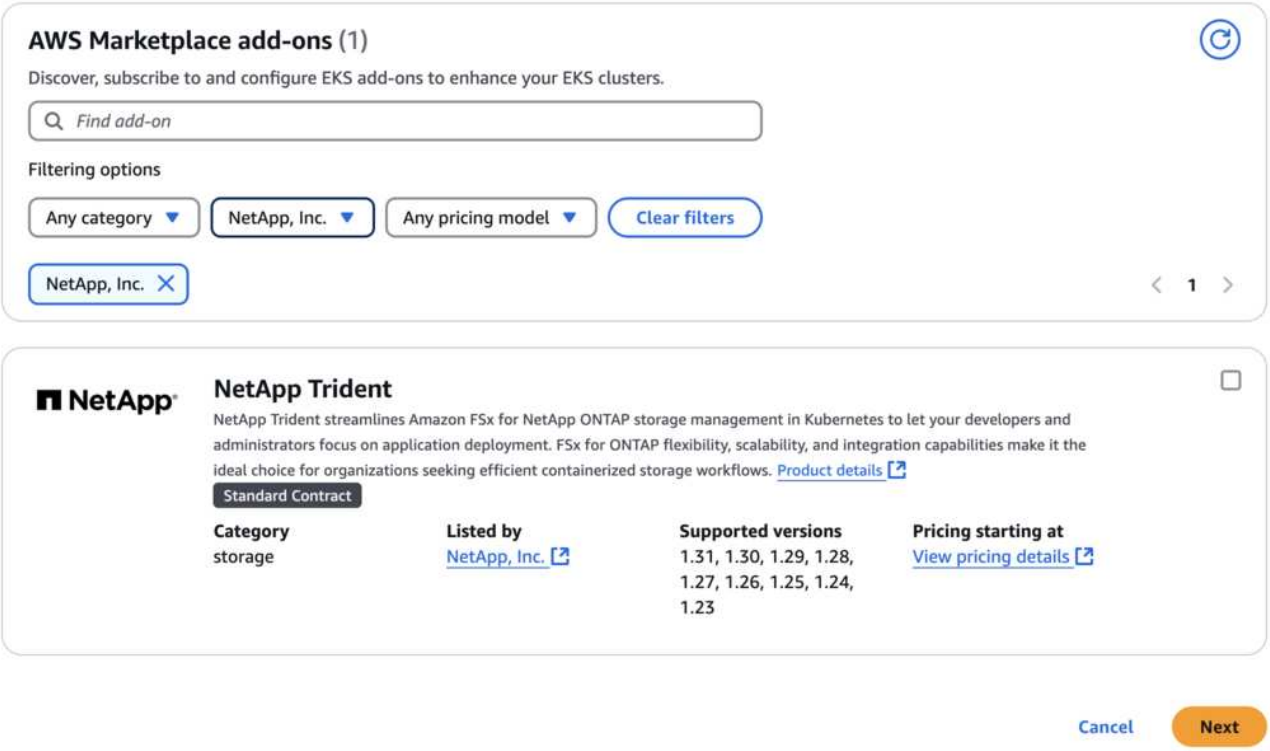
- アドオンを使用する権限を持つAmazon EKSクラスタアカウント。を参照してください "Amazon EKSアドオン"。
- AWS MarketplaceへのAWS権限：
  `"aws-marketplace:ViewSubscriptions",`
  `"aws-marketplace:Subscribe",`
  `"aws-marketplace:Unsubscribe`
- AMIタイプ：Amazon Linux 2（AL2_x86_64）またはAmazon Linux 2 ARM（AL2_Linux_64 ARM）
- ノードタイプ：AMDまたはARM
- 既存のAmazon FSx for NetApp ONTAPファイルシステム

手順

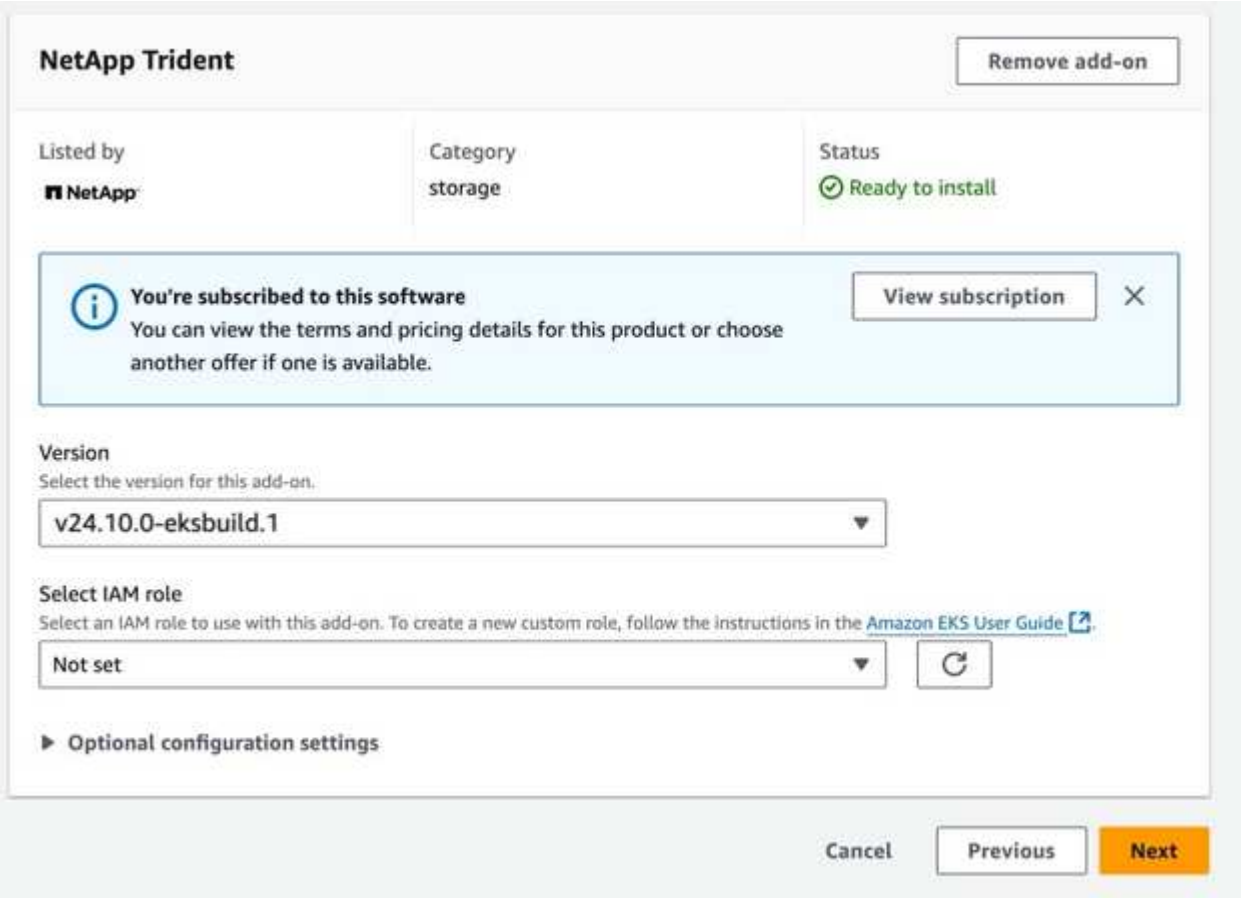
1. EKSポッドがAWSリソースにアクセスできるようにするために、IAMロールとAWSシークレットを作成してください。手順については、を参照してください"IAMロールとAWS Secretを作成する"。
2. EKS Kubernetesクラスタで、*[アドオン]*タブに移動します。



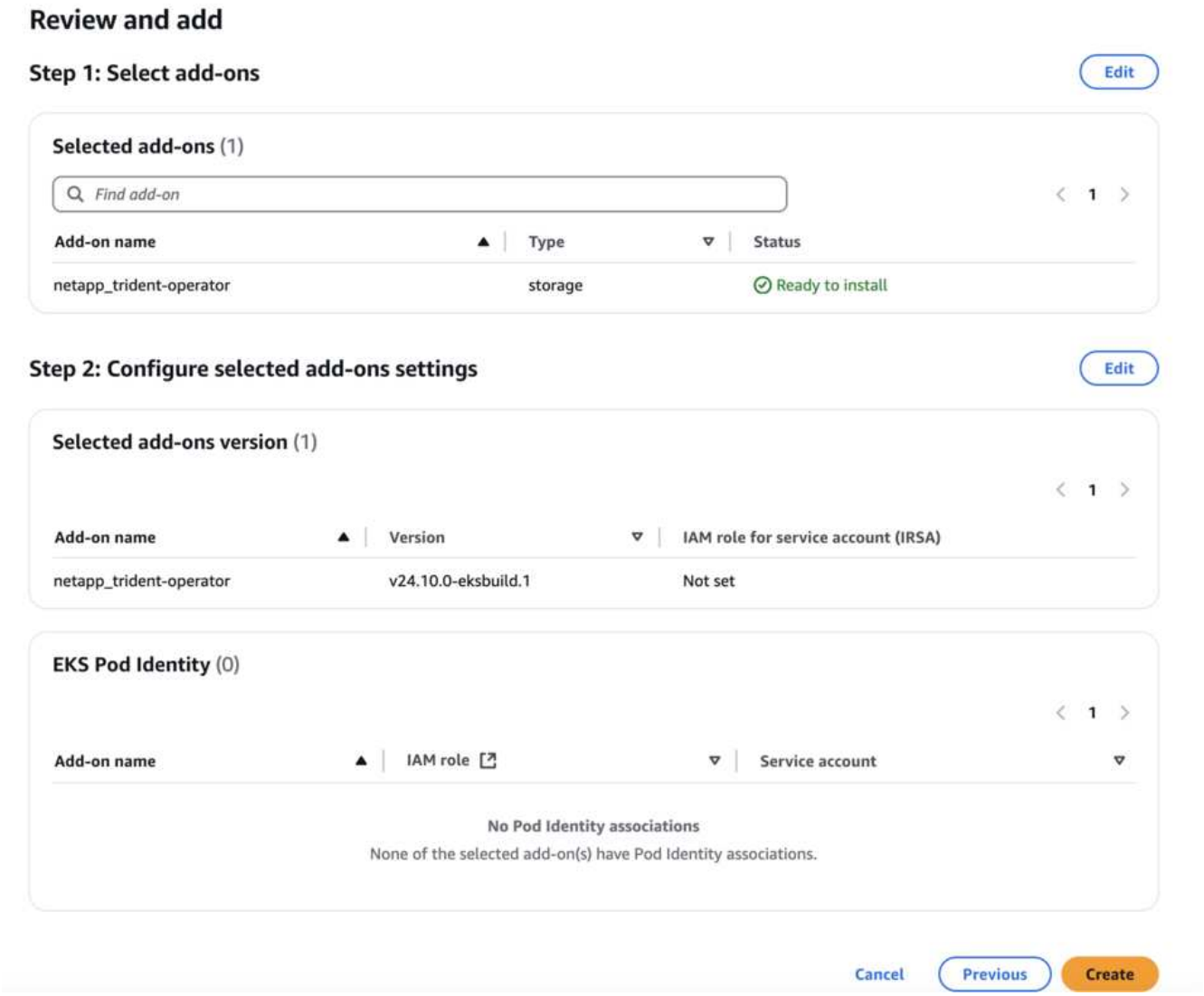
3. [AWS Marketplace add-ons]*にアクセスし、_storage_categoryを選択します。



4. 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。
5. 必要なアドオンのバージョンを選択します。



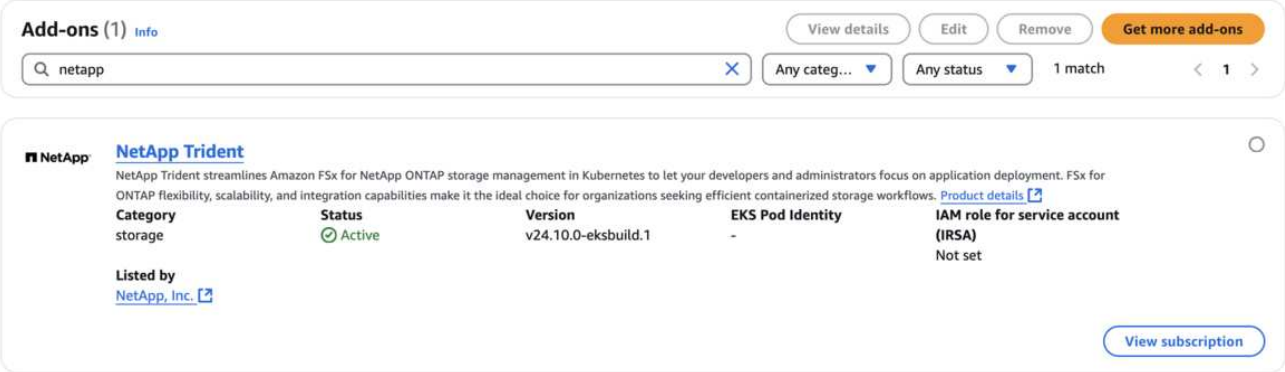
6. ノードから継承するIAMロールオプションを選択します。



7. 必要に応じてオプションの設定を行い、* Next *を選択します。

   Add-on構成スキーマ*に従って、* Configuration Values *セクションのConfiguration Valuesパラメーターを、前の手順（手順1）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）。`eks.amazonaws.com/role-arn: arn:aws:iam::464262061435:role/AmazonEKS_FSXN_CSI_DriverRole`注:[競合解決方法]で[上書き]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。

8. 「 * Create * 」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。

次の例では、Trident EKSアドオンをインストールします。

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

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 © 2025 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権を有し、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権については、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。