Amazon FSx for NetApp ONTAP

Trident

NetApp
November 02, 2026

This PDF was generated from https://docs.netapp.com/ko-kr/trident-2602/trident-use/trident-fsx.html on November 02, 2026. Always check docs.netapp.com for the latest.

목차

Amazon FSx for NetApp ONTAP . . . 1
- Amazon FSx for NetApp ONTAP와 함께 Trident 사용 . . . 1
  - 요구 사항 . . . 1
  - 고려 사항 . . . 1
  - 인증 . . . 2
  - 테스트된 Amazon Machine Images(AMI) . . . 2
  - 자세한 정보 찾기 . . . 3
- IAM 역할 및 AWS Secret을 생성합니다 . . . 3
  - AWS Secrets Manager 시크릿 생성 . . . 3
  - IAM 정책 생성 . . . 4
- Trident 설치 . . . 8
  - Helm을 통해 Trident 설치 . . . 8
  - EKS 애드온을 통해 Trident를 설치하십시오 . . . 10
- 스토리지 클래스 구성 . . . 14
  - StorageClass 객체를 생성합니다 . . . 14
  - SMB 볼륨 프로비저닝 . . . 22
  - 백엔드 고급 구성 및 예 . . . 22
  - 볼륨 프로비저닝을 위한 백엔드 구성 옵션 . . . 26
- PVC 구성 . . . 28
  - PVC를 생성합니다 . . . 28
- 애플리케이션 배포 . . . 29
- 샘플 애플리케이션 배포 . . . 29
- EKS 클러스터에 Trident EKS 애드온을 구성합니다 . . . 30
  - 필수 구성 요소 . . . 31
  - 단계 . . . 31
  - CLI를 사용하여 Trident EKS 애드온 설치/제거 . . . 34

Amazon FSx for NetApp ONTAP

Amazon FSx for NetApp ONTAP와 함께 Trident 사용

"Amazon FSx for NetApp ONTAP"는 NetApp ONTAP 스토리지 운영 체제를 기반으로 하는 파일 시스템을 실행하는 완전 관리형 AWS 서비스입니다. AWS의 확장성과 운영 간편성과 함께 ONTAP의 기능, 성능 및 관리를 제공합니다. 파일 시스템은 Amazon FSx의 기본 리소스이며 온프레미스 ONTAP 클러스터와 유사합니다. 각 파일 시스템에는 하나 이상의 스토리지 가상 머신(SVM)이 포함되고, 각 SVM에는 파일과 디렉터리를 저장하는 하나 이상의 볼륨이 포함됩니다. 이 통합을 통해 Amazon Elastic Kubernetes Service(EKS)에서 실행되는 Kubernetes 클러스터는 블록 및 파일 워크로드를 위해 ONTAP 기반 영구 볼륨을 프로비저닝할 수 있습니다.

요구 사항

"Trident 요구 사항" 외에도 FSx for ONTAP을 Trident와 통합하려면 다음이 필요합니다.

- `kubectl`가 설치된 기존 Amazon EKS 클러스터 또는 자체 관리형 Kubernetes 클러스터.
- 클러스터의 워커 노드에서 연결할 수 있는 기존 Amazon FSx for NetApp ONTAP 파일 시스템 및 스토리지 가상 머신(SVM).
- "NFS 또는 iSCSI"에 대해 준비된 작업자 노드.

> EKS AMI 유형에 따라 Amazon Linux 및 Ubuntu "Amazon Machine Images"(AMI)에 필요한 노드 준비 단계를 반드시 따르십시오.

고려 사항

- SMB 볼륨:
  - SMB 볼륨은 `ontap-nas` 드라이버를 통해서만 지원됩니다.
  - SMB 볼륨은 Trident EKS 애드온에서 지원되지 않습니다.
  - Trident는 Windows 노드에서 실행되는 Pod에 마운트된 SMB 볼륨만 지원합니다. 자세한 내용은 "SMB 볼륨 프로비저닝 준비"를 참조하십시오.
- Trident 24.02 이전 버전에서는 자동 백업이 활성화된 Amazon FSx 파일 시스템에 생성된 볼륨을 Trident에서 삭제할 수 없었습니다. Trident 24.02 이상 버전에서 이 문제를 방지하려면 Amazon FSx for NetApp ONTAP의 백엔드 구성 파일에서 `fsxFilesystemID`, AWS `apiRegion`, AWS `apikey`, AWS `secretKey`를 지정하십시오.

> Trident에 IAM 역할을 지정하는 경우 `apiRegion`, `apiKey` 및 `secretKey` 필드를 Trident에 명시적으로 지정하지 않아도 됩니다. 자세한 내용은 "FSx for ONTAP 구성 옵션 및 예"을 참조하십시오.

Trident SAN/iSCSI 및 EBS-CSI 드라이버의 동시 사용

AWS(EKS, ROSA, EC2 또는 기타 인스턴스)에서 ontap-san 드라이버(예: iSCSI)를 사용하려는 경우 노드에 필요한

다중 경로 구성이 Amazon Elastic Block Store(EBS) CSI 드라이버와 충돌할 수 있습니다. 동일한 노드의 EBS 디스크와 충돌 없이 다중 경로가 작동하도록 하려면 다중 경로 설정에서 EBS를 제외해야 합니다. 다음 예는 다중 경로에서 EBS 디스크를 제외하면서 필요한 Trident 설정을 포함하는 `multipath.conf` 파일입니다.

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

인증

Trident는 두 가지 인증 방식을 제공합니다.

- 자격 증명 기반(권장): 자격 증명을 AWS Secrets Manager에 안전하게 저장합니다. `fsxadmin` 사용자를 파일 시스템에 사용하거나 `vsadmin` 사용자를 SVM에 구성하여 사용할 수 있습니다.

> Trident는 `vsadmin` SVM 사용자 또는 동일한 역할을 가진 다른 이름의 사용자로 실행되어야 합니다. Amazon FSx for NetApp ONTAP에는 ONTAP `admin` 클러스터 사용자의 제한된 대체 역할을 하는 `fsxadmin` 사용자가 있습니다. Trident와 함께 `vsadmin` 사용을 강력히 권장합니다.

- 인증서 기반: Trident는 SVM에 설치된 인증서를 사용하여 FSx 파일 시스템의 SVM과 통신합니다.

인증 활성화에 대한 자세한 내용은 드라이버 유형에 대한 인증을 참조하십시오.

- "ONTAP NAS 인증"
- "ONTAP SAN 인증"

테스트된 **Amazon Machine Images(AMI)**

EKS 클러스터는 다양한 운영 체제를 지원하지만, AWS는 컨테이너 및 EKS에 최적화된 특정 Amazon Machine Image(AMI)를 제공합니다. 다음 AMI는 NetApp Trident 25.02에서 테스트되었습니다.

AMI	NAS	NAS-이코노미	iSCSI	iSCSI-경제성
AL2023_x86_64_STANDARD	예	예	예	예
AL2_x86_64	예	예	예*	예*
BOTTLEROCKET_x86_64	예**	예	해당 없음	해당 없음
AL2023_ARM_64_STANDARD	예	예	예	예

AL2_ARM_64	예	예	예*	예*
BOTTLEROCKET_ARM_64	예**	예	해당 없음	해당 없음

- * 노드를 재시작하지 않고는 PV를 삭제할 수 없습니다
- ** Trident 버전 25.02에서는 NFSv3와 호환되지 않습니다.

원하는 AMI가 이 목록에 없다고 해서 지원되지 않는다는 의미는 아닙니다. 단지 테스트가 완료되지 않았다는 뜻입니다. 이 목록은 정상 작동하는 것으로 확인된 AMI를 안내하기 위한 것입니다.

다음을 사용하여 수행된 테스트:

- EKS 버전: 1.32
- 설치 방법: Helm 25.06 및 AWS 추가 기능 25.06
- NAS의 경우 NFSv3과 NFSv4.1이 모두 테스트되었습니다.
- SAN의 경우 iSCSI만 테스트되었으며 NVMe-oF는 테스트되지 않았습니다.

수행된 테스트:

- 생성: Storage Class, pvc, pod
- 삭제: pod, pvc(일반, qtree/lun – economy, AWS 백업이 있는 NAS)

자세한 정보 찾기

- "Amazon FSx for NetApp ONTAP 설명서"
- "Amazon FSx for NetApp ONTAP에 대한 블로그 게시물"

IAM 역할 및 AWS Secret을 생성합니다

명시적인 AWS 자격 증명을 제공하는 대신 AWS IAM 역할로 인증하여 Kubernetes Pod가 AWS 리소스에 액세스하도록 구성할 수 있습니다.

AWS IAM 역할을 사용하여 인증하려면 EKS를 사용하여 배포된 Kubernetes 클러스터가 있어야 합니다.

AWS Secrets Manager 시크릿 생성

Trident는 스토리지 관리를 위해 FSx vserver에 대한 API를 발행하므로 자격 증명이 필요합니다. 이러한 자격 증명을 안전하게 전달하는 방법은 AWS Secrets Manager 시크릿을 사용하는 것입니다. 따라서 아직 AWS Secrets Manager 시크릿이 없다면 vsadmin 계정의 자격 증명이 포함된 시크릿을 생성해야 합니다.

이 예제는 Trident CSI 자격 증명을 저장하기 위한 AWS Secrets Manager 시크릿을 생성합니다.

```
aws secretsmanager create-secret --name trident-secret --description
```

```
"Trident CSI credentials"\
  --secret-string
"{\"username\":\"vsadmin\",\"password\":\"<svmpassword>\"}"
```

IAM 정책 생성

Trident가 제대로 실행되려면 AWS 권한도 필요합니다. 따라서 Trident에 필요한 권한을 부여하는 정책을 생성해야 합니다.

다음 예제는 AWS CLI를 사용하여 IAM 정책을 생성합니다.

```
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"
```

정책 **JSON** 예:

```
{
  "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"
}
```

서비스 계정 연결(**IRSA**)을 위한 **Pod Identity** 또는 **IAM** 역할을 생성합니다**.**

Kubernetes 서비스 계정이 EKS Pod Identity 또는 IRSA(서비스 계정 연결용 IAM 역할)를 사용하는 AWS Identity and Access Management(IAM) 역할을 수임하도록 구성할 수 있습니다. 해당 서비스 계정을 사용하도록 구성된 모든 Pod는 해당 역할에 액세스 권한이 있는 모든 AWS 서비스에 액세스할 수 있습니다.

Pod Identity

Amazon EKS Pod Identity 연결은 Amazon EC2 인스턴스 프로필이 Amazon EC2 인스턴스에 자격 증명을 제공하는 방식과 유사하게 애플리케이션의 자격 증명을 관리하는 기능을 제공합니다.

EKS 클러스터에 **Pod Identity** 설치:

AWS 콘솔을 사용하거나 다음 AWS CLI 명령을 사용하여 Pod ID를 생성할 수 있습니다.

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

자세한 내용은 "Amazon EKS Pod Identity Agent를 설정합니다"을(를) 참조하십시오.

trust-relationship.json 생성:

EKS 서비스 주체가 Pod Identity에 대한 이 역할을 수임할 수 있도록 trust-relationship.json을 생성합니다. 그런 다음 이 신뢰 정책을 사용하여 역할을 생성합니다.

```
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 파일:

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

```
}
```

IAM 역할에 역할 정책 연결:

이전 단계의 역할 정책을 생성된 IAM 역할에 연결합니다.

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

포드 **ID** 연결 생성:

IAM 역할과 Trident 서비스 계정(trident-controller) 간에 Pod ID 연결을 생성합니다.

```
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
```

서비스 계정 연결을 위한 **IAM** 역할**(IRSA)**

AWS CLI 사용:

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

trust-relationship.json 파일**:**

```
{
  "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` 파일에서 다음 값을 업데이트합니다.

- **<account_id>** - AWS 계정 ID
- **<oidc_provider>** - EKS 클러스터의 OIDC입니다. 다음 명령을 실행하여 oidc_provider를 얻을 수 있습니다:

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

IAM 정책을 사용하여 **IAM** 역할을 연결합니다:

역할이 생성되면 이 명령을 사용하여 (위 단계에서 생성한) 정책을 역할에 연결합니다.

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

OICD 공급자가 연결되어 있는지 확인:

OIDC 공급자가 클러스터에 연결되어 있는지 확인하십시오. 다음 명령을 사용하여 확인할 수 있습니다.

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

출력이 비어 있는 경우 다음 명령을 사용하여 IAM OIDC를 클러스터에 연결하십시오.

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

eksctl을 사용하는 경우 다음 예제를 사용하여 EKS에서 서비스 계정에 대한 IAM 역할을 생성하십시오.

```
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 설치

Trident는 Amazon FSx for NetApp ONTAP 스토리지 관리를 Kubernetes에서 간소화하여 개발자와 관리자가 애플리케이션 배포에 집중할 수 있도록 지원합니다. 다음 방법 중 하나를 사용하여 Trident를 설치할 수 있습니다.

- Helm
- EKS 추가 기능

스냅샷 기능을 사용하려면 CSI 스냅샷 컨트롤러 추가 기능을 설치하십시오. 자세한 내용은 "CSI 볼륨에 대한 스냅샷 기능 활성화"을 참조하십시오.

Helm을 통해 Trident 설치

Pod Identity

1. Trident Helm 리포지토리 추가:

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

2. 다음 예시를 사용하여 Trident를 설치합니다.

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

`helm list` 명령을 사용하여 이름, 네임스페이스, 차트, 상태, 앱 버전 및 개정 번호와 같은 설치 세부 정보를 검토할 수 있습니다.

```
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
```

서비스 계정 연결(**IRSA**)

1. Trident Helm 리포지토리 추가:

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

2. 클라우드 공급자 및 클라우드 **ID** 값을 설정합니다.

```
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` 명령을 사용하여 이름, 네임스페이스, 차트, 상태, 앱 버전 및 개정 번호와 같은 설치 세부 정보를 검토할 수 있습니다.

```
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
```

iSCSI를 사용하려는 경우 클라이언트 시스템에서 iSCSI가 활성화되어 있는지 확인하십시오. AL2023 Worker 노드 OS를 사용하는 경우 helm 설치에 node prep 매개 변수를 추가하여 iSCSI 클라이언트 설치를 자동화할 수 있습니다.

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

EKS 애드온을 통해 Trident를 설치하십시오

Trident EKS 애드온에는 최신 보안 패치와 버그 수정 사항이 포함되어 있으며, AWS에서 Amazon EKS와의 호환성을 검증받았습니다. EKS 애드온을 사용하면 Amazon EKS 클러스터의 보안과 안정성을 지속적으로 유지할 수 있으며, 애드온 설치, 구성 및 업데이트에 필요한 작업량을 줄일 수 있습니다.

필수 구성 요소

AWS EKS용 Trident 애드온을 구성하기 전에 다음 사항을 확인하십시오.

- 추가 기능 구독이 포함된 Amazon EKS 클러스터 계정
- AWS Marketplace에 대한 AWS 권한:
  `"aws-marketplace:ViewSubscriptions",`
  `"aws-marketplace:Subscribe",`
  `"aws-marketplace:Unsubscribe`
- AMI 유형: Amazon Linux 2(AL2_x86_64) 또는 Amazon Linux 2 Arm(AL2_ARM_64)
- 노드 유형: AMD 또는 ARM
- 기존 Amazon FSx for NetApp ONTAP 파일 시스템

AWS용 Trident 추가 기능을 활성화하세요

관리 콘솔

1. Amazon EKS 콘솔을 엽니다 https://console.aws.amazon.com/eks/home#/clusters.
2. 왼쪽 탐색 창에서 *클러스터*를 선택합니다.
3. NetApp Trident CSI 추가 기능을 구성할 클러스터의 이름을 선택하십시오.
4. * 추가 기능 * 을 선택한 다음 * 추가 기능 더 보기 * 를 선택합니다.
5. 추가 기능을 선택하려면 다음 단계를 따르십시오.
   a. **AWS Marketplace** 애드온 섹션으로 스크롤하여 검색 상자에 *'"Trident"*를 입력하세요.
   b. Trident by NetApp 상자의 오른쪽 상단에 있는 확인란을 선택하십시오.
   c. * 다음 * 을 선택합니다.
6. 선택한 애드온 구성 설정 페이지에서 다음을 수행하십시오.

   ⓘ **Pod Identity** 연결을 사용하는 경우 이 단계를 건너뛰십시오.

   a. 사용할 *버전*을 선택합니다.
   b. IRSA 인증을 사용하는 경우 선택적 구성 설정에서 사용 가능한 구성 값을 설정했는지 확인하십시오.
      - 사용할 *버전*을 선택합니다.
      - **Add-on configuration schema***를 따르고 *구성 값 섹션의 **configurationValues** 매개 변수를 이전 단계에서 생성한 role-arn으로 설정합니다(값은 다음 형식이어야 합니다).

```
{
```

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

  }
```

+
충돌 해결 방법으로 Override를 선택하면 기존 애드온 소프트웨어의 설정 중 하나 이상이 Amazon EKS 애드온 소프트웨어 설정으로 덮어쓰여질 수 있습니다. 이 옵션을 사용하지 않고 기존 설정과 충돌이 발생하면 작업이 실패합니다. 오류 메시지를 사용하여 충돌 문제를 해결할 수 있습니다. 이 옵션을 선택하기 전에 Amazon EKS 애드온 소프트웨어가 사용자가 직접 관리해야 하는 설정을 관리하지 않는지 확인하십시오.

7. * 다음 * 을 선택합니다.
8. 검토 및 추가 페이지에서 * 생성 * 을 선택합니다.

   애드온 소프트웨어 설치가 완료되면 설치된 애드온 소프트웨어를 확인할 수 있습니다.

AWS CLI

1. `add-on.json` 파일 생성:

Pod Identity의 경우 다음 형식을 사용하십시오:

ⓘ 다음을 사용하세요

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

IRSA 인증의 경우 다음 형식을 사용하십시오:

```
{
  "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>`을 이전 단계에서 생성한 역할의 ARN으로 바꿉니다.

2. Trident EKS 애드온을 설치합니다.

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

eksctl

다음 예시 명령은 Trident EKS 애드온을 설치합니다.

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

Trident EKS 애드온 업데이트

관리 콘솔

1. Amazon EKS 콘솔을 엽니다 https://console.aws.amazon.com/eks/home#/clusters.
2. 왼쪽 탐색 창에서 *클러스터*를 선택합니다.
3. NetApp Trident CSI 애드온을 업데이트할 클러스터의 이름을 선택합니다.
4. 애드온 탭을 선택합니다.
5. *Trident by NetApp*을 선택한 다음 *편집*을 선택합니다.
6. **NetApp**에서 **Trident** 구성 페이지에서 다음을 수행합니다.
   a. 사용할 *버전*을 선택합니다.
   b. *선택적 구성 설정*을 확장하고 필요에 따라 수정하십시오.
   c. *변경 사항 저장*을 선택합니다.

AWS CLI

다음 예에서는 EKS 애드온을 업데이트합니다.

```
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 애드온의 현재 버전을 확인하십시오. `my-cluster`를 클러스터 이름으로 바꾸십시오.

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

  예시 출력:

```
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'"}
```

- 이전 단계 출력의 UPDATE AVAILABLE에 반환된 버전으로 애드온 소프트웨어를 업데이트하십시오.

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

`--force` 옵션을 제거하고 Amazon EKS 애드온 설정이 기존 설정과 충돌하는 경우 Amazon EKS 애드온 업데이트가 실패하며 충돌 해결에 도움이 되는 오류 메시지가 표시됩니다. 이 옵션을 지정하기 전에 Amazon EKS 애드온이 관리해야 하는 설정을 관리하지 않는지 확인하십시오. 이 옵션을 사용하면 해당 설정이 덮어쓰여지기 때문입니다. 이 설정의 다른 옵션에 대한 자세한 내용은 link:https://eksctl.io/usage/addons/["애드온"]을 참조하십시오. Amazon EKS Kubernetes 필드 관리에 대한 자세한 내용은 link:https://docs.aws.amazon.com/eks/latest/userguide/kubernetes-field-management.html["Kubernetes 필드 관리"]을 참조하십시오.

Trident EKS 애드온 제거/삭제

Amazon EKS 애드온을 제거하는 두 가지 옵션이 있습니다.

- 클러스터에 애드온 소프트웨어 유지 – 이 옵션을 선택하면 Amazon EKS에서 설정 관리가 제거됩니다. 또한 Amazon EKS에서 업데이트 알림을 보내거나 업데이트를 시작한 후 Amazon EKS 애드온을 자동으로 업데이트하는 기능도 제거됩니다. 하지만 클러스터에 애드온 소프트웨어는 유지됩니다. 이 옵션을 사용하면 애드온이 Amazon EKS 애드온이 아닌 자체 관리형 설치로 작동합니다. 이 옵션을 사용해도 애드온에 대한 다운타임은 발생하지 않습니다. 애드온을 유지하려면 명령에 `--preserve` 옵션을 포함하십시오.
- 클러스터에서 애드온 소프트웨어를 완전히 제거하기 – NetApp에서는 클러스터에 해당 애드온에 의존하는 리소스가 없는 경우에만 Amazon EKS 애드온을 클러스터에서 제거할 것을 권장합니다. 애드온을 제거하려면 `--preserve` 옵션을 `delete` 명령에서 제거하세요.

애드온 소프트웨어에 IAM 계정이 연결되어 있는 경우 해당 IAM 계정은 제거되지 않습니다.

관리 콘솔

1. Amazon EKS 콘솔을 엽니다 https://console.aws.amazon.com/eks/home#/clusters.
2. 왼쪽 탐색 창에서 *클러스터*를 선택합니다.
3. NetApp Trident CSI 애드온을 제거할 클러스터의 이름을 선택하십시오.

4. **Add-ons** 탭을 선택한 다음 *Trident by NetApp*을 선택합니다.
5. *제거*를 선택합니다.
6. **netapp_trident-operator** 제거 확인 대화 상자에서 다음을 수행합니다.
   a. Amazon EKS에서 애드온 설정을 더 이상 관리하지 않도록 하려면 *클러스터에 유지*를 선택하십시오. 클러스터에 애드온 소프트웨어를 유지하여 애드온의 모든 설정을 직접 관리하려면 이 옵션을 선택하십시오.
   b. *netapp_trident-operator*를 입력합니다.
   c. *제거*를 선택합니다.

AWS CLI

`my-cluster`를 클러스터 이름으로 바꾼 다음 명령을 실행합니다.

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

eksctl

다음 명령을 실행하면 Trident EKS 추가 기능이 제거됩니다.

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

스토리지 클래스 구성

이 "Kubernetes StorageClass 오브젝트" 설정은 프로비저너를 식별하고 프로비저너에게 볼륨을 프로비저닝하는 방법을 지시합니다. 이 섹션에서는 Trident를 프로비저너로 지정하는 Kubernetes StorageClass 객체를 구성하는 방법을 보여줍니다.

StorageClass 객체를 생성합니다

FSx for ONTAP용 StorageClass를 생성하면 Trident가 백엔드 구성을 자동으로 생성합니다.

> 스토리지 백엔드를 수동으로 구성하려면 Trident 백엔드와 스토리지 클래스를 별도로 생성하는 방법에 대한 자동 백엔드 구성 없이 Kubernetes StorageClass 생성 섹션을 참조하십시오.

필수 **StorageClass** 매개변수를 지정하세요

StorageClass를 생성할 때 다음 세 가지 매개 변수를 정의해야 합니다.

매개변수	필수	유형	설명
`fsxFilesystemID`	예	문자열	FSx for NetApp ONTAP 파일 시스템 ID

매개변수	필수	유형	설명
`storageDriverName`	예	문자열	Trident 스토리지 드라이버(예: `ontap-nas` 또는 `ontap-san`)
`credentialsName`	예	문자열	ONTAP 자격 증명용 FSx가 포함된 Kubernetes Secret의 이름

선택적 매개변수 지정

StorageClass를 통해 선택적 백엔드 매개 변수를 전달할 수 있습니다. StorageClass `parameters` 섹션에서 모든 선택적 값을 문자열로 정의하십시오. 백엔드 매개변수의 전체 목록은 다음을 참조하십시오: "FSx for NetApp ONTAP 백엔드 구성".

StorageClass 구성 파일의 예입니다.

다음 예는 자동 백엔드 구성을 트리거하는 StorageClass를 보여줍니다.

YAML

```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

```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 생성

구성 파일을 생성한 후 다음 명령을 실행하여 스토리지 클래스를 생성하십시오.

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

이제 Kubernetes와 Trident 모두에서 **basic-csi** 스토리지 클래스가 표시되어야 하며, Trident가 백엔드에서 풀을 검색했을 것입니다.

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

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

StorageClass를 적용하면 Trident가 백엔드를 자동으로 생성합니다. 그런 다음, 이 StorageClass를 참조하는 PersistentVolumeClaims를 생성할 수 있습니다.

백엔드 구성 상태를 확인합니다

Trident는 백엔드 생성 결과를 StorageClass 어노테이션에 기록합니다.

주석	설명
`trident.netapp.io/configuratorStatus`	구성 결과 (`Success` 또는 `Failure`)
`trident.netapp.io/configuratorMessage`	자세한 상태 또는 오류 메시지
`trident.netapp.io/configuratorName`	내부 구성자 리소스의 이름
`trident.netapp.io/managed`	StorageClass가 Trident에 의해 관리됨을 나타냅니다
`trident.netapp.io/additionalStoragePools`	이 백엔드용으로 생성된 스토리지 풀

상태를 확인하려면 다음을 실행합니다.

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

`trident.netapp.io/configuratorStatus`이(가) `Success`(으)로 설정되어 있는지 확인하십시오. 값이 `Failure`인 경우 오류에 대해 `trident.netapp.io/configuratorMessage`을(를) 검토하십시오.

추가 FSxN 파일 시스템 추가

동일한 StorageClass를 계속 사용하면서 추가 스토리지 용량이 필요한 경우 추가 FSxN 파일 시스템 ID를 추가하십시오.

StorageClass를 편집하고 다음 주석을 추가하세요.

```
metadata:
  annotations:
    trident.netapp.io/additionalFsxNFileSystemID: '["fs-
xxxxxxxxxxxxxxxxxx"]'
```

변경 사항을 적용하면 Trident가 백엔드 구성을 업데이트하고 StorageClass 어노테이션을 업데이트합니다.

운영상의 고려사항 및 제한사항

- 자동 백엔드 구성이 있는 StorageClass를 삭제하면 일반적으로 연결된 Trident 백엔드도 함께 삭제됩니다. 이로 인해 스토리지 연결이 중단되고 실행 중인 워크로드가 중단될 수 있습니다. 관리형 StorageClass를 삭제하기 전에 영향을 검증하십시오.
- 자동 백엔드 구성은 AWS FSx for NetApp ONTAP에서만 지원됩니다.

자동 백엔드 구성 없이 Kubernetes StorageClass 생성

Trident 백엔드와 StorageClass를 별도로 생성하려면 다음 단계를 따르세요.

자동 백엔드 구성 작동 방식 이해

Trident는 StorageClass 정의에서 백엔드 구성을 가져옵니다. StorageClass를 적용하면 Trident는 필수 매개변수를 검증하고 백엔드를 생성한 다음 StorageClass에 상태 주석을 추가합니다.

Trident는 VolumeSnapshotClass를 한 번만 생성합니다. Trident는 이후의 StorageClasses에 대해 동일한 VolumeSnapshotClass를 재사용합니다.

Trident 백엔드를 생성합니다

Trident 백엔드를 생성하려면 JSON 또는 YAML 형식의 구성 파일을 만들어야 합니다. 이 파일에는 사용할 스토리지 유형(NAS 또는 SAN), 파일 시스템, 스토리지를 가져올 SVM, 그리고 인증 방법을 지정해야 합니다. 다음 예는 NAS 기반 스토리지를 정의하고 AWS 시크릿을 사용하여 사용할 SVM에 대한 자격 증명을 저장하는 방법을 보여줍니다.

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"
    }
  }
}
```

FSx for ONTAP 드라이버 세부 정보

다음 드라이버를 사용하여 Trident를 Amazon FSx for NetApp ONTAP과 통합할 수 있습니다.

드라이버 이름	설명
ontap-san	프로비저닝된 각 PV는 자체 Amazon FSx for NetApp ONTAP 볼륨 내의 LUN입니다. 블록 스토리지에 권장됩니다.
ontap-nas	프로비저닝된 각 PV는 전체 Amazon FSx for NetApp ONTAP 볼륨입니다. NFS 및 SMB에 권장됩니다.
ontap-san-economy	프로비저닝된 각 PV는 Amazon FSx for NetApp ONTAP 볼륨당 구성 가능한 LUN 수를 가진 LUN입니다.
ontap-nas-economy	프로비저닝된 각 PV는 qtree이며, Amazon FSx for NetApp ONTAP 볼륨당 구성 가능한 수의 qtree가 할당됩니다.
ontap-nas-flexgroup	프로비저닝된 각 PV는 Amazon FSx for NetApp ONTAP FlexGroup 볼륨입니다.

드라이버 세부 정보는 "NAS 드라이버" 및 "SAN 드라이버"을 참조하십시오.

백엔드 생성

구성 파일을 생성한 후 다음 명령을 실행하여 Trident 백엔드 구성(TBC)을 생성하고 유효성을 검사하십시오.

- YAML 파일에서 Trident 백엔드 구성(TBC)을 생성하고 다음 명령을 실행하십시오.

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

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

- Trident 백엔드 구성(TBC)이 성공적으로 생성되었는지 확인합니다.

```
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
```

다른 구성 옵션에 대한 자세한 내용은 아래 [Backend-advanced-configuration-and-examples] 섹션을 참조하십시오.

자동 백엔드 구성 없이 스토리지 클래스 구성

다음은 Trident 및 FSx for ONTAP에서 사용할 스토리지 클래스 구성의 예입니다.

NFS용 스토리지 클래스

이 예제를 사용하여 NFS를 사용하는 볼륨에 대한 StorageClass를 설정할 수 있습니다(전체 속성 목록은 아래 Trident 속성 섹션을 참조하십시오).

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

iSCSI용 스토리지 클래스

iSCSI를 사용하여 볼륨에 대한 StorageClass를 설정하려면 다음 예를 사용하십시오.

```
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 및 AWS Bottlerocket을 사용하는 스토리지 클래스

AWS Bottlerocket에서 NFSv3 볼륨을 프로비저닝하려면 스토리지 클래스에 필요한 `mountOptions`을(를) 추가하십시오.

```
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 속성

이러한 매개변수는 특정 유형의 볼륨을 프로비저닝하는 데 사용할 Trident 관리 스토리지 풀을 결정합니다.

속성	유형	값	제공	요청	지원 대상:
미디어[1]	문자열	HDD, 하이브리드, SSD	풀에는 이러한 유형의 미디어가 포함되어 있습니다. 하이브리드는 둘 다를 의미합니다	지정된 미디어 유형	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, solidfire-san
provisioningType	문자열	씬, 씩	풀은 이 프로비저닝 방식을 지원합니다	프로비저닝 방법이 지정되었습니다	thick: 모든 ONTAP, thin: 모든 ONTAP 및 SolidFire-SAN
backendType	문자열	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, solidfire-san, azure-netapp-files, ontap-san-economy	풀은 이러한 유형의 백엔드에 속합니다	지정된 백엔드	모든 드라이버
스냅샷	불	참, 거짓	풀은 스냅샷이 있는 볼륨을 지원합니다	스냅샷이 활성화된 볼륨	ontap-nas, ontap-san, solidfire-san
클론	불	참, 거짓	풀은 볼륨 복제를 지원합니다	클론이 활성화된 볼륨	ontap-nas, ontap-san, solidfire-san
암호화	불	참, 거짓	스토리지 풀은 암호화된 볼륨을 지원합니다.	암호화가 설정된 볼륨	ontap-nas, ontap-nas-economy, ontap-nas-flexgroups, ontap-san
IOPS	int	양의 정수	풀은 이 범위의 IOPS를 보장할 수 있습니다	볼륨에서 이러한 IOPS를 보장합니다	SolidFire-SAN

[1]: ONTAP Select 또는 FSx for ONTAP 시스템에서 지원되지 않습니다

스토리지 클래스가 "Kubernetes 및 Trident 객체"와 상호 작용하는 방법 및 Trident가 볼륨을 프로비저닝하는 방식을 제어하는 매개변수에 대한 자세한 내용은 `PersistentVolumeClaim`를 참조하십시오.

StorageClass를 구성한 후 Kubernetes에서 생성할 수 있습니다.

단계

1. 이것은 Kubernetes 객체이므로 `kubectl`을 사용하여 Kubernetes에서 생성합니다.

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

2. 이제 Kubernetes와 Trident 모두에서 **basic-csi** 스토리지 클래스가 표시되어야 하며, Trident가 백엔드에서 풀을 검색했을 것입니다.

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

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

SMB 볼륨 프로비저닝

`ontap-nas` 드라이버를 사용하여 SMB 볼륨을 프로비저닝할 수 있습니다. 하지만 그러려면 다음 단계를 완료해야 합니다: xref:{relative_path}worker-node-prep.html#prepare-to-provision-smb-volumes["SMB 볼륨 프로비저닝 준비"].

백엔드 고급 구성 및 예

백엔드 구성 옵션은 다음 표를 참조하십시오.

매개변수	설명	예
version		항상 1
storageDriverName	스토리지 드라이버의 이름	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, ontap-san-economy
backendName	사용자 지정 이름 또는 스토리지 백엔드	드라이버 이름 + "_" + dataLIF

매개변수	설명	예
managementLIF	클러스터 또는 SVM 관리 LIF의 IP 주소입니다. 정규화된 도메인 이름(FQDN)도 지정할 수 있습니다. Trident를 IPv6 플래그를 사용하여 설치한 경우 IPv6 주소를 사용하도록 설정할 수 있습니다. IPv6 주소는 [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]와 같이 대괄호 안에 정의해야 합니다. 만약 `fsxFilesystemID`를 `aws` 필드에 제공하면 `managementLIF`를 제공할 필요가 없습니다. Trident가 AWS에서 SVM `managementLIF` 정보를 가져오기 때문입니다. 따라서 SVM(예: vsadmin) 아래의 사용자에 대한 자격 증명을 제공해야 하며, 해당 사용자는 vsadmin 역할을 가지고 있어야 합니다.	"10.0.0.1", "[2001:1234:abcd::fefe]"
dataLIF	프로토콜 LIF의 IP 주소입니다. * ONTAP NAS 드라이버*: NetApp dataLIF를 지정하는 것을 권장합니다. 지정하지 않으면 Trident는 SVM에서 dataLIF를 가져옵니다. NFS 마운트 작업에 사용할 정규화된 도메인 이름(FQDN)을 지정하여 여러 dataLIF에 걸쳐 로드 밸런싱을 수행하는 라운드 로빈 DNS를 구성할 수 있습니다. 초기 설정 후 변경할 수 있습니다. * ONTAP SAN 드라이버*: iSCSI의 경우 지정하지 마십시오. Trident는 ONTAP 선택적 LUN 맵을 사용하여 멀티 패스 세션을 설정하는 데 필요한 iSCSI LIF를 검색합니다. dataLIF가 명시적으로 정의된 경우 경고가 생성됩니다. Trident를 IPv6 플래그를 사용하여 설치한 경우 IPv6 주소를 사용하도록 설정할 수 있습니다. IPv6 주소는 [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]와 같이 대괄호 안에 정의해야 합니다.	
autoExportPolicy	자동 엑스포트 정책 생성 및 업데이트를 활성화합니다[Boolean]. autoExportPolicy 및 autoExportCIDRs 옵션을 사용하면 Trident가 내보내기 정책을 자동으로 관리할 수 있습니다.	false

매개변수	설명	예
autoExportCIDRs	autoExportPolicy`이 활성화된 경우 Kubernetes 노드 IP를 필터링하는 데 사용할 CIDR 목록입니다. `autoExportPolicy 및 autoExportCIDRs 옵션을 사용하면 Trident가 내보내기 정책을 자동으로 관리할 수 있습니다.	["0.0.0.0/0", "::/0"]"
labels	볼륨에 적용할 임의의 JSON 형식 레이블 세트	""
clientCertificate	클라이언트 인증서의 Base64 인코딩 값입니다. 인증서 기반 인증에 사용됩니다	""
clientPrivateKey	클라이언트 개인 키를 Base64로 인코딩한 값입니다. 인증서 기반 인증에 사용됩니다	""
trustedCACertificate	신뢰할 수 있는 CA 인증서의 Base64 인코딩 값입니다. 선택 사항입니다. 인증서 기반 인증에 사용됩니다.	""
username	클러스터 또는 SVM에 연결하는 데 사용할 사용자 이름입니다. 자격 증명 기반 인증에 사용됩니다. 예: vsadmin.	
password	클러스터 또는 SVM에 연결하는 데 사용되는 비밀번호입니다. 자격 증명 기반 인증에 사용됩니다.	
svm	사용할 스토리지 가상 머신	SVM 관리 LIF가 지정된 경우 파생됩니다.
storagePrefix	SVM에서 새 볼륨을 프로비저닝할 때 사용되는 접두사입니다. 생성 후에는 수정할 수 없습니다. 이 매개 변수를 업데이트하려면 새 백엔드를 생성해야 합니다.	trident
limitAggregateUsage	**Amazon FSx for NetApp ONTAP**에는 지정하지 마십시오. 제공된 fsxadmin 및 `vsadmin`에는 Trident를 사용하여 애그리게이트 사용량을 검색하고 제한하는 데 필요한 권한이 포함되어 있지 않습니다.	사용하지 마십시오.
limitVolumeSize	요청된 볼륨 크기가 이 값을 초과하면 프로비저닝이 실패합니다. 또한 관리하는 qtree 및 LUN의 최대 크기를 제한하며, qtreesPerFlexvol 옵션을 통해 FlexVol 볼륨당 최대 qtree 수를 사용자 지정할 수 있습니다.	"" (기본적으로 적용되지 않음)

매개변수	설명	예
`lunsPerFlexvol`	FlexVol 볼륨당 최대 LUN 수는 [50, 200] 범위 내에 있어야 합니다. SAN 전용입니다.	“100”
`debugTraceFlags`	문제 해결 시 사용할 디버그 플래그입니다. 예, {"api":false, "method":true} 문제 해결 중이거나 자세한 로그 덤프가 필요한 경우가 아니면 `debugTraceFlags` 사용하지 마십시오.	null
`nfsMountOptions`	쉼표로 구분된 NFS 마운트 옵션 목록입니다. Kubernetes 영구 볼륨의 마운트 옵션은 일반적으로 스토리지 클래스에 지정되지만, 스토리지 클래스에 마운트 옵션이 지정되지 않은 경우 Trident는 스토리지 백엔드의 구성 파일에 지정된 마운트 옵션을 사용합니다. 스토리지 클래스 또는 구성 파일에 마운트 옵션이 지정되지 않은 경우 Trident는 연결된 영구 볼륨에 마운트 옵션을 설정하지 않습니다.	""
`nasType`	NFS 또는 SMB 볼륨 생성을 구성합니다. 옵션은 `nfs`, `smb` 또는 null입니다. **SMB** 볼륨의 경우 **`smb`**로 설정해야 합니다**.** null로 설정하면 기본적으로 NFS 볼륨이 사용됩니다.	`nfs`
`qtreesPerFlexvol`	FlexVol 볼륨당 최대 Qtree 수는 [50, 300] 범위 내에 있어야 합니다.	`"200"`
`smbShare`	다음 중 하나를 지정할 수 있습니다. Microsoft Management Console 또는 ONTAP CLI를 사용하여 생성한 SMB 공유의 이름 또는 Trident가 SMB 공유를 생성할 수 있도록 허용하는 이름입니다. 이 매개 변수는 Amazon FSx for NetApp ONTAP 백엔드에 필수입니다.	`smb-share`

매개변수	설명	예
useREST	ONTAP REST API를 사용하기 위한 부울 매개 변수입니다. `true`로 설정하면 Trident는 ONTAP REST API를 사용하여 백엔드와 통신합니다. 이 기능을 사용하려면 ONTAP 9.11.1 이상이 필요합니다. 또한 사용되는 ONTAP 로그인 역할은 `ontap` 애플리케이션에 대한 액세스 권한이 있어야 합니다. 이는 사전 정의된 vsadmin 및 cluster-admin 역할로 충족됩니다.	false
aws	AWS FSx for ONTAP의 구성 파일에서 다음을 지정할 수 있습니다. - fsxFilesystemID: AWS FSx 파일 시스템의 ID를 지정합니다. - apiRegion: AWS API 리전 이름입니다. - apikey: AWS API 키입니다. - secretKey: AWS 비밀 키입니다.	"" "" ""
credentials	AWS Secrets Manager에 저장할 FSx SVM 자격 증명을 지정합니다. - name: SVM 자격 증명이 포함된 시크릿의 Amazon 리소스 이름(ARN)입니다. - type: `awsarn`로 설정합니다. 자세한 내용은 "AWS Secrets Manager 암호를 생성합니다"을 참조하십시오.	

볼륨 프로비저닝을 위한 백엔드 구성 옵션

구성의 defaults 섹션에서 이러한 옵션을 사용하여 기본 프로비저닝을 제어할 수 있습니다. 예를 들어, 아래 구성 예를 참조하십시오.

매개변수	설명	기본값
spaceAllocation	LUN의 공간 할당	true
spaceReserve	공간 예약 모드, "none"(씬) 또는 "volume"(씩)	none
snapshotPolicy	사용할 스냅샷 정책	none

매개변수	설명	기본값
qosPolicy	생성된 볼륨에 할당할 QoS 정책 그룹입니다. 스토리지 풀 또는 백엔드별로 qosPolicy 또는 adaptiveQosPolicy 중 하나를 선택하십시오. Trident에서 QoS 정책 그룹을 사용하려면 ONTAP 9.8 이상이 필요합니다. 공유되지 않는 QoS 정책 그룹을 사용하고 정책 그룹이 각 구성 요소에 개별적으로 적용되도록 해야 합니다. 공유 QoS 정책 그룹은 모든 워크로드의 총 처리량에 대한 상한선을 적용합니다.	""
adaptiveQosPolicy	생성된 볼륨에 할당할 적응형 QoS 정책 그룹입니다. 스토리지 풀 또는 백엔드별로 qosPolicy 또는 adaptiveQosPolicy 중 하나를 선택하십시오. ontap-nas-economy에서는 지원되지 않습니다.	""
snapshotReserve	스냅샷용으로 예약된 볼륨의 비율 "0"	`snapshotPolicy`이 `none`인 경우, `else` ""
splitOnClone	생성 시 상위 항목에서 클론 분할	false
encryption	새 볼륨에서 NetApp Volume Encryption(NVE)을 활성화합니다. 기본값은 `false`입니다. 이 옵션을 사용하려면 클러스터에서 NVE 라이센스가 있고 활성화되어 있어야 합니다. 백엔드에서 NAE가 활성화된 경우 Trident에서 프로비저닝된 모든 볼륨은 NAE가 활성화됩니다. 자세한 내용은 다음을 참조하십시오. "Trident가 NVE 및 NAE와 작동하는 방식"	false
luksEncryption	LUKS 암호화를 활성화합니다. 다음을 참조하십시오. "Linux Unified Key Setup(LUKS) 사용". SAN 전용입니다.	""
tieringPolicy	사용할 계층화 정책 none	
unixPermissions	새 볼륨 모드입니다. **SMB** 볼륨의 경우 비워 두십시오.	""
securityStyle	새 볼륨에 대한 보안 스타일입니다. NFS는 mixed 및 unix 보안 스타일을 지원합니다. SMB는 mixed 및 ntfs 보안 스타일을 지원합니다.	NFS 기본값은 `unix`입니다. SMB 기본값은 `ntfs`입니다.

PVC 구성

이 섹션에는 구성된 Kubernetes StorageClass를 사용하여 PV를 요청하는 PersistentVolumeClaim(PVC)을 생성하는 방법에 대한 지침이 포함되어 있습니다. 성공하면 PV를 파드에 마운트할 수 있습니다.

PVC를 생성합니다

`https://kubernetes.io/docs/concepts/storage/persistent-volumes["_PersistentVolumeClaim_"^](PVC)`는 클러스터의 `PersistentVolume`에 대한 액세스 요청입니다. `PVC`는 특정 크기의 스토리지 또는 액세스 모드를 요청하도록 구성할 수 있습니다. 연결된 `StorageClass`를 사용하여 클러스터 관리자는 `PersistentVolume` 크기 및 액세스 모드 외에도 성능이나 서비스 수준과 같은 다양한 요소를 제어할 수 있습니다.

Trident 백엔드와 StorageClass를 생성한 후 PVC를 생성할 수 있습니다. PVC가 생성되면 해당 볼륨을 Pod에 마운트할 수 있습니다.

샘플 매니페스트

다음 예시는 기본적인 PVC 구성 옵션을 보여줍니다.

RWX 액세스가 있는 **PVC**

이 예제는 `basic-csi`라는 이름의 StorageClass와 연결된 RWX 액세스 권한이 있는 기본 PVC를 보여줍니다.

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

iSCSI를 사용하는 **PVC** 예

이 예제는 RWO 액세스가 가능한 iSCSI용 기본 PVC를 보여주며, 이는 StorageClass라는 이름의 `protection-gold`와(과) 연결되어 있습니다.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-san
```

```
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: protection-gold
```

PVC 생성

단계

1. PVC를 생성합니다.

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

2. PVC 상태를 확인합니다.

```
kubectl get pvc
```

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

스토리지 클래스가 "Kubernetes 및 Trident 객체"와 상호 작용하는 방법 및 Trident가 볼륨을 프로비저닝하는 방식을 제어하는 매개변수에 대한 자세한 내용은 `PersistentVolumeClaim`를 참조하십시오.

애플리케이션 배포

스토리지 클래스와 PVC가 생성되면 PV를 파드에 마운트할 수 있습니다. 이 섹션에서는 PV를 파드에 연결하는 예제 명령과 구성을 설명합니다.

샘플 애플리케이션 배포

단계

1. POD에 볼륨을 마운트합니다.

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

다음 예시는 PVC를 포드에 연결하는 기본 구성을 보여줍니다. 기본 구성:

```
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
```

`kubectl get pod --watch`를 사용하여 진행 상황을 모니터링할 수 있습니다.

2. 볼륨이 `/my/mount/path`에 마운트되었는지 확인하십시오.

```
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
```

이제 Pod를 삭제할 수 있습니다. Pod 애플리케이션은 더 이상 존재하지 않지만 볼륨은 그대로 유지됩니다.

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

EKS 클러스터에 Trident EKS 애드온을 구성합니다

NetApp Trident는 Kubernetes 환경에서 Amazon FSx for NetApp ONTAP 스토리지 관리를 간소화하여 개발자와 관리자가 애플리케이션 배포에 집중할 수 있도록 지원합니다. NetApp Trident EKS 애드온에는 최신 보안 패치와 버그 수정 사항이 포함되어 있으며, AWS에서 Amazon EKS와의 호환성을 검증받았습니다. EKS 애드온을 사용하면 Amazon EKS 클러스터의 보안과 안정성을 지속적으로 유지할 수 있으며, 애드온 설치, 구성 및 업데이트에 필요한 작업량을 줄일 수 있습니다.

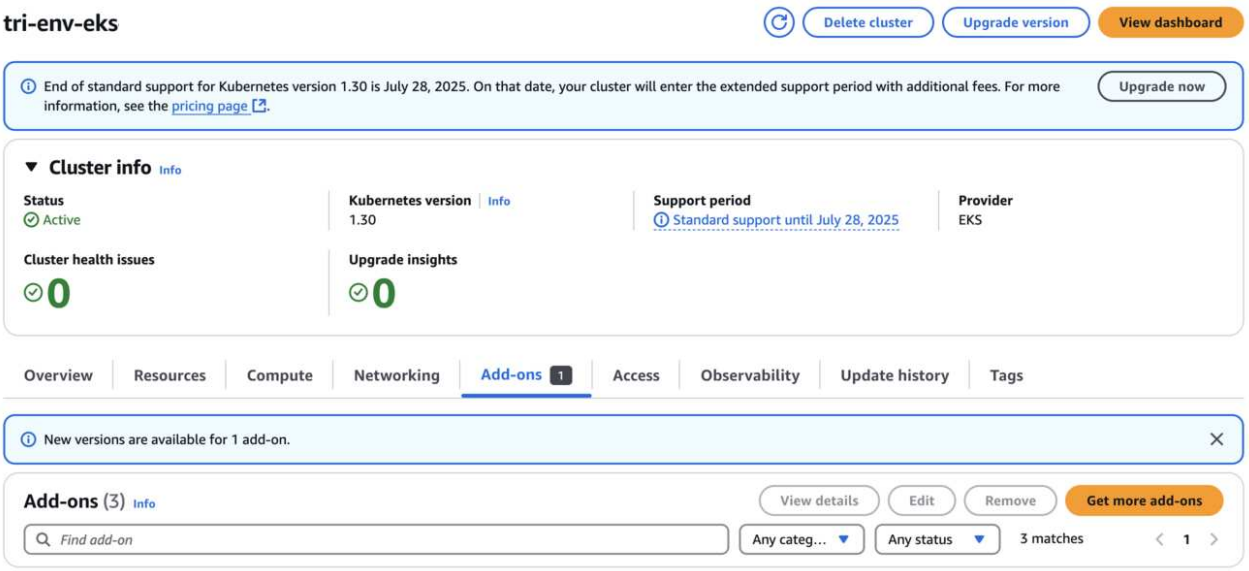
필수 구성 요소

AWS EKS용 Trident 애드온을 구성하기 전에 다음 사항을 확인하십시오.

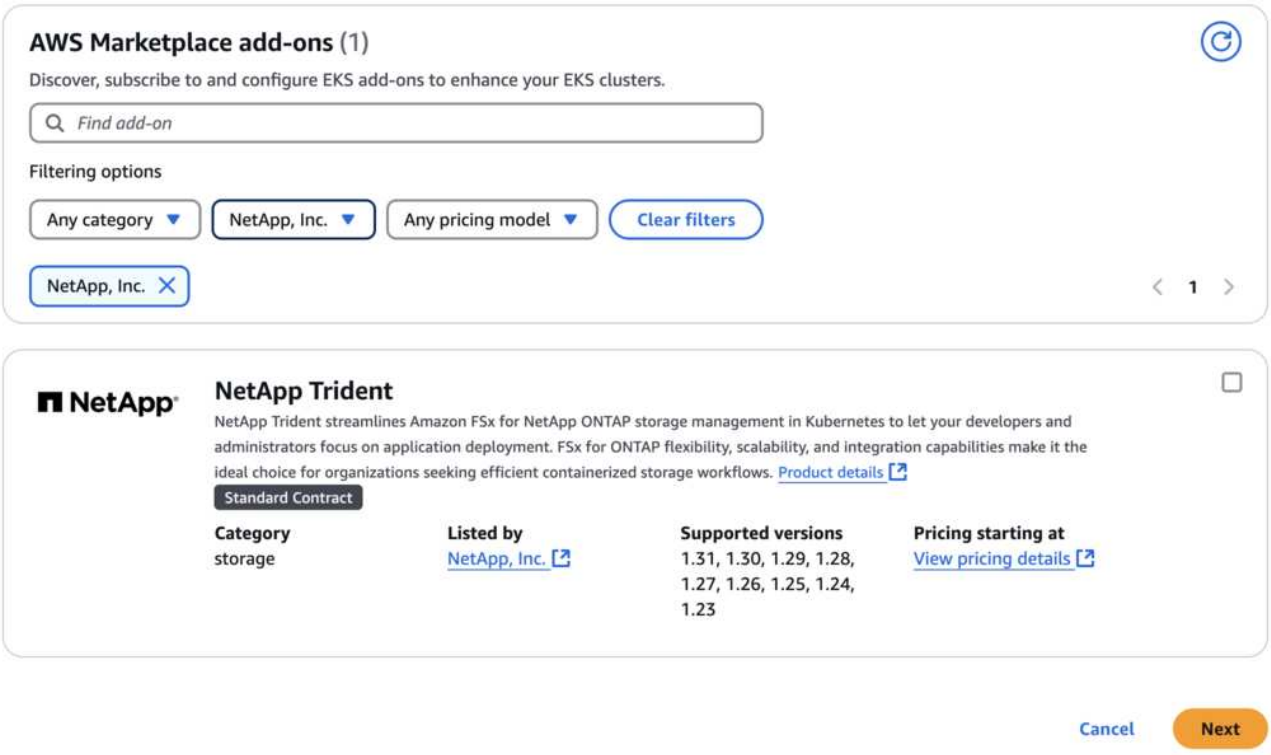
- 추가 기능을 사용할 수 있는 권한이 있는 Amazon EKS 클러스터 계정입니다. 다음을 참조하십시오. "Amazon EKS 추가 기능".
- AWS Marketplace에 대한 AWS 권한:
  ```
  "aws-marketplace:ViewSubscriptions",
  "aws-marketplace:Subscribe",
  "aws-marketplace:Unsubscribe
  ```
- AMI 유형: Amazon Linux 2(AL2_x86_64) 또는 Amazon Linux 2 Arm(AL2_ARM_64)
- 노드 유형: AMD 또는 ARM
- 기존 Amazon FSx for NetApp ONTAP 파일 시스템

단계

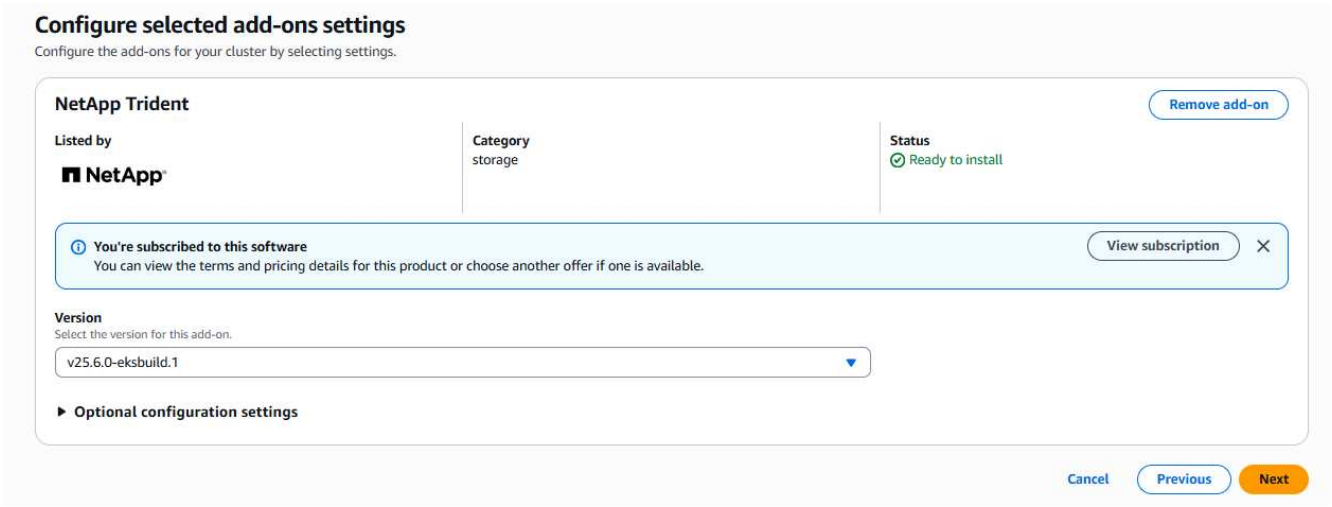
1. EKS Pod가 AWS 리소스에 액세스할 수 있도록 IAM 역할과 AWS 시크릿을 생성해야 합니다. 지침은 "IAM 역할 및 AWS Secret을 생성합니다"을 참조하십시오.
2. EKS Kubernetes 클러스터에서 **Add-ons** 탭으로 이동합니다.



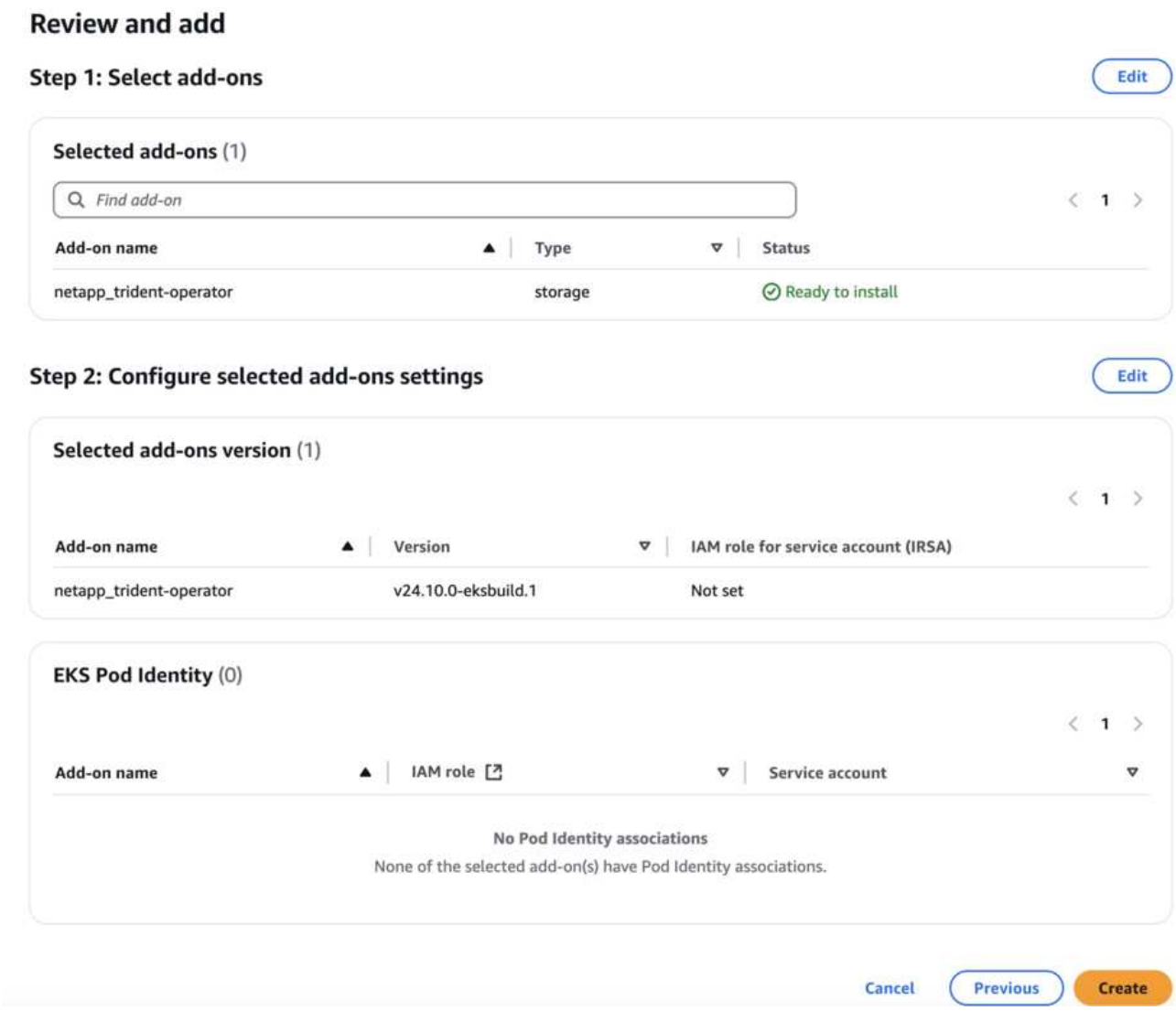
3. *AWS Marketplace add-ons*로 이동하여 *storage* 카테고리를 선택하세요.



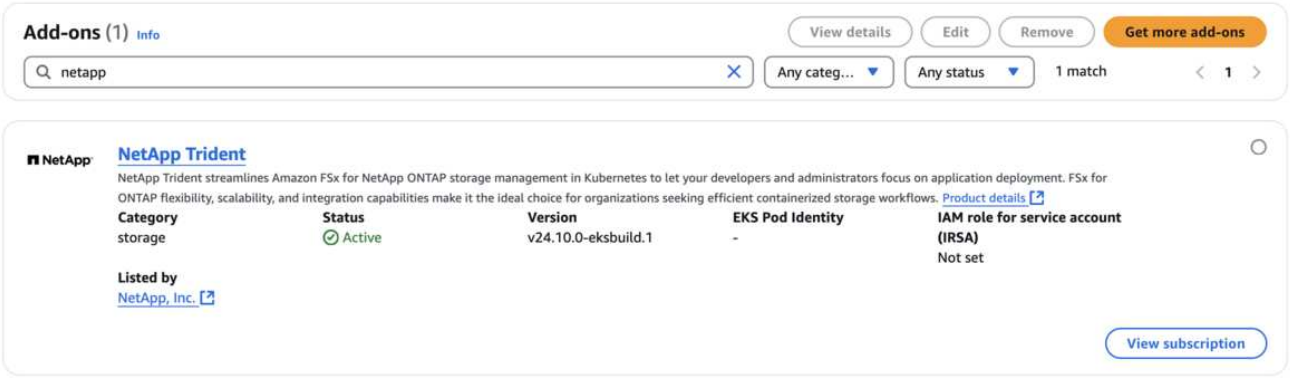
4. *NetApp Trident*를 찾아 Trident 추가 기능 확인란을 선택한 다음 *다음*을 클릭합니다.
5. 원하는 추가 기능 버전을 선택합니다.



6. 필요한 추가 기능 설정을 구성하십시오.



7. IRSA(서비스 계정용 IAM 역할)를 사용하는 경우 추가 구성 단계"여기"를 참조하십시오.
8. *생성*을 선택합니다.
9. 추가 기능의 상태가 _Active_인지 확인하십시오.



10. 다음 명령을 실행하여 Trident가 클러스터에 제대로 설치되었는지 확인하십시오.

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

11. 설정을 계속 진행하여 스토리지 백엔드를 구성하십시오. 자세한 내용은 "스토리지 백엔드 구성"를 참조하십시오.

CLI를 사용하여 Trident EKS 애드온 설치/제거

CLI를 사용하여 **NetApp Trident EKS** 추가 기능을 설치합니다**.**

다음 예시 명령어는 Trident EKS 애드온을 설치합니다.
`eksctl create addon --cluster clusterName --name netapp_trident-operator --version v25.6.0-eksbuild.1`(전용 버전 포함)

다음 예시 명령은 Trident EKS 애드온 버전 25.6.1를 설치합니다:
`eksctl create addon --cluster clusterName --name netapp_trident-operator --version v25.6.1-eksbuild.1` (전용 버전 포함)

다음 예시 명령은 Trident EKS 애드온 버전 25.6.2를 설치합니다:
`eksctl create addon --cluster clusterName --name netapp_trident-operator --version v25.6.2-eksbuild.1` (전용 버전 포함)

CLI를 사용하여 **NetApp Trident EKS** 추가 기능을 제거합니다**.**

다음 명령을 실행하면 Trident EKS 추가 기능이 제거됩니다.

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

저작권 정보

Copyright © 2026 NetApp, Inc. All Rights Reserved. 미국에서 인쇄됨 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이센스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이센스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이센스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

NETAPP, NETAPP 로고 및 http://www.netapp.com/TM에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.