



NetApp Trident 구축 및 구성

NetApp Solutions

NetApp
April 20, 2024

This PDF was generated from https://docs.netapp.com/ko-kr/netapp-solutions/ai/aicp_netapp_trident_deployment_and_configuration_overview.html on April 20, 2024. Always check docs.netapp.com for the latest.

목차

NetApp Trident 구축 및 구성	1
NetApp Trident 구축 및 구성	1
ONTAP AI 배포에 대한 Trident 백엔드 예	1
ONTAP AI 구축을 위한 Kubernetes StorageClasses의 예	4

NetApp Trident 구축 및 구성

NetApp Trident 구축 및 구성

이 섹션에서는 Kubernetes 클러스터에 NetApp Trident를 설치 및 구성하기 위해 완료해야 하는 작업에 대해 설명합니다.

필수 구성 요소

이 섹션에 요약된 배포 연습을 수행하기 전에 이미 다음 작업을 수행했다고 가정합니다.

1. Kubernetes 작업 클러스터가 이미 있으며, Trident에서 지원하는 Kubernetes 버전을 실행 중입니다. 지원되는 버전 목록은 [Trident 문서](#)를 참조하십시오.
2. Trident에서 지원하는 작업 중인 NetApp 스토리지 어플라이언스, 소프트웨어 정의 인스턴스 또는 클라우드 스토리지 서비스가 이미 있습니다.

Trident를 설치합니다

Kubernetes 클러스터에 NetApp Trident를 설치 및 구성하려면 배포 점프 호스트에서 다음 작업을 수행하십시오.

1. 다음 방법 중 하나를 사용하여 Trident를 배포합니다.
 - NVIDIA DeepOps를 사용하여 Kubernetes 클러스터를 구축한 경우, NVIDIA DeepOps를 사용하여 Kubernetes 클러스터에 Trident를 구축할 수도 있습니다. DeepOps를 사용하여 Trident를 배포하려면 [따릅니다 "Trident 배포 지침"](#) NVIDIA DeepOps GitHub 사이트에서 다운로드할 수 있습니다.
 - NVIDIA DeepOps를 사용하여 Kubernetes 클러스터를 배포하지 않았거나 Trident를 수동으로 배포하려는 경우 [에 따라 Trident를 배포할 수 있습니다 "배포 지침"](#) Trident 문서 구성 방법에 대한 자세한 내용은 Trident 백엔드와 하나 이상의 Kubernetes StorageClass를 생성해야 합니다 ["백엔드"](#) 및 ["StorageClaes를 참조하십시오"](#) NetApp Docs에서 연결된 하위 섹션을 참조하십시오.



ONTAP AI Pod에 NetApp AI Control Plane 솔루션을 구축하는 경우 [참조하십시오 "ONTAP AI 배포에 대한 Trident 백엔드 예"](#) 및 [참조하십시오 "ONTAP AI 배포를 위한 Kubernetes StorageClasses의 예"](#) 생성할 수 있는 여러 Kubernetes StorageClasses의 몇 가지 예를 확인하십시오.

ONTAP AI 배포에 대한 Trident 백엔드 예

Trident를 사용하여 Kubernetes 클러스터 내에서 스토리지 리소스를 동적으로 프로비저닝하려면 먼저 하나 이상의 Trident 백엔드를 생성해야 합니다. 다음 예는 ONTAP AI 포드에 NetApp AI Control Plane 솔루션을 구축할 경우 생성할 수 있는 다양한 유형의 백엔드를 보여줍니다. 백엔드에 대한 자세한 내용은 [참조하십시오 "Trident 문서"](#).

1. NetApp AFF 시스템에서 사용할 각 데이터 LIF(데이터 액세스를 제공하는 논리적 네트워크 인터페이스)에 대해 FlexGroup 지원 Trident 백엔드를 생성하는 것이 좋습니다. LIF 간 볼륨 마운트의 균형을 조정할 수 있습니다

다음 명령 예는 동일한 ONTAP 스토리지 가상 시스템(SVM)과 연관된 2개의 서로 다른 데이터 LIF에 대해 2개의 FlexGroup 지원 Trident 백엔드를 생성하는 것을 보여줍니다. 이러한 백엔드는 'ONTAP-NAS-Flexgroup'

스토리지 드라이버를 사용합니다. ONTAP은 FlexVol과 FlexGroup의 두 가지 기본 데이터 볼륨 유형을 지원합니다. FlexVol 볼륨의 크기는 제한되어 있습니다(이 쓰기 작업 시 최대 크기는 특정 구축에 따라 다름). 반면 FlexGroup 볼륨은 최대 20PB 및 4천억 개 파일까지 선형적으로 확장할 수 있으므로 데이터 관리를 크게 간소화하는 단일 네임스페이스를 제공합니다. 따라서 FlexGroup 볼륨은 대량의 데이터를 사용하는 AI 및 ML 워크로드에 최적화되어 있습니다.

소량의 데이터로 작업하고 FlexGroup 볼륨 대신 FlexVol 볼륨을 사용하려는 경우, ONTAP-NAS-Flexgroup 스토리지 드라이버 대신 'ONTAP-NAS' 스토리지 드라이버를 사용하는 Trident 백엔드를 생성할 수 있습니다.

```
$ cat << EOF > ./trident-backend-ontap-ai-flexgroups-iface1.json
{
    "version": 1,
    "storageDriverName": "ontap-nas-flexgroup",
    "backendName": "ontap-ai-flexgroups-iface1",
    "managementLIF": "10.61.218.100",
    "dataLIF": "192.168.11.11",
    "svm": "ontapai_nfs",
    "username": "admin",
    "password": "ontapai"
}
EOF
$ tridentctl create backend -f ./trident-backend-ontap-ai-flexgroups-
iface1.json -n trident
+-----+-----+
+-----+-----+-----+-----+
|          NAME          | STORAGE DRIVER |
+-----+-----+-----+-----+
|          UUID          | STATE  | VOLUMES |
+-----+-----+-----+-----+
| ontap-ai-flexgroups-iface1 | ontap-nas-flexgroup | b74cbddb-e0b8-40b7-
b263-b6da6dec0bdd | online |      0 |
+-----+-----+-----+-----+
$ cat << EOF > ./trident-backend-ontap-ai-flexgroups-iface2.json
{
    "version": 1,
    "storageDriverName": "ontap-nas-flexgroup",
    "backendName": "ontap-ai-flexgroups-iface2",
    "managementLIF": "10.61.218.100",
    "dataLIF": "192.168.12.12",
    "svm": "ontapai_nfs",
    "username": "admin",
    "password": "ontapai"
}
EOF
$ tridentctl create backend -f ./trident-backend-ontap-ai-flexgroups-
iface2.json -n trident
```

```

+-----+-----+
+-----+-----+-----+-----+
|          NAME          |  STORAGE DRIVER  |
UUID                   | STATE | VOLUMES |
+-----+-----+
+-----+-----+-----+-----+
| ontap-ai-flexgroups-iface2 | ontap-nas-flexgroup | 61814d48-c770-436b-9cb4-cf7ee661274d | online | 0 |
+-----+-----+
+-----+-----+-----+-----+
$ tridentctl get backend -n trident
+-----+-----+
+-----+-----+-----+-----+
|          NAME          |  STORAGE DRIVER  |
UUID                   | STATE | VOLUMES |
+-----+-----+
+-----+-----+-----+-----+
| ontap-ai-flexgroups-iface1 | ontap-nas-flexgroup | b74cbddb-e0b8-40b7-b263-b6da6dec0bdd | online | 0 |
| ontap-ai-flexgroups-iface2 | ontap-nas-flexgroup | 61814d48-c770-436b-9cb4-cf7ee661274d | online | 0 |
+-----+-----+
+-----+-----+-----+-----+

```

2. 또한 하나 이상의 FlexVol 지원 Trident 백엔드를 생성하는 것이 좋습니다. 데이터 세트 스토리지를 훈련하는 데 FlexGroup 볼륨을 사용하는 경우 FlexVol 볼륨을 사용하여 결과, 출력, 디버그 정보 등을 저장할 수 있습니다. FlexVol 볼륨을 사용하려면 하나 이상의 FlexVol 지원 Trident 백엔드를 생성해야 합니다. 다음 명령의 예는 단일 데이터 LIF를 사용하는 단일 FlexVol 지원 Trident 백엔드를 생성하는 것입니다.

```
$ cat << EOF > ./trident-backend-ontap-ai-flexvols.json
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "ontap-ai-flexvols",
  "managementLIF": "10.61.218.100",
  "dataLIF": "192.168.11.11",
  "svm": "ontapai_nfs",
  "username": "admin",
  "password": "ontapai"
}
EOF
$ tridentctl create backend -f ./trident-backend-ontap-ai-flexvols.json -n
trident
+-----+-----+-----+
+-----+-----+-----+
|          NAME          | STORAGE DRIVER |          UUID          |
| STATE | VOLUMES | |
+-----+-----+-----+
+-----+-----+-----+
| ontap-ai-flexvols      | ontap-nas      | 52bdb3b1-13a5-4513-   |
a9c1-52a69657fabe | online |      0 |
+-----+-----+-----+
+-----+-----+-----+
$ tridentctl get backend -n trident
+-----+-----+-----+
+-----+-----+-----+
|          NAME          | STORAGE DRIVER |          UUID          |
| STATE | VOLUMES | |
+-----+-----+-----+
+-----+-----+-----+
| ontap-ai-flexvols      | ontap-nas      | 52bdb3b1-13a5-4513-   |
a9c1-52a69657fabe | online |      0 |
| ontap-ai-flexgroups-iface1 | ontap-nas-flexgroup | b74cbddb-e0b8-40b7-   |
b263-b6da6dec0bdd | online |      0 |
| ontap-ai-flexgroups-iface2 | ontap-nas-flexgroup | 61814d48-c770-436b-   |
9cb4-cf7ee661274d | online |      0 |
+-----+-----+-----+
+-----+-----+-----+
```

ONTAP AI 구축을 위한 Kubernetes StorageClasses의 예

Trident를 사용하여 Kubernetes 클러스터 내에서 스토리지 리소스를 동적으로 프로비저닝하려면 먼저 하나 이상의 Kubernetes StorageClasses를 생성해야 합니다. 다음

예제는 ONTAP AI POD에 NetApp AI Control Plane 솔루션을 구축할 경우 생성할 수 있는 다양한 유형의 StorageClasses를 보여줍니다. StorageClasses에 대한 자세한 내용은 를 참조하십시오 ["Trident 문서"](#).

1. NetApp은 섹션에 생성한 각 FlexGroup 지원 Trident 백엔드에 대해 별도의 StorageClass를 생성할 것을 권장합니다 ["ONTAP AI 배포에 대한 Trident 백엔드 예"](#), 1단계. 이러한 세분화된 StorageClasses를 사용하면 특정 LIF(Trident 백엔드를 생성할 때 지정한 LIF)에 해당하는 NFS 마운트를 StorageClass 사양 파일에 지정된 특정 백엔드에서 추가할 수 있습니다. 다음 예제 명령은 섹션에 생성된 두 예제 백엔드에 해당하는 두 개의 StorageClasses를 생성하는 방법을 보여 줍니다 ["ONTAP AI 배포에 대한 Trident 백엔드 예"](#), 1단계. StorageClasses에 대한 자세한 내용은 를 참조하십시오 ["Trident 문서"](#).

영구 볼륨은 해당 PersistentVolumeClaim(PVC)이 삭제되어도 삭제되지 않도록 다음 예에서는 "Retain"의 "reclaimPolicy" 값을 사용합니다. '청구 정책' 필드에 대한 자세한 내용은 공식 을 참조하십시오 ["Kubernetes 문서"](#).

```
$ cat << EOF > ./storage-class-ontap-ai-flexgroups-retain-iface1.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-ai-flexgroups-retain-iface1
provisioner: netapp.io/trident
parameters:
  backendType: "ontap-nas-flexgroup"
  storagePools: "ontap-ai-flexgroups-iface1:.*"
reclaimPolicy: Retain
EOF
$ kubectl create -f ./storage-class-ontap-ai-flexgroups-retain-iface1.yaml
storageclass.storage.k8s.io/ontap-ai-flexgroups-retain-iface1 created
$ cat << EOF > ./storage-class-ontap-ai-flexgroups-retain-iface2.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-ai-flexgroups-retain-iface2
provisioner: netapp.io/trident
parameters:
  backendType: "ontap-nas-flexgroup"
  storagePools: "ontap-ai-flexgroups-iface2:.*"
reclaimPolicy: Retain
EOF
$ kubectl create -f ./storage-class-ontap-ai-flexgroups-retain-iface2.yaml
storageclass.storage.k8s.io/ontap-ai-flexgroups-retain-iface2 created
$ kubectl get storageclass
NAME                                     PROVISIONER                AGE
ontap-ai-flexgroups-retain-iface1      netapp.io/trident          0m
ontap-ai-flexgroups-retain-iface2      netapp.io/trident          0m
```

2. 또한 섹션에서 생성한 FlexVol 지원 Trident 백엔드에 해당하는 StorageClass를 생성하는 것이 좋습니다 ["ONTAP AI 배포에 대한 Trident 백엔드 예"](#), 2단계. 다음 명령 예에서는 FlexVol 볼륨에 대한 단일 StorageClass를 생성하는 것을 보여 줍니다.

다음 예에서는 FlexVol 지원 Trident 백엔드가 하나만 생성되었기 때문에 StorageClass 정의 파일에 특정 백엔드가 지정되지 않습니다. Kubernetes를 사용하여 이 StorageClass를 사용하는 볼륨을 관리할 경우 Trident는 'ONTAP-NAS' 드라이버를 사용하는 사용 가능한 백엔드를 사용하려고 합니다.

```
$ cat << EOF > ./storage-class-ontap-ai-flexvols-retain.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-ai-flexvols-retain
provisioner: netapp.io/trident
parameters:
  backendType: "ontap-nas"
reclaimPolicy: Retain
EOF
$ kubectl create -f ./storage-class-ontap-ai-flexvols-retain.yaml
storageclass.storage.k8s.io/ontap-ai-flexvols-retain created
$ kubectl get storageclass
```

NAME	PROVISIONER	AGE
ontap-ai-flexgroups-retain-iface1	netapp.io/trident	1m
ontap-ai-flexgroups-retain-iface2	netapp.io/trident	1m
ontap-ai-flexvols-retain	netapp.io/trident	0m

3. 또한 FlexGroup 볼륨에 대한 일반 StorageClass를 생성하는 것이 좋습니다. 다음 예제 명령은 FlexGroup 볼륨에 대한 단일 일반 StorageClass를 생성하는 방법을 보여 줍니다.

StorageClass 정의 파일에 특정 백엔드가 지정되지 않았습니다. 따라서 Kubernetes를 사용하여 이 StorageClass를 사용하는 볼륨을 관리할 때 Trident는 'ONTAP-NAS-Flexgroup' 드라이버를 사용하는 사용 가능한 백엔드를 사용하려고 합니다.


```

$ cat << EOF > ./storage-class-ontap-ai-flexgroups-retain.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-ai-flexgroups-retain
provisioner: netapp.io/trident
parameters:
  backendType: "ontap-nas-flexgroup"
reclaimPolicy: Retain
EOF
$ kubectl create -f ./storage-class-ontap-ai-flexgroups-retain.yaml
storageclass.storage.k8s.io/ontap-ai-flexgroups-retain created
$ kubectl get storageclass

```

NAME	PROVISIONER	AGE
ontap-ai-flexgroups-retain	netapp.io/trident	0m
ontap-ai-flexgroups-retain-iface1	netapp.io/trident	2m
ontap-ai-flexgroups-retain-iface2	netapp.io/trident	2m
ontap-ai-flexvols-retain	netapp.io/trident	1m

저작권 정보

Copyright © 2024 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 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 라이선스와 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 라이선스도 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 라이선스를 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 라이선스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

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