



NetApp Trident 概述

NetApp Solutions

NetApp
December 19, 2024

目录

NetApp Trident 概述	1
Trident概述	1
NetApp ONTAP NFS 配置	3
NetApp ONTAP iSCSI 配置	9

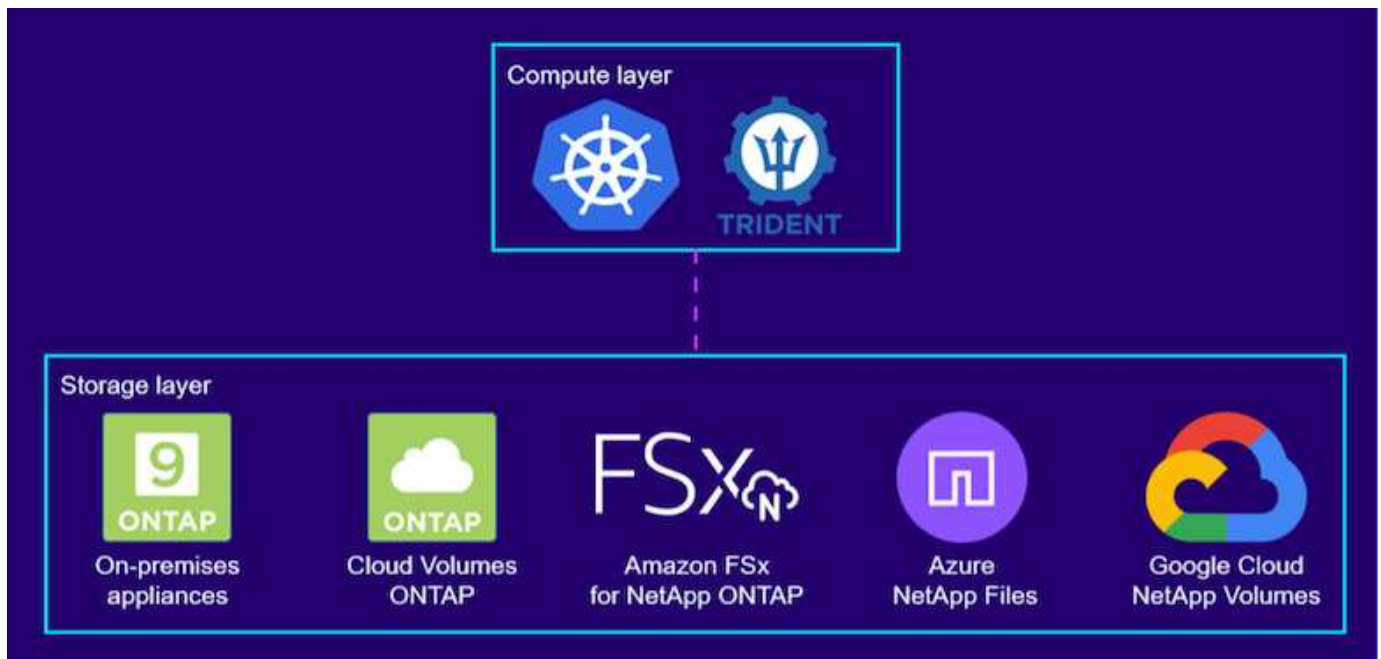
NetApp Trident 概述

Trident概述

Trident是一款完全受支持的开源存储流程编排程序、适用于容器和Kubernetes分发版(包括VMware Tanzu)。

Trident是一款完全受支持的开源存储流程编排程序、适用于容器和KubeNet分发(如 {k8s_Distribution_name})。Trident 可与包括 NetApp ONTAP 和 Element 存储系统在内的整个 NetApp 存储产品组合配合使用，并且还支持 NFS 和 iSCSI 连接。Trident 允许最终用户从其 NetApp 存储系统配置和管理存储，而无需存储管理员干预，从而加快了 DevOps 工作流的速度。

管理员可以根据项目需求和存储系统型号配置多个存储后端，以实现高级存储功能，包括数据压缩，特定磁盘类型或 QoS 级别，以保证一定水平的性能。定义后，开发人员可以在其项目中使用这些后端创建永久性卷声明（PVC），并按需将永久性存储附加到容器。



Trident的开发周期较短、与Kubernetes一样、它每年发布四次。

Trident的最新版本为22.04、于2022年4月发布。一个支持表，用于说明已使用哪个版本的Trident进行了测试，可以找到Kubernetes分发 ["此处"](#)。

从 20.04 版开始，Trident 设置由 Trident 操作员执行。操作员可以简化大规模部署、并提供额外的支持、包括为在Trident安装过程中部署的Pod提供自我修复。

在 21.01 版中，我们提供了一个 Helm 图表，用于简化 Trident 操作员的安装。

使用Helm部署Trident操作员

1. 首先将用户集群的 kubeconfig 文件的位置设置为环境变量，以便您不必引用该文件，因为 Trident 没有传递此文件的选项。

```
[netapp-user@rhel7]$ export KUBECONFIG=~/.tanzu-install/auth/kubeconfig
```

2. 添加NetApp Trident Helm存储库。

```
[netapp-user@rhel7]$ helm repo add netapp-trident  
https://netapp.github.io/trident-helm-chart  
"netapp-trident" has been added to your repositories
```

3. 更新Helm存储库。

```
[netapp-user@rhel7]$ helm repo update  
Hang tight while we grab the latest from your chart repositories...  
...Successfully got an update from the "netapp-trident" chart repository  
...Successfully got an update from the "bitnami" chart repository  
Update Complete. ☐Happy Helming!☐
```

4. 为Trident的安装创建新的命名空间。

```
[netapp-user@rhel7]$ kubectl create ns trident
```

5. 使用多端口集线器凭据创建一个密钥以下载Trident映像。

```
[netapp-user@rhel7]$ kubectl create secret docker-registry docker-  
registry-cred --docker-server=docker.io --docker-username=netapp  
-solutions-tme --docker-password=xxxxxxx -n trident
```

6. 对于由TKGS (采用Tanzu的vSphere)管理的用户或工作负载集群或部署了管理集群的TKG管理的用户或工作负载集群、请完成以下过程以安装Trident：

- a. 确保已登录用户有权在Trident命名空间中创建服务帐户、并且Trident命名空间中的服务帐户有权创建Pod。
- b. 运行以下helm命令、在创建的命名空间中安装Trident操作员。

```
[netapp-user@rhel7]$ helm install trident netapp-trident/trident-  
operator -n trident --set imagePullSecrets[0]=docker-registry-cred
```

7. 对于由TKGI部署管理的用户或工作负载集群、请运行以下helm命令在创建的命名空间中安装Trident操作员。

```
[netapp-user@rhel7]$ helm install trident netapp-trident/trident-operator -n trident --set imagePullSecrets[0]=docker-registry-cred,kubeletDir="/var/vcap/data/kubelet"
```

8. 验证Trident Pod是否已启动且正在运行。

```
NAME                                READY   STATUS    RESTARTS
AGE
trident-csi-6vv62                    2/2    Running   0
14m
trident-csi-cfd844bcc-sqhcg         6/6    Running   0
12m
trident-csi-dfcmz                    2/2    Running   0
14m
trident-csi-pb2n7                    2/2    Running   0
14m
trident-csi-qsw6z                    2/2    Running   0
14m
trident-operator-67c94c4768-xw978  1/1    Running   0
14m
```

```
[netapp-user@rhel7]$ ./tridentctl -n trident version
+-----+-----+
| SERVER VERSION | CLIENT VERSION |
+-----+-----+
| 22.04.0        | 22.04.0        |
+-----+-----+
```

创建存储系统后端

完成Trident操作员安装后、您必须为所使用的特定NetApp存储平台配置后端。请单击以下链接继续设置和配置Trident。

- ["NetApp ONTAP NFS"](#)
- ["NetApp ONTAP iSCSI"](#)

NetApp ONTAP NFS 配置

要通过NFS与NetApp ONTAP 存储系统实现Trident集成、您必须创建一个后端、以便与存储系统进行通信。我们在此解决方案 中配置了一个基本后端、但如果您要查找更多自定义选项、请访问相关文档 ["此处"](#)。

在ONTAP 中创建SVM

1. 登录到ONTAP 系统管理器、导航到存储> Storage VM、然后单击添加。
2. 输入SVM的名称、启用NFS协议、选中允许NFS客户端访问复选框、然后在导出策略规则中添加工作节点所在的子网、以便允许将卷作为PV挂载到工作负载集群中。

Add Storage VM ×

STORAGE VM NAME

trident_svm

Access Protocol

SMB/CIFS, NFS, S3

iSCSI

Enable SMB/CIFS

Enable NFS

Allow NFS client access

Add at least one rule to allow NFS clients to access volumes in this storage VM. [?](#)

EXPORT POLICY

Default

RULES

Rule Index	Clients	Access Protocols	Read-Only Rule	Read/Wr
	0.0.0.0/0	Any	Any	Any



如果您要在NSX-T中使用NAT部署用户集群或工作负载集群、则需要将出口子网(对于TKGS0)或浮动IP子网(对于TKGI)添加到导出策略规则中。

3. 提供数据LIF的详细信息以及SVM管理帐户的详细信息、然后单击保存。

NETWORK INTERFACE

Use multiple network interfaces when client traffic is high.

K8s-Ontap-01

IP ADDRESS

172.21.252.180

SUBNET MASK

24

GATEWAY

172.21.252.1



BROADCAST DOMAIN

Default



Storage VM Administration

Manage administrator account

USER NAME

vsadmin

PASSWORD

.....

CONFIRM PASSWORD

.....

Add a network interface for storage VM management.

4. 将聚合分配给SVM。导航到存储>存储VM、单击新创建的SVM旁边的省略号、然后单击编辑。选中将卷创建限制为首选本地层复选框、并将所需聚合附加到其中。

Edit Storage VM



STORAGE VM NAME

trident_svm

DEFAULT LANGUAGE

c.utf_8



DELETED VOLUME RETENTION PERIOD 

12

HOURS

Resource Allocation

Limit volume creation to preferred local tiers

LOCAL TIERS

K8s_Ontap_01_SSD_1 

Cancel

Save

5. 如果要安装Trident的用户或工作负载集群采用NAT方式进行部署、则由于SNAT、存储挂载请求可能来自非标准端口。默认情况下、ONTAP 仅允许从根端口发出卷挂载请求。因此、请登录到ONTAP 命令行界面并修改设置、以允许来自非标准端口的挂载请求。


```
ontap-01> vserver nfs modify -vserver tanzu_svm -mount-rootonly disabled
```

创建后端和StorageClasses

1. 对于提供NFS服务的NetApp ONTAP 系统、请使用backendName、managementLIF、dataLIF、SVM、username在jumphost上创建一个后端配置文件。密码和其他详细信息。

```
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "ontap-nas+10.61.181.221",
  "managementLIF": "172.21.224.201",
  "dataLIF": "10.61.181.221",
  "svm": "trident_svm",
  "username": "admin",
  "password": "password"
}
```



最佳做法是，将自定义 backendName 值定义为 storageDriverName 和为 NFS 提供服务的 dataLIF 的组合，以便于识别。

2. 运行以下命令创建Trident后端。

```
[netapp-user@rhel7]$ ./tridentctl -n trident create backend -f backend-ontap-nas.json
+-----+-----+
+-----+-----+-----+
|          NAME          | STORAGE DRIVER |                UUID                |
| STATE | VOLUMES | |
+-----+-----+-----+
+-----+-----+-----+
| ontap-nas+10.61.181.221 | ontap-nas      | be7a619d-c81d-445c-b80c-5c87a73c5b1e |
| online |         0 |
+-----+-----+-----+
+-----+-----+-----+
```

3. 创建后端后，您接下来必须创建一个存储类。以下存储类定义示例突出显示了必填字段和基本字段。参数`backendType`应反映新创建的Trident后端的存储驱动程序。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nfs
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
```

4. 运行kubect命令创建存储类。

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f storage-class-nfs.yaml
storageclass.storage.k8s.io/ontap-nfs created
```

5. 创建存储类后，您必须创建第一个永久性卷请求（PVC）。下面提供了一个PVC定义示例。确保`storageClassName`字段与刚刚创建的存储类的名称匹配。根据要配置的工作负载、可以根据需要进一步自定义PVC定义。

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: ontap-nfs
```

6. 发出kubect命令创建PVC。根据所创建的后备卷的大小，创建可能需要一些时间，因此您可以在该过程完成后进行观察。

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f pvc-basic.yaml
persistentvolumeclaim/basic created
```

```
[netapp-user@rhel7 trident-installer]$ kubectl get pvc
NAME      STATUS   VOLUME                                     CAPACITY
ACCESS MODES  STORAGECLASS  AGE
basic      Bound     pvc-b4370d37-0fa4-4c17-bd86-94f96c94b42d  1Gi
RWO                ontap-nfs      7s
```

NetApp ONTAP iSCSI 配置

要通过iSCSI将NetApp ONTAP 存储系统与适用于永久性卷的VMware Tanzu Kubernetes 集群集成、第一步是通过登录到每个节点并配置iSCSI实用程序或软件包以挂载iSCSI卷来准备节点。为此、请按照本节中所述的操作步骤 进行操作 "[链接。](#)"。



NetApp不建议在采用NAT方式部署VMware Tanzu Kubernetes集群时使用此操作步骤。



TKGI使用Bossh VM作为运行不可变配置映像的Tanzu Kubernetes集群的节点、而在重新启动后、对Bossh VM上的iSCSI软件包进行的任何手动更改都不会保持持久性。因此、NetApp建议对由TKGI部署和操作的Tanzu Kubernetes集群使用NFS卷作为永久性存储。

为iSCSI卷准备好集群节点后、您必须创建一个后端、以便与存储系统进行通信。我们在此解决方案 中配置了一个基本后端、但是、如果您要查找更多自定义选项、请访问文档 "[此处](#)"。

在ONTAP 中创建SVM

要在ONTAP 中创建SVM、请完成以下步骤：

1. 登录到ONTAP 系统管理器、导航到存储> Storage VM、然后单击添加。
2. 输入SVM的名称、启用iSCSI协议、然后提供数据LIF的详细信息。

Add Storage VM



STORAGE VM NAME

trident_svm_iscsi

Access Protocol

SMB/CIFS, NFS, S3

iSCSI

Enable iSCSI

NETWORK INTERFACE

K8s-Ontap-01

IP ADDRESS

10.61.181.231

SUBNET MASK

24

GATEWAY

10.61.181.1

BROADCAST DOMAIN

Defa...

Use the same subnet mask, gateway, and broadcast domain for all of the following interfaces

IP ADDRESS

10.61.181.232

SUBNET MASK

24

GATEWAY

10.61.181.1

BROADCAST DOMAIN

Defa...

3. 输入SVM管理帐户的详细信息、然后单击保存。

Storage VM Administration

Manage administrator account

USER NAME

vsadmin

PASSWORD

.....

CONFIRM PASSWORD

.....

Add a network interface for storage VM management.

Save

Cancel

4. 要将聚合分配给SVM、请导航到存储>存储VM、单击新创建的SVM旁边的省略号、然后单击编辑。选中将卷创建限制为首选本地层复选框、并将所需聚合附加到其中。

Edit Storage VM



STORAGE VM NAME

trident_svm_iscsi

DEFAULT LANGUAGE

c.utf_8



DELETED VOLUME RETENTION PERIOD 

12

HOURS

Resource Allocation

Limit volume creation to preferred local tiers

LOCAL TIERS

K8s_Ontap_01_SSD_1 

Cancel

Save

创建后端和StorageClasses

1. 对于提供NFS服务的NetApp ONTAP 系统、请使用backendName、managementLIF、dataLIF、SVM、username在jumphost上创建一个后端配置文件。 密码和其他详细信息。

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "ontap-san+10.61.181.231",
  "managementLIF": "172.21.224.201",
  "dataLIF": "10.61.181.231",
  "svm": "trident_svm_iscsi",
  "username": "admin",
  "password": "password"
}
```

2. 运行以下命令创建Trident后端。

```
[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident create
backend -f backend-ontap-san.json
+-----+-----+-----+
+-----+-----+-----+
|           NAME           | STORAGE DRIVER |           UUID           |
| STATE | VOLUMES | |           |           |
+-----+-----+-----+
+-----+-----+-----+
| ontap-san+10.61.181.231 | ontap-san      | 6788533c-7fea-4a35-b797- |
| fb9bb3322b91 | online |          0 |
+-----+-----+-----+
+-----+-----+-----+
```

3. 创建后端之后、您必须接下来创建一个存储类。以下存储类定义示例突出显示了必填字段和基本字段。参数`backendType`应反映新创建的Trident后端的存储驱动程序。另请注意 name-field 值，稍后必须引用该值。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-iscsi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
```



此文件中定义了一个名为 `fSType` 的可选字段。在iSCSI后端、可以将此值设置为特定的Linux文件系统类型(XFS、ext4等)、也可以删除此值、以便Tanzu Kubernetes集群决定要使用的文件系统。

4. 运行kubect命令创建存储类。

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f storage-class-iscsi.yaml
storageclass.storage.k8s.io/ontap-iscsi created
```

5. 创建存储类后，您必须创建第一个永久性卷请求（PVC）。下面提供了一个PVC定义示例。确保`storageClassName`字段与刚刚创建的存储类的名称匹配。根据要配置的工作负载、可以根据需要进一步自定义PVC定义。

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: ontap-iscsi
```

6. 发出kubectl命令创建PVC。根据所创建的后备卷的大小，创建可能需要一些时间，因此您可以在该过程完成后进行观察。

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f pvc-basic.yaml
persistentvolumeclaim/basic created
```

```
[netapp-user@rhel7 trident-installer]$ kubectl get pvc
```

NAME	STATUS	VOLUME	CAPACITY
basic	Bound	pvc-7ceac1ba-0189-43c7-8f98-094719f7956c	1Gi
RWO		ontap-iscsi	3s

版权信息

版权所有 © 2024 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。