



Azure NetApp Files

Trident

NetApp
January 15, 2026

This PDF was generated from <https://docs.netapp.com/zh-cn/trident-2506/trident-use/anf.html> on January 15, 2026. Always check docs.netapp.com for the latest.

目录

- Azure NetApp Files 1
 - 配置Azure NetApp Files后端 1
 - Azure NetApp Files驱动程序详细信息 1
 - 为 AKS 管理身份 1
 - AKS 的云身份 2
 - 准备配置Azure NetApp Files后端 4
 - NFS 和 SMB 卷的先决条件 4
 - 中小企业卷的附加要求 6
 - Azure NetApp Files后端配置选项和示例 6
 - 后端配置选项 7
 - 示例配置 9
 - 存储类定义 16
 - 创建后端 19

Azure NetApp Files

配置Azure NetApp Files后端

您可以将Azure NetApp Files配置为Trident的后端。您可以使用Azure NetApp Files后端附加 NFS 和 SMB 卷。Trident还支持使用托管标识对 Azure Kubernetes 服务 (AKS) 集群进行凭据管理。

Azure NetApp Files驱动程序详细信息

Trident提供以下Azure NetApp Files存储驱动程序，以便与集群通信。支持的访问模式有：*ReadWriteOnce* (RWO)、*ReadOnlyMany* (ROX)、*ReadWriteMany* (RWX)、*ReadWriteOncePod* (RWOP)。

驱动程序	协议	音量模式	支持的访问模式	支持的文件系统
azure-netapp-files	NFS SMB	Filesystem	RWO、ROX、RWX、RWOP	nfs, smb

注意事项

- Azure NetApp Files服务不支持小于 50 GiB 的卷。如果请求的卷较小，Trident会自动创建 50GiB 的卷。
- Trident仅支持挂载到运行在 Windows 节点上的 pod 的 SMB 卷。

为 AKS 管理身份

Trident支持"托管身份"适用于 Azure Kubernetes 服务集群。要利用托管身份提供的简化凭证管理功能，您必须具备以下条件：

- 使用 AKS 部署的 Kubernetes 集群
- 在 AKS Kubernetes 集群上配置的托管身份
- 已安装的Trident包括 `cloudProvider`指定`"Azure"`。

Trident操作员

要使用Trident操作员安装Trident，请编辑 `tridentorchestrator_cr.yaml` 设置 `cloudProvider` 到 `"Azure"`。例如：

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "Azure"
```

舵

以下示例安装Trident套装 `cloudProvider` 使用环境变量连接到 Azure `$CP`：

```
helm install trident trident-operator-100.2506.0.tgz --create
--namespace --namespace <trident-namespace> --set cloudProvider=$CP
```

`tridentctl`

以下示例安装Trident并进行设置 `cloudProvider` 标记 `"Azure"`：

```
tridentctl install --cloud-provider="Azure" -n trident
```

AKS 的云身份

云身份使 Kubernetes pod 能够通过作为工作负载身份进行身份验证来访问 Azure 资源，而无需提供显式的 Azure 凭据。

要在 Azure 中利用云身份功能，您必须具备以下条件：

- 使用 AKS 部署的 Kubernetes 集群
- 在 AKS Kubernetes 集群上配置工作负载身份和 `oidc-issuer`
- 已安装的Trident包括 `cloudProvider` 指定 `"Azure"` 和 `cloudIdentity` 指定工作负载标识

Trident操作员

要使用Trident操作员安装Trident，请编辑 `tridentorchestrator_cr.yaml` 设置 `cloudProvider` 到 `"Azure"` 并设置 `cloudIdentity` 到 `azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`。

例如：

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "Azure"
  cloudIdentity: 'azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx' # Edit
```

舵

使用以下环境变量设置 **cloud-provider (CP)** 和 **cloud-identity (CI)** 标志的值：

```
export CP="Azure"
export CI="'azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx'"
```

以下示例安装Trident并进行设置 `cloudProvider` 使用环境变量连接到 `Azure` `$CP` 并设置 `cloudIdentity` 使用环境变量 `$CI`：

```
helm install trident trident-operator-100.6.0.tgz --set
cloudProvider=$CP --set cloudIdentity="$CI"
```

`tridentctl`

请使用以下环境变量设置 云提供商 和 云身份 标志的值：

```
export CP="Azure"
export CI="azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
```

以下示例安装Trident并进行设置 `cloud-provider` 标记 `$CP`，和 `cloud-identity` 到 `$CI`：

```
tridentctl install --cloud-provider=$CP --cloud-identity="$CI" -n
trident
```

准备配置Azure NetApp Files后端

在配置Azure NetApp Files后端之前，需要确保满足以下要求。

NFS 和 SMB 卷的先决条件

如果您是第一次使用Azure NetApp Files或在新的位置使用，则需要进行一些初始配置来设置 Azure NetApp文件并创建 NFS 卷。参考 ["Azure：设置Azure NetApp Files并创建 NFS 卷"](#)。

配置和使用 ["Azure NetApp Files"](#)后端需要以下组件：



- subscriptionID, tenantID, clientID, location, 和 `clientSecret` 在 AKS 集群上使用托管身份时，这些是可选的。
- tenantID, clientID, 和 `clientSecret` 在 AKS 集群上使用云身份时，这些是可选的。

- 容量池。参考["微软：为Azure NetApp Files创建容量池"](#)。
- 委派给Azure NetApp Files 的子网。参考["Microsoft：将子网委派给Azure NetApp Files"](#)。
- `subscriptionID` 来自已启用Azure NetApp Files功能的 Azure 订阅。
- tenantID, clientID, 和 `clientSecret` 来自["应用程序注册"](#)在 Azure Active Directory 中拥有对Azure NetApp Files服务的足够权限。应用程序注册应使用以下任一方式：
 - 所有者或贡献者角色["Azure 预定义"](#)。
 - 一个["自定义贡献者角色"](#)订阅级别(assignableScopes) 但权限仅限于Trident所需的权限。创建自定义角色后，["使用 Azure 门户分配角色"](#)。

```

{
  "id": "/subscriptions/<subscription-
id>/providers/Microsoft.Authorization/roleDefinitions/<role-
definition-id>",
  "properties": {
    "roleName": "custom-role-with-limited-perms",
    "description": "custom role providing limited permissions",
    "assignableScopes": [
      "/subscriptions/<subscription-id>"
    ],
    "permissions": [
      {
        "actions": [
          "Microsoft.NetApp/netAppAccounts/capacityPools/read",
          "Microsoft.NetApp/netAppAccounts/capacityPools/write",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/read",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/write",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/delete",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/
read",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/
write",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/
delete",

          "Microsoft.NetApp/netAppAccounts/capacityPools/volumes/MountTarge
ts/read",
          "Microsoft.Network/virtualNetworks/read",
          "Microsoft.Network/virtualNetworks/subnets/read",

          "Microsoft.Features/featureProviders/subscriptionFeatureRegistrat
ions/read",

          "Microsoft.Features/featureProviders/subscriptionFeatureRegistrat
ions/write",

          "Microsoft.Features/featureProviders/subscriptionFeatureRegistrat

```

```

ions/delete",
    "Microsoft.Features/features/read",
    "Microsoft.Features/operations/read",
    "Microsoft.Features/providers/features/read",

"Microsoft.Features/providers/features/register/action",

"Microsoft.Features/providers/features/unregister/action",

"Microsoft.Features/subscriptionFeatureRegistrations/read"
    ],
    "notActions": [],
    "dataActions": [],
    "notDataActions": []
  }
]
}
}

```

- 蔚蓝 `location` 包含至少一个 **"委托子网"**。截至 Trident 22.01 版本，`location` 参数是后端配置文件顶层的一个必填字段。虚拟池中指定的位置值将被忽略。
- 使用 Cloud Identity 得到 `client ID` 从一个 **"用户分配的托管身份"** 并指定该 ID
`azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`。

中小企业卷的附加要求

要创建 SMB 卷，您必须具备以下条件：

- Active Directory 已配置并连接到 Azure NetApp Files。参考["Microsoft：创建和管理 Azure NetApp Files 管理器的 Active Directory 连接"](#)。
- 一个 Kubernetes 集群，包含一个 Linux 控制器节点和至少一个运行 Windows Server 2022 的 Windows 工作节点。Trident 仅支持挂载到运行在 Windows 节点上的 pod 的 SMB 卷。
- 至少需要一个包含 Active Directory 凭据的 Trident 密钥，以便 Azure NetApp Files 可以向 Active Directory 进行身份验证。生成秘密 smbcreds：

```

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

```

- 配置为 Windows 服务的 CSI 代理。要配置 csi-proxy，请参阅["GitHub：CSI 代理"](#)或者["GitHub：适用于 Windows 的 CSI 代理"](#)适用于在 Windows 上运行的 Kubernetes 节点。

Azure NetApp Files 后端配置选项和示例

了解 Azure NetApp Files 的 NFS 和 SMB 后端配置选项，并查看配置示例。

后端配置选项

Trident使用您的后端配置（子网、虚拟网络、服务级别和位置），在请求的位置中可用的容量池上创建Azure NetApp Files卷，并与请求的服务级别和子网相匹配。



* 从NetApp Trident 25.06 版本开始，手动 QoS 容量池作为技术预览版受到支持。*

Azure NetApp Files后端提供以下配置选项。

参数	描述	默认
version		始终为 1
storageDriverName	存储驱动程序的名称	"azure-netapp-files"
backendName	自定义名称或存储后端	司机姓名 + "_" + 随机字符
subscriptionID	Azure 订阅的订阅 ID（在 AKS 群集上启用托管标识时为可选）。	
tenantID	在 AKS 集群上使用托管身份或云身份时，应用程序注册中的租户 ID 是可选的。	
clientID	在 AKS 集群上使用托管身份或云身份时，应用程序注册中的客户端 ID 是可选的。	
clientSecret	在 AKS 集群上使用托管身份或云身份时，应用程序注册中的客户端密钥是可选的。	
serviceLevel	之一 Standard, Premium, 或者 Ultra	""（随机的）
location	将在 Azure 中创建新卷的位置名称（在 AKS 群集上启用托管标识时为可选）。	
resourceGroups	用于筛选已发现资源的资源组列表	（无滤镜）
netappAccounts	用于筛选已发现资源的NetApp帐户列表	（无滤镜）
capacityPools	用于筛选已发现资源的容量池列表	（无过滤，随机）
virtualNetwork	具有委派子网的虚拟网络的名称	""
subnet	委派给子网的名称 Microsoft.Netapp/volumes	""
networkFeatures	卷的 VNet 功能集，可能是 Basic 或者 Standard。网络功能并非在所有地区都可用，可能需要通过订阅才能启用。指定 networkFeatures 未启用该功能会导致卷配置失败。	""

参数	描述	默认
nfsMountOptions	对 NFS 挂载选项进行精细控制。对于 SMB 卷，此设置将被忽略。要使用 NFS 版本 4.1 挂载卷，请包含以下内容 `nfsvers=4` 在以逗号分隔的挂载选项列表中选择 NFS v4.1。存储类定义中设置的挂载选项会覆盖后端配置中设置的挂载选项。	"nfsvers=3"
limitVolumeSize	如果请求的卷大小大于此值，则配置失败。	(默认情况下不强制执行)
debugTraceFlags	故障排除时要使用的调试标志。例子， <code>\{"api": false, "method": true, "discovery": true\}</code> 。除非您正在进行故障排除并需要详细的日志转储，否则请勿使用此功能。	无效的
nasType	配置 NFS 或 SMB 卷的创建。选项有 <code>nfs</code> ， <code>smb</code> 或空值。设置为 <code>null</code> 则默认使用 NFS 卷。	nfs
supportedTopologies	表示此后端支持的区域和区域列表。更多信息，请参阅 "使用 CSI 拓扑" 。	
qosType	表示 QoS 类型：自动或手动。* Trident 25.06 技术预览版*	自动
maxThroughput	设置允许的最大吞吐量，单位为 MiB/秒。仅支持手动 QoS 容量池。* Trident 25.06 技术预览版*	4 MiB/sec



有关网络功能的更多信息，请参阅["为 Azure NetApp Files 卷配置网络功能"](#)。

所需权限和资源

如果在创建 PVC 时收到“未找到容量池”错误，则可能是您的应用程序注册没有关联的必要权限和资源（子网、虚拟网络、容量池）。如果启用调试功能，Trident 将记录在创建后端时发现的 Azure 资源。请确认是否使用了合适的角色。

值 `resourceGroups`，`netappAccounts`，`capacityPools`，`virtualNetwork`，和 `subnet` 可以使用简称或完全限定名称来指定。大多数情况下建议使用完全限定名称，因为短名称可能会匹配多个同名资源。

这 `resourceGroups`，`netappAccounts`，和 `capacityPools` 值是过滤器，用于将发现的资源集限制为该存储后端可用的资源，并且可以以任意组合指定。完全限定名称遵循以下格式：

类型	格式
资源组	<资源组>
NetApp 帐户	<资源组>/<NetApp 帐户>
容量池	<资源组>/<NetApp 帐户>/<容量池>

类型	格式
虚拟网络	<资源组>/<虚拟网络>
子网	<资源组>/<虚拟网络>/<子网>

卷配置

您可以通过在配置文件的特定部分中指定以下选项来控制默认卷配置。参考 [\[示例配置\]](#) 了解详情。

参数	描述	默认
exportRule	新卷的出口规则。 `exportRule` 必须是以逗号分隔的 IPv4 地址或 IPv4 子网的任意组合列表，采用 CIDR 表示法。对于 SMB 卷，此设置将被忽略。	“0.0.0.0/0”
snapshotDir	控制 .snapshot 目录的可见性	NFSv4 为“true”，NFSv3 为“false”。
size	新卷的默认大小	100G
unixPermissions	新卷的 Unix 权限（4 位八进制数字）。对于 SMB 卷，此设置将被忽略。	（预览功能，需订阅并加入白名单）

示例配置

以下示例展示了基本配置，其中大多数参数都保留默认值。这是定义后端最简单的方法。

最小配置

这是最基本的后端配置。通过此配置，Trident会发现配置位置中所有委派给Azure NetApp Files存储的NetApp帐户、容量池和子网，并将新卷随机放置在其中一个池和子网上。因为`nasType`省略了`nfs`默认设置生效，后端将为NFS卷进行配置。

如果您刚开始使用Azure NetApp Files并进行尝试，此配置是理想的选择，但在实践中，您需要为预配的卷提供额外的范围。

```
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
  tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
  clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
  clientSecret: SECRET
  location: eastus
```

为 AKS 管理身份

此后端配置省略了 subscriptionID, tenantID, clientID, 和 `clientSecret` 在使用托管身份时, 这些是可选的。

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  capacityPools:
    - ultra-pool
  resourceGroups:
    - aks-ami-eastus-rg
  netappAccounts:
    - smb-na
  virtualNetwork: eastus-prod-vnet
  subnet: eastus-anf-subnet
```

AKS 的云身份

此后端配置省略了 `tenantID`，`clientID`，和 `clientSecret` 在使用云身份时，这些是可选的。

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  capacityPools:
    - ultra-pool
  resourceGroups:
    - aks-ami-eastus-rg
  netappAccounts:
    - smb-na
  virtualNetwork: eastus-prod-vnet
  subnet: eastus-anf-subnet
  location: eastus
  subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
```

具有容量池过滤器的特定服务级别配置

此后端配置将卷放置在 Azure 中 `eastus` 位置 `Ultra` 容量池。Trident 会自动发现该位置中委派给 Azure NetApp Files 的所有子网，并随机在其中一个子网上放置一个新卷。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
serviceLevel: Ultra
capacityPools:
  - application-group-1/account-1/ultra-1
  - application-group-1/account-1/ultra-2
```

此后端配置将卷放置在 Azure 中 `eastus` 具有手动 QoS 容量池的位置。* NetApp Trident 25.06 中的技术预览*。

```
---
version: 1
storageDriverName: azure-netapp-files
backendName: anf1
location: eastus
labels:
  clusterName: test-cluster-1
  cloud: anf
  nasType: nfs
defaults:
  qosType: Manual
storage:
  - serviceLevel: Ultra
    labels:
      performance: gold
    defaults:
      maxThroughput: 10
  - serviceLevel: Premium
    labels:
      performance: silver
    defaults:
      maxThroughput: 5
  - serviceLevel: Standard
    labels:
      performance: bronze
    defaults:
      maxThroughput: 3
```

此后端配置进一步缩小了卷放置范围到单个子网，并且还修改了一些卷配置默认值。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
serviceLevel: Ultra
capacityPools:
  - application-group-1/account-1/ultra-1
  - application-group-1/account-1/ultra-2
virtualNetwork: my-virtual-network
subnet: my-subnet
networkFeatures: Standard
nfsMountOptions: vers=3,proto=tcp,timeo=600
limitVolumeSize: 500Gi
defaults:
  exportRule: 10.0.0.0/24,10.0.1.0/24,10.0.2.100
  snapshotDir: "true"
  size: 200Gi
  unixPermissions: "0777"
```


此后端配置在单个文件中定义了多个存储池。当您有多个容量池支持不同的服务级别，并且想要在 Kubernetes 中创建代表这些级别的存储类时，这将非常有用。虚拟池标签用于根据以下因素区分池子：performance。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
resourceGroups:
  - application-group-1
networkFeatures: Basic
nfsMountOptions: vers=3,proto=tcp,timeo=600
labels:
  cloud: azure
storage:
  - labels:
      performance: gold
      serviceLevel: Ultra
      capacityPools:
        - ultra-1
        - ultra-2
      networkFeatures: Standard
  - labels:
      performance: silver
      serviceLevel: Premium
      capacityPools:
        - premium-1
  - labels:
      performance: bronze
      serviceLevel: Standard
      capacityPools:
        - standard-1
        - standard-2
```

Trident可根据区域和可用区为工作负载提供卷。这 `supportedTopologies` 此后端配置中的块用于提供每个后端的区域和区域列表。此处指定的区域和区域值必须与每个 Kubernetes 集群节点上的标签中的区域和区域值相匹配。这些区域和分区代表存储类中可以提供的允许值的列表。对于包含后端提供的区域和可用区子集的存储类，Trident会在所述区域和可用区中创建卷。更多信息，请参阅["使用 CSI 拓扑"](#)。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
serviceLevel: Ultra
capacityPools:
  - application-group-1/account-1/ultra-1
  - application-group-1/account-1/ultra-2
supportedTopologies:
  - topology.kubernetes.io/region: eastus
    topology.kubernetes.io/zone: eastus-1
  - topology.kubernetes.io/region: eastus
    topology.kubernetes.io/zone: eastus-2
```

存储类定义

下列 `StorageClass` 上述定义指的是存储池。

使用示例定义 `parameter.selector` 场地

使用 `parameter.selector` 你可以为每个对象指定。`StorageClass` 用于托管卷的虚拟池。该卷将具有所选池中定义的方面。

```

---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: gold
provisioner: csi.trident.netapp.io
parameters:
  selector: performance=gold
allowVolumeExpansion: true

---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: silver
provisioner: csi.trident.netapp.io
parameters:
  selector: performance=silver
allowVolumeExpansion: true

---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: bronze
provisioner: csi.trident.netapp.io
parameters:
  selector: performance=bronze
allowVolumeExpansion: true

```

SMB卷的示例定义

使用 `nasType`, `node-stage-secret-name`, 和 `node-stage-secret-namespace` 您可以指定 SMB 卷并提供所需的 Active Directory 凭据。

默认命名空间上的基本配置

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: "default"
```

每个命名空间使用不同的密钥

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

每个卷使用不同的密钥

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: ${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```



`nasType: smb`筛选支持 SMB 卷的存储池。`nasType: nfs`或者`nasType: null`NFS池过滤器。

创建后端

创建后端配置文件后，运行以下命令：

```
tridentctl create backend -f <backend-file>
```

如果后端创建失败，则后端配置存在问题。您可以通过运行以下命令查看日志以确定原因：

```
tridentctl logs
```

在您发现并纠正配置文件中的问题后，您可以再次运行创建命令。

版权信息

版权所有 © 2026 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，本文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。