ONTAP SAN 驱动程序

Trident

NetApp
January 15, 2026

目录

ONTAP SAN 驱动程序

ONTAP SAN 驱动程序概述

了解如何使用ONTAP和Cloud Volumes ONTAP SAN 驱动程序配置ONTAP后端。

ONTAP SAN 驱动程序详情

Trident提供以下 SAN 存储驱动程序，用于与ONTAP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
`ontap-san`	iSCSI 通过光纤通道提供 SCSI 服务	块	RWO、ROX、RWX、RWOP	无文件系统；原始块设备
`ontap-san`	iSCSI 通过光纤通道提供 SCSI 服务	Filesystem	RWO，RWOP ROX 和 RWX 在文件系统卷模式下不可用。	`xfs`， `ext3` ， `ext4`
`ontap-san`	NVMe/TCP 参考NVMe/TCP 的其他注意事项。	块	RWO、ROX、RWX、RWOP	无文件系统；原始块设备
`ontap-san`	NVMe/TCP 参考NVMe/TCP 的其他注意事项。	Filesystem	RWO，RWOP ROX 和 RWX 在文件系统卷模式下不可用。	`xfs`， `ext3` ， `ext4`
`ontap-san-economy`	iSCSI	块	RWO、ROX、RWX、RWOP	无文件系统；原始块设备
`ontap-san-economy`	iSCSI	Filesystem	RWO，RWOP ROX 和 RWX 在文件系统卷模式下不可用。	`xfs`， `ext3` ， `ext4`

- 使用 `ontap-san-economy`仅当预计持续卷使用量高于……时"支持的ONTAP容量限制"。
- 使用 `ontap-nas-economy`仅当预计持续卷使用量高于……时"支持的ONTAP容量限制"以及 `ontap-san-economy`驱动程序无法使用。
- 请勿使用 `ontap-nas-economy`如果您预计需要数据保护、灾难恢复或移动办公。
- NetApp不建议在所有ONTAP驱动程序中使用 Flexvol 自动增长，ontap-san 除外。作为一种变通方法， Trident支持使用快照储备，并相应地调整 Flexvol 容量。

用户权限

Trident预期以ONTAP或 SVM 管理员身份运行，通常使用以下方式： `admin`集群用户或 `vsadmin`SVM 用户，或者具有相同角色但名称不同的用户。对于Amazon FSx for NetApp ONTAP部署， Trident需要以ONTAP或 SVM 管理员身份运行，并使用集群。 `fsxadmin`用户或 `vsadmin`SVM 用户，或者具有相同角色但名称不同的用户。这 `fsxadmin`用户是集群管理员用户的有限替代品。

如果你使用 `limitAggregateUsage`需要参数和集群管理员权限。当使用Amazon FSx for NetApp ONTAP和Trident时， `limitAggregateUsage`参数将无法与 `vsadmin`和 `fsxadmin`用户账户。如果指定此参数，配置操作将失败。

虽然可以在ONTAP中创建一个Trident驱动程序可以使用的限制性更强的角色，但我们不建议这样做。 Trident的大多数新版本都会调用额外的 API，这些 API 必须加以考虑，这使得升级变得困难且容易出错。

NVMe/TCP 的其他注意事项

Trident支持使用非易失性存储器高速接口 (NVMe) 协议 `ontap-san`驱动程序包括：

- IPv6
- NVMe卷的快照和克隆
- 调整 NVMe 卷的大小
- 导入在Trident外部创建的 NVMe 卷，以便Trident可以管理其生命周期。
- NVMe原生多路径
- K8s节点的优雅关闭或非优雅关闭（24.06）

Trident不支持：

- NVMe 原生支持的 DH-HMAC-CHAP
- 设备映射器 (DM) 多路径
- LUKS 加密

NVMe 仅支持ONTAP REST API，不支持 ONTAPI (ZAPI)。

准备配置后端ONTAP SAN 驱动程序

了解配置ONTAP后端和ONTAP SAN 驱动程序的要求和身份验证选项。

要求

对于所有ONTAP后端， Trident要求至少将一个聚合分配给 SVM。

"ASA r2 系统"与其他ONTAP系统（ASA、 AFF和FAS）在存储层的实现上有所不同。在ASA r2系统中，使用存储可用区而不是聚合。请参阅"这"知识库文章，介绍如何在ASA r2 系统中将聚合分配给 SVM。

请记住，您还可以运行多个驱动程序，并创建指向其中一个或另一个驱动程序的存储类。例如，您可以配置一个 `san-dev`使用类 `ontap-san`司机和 `san-default`使用类 `ontap-san-economy`一。

所有 Kubernetes 工作节点都必须安装相应的 iSCSI 工具。参考 "准备工作节点" 了解详情。

对**ONTAP**后端进行身份验证

Trident提供两种ONTAP后端身份验证方式。

- 基于凭证：具有所需权限的ONTAP用户的用户名和密码。建议使用预定义的安全登录角色，例如： `admin`或者 `vsadmin`确保与ONTAP版本最大程度兼容。
- 基于证书： Trident还可以使用安装在后端的证书与ONTAP集群通信。此处，后端定义必须包含客户端证书、密钥和受信任 CA 证书（如果使用，建议使用）的 Base64 编码值。

您可以更新现有后端，以在基于凭据的方法和基于证书的方法之间进行切换。但是，一次只能支持一种身份验证方法。要切换到不同的身份验证方法，必须从后端配置中删除现有方法。

如果您尝试同时提供凭据和证书，则后端创建将失败，并出现错误，提示配置文件中提供了多个身份验证方法。

启用基于凭据的身份验证

Trident需要 SVM 范围/集群范围管理员的凭据才能与ONTAP后端通信。建议使用标准的、预定义的角色，例如： `admin`或者 `vsadmin`。这样可以确保与未来ONTAP版本向前兼容，这些版本可能会公开一些功能 API，供未来的Trident版本使用。虽然可以创建自定义安全登录角色并将其与Trident一起使用，但不建议这样做。

后端定义示例如下所示：

YAML

```yaml
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: password
```

JSON

```json
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-san",
  "managementLIF": "10.0.0.1",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "password"
}
```

请注意，后端定义是唯一以纯文本形式存储凭据的地方。后端创建完成后，用户名/密码将使用 Base64 进行编码，并存储为 Kubernetes 密钥。只有创建或更新后端时才需要了解凭据。因此，这是一项仅限管理员执行的操作，由 Kubernetes/存储管理员执行。

启用基于证书的身份验证

新的和现有的后端都可以使用证书与ONTAP后端通信。后端定义需要三个参数。

- clientCertificate：客户端证书的 Base64 编码值。
- clientPrivateKey：关联私钥的 Base64 编码值。
- trustedCACertificate：受信任 CA 证书的 Base64 编码值。如果使用受信任的 CA，则必须提供此参数。如果没有使用受信任的证书颁发机构，则可以忽略此步骤。

典型的工作流程包括以下步骤。

步骤

1. 生成客户端证书和密钥。生成时，将通用名称 (CN) 设置为要进行身份验证的ONTAP用户。

```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=admin"
```

2. 向ONTAP集群添加受信任的 CA 证书。这可能已经由存储管理员处理了。如果没有使用受信任的证书颁发机构，则忽略此操作。

```
security certificate install -type server -cert-name <trusted-ca-cert-
name> -vserver <vserver-name>
ssl modify -vserver <vserver-name> -server-enabled true -client-enabled
true -common-name <common-name> -serial <SN-from-trusted-CA-cert> -ca
<cert-authority>
```

3. 在ONTAP集群上安装客户端证书和密钥（来自步骤 1）。

```
security certificate install -type client-ca -cert-name <certificate-
name> -vserver <vserver-name>
security ssl modify -vserver <vserver-name> -client-enabled true
```

4. 确认ONTAP安全登录角色支持 `cert`身份验证方法。

```
security login create -user-or-group-name admin -application ontapi
-authentication-method cert
security login create -user-or-group-name admin -application http
-authentication-method cert
```

5. 使用生成的证书进行身份验证。请将 < ONTAP管理 LIF> 和 <vserver 名称> 替换为管理 LIF IP 地址和 SVM 名称。

```
curl -X POST -Lk https://<ONTAP-Management-
LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key
--cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp
xmlns="http://www.netapp.com/filer/admin" version="1.21"
vfiler="<vserver-name>"><vserver-get></vserver-get></netapp>'
```

6. 使用 Base64 对证书、密钥和受信任的 CA 证书进行编码。

```
base64 -w 0 k8senv.pem >> cert_base64
base64 -w 0 k8senv.key >> key_base64
base64 -w 0 trustedca.pem >> trustedca_base64
```

7. 使用上一步获得的值创建后端。

```
cat cert-backend.json
{
"version": 1,
"storageDriverName": "ontap-san",
"backendName": "SanBackend",
"managementLIF": "1.2.3.4",
"svm": "vserver_test",
"clientCertificate": "Faaaakkkkeeee...Vaaalllluuuueeee",
"clientPrivateKey": "LS0tFaKE...0VaLuES0tLS0K",
"trustedCACertificate": "QNFinfO...SiqOyN",
"storagePrefix": "myPrefix_"
}

tridentctl create backend -f cert-backend.json -n trident
+------------+----------------+--------------------------------------
+--------+---------+
|    NAME    | STORAGE DRIVER |                 UUID                 |
STATE  | VOLUMES |
+------------+----------------+--------------------------------------
+--------+---------+
| SanBackend | ontap-san      | 586b1cd5-8cf8-428d-a76c-2872713612c1 |
online |       0 |
+------------+----------------+--------------------------------------
+--------+---------+
```

更新身份验证方法或轮换凭据

您可以更新现有后端，以使用不同的身份验证方法或轮换其凭据。这种方法是双向的：使用用户名/密码的后端可以更新为使用证书；使用证书的后端可以更新为基于用户名/密码的后端。为此，您必须删除现有的身份验证方法并添加新的身份验证方法。然后使用包含所需参数的更新后的 backend.json 文件来执行 `tridentctl backend update`。

```
cat cert-backend-updated.json
{
"version": 1,
"storageDriverName": "ontap-san",
"backendName": "SanBackend",
"managementLIF": "1.2.3.4",
"svm": "vserver_test",
"username": "vsadmin",
"password": "password",
"storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
tridentctl update backend SanBackend -f cert-backend-updated.json -n
trident
+------------+----------------+--------------------------------------
+--------+---------+
|    NAME    | STORAGE DRIVER |                 UUID                 |
STATE   | VOLUMES |
+------------+----------------+--------------------------------------
+--------+---------+
| SanBackend | ontap-san      | 586b1cd5-8cf8-428d-a76c-2872713612c1 |
online |       9 |
+------------+----------------+--------------------------------------
+--------+---------+
```

轮换密码时，存储管理员必须首先更新ONTAP上用户的密码。接下来将进行后端更新。轮换证书时，可以为用户添加多个证书。然后更新后端以使用新证书，之后即可从ONTAP集群中删除旧证书。

更新后端不会中断对已创建卷的访问，也不会影响之后建立的卷连接。后端更新成功表明Trident可以与ONTAP后端通信并处理未来的卷操作。

为Trident创建自定义ONTAP角色

您可以创建一个具有最低权限的ONTAP集群角色，这样您就不必使用ONTAP管理员角色在Trident中执行操作。在Trident后端配置中包含用户名时， Trident将使用您创建的ONTAP集群角色来执行操作。

请参阅"Trident自定义角色生成器"有关创建Trident自定义角色的更多信息。

使用**ONTAP CLI**

1. 使用以下命令创建新角色：

```
security login role create <role_name\> -cmddirname "command" -access all
-vserver <svm_name\>
```

2. 为Trident用户创建用户名：

```
security login create -username <user_name\> -application ontapi
-authmethod <password\> -role <name_of_role_in_step_1\> -vserver
<svm_name\> -comment "user_description"
```

3. 将角色映射到用户：

```
security login modify username <user_name\> -vserver <svm_name\> -role
<role_name\> -application ontapi -application console -authmethod
<password\>
```

使用系统管理器

在ONTAP系统管理器中执行以下步骤：

1. 创建自定义角色：
   a. 要在集群级别创建自定义角色，请选择“集群 > 设置”。

      （或者）要在 SVM 级别创建自定义角色，请选择“存储”>“存储虚拟机”> `required SVM` > 设置 > 用户和角色*。

   b. 选择“用户和角色”旁边的箭头图标（→）。
   c. 在“角色”下选择“+添加”。
   d. 定义角色规则，然后点击“保存”。
2. 将角色映射到Trident用户： + 在“用户和角色”页面上执行以下步骤：
   a. 在“用户”下方选择“添加”图标 **+**。
   b. 选择所需的用户名，然后在“角色”下拉菜单中选择角色。
   c. 单击“保存”。

更多信息请参阅以下页面：

- "用于管理ONTAP的自定义角色"或者"定义自定义角色"
- "与角色和用户协作"

使用双向 **CHAP** 验证连接

Trident可以使用双向 CHAP 对 iSCSI 会话进行身份验证。 `ontap-san`和 `ontap-san-economy`司机。这需要启用 `useCHAP`在后端定义中添加选项。设置为 `true`Trident将 SVM 的默认发起程序安全配置为双向 CHAP，并从后端文件中设置用户名和密钥。 NetApp建议使用双向 CHAP 协议对连接进行身份验证。请参见以下示例配置

：

```
---
version: 1
storageDriverName: ontap-san
backendName: ontap_san_chap
managementLIF: 192.168.0.135
svm: ontap_iscsi_svm
useCHAP: true
username: vsadmin
password: password
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSd6cNwxyz
```

这 `useCHAP`该参数是一个布尔选项，只能配置一次。默认值为 false。一旦将其设置为 true，就无法再将其设置为 false。

此外 `useCHAP=true`， 这 `chapInitiatorSecret`， `chapTargetInitiatorSecret` ， `chapTargetUsername` ， 和 `chapUsername`字段必须包含在后端定义中。创建后端后，可以通过运行以下命令来更改密钥： `tridentctl update`。

工作原理

通过设置 `useCHAP`如果设置为 true，则存储管理员指示Trident在存储后端配置 CHAP。其中包括以下内容：

- 在SVM上设置CHAP：
  - 如果 SVM 的默认启动器安全类型为“无”（默认设置）*并且*卷中不存在任何预先存在的 LUN，Trident会将默认安全类型设置为“无”。 `CHAP`然后继续配置 CHAP 发起程序和目标用户名及密钥。
  - 如果 SVM 包含 LUN， Trident将不会在 SVM 上启用 CHAP。这样可以确保对 SVM 上已存在的 LUN 的访问不受限制。
- 配置 CHAP 发起程序和目标用户名和密钥；这些选项必须在后端配置中指定（如上所示）。

后端创建完成后， Trident会创建一个相应的实例。 `tridentbackend` CRD 并将 CHAP 密钥和用户名存储为 Kubernetes 密钥。 Trident在此后端创建的所有 PV 都将通过 CHAP 进行安装和连接。

轮换凭证并更新后端

您可以通过更新 CHAP 参数来更新 CHAP 凭据。 `backend.json`文件。这将需要更新 CHAP 密钥并使用 `tridentctl update`命令反映这些更改。

更新后端 CHAP 密钥时，必须使用 `tridentctl`更新后端。请勿使用ONTAP CLI 或ONTAP系统管理器更新存储集群上的凭据，因为Trident将无法检测到这些更改。

```
cat backend-san.json
{
    "version": 1,
    "storageDriverName": "ontap-san",
    "backendName": "ontap_san_chap",
    "managementLIF": "192.168.0.135",
    "svm": "ontap_iscsi_svm",
    "useCHAP": true,
    "username": "vsadmin",
    "password": "password",
    "chapInitiatorSecret": "cl9qxUpDaTeD",
    "chapTargetInitiatorSecret": "rqxigXgkeUpDaTeD",
    "chapTargetUsername": "iJF4heBRT0TCwxyz",
    "chapUsername": "uh2aNCLSd6cNwxyz",
}

./tridentctl update backend ontap_san_chap -f backend-san.json -n trident
+----------------+----------------+--------------------------------------
+--------+---------+
|   NAME         | STORAGE DRIVER |                 UUID                 |
STATE   | VOLUMES |
+----------------+----------------+--------------------------------------
+--------+---------+
| ontap_san_chap | ontap-san      | aa458f3b-ad2d-4378-8a33-1a472ffbeb5c |
online |       7 |
+----------------+----------------+--------------------------------------
+--------+---------+
```

现有连接将不受影响；如果Trident在 SVM 上更新凭据，则这些连接将继续保持活动状态。新连接使用更新后的凭据，现有连接将继续保持活动状态。断开并重新连接旧的PV将使它们使用更新后的凭据。

ONTAP SAN 配置选项和示例

了解如何在Trident安装中创建和使用ONTAP SAN 驱动程序。本节提供后端配置示例以及将后端映射到 StorageClasses 的详细信息。

"ASA r2 系统"与其他ONTAP系统（ASA、 AFF和FAS）在存储层的实现上有所不同。这些变化会影响某些参数的使用，如注释中所述。"了解更多关于ASA r2 系统与其他ONTAP系统之间的区别"。

只有 `ontap-san`ASA r2 系统支持驱动程序（支持 iSCSI 和 NVMe/TCP 协议）。

在Trident后端配置中，无需指定您的系统是ASA r2。当您选择 `ontap-san`作为 `storageDriverName`Trident可自动检测ASA r2 或传统的ONTAP系统。如下表所示，某些后端配置参数不适用于ASA r2 系统。

后端配置选项

请参阅下表了解后端配置选项：

参数	描述	默认
`version`		始终为 1
`storageDriverName`	存储驱动程序的名称	`ontap-san`或者 `ontap-san-economy`
`backendName`	自定义名称或存储后端	驱动程序名称 + "_" + dataLIF
`managementLIF`	集群或 SVM 管理 LIF 的 IP 地址。 可以指定一个完全限定域名（FQDN）。 如果Trident安装时使用了 IPv6 标志，则可以设置为使用 IPv6 地址。 IPv6 地址必须用方括号定义，例如：`[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]` 。 为了实现MetroCluster 的无缝切换，请参阅MetroCluster示例。 ⓘ 如果您使用的是“vsadmin”凭据，`managementLIF`必须是SVM的凭据；如果使用“admin”凭据，`managementLIF`必须是集群的那个。	"10.0.0.1", "[2001:1234:abcd::fefe]"
`dataLIF`	LIF协议的IP地址。如果Trident安装时使用了 IPv6 标志，则可以设置为使用 IPv6 地址。 IPv6 地址必须用方括号定义，例如：`[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]` 。 *请勿指定使用 iSCSI。*Trident的使用"ONTAP选择性 LUN 地图"发现建立多路径会话所需的 iSCSI LIF。如果出现以下情况，则会生成警告： `dataLIF`已明确定义。 *Metrocluster 除外。*查看MetroCluster示例。	由支持向量机导出
`svm`	要使用的存储虚拟机 *Metrocluster 除外。*查看MetroCluster示例。	如果是 SVM 则推导而来`managementLIF`已指定
`useCHAP`	使用 CHAP 对ONTAP SAN 驱动程序的 iSCSI 进行身份验证 [布尔值]。设置为 `true`让Trident配置并使用双向 CHAP 作为后端给定 SVM 的默认身份验证。参考 "准备配置后端ONTAP SAN 驱动程序" 了解详情。 不支持**FCP**或**NVMe/TCP**协议。	`false`
`chapInitiatorSecret`	CHAP 发起者密钥。如果是必填项 `useCHAP=true`	""
`labels`	要应用于卷的任意 JSON 格式标签集	""

参数	描述	默认
chapTargetInitiatorSecret	CHAP 目标发起者密钥。如果是必填项 useCHAP=true	""
chapUsername	入站用户名。如果是必填项 useCHAP=true	""
chapTargetUsername	目标用户名。如果是必填项 useCHAP=true	""
clientCertificate	客户端证书的 Base64 编码值。用于基于证书的身份验证	""
clientPrivateKey	客户端私钥的 Base64 编码值。用于基于证书的身份验证	""
trustedCACertificate	受信任 CA 证书的 Base64 编码值。可选。用于基于证书的身份验证。	""
username	与ONTAP集群通信所需的用户名。用于基于凭证的身份验证。有关 Active Directory 身份验证，请参阅 "使用 Active Directory 凭据向后端 SVM 验证Trident 的身份"。	""
password	与ONTAP集群通信所需的密码。用于基于凭证的身份验证。有关 Active Directory 身份验证，请参阅 "使用 Active Directory 凭据向后端 SVM 验证Trident 的身份"。	""
svm	使用的存储虚拟机	如果是 SVM 则推导而来 `managementLIF`已指定
storagePrefix	在 SVM 中配置新卷时使用的前缀。之后无法修改。要更新此参数，您需要创建一个新的后端。	trident
aggregate	用于配置的聚合（可选；如果设置，则必须分配给 SVM）。对于 `ontap-nas-flexgroup`驱动程序，此选项将被忽略。如果未分配，则可以使用任何可用的聚合来配置FlexGroup卷。 当 SVM 中的聚合数据更新时， Trident 会自动轮询 SVM 进行更新，而无需重启Trident控制器。当您在Trident中配置特定聚合以配置卷时，如果该聚合被重命名或移出 SVM，则在轮询 SVM 聚合时， Trident中的后端将变为失败状态。您必须将聚合更改为 SVM 上存在的聚合，或者将其完全删除，才能使后端恢复联机。 请勿指定用于**ASA r2** 系统。	""

参数	描述	默认
`limitAggregateUsage`	如果使用率超过此百分比，则配置失败。如果您使用的是Amazon FSx for NetApp ONTAP后端，请勿指定 `limitAggregateUsage`。提供的 `fsxadmin`和 `vsadmin`不包含检索汇总使用情况和使用Trident限制它所需的权限。 请勿指定用于**ASA r2** 系统。	（默认情况下不强制执行）
`limitVolumeSize`	如果请求的卷大小大于此值，则配置失败。同时限制其管理的 LUN 卷的最大大小。	（默认情况下不强制执行）
`lunsPerFlexvol`	每个 Flexvol 的最大 LUN 数量必须在 [50, 200] 范围内	`100`
`debugTraceFlags`	故障排除时要使用的调试标志。例如，{"api":false, "method":true} 除非您正在进行故障排除并且需要详细的日志转储，否则请勿使用此方法。	`null`
`useREST`	使用ONTAP REST API 的布尔参数。 `useREST`设置为 `true`Trident 使用ONTAP REST API 与后端通信；当设置为 `false`Trident使用 ONTAPI (ZAPI) 调用与后端通信。此功能需要ONTAP 9.11.1 及更高版本。此外，所使用的ONTAP登录角色必须具有访问权限。 `ontapi`应用。预定义项满足了这一点。 `vsadmin`和 `cluster-admin` 角色。从Trident 24.06 版本和ONTAP 9.15.1 或更高版本开始， `useREST`设置为 `true`默认；更改 `useREST`到 `false`使用 ONTAPI (ZAPI) 调用。 `useREST`完全符合 NVMe/TCP 标准。 NVMe 仅支持ONTAP REST API，不支持 ONTAPI (ZAPI)。 如果指定，则始终设置为 **`true`**适用于**ASA r2** 系统。	`true`适用于ONTAP 9.15.1 或更高版本，否则 `false`。
`sanType`	用于选择 `iscsi`对于 iSCSI， `nvme`适用于 NVMe/TCP 或 `fcp`用于光纤通道 (FC) 上的 SCSI。	`iscsi`如果为空

参数	描述	默认
`formatOptions`	使用 `formatOptions`为以下情况指定命令行参数 `mkfs`该命令将在每次格式化卷时应用。这样您就可以根据自己的喜好格式化音量。请确保指定与 mkfs 命令选项类似的 formatOptions，但不包括设备路径。例如：“-E nodiscard” 支持 `**ontap-san**`和 `**ontap-san-economy**`支持 **iSCSI** 协议的驱动程序。 此外，在使用 **iSCSI** 和 **NVMe/TCP** 协议时， **ASA r2** 系统也受支持。	
`limitVolumePoolSize`	在 ontap-san-economy 后端中使用 LUN 时可请求的最大FlexVol大小。	（默认情况下不强制执行）
`denyNewVolumePools`	限制 `ontap-san-economy`后端创建新的FlexVol卷来包含它们的 LUN。只有预先存在的 Flexvol 才能用于配置新的 PV。	

使用 **formatOptions** 的建议

Trident建议采用以下选项来加快格式化过程：

-E nodiscard:

- 保留，不要在执行 mkfs 时尝试丢弃块（最初丢弃块对固态设备和稀疏/精简配置存储很有用）。这取代了已弃用的选项“-K”，并且适用于所有文件系统（xfs、ext3 和 ext4）。

使用 **Active Directory** 凭据向后端 **SVM** 验证**Trident** 的身份

您可以配置Trident以使用 Active Directory (AD) 凭据对后端 SVM 进行身份验证。在 AD 帐户可以访问 SVM 之前，您必须配置 AD 域控制器对集群或 SVM 的访问权限。对于使用 AD 帐户进行集群管理，您必须创建域隧道。参考 "在ONTAP中配置 Active Directory 域控制器访问" 了解详情。

步骤

1. 为后端 SVM 配置域名系统 (DNS) 设置：

   ```
   vserver services dns create -vserver <svm_name> -dns-servers
   <dns_server_ip1>,<dns_server_ip2>
   ```

2. 运行以下命令在 Active Directory 中为 SVM 创建计算机帐户：

   ```
   vserver active-directory create -vserver DataSVM -account-name ADSERVER1
   -domain demo.netapp.com
   ```

3. 使用此命令创建 AD 用户或组来管理集群或 SVM

   ```
   security login create -vserver <svm_name> -user-or-group-name
   <ad_user_or_group> -application <application> -authentication-method domain
   -role vsadmin
   ```

4. 在Trident后端配置文件中，设置 `username` 和 `password` 参数分别为 AD 用户或组名称和密码。

卷配置的后端配置选项

您可以使用以下选项控制默认配置。 `defaults`配置部分。例如，请参见下面的配置示例。

参数	描述	默认
`spaceAllocation`	LUN 的空间分配	“true” 如果指定，则设置为 **`true`**适用于**ASA r2** 系统。
`spaceReserve`	空间预留模式；“无”（细）或“大量”（粗）。 设置为 **`none`**适用于**ASA r2** 系统。	“没有任何”
`snapshotPolicy`	要使用的快照策略。 设置为 **`none`**适用于**ASA r2** 系统。	“没有任何”
`qosPolicy`	要为创建的卷分配的 QoS 策略组。每个存储池/后端选择 qosPolicy 或 adaptiveQosPolicy 之一。将 QoS 策略组与Trident结合使用需要ONTAP 9.8 或更高版本。您应该使用非共享的 QoS 策略组，并确保该策略组单独应用于每个成员。共享的 QoS 策略组强制规定所有工作负载的总吞吐量上限。	""
`adaptiveQosPolicy`	要为创建的卷分配的自适应 QoS 策略组。每个存储池/后端选择 qosPolicy 或 adaptiveQosPolicy 之一	""
`snapshotReserve`	为快照预留的卷百分比。 请勿指定用于**ASA r2** 系统。	如果为“0”， `snapshotPolicy`为“无”，否则为“
`splitOnClone`	创建时将克隆体从其母体中分离出来	"false"
`encryption`	在新卷上启用NetApp卷加密 (NVE)；默认设置为 `false`。要使用此选项，必须在集群上获得 NVE 许可并启用 NVE。如果后端启用了 NAE，则在Trident中配置的任何卷都将启用 NAE。更多信息，请参阅：“Trident如何与 NVE 和 NAE 协同工作” 。	“false” 如果指定，则设置为 **`true`**适用于**ASA r2** 系统。
`luksEncryption`	启用LUKS加密。参考“使用 Linux 统一密钥设置 (LUKS)”。	设置为 **`false`**适用于**ASA r2** 系统。
`tieringPolicy`	分层策略使用“无” 请勿为**ASA r2** 系统指定 。	
`nameTemplate`	用于创建自定义卷名称的模板。	""

卷配置示例

以下是一个定义了默认值的示例：

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: trident_svm
username: admin
password: <password>
labels:
  k8scluster: dev2
  backend: dev2-sanbackend
storagePrefix: alternate-trident
debugTraceFlags:
  api: false
  method: true
defaults:
  spaceReserve: volume
  qosPolicy: standard
  spaceAllocation: 'false'
  snapshotPolicy: default
  snapshotReserve: '10'
```

对于使用以下方式创建的所有卷 `ontap-san`驱动程序Trident为FlexVol增加了 10% 的额外容量，以容纳 LUN 元数据。 LUN 将按照用户在 PVC 中请求的确切大小进行配置。 Trident使FlexVol增加 10%（在ONTAP中显示为可用尺寸）。用户现在将获得他们所申请的可用容量。此项更改还可以防止 LUN 在可用空间未完全利用之前变为只读。这不适用于 ontap-san-economy。

对于定义后端 `snapshotReserve`Trident计算体积大小的方法如下：

```
Total volume size = [(PVC requested size) / (1 - (snapshotReserve
percentage) / 100)] * 1.1
```

1.1 是Trident为容纳 LUN 元数据而额外添加到FlexVol 的10%。为了 `snapshotReserve`= 5%，PVC 请求 = 5 GiB，总体积大小为 5.79 GiB，可用大小为 5.5 GiB。这 `volume show`该命令应显示与此示例类似的结果：

```
Vserver   Volume       Aggregate    State      Type       Size  Available Used%
--------- ------------ ------------ ---------- ---- ---------- ---------- -----
          _pvc_89f1c156_3801_4de4_9f9d_034d54c395f4
                       online     RW         10GB     5.00GB    0%
          _pvc_e42ec6fe_3baa_4af6_996d_134adbbb8e6d
                       online     RW       5.79GB     5.50GB    0%
          _pvc_e8372153_9ad9_474a_951a_08ae15e1c0ba
                       online     RW          1GB    511.8MB    0%
3 entries were displayed.
```

目前，调整大小是将新计算方法应用于现有体积的唯一途径。

最小配置示例

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。

> 如果您在NetApp ONTAP上使用Amazon FSx和Trident， NetApp建议您为 LIF 指定 DNS 名称而不是 IP 地址。

ONTAP SAN 示例

这是使用以下方法的基本配置： `ontap-san`司机。

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
labels:
  k8scluster: test-cluster-1
  backend: testcluster1-sanbackend
username: vsadmin
password: <password>
```

MetroCluster示例

您可以配置后端，以避免在切换和切换回后端后手动更新后端定义。"SVM复制和恢复" 。

为了实现无缝切换和切换回，请指定 SVM `managementLIF`并省略 `svm`参数。例如：

```
version: 1
storageDriverName: ontap-san
managementLIF: 192.168.1.66
username: vsadmin
password: password
```

ONTAP SAN 经济示例

```
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
username: vsadmin
password: <password>
```

基于证书的身份验证示例

在这个基本配置示例中 `clientCertificate`， `clientPrivateKey`， 和 `trustedCACertificate`（如果使用受信任的 CA，则为可选） `backend.json`分别取客户端证书、私钥和受信任 CA 证书的 base64 编码值。

```
---
version: 1
storageDriverName: ontap-san
backendName: DefaultSANBackend
managementLIF: 10.0.0.1
svm: svm_iscsi
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSd6cNwxyz
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
```

双向 **CHAP** 示例

这些示例创建了一个后端。 `useCHAP`设置为 `true`。

ONTAP SAN CHAP 示例

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
labels:
  k8scluster: test-cluster-1
  backend: testcluster1-sanbackend
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSd6cNwxyz
username: vsadmin
password: <password>
```

ONTAP SAN 经济 **CHAP** 示例

```
---
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSd6cNwxyz
username: vsadmin
password: <password>
```

NVMe/TCP 示例

您的ONTAP后端必须配置有使用 NVMe 的 SVM。这是 NVMe/TCP 的基本后端配置。

```
---
version: 1
backendName: NVMeBackend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_nvme
username: vsadmin
password: password
sanType: nvme
useREST: true
```

SCSI over FC (FCP) 示例

您的ONTAP后端必须配置有 FC 的 SVM。这是 FC 的基本后端配置。

```
---
version: 1
backendName: fcp-backend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_fc
username: vsadmin
password: password
sanType: fcp
useREST: true
```

使用 **nameTemplate** 的后端配置示例

```
---
version: 1
storageDriverName: ontap-san
backendName: ontap-san-backend
managementLIF: <ip address>
svm: svm0
username: <admin>
password: <password>
defaults:
  nameTemplate:
"{{.volume.Name}}_{{.labels.cluster}}_{{.volume.Namespace}}_{{.vo\
    lume.RequestName}}"
labels:
  cluster: ClusterA
  PVC: "{{.volume.Namespace}}_{{.volume.RequestName}}"
```

ontap-san-economy 驱动程序的 **formatOptions** 示例

```
---
version: 1
storageDriverName: ontap-san-economy
managementLIF: ""
svm: svm1
username: ""
password: "!"
storagePrefix: whelk_
debugTraceFlags:
  method: true
  api: true
defaults:
  formatOptions: -E nodiscard
```

具有虚拟池的后端示例

在这些示例后端定义文件中，所有存储池都设置了特定的默认值，例如： `spaceReserve`没有，`spaceAllocation`为假，并且 `encryption`为假。虚拟池在存储部分中定义。

Trident在“备注”字段中设置配置标签。在FlexVol volume上设置注释。Trident 在配置时Trident虚拟池上存在的所有标签复制到存储卷。为了方便起见，存储管理员可以为每个虚拟池定义标签，并按标签对卷进行分组。

在这些示例中，一些存储池会设置自己的参数。 `spaceReserve` ， `spaceAllocation` ， 和 `encryption`有

些值会覆盖默认值，有些池会覆盖默认值。

ONTAP SAN 示例

```yaml
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSd6cNwxyz
username: vsadmin
password: <password>
defaults:
  spaceAllocation: "false"
  encryption: "false"
  qosPolicy: standard
labels:
  store: san_store
  kubernetes-cluster: prod-cluster-1
region: us_east_1
storage:
  - labels:
      protection: gold
      creditpoints: "40000"
    zone: us_east_1a
    defaults:
      spaceAllocation: "true"
      encryption: "true"
      adaptiveQosPolicy: adaptive-extreme
  - labels:
      protection: silver
      creditpoints: "20000"
    zone: us_east_1b
    defaults:
      spaceAllocation: "false"
      encryption: "true"
      qosPolicy: premium
  - labels:
      protection: bronze
      creditpoints: "5000"
    zone: us_east_1c
    defaults:
      spaceAllocation: "true"
      encryption: "false"
```

ONTAP SAN 经济示例

```
---
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSd6cNwxyz
username: vsadmin
password: <password>
defaults:
  spaceAllocation: "false"
  encryption: "false"
labels:
  store: san_economy_store
region: us_east_1
storage:
  - labels:
      app: oracledb
      cost: "30"
    zone: us_east_1a
    defaults:
      spaceAllocation: "true"
      encryption: "true"
  - labels:
      app: postgresdb
      cost: "20"
    zone: us_east_1b
    defaults:
      spaceAllocation: "false"
      encryption: "true"
  - labels:
      app: mysqldb
      cost: "10"
    zone: us_east_1c
    defaults:
      spaceAllocation: "true"
      encryption: "false"
  - labels:
      department: legal
      creditpoints: "5000"
```

```
    zone: us_east_1c
    defaults:
      spaceAllocation: "true"
      encryption: "false"
```

NVMe/TCP 示例

```
---
version: 1
storageDriverName: ontap-san
sanType: nvme
managementLIF: 10.0.0.1
svm: nvme_svm
username: vsadmin
password: <password>
useREST: true
defaults:
  spaceAllocation: "false"
  encryption: "true"
storage:
  - labels:
      app: testApp
      cost: "20"
    defaults:
      spaceAllocation: "false"
      encryption: "false"
```

将后端映射到存储类

以下 StorageClass 定义指的是[具有虚拟池的后端示例]。使用 `parameters.selector`字段中，每个 StorageClass 都会指出哪些虚拟池可用于托管卷。卷将具有所选虚拟池中定义的各个方面。

- 这 `protection-gold`StorageClass 将映射到第一个虚拟池。 `ontap-san`后端。这是唯一提供黄金级保护的泳池。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=gold"
  fsType: "ext4"
```

- 这 `protection-not-gold`StorageClass 将映射到第二个和第三个虚拟池。 `ontap-san`后端。除了黄金级别之外，只有这些金池提供其他级别的保护。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection!=gold"
  fsType: "ext4"
```

- 这 `app-mysqldb`StorageClass 将映射到第三个虚拟池 `ontap-san-economy`后端。这是唯一一个为mysqldb类型应用程序提供存储池配置的存储池。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"
```

- 这 `protection-silver-creditpoints-20k`StorageClass 将映射到第二个虚拟池 `ontap-san`后端。这是唯一提供银级保护和 20000 积分的彩池。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"
```

- 这 `creditpoints-5k`StorageClass 将映射到第三个虚拟池 `ontap-san`后端和第四个虚拟池 `ontap-san-economy`后端。这是唯一提供 5000 积分的彩池产品。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: csi.trident.netapp.io
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"
```

- 这 `my-test-app-sc`StorageClass 将映射到 `testAPP`虚拟池 `ontap-san`司机 `sanType: nvme`。这是唯一一家提供泳池的公司 `testApp`。

```
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: my-test-app-sc
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=testApp"
  fsType: "ext4"
```

Trident将决定选择哪个虚拟池，并确保满足存储需求。