ONTAP SAN 驱动程序

Trident

NetApp
July 01, 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 驱动程序详细信息

Trident 提供以下 SAN 存储驱动程序以与 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	**volumeMode**	支持的访问模式	支持的文件系统
`ontap-san`	iSCSI SCSI over FC	块	RWO、ROX、RWX、RWOP	无文件系统；原始块设备
`ontap-san`	iSCSI SCSI over FC	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 请参见 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 请参见 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`

- 仅当预期持久卷使用次数高于"支持的 ONTAP 卷限制"时，才使用 `ontap-san-economy`。
- 仅当预期持久卷使用次数高于"支持的 ONTAP 卷限制"且无法使用 `ontap-san-economy`驱动程序时，才使用 `ontap-nas-economy`。
- 如果您预计需要数据保护、灾难恢复或移动性，请勿使用 `ontap-nas-economy`。
- NetApp 不建议在所有 ONTAP 驱动程序中使用 Flexvol 自动增长，除了 ontap-san。作为一种解决方法，Trident 支持使用快照保留并相应地扩展 Flexvol 卷。

用户权限

Trident 希望以 ONTAP 或 SVM 管理员的身份运行，通常使用 `admin`集群用户或 `vsadmin`SVM 用户，或具有相同角色的不同名称的用户。对于 Amazon FSx for NetApp ONTAP 部署，Trident 希望以 ONTAP 或 SVM 管理员的身份运行，使用集群 `fsxadmin`用户或 `vsadmin`SVM 用户，或具有相同角色的不同名称的用户。`fsxadmin`用户是集群管理员用户的有限替代品。

如果使用 `limitAggregateUsage` 参数，则需要群集管理员权限。将 Amazon FSx for NetApp ONTAP 与 Trident 结合使用时， `limitAggregateUsage` 参数不适用于 `vsadmin` 和 `fsxadmin` 用户帐户。如果指定此参数，配置操作将失败。

虽然可以在 ONTAP 中创建 Trident 驱动程序可以使用的更具限制性的角色，但我们不建议这样做。大多数新版本的 Trident 会调用必须考虑到的其他 API，这使得升级变得困难且容易出错。

NVMe/TCP 的其他注意事项

Trident 支持非易失性存储器 Express (NVMe) 协议，使用 `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 之外创建的 NVMe 卷，以便 Trident 可以管理其生命周期
- NVMe 原生多路径
- K8s 节点的优雅或不优雅关闭（24.06）

Trident 不支持：

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

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

准备使用 ONTAP SAN 驱动程序配置后端

了解使用 ONTAP SAN 驱动程序配置 ONTAP 后端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" 不同于其他 ONTAP 系统（ASA、AFF 和 FAS）的存储层实现。在 ASA r2 系统中，使用存储可用区而不是聚合。请参阅 "此" 知识库文章，了解如何在 ASA r2 系统中为 SVM 分配聚合。

请记住，您还可以运行多个驱动程序，并创建指向一个或另一个的存储类。例如，您可以配置一个 `san-dev`类，该类使用 `ontap-san`驱动程序，以及一个 `san-default`类，该类使用 `ontap-san-economy`驱动程序。

所有 Kubernetes 工作节点都必须安装相应的 iSCSI 工具。有关详细信息，请参见 "准备工作节点"。

对 **ONTAP** 后端进行身份验证

Trident 提供两种身份验证 ONTAP 后端的模式。

- 基于凭据：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 `vsadmin`以确保与 ONTAP 版本的最大兼容性。
- 基于证书：Trident 还可以使用后端安装的证书与 ONTAP 集群进行通信。此处，后端定义必须包含客户端证书、密钥和可信 CA 证书的 Base64 编码值（如果使用）（推荐）。

您可以更新现有后端以在基于凭据和基于证书的方法之间移动。但是，一次仅支持一种身份验证方法。要切换到其他身份验证方法，必须从后端配置中删除现有方法。

如果您尝试提供*凭据和证书*，则后端创建将失败，错误为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 需要向 SVM 范围/集群范围的管理员提供凭据，以便与 ONTAP 后端进行通信。建议使用标准、预定义的角色，如 `admin`或 `vsadmin`。这确保了与未来 ONTAP 版本的向前兼容性，这些版本可能会公开未来 Trident 版本使用的功能 API。可以创建自定义安全登录角色并与 Trident 一起使用，但不建议这样做。

示例后端定义如下所示：

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"
}
```

请记住，后端定义是凭据以纯文本形式存储的唯一位置。后端创建后，用户名/密码使用 Base64 进行编码，并存储为 Kubernetes 密码。创建或更新后端是唯一需要了解凭据的步骤。因此，它是一个仅限管理员的操作，由 Kubernetes/存储管理员执行。

启用基于证书的身份验证

新的和现有的后端可以使用证书并与 ONTAP 后端通信。后端定义中需要三个参数。

- clientCertificate: 客户端证书的 Base64 编码值。
- clientPrivateKey: 关联专用密钥的 Base64 编码值。
- trustedCACertificate：受信任的 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。如果未使用受信任的 CA，则可以忽略此设置。

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，将公用名 (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. 将受信任的 CA 证书添加到 ONTAP 集群。这可能已由存储管理员处理。如果未使用受信任的 CA，则忽略。

```
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
```

> 运行此命令后，ONTAP 提示输入证书。粘贴步骤 1 中生成的 `k8senv.pem`文件内容，然后输入 `END`以完成安装。

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. 使用生成的证书测试身份验证。将 <ONTAP Management LIF> 和 <vserver name> 替换为管理 LIF IP 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、密钥和可信 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. 使用从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。这可以双向工作：可以将使用用户名/密码的后端更新为使用证书；可以将使用证书的后端更新为基于用户名/密码。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。然后使用更新的 backend.json 文件，其中包含执行 `tridentctl backend update` 所需的参数。

```
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。然后进行后端更新。轮换证书时，可以向用户添加多个证书。然后更新后端以使用新证书，之后可以从 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。成功的后端更新表明，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

您可以使用最低权限创建 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 使用您创建的 ONTAP 集群角色来执行操作。

有关创建 Trident 自定义角色的详细信息，请参见 "Trident 自定义角色生成器"。

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 **System Manager**

在 ONTAP System Manager 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 **Cluster > Settings**。

      （或）要在 SVM 级别创建自定义角色，请选择*存储 > Storage VM > `required SVM`> 设置 > 用户和角色*。

   b. 选择 **Users and Roles** 旁边的箭头图标 (→)。
   c. 在 **Roles** 下选择 **+Add**。
   d. 定义角色的规则并单击 **Save**。
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. 选择 **Users** 下的添加图标 **+**。
   b. 选择所需的用户名，然后在 **Role** 下拉菜单中选择一个角色。
   c. 单击 **Save**。

有关详细信息，请参见以下页面：

- "用于管理 ONTAP 的自定义角色" 或 "定义自定义角色"
- "使用角色和用户"

使用双向 **CHAP** 验证连接

Trident 可以使用 `ontap-san`和 `ontap-san-economy`驱动程序的双向 CHAP 对 iSCSI 会话进行身份验证。这需要在后端定义中启用 `useCHAP`选项。当设置为 `true`时，Trident 将 SVM 的默认启动器安全配置为双向 CHAP，并从后端文件设置用户名和密码。NetApp 建议使用双向 CHAP 来验证连接。请参见以下配置示例：

```
---
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
```

useCHAP 参数是一个布尔选项，只能配置一次。默认设置为 false。将其设置为 true 后，无法将其设置为 false。

除了 `useCHAP=true`之外，后端定义中还必须包含 `chapInitiatorSecret`、`chapTargetInitiatorSecret`、`chapTargetUsername`和 `chapUsername`字段。创建后端后，可以通过运行 `tridentctl update`来更改密钥。

工作原理

通过设置 `useCHAP`为 true，存储管理员指示 Trident 在存储后端上配置 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 none（默认设置）*和*卷中没有预先存在的 LUN，Trident 会将默认安全类型设置为 `CHAP`并继续配置 CHAP 启动器以及目标用户名和密码。
  - 如果 SVM 包含 LUN，Trident 将不会在 SVM 上启用 CHAP。这可确保对 SVM 上已存在的 LUN 的访问不受限制。
- 配置 CHAP 启动器以及目标用户名和密码；这些选项必须在后端配置中指定（如上所示）。

创建后端后，Trident 创建相应的 `tridentbackend` CRD，并将 CHAP secrets 和用户名存储为 Kubernetes secrets。由 Trident 在此后端上创建的所有 PV 都将通过 CHAP 进行挂载和连接。

轮换凭据并更新后端

您可以通过更新 `backend.json` 文件中的 CHAP 参数来更新 CHAP 凭据。这将需要更新 CHAP 密码并使用 `tridentctl update` 命令来反映这些更改。

更新后端的 CHAP 密码时，必须使用 `tridentctl`来更新后端。请勿使用 ONTAP CLI 或 ONTAP System Manager 更新存储集群上的凭据，因为 Trident 将无法获取这些更改。

```
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 上更新了凭据，它们将继续保持活动状态。新连接使用更新的凭据，现有连接继续保持活动状态。断开和重新连接旧的 PV 将导致它们使用更新的凭据。

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。本节提供了将后端映射到 StorageClasses 的后端配置示例和详细信息。"ASA r2 系统" 不同于其他 ONTAP 系统（ASA、AFF 和 FAS）的存储层实现。这些变化会影响某些标注参数的使用。"详细了解 ASA r2 系统与其他 ONTAP 系统之间的差异".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`时，Trident 会自动检测 ASA r2 或其他 ONTAP 系统。某些后端配置参数不适用于 ASA r2 系统，如下表所示。

ASA r2 系统仅支持 `ontap-san` 驱动程序（具有 iSCSI、NVMe/TCP 和 FC 协议）。

后端配置选项

有关后端配置选项，请参见下表：

参数	说明	默认
`version`		始终为 1
`storageDriverName`	存储驱动程序的名称	`ontap-san` 或 `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。 可以指定完全限定的域名 (FQDN)。 如果使用 IPv6 标志安装了 Trident，则可以设置为使用 IPv6 地址。IPv6 地址必须在方括号中定义，例如 `[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]`。 有关无缝 MetroCluster 切换，请参见 MetroCluster 示例。 ⓘ 如果使用 "vsadmin" 凭据，`managementLIF` 必须是 SVM 的凭据；如果使用 "admin" 凭据，`managementLIF` 必须是集群的凭据。	“10.0.0.1”，“[2001:1234:abcd::fefe]”
`dataLIF`	协议 LIF 的 IP 地址。如果使用 IPv6 标志安装了 Trident，则可以设置为使用 IPv6 地址。IPv6 地址必须在方括号中定义，例如 `[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]`。*不为 iSCSI 指定。*Trident 使用"ONTAP 选择性 LUN 映射"来发现建立多路径会话所需的 iSCSI LIF。如果明确定义了 `dataLIF`，则会生成警告。*省略 MetroCluster。*请参阅MetroCluster 示例。	由 SVM 派生
`svm`	要使用的 Storage Virtual Machine *对于 MetroCluster 请省略。*请参阅MetroCluster 示例。	如果指定了 SVM `managementLIF`，则派生
`useCHAP`	使用 CHAP 对 ONTAP SAN 驱动程序的 iSCSI 进行身份验证 [Boolean]。设置为 `true`，Trident 将配置和使用双向 CHAP 作为后端给定的 SVM 的默认身份验证。有关详细信息，请参见 "准备使用 ONTAP SAN 驱动程序配置后端"。不支持 **FCP** 或 **NVMe/TCP**。	`false`
`chapInitiatorSecret`	CHAP 启动器密钥。如果 `useCHAP=true`为必需	""
`labels`	要应用于卷的任意 JSON 格式标签集	""

参数	说明	默认
`chapTargetInitiatorSecret`	CHAP 目标发起者密钥。如果 `useCHAP=true`为必需	""
`chapUsername`	入站用户名。如果 `useCHAP=true`为必需	""
`chapTargetUsername`	目标用户名。如果 `useCHAP=true`为必需	""
`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。用于基于凭据的身份验证。有关 Active Directory 身份验证，请参阅"使用 Active Directory 凭据向后端 SVM 验证 Trident"。	""
`password`	与 ONTAP 集群通信所需的密码。用于基于凭据的身份验证。有关 Active Directory 身份验证，请参阅"使用 Active Directory 凭据向后端 SVM 验证 Trident"。	""
`svm`	要使用的 Storage Virtual Machine	如果指定了 SVM `managementLIF`，则派生
`storagePrefix`	在 SVM 中配置新卷时使用的前缀。以后无法修改。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。 当聚合在 SVM 中更新时，它会通过轮询 SVM 在 Trident 中自动更新，而无需重新启动 Trident Controller。当您在 Trident 中配置了特定聚合来配置卷时，如果聚合被重命名或移出 SVM，则在轮询 SVM 聚合时，后端将在 Trident 中移至失败状态。您必须将聚合更改为存在于 SVM 上的聚合，或者将其完全删除，以使后端恢复联机。 请勿为 **ASA r2** 系统指定。	""
`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，请不要指定 `limitAggregateUsage`。提供的 `fsxadmin`和 `vsadmin`不包含检索聚合使用情况并使用 Trident 限制它所需的权限。请勿为 **ASA r2** 系统指定。	"（默认情况下不强制执行）

参数	说明	默认
limitVolumeSize	如果请求的卷大小高于此值，则设置失败。还限制它为 LUN 管理的卷的最大大小。	"（默认情况下不强制执行）
lunsPerFlexvol	每个 FlexVol 的最大 LUN 数，必须在 [50, 200] 范围内	100
debugTraceFlags	故障排除时使用的调试标志。例如，{"api":false, "method":true} 除非正在进行故障排除并需要详细的日志转储，否则不要使用。	null
useREST	使用 ONTAP REST API 的布尔参数。 `useREST`设置为 `true`时，Trident 使用 ONTAP REST API 与后端通信；设置为 `false`时，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 登录角色必须能够访问 `ontapi`应用程序。这通过预定义的 `vsadmin`和 `cluster-admin`角色来满足。从 Trident 24.06 版本和 ONTAP 9.15.1 或更高版本开始，`useREST`默认设置为 `true`；将 `useREST`更改为 `false`以使用 ONTAPI (ZAPI) 调用。 useREST 完全符合 NVMe/TCP。 仅 ONTAP REST API 支持 NVMe，ONTAPI (ZAPI) 不支持 NVMe。 如果指定，则对于 **ASA r2** 系统始终设置为 **true**。	true 适用于 ONTAP 9.15.1 或更高版本，否则 false。
sanType	用于选择 iscsi iSCSI、 nvme NVMe/TCP 或 fcp 光纤通道 (FC) 上的 SCSI。	iscsi 如果为空
formatOptions	使用 formatOptions 为 mkfs 命令指定命令行参数，该参数将在卷格式化时应用。这允许您根据您的偏好格式化卷。请务必指定与 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** 驱动程序的 **iSCSI** 协议。*此外，使用 iSCSI 和 NVMe/TCP 协议时，支持 ASA r2 系统。*	
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）

参数	说明	默认
`denyNewVolumePools`	限制 `ontap-san-economy`后端创建包含其 LUN 的新 FlexVol 卷。仅预先存在的 FlexVol 用于配置新的 PV。	

有关使用 **formatOptions** 的建议

Trident 建议使用以下选项来加快格式化过程：

- **-E nodiscard (ext3, ext4):** 不要尝试在 mkfs 时间丢弃块（丢弃块最初在固态设备和稀疏/精简配置的存储上很有用）。这将替换已弃用的选项 "-K"，并且适用于 ext3、ext4 文件系统。
- **-K (xfs):** 不要尝试在 mkfs 时间丢弃块。此选项适用于 xfs 文件系统。

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

您可以配置 Trident 使用 Active Directory (AD) 凭据向后端 SVM 进行身份验证。在 AD 帐户可以访问 SVM 之前，必须配置 AD 域控制器对集群或 SVM 的访问。对于使用 AD 帐户的集群管理，必须创建域隧道。有关详细信息，请参见 "在 ONTAP 中配置 Active Directory 域控制器访问"。

步骤

1. 配置后端 SVM 的域名系统 (DNS) 设置：

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

2. 运行以下命令为 Active Directory 中的 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 用户或组以管理集群或 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，将 `username` 和 `password` 参数分别设置为 AD 用户或组名称和密码。

用于配置卷的后端配置选项

您可以使用配置的 `defaults` 部分中的这些选项来控制默认配置。有关示例，请参阅下面的配置示例。

参数	说明	默认
`spaceAllocation`	LUN 的空间分配	"true" 如果指定，则对于 **ASA r2** 系统设置为 **`true`**。
`spaceReserve`	空间预留模式；"none"（精简）或"volume"（厚）。对于 **ASA r2** 系统，设置为 **`none`**。	"无"
`snapshotPolicy`	要使用的 Snapshot 策略。对于 **ASA r2** 系统设置为 **`none`**。	"无"

参数	说明	默认
`qosPolicy`	要为创建的卷分配的 QoS 策略组。为每个存储池/后端选择 qosPolicy 或 adaptiveQosPolicy 中的一个。在 Trident 中使用 QoS 策略组需要 ONTAP 9.8 或更高版本。您应该使用非共享 QoS 策略组，并确保该策略组单独应用于每个组成部分。共享 QoS 策略组强制执行所有工作负载总吞吐量的上限。	""
`adaptiveQosPolicy`	要为创建的卷分配的自适应 QoS 策略组。为每个存储池/后端选择 qosPolicy 或 adaptiveQosPolicy 中的一个	""
`snapshotReserve`	为快照保留的卷的百分比。请勿为 **ASA r2** 系统指定。	"0" 如果 `snapshotPolicy` 为 "none"，否则为 "
`splitOnClone`	创建时从其父级拆分克隆	"false"
`encryption`	在新卷上启用 NetApp Volume Encryption (NVE)；默认为 `false`。必须在群集上许可并启用 NVE 才能使用此选项。如果在后端启用了 NAE，则在 Trident 中配置的任何卷都将启用 NAE。有关更多信息，请参阅：“Trident 如何与 NVE 和 NAE 配合使用”。	"false" 如果指定，则对于 **ASA r2** 系统设置为 **`true`**。
`luksEncryption`	启用 LUKS 加密。请参见 “使用 Linux Unified Key Setup (LUKS)”。	" 对于 **ASA r2** 系统，设置为 **`false`**。
`tieringPolicy`	使用 "none" 的分层策略 不要为 **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 `ontap-san` 驱动程序创建的所有卷，Trident 会为 FlexVol 增加 10% 的额外容量以容纳 LUN 元数据。LUN 将使用用户在 PVC 中请求的确切大小进行配置。Trident 为 FlexVol 增加 10%（在 ONTAP 中显示为可用大小）。用户现在将获得他们请求的可用容量。此更改还可防止 LUN 变为只读，除非可用空间得到充分利用。这不适用于 ontap-san-economy。

对于定义 `snapshotReserve` 的后端，Trident 计算卷的大小如下：

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

1.1 是 Trident 添加到 FlexVol 以容纳 LUN 元数据的额外 10%。对于 `snapshotReserve` = 5%，且 PVC 请求 = 5 GiB，总体积大小为 5.79 GiB，可用大小为 5.5 GiB。 `volume show`命令应显示类似于以下示例的结果：

当前，对现有卷使用新计算的唯一方法是调整大小。

最小配置示例

以下示例显示了将大多数参数保留为默认值的基本配置。这是定义后端的最简单方法。

> 如果您在 Trident 中使用 Amazon FSx for NetApp ONTAP，NetApp 建议为 LIF 指定 DNS 名称而不是 IP 地址。

ONTAP SAN 示例

这是使用 `ontap-san` 驱动程序的基本配置。

```
---
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 "SVM 复制和恢复" 期间进行切换和切换后手动更新后端定义。

对于无缝切换和切回，使用 `managementLIF` 指定 SVM 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

在此基本配置示例中， `clientCertificate`、 `clientPrivateKey` 和 `trustedCACertificate`（可选，如果使用受信任的 CA）填充在 `backend.json` 中，并分别采用客户端证书、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

这些示例创建一个后端，其中 `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 economy 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

必须在 ONTAP 后端上配置具有 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
```

基于 FC 的 SCSI (FCP) 示例

您必须在 ONTAP 后端上配置带有 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
```

带有 **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}}"
```

formatOptions 示例，适用于 **ontap-san-economy** 驱动程序

```
---
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，为所有存储池设置了特定的默认值，例如 `spaceReserve` 为 none、`spaceAllocation` 为 false 和 `encryption` 为 false。虚拟池在存储部分中定义。

Trident 在"Comments"字段中设置配置标签。注释在 FlexVol 卷上设置，Trident 在配置时将虚拟池中存在的所有标签复制到存储卷。为方便起见，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，一些存储池设置了自己的 `spaceReserve`、 `spaceAllocation` 和 `encryption` 值，而一些

池覆盖了默认值。

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 StorageClasses

以下 StorageClass 定义请参阅 [具有虚拟池的后端示例]。使用 `parameters.selector` 字段，每个 StorageClass 调用哪些虚拟池可用于托管卷。卷将具有所选虚拟池中定义的方面。

- `protection-gold` StorageClass 将映射到 `ontap-san`后端中的第一个虚拟池。这是唯一提供黄金级保护的池。

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

- `protection-not-gold` StorageClass 将映射到 `ontap-san`后端的第二个和第三个虚拟池。这些是唯一提供黄金以外保护级别的池。

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

- `app-mysqldb` StorageClass 将映射到 `ontap-san-economy`后端的第三个虚拟池。这是唯一为 mysqldb 类型应用程序提供存储池配置的池。

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

- `protection-silver-creditpoints-20k` StorageClass 将映射到 `ontap-san` 后端的第二个虚拟池。这是唯一提供银级保护和 20000 creditpoints 的池。

```
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"
```

- `creditpoints-5k` StorageClass 将映射到 `ontap-san`后端的第三个虚拟池和 `ontap-san-economy`后端的第四个虚拟池。这些是唯一拥有 5000 个信用点的池产品。

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

- 该 `my-test-app-sc` StorageClass 将映射到 `testAPP` 驱动程序中的 `ontap-san` 虚拟池，并带有 `sanType: nvme`。这是唯一提供 `testApp` 的池。

```
---
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，并确保满足存储要求。