



ONTAP SAN ドライバ

Trident

NetApp
February 20, 2026

This PDF was generated from <https://docs.netapp.com/ja-jp/trident/trident-use/ontap-san.html> on February 20, 2026. Always check docs.netapp.com for the latest.

目次

ONTAP SAN ドライバ	1
ONTAP SAN ドライバの概要	1
ONTAP SAN ドライバの詳細	1
ユーザ権限	2
NVMe/TCPに関するその他の考慮事項	2
バックエンドにONTAP SAN ドライバを設定する準備をします	3
要件	3
ONTAP バックエンドの認証	3
双向CHAPによる接続の認証	9
ONTAP のSAN構成オプションと例	10
バックエンド構成オプション	11
ボリュームのプロビジョニング用のバックエンド構成オプション	17
最小限の設定例	19
仮想プールを使用するバックエンドの例	24
バックエンドを StorageClasses にマッピングします	29

ONTAP SAN ドライバ

ONTAP SAN ドライバの概要

ONTAP および Cloud Volumes ONTAP SAN ドライバを使用した ONTAP バックエンドの設定について説明します。

ONTAP SAN ドライバの詳細

Tridentは、ONTAPクラスタと通信するための次のSANストレージドライバを提供します。サポートされているアクセスモードは、*ReadWriteOnce*(RWO)、*ReadOnlyMany*(ROX)、*ReadWriteMany*(RWX)、*ReadWriteOncePod*(RWOP)です。

ドライバ	プロトコル	ボリュームモード	サポートされているアクセスモード	サポートされるファイルシステム
「ontap - san」	iSCSI SCSI over FC	ブロック	RWO、ROX、RWX、RWOP	ファイルシステムなし。rawブロックデバイスです
「ontap - san」	iSCSI SCSI over FC	ファイルシステム	RWO、RWOP ROXおよびRWXは、ファイルシステムボリュームモードでは使用できません。	「xfs」、「ext3」、「ext4」
「ontap - san」	NVMe/FC を参照してください NVMe/TCP に関するその他の考慮事項。	ブロック	RWO、ROX、RWX、RWOP	ファイルシステムなし。rawブロックデバイスです
「ontap - san」	NVMe/FC を参照してください NVMe/TCP に関するその他の考慮事項。	ファイルシステム	RWO、RWOP ROXおよびRWXは、ファイルシステムボリュームモードでは使用できません。	「xfs」、「ext3」、「ext4」
「ONTAP - SAN - エコノミー」	iSCSI	ブロック	RWO、ROX、RWX、RWOP	ファイルシステムなし。rawブロックデバイスです

ドライバ	プロトコル	ボリュームモード	サポートされているアクセスモード	サポートされるファイルシステム
「ONTAP - SAN - エコノミー」	iSCSI	ファイルシステム	RWO、RWOP ROXおよびRWXは、ファイルシステムボリュームモードでは使用できません。	「xfs」、「ext3」、「ext4」

- ! • 使用 `ontap-san-economy` 永続的ボリュームの使用数が次の値よりも多いと予想される場合のみ "サポートされるONTAPの制限"。
- ! • 使用 `ontap-nas-economy` 永続的ボリュームの使用数が次の値よりも多いと予想される場合のみ "サポートされるONTAPの制限" および `ontap-san-economy` ドライバは使用できません。
- ! • 使用しないでください `ontap-nas-economy` データ保護、ディザスタリカバリ、モビリティのニーズが予想される場合。
- ! • NetAppでは、ONTAP SANを除くすべてのONTAPドライバでFlexVol自動拡張を使用することは推奨されていません。回避策として、Tridentはスナップショット予約の使用をサポートし、それに応じてFlexVolボリュームを拡張します。

ユーザ権限

Tridentは、ONTAP管理者またはSVM管理者（通常はクラスタユーザ、`vsadmin`SVM`ユーザ、または別の名前で同じロールのユーザを使用）として実行することを想定しています ``admin`。Amazon FSx for NetApp ONTAP環境では、Tridentは、クラスタユーザまたは`vsadmin`SVM`ユーザを使用するONTAP管理者またはSVM管理者、または同じロールの別の名前のユーザとして実行される必要があります ``fsxadmin`。この``fsxadmin``ユーザは、クラスタ管理者ユーザに代わる限定的なユーザです。

! パラメータを使用する場合は `limitAggregateUsage`、クラスタ管理者の権限が必要です。TridentでAmazon FSx for NetApp ONTAPを使用している場合、`limitAggregateUsage``パラメータはユーザアカウントと ``fsxadmin``ユーザアカウントでは機能しません ``vsadmin`。このパラメータを指定すると設定処理は失敗します。

ONTAP内でTridentドライバが使用できる、より制限の厳しいロールを作成することは可能ですが、推奨しません。Tridentの新リリースでは、多くの場合、考慮すべきAPIが追加で必要になるため、アップグレードが難しく、エラーも起こりやすくなります。

NVMe/TCPに関するその他の考慮事項

Tridentは、次のドライバを使用してNon-Volatile Memory Express (NVMe) プロトコルをサポートします `ontap-san`。

- IPv6
- NVMeボリュームのSnapshotとクローン
- NVMeボリュームのサイズ変更
- Tridentの外部で作成されたNVMeボリュームをインポートして、そのライフサイクルをTridentで管理でき

るようになります

- NVMeネイティブマルチパス
- Kubernetesノードのグレースフルシャットダウンまたはグレースフルシャットダウン (24.06)

Tridentは以下をサポートしていません。

- NVMeでネイティブにサポートされているDH-HMAC-CHAP
- Device Mapper (DM ; デバイスマッパー) マルチパス
- LUKS暗号化



NVMe はONTAP REST API でのみサポートされ、ONTAPI (ZAPI) ではサポートされません。

バックエンドにONTAP SAN ドライバを設定する準備をします

ONTAP SAN ドライバでONTAPバックエンドを構成するための要件と認証オプションを理解します。

要件

すべてのONTAP バックエンドでは、Trident では少なくとも 1 つのアグリゲートを SVM に割り当てる必要があります。



"ASA r2システム"ストレージ層の実装において他のONTAPシステム (ASA、 AFF、 FAS) と異なります。ASA r2 システムでは、集約の代わりにストレージ可用性ゾーンが使用されます。参照["これ"ASA r2 システムで SVM にアグリゲートを割り当てる方法に関するナレッジベースの記事。](#)

複数のドライバを実行し、1つまたは複数のドライバを参照するストレージクラスを作成することもできます。たとえば 'ONTAP-SAN' ドライバを使用する「-dev」クラスと 'ONTAP-SAN-エコノミー'one を使用する「デフォルト」クラスを設定できます

すべてのKubernetesワーカーノードに適切なiSCSIツールをインストールしておく必要があります。を参照してください ["ワーカーノードを準備します"](#) を参照してください。

ONTAPバックエンドの認証

Tridentには、ONTAPバックエンドの認証に2つのモードがあります。

- credential based : 必要な権限を持つONTAP ユーザのユーザ名とパスワード。ONTAP バージョンとの互換性を最大限に高めるために 'admin' または vsadmin などの事前定義されたセキュリティ・ログイン・ロールを使用することを推奨します
- 証明書ベース : Tridentは、バックエンドにインストールされている証明書を使用してONTAPクラスタと通信することもできます。この場合、バックエンド定義には、Base64 でエンコードされたクライアント証明書、キー、および信頼された CA 証明書（推奨）が含まれている必要があります。

既存のバックエンドを更新して、クレデンシャルベースの方式と証明書ベースの方式を切り替えることができます。ただし、一度にサポートされる認証方法は1つだけです。別の認証方式に切り替えるには、バックエン

ド設定から既存の方式を削除する必要があります。



クレデンシャルと証明書の両方を*指定しようとすると、バックエンドの作成が失敗し、構成ファイルに複数の認証方法が指定されているというエラーが表示されます。

クレデンシャルベースの認証を有効にします

TridentがONTAPバックエンドと通信するには、SVMを対象としたクラスタを対象とした管理者に対するクレデンシャルが必要です。や vsadmin、などの事前定義された標準のロールを使用することを推奨します`admin。これにより、今後のONTAPリリースで使用する機能APIが公開される可能性がある将来のTridentリリースとの前方互換性が確保されます。Tridentでは、カスタムのセキュリティログインロールを作成して使用できますが、推奨されません。

バックエンド定義の例は次のようにになります。

YAML

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

JSON

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

バックエンド定義は、クレデンシャルがプレーンテキストで保存される唯一の場所であることに注意してください。バックエンドが作成されると、ユーザ名とパスワードが Base64 でエンコードされ、Kubernetes シークレットとして格納されます。クレデンシャルの知識が必要なのは、バックエンドの作成または更新だけです。この処理は管理者専用で、Kubernetes / ストレージ管理者が実行します。

証明書ベースの認証の有効化

新規または既存のバックエンドは証明書を使用して ONTAP バックエンドと通信できます。バックエンド定義

には3つのパラメータが必要です。

- clientCertificate : Base64でエンコードされたクライアント証明書の値。
- clientPrivateKey : Base64でエンコードされた、関連付けられた秘密鍵の値。
- trustedCACertificate: 信頼されたCA証明書のBase64エンコード値。信頼されたCAを使用する場合は、このパラメータを指定する必要があります。信頼されたCAが使用されていない場合は無視してかまいません。

一般的なワークフローは次の手順で構成されます。

手順

1. クライアント証明書とキーを生成します。生成時に、ONTAPユーザとして認証するようにCommon Name (CN; 共通名)を設定します。

```
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. 信頼されたCA証明書をONTAPクラスタに追加します。この処理は、ストレージ管理者がすでに行っている可能性があります。信頼できるCAが使用されていない場合は無視します。

```
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. ONTAPクラスタにクライアント証明書とキーをインストールします(手順1)。

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



このコマンドを実行すると、ONTAPは証明書の入力を求めます。手順1で生成された`k8senv.pem`ファイルの内容を貼り付け、`END`を入力してインストールを完了します。

4. ONTAPセキュリティ・ログイン・ロールが'cert'認証方式をサポートしていることを確認します

```
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. 生成された証明書を使用して認証をテストONTAP管理LIF >と<vserver name>は、管理LIFのIPアド

レスおよび SVM 名に置き換えてください。

```
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. Base64 で証明書、キー、および信頼された CA 証明書をエンコードする。

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

7. 前の手順で得た値を使用してバックエンドを作成します。

```
cat cert-backend.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "SanBackend",
  "managementLIF": "1.2.3.4",
  "svm": "vserver_test",
  "clientCertificate": "Faaaakkkeeee...Vaaalllluuuuueeee",
  "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  |
+-----+-----+
+-----+-----+
```

認証方法を更新するか、クレデンシャルをローテーションして

既存のバックエンドを更新して、別の認証方法を使用したり、クレデンシャルをローテーションしたりできます。これはどちらの方法でも機能します。ユーザ名とパスワードを使用するバックエンドは証明書を使用するように更新できますが、証明書を使用するバックエンドはユーザ名とパスワードに基づいて更新できます。これを行うには、既存の認証方法を削除して、新しい認証方法を追加する必要があります。次に必要なパラメータを含む更新されたbackend.jsonファイルを使用してtridentctl backend updateを実行します

```
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 |
+-----+-----+
+-----+-----+
```

 パスワードのローテーションを実行する際には、ストレージ管理者が最初にONTAPでユーザのパスワードを更新する必要があります。この後にバックエンドアップデートが続けます。証明書のローテーションを実行する際に、複数の証明書をユーザに追加することができます。その後、バックエンドが更新されて新しい証明書が使用されるようになります。この証明書に続く古い証明書は、ONTAPクラスタから削除できます。

バックエンドを更新しても、すでに作成されているボリュームへのアクセスは中断されず、その後のボリューム接続にも影響しません。バックエンドの更新が成功すると、TridentがONTAPバックエンドと通信し、以降のボリューム処理を処理できるようになります。

Trident用のカスタムONTAPロールの作成

Tridentで処理を実行するためにONTAP adminロールを使用する必要がないように、最小Privilegesを持

つONTAPクラスタロールを作成できます。Tridentバックエンド構成にユーザ名を含めると、Trident作成したONTAPクラスタロールが使用されて処理が実行されます。

Tridentカスタムロールの作成の詳細については、を参照してください["Tridentカスタムロールジェネレータ"](#)。

ONTAP CLIノショウ

1. 次のコマンドを使用して新しいロールを作成します。

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

2. Tridentユーザのユーザ名を作成します。

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

3. ユーザにロールをマッピングします。

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

System Managerの使用

ONTAPシステムマネージャで、次の手順を実行します。

1. カスタムロールの作成：

- a. クラスタレベルでカスタムロールを作成するには、*[クラスタ]>[設定]*を選択します。

(または) SVMレベルでカスタムロールを作成するには、*[ストレージ]>[Storage VM]>[設定]>[ユーザとロール]*を選択し`required SVM`ます。

- b. の横にある矢印アイコン (→*) を選択します。

- c. [Roles]*で[+Add]*を選択します。

- d. ロールのルールを定義し、*[保存]*をクリックします。

2. ロールをTridentユーザにマップする:+[ユーザとロール]ページで次の手順を実行します。

- a. で[アイコンの追加]+*を選択します。

- b. 必要なユーザ名を選択し、* Role *のドロップダウンメニューでロールを選択します。

- c. [保存 (Save)]をクリックします。

詳細については、次のページを参照してください。

- "ONTAPの管理用のカスタムロール"または"カスタムロールの定義"
- "ロールとユーザを使用する"

双方向CHAPによる接続の認証

Tridentでは、ドライバと `ontap-san-economy`、ドライバの双方向CHAPを使用してiSCSIセッションを認証できます。`ontap-san`。これには、バックエンド定義でオプションを有効にする必要があり `useCHAP` ます。に設定する `true` と、TridentはSVMのデフォルトのイニシエータセキュリティを双方向CHAPに設定し、ユーザ名とシークレットをバックエンドファイルに設定します。接続の認証には双方向 CHAP を使用することを推奨します。次の設定例を参照してください。

```
---
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」パラメータは、1回だけ設定できるブール型のオプションです。デフォルトでは `false` に設定されています。`true` に設定したあとで、`false` に設定することはできません。

「`useCHAP=true`」に加えて、「`chapInitiatorSecret`」、「`chapTargetInitiatorSecret`」、「`chapTargetUsername`」、および「`chapUsername`」フィールドもバックエンド定義に含める必要があります。シークレットは `tridentctl update` を実行してバックエンドを作成した後に変更できます

仕組み

`true`に設定する `useCHAP` と、ストレージ管理者はTridentにストレージバックエンドでCHAPを構成するように指示します。これには次のものが含まれます。

- SVMでCHAPをセットアップします。
 - SVMのデフォルトのイニシエータセキュリティタイプが`none`（デフォルトで設定）*で、*ボリュームに既存のLUNがない場合、Tridentはデフォルトのセキュリティタイプをに設定し CHAP、CHAPイニシエータとターゲットのユーザ名とシークレットの設定に進みます。
 - SVMにLUNが含まれている場合、TridentはSVMでCHAPを有効にしません。これにより、SVMにすでに存在するLUNへのアクセスが制限されなくなります。
- CHAPイニシエータとターゲットのユーザ名とシークレットを設定します。これらのオプションは、バックエンド構成で指定する必要があります（上記を参照）。

バックエンドが作成されると、Tridentは対応するCRDを作成し `tridentbackend`、CHAPシークレットとユーザ名をKubernetesシークレットとして格納します。このバックエンドでTridentによって作成されたすべてのPVSがマウントされ、CHAP経由で接続されます。

認証情報をローテーションしてバックエンドを更新する

CHAP 証明書を更新するには 'backend.json' ファイルの CHAP パラメータを更新します。これには 'CHAP シークレット' を更新し 'tridentctl update' コマンドを使用してこれらの変更を反映する必要があります。

 バックエンドの CHAP シークレットを更新する場合は、を使用してバックエンドを更新する必要があります。tridentctl、ONTAP CLI または ONTAP システムマネージャを使用してストレージクラスタのクレデンシャルを更新しないでください。Trident ではこれらの変更を反映できません。

```
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": "c19qxUpDaTeD",
  "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 |
+-----+-----+
+-----+-----+
```

既存の接続は影響を受けず、SVM上のTridentによってクレデンシャルが更新されてもアクティブなままであります。新しい接続では更新されたクレデンシャルが使用され、既存の接続は引き続きアクティブになります。古いPVSを切断して再接続すると、更新されたクレデンシャルが使用されます。

ONTAP の SAN構成オプションと例

Tridentのインストール時にONTAP SANドライバを作成して使用する方法について説明します。このセクションでは、バックエンドの構成例と、バックエンド

をStorageClassesにマッピングするための詳細を示します。

"ASA r2システム"ストレージ層の実装において他のONTAPシステム(ASA、AFF、FAS)と異なります。これらのバリエーションは、記載されている特定のパラメータの使用に影響します。"ASA r2システムと他のONTAPシステムの違いについて詳しくは、こちらをご覧ください。"。



のみ`ontap-san`ドライバー(iSCSI、NVMe/TCP、およびFCプロトコル付き)は、ASA r2システムでサポートされています。

Tridentバックエンド構成では、システムがASA r2であることを指定する必要はありません。選択すると`ontap-san`として`storageDriverName`TridentはASA r2またはその他のONTAPシステムを自動的に検出します。以下の表に示すように、一部のバックエンド構成パラメータはASA r2システムには適用されません。

バックエンド構成オプション

バックエンド設定オプションについては、次の表を参照してください。

パラメータ	説明	デフォルト
「バージョン」		常に1
'storageDriverName'	ストレージドライバの名前	ontap-san`または`ontap-san-economy`
backendName`	カスタム名またはストレージバックエンド	ドライバ名+"_"+dataLIF
「管理LIF」	<p>クラスタ管理LIFまたはSVM管理LIFのIPアドレス。</p> <p>Fully Qualified Domain Name (FQDN; 完全修飾ドメイン名)を指定できます。</p> <p>IPv6フラグを使用してTridentがインストールされている場合は、IPv6アドレスを使用するように設定できます。IPv6アドレスは、のように角かっこで定義する必要があります</p> <p>[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:355]。</p> <p>シームレスなMetroClusterスイッチオーバーについては、を参照してMetroClusterの例ください。</p> <p>「vsadmin」のクレデンシャルを使用する場合はSVMのクレデン`managementLIF`シャル、「admin」のクレデンシャルを使用する場合はクラスタのクレデンシャル`managementLIF`を使用する必要があります。</p>	"10.0.0.1","[2001:1234:abcd:fe]"

パラメータ	説明	デフォルト
「重複排除」	プロトコル LIF の IP アドレス。IPv6 フラグを使用して Trident がインストールされている場合は、IPv6 アドレスを使用するように設定できます。IPv6 アドレスは、のように角かっこで定義する必要があります [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]。* iSCSI の場合は指定しないでください。Trident は、を使用して " ONTAP の選択的LUNマップ "、マルチパスセッションの確立に必要な iSCSI LIF を検出します。が明示的に定義されている場合は、警告が生成され、`dataLIF` ます。 MetroCluster の場合は省略してください。*を参照してください MetroCluster の例 。	SVMの派生物です
「VM」	使用する Storage Virtual Machine * MetroCluster の場合は省略してください。 MetroCluster の例 。	SVM 「管理 LIF」 が指定されている場合に生成されます
「useCHAP」	CHAP を使用して ONTAP SAN ドライバの iSCSI を認証します（ブーリアン）。バックエンドで指定された SVM のデフォルト認証として 双方向 CHAP を設定して 使用する場合は、Trident のをに設定し、`true` ます。 詳細については、を参照してください "バックエンドに ONTAP SAN ドライバを設定する準備をします" 。 FCP または NVMe/TCP ではサポートされません。	「偽」
「chapInitiatorSecret」	CHAP イニシエータシークレット。「useCHAP = TRUE」 の場合は必須	""
「ラベル」	ボリュームに適用する任意の JSON 形式の ラベル の セット	""
「chapTargetInitiatorSecret」	CHAP ターゲットイニシエータシークレット。「useCHAP = TRUE」 の場合は必須	""
「chapUsername」	インバウンドユーザ名。「useCHAP = TRUE」 の場合は必須	""
「chapTargetUsername」	ターゲットユーザ名。「useCHAP = TRUE」 の場合は必須	""
「clientCertificate」をクリックします	クライアント証明書の Base64 エンコード値。証明書ベースの認証に使用されます	""
「clientPrivateKey」	クライアント秘密鍵の Base64 エンコード値。証明書ベースの認証に使用されます	""
「trustedCACertificate」	信頼された CA 証明書の Base64 エンコード値。任意。証明書ベースの認証に使用されます。	""

パラメータ	説明	デフォルト
「ユーザ名」	ONTAPクラスタと通信するために必要なユーザー名。資格情報ベースの認証に使用されます。Active Directory認証については、"Active Directory の認証情報を使用して、バックエンド SVM に対してTrident を認証する"。	""
「password」と入力します	ONTAPクラスタと通信するために必要なパスワード。資格情報ベースの認証に使用されます。Active Directory認証については、"Active Directory の認証情報を使用して、バックエンド SVM に対してTrident を認証する"。	""
'VM'	使用する Storage Virtual Machine	SVM 「管理 LIF 」が指定されている場合に生成されます
'storagePrefix'	SVM で新しいボリュームをプロビジョニングする際に使用するプレフィックスを指定します。あとから変更することはできません。このパラメータを更新するには、新しいバックエンドを作成する必要があります。	trident
「集約」	<p>プロビジョニング用のアグリゲート（オプション）。設定する場合は SVM に割り当てる必要があります）。ドライバの場合 <code>ontap-nas-flexgroup</code>、このオプションは無視されます。割り当てられていない場合は、使用可能ないずれかのアグリゲートを使用してFlexGroupボリュームをプロビジョニングできます。</p> <p> SVMでアグリゲートが更新されると、Tridentコントローラを再起動せずにSVMをポーリングすることで、Tridentでアグリゲートが自動的に更新されます。ボリュームをプロビジョニングするようにTridentで特定のアグリゲートを設定している場合、アグリゲートの名前を変更するかSVMから移動すると、SVMアグリゲートのポーリング中にTridentでバックエンドが障害状態になります。アグリゲートをSVMにあるアグリゲートに変更するか、アグリゲートを完全に削除してバックエンドをオンラインに戻す必要があります。</p> <p>ASA r2 システムには指定しないでください。</p>	""

パラメータ	説明	デフォルト
「 <code>AggreglimitateUsage</code> 」と入力します	使用率がこの割合を超えている場合は、プロビジョニングが失敗します。Amazon FSx for NetApp ONTAP バックエンドを使用している場合は、を指定しないで <code>limitAggregateUsage</code> `ください。指定されたと `vsadmin` には `fsxadmin、アグリゲートの使用量を取得してTridentを使用して制限するために必要な権限が含まれていません。ASA r2 システムには指定しないでください。	"" (デフォルトでは適用されません)
「 <code>limitVolumeSize</code> 」と入力します	要求されたボリュームサイズがこの値を超えている場合、プロビジョニングが失敗します。また、LUNで管理するボリュームの最大サイズも制限します。	"" (デフォルトでは適用されません)
<code>'lunsPerFlexvol</code>	FlexVolあたりの最大 LUN 数。有効な範囲は 50、200 です	100
「バグトレースフラグ」	トラブルシューティング時に使用するデバッグフラグ。例： <code>{"api": false, "method": true}</code> トラブルシューティングを行い、詳細なログダンプが必要な場合を除き、は使用しないでください。	null

パラメータ	説明	デフォルト
「useREST」	<p>ONTAP REST API を使用するためのブール パラメーター。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>`useREST`に設定すると `true`、 TridentはONTAP REST APIを使用してバックエンドと通信します。 `false` Trident は、バックエンドとの通信に ONTAPI (ZAPI) 呼び出しを使用します。この機能にはONTAP 9.11.1 以降が必要です。さらに、使用するONTAPロ グインロールには、 `ontapi`応用。これは、事前に定義された `vsadmin`そして `cluster- admin`役割。 Trident 24.06リリースおよびONTAP 9.15.1以降では、 `useREST`設定されている `true`デフォルト; 変更 `useREST`に `false`ONTAPI (ZAPI) 呼び出しを使用します。</p> </div> <p>`useREST`NVMe/TCP に完全対応しています。</p> <p> NVMe はONTAP REST API でのみサ ポートされ、ONTAPI (ZAPI) ではサポ ートされません。</p> <p>指定されている場合、常に `true` ASA r2 システムの 場合。</p>	true ONTAP 9.15.1以降の場 合は、それ以外の場合は false。
sanType	iSCSI、nvme`NVMe/TCP、または `fcp`SCSI over Fibre Channel (FC; SCSI over Fibre Channel) に対してを選択します `iscsi`。	iscsi 空白の場合

パラメータ	説明	デフォルト
formatOptions	<p>を使用して、`formatOptions`コマンドのコマンドライン引数を指定します。この引数 `mkfs` は、ボリュームがフォーマットされるたびに適用されます。これにより、好みに応じてボリュームをフォーマットできます。デバイスパスを除いて、`mkfs`コマンドオプションと同様に `formatOptions` を指定してください。例：「-E nodiscard」</p> <p>対応機種 <code>ontap-san</code> そして <code>ontap-san-economy</code> iSCSI プロトコルを使用したドライバー。 <code>**iSCSI</code> および <code>NVMe/TCP</code> プロトコルを使用する場合、ASA r2 システムでもサポートされます。</p>	
limitVolumePoolSize	ONTAP SANエコノミーバックエンドでLUNを使用する場合の、要求可能な最大FlexVolサイズ。	"" (デフォルトでは適用されません)
denyNewVolumePools	バックエンドがLUNを格納するために新しいFlexVolボリュームを作成することを制限します <code>ontap-san-economy</code> 。新しいPVのプロビジョニングには、既存のFlexVolのみが使用されます。	

formatOptionsの使用に関する推奨事項

Trident は、フォーマット処理を高速化するために次のオプションを推奨しています。

- **-E nodiscard (ext3、ext4):** `mkfs` 時にブロックを破棄しません (最初にブロックを破棄することは、ソリッドステートデバイスおよびスパース / シンプロビジョニングストレージで役立ちます)。これは非推奨のオプション「-K」に代わるもので、ext3、ext4 ファイルシステムに適用できます。
- **-K (xfs):** `mkfs` 時にブロックを破棄しません。このオプションは xfs ファイルシステムに適用できます。

Active Directory の認証情報を使用して、バックエンド SVM に対して Trident を認証する

Active Directory (AD) 認証情報を使用してバックエンド SVM に対して認証するように Trident を設定できます。AD アカウントが SVM にアクセスする前に、クラスタまたは SVM への AD ドメインコントローラー アクセスを設定する必要があります。AD アカウントを使用してクラスターを管理するには、ドメイントンネルを作成する必要があります。参照 ["ONTAPでActive Directoryドメインコントローラのアクセスを構成する" 詳細](#) については。

手順

1. バックエンド SVM のドメインネームシステム (DNS) 設定を構成します。

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

2. 次のコマンドを実行して、Active Directory に SVM のコンピュータアカウントを作成します。

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

3. このコマンドを使用して、クラスタまたは SVM を管理するための AD ユーザーまたはグループを作成しま

す。

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

4. Tridentバックエンド設定ファイルで、`username` そして `password` パラメータをそれぞれ AD ユーザー名またはグループ名とパスワードに渡します。

ボリュームのプロビジョニング用のバックエンド構成オプション

これらのオプションを使用して、のデフォルトプロビジョニングを制御できます `defaults` 設定のセクション。例については、以下の設定例を参照してください。

パラメータ	説明	デフォルト
「平和の配分」	space-allocation for LUN のコマンドを指定します	"true" 指定されている場合は、 true ASA r2 システムの場合。
「平和のための準備」を参照してください	スペースリザベーションモード：「none」（シン）または「volume」（シック）。設定 none ASA r2 システムの場合。	"なし"
「ナップショットポリシー」	使用するSnapshotポリシー。設定 none ASA r2 システムの場合。	"なし"
「QOSPolicy」	作成したボリュームに割り当てる QoS ポリシーグループ。ストレージプール / バックエンドごとに QOSPolicy または adaptiveQosPolicy のいずれかを選択します。TridentでQoSポリシーグループを使用するには、ONTAP 9.8以降が必要です。共有されていないQoSポリシーグループを使用し、ポリシーグループが各コンステイチュエントに個別に適用されるようにします。QoSポリシーグループを共有すると、すべてのワーカロードの合計スループットの上限が適用されます。	""
「adaptiveQosPolicy」を参照してください	アダプティブ QoS ポリシーグループ：作成したボリュームに割り当てます。ストレージプール / バックエンドごとに QOSPolicy または adaptiveQosPolicy のいずれかを選択します	""
「スナップショット予約」	Snapshot用にリザーブされているボリュームの割合。 ASA r2 システムには指定しないでください。	次の場合は「0」 <code>snapshotPolicy</code> は「none」、それ以外の場合は「」です。
'plitOnClone	作成時にクローンを親からスプリットします	いいえ

パラメータ	説明	デフォルト
「暗号化」	新しいボリュームでNetApp Volume Encryption (NVE) を有効にします。デフォルトはです。`false`このオプションを使用するには、クラスタで NVE のライセンスが設定され、有効になっている必要があります。バックエンドでNAEが有効になっている場合、TridentでプロビジョニングされたすべてのボリュームでNAEが有効になります。詳細については、を参照してください "TridentとNVEおよびNAEとの連携" 。	"false" 指定されている場合は、 true ASA r2 システムの場合。
luksEncryption	LUKS暗号化を有効にします。を参照してください "Linux Unified Key Setup (LUKS；統合キーセットアップ) を使用" 。	"" 設定 false ASA r2 システムの場合。
階層ポリシー	階層化ポリシーは「なし」を使用します。 ASA r2 システムでは指定しないでください。	
nameTemplate	カスタムボリューム名を作成するためのテンプレート。	""

ボリュームプロビジョニングの例

デフォルトが定義されている例を次に示します。

```
---
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'
```



ドライバを使用して作成されたすべてのボリュームについて、`ontap-san` TridentはLUNメタデータに対応するために10%の容量をFlexVolに追加します。LUNは、ユーザが PVC で要求したサイズとまったく同じサイズでプロビジョニングされます。Tridentは、FlexVolに10%を追加します（ONTAPでは使用可能なサイズとして表示されます）。ユーザには、要求した使用可能容量が割り当てられます。また、利用可能なスペースがフルに活用されていないかぎり、LUNが読み取り専用になることもありません。これは、ONTAPとSANの経済性には該当しません。

定義されたバックエンドの場合 `snapshotReserve`、Tridentは次のようにボリュームのサイズを計算します。

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

にTridentがFlexVolに追加する10%の容量です。`snapshotReserve = 5%`、`PVC要求 = 5 GiB`の場合、ボリュームの合計サイズは5.79 GiB、使用可能なサイズは5.5 GiBです。`'volume show'`コマンドを実行すると、次の例のような結果が表示されます。

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_134adb8e6d		online	RW	5.79GB	5.50GB	0%
	_pvc_e8372153_9ad9_474a_951a_08ae15e1c0ba		online	RW	1GB	511.8MB	0%
3 entries were displayed.							

現在、既存のボリュームに対して新しい計算を行うには、サイズ変更だけを使用します。

最小限の設定例

次の例は、ほとんどのパラメータをデフォルトのままにする基本的な設定を示しています。これは、バックエンドを定義する最も簡単な方法です。



TridentでAmazon FSx on NetApp ONTAPを使用している場合、NetAppでは、IPアドレスではなく、LIFのDNS名を指定することを推奨します。

ONTAP SANの例

これは、ontap-san ドライバ。

```
---  
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の例

スイッチオーバーやスイッチバックの実行中にバックエンド定義を手動で更新する必要がないようにバックエンドを設定できます。 "SVMのレプリケーションとリカバリ"。

スイッチオーバーとスイッチバックをシームレスに実行するには、を使用してSVMを指定しmanagementLIF、パラメータは省略します svm。例：

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

ONTAP SANの経済性の例

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

証明書ベースの認証の例

この基本的な設定例では、`clientCertificate`、`clientPrivateKey`、および
`trustedCACertificate`（信頼されたCAを使用している場合はオプション）がに入力されます
`backend.json` およびは、クライアント証明書、秘密鍵、信頼されたCA証明書のbase64エンコード値
をそれぞれ取得します。

```
---  
version: 1  
storageDriverName: ontap-san  
backendName: DefaultSANBackend  
managementLIF: 10.0.0.1  
svm: svm_iscsi  
useCHAP: true  
chapInitiatorSecret: c19qxIm36DKyawxy  
chapTargetInitiatorSecret: rqxigXgkesIpwxyz  
chapTargetUsername: iJF4heBRT0TCwxyz  
chapUsername: uh2aNCLSd6cNwxyz  
clientCertificate: ZXROZXJwYXB...ICMgJ3BhcGVyc2  
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX  
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
```

双方向CHAPの例

次の例では、useCHAP をに設定します true。

ONTAP SAN CHAPの例

```
---
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エコノミーCHAPの例

```
---
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の例

ONTAPバックエンドでNVMeを使用するSVMを設定しておく必要があります。これはNVMe/TCPの基本的なバックエンド構成です。

```
---  
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) の例

ONTAPバックエンドでFCを使用してSVMを設定しておく必要があります。これはFCの基本的なバックエンド構成です。

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

nameTemplateを使用したバックエンド構成の例

```
---
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エコノミードライバのformatOptionsの例

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

仮想プールを使用するバックエンドの例

これらのサンプルバックエンド定義ファイルでは、次のような特定のデフォルトがすべてのストレージプールに設定されています。spaceReserve 「なし」 の場合は、spaceAllocation との誤り encryption 実行されます。仮想プールは、ストレージセクションで定義します。

Tridentでは、[Comments]フィールドにプロビジョニングラベルが設定されます。コメントは、仮想プール上のすべてのラベルをプロビジョニング時にストレージボリュームにコピーするFlexVol volume Tridentに設定されます。ストレージ管理者は、仮想プールごとにラベルを定義したり、ボリュームをラベルでグループ化し

たりできます。

これらの例では、一部のストレージプールが独自の `spaceReserve`、`spaceAllocation`、および `encryption` 値、および一部のプールはデフォルト値よりも優先されます。

ONTAP SANの例

```
---  
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の経済性の例

```
---  
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: mysql ldb  
    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の例

```
---
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"
```

バックエンドを **StorageClasses** にマッピングします

次のStorageClass定義は、[\[仮想プールを使用するバックエンドの例\]](#)。を使用する parameters.selector フィールドでは、各StorageClassがボリュームのホストに使用できる仮想プールを呼び出します。ボリュームには、選択した仮想プール内で定義された要素があります。

- 。 protection-gold StorageClassは、ontap-san バックエンド：ゴールドレベルの保護を提供する唯一のプールです。

```
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は、内の2番目と3番目の仮想プールにマッピングされます。
ontap-san バックエンド：これらは、ゴールド以外の保護レベルを提供する唯一のプールです。

```
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は内の3番目の仮想プールにマッピングされます ontap-san-economy バックエンド：これは、mysqldbタイプアプリケーション用のストレージプール構成を提供する唯一のプールです。

```
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は内の2番目の仮想プールにマッピングされます ontap-san バックエンド：シルバーレベルの保護と20000クレジットポイントを提供する唯一のプールです。

```

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は内の3番目の仮想プールにマッピングされます ontap-san バックエンドと内の4番目の仮想プール ontap-san-economy バックエンド：これらは、5000クレジットポイントを持つ唯一のプールオファリングです。

```

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

```

- 。 my-test-app-sc StorageClassはにマッピングされます testAPP 内の仮想プール ontap-san ドライバ sanType: nvme。これは唯一のプールサービスです。 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が選択する仮想プールを決定し、ストレージ要件が満たされるようにします。

著作権に関する情報

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を隨時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5225.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用権を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用権については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。