Red Hat Enterprise Linux 8

RHEL で論理ボリュームの重複排除および圧縮

VDO を使用した LVM ストレージ容量の拡張

Last Updated: 2021-07-02

Red Hat Enterprise Linux 8 RHEL で論理ボリュームの重複排除および圧縮

VDO を使用した LVM ストレージ容量の拡張

法律上の通知

概要

本書は、LVM で Virtual Data Optimizer (VDO) 機能を使用して、RHEL で重複排除した論理ボリュームと圧縮した論理ボリュームを管理する方法を説明します。

目次

多様性を受け入れるオープンソースの強化 3

RED HAT ドキュメントへのフィードバック (英語のみ) 4

第1章 LVM での VDO の概要 5

第2章 LVM-VDO 要件 6
2.1. VDO メモリー要件 6
2.2. VDO ストレージの領域要件 6
2.3. 物理サイズによる VDO 要件の例 7
2.4. ストレージスタックの LVM-VDO の配置 8

第3章 重複排除した論理ボリュームと圧縮論理ボリュームの作成 9
3.1. LVM-VDO デプロイメントシナリオ 9
3.2. LVM-VDO ボリュームの物理サイズおよび論理サイズ 11
3.3. VDO 論理ボリュームに推奨される論理サイズ 11
3.4. VDO のスラブサイズ 12
3.5. VDO のインストール 12
3.6. LVM-VDO ボリュームの作成 12
3.7. LVM-VDO ボリュームのマウント 14
3.8. LVM-VDO ボリュームでの圧縮および重複排除設定の変更 14

第4章 LVM-VDO ボリュームの縮小オプション 16
4.1. VDO で DISCARD マウントオプションの有効化 16
4.2. 定期的な TRIM 操作の設定 16

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティーにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、弊社 の CTO、Chris Wright のメッセージ を参照してください。

RED HAT ドキュメントへのフィードバック (英語のみ)

ご意見ご要望をお聞かせください。ドキュメントの改善点はございませんか。改善点を報告する場合は、以下のように行います。

- 特定の文章に簡単なコメントを記入する場合は、以下の手順を行います。
  1. ドキュメントの表示が **Multi-page HTML** 形式になっていて、ドキュメントの右上端に **Feedback** ボタンがあることを確認してください。
  2. マウスカーソルで、コメントを追加する部分を強調表示します。
  3. そのテキストの下に表示される **Add Feedback** ポップアップをクリックします。
  4. 表示される手順に従ってください。
- より詳細なフィードバックを行う場合は、Bugzilla のチケットを作成します。
  1. Bugzilla の Web サイトにアクセスします。
  2. Component で **Documentation** を選択します。
  3. **Description** フィールドに、ドキュメントの改善に関するご意見を記入してください。ドキュメントの該当部分へのリンクも記入してください。
  4. **Submit Bug** をクリックします。

第1章 LVM での VDO の概要

VDO (Virtual Data Optimizer) 機能は、ストレージ用にインラインのブロックレベルの重複排除、圧縮、およびシンプロビジョニングを提供します。VDO は、LVM シンプロビジョニングのボリュームと同様に、LVM 論理ボリューム (LV) として管理できます。

LVM (LVM-VDO) 上の VDO ボリュームは、以下の論理ボリュームで構成されます。

VDO プール LV (論理ボリューム)

これは、VDO LV 用のデータを保存し、重複排除して、圧縮するバッキング物理デバイスです。VDO プール LV は、VDO ボリュームの物理サイズを設定します。これは、VDO がディスクに保存できるデータ量です。

現在、各 VDO プール LV は 1 つの VDO LV のみを保持できます。その結果、VDO は各 VDO LV を重複排除し、圧縮します。つまり、VDO は、複数の VDO LV 間で共有されるデータの重複排除または圧縮ができません。

VDO LV

これは、VDO プール LV 上の仮想、プロビジョニングされたデバイスです。VDO LV は、VDO ボリュームのプロビジョニングされた論理サイズを設定します。これは、重複排除と圧縮が行われる前にアプリケーションがボリュームに書き込みできるデータ量です。

表1.1 LVM および LVM シンプロビジョニングにおける VDO 内のコンポーネントの比較

	物理デバイス	プロビジョニングされたデバイス
LVM 上の VDO	VDO プール LV (論理ボリューム)	VDO LV
LVM シンプロビジョニング	シンプール	シンボリューム

VDO はシンプロビジョニングされているため、ファイルシステムとアプリケーションは、使用中の論理領域だけを認識し、実際に利用可能な物理領域は認識しません。スクリプトを使用して、実際に利用可能な領域を監視し、使用量がしきい値を超えた場合 (たとえば、VDO プール LV の使用量が 80% になった場合) にアラートを生成します。

関連情報

- スタンドアロン VDO の詳細は、「ストレージの重複排除および圧縮」を参照してください。
- LVM シンプロビジョニングの詳細は、「シンプロビジョニングされた論理ボリューム (シンボリューム) の作成および管理」を参照してください。

第2章 LVM-VDO 要件

LVM 上の VDO には、配置とシステムリソースに関する特定の要件があります。

2.1. VDO メモリー要件

各 VDO ボリュームには、2 つの異なるメモリー要件があります。

VDO モジュール

VDO には、固定メモリー 38 MB と変動用に容量を確保する必要があります。

- 設定済みのブロックマップキャッシュサイズ 1 MB ごとに 1.15 MB のメモリー。ブロックマップキャッシュには、少なくとも 150 MB のメモリーが必要です。
- 1 TB の論理領域ごとに 1.6 MB のメモリー
- ボリュームが管理する物理ストレージの 1 TB ごとに 268 MB のメモリー。

UDS インデックス

Universal Deduplication Service (UDS) には、最低 250 MB のメモリーが必要です。このメモリー量は、重複排除が使用するデフォルトの容量です。この値は、インデックスで必要なストレージ容量も左右するので、VDO ボリュームのフォーマット時にこの値を設定してください。

UDS インデックスに必要なメモリーは、インデックスタイプと、重複排除ウィンドウに必要なサイズにより指定されます。

インデックスタイプ	重複排除ウィンドウ	備考
Dense	RAM 1 GB あたり 1 TB	通常、最大 4 TB の物理ストレージには、1 GB の dense インデックスで十分です。
Sparse	RAM 1 GB あたり 10 TB	通常、最大 40 TB の物理ストレージには、1 GB の sparse インデックスで十分です。

VDO で推奨されるモードは、UDS の sparse インデックス機能です。データの一時的な局所性に依存し、メモリー内で最も関連するインデックスエントリーのみを保持しようとします。sparse インデックスでは、UDS は、同じ量のメモリーを使用しながら、dense を使用したときの 10 倍以上長い重複排除ウィンドウを維持できます。

sparse インデックスを使用すると対象範囲が広くなりますが、dense インデックスの方が提供する重複排除アドバイスが多くなります。ワークロードの観点では、多くの場合、同じメモリー量であれば、dense インデックスと sparse インデックスにおける重複排除率の差異はごくわずかとなります。

関連情報

- UDS インデックスのメモリー要件の具体例は、「物理サイズによる VDO 要件の例」 を参照してください。

2.2. VDO ストレージの領域要件

VDO ボリュームを設定して、最大 256 TB の物理ストレージを使用するように設定できます。データを格納するのに使用できるのは、物理ストレージの一部のみです。このセクションでは、VDO に管理されるボリュームで使用可能なサイズを特定するための計算方法を説明します。

VDO では、2 種類の VDO メタデータと UDS インデックスにストレージが必要です。

- 1 つ目の VDO メタデータは、**物理ストレージ** 4 GB ごとに約 1 MB が使用され、スラブごとに 1 MB ずつ使用します。
- 2 つ目の VDO メタデータでは、**論理ストレージ** 1 GB ごとに約 1.25 MB が使用され、最も近いスラブに切り上げられます。
- UDS インデックスに必要なストレージの容量は、インデックスの種類と、インデックスに割り当てられている RAM の容量によって異なります。RAM 1 GB ごとに、dense の UDS インデックスはストレージを 17 GB 使用し、sparse の UDS インデックスはストレージを 170 GB 使用します。

関連情報

- VDO ストレージ要件の具体例は、「物理サイズによる VDO 要件の例」を参照してください。
- スラブの説明は、「VDO のスラブサイズ」を参照してください。

2.3. 物理サイズによる VDO 要件の例

以下の表は、基となるボリュームの物理サイズに基づいた、VDO のシステム要件の概算を示しています。それぞれの表には、プライマリーストレージ、バックアップストレージなどの、目的のデプロイメントに適した要件が記載されています。

正確な数値は、VDO ボリュームの設定により異なります。

プライマリーストレージのデプロイメント

プライマリーストレージの場合、UDS インデックスのサイズは、物理サイズの 0.01% から 25% になります。

表2.1 プライマリストレージのストレージ要件およびメモリー要件

物理サイズ	メモリー使用率: UDS	メモリー使用率: VDO	ディスク使用率	インデックスタイプ
10GB ~ 1TB	250MB	472MB	2.5GB	Dense
2 ~ 10TB	1GB	3GB	10GB	Dense
	250MB		22GB	Sparse
11 ~ 50TB	2GB	14GB	170GB	Sparse
51 ~ 100TB	3GB	27GB	255GB	Sparse
101 ~ 256TB	12GB	69GB	1020GB	Sparse

バックアップストレージのデプロイメント

バックアップストレージの場合、UDS インデックスは、バックアップセットのサイズよりは大きくなりますが、物理サイズと同じか、より小さくなります。バックアップセットや物理サイズが今後大きくなる可能性がある場合は、これをインデックスサイズに組み込んでください。

表2.2 バックアップストレージのストレージ要件およびメモリ要件

物理サイズ	メモリー使用率: UDS	メモリー使用率: VDO	ディスク使用率	インデックスタイプ
10GB ~ 1TB	250MB	472MB	2.5 GB	Dense
2 ~ 10TB	2GB	3GB	170GB	Sparse
11 ~ 50TB	10GB	14GB	850GB	Sparse
51 ~ 100TB	20GB	27GB	1700GB	Sparse
101 ~ 256TB	26GB	69GB	3400GB	Sparse

2.4. ストレージスタックの LVM-VDO の配置

VDO 論理ボリュームに特定のストレージ層を配置する必要があります。

シックプロビジョニングの層を VDO の上に配置することができますが、その場合はこのようなシックプロビジョニングの保証に頼ることはできません。VDO 層はシンプロビジョニングされているため、シンプロビジョニングの影響は、上記のすべての層に及びます。VDO ボリュームが監視されていない場合には、VDO 層の上にあるシックプロビジョニングのボリュームで、物理領域が不足する可能性があります。

以下の層は、VDO の配下にある場合にサポートされます。VDO 層の上に配置しないでください。

- DM Multipath
- DM Crypt
- ソフトウェア RAID (LVM または MD RAID)

以下の設定には **対応していません。**

- ループバックデバイスの上に VDO
- VDO の上に暗号化されたボリューム
- VDO ボリュームにパーティション作成
- VDO ボリュームの上に RAID (LVM RAID、MD RAID、またはその他のタイプ)

関連情報

- 他の LVM レイヤーによる VDO のスタックの詳細は、「LVM ボリュームのスタック」を参照してください。

第3章 重複排除した論理ボリュームと圧縮論理ボリュームの作成

VDO 機能を使用する LVM 論理ボリュームを作成して、データを重複排除して圧縮できます。

3.1. LVM-VDO デプロイメントシナリオ

LVM 上の VDO (LVM-VDO) は、さまざまな方法でデプロイして、以下に対して、重複排除したストレージを提供できます。

- ブロックアクセス
- ファイルアクセス
- ローカルストレージ
- リモートストレージ

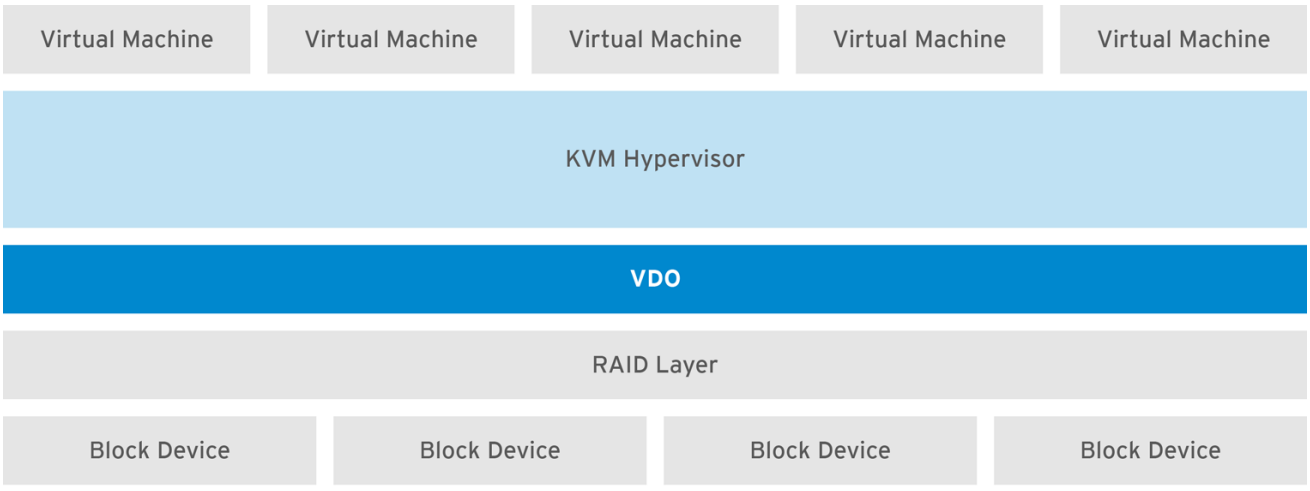
LVM-VDO は、通常の論理ボリューム (LV) として重複排除したストレージを公開するため、そのストレージを標準ファイルシステム、iSCSI および FC のターゲットドライバー、または統合ストレージとして使用できます。

注記

現在、LVM-VDO への Ceph Storage のデプロイメントはサポートされていません。

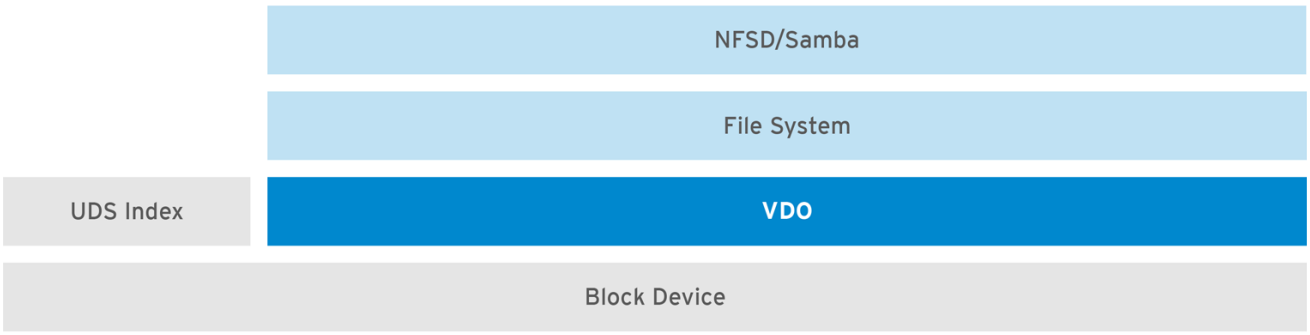
KVM

DAS (Direct Attached Storage) を使用して設定した KVM サーバーに LVM-VDO をデプロイできます。



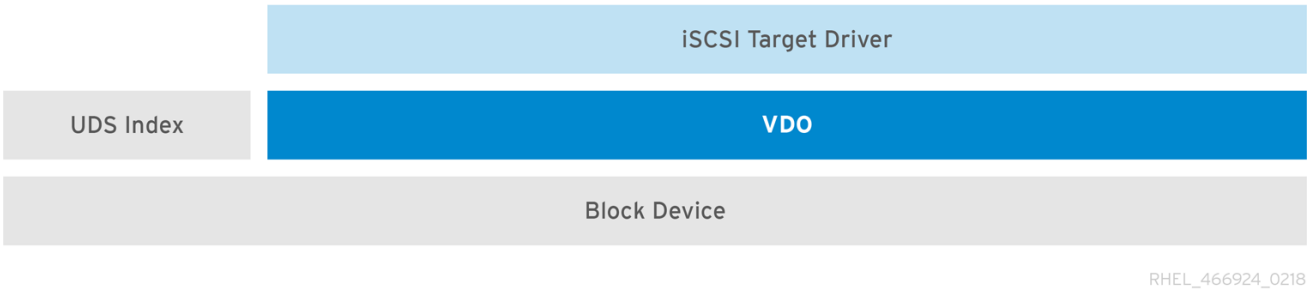
ファイルシステム

VDO LV にファイルシステムを作成して、NFS サーバーまたは Samba で、NFS ユーザーまたは CIFS ユーザーに公開します。



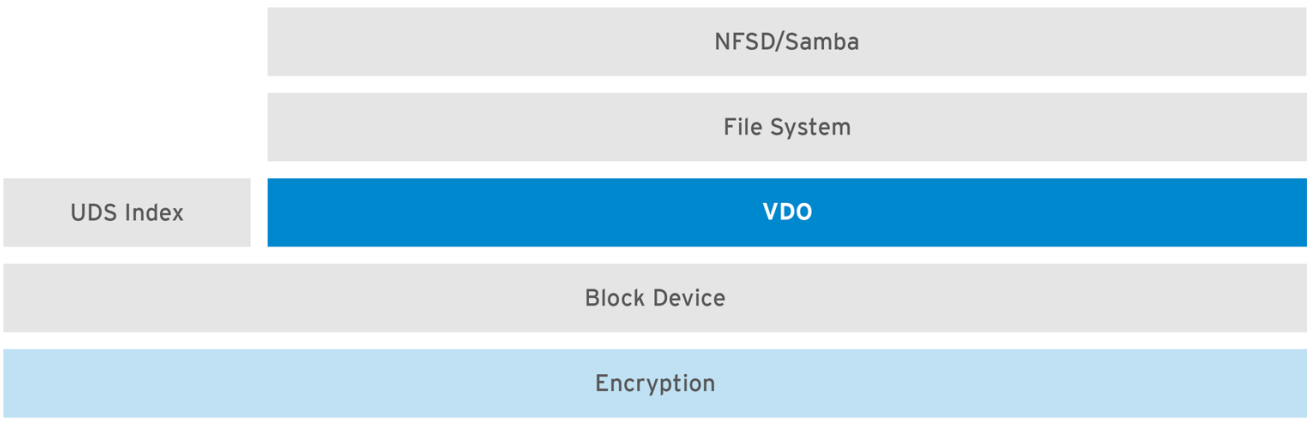
iSCSI ターゲット

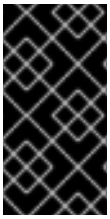
VDO LV 全体を、iSCSI ターゲットとしてリモート iSCSI イニシエーターにエクスポートできます。



暗号化

DM Crypt などのデバイスマッパー (DM) メカニズムは VDO と互換性があります。VDO LV ボリュームの暗号化により、データセキュリティーと、VDO LV にある全ファイルシステムが重複排除されるようになります。





重要

VDO LV で暗号化層を適用すると、データの重複排除が行われてもほとんど行われません。暗号化により、VDO が重複を排除する前に、重複ブロックを変更します。

常に VDO LV の下に暗号化層を配置します。

3.2. LVM-VDO ボリュームの物理サイズおよび論理サイズ

このセクションでは、VDO が使用できる物理サイズ、利用可能な物理サイズ、論理サイズを説明します。

物理サイズ

これは、VDO プール LV に割り当てられた物理エクステントと同じサイズです。VDO は、以下の目的でこのストレージを使用します。

- 重複排除および圧縮される可能性があるユーザーデータ
- UDS インデックスなどの VDO メタデータ

利用可能な物理サイズ

これは、VDO がユーザーデータに使用できる物理サイズの一部です。
これは、メタデータのサイズを引いた物理サイズと同等で、指定のスラブサイズでボリュームをスラブに分割した後の残りを引いたものと同じです。

論理サイズ

これは、VDO LV がアプリケーションに提示するプロビジョニングされたサイズです。通常、これは利用可能な物理サイズよりも大きくなります。
--virtualsize オプションを指定しないと、VDO はボリュームの比率を **1:1** にプロビジョニングします。たとえば、VDO LV を 20 GB の VDO プール LV 上に配置すると、VDO はデフォルトのインデックスサイズが使用されている場合に UDS インデックス用に 2.5 GB を確保します。残りの 17.5 GB は、VDO メタデータおよびユーザーデータに提供されます。そのため、消費する利用可能なストレージは 17.5 GB を超えません。実際の VDO ボリュームを構成するメタデータにより、これよりも少なくなる可能性があります。

VDO は現在、絶対最大論理サイズ 4PB の物理ボリュームの最大 254 倍の論理サイズに対応します。

関連情報

- さまざまな物理サイズで必要となるストレージ VDO メタデータのサイズは、「物理サイズによる VDO 要件の例」 を参照してください。

3.3. VDO 論理ボリュームに推奨される論理サイズ

VDO 論理ボリューム (LV) を設定する場合は、VDO LV が提示する論理ストレージのサイズを指定します。Red Hat では、以下の論理サイズを推奨します。

- アクティブな仮想マシンまたはコンテナーをホストする場合、Red Hat は、物理と論理の割合を 1 対 10 にすることを推奨します。つまり、物理ストレージを 1 TB にした場合は、論理ストレージを 10 TB にします。
- Ceph が提供するタイプなどのオブジェクトストレージの場合、Red Hat は、物理と論理の割合を 1 対 3 にすることを推奨します。つまり、物理ストレージを 1 TB にした場合は、論理ストレージを 3 TB にします。

いずれの場合も、VDO LV にファイルシステムを置くだけで、直接使用することも、分散クラウドストレージアーキテクチャーの一部として使用することもできます。

3.4. VDO のスラブサイズ

VDO ボリュームの物理ストレージは、複数のスラブに分割されます。各スラブは、物理領域の連続するリージョンです。指定のボリュームのスラブはすべて同じサイズになります。この値は、128 MB の 2 乗で、最大 32 GB にすることができます。

小規模なテストシステムで VDO の評価を容易にするために、デフォルトのスラブサイズは 2 GB です。1 つの VDO ボリュームに最大 8192 個のスラブを持たせることができます。したがって、2 GB のスラブを使用したデフォルト設定では、許容される物理ストレージの最大値は 16 TB です。32 GB のスラブを使用する場合、許可される物理ストレージは最大 256 TB です。VDO は、メタデータ用に少なくともスラブ全体を予約するため、予約されたスラブはユーザーデータの格納には使用できません。

スラブサイズは、VDO ボリュームのパフォーマンスには影響しません。

表3.1 物理ボリュームサイズによる推奨される VDO スラブサイズ

物理ボリュームのサイズ	推奨されるスラブサイズ
10–99 GB	1 GB
100 GB – 1 TB	2 GB
2–256 TB	32 GB

--vdoSlabSize=megabytes オプションを **vdo create** コマンドに指定することで、スラブサイズを制御できます。

3.5. VDO のインストール

この手順では、VDO ボリュームの作成、マウント、および管理に必要なソフトウェアをインストールします。

手順

- **vdo** パッケージおよび **kmod-kvdo** パッケージをインストールします。

```
# yum install vdo kmod-kvdo
```

3.6. LVM-VDO ボリュームの作成

この手順では、VDO プール LV に VDO 論理ボリューム (LV) を作成します。

前提条件

- VDO ソフトウェアをインストールしている。「VDO のインストール」 を参照してください。
- 空きストレージ容量を持つ LVM ボリュームグループがシステムに存在する。

手順

1. **vdo1** などの VDO LV の名前を選択します。システムの VDO LV ごとに、異なる名前とデバイスを使用する必要があります。

以下の手順では、**vdo-name** をその名前に置き換えます。

2. VDO LV を作成します。

```
# lvcreate --type vdo \
        --name vdo-name
        --size physical-size
        --virtualsize logical-size \
        vg-name
```

- **vg-name** を、VDO LV を配置する既存の LVM ボリュームグループの名前に置き換えます。
- **logical-size** を、VDO LV が含まれる論理ストレージのサイズに置き換えます。
- 物理サイズが 16TiB を超える場合は、以下のオプションを指定して、ボリューム上のスラブサイズを 32GiB に増やします。

```
--config 'allocation/vdo_slab_size_mb=32768'
```

物理サイズが 16TiB を超えるデフォルトのスラブサイズ 2GiB を使用すると、**lvcreate** コマンドは以下のエラーで失敗します。

```
ERROR - vdoformat: formatVDO failed on '/dev/device': VDO Status: Exceeds maximum number of slabs supported
```

例3.1 コンテナーストレージ用の VDO LV の作成

たとえば、1TB の VDO プール LV で、コンテナーストレージ用に VDO LV を作成するには、次のコマンドを実行します。

```
# lvcreate --type vdo \
        --name vdo1
        --size 1T
        --virtualsize 10T \
        vg-name
```

重要

VDO ボリュームの作成中に問題が発生した場合は、ボリュームを削除します。

3. VDO LV にファイルシステムを作成します。

- XFS ファイルシステムの場合:

```
# mkfs.xfs -K /dev/vg-name/vdo-name
```

- ext4 ファイルシステムの場合:

```
# mkfs.ext4 -E nodiscard /dev/vg-name/vdo-name
```

関連情報

関連情報

- **lvmvdo(7)** の man ページ

3.7. LVM-VDO ボリュームのマウント

この手順では、手動で、または永続的に、LVM-VDO ボリュームにファイルシステムをマウントします。

前提条件

- LVM-VDO ボリュームがシステム上にある。詳細は 「LVM-VDO ボリュームの作成」 を参照してください。

手順

- LVM-VDO ボリュームに手動でファイルシステムをマウントするには、以下のコマンドを使用します。

```
# mount /dev/vg/vdo-name mount-point
```

- システムの起動時にファイルシステムを自動的にマウントするように設定するには、**/etc/fstab** ファイルに以下の行を追加します。
  - XFS ファイルシステムの場合:

```
/dev/vg/vdo-name mount-point xfs defaults,x-systemd.device-timeout=0,x-systemd.requires=vdo.service 0 0
```

  - ext4 ファイルシステムの場合:

```
/dev/vg/vdo-name mount-point ext4 defaults,x-systemd.device-timeout=0,x-systemd.requires=vdo.service 0 0
```

  LVM-VDO ボリュームが、iSCSI などのネットワークを必要とするブロックデバイスに配置されている場合は、**_netdev** マウントオプションを追加します。

関連情報

- iSCSI や、ネットワークを必要とするその他のブロックデバイスの **_netdev** マウントオプションに関する情報は、**systemd.mount(5)** の man ページを参照してください。

3.8. LVM-VDO ボリュームでの圧縮および重複排除設定の変更

この手順では、VDO プール論理ボリューム (LV) の圧縮および重複排除を有効または無効にします。

注記

圧縮と重複排除はデフォルトで有効になっています。

前提条件

- LVM-VDO ボリュームがシステム上にある。

手順

1. 論理ボリュームで圧縮および重複排除が有効かどうかを調べるには、次のコマンドを実行します。

```
# lvs -o+vdo_compression,vdo_deduplication
```

2. 稼働中の VDOPoolLV の重複排除インデックスと圧縮のステータスを確認します。

```
# lvs -o+vdo_compression_state,vdo_index_state
```

vdo_index_state は、**error**、**close**、**opening**、**closing**、**online** および **offline** として表示されます。

3. VDOPoolLV の圧縮を有効または無効にするには、以下を行います。

```
# lvchange --compression y|n  vgname/vdopoolname
```

4. VDOPoolLV の重複排除を有効または無効にするには、以下を行います。

```
# lvchange --deduplication y|n vgname/vdopoolname
```

関連情報

- 詳細は、**lvmvdo(7)** の man ページを参照してください。

第4章 LVM-VDO ボリュームの縮小オプション

VDO ボリュームで **discard** オプションを使用すると、ブロックデバイスがファイルシステムにより未使用の領域を回収できます。もう 1 つの方法として、**fstrim** アプリケーション (オンデマンドでの破棄) または **mount -o discard** コマンドを実行して即座に破棄します。**fstrim** アプリケーションを使用する場合には、管理者は追加のプロセスをスケジュールおよび監視する必要がありますが、**mount -o discard** コマンドを使用すると、可能な場合には、領域の即時に復元できます。

このオプションのパフォーマンスへの影響は非常に大きく可能性があるため、**fstrim** アプリケーションを使用する場合は **discard** マウントオプションではなく未使用のブロックを破棄することが推奨されます。このため、**nodiscard** がデフォルトです。

4.1. VDO で DISCARD マウントオプションの有効化

この手順では、VDO ボリュームで **discard** オプションを有効にします。

前提条件

- LVM-VDO ボリュームがシステム上にある。

手順

- ボリュームで **discard** を有効にします。

```
# mount -o discard /dev/vgname/vdoname mount-point
```

関連情報

- **xfs(5)** の man ページ
- **mount(8)** の man ページ
- **lvmvdo(7)** の man ページ

4.2. 定期的な TRIM 操作の設定

この手順では、システムで TRIM 操作のスケジューリングを有効にします。

前提条件

- LVM-VDO ボリュームがシステム上にある。

手順

- タイマーを有効にして起動します。

```
# systemctl enable --now fstrim.timer
```

検証

- タイマーが有効化されていることを確認します。

```
# systemctl list-timers fstrim.timer
```

例4.1 検証手順での出力候補

```
# systemctl list-timers fstrim.timer
NEXT                         LEFT        LAST  PASSED  UNIT         ACTIVATES
Mon 2021-05-10 00:00:00 EDT  5 days left  n/a   n/a     fstrim.timer fstrim.service
```

注記

fstrim.timer は、マウントされている全ファイルシステムで実行されるため、VDO ボリュームへの参照は表示されません。

関連情報

- **fstrim(8)** の man ページ