



Red Hat Enterprise Linux 8

Device Mapper Multipath の設定

Device Mapper Multipath 機能の使用

Red Hat Enterprise Linux 8 Device Mapper Multipath の設定

Device Mapper Multipath 機能の使用

法律上の通知

Copyright © 2019 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本書は、Red Hat Enterprise Linux 8 で Device Mapper Multipath (DM-Multipath) 機能を設定および管理する方法を説明します。

目次

RED HAT ドキュメントへのフィードバック	3
第1章 DM MULTIPATH の設定	4
1.1. DM-MULTIPATH の基本設定	4
1.2. マルチパスデバイスの作成時にローカルのディスクを無視	5
1.3. 追加のストレージデバイスの設定	6
1.4. INITRAMFS ファイルシステムでマルチパスの設定	7
第2章 DM MULTIPATH 設定ファイルの編集	8
2.1. 設定ファイルの概要	8
2.2. DM MULTIPATH からのデバイスのブラックリスト	9
2.2.1. WWID でブラックリストの指定	10
2.2.2. デバイス名でのブラックリストの指定	10
2.2.3. デバイスタイプでのブラックリストの指定	11
2.2.4. udev プロパティーでのブラックリスト指定	11
2.2.5. デバイスプロトコルでのブラックリストの指定	11
2.2.6. ブラックリストの例外	11
2.3. マルチパス設定ファイルのデフォルトの編集	12
2.4. 特定デバイスのマルチパス設定の編集	13
2.5. ストレージコントローラーのマルチパス設定の編集	14
2.6. すべてのデバイスへのマルチパス値の設定	15
第3章 マルチパス化されたボリュームの管理	16
3.1. MULTIPATH コマンド	16
3.1.1. Multipath コマンドの出力	16
3.1.2. マルチパス設定の表示	17
3.2. オンラインのマルチパスデバイスのサイズ変更	18
3.3. ROOT ファイルシステムをシングルパスデバイスからマルチパスデバイスに移動	19
3.4. SWAP ファイルシステムをシングルパスデバイスからマルチパスデバイスに移動	20
3.5. DMSETUP コマンドを使用したデバイスマッパーエントリーの特定	21
3.6. MULTIPATHD デーモンの管理	21
3.7. パッケージ削除時におけるマルチパスファイルのクリーンアップ	22
第4章 DM MULTIPATH のトラブルシューティング	23
4.1. DM MULTIPATH トラブルシューティングチェックリスト	23
4.1.1. マルチパスデーモンが実行していない	23
4.1.2. queue_if_no_path 機能に関する問題	23
4.2. MULTIPATHD 対話式コンソールを使用したトラブルシューティング	23

RED HAT ドキュメントへのフィードバック

ドキュメントの改善に関するご意見やご要望をお聞かせください。

- 特定の文章に簡単なコメントを記入する場合は、ドキュメントが Multi-page HTML 形式になっているのを確認してください。コメントを追加する部分を強調表示し、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、表示された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 DM MULTIPATH の設定

システムに DM-Multipath をセットアップする前に、システムが更新されており、**device-mapper-multipath** パッケージがインストールされていることを確認してください。

1.1. DM-MULTIPATH の基本設定

mpathconf ユーティリティーを使用して DM Multipath を設定します。マルチパス設定ファイル **/etc/multipath.conf** が作成されます。

- **/etc/multipath.conf** ファイルがすでに存在する場合は、**mpathconf** ユーティリティーで編集します。
- **/etc/multipath.conf** ファイルが存在しない場合は、**mpathconf** ユーティリティーにより **/etc/multipath.conf** ファイルが作成されます。

mpathconf ユーティリティーの詳細は、man ページの **mpathconf(8)** を参照してください。

/etc/multipath.conf ファイルを編集する必要がない場合は、以下の **mpathconf** コマンドを実行すると、基本的なフェイルオーバー設定の DM Multipath をセットアップできます。このコマンドでマルチパス設定が有効になり、**multipathd** デーモンが起動します。

```
# mpathconf --enable --with_multipathd y
```

multipathd デーモンを起動する前に **/etc/multipath.conf** ファイルを編集する必要がある場合は、以下の手順に従って基本的なフェイルオーバー設定の DM Multipath をセットアップします。

1. **--enable** オプションを指定して **mpathconf** コマンドを入力します。

```
# mpathconf --enable
```

mpathconf コマンドに必要な追加オプションの詳細は、man ページの **mpathconf(8)** を参照するか、**--help** オプションを指定して **mpathconf** コマンドを実行してください。

```
# mpathconf --help
usage: /sbin/mpathconf <command>

Commands:
Enable: --enable
Disable: --disable
Set user_friendly_names (Default y): --user_friendly_names <y|n>
Set find_multipaths (Default y): --find_multipaths <y|n>
Load the dm-multipath modules on enable (Default y): --with_module <y|n>
start/stop/reload multipathd (Default n): --with_multipathd <y|n>
```

2. 必要に応じて **/etc/multipath.conf** ファイルを編集します。DM Multipath のデフォルト設定はシステムにコンパイルされているため、**/etc/multipath.conf** ファイルで明示的に指定する必要はありません。

path_grouping_policy のデフォルト値は **failover** に設定されているため、この例では **/etc/multipath.conf** ファイルを編集する必要はありません。

設定ファイル内で初めてのデフォルトセクションはシステムを設定するため、マルチパスデバイスの名前の形式は **/dev/mapper/mpathn** になります。この設定がないと、マルチパスデバイスの WWID がマルチパスデバイスの名前になります。ユーザーフレンドリー名を使用しない場

合は、以下のコマンドを入力します。

```
# mpathconf --enable --user_friendly_names n
```

3. 必要に応じて設定ファイルを保存し、エディターを終了します。
4. 以下のコマンドを実行します。

```
# systemctl start multipathd.service
```



注記

マルチパスデーモンを起動してからマルチパス設定ファイルの編集が必要になった場合は、**systemctl reload multipathd.service** コマンドを実行して変更を反映する必要があります。

1.2. マルチパスデバイスの作成時にローカルのディスクを無視

ローカルの SCSI カードが内蔵ディスクに搭載されているマシンがありますが、DM Multipath をこのようなデバイスで使用することは推奨されません。**find_multipaths** 設定パラメーターを **on** に設定すると、このようなリストをブラックリストに指定する必要がなくなります。**find_multipaths** 設定パラメーターと、これらのパラメーターに設定する値に関する詳細は、man ページの **multipath.conf(5)** を参照してください。

find_multipaths 設定パラメーターを **on** にセットしない場合は、以下の手順に従ってマルチパス設定ファイルを編集すると、マルチパスの設定時にローカルのディスクを無視できます。

1. どのディスクが内蔵ディスクかを確認し、そのディスクに、ブラックリストとして指定する印を付けます。
以下の例では **/dev/sda** が内蔵ディスクです。デフォルトのマルチパス設定ファイル内で元々設定されているように、**multipath -v2** を実行するとマルチパスマップ内にローカルディスク **/dev/sda** が表示されます。

この例では、**multipath** コマンドで **-d** オプションを指定し、マルチパスデバイスを作成しないドライ実行であることを示しています。

```
# multipath -v2 -d
: SIBM-ESXSST336732LC____F3ET0EP0Q000072428BX1 undef WINSYS,SF2372
size=33 GB features="0" hwhandler="0" wp=undef
`-- policy='round-robin 0' prio=1 status=undef
   |- 0:0:0:0 sda 8:0 [-----]
   `--

: 3600a0b80001327d80000006d43621677 undef WINSYS,SF2372
size=12G features='0' hwhandler='0' wp=undef
`-- policy='round-robin 0' prio=1 status=undef
   |- 2:0:0:0 sdb 8:16 undef ready running
   `-- 3:0:0:0 sdf 8:80 undef ready running

: 3600a0b80001327510000009a436215ec undef WINSYS,SF2372
size=12G features='0' hwhandler='0' wp=undef
`-- policy='round-robin 0' prio=1 status=undef
   |- 2:0:0:1 sdc 8:32 undef ready running
   `-- 3:0:0:1 sdg 8:96 undef ready running
```

```

: 3600a0b80001327d800000070436216b3 undef WINSYS,SF2372
size=12G features='0' hwhandler='0' wp=undef
`-- policy='round-robin 0' prio=1 status=undef
  |- 2:0:0:2 sdd 8:48 undef ready running
  `-- 3:0:0:2 sdg 8:112 undef ready running

: 3600a0b80001327510000009b4362163e undef WINSYS,SF2372
size=12G features='0' hwhandler='0' wp=undef
`+- policy='round-robin 0' prio=1 status=undef
  |- 2:0:0:3 sdd 8:64 undef ready running
  `-- 3:0:0:3 sdg 8:128 undef ready running

```

2. デバイスマッパーによる `/dev/sda` のマルチパスマップへのマッピングを阻止するため、`/etc/multipath.conf` ファイルの `blacklist` セクションを編集してこのデバイスを追加します。 `devnode` タイプで `sda` デバイスをブラックリストに指定することも可能ですが、`/dev/sda` が再起動後に同じく `/dev/sda` になる保証はないため安全な方法ではありません。個々のデバイスをブラックリストに指定する場合は、そのデバイスの WWID を使用します。`multipath -v2` コマンドの出力を見ると、`/dev/sda` デバイスの WWID は `SIBM-ESXSST336732LC___F3ET0EP0Q000072428BX1` になっています。このデバイスをブラックリストに指定する場合は `/etc/multipath.conf` ファイルに以下の記述を追加します。

```

blacklist {
    wwid SIBM-ESXSST336732LC___F3ET0EP0Q000072428BX1
}

```

3. `/etc/multipath.conf` ファイルを更新したら、`multipathd` デーモンにファイルを再ロードするよう、手動で指示する必要があります。以下のコマンドで、更新した `/etc/multipath.conf` ファイルを再ロードします。

```
# systemctl reload multipathd.service
```

1.3. 追加のストレージデバイスの設定

DM-Multipath に対応しているストレージアレイで最も一般的なものは、DM-Multipath の設定にデフォルトで含まれています。対応デバイスなど、デフォルトの設定値は、以下のいずれかのコマンドを実行して確認してください。

```
# multipathd show config
# multipath -t
```

デフォルトではサポートされないストレージデバイスを既知のマルチパスデバイスとして追加する必要がある場合は、`/etc/multipath.conf` ファイルを編集し、適切なデバイス情報を挿入します。

たとえば、HP Open-V シリーズに関する情報を追加する場合は、以下のようなエントリーを記述します。この例では、すべてのパスに障害が発生した場合には、1分間（または5秒ごとの再試行で12回の再試行を行う）のキュー待ちを行うようデバイスを設定しています。

```

devices {
    device {
        vendor "HP"
        product "OPEN-V"
        no_path_retry 12
    }
}

```

1.4. INITRAMFS ファイルシステムでマルチパスの設定

initramfs ファイルシステムでマルチパスを設定できます。マルチパスの設定後、以下のオプションを追加して **dracut** を実行すると、マルチパス設定ファイルで **initramfs** ファイルシステムを再構築できます。

```
# dracut --force --add multipath --include /etc/multipath
```

initramfs ファイルシステムからマルチパスを実行してマルチパスの設定ファイルを変更した場合は、**initramfs** ファイルシステムを再構築して変更を有効にする必要があります。

第2章 DM MULTIPATH 設定ファイルの編集

DM Multipath では、マルチパスで最も一般的に使用する設定値がデフォルトで提供されています。また DM Multipath をサポートする最も一般的なストレージレイが DM Multipath に含まれています。サポートされるデバイスなどのデフォルトの設定値は、以下のいずれかのコマンドを実行して確認してください。

```
# multipathd show config
# multipath -t
```

DM-Multipath のデフォルトの設定値は、`/etc/multipath.conf` 設定ファイルを編集するとオーバーライドできます。必要な場合は、デフォルトではサポートされないストレージレイを設定ファイルに追加することもできます。



注記

initramfs ファイルシステムでマルチパスの設定を行うことができます。**initramfs** ファイルシステムからマルチパスを実行してマルチパスの設定ファイルを変更した場合は、**initramfs** ファイルシステムを再構築して変更を有効にする必要があります。

マルチパス設定ファイルで指定するのは、使用状況に応じて必要とされるセクションや、デフォルト値の変更が必要なセクションのみです。使用環境には無関係なセクションや、デフォルト値を無効にする必要がないセクションは、初期ファイルに指定されているコメントアウトを削除する必要はありません。

設定ファイルでは、正規表現の記述構文を使用できます。

設定ファイルの詳細は、man ページの **multipath.conf(5)** を参照してください。

2.1. 設定ファイルの概要

マルチパス設定ファイルは以下のセクションに分かれています。

blacklist

マルチパス設定の対象として考慮しないデバイスの一覧。

blacklist_exceptions

ブラックリストの例外としてマルチパス設定の対象となるデバイスの一覧。この例外セクションに指定されていないければ、`blacklist` セクションのパラメーターに従ってブラックリストに指定されません。

defaults

DM Multipath のデフォルトの全般設定。

multipaths

マルチパスデバイスの特性に関する個別設定。ここで指定する値は、設定ファイルの **overrides**、**devices**、および **defaults** のセクションで指定されている値より優先されます。

devices

ストレージコントローラーの個別設定。ここで指定する値は、設定ファイル内の **defaults** セクションで指定されている値より優先されます。デフォルトでは対応していないストレージレイを使用している場合は、そのレイ用の **devices** サブセクションを作成する必要があります。

overrides

すべてのデバイスに適用される設定。ここで指定する値は、設定ファイルの **devices** および **defaults** のセクションで指定されている値より優先されます。

システムでマルチパスデバイスの属性が確認される場合は、multipath セクション、devices セクション、defaults セクションの順に設定がチェックされます。

2.2. DM MULTIPATH からのデバイスのブラックリスト

マルチパス設定ファイルの **blacklist** セクションには、システムがマルチパスデバイスを設定する際に使用しないデバイスを指定します。ブラックリストに指定したデバイスは、マルチパスデバイスには含まれません。

find_multipaths 設定パラメーターを **off** に設定すると、マルチパスは明示的にブラックリストに指定されていないパスに対して常にマルチパスデバイスを作成しようとします。**find_multipaths** 設定パラメーターが **on** に設定されている場合は、3つの条件のうち1つを満たす場合に限り、マルチパスによりデバイスが作成されます。

- ブラックリストに指定されておらず、同一の WWID を持つパスが少なくとも2つある場合。
- ユーザーが **multipath** コマンドでデバイスを指定して、デバイスの作成を手動で強制する場合。
- 任意のパスが、以前に作成されたマルチパスデバイスと同一の WWID を持っている場合 (そのマルチパスデバイスがその時点で存在していない場合も含む)。マルチパスデバイスを作成すると、常にそのデバイスの WWID が記憶されるため、マルチパスは、同一の WWID を持つパスを認識すると、自動的にそのデバイスを再度作成します。これにより、マルチパスのブラックリストを編集しなくても、マルチパスが自動的に正しいパスを選択してマルチパスデバイスにできます。

find_multipaths パラメーターを使用しないでマルチパスデバイスを作成してから、パラメーターを **on** に設定した場合は、**/etc/multipath/wwids** ファイルから、マルチデバイスとして作成しないデバイスの WWID を削除しなければならないことがあります。以下は、**/etc/multipath/wwids** ファイルの例を示しています。スラッシュ (/) で囲まれている部分が WWID です。

```
# Multipath wwids, Version : 1.0
# NOTE: This file is automatically maintained by multipath and multipathd.
# You should not need to edit this file in normal circumstances.
#
# Valid WWIDs:
/3600d0230000000000e13955cc3757802/
/3600d0230000000000e13955cc3757801/
/3600d0230000000000e13955cc3757800/
/3600d02300069c9ce09d41c31f29d4c00/
/SWINSYS SF2372    0E13955CC3757802/
/3600d0230000000000e13955cc3757803/
```

find_multipaths には、**on** と **off** 以外に以下の値を設定することもできます。

- **strict** - マルチパスは、マルチパス化されていなかったパスを許可しないため、そのパスは **/etc/multipath/wwids** ファイルに記載されていません。
- **smart** - マルチパスは常に **udev** のブラックリストに指定されていないデバイスを許可しますが、**find_multipaths_timeout** パラメーターで設定されたタイムアウト期間内に **multipathd** がデバイスを作成しないとそのクレームをデバイス上で解放します。**find_multipaths_timeout** パラメーターの詳細は、man ページの **multipath.conf(5)** を参照してください。

`find_multipaths` の本来のデフォルト値は **off** ですが、`mpathconf` により作成されたデフォルトの `multipath.conf` ファイルは `find_multipaths` の値を **on** に設定します。

`find_multipaths` に設定できる値の詳細は、man ページの `multipath.conf(5)` を参照してください。

`find_multipaths` パラメーターを **on** に設定した場合は、マルチパス化しない複数のパスを持つデバイスのみをブラックリストに指定する必要があります。そのため、通常はデバイスをブラックリストに指定する必要がありません。

デバイスをブラックリストに指定する必要がある場合は、WWID、デバイス名、デバイスタイプ、プロパティ、およびプロトコルでデバイスをブラックリストに指定できます。これら5つの基準は各デバイスで「property、devnode、device、protocol、wwid」の順に評価されます。いずれかの基準によりデバイスがブラックリストに指定されていることが判明した場合は、`multipathd` により処理から除外され、それ以降の基準は評価されません。各基準において、デバイスがホワイトリストとブラックリストの両方に一致した場合はホワイトリストが優先されます。

デフォルトでは、設定ファイルの初期ブラックリストセクションをコメントアウトした後でも、各種のデバイスタイプがブラックリストに指定されます。詳細は「[デバイス名でのブラックリストの指定](#)」を参照してください。

2.2.1. WWID でブラックリストの指定

設定ファイルの `blacklist` セクションで `wwid` エントリーを使用し、World-Wide IDentification でデバイスをブラックリストに指定できます。

以下の例では、WWID が 26353900f02796769 であるデバイスをブラックリストに指定します。

```
blacklist {
    wwid 26353900f02796769
}
```

2.2.2. デバイス名でのブラックリストの指定

設定ファイルの `blacklist` セクションで `devnode` エントリーを使用し、デバイス名でデバイスタイプをブラックリストに指定し、マルチパスデバイスに含まれないようにできます。

以下の例では、`sd*` のデバイスをすべてブラックリストに指定することで、すべての SCSI デバイスをブラックリストに指定します。

```
blacklist {
    devnode "^sd[a-z]"
}
```

`blacklist` セクションで `devnode` エントリーを使用すると、特定タイプに該当する全デバイスではなく、個々のデバイスをブラックリストに指定できます。ただし、デバイスが `udev` ルールで静的にマッピングされていない限り、再起動後にそのデバイス名が同じ名前になる保証がないため、この方法は推奨できません。たとえば、再起動するとデバイス名が `/dev/sda` から `/dev/sdb` に変わる可能性があります。

デフォルトでは、以下の `devnode` エントリーがデフォルトのブラックリストにコンパイルされています。これらのエントリーによりブラックリストに指定されるデバイスは、通常 DM Multipath をサポートしません。これらのデバイスでマルチパスを有効にするには、「[ブラックリストの例外](#)」の説明に従って、設定ファイルの `blacklist_exceptions` セクションにこれらのデバイスを指定する必要があります。

```
blacklist {
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
    devnode "^(td|ha)d[a-z]"
}
```

2.2.3. デバイスタイプでのブラックリストの指定

設定ファイルの **blacklist** セクションに **device** セクションを使用して、特定のデバイスタイプを指定できます。以下の例では、IBM DS4200 デバイスと HP デバイスをすべてブラックリストに指定します。

```
blacklist {
    device {
        vendor "IBM"
        product "3S42"    #DS4200 Product 10
    }
    device {
        vendor "HP"
        product ".*"
    }
}
```

2.2.4. udev プロパティでのブラックリスト指定

multipath.conf 設定ファイルの **blacklist** および **blacklist_exceptions** セクションは **property** パラメーターをサポートします。このパラメーターを使用すると、ユーザーは特定タイプのデバイスをブラックリストに指定できます。**property** パラメーターは、デバイスの **udev** 環境変数名と照合される正規表現の文字列を取ります。

以下の例は、udev プロパティ **ID_ATA** を持つすべてのデバイスをブラックリストに指定します。

```
blacklist {
    property "ID_ATA"
}
```

2.2.5. デバイスプロトコルでのブラックリストの指定

protocol セクションのある設定ファイルの **blacklist** セクションに、マルチパス化から除外されるデバイスのプロトコルを指定できます。マルチパスが認識するプロトコル文字列は `scsi:fc`、`scsi:spi`、`scsi:ssa`、`scsi:sbp`、`scsi:srp`、`scsi:iscsi`、`scsi:sas`、`scsi:adt`、`scsi:ata`、`scsi:unspec`、`ccw`、`cciss`、`nvme`、および `undef` です。パスが使用しているプロトコルを表示するには、**multipathd show paths format "%d %P"** コマンドを実行します。

以下の例は、未定義のプロトコルまたは未知の SCSI トランスポートタイプを持つすべてのデバイスをブラックリストに指定します。

```
blacklist {
    protocol "scsi:unspec"
    protocol "undef"
}
```

2.2.6. ブラックリストの例外

設定ファイルの **blacklist_exceptions** セクションを使用すると、デフォルトではブラックリストに指定されているデバイスで、マルチパスを有効にできます。

たとえば、デバイスが多数あり、その中の1つのデバイス (以下の例では WWID が 3600d023000000000000e13955cc3757803 のデバイス) でのみマルチパスを有効にする場合は、有効にするデバイス以外のものを1つ1つブラックリストに指定するのではなく、一旦すべてのデバイスをブラックリストに指定してから、**/etc/multipath.conf** ファイルに以下の行を追加します。これにより、目的のデバイスにだけマルチパスを許可できます。

```
blacklist {
    wwid ".*"
}

blacklist_exceptions {
    wwid "3600d023000000000000e13955cc3757803"
}
```

設定ファイルの **blacklist_exceptions** セクションでデバイスを指定する場合は、**blacklist** で指定したのと同じ方法で指定しなければなりません。たとえば、**blacklist** で **devnode** エントリーを使用してデバイスを指定している場合は、**exception** で除外指定に使用している WWID が正しくても適用されません。同様に **devnode** による除外は、**devnode** エントリーでブラックリストを指定したデバイスにしか適用されず、**device** による除外は、**device** エントリーでブラックリストを指定したデバイスにしか適用されません。

property パラメーターの挙動は、他の **blacklist_exception** パラメーターとは異なります。このパラメーターが設定された場合、デバイスには一致する **udev** 変数が必要になります。この変数がないと、デバイスはブラックリストに指定されます。このパラメーターを使用すると、USB スティックやローカルハードドライブなど、マルチパスが無視する必要のある SCSI デバイスをブラックリストに登録できます。合理的にマルチパス化できる SCSI デバイスのみを許可するには、以下の例のようにこのパラメーターを **SCSI_IDENT_ID_WWN** に設定します。

```
blacklist_exceptions {
    property "(SCSI_IDENT_ID_WWN)"
}
```

2.3. マルチパス設定ファイルのデフォルトの編集

/etc/multipath.conf 設定ファイルには **defaults** セクションがあり、以下のように **user_friendly_names** パラメーターを **yes** にセットできます。

```
defaults {
    user_friendly_names yes
}
```

上記は、**user_friendly_names** パラメーターのデフォルト値を上書きします。

設定ファイルに **defaults** のテンプレートが含まれています。このセクションは、以下のようにコメントアウトされています。

```
#defaults {
#    polling_interval    10
#    path_selector       "round-robin 0"
#    path_grouping_policy multibus
#    uid_attribute       ID_SERIAL
```



```
# prio          alua
# path_checker  readsector0
# rr_min_io     100
# max_fds       8192
# rr_weight     priorities
# failback      immediate
# no_path_retry fail
# user_friendly_names yes
#}
```

設定パラメーターのデフォルト値を上書きする場合は、このテンプレートの該当行を **defaults** セクションにコピーして、コメントを外します。たとえば、**path_grouping_policy** パラメーターを、デフォルト値の **failover** から **multibus** に上書きするには、以下のように、テンプレートで該当行を見つけて設定ファイルの **defaults** セクションにコピーし、そのコメントを外します。

```
defaults {
    user_friendly_names yes
    path_grouping_policy multibus
}
```

multipath.conf 設定ファイルの **defaults** セクションで設定される属性の詳細は、man ページの **multipath.conf(5)** を参照してください。**multipath.conf** ファイルの **devices**、**multipaths**、または **overrides** の各セクションに指定された属性により上書きされた場合を除き、これらの値は DM Multipath により使用されます。

2.4. 特定デバイスのマルチパス設定の編集

multipath.conf 設定ファイルの **multipaths** セクションにある属性は、指定された1つのマルチパスにのみ適用されます。これらのデフォルトは DM Multipath により使用され、**multipath.conf** ファイルの **overrides**、**defaults**、および **devices** セクションに設定された属性を上書きします。

multipath.conf 設定ファイルの **defaults** セクションで設定される属性の詳細は、man ページの **multipath.conf(5)** を参照してください。

設定ファイルで、2つの特定マルチパスデバイスに対して **multipath** 属性を指定している例を以下に示します。1番目のデバイスの WWID は **3600508b4000156d70001200000b0000** で、シンボリック名 **yellow** があります。

この例では、2番目のマルチパスデバイスの WWID は **1DEC_321816758474** で、シンボリック名は **red** です。**rr_weight** 属性は **priorities** に設定されます。

```
multipaths {
    multipath {
        wwid          3600508b4000156d70001200000b0000
        alias         yellow
        path_grouping_policy multibus
        path_selector "round-robin 0"
        failback      manual
        rr_weight     priorities
        no_path_retry 5
    }
    multipath {
        wwid          1DEC_321816758474
        alias         red
    }
}
```

```

    rr_weight    priorities
  }
}

```

2.5. ストレージコントローラーのマルチパス設定の編集

multipath.conf 設定ファイルの **devices** セクションは、各ストレージデバイスの属性を設定します。デバイスを含むパスの **multipath.conf** ファイルの **multipaths** セクションまたは **overrides** セクションに指定された属性により上書きされた場合を除き、これらの値は DM Multipath により使用されます。これらの属性は、**multipath.conf** ファイルの **defaults** セクションに設定された属性を上書きします。

multipath.conf 設定ファイルの **devices** セクションで設定される属性の詳細は、man ページの **multipath.conf(5)** を参照してください。

マルチパスに対応しているデバイスの多くは、デフォルトでマルチパスの設定に含まれています。対応デバイスなど、デフォルトの設定値は、以下のいずれかのコマンドを実行して確認してください。

```

# multipathd show config
# multipath -t

```

おそらく、こうしたデバイスの値を修正する必要はありませんが、必要が生じた場合は、設定ファイルにエントリーを追加すればデフォルト値を上書きできます。**multipathd show config** コマンドが表示する、目的のデバイスのデフォルト値をコピーして、変更したい値に書き換えることができます。

設定ファイルのこのセクションに、デフォルトでは自動で設定されないデバイスを追加する場合は、**vendor** パラメーターおよび **product** パラメーターを設定する必要があります。以下の例のように、これらの値は **/sys/block/device_name/device/vendor** および **/sys/block/device_name/device/model** で見つけることができます。*device_name* はマルチパス化するデバイスになります。

```

# cat /sys/block/sda/device/vendor
WINSYS
# cat /sys/block/sda/device/model
SF2372

```

追加で指定するパラメーターは、使用するデバイスにより異なります。デバイスが **active/active** の場合は通常、追加でパラメーターを設定する必要はありません。**path_grouping_policy** の **multibus** への設定が必要になることがあります。この他に、設定が必要となる可能性があるパラメーターは **no_path_retry** と **rr_min_io** です。

デバイスが **active/passive** であるにも関わらず、I/O を持つパスが自動的に **passive** パスに切り替えられる場合は、チェッカーの関数を、I/O をパスに送信しない関数に変更し、それが正しく動作するかどうかを検証する必要があります (これを行わないとデバイスはフェイルオーバーし続けます)。つまり、ほとんどの場合は、**path_checker** を **tur** に設定するということになります。この設定は、Test Unit Ready コマンドに対応するすべての SCSI デバイスで機能します。Test Unit Ready コマンドは大半のデバイスで対応しています。

パスの切り替えに特殊なコマンドを必要とするデバイスにマルチパスを設定するには、ハードウェアハンドラーカーネルモジュールが必要になります。現在、利用可能なハードウェアハンドラーは **emc** です。このハンドラーが目的のデバイスに使用できない場合は、そのデバイスにマルチパスを設定できない可能性があります。

マルチパス設定ファイルの **device** エントリーの例を以下に示します。

```

# }

```

```
# device {  
# vendor "COMPAQ "  
# product "MSA1000      "  
# path_grouping_policy multibus  
# path_checker tur  
# rr_weight priorities  
#}  
#}
```

2.6. すべてのデバイスへのマルチパス値の設定

multipath.conf 設定ファイルの **overrides** セクションでは、すべてのデバイスの設定値を設定できます。たとえば、すべてのデバイスで **no_path_retry** を **fail** に設定する場合があります。このセクションは、**multipath.conf** 設定ファイルの **devices** セクションおよび **defaults** セクションの両方によりサポートされるすべての属性をサポートします。これは、**vendor**、**product**、および **revision** 以外の **devices** セクション属性になります。このデバイスを含むパスの **multipath.conf** ファイルで、**multipaths** セクションに指定された属性により上書きされた場合を除き、これらの属性はすべてのデバイスの DM Multipath により使用されます。これらの属性は、**multipath.conf** ファイルの **devices** セクションおよび **defaults** セクションに設定された属性を上書きします。

multipath.conf 設定ファイルの **devices** セクションおよび **defaults** セクションで設定される属性の詳細は、man ページの **multipath.conf(5)** を参照してください。

第3章 マルチパス化されたボリュームの管理

DM Multipath は、マルチパスボリュームの管理に使用できるさまざまなツールとコマンドを提供します。

3.1. MULTIPATH コマンド

multipath コマンドは、デバイスの複数のパスを検出および結合するために使用されます。マルチパス化されたデバイスの管理に使用できるさまざまなオプションを提供します。

表3.1「便利な **multipath** コマンドオプション」では、**multipath** コマンドの便利なオプションの一部を説明します。

表3.1 便利な **multipath** コマンドオプション

オプション	説明
-l	sysfs およびデバイスマッパーから取得した現在のマルチパス設定を表示します。
-ll	sysfs およびデバイスマッパーと、システムで利用可能なその他の全コンポーネントから取得した現在のマルチパス設定を表示します。
-f device	指定されたマルチパスデバイスを削除します。
-F	不要なマルチパスデバイスをすべて削除します。
-w device	wwids ファイルから指定デバイスの wwid を削除します。
-W	現在のマルチパスデバイスのみが含まれるように、 wwids ファイルをリセットします。

3.1.1. Multipath コマンドの出力

マルチパスデバイスの作成、修正、一覧表示を行う際、現在のデバイス設定が表示されます。形式を以下に示します。

マルチパスデバイス

```
action_if_any: alias (wwid_if_different_from_alias) dm_device_name_if_known
vendor,product size=size features='features' hwhandler='hardware_handler'
wp=write_permission_if_known
```

パスグループ

```
-- policy='scheduling_policy' prio=prio_if_known status=path_group_status_if_known
```

パス

```
\`- host:channel:id:lun devnode major:minor dm_status_if_known path_status online_status
```

マルチパスコマンドの出力例を以下に示します。

```
3600d0230000000000e13955cc3757800 dm-1 WINSYS,SF2372
size=269G features='0' hwhandler='0' wp=rw
|+- policy='round-robin 0' prio=1 status=active
|`- 6:0:0:0 sdb 8:16 active ready running
\`+- policy='round-robin 0' prio=1 status=enabled
  `-- 7:0:0:0 sdf 8:80 active ready running
```

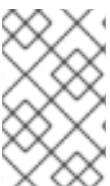
パスが有効になり、I/O が可能になると、パスの状態が **ready** または **ghost** になります。パスが停止している場合は、状態が **faulty** または **shaky** になります。パスの状態は、`/etc/multipath.conf` ファイルに定義されているポーリング間隔に基づき、**multipathd** デーモンにより定期的に更新されます。

その他の可能なパス状態値は以下のとおりです。

- **i/o pending** - チェッカーがアクティブにこのパスをチェックし、状態はまもなく更新されます。
- **i/o timeout - faulty** と同じです。タイムアウトする前にチェッカーが成功または失敗のいずれかを返さなかったことをユーザーに通知します。
- **removed** - システムからパスが削除され、マルチパスデバイスからまもなく削除されます。**faulty** と同様に処理されます。
- **wild** - 内部エラーまたは設定の問題によって、**multipathd** はパスチェッカーを実行できませんでした。**faulty** とほぼ同じですが、マルチパスはパス上で多くのアクションをスキップします。
- **unchecked** - パスが今検出された、割り当てられたパスチェッカーがない、パスチェッカーにエラーが発生したなどの理由で、パスチェッカーがこのパス上で実行されませんでした。これは **wild** と同様に処理されます。
- **delayed** - パスチェッカーはパスが有効であると返したにも関わらず、パスが複数回失敗し、パスを遅延するようマルチパスが設定されたため、マルチパスがパスの復元を遅延しています。

dm の状態はパスの状態と似ていますが、視点はカーネルのものになります。dm 状態が **active** の場合は、パスの状態が **ready** および **ghost** になります。**pending** パスの状態と同等の dm の状態はありません。その他のパスの状態はすべて、dm の状態では **failed** となります。dm 状態は、パスチェッカーが完了するまで現在のステータスを保持します。

online_status に使用できる値は **running** と **offline** です。**offline** は、この SCSI デバイスが無効になっていることを表します。



注記

マルチパスデバイスを作成中または修正中の場合は、パスグループの状態、dm のデバイス名、書き込みパーミッション、dm の状態などが不明です。また、機能も常に正しいとは限りません。

3.1.2. マルチパス設定の表示

multipath コマンドで **-l** および **-ll** オプションを使用すると、現在のマルチパス設定を表示できます。**-l**

オプションは、**sysfs** の情報から収集したマルチパスポートロギーと、デバイスマッパーを表示します。**-ll** オプションは、**-l** オプションが表示する情報の他に、使用できる他のすべてのシステムコンポーネントを表示します。

マルチパス設定を表示するとき、**multipath** コマンドの **-v** オプションを使用すると詳細度を指定できます。**-v0** を指定すると何も出力されません。**-v1** を指定すると、作成または更新されたマルチパス名のみが出力され、**kpartx** などの別のツールに提供できます。**-v2** を指定すると、検出されたパス、マルチパス、およびデバイスマップがすべて出力されます。より詳細な情報を出力する場合は、**-v3**、**-v4**、または **-v5** を指定することもできます。

以下は、**multipath -l** コマンドの出力例になります。

```
# multipath -l
3600d0230000000000e13955cc3757800 dm-1 WINSYS,SF2372
size=269G features='0' hwhandler='0' wp=rw
|-- policy='round-robin 0' prio=1 status=active
|`- 6:0:0:0 sdb 8:16 active ready running
`-- policy='round-robin 0' prio=1 status=enabled
   `- 7:0:0:0 sdf 8:80 active ready running
```

以下は、**multipath -ll** コマンドの出力例になります。

```
# multipath -ll
3600d0230000000000e13955cc3757801 dm-10 WINSYS,SF2372
size=269G features='0' hwhandler='0' wp=rw
|-- policy='round-robin 0' prio=1 status=enabled
|`- 19:0:0:1 sdc 8:32 active ready running
`-- policy='round-robin 0' prio=1 status=enabled
   `- 18:0:0:1 sdh 8:112 active ready running
3600d0230000000000e13955cc3757803 dm-2 WINSYS,SF2372
size=125G features='0' hwhandler='0' wp=rw
`+-- policy='round-robin 0' prio=1 status=active
    |- 19:0:0:3 sde 8:64 active ready running
    `- 18:0:0:3 sdj 8:144 active ready running
```

3.2. オンラインのマルチパスデバイスのサイズ変更

オンラインのマルチパスデバイスのサイズを変更する必要がある場合は、以下の手順に従ってください。

1. 物理デバイスのサイズを変更します。
2. 以下のコマンドを実行して、LUN へのパスを検索します。

```
# multipath -l
```

3. パスのサイズを変更します。SCSI デバイスの場合は、デバイスの **rescan** ファイルに 1 と書き込むと、SCSI ドライバーによる再スキャンが行われます。以下にコマンド例を示します。

```
# echo 1 > /sys/block/path_device/device/rescan
```

各パスデバイスに対してこのコマンドを実行します。たとえば、パスデバイスが **sda**、**sdb**、**sde**、および **sdf** の場合は、以下のコマンドを実行します。

```
# echo 1 > /sys/block/sda/device/rescan
# echo 1 > /sys/block/sdb/device/rescan
# echo 1 > /sys/block/sde/device/rescan
# echo 1 > /sys/block/sdf/device/rescan
```

4. **multipathd resize** コマンドを実行して、マルチパスデバイスのサイズを変更します。

```
# multipathd resize map multipath_device
```

5. ファイルシステムのサイズを変更します (LVM または DOS のパーティションが使用されていないことを前提とします)。

```
# resize2fs /dev/mapper/mpatha
```

3.3. ROOT ファイルシステムをシングルパスデバイスからマルチパスデバイスに移動

シングルパスのデバイスにシステムをインストールしてから、別のパスを root ファイルシステムに追加する場合は、root ファイルシステムをマルチパスのデバイスに移行する必要があります。本セクションでは、シングルパスからマルチパスのデバイスへの移行手順を説明します。

device-mapper-multipath パッケージがインストールされていることを確認した上で、以下の手順を実行します。

1. 以下のコマンドを実行して **/etc/multipath.conf** 設定ファイルを作成し、マルチパスモジュールをロードして **multipathd** の **chkconfig** を **on** に設定します。

```
# mpathconf --enable
```

2. **find_multipaths** 設定パラメーターが **yes** に設定されていない場合は、「[DM Multipath からのデバイスのブラックリスト](#)」の説明に従って、**/etc/multipath.conf** ファイルの **blacklist** セクションおよび **blacklist_exceptions** セクションを変更します。
3. 検出され次第、root デバイスにマルチパスデバイスを構築させるため、次のコマンドを実行します。また、このコマンドを実行すると、パスが1つしかない場合でも必ずマルチパスデバイスが構築されるようになります。

```
# multipath -a root_devname
```

たとえば、root デバイスが **/dev/sdb** の場合は、次のコマンドを実行します。

```
# multipath -a /dev/sdb
wwid '3600d02300069c9ce09d41c4ac9c53200' added
```

4. 設定ファイルの設定が正しく行われたことを確認するため、**multipath** コマンドを実行して、以下の形式の行の出力を見つけます。これは、コマンドがマルチパスデバイスの作成に失敗したことを示しています。

```
date wwid: ignoring map
```

たとえば、デバイスの WWID が **3600d02300069c9ce09d41c4ac9c53200** の場合は、次のような行が出力に表示されます。

multipath

Oct 21 09:37:19 | 3600d02300069c9ce09d41c4ac9c53200: ignoring map

5. **multipath** で **initramfs** ファイルシステムを再構築するため、以下のオプションを使用して **dracut** コマンドを実行します。

dracut --force -H --add multipath

6. マシンをシャットダウンします。
7. マシンから他のパスが見えるように、FC スイッチを設定します。
8. マシンを起動します。
9. root ファイルシステム ('/') が、マルチパスを設定したデバイス上にあることを確認します。

3.4. SWAP ファイルシステムをシングルパスデバイスからマルチパスデバイスに移動

デフォルトでは、swap デバイスは論理ボリュームとして設定されます。論理ボリュームグループを構成する物理ボリュームでマルチパスを設定している限り、これらのデバイスをマルチパスデバイスとして設定する特別な手順は必要ありません。ただし、swap デバイスが LVM ボリュームではなく、デバイス名でマウントする場合には、**/etc/fstab** ファイルに、適切なマルチパスデバイス名を設定しないといけない場合があります。

1. **-v3** オプションを使用して **/sbin/multipath** コマンドを実行し、swap デバイスの WWID 番号を確認します。コマンドの出力のパス一覧に swap デバイスが表示されます。swap デバイスを示す、次のような行のコマンド出力を探します。

WWID H:B:T:L devname MAJOR:MINOR

たとえば、swap ファイルシステムを **sda** か、そのパーティションの1つに設定している場合は、以下のような行が表示されます。

```
===== paths list =====
...
1ATA   WDC WD800JD-75MSA3           WD-WMAM9F 1:0:0:0 sda 8:0
...
```

2. **/etc/multipath.conf** ファイルに、swap デバイスのエイリアスを設定します。

```

multipaths {
  multipath {
    wwid WWID_of_swap_device
    alias swapdev
  }
}
```

3. **/etc/fstab** で、root デバイスへの古いデバイスパスを、マルチパスデバイスに置き換えます。例として、**/etc/fstab** ファイルに、以下のようなエントリーがあるとします。

```
/dev/sda2 swap          swap defaults    0 0
```


そのエントリーを以下のように変更します。

```
/dev/mapper/swapdev swap      swap defaults    0 0
```

3.5. DMSETUP コマンドを使用したデバイスマッパーエントリーの特定

dmsetup コマンドを使用すると、マルチパスを設定したデバイスに一致するデバイスマッパーのエントリーを検索できます。

以下のコマンドでは、すべてのデバイスマッパーデバイスと、そのメジャー番号およびマイナー番号を表示します。dm デバイス名は、マイナー番号で特定できます。たとえば、マイナー番号 3 は、マルチパスを設定したデバイス **/dev/dm-3** になります。

```
# dmsetup ls
mpathd (253:4)
mpathep1 (253:12)
mpathfp1 (253:11)
mpathb (253:3)
mpathgp1 (253:14)
mpathhp1 (253:13)
mpatha (253:2)
mpathh (253:9)
mpathg (253:8)
VolGroup00-LogVol01 (253:1)
mpathf (253:7)
VolGroup00-LogVol00 (253:0)
mpathe (253:6)
mpathbp1 (253:10)
mpathd (253:5)
```

3.6. MULTIPATHD デーモンの管理

multipathd コマンドを使用して、**multipathd** デーモンを管理できます。利用可能な **multipathd** コマンドは、man ページの **multipathd(8)** を参照してください。

以下のコマンドは、**multipathd show maps** コマンドの出力の標準デフォルト形式を示しています。

```
# multipathd show maps
name sysfs uuid
mpathc dm-0 360a98000324669436c2b45666c567942
```

一部の **multipathd** コマンドには、後にワイルドカードが付いた **format** オプションを含むものがあります。次のコマンドを実行すると、使用できるワイルドカードの一覧を表示できます。

```
# multipathd show wildcards
```

multipathd コマンドは、「raw」形式のバージョンでマルチパスデバイスとパスの状態を表示する、**format** コマンドをサポートしています。raw 形式ではヘッダーは出力されず、フィールドがパディングされていないため、列とヘッダーが調整されていません。代わりに、フィールドは **format** 文字列の指定どおりに出力します。このため、出力はスクリプトで使いやすくなります。**multipathd show wildcards** コマンドを使用すると、**format** 文字列で使用されるワイルドカードを表示できます。

以下の **multipathd** コマンドは、format 文字列でマルチパスのワイルドカードを使用し、**multipathd** が監視しているマルチパスデバイスを通常の形式と raw 形式で表示します。

```
list|show maps|multipaths format $format
list|show maps|multipaths raw format $format
```

以下の **multipathd** コマンドは、format 文字列でマルチパスのワイルドカードを使用し、**multipathd** が監視しているパスを通常の形式と raw 形式で表示します。

```
list|show paths format $format
list|show paths raw format $format
```

以下のコマンドは、**multipathd show maps** の出力を、raw 形式と raw でない形式で示した場合の違いを表示します。**raw** 形式ではヘッダーがなく、単一のスペースのみが列の間にあることに注意してください。

```
# multipathd show maps format "%n %w %d %s"
name uuid sysfs vend/prod/rev
mpathc 360a98000324669436c2b45666c567942 dm-0 NETAPP,LUN

# multipathd show maps raw format "%n %w %d %s"
mpathc 360a98000324669436c2b45666c567942 dm-0 NETAPP,LUN
```

3.7. パッケージ削除時におけるマルチパスファイルのクリーンアップ

device-mapper-multipath rpm. ファイルを削除して

も、**/etc/multipath.conf**、**/etc/multipath/bindings**、および **/etc/multipath/wwids** の各ファイルは削除されないことに注意してください。これらのファイルは、**device-mapper-multipath** パッケージの次回インストール時に手動で削除する必要があります。

第4章 DM MULTIPATH のトラブルシューティング

マルチパス設定の実装に問題がある場合は、さまざまな項目を確認できます。

4.1. DM MULTIPATH トラブルシューティングチェックリスト

マルチパス設定の反応が遅かったり、機能しない場合は、以下の問題が原因として考えられます。

4.1.1. マルチパスデーモンが実行していない

マルチパス設定の実装に問題があることが判明した場合は、[1章DM Multipath の設定](#)の説明に従って、マルチパスデーモンが実行していることを確認する必要があります。

マルチパスが設定されたデバイスを使用するには、**multipathd** デーモンが実行している必要があります。

4.1.2. queue_if_no_path 機能に関する問題

features "1 queue_if_no_path" でマルチパスデバイスを設定した場合は、1つ以上のパスが復旧するまで、I/O を実行するプロセスがすべてハングします。これを回避するには、**/etc/multipath.conf** ファイルに **no_path_retry N** パラメーターを設定します (**N** はシステムでパスを再試行する回数に置き換えます)。

features "1 queue_if_no_path" オプションを使用する必要があり、ここで説明する問題が発生した場合は、特定の LUN に対して起動時にキューイングポリシーを無効にできます (すべてのパスが利用できません)。以下のコマンドは特定デバイスのキューイングを無効にします。

```
multipathd disablequeueing map device
```

以下のコマンドはすべてのデバイスのキューイングを無効にします。

```
multipathd disablequeueing maps
```

デバイスのキューイングを無効にしたら、**multipathd** が再起動または再ロードされるか、以下のコマンドのいずれかを実行するまで、そのデバイスは無効になります。

以下のコマンドは、特定デバイスのキューイングを以前の値にリセットします。

```
multipathd restorequeueing map device
```

以下のコマンドは、すべてのデバイスのキューイングを以前の値にリセットします。

```
multipathd restorequeueing maps
```

4.2. MULTIPATHD 対話式コンソールを使用したトラブルシューティング

multipathd -k コマンドは、**multipathd** デーモンに対する対話式のインターフェースです。このコマンドを実行すると対話式のマルチパスコンソールが立ち上がります。このコマンドを実行してから **help** を入力すると、使用可能なコマンドの一覧が表示され、対話式コマンドを入力できます。**CTRL-D** を入力すると終了します。

multipathd 対話式コンソールを使用すると、システムで発生している問題のトラブルシューティングを行うことができます。たとえば、以下のようなコマンドを順番に入力すると、デフォルト値などのマルチパス設定を表示して、その後コンソールを終了します。

```
# multipathd -k  
> > show config  
> > CTRL-D
```

以下のコマンドシーケンスを実行すると、**multipath.conf** への変更が確実に反映されます。

```
# multipathd -k  
> > reconfigure  
> > CTRL-D
```

以下のコマンドシーケンスを実行し、パスチェッカーが正しく動作していることを確認します。

```
# multipathd -k  
> > show paths  
> > CTRL-D
```