



# Red Hat Enterprise Linux 7 DM Multipath

---

DM Multipath 的配置與管理



## DM Multipath 的配置與管理

## 法律聲明

Copyright © 2015 Red Hat, Inc. and others.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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.

## 摘要

本書提供了在 Red Hat Enterprise Linux 7 中，使用 Device-Mapper Multipath 功能的資訊。

## 內容目錄

<b>章 1. Device Mapper Multipathing</b> .....	<b>2</b>
1.1. 新功能與更改的功能	2
1.2. DM Multipath 總覽	2
1.3. 儲存陣列的支援	5
1.4. DM Multipath 元件	5
1.5. DM Multipath 設定總覽	6
<b>章 2. Multipath 裝置</b> .....	<b>7</b>
2.1. Multipath 的裝置識別子	7
2.2. 叢集裡一致的 Multipath 裝置名稱	7
2.3. Multipath 裝置屬性	8
2.4. 邏輯卷冊中的 multipath 裝置	8
<b>章 3. 設定 DM Multipath</b> .....	<b>10</b>
3.1. 設定 DM Multipath	10
3.2. 產生 multipath 裝置時，忽略本機磁碟	11
3.3. 配置儲存裝置	13
3.4. 在 initramfs 檔案系統中設定 multipath	13
<b>章 4. DM Multipath 配置檔案</b> .....	<b>14</b>
4.1. 配置檔案總覽	14
4.2. 配置檔案的黑名單	15
4.3. 配置檔案的預設值	17
4.4. Multipath 裝置配置的屬性	23
4.5. 配置檔案裝置	25
<b>章 5. DM Multipath 管理與疑難排解</b> .....	<b>31</b>
5.1. 透過 Multipath Helper 自動產生配置檔案	31
5.2. 重設線上 Multipath 裝置的大小	31
5.3. 將 root 檔案系統從單路徑裝置移到多路徑裝置	31
5.4. 將 swap 檔案系統從單路徑裝置移到多路徑裝置	32
5.5. Multipath Daemon	33
5.6. 大量 LUN 的問題	33
5.7. queue_if_no_path 功能上的問題	33
5.8. Multipath 指令的輸出	34
5.9. 使用 multipath 指令來進行 Multipath 查詢	34
5.10. Multipath 的指令選項	35
5.11. 使用 dmsetup 指令，決定 device mapper 的項目	35
5.12. 使用 multipathd 的互動式主控台進行疑難排解	36
<b>附錄 A. 修訂記錄</b> .....	<b>37</b>
<b>索引</b> .....	<b>38</b>

# 章 1. Device Mapper Multipathing

Device mapper multipathing (DM Multipath) 能讓您配置伺服器節點與儲存陣列之間的 multipath I/O 路徑，將其整合成一個單一裝置。這些 I/O 路徑乃實體的 SAN 連線，包括獨立的網路線、交換器與控制器。multipath 會整合 I/O 路徑，建立包含了已整合路徑的新裝置。

本章節提供了在 Red Hat Enterprise Linux 7 初始發行版發行後加入的 DM-Multipath 功能之概述。本章同時亦提供了高層的 DM-Multipath 與其元件之總覽，以及 DM-Multipath 設定上的總覽。

## 1.1. 新功能與更改的功能

這部分列出了在 Red Hat Enterprise Linux 7 初始發行後加入的新 DM Multipath 功能。

### 1.1.1. Red Hat Enterprise Linux 7.1 的新功能與更改的功能

Red Hat Enterprise Linux 7.1 包含了以下文件、功能上的更新，以及變革。

- ✦ [表格 5.1, “實用的 multipath 指令選項”](#) 現在包含了 `multipath` 指令的 `-w` 與 `-W` 選項項目，並讓您更有效率地管理 `wwids` 檔案。
- ✦ `multipath.conf` 檔案中的 `features` 參數的 `values` 引述記載在 [〈章 4, DM Multipath 配置檔案〉](#) 中。
- ✦ [表格 4.1, “Multipath 配置預設值”](#) 包含了一項 `force_sync` 參數的項目，當設為「yes」時，能避免路徑檢查程式在非同步模式下執行。

此外，亦針對整份文件進行了小幅的技術資訊修正與闡明。

### 1.1.2. Red Hat Enterprise Linux 7.2 的新功能與更改的功能

Red Hat Enterprise Linux 7.2 包含了以下文件、功能上的更新，以及變革。

- ✦ 這份文件包含了一個新的部分：[節 5.1, “透過 Multipath Helper 自動產生配置檔案”](#)。Multipath Helper 應用程式能提供您選項建立 multipath 配置，包含自訂別名、裝置黑名單，以及個別 multipath 裝置的行為設定。
- ✦ `multipath.conf` 配置檔案的 `defaults` 部分支援了新的 `config_dir`、`new_bindings_in_boot`、`ignore_new_boot_devs`、`retrigger_tries`，以及 `retrigger_delays` 參數。`multipath.conf` 檔案的 `defaults` 部分記錄在 [〈表格 4.1, “Multipath 配置預設值”〉](#) 中。
- ✦ `multipath.conf` 配置檔案的 `defaults`、`devices` 與 `multipaths` 部分現在已支援 `delay_watch_checks` 和 `delay_wait_checks` 配置參數。欲取得配置參數上的相關資訊，請參閱 [〈章 4, DM Multipath 配置檔案〉](#)。

此外，亦針對整份文件進行了小幅的技術資訊修正與闡明。

## 1.2. DM Multipath 總覽

DM Multipath 可被使用來提供：

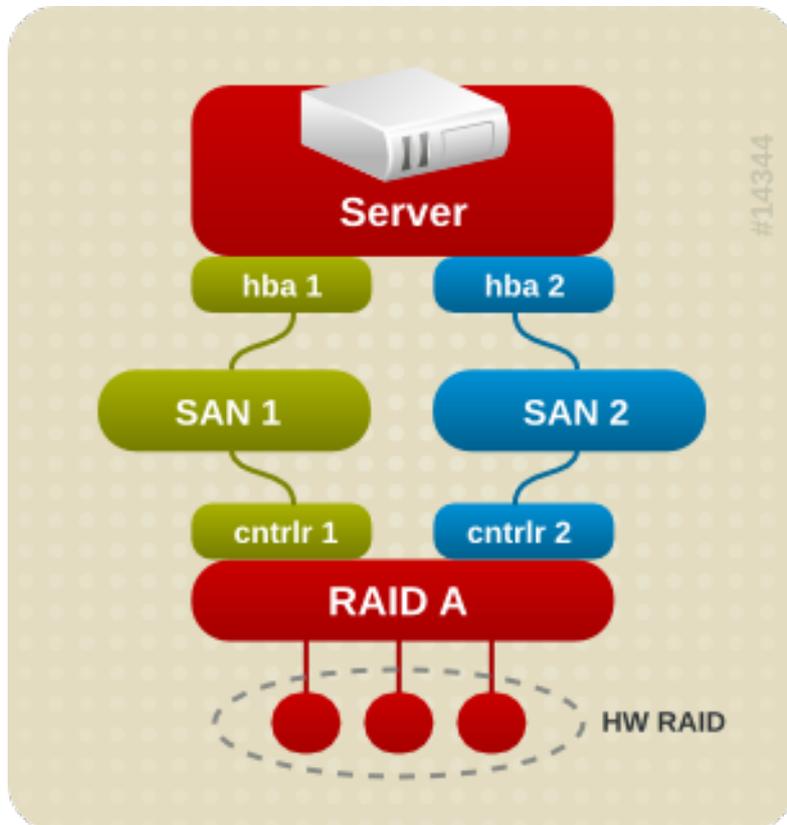
- ✦ 冗餘

DM Multipath 可透過 active/passive (主動/被動) 配置，提供備援功能。在 active/passive 配置中，只有一半的路徑會在任何時間被使用來進行 I/O。如果 I/O 路徑的任何元素 (網路線、交換器、或控制器) 失效，DM Multipath 就會切換到替代的路徑。

» 改善的效能

DM Multipath 可配置為 active/active (主動/主動) 模式，其中 I/O 會以循環排程法 (round-robin) 分散到路徑去。在某些配置中，DM Multipath 可偵測 I/O 路徑的負載，並動態式地平衡此負載。

[圖形 1.1, “以一個 RAID 裝置來進行 Active/Passive Multipath 的配置”](#) 顯示了 active/passive 配置，其中伺服器有兩條 I/O 路徑連至一組 RAID 裝置。伺服器上有兩組 HBA、兩台 SAN 交換器、以及兩組 RAID 控制器。



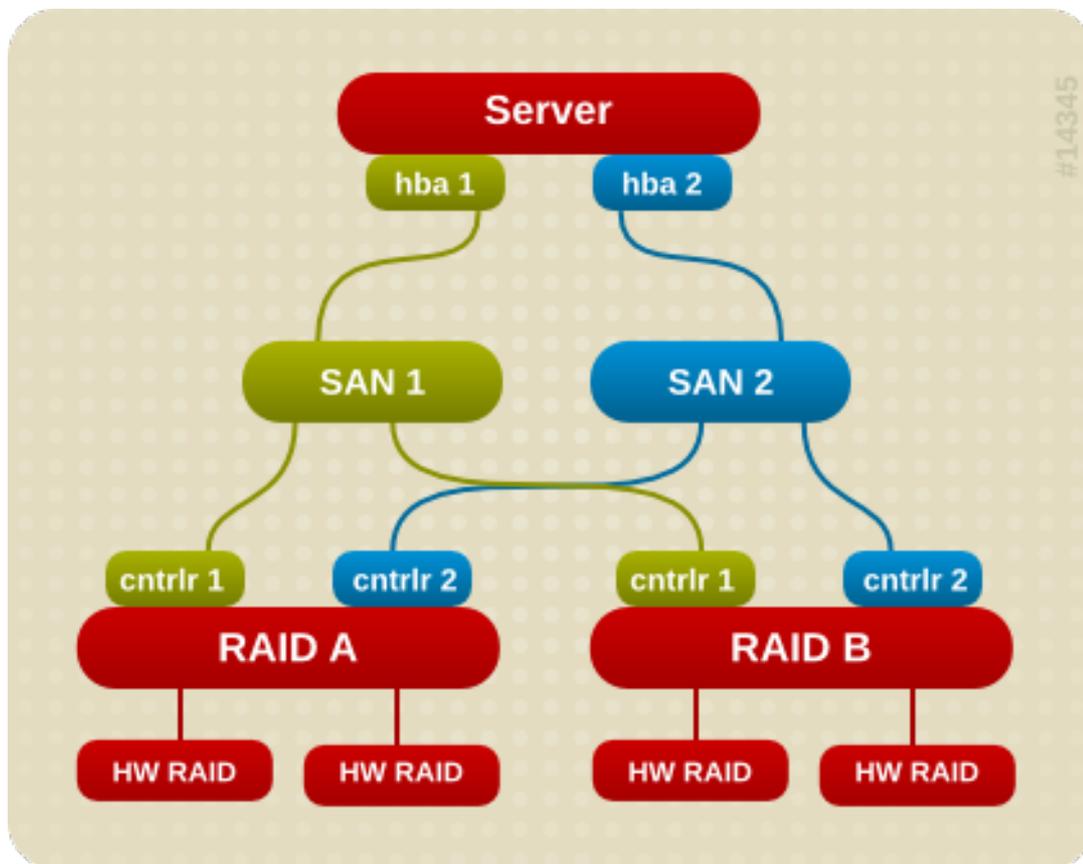
圖形 1.1. 以一個 RAID 裝置來進行 Active/Passive Multipath 的配置

在此配置中，有一組 I/O 路徑透過 hba1、SAN1、以及 controller1 連接，第二組 I/O 路徑則透過 hba2、SAN2、以及 controller2 連接。在此配置中，有許多可能失效的情況會發生：

- » HBA 失效
- » 光纖電纜失效
- » SAN 交換器失效
- » 陣列控制器連接埠失效

配置了 DM Multipath 之後，以上這些失效皆會導致 DM Multipath 切換至其它 I/O 路徑。

[圖形 1.2, “以兩個 RAID 裝置進行 Active/Passive Multipath 的配置”](#) 顯示了較為複雜的 active/passive 配置，伺服器上有兩組 HBA、兩台 SAN 交換器、兩組 RAID 裝置，並且各有兩組 RAID 控制器。



圖形 1.2. 以兩個 RAID 裝置進行 Active/Passive Multipath 的配置

在 [圖形 1.2. “以兩個 RAID 裝置進行 Active/Passive Multipath 的配置”](#) 中所示的範例中，各個 RAID 裝置皆有兩條 I/O 路徑。（類似 [圖形 1.1. “以一個 RAID 裝置來進行 Active/Passive Multipath 的配置”](#) 中的範例所示）。當配置了 DM Multipath 之後，連至 RAID 裝置的 I/O 路徑上的任何失效情況，皆會造成 DM Multipath 切換至該裝置的其它 I/O 路徑。

[圖形 1.3. “以一個 RAID 裝置來進行 Active/Active Multipath 的配置”](#) shows an active/active configuration with 2 HBAs on the server, 1 SAN switch, and 2 RAID controllers. There are four I/O paths from the server to a storage device:

- hba1 到 controller1
- hba1 到 controller2
- hba2 到 controller1
- hba2 到 controller2

在此配置中，I/O 可以分散至這四條路徑。



圖形 1.3. 以一個 RAID 裝置來進行 Active/Active Multipath 的配置

### 1.3. 儲存陣列的支援

就預設值，DM Multipath 會支援大部分本身也支援 DM Multipath 的常見儲存陣列。欲取得更多有關於預設配置值（包括支援的裝置）上的相關資訊，請執行下列指令。

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

若您的儲存陣列支援 DM Multipath 並且就預設值尚未經過配置，您將需要將它加入到 DM Multipath 配置檔案 **multipath.conf** 中。欲取得更多有關於 DM Multipath 配置檔案上的相關資訊，請參閱 [〈章 4. DM Multipath 配置檔案〉](#)。

某些儲存陣列需要透過一些特別的方式，來處理 I/O 錯誤與路徑切換。這需要個別的硬體處理 kernel 模組。

### 1.4. DM Multipath 元件

[表格 1.1, “DM Multipath 元件”](#) 描述了 DM Multipath 的元件。

表格 1.1. DM Multipath 元件

元件	描述
<b>dm_multipath</b> kernel 模組	將 I/O 路由重設，並支援路徑與路徑群組的備援。
<b>mpathconf</b> 工具程式	配置並啟用 device mapper multipathing。
<b>multipath</b> 指令	列出並配置 multipath 裝置。正常來說會以 <b>/etc/rc.sysinit</b> 開始，但每當有區塊裝置加入時，也可以透過 <b>udev</b> 程式來啟動。

元件	描述
<b>multipathd</b> daemon	監控路徑；當路徑失效後又重新啟用，它會切換路徑群組。它也能用於互動式地變更 multipath 裝置。 <code>/etc/multipath.conf</code> 檔案如有任何變更，這個 daemon 便必須重新啟動。
<b>kpartx</b> 指令	要為裝置上的分割區建立 device mapper 裝置，您需要用這項指令來分割 DOS 類的分割區供 DM-multipath 使用。 <b>kpartx</b> 指令是個獨立套件，但是 <b>device-mapper-multipath</b> 套件相依於 <b>kpartx</b> 。

## 1.5. DM Multipath 設定總覽

DM Multipath 包含了已編譯的預設設定，適用於一般的 multipath 配置。為您系統配置 DM Multipath 的基本程序如下：

1. 安裝 **device-mapper-multipath** rpm。
2. 透過 **mpathconf** 指令建立配置檔案並啟用 multipath。若您無須編輯配置檔案，您亦可透過這項指令來啟用 multipath daemon。
3. 視需求編輯 **multipath.conf** 配置檔案，以修改預設值並儲存已更新的檔案。
4. 啟動 multipath daemon。

欲取得 multipath 配置上的詳細設定指示，請參閱 [〈章 3. 設定 DM Multipath〉](#)。

## 章 2. Multipath 裝置

若沒有 DM Multipath，從伺服器節點到儲存控制器的每條路徑皆會被系統視為各別裝置，即使當 I/O 路徑將同樣的伺服器節點連到同樣的儲存控制器亦然。DM Multipath 提供了整理 I/O 路徑的邏輯方法，作法是在其下的裝置上方，建立單一的 multipath 裝置。

### 2.1. Multipath 的裝置識別子

每個 multipath 裝置皆有一個 WWID (World Wide Identifier, 全球識別子)，它是全世界獨一無二、不會改變的。預設上，multipath 裝置名稱會設定為它的 WWID。您可以設定 multipath 設定檔裡面的 `user_friendly_names` 選項，以 `mpathn` 格式設定別名。

舉例來說，有兩個 HBA 的節點透過單一、不分區的光纖交換器連上擁有兩個連接埠的儲存控制器，會看到四個裝置：`/dev/sda`、`/dev/sdb`、`dev/sdc` 以及 `/dev/sdd`。DM Multipath 會使用一組獨一無二的 WWID 建立單一裝置，並依據 multipath 的設定，將這四組裝置的 I/O 重新導向到這單一裝置上。當 `user_friendly_names` 設定設為 `yes` 的時候，multipath 裝置便會設成 `mpathn`。

當新裝置納入 DM Multipath 的控制之下時，新裝置可能會在 `/dev` 目錄下的兩個地方看到：`/dev/mapper/mpathn` 以及 `/dev/dm-n`。

- `/dev/mapper` 裡的裝置是在開機過程中建立的。您可使用這些裝置來存取 multipath 裝置，比方說當建立邏輯卷冊時。
- 所有 `/dev/dm-n` 形式的裝置都僅供內部使用，管理員不該直接這些裝置。

欲取得更多 multipath 預設配置上的相關資訊，包括 `user_friendly_names` 配置選項，請參閱〈[節 4.3, “配置檔案的預設值”](#)〉。

您亦可依您的選擇來設置 multipath 裝置的名稱，方法就是使用 multipath 配置檔案的 `multipaths` 部份中的 `alias` 選項。欲取得更多有關於 multipath 配置檔案中的 `multipaths` 一節的相關資訊，請參閱〈[節 4.4, “Multipath 裝置配置的屬性”](#)〉。

### 2.2. 叢集裡一致的 Multipath 裝置名稱

當 `user_friendly_names` 配置選項設為 `yes` 時，multipath 裝置的名稱對節點來說將會是獨特的；但對使用該 multipath 裝置的所有節點來說，不能保證是唯一的。相同地，若您在 `multipath.conf` 配置檔案的 `multipaths` 部份中為某個裝置設置了 `alias` 選項，該名稱不會自動在叢集中的所有節點上維持其一致性。若您使用了 LVM 來由 multipath 裝置建立邏輯裝置的話，這應該不會造成任何問題。不過，若您希望每個節點中皆使用一致的 multipath 裝置名稱，建議您不要將 `user_friendly_names` 選項設為 `yes`，並且不要為裝置配置別名。就預設值，若您不將 `user_friendly_names` 設為 `yes` 或為裝置配置別名的話，裝置名稱將會是裝置的 WWID，而這將總是相同的。

然而，若您希望叢集中所有節點的系統定義、使用者易記的名稱一致的話，您可依照下列程序進行：

1. 在一部機器上設定所有的 multipath 裝置。
2. 請透過以下指令來停用其它機器上的所有 multipath 裝置：

```
# service multipathd stop
# multipath -F
```

3. 將 `/etc/multipath/bindings` 檔案由第一部機器複製至叢集中所有其它機器上。
4. 請透過下列指令來將叢集中，其它機器上的所有 `multipathd` daemon 重新啟用：

```
# service multipathd start
```

若您新增了裝置，您將需要重複這項程序。

相同地，若您為了您希望使其在叢集中所有節點上皆能保持一致的裝置配置了別名，您應依照相同程序，確認叢集中各個節點的 `/etc/multipath.conf` 檔案皆相同：

1. 為一部機器上的 `multipath.conf` 檔案中的 `multipath` 裝置配置別名。
2. 請透過以下指令來停用其它機器上的所有 `multipath` 裝置：

```
# service multipathd stop
# multipath -F
```

3. 由第一部機器將 `/etc/multipath.conf` 檔案複製至叢集中所有其它機器上。
4. 請透過下列指令來將叢集中，其它機器上的所有 `multipathd` daemon 重新啟用：

```
# service multipathd start
```

當您新增裝置時，您將需要重複這項程序。

## 2.3. Multipath 裝置屬性

除了 `user_friendly_names` 與 `alias` 選項外，`multipath` 裝置有多種屬性。您可以修改特定 `multipath` 裝置的這些屬性，方法是在 `multipath` 配置檔的 `multipaths` 一節裡建立該裝置的條目。預知 `multipath` 配置檔案的 `multipaths` 一節之相關資訊，請參閱 [〈節 4.4. “Multipath 裝置配置的屬性”〉](#)。

## 2.4. 邏輯卷冊中的 multipath 裝置

建立 `multipath` 裝置後，您可以使用 `multipath` 裝置名稱，如建立 LVM 實體卷冊時使用的實體裝置名稱一樣。舉例來說，如果 `/dev/mapper/mpatha` 是 `multipath` 裝置的名稱，以下指令會將 `/dev/mapper/mpatha` 標示為實體卷冊。

```
pvcreate /dev/mapper/mpatha
```

當您建立 LVM 卷冊群組時，您可以使用產生出的 LVM 實體裝置，一如您使用其他 LVM 實體裝置一樣。

### 注意

若您嘗試在您已配置分割區的整個裝置上，建立 LVM 實體卷冊的話，`pvcreate` 指令將會失敗。請注意，若您不為各個區塊裝置指定分割表的話，Anaconda 和 Kickstart 安裝程式將會建立空的分割表。若您希望使用整個裝置而不是一個分割區，您必須由裝置上移除現有的分割區。您可透過 `kpartx -d` 和 `fdisk` 指令來移除既有的分割區。若您的系統擁有超過 2TB 的區塊裝置，您可使用 `parted` 指令來移除分割區。

當您建立了使用 `active/passive` (主動 / 被動) 的 `multipath` 陣列作為實體裝置的 LVM 邏輯卷冊時，您應該在 `/etc/lvm/lvm.conf` 檔案裡加入過濾程式，將這些磁碟排除在外。這是因為若裝置未經過濾，陣列在收到 I/O 時自動由主動路徑改變為被動路徑的時候，`multipath` 會在 LVM 掃描被動路徑時進行障礙復原和容錯移轉。對於需要指令來啟動被動路徑的 `active/passive` (主動 / 被動) 陣列，LVM 會在此情況發生時顯示警告訊息。

若要篩選 LVM 配置檔案 (`lvm.conf`) 中的所有 SCSI 裝置，請將以下篩選程式附加至該檔案的 **devices** 一節中。

```
filter = [ "r/block/", "r/disk/", "r/sd.*/", "a./." ]
```

## 章 3. 設定 DM Multipath

本章的範例會逐步帶領您配置 DM Multipath。其中包含了以下程序：

- 基本 DM Multipath 設定
- 忽略本機磁碟
- 新增更多裝置至配置檔案中
- 在 **initramfs** 檔案系統中啟用 multipath

### 3.1. 設定 DM Multipath

要在系統上設定 DM Multipath 之前，請確定您的系統已安裝 **device-mapper-multipath** 套件，或已更新至最新版。

您透過 **mpathconf** 工具程式來設定 multipath，這將會建立 multipath 配置檔案 **/etc/multipath.conf**。

- 若 **/etc/multipath.conf** 檔案已存在，**mpathconf** 工具程式便會編輯該檔案。
- 若 **/etc/multipath.conf** 檔案不存在的話，**mpathconf** 工具程式將會使用 **/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf** 檔案來作為起始檔案。
- 若 **/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf** 檔案不存在的話，**mpathconf** 工具程式將會重新建立一個 **/etc/multipath.conf** 檔案。

若您不需要編輯 **/etc/multipath.conf** 檔案，您可透過執行下列指令來為基本備援配置設定 DM Multipath。這項指令將啟用 multipath 配置檔案並啟動 **multipathd** daemon。

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

若您在啟動 **multipathd** daemon 之前，需要編輯 **/etc/multipath.conf** 檔案的話，請使用下列程序來為基本備援配置設定 DM Multipath。

1. 請執行 **mpathconf** 指令，並指定 **--enable** 選項：

```
# mpathconf --enable
```

欲取得更多有關於 **mpathconf** 指令選項上的相關資訊，您可能需要參閱 **mpathconf** 的 man page 或是執行 **mpathconf** 指令並指定 **--help** 選項。

```
# 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. 請視需求編輯 **/etc/multipath.conf** 檔案。DM Multipath 的預設設定會被編譯入系統中，並且無須明確地設置於 **/etc/multipath.conf** 檔案中。

`path_grouping_policy` 的預設值設為 `failover`，因此在本範例中，您無須編輯 `/etc/multipath.conf` 檔案。欲取得如何將配置檔案中的預設值改為其它值的相關資訊，請參閱〈[章 4, DM Multipath 配置檔案](#)〉。

配置檔案的初始預設值一節會將 multipath 裝置名稱設為 `mpathn` 這個格式；若沒有這項設定，multipath 裝置的名稱將會是裝置 WWID 的別名。

- 視需求儲存配置檔案並退出文字編輯器。
- 執行下列指令：

```
# service multipathd start
```

由於配置檔案中，`user_friendly_name` 的值被設為了 `yes`，因此 multipath 裝置會以 `/dev/mapper/mpathn` 格式被建立。欲取得如何為裝置名稱設置別名的相關資訊，請參閱〈[章 4, DM Multipath 配置檔案](#)〉。

若您不希望採用使用者容易記得的名稱，您可執行下列指令：

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



### 注意

如果您需要在 multipath daemon 啟動後編輯 multipath 配置檔，那麼在編輯後請務必執行 `service multipathd reload` 指令，好讓變更生效。

## 3.2. 產生 multipath 裝置時，忽略本機磁碟

有些電腦上的內部磁碟含有本機 SCSI 卡。不建議為這些裝置使用 DM Multipath。若您將 `find_multipaths` 配置參數設為 `yes`，您便無須將這些裝置納入黑名單中。欲取得更多有關於 `find_multipaths` 配置參數上的相關資訊，請參閱〈[節 4.3, “配置檔案的預設值”](#)〉。

若您不將 `find_multipaths` 配置參數設為 `yes`，您可使用下列程序來修改 multipath 配置檔案，以在配置 multipath 時忽略本機磁碟。

- 決定哪些磁碟為本機磁碟，並將這些磁碟標示於黑名單中。

在此範例中，`/dev/sda` 是內部磁碟。請注意預設的 multipath 配置檔案中原有的值為何，執行 `multipath -v2` 指令會在 multipath 映射中顯示本機硬碟 `/dev/sda`。

欲取得更多有關於 `multipath` 指令輸出的相關資訊，請參閱〈[節 5.8, “Multipath 指令的輸出”](#)〉。

```
# multipath -v2
create: 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 [------

device-mapper ioctl cmd 9 failed: Invalid argument
device-mapper ioctl cmd 14 failed: No such device or address
create: 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

create: 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

create: 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

create: 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

```

- 為了要避免裝置映射程式映射 `/dev/sda`，請編輯 `/etc/multipath.conf` 檔案裡的黑名單 (blacklist) 一節，以包含此裝置。雖然您可透過使用 `devnode` 類型來將 `sda` 裝置納入黑名單中，但這並不安全，因為重新開機之後，`/dev/sda` 並不一定會是同一台磁碟機。若要將各別裝置納入黑名單中，您可使用該裝置的 WWID。

請注意 `multipath -v2` 指令的執行結果。`/dev/sda` 裝置的 WWID 是「SIBM-ESXSST336732LC\_\_\_\_F3ET0EP0Q000072428BX1」。要將此裝置納入黑名單中，請在 `/etc/multipath.conf` 檔案中加入以下幾行。

```

blacklist {
    wwid SIBM-ESXSST336732LC____F3ET0EP0Q000072428BX1
}

```

- 在您升級了 `/etc/multipath.conf` 檔案之後，您必須手動告訴 `multipathd` daemon 重新載入這檔案。以下指令可以重新載入更新過的 `/etc/multipath.conf` 檔案。

```
# service multipathd reload
```

- 請執行下列指令，以移除 multipath 裝置：

```
# multipath -f SIBM-ESXSST336732LC____F3ET0EP0Q000072428BX1
```

- 若要檢查裝置移除是否成功，您可執行 `multipath -ll` 指令以顯示目前的 multipath 配置。若希望取得 `multipath -ll` 指令上的相關資訊，請參閱〈[節 5.9 “使用 multipath 指令來進行 Multipath 查詢”](#)〉。

若要檢查被納入黑名單中的裝置是否沒有被加回，您可執行 `multipath` 指令，如以下範例所示。若您沒有指定 `-v` 選項的話，`multipath` 指令的資訊詳細等級預設值將會是 `v2`。

```

# multipath

create: 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

```

```

create: 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

create: 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

create: 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

```

### 3.3. 配置儲存裝置

就預設值，DM Multipath 會支援大部分本身也支援 DM Multipath 的常見儲存陣列。欲取得更多有關於預設配置值（包括支援的裝置）上的相關資訊，請執行下列指令。

```

# multipathd show config
# multipath -t

```

若您需要新增預設上不受支援的儲存裝置，使其成為 multipath 裝置的一員，請編輯 `/etc/multipath.conf` 檔案，加入適當的裝置資訊。

舉例來說，若要新增有關於 HP Open-V 系列裝置的相關資訊，此項目將會看似以下範例。此範例會在所有路徑失敗時，將裝置設置為排在佇列中一分鐘（或嘗試 12 次，一次等待五秒鐘）。

```

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

```

欲取得更多配置檔案的 `devices` 一節的相關資訊，請參閱 [〈節 4.5, “配置檔案裝置”〉](#)。

### 3.4. 在 initramfs 檔案系統中設定 multipath

您可以在 `initramfs` 檔案系統中設定 multipath。配置 multipath 之後，您可以使用 multipath 配置檔案重建 `initramfs` 檔案系統，方法適用以下選項執行 `dracut` 指令：

```

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

```

如果您是從 `initramfs` 檔案系統上執行 multipath，並變更了 multipath 配置檔案，那您必須重建 `initramfs` 檔案系統，好讓變更生效。

## 章 4. DM Multipath 配置檔案

就預設值，DM Multipath 提供的配置值足以應付 multipath 大部分的一般使用方式。除此之外，DM Multipath 支援常見、本身支援 DM Multipath 功能的儲存裝置陣列。欲取得有關於預設配置值上的相關資訊（包括支援的裝置）請執行以下其中一項指令。

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

您可以編輯 `/etc/multipath.conf` 配置檔案，覆寫 DM Multipath 的預設配置值。若有必要，您也能將預設上不受支援的儲存陣列加入配置檔案中。

### 注意

您可以在 **initramfs** 檔案系統上執行設定 multipath。如果您可以從 **initramfs** 檔案系統上執行 multipath，並更改了任何 multipath 配置檔，那就必須重建 **initramfs** 檔案系統，以讓變更生效。欲知重建使用 multipath 的 **initramfs** 檔案系統的方法，請參閱〈[節 3.4, “在 initramfs 檔案系統中設定 multipath”](#)〉。

本章提供了關於剖析與修改 `multipath.conf` 檔案的資訊，其中包含了以下章節：

- 配置檔案總覽
- 配置檔案黑名單
- 配置檔案預設值
- 配置檔案 multipath
- 配置檔案裝置

在 multipath 配置檔案中，您僅需要指定您配置所需的部份，或需改變的預設值。如果配置檔案中有些部份與您的使用環境無關，或者您不想要蓋過預設值，您可將這些部份設為註釋，就如在初始檔案中一樣。

您可使用正規表示式（regular expression）語法來撰寫配置檔案。

您可在 `multipath.conf(5)` man page 中找到更多有關於配置檔案上的相關資訊。

### 4.1. 配置檔案總覽

multipath 的配置檔案可分成以下幾節：

#### **blacklist**

不會被用來作為 multipath 的特定裝置清單。

#### **blacklist\_exceptions**

根據 blacklist 一節的參數，列出 multipath 黑名單以外的裝置。

#### **defaults**

DM Multipath 的一般預設設定。

#### **multipaths**

個別 multipath 裝置特性的設定。這些值會覆蓋過配置檔案的 **defaults** 與 **devices** 兩節中所指定的設定。

### devices

個別儲存控制卡的設定。這些值會覆蓋過配置檔案的 **defaults** 一節中的設定。如果預設設定並不支援您使用的儲存陣列，您可能需要為其建立 **devices** 一節。

當系統決定了一組 multipath 裝置的屬性，它會先檢查 multipath 的設定、裝置的設定，然後則是 multipath 的系統預設值。

## 4.2. 配置檔案的黑名單

multipath 配置檔案的 **blacklist** (黑名單) 一節詳述了當系統配置 multipath 裝置時，不會被使用的裝置。在此清單中的裝置將不會被納入 multipath 裝置的群組中。

在較舊的 RHEL 發行版中，multipath 總是會試圖為每個未明確納入黑名單中的路徑，建立 multipath 裝置。然而從 RHEL 6 起，若 **find\_multipaths** 配置參數設為了 **yes**，multipath 僅會在滿足了下列三項條件之一的情況下建立裝置：

- 至少兩個擁有相同 WWID、未納入黑名單的路徑。
- 使用者透過 **multipath** 指令來指定裝置，以手動式強制建立裝置。
- 一個含有與先前建立之 multipath 裝置相同 WWID 的路徑（儘管該 multipath 裝置目前已不存在）。每當某個 multipath 裝置被建立時，multipath 會記得裝置的 WWID，如此一來當它看見含有該 WWID 的路徑時，它便會自動地再次建立裝置。這能讓 multipath 自動選擇正確的路徑，在無須編輯 multipath 黑名單的情況下，製成 multipath 裝置。

若您先前在未使用 **find\_multipaths** 參數的情況下建立 multipath 裝置，而之後將參數設為 **yes** 的話，您可能需要由 **/etc/multipath/wwids** 檔案中，移除任何您不希望將其建立為 multipath 裝置的裝置 WWID。以下顯示了範例 **/etc/multipath/wwids** 檔案。WWID 包含在斜線符號之間 (/)：

```
# 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/
```

若 **find\_multipaths** 參數設為 **yes**，您只需要將含有多重路徑，而您不希望 multipath 這些路徑的裝置納入黑名單中。因為如此，一般來講您無須將裝置納入黑名單中。

若您需要將裝置納入黑名單中，您可根據下列條件進行：

- 根據 WWID，如〈[節 4.2.1, “根據 WWID 建立黑名單”](#)〉中所描述
- 根據裝置名稱，如〈[節 4.2.2, “根據裝置名稱來建立黑名單”](#)〉中所描述
- 根據裝置類型，如〈[節 4.2.3, “根據裝置類型來建立黑名單”](#)〉中所描述

就預設值，即使您將配置檔案中的黑名單一節標記為註解，各種裝置類型依然還是會位於黑名單中。欲取得更詳盡的資訊，請參閱〈[節 4.2.2, “根據裝置名稱來建立黑名單”](#)〉。

### 4.2.1. 根據 WWID 建立黑名單

您可以根據裝置的 WWID (World-Wide IDentification) 將裝置放入黑名單中，方法是在配置檔案的 **blacklist** (黑名單) 一節裡放入 **wwid** 這個項目。

以下範例顯示配置檔案如何將 WWID 為 26353900f02796769 的裝置加入黑名單中。

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

### 4.2.2. 根據裝置名稱來建立黑名單

您可根據裝置名稱將裝置類型納入黑名單中，如此一來這些裝置便不會被併入 multipath 裝置的群組中，方法是在配置檔案的 **blacklist** 一節中，指定 **devnode** 項目。

下列範例顯示了在配置檔案中，所有 SCSI 裝置皆納入了黑名單中，因為它將所有 sd\* 裝置納入了黑名單中。

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

您可以在配置檔案的 **blacklist** 一節中，使用 **devnode** 項目來指定各別的裝置以納入黑名單中，而非納入屬於特定類型的所有裝置。然而我們並不建議這麼作。除非某個裝置是以 **udev** 規則靜態對映，否則系統無法保證重新開機後，該裝置能保有原來的名稱。舉例來說，重新開機後，某個裝置可能會從 **/dev/sda** 變為 **/dev/sdb**。

就預設值，以下 **devnode** 項目會被編譯入預設的黑名單中；在黑名單中的裝置一般不支援 DM Multipath。若要啟用這些裝置的 multipath 功能，您需要將它們指定於配置檔案的 **blacklist\_exceptions** 部份中，如〈[節 4.2.4. “黑名單的例外”](#)〉中所述。

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

### 4.2.3. 根據裝置類型來建立黑名單

您可在配置檔案中的 **blacklist** 的 **device** 部分中，指定特定裝置類型。下列範例將所有 IBM DS4200 和 HP 裝置納入了黑名單。

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

### 4.2.4. 黑名單的例外

您可透過使用配置檔案的 **blacklist\_exceptions** 部份，將那些就預設值被納入黑名單中的裝置的 multipath 功能啟用。

比方說，若您擁有大量裝置並且僅希望 multipath 其中之一（其 WWID 為 3600d0230000000000e13955cc3757803），與其各別將除了您所希望 multipath 的裝置之外的所有裝置納入黑名單中，您可將所有裝置納入黑名單中，並透過在 `/etc/multipath.conf` 檔案中附加下列行列，以允許您所希望 multipath 的裝置。

```
blacklist {
    wwid ""
}

blacklist_exceptions {
    wwid "3600d0230000000000e13955cc3757803"
}
```

在指定配置檔案中 **blacklist\_exceptions** 一節裡的裝置時，您必須用處理黑名單的同樣方式，將其指定至此。舉例來說，WWID 的例外並不會套用至 **devnode** 黑名單項目中的裝置上，即使這黑名單上的裝置與這組 WWID 有所關聯亦然。同樣，**devnode** 的例外也僅套用於 **devnode** 項目上，而 **device** 的例外也僅會套用於裝置項目上。

### 4.3. 配置檔案的預設值

`/etc/multipath.conf` 配置檔案包括 **defaults**（預設值）一節，可以將 **user\_friendly\_names** 設為 **yes**，語法如下。

```
defaults {
    user_friendly_names yes
}
```

這會覆蓋 **user\_friendly\_names** 參數的預設值。

這配置檔案包括了所有預設設定的範本。這一部份如下所示，已被設為註解。

```
#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
#}
```

若要覆蓋任何參數的預設值，您可將此範本裡的相關資訊複製至 **defaults** 一節裡，並拿掉註解符號。舉例來說，要覆蓋 **path\_grouping\_policy** 的原有參數 **multibus**，使其成為 **failover**，請將範本中相對應的部份複製至配置檔案的 **defaults** 一節裡，拿掉註解符號，如同以下顯示：

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

[表格 4.1. “Multipath 配置預設值”](#) 描述了 `multipath.conf` 配置檔案中，`defaults` 一節裡所設定的屬性。DM Multipath 會使用這些值，除非它們被 `multipath.conf` 檔案中的 `devices` 與 `multipaths` 兩節裡所指定的屬性覆蓋過去。

表格 4.1. Multipath 配置預設值

屬性	描述
<code>polling_interval</code>	以秒為單位指定兩條路徑之間的檢查時間間隔。對於正確運作的路徑來說，檢查程序之間的時間間隔會逐漸增加（ $4 * \text{polling\_interval}$ ）。預設值為 5。
<code>multipath_dir</code>	存放動態共享物件的目錄。預設值根據系統而定，一般為 <code>/lib/multipath</code> 。
<code>find_multipaths</code>	<p>定義用來設定 multipath 裝置的模式。若此參數設為 <b>yes</b> 的話，multipath 將不會試圖為各個非黑名單中的路徑建立裝置。multipath 只會在滿足了下列三種條件之一的情況下，建立裝置：</p> <ul style="list-style-type: none"> <li>- 至少兩個擁有相同 WWID、非黑名單中的路徑。</li> <li>- 使用者透過 <code>multipath</code> 指令指定裝置，以手動式強制建立裝置。</li> <li>- 一條擁有與先前建立之 multipath 裝置相同 WWID 的路徑。每當有個 multipath 裝置在設置了 <code>find_multipaths</code> 的情況下被建立時，multipath 便會記得該裝置的 WWID，如此一來每當它看見含有該 WWID 的路徑時，它便會自動地再次建立該裝置。這能讓 multipath 自動地選擇正確的路徑，在無須編輯 multipath 黑名單的情況下，製成 multipath 裝置。若您先前在未設置 <code>find_multipaths</code> 參數的情況下建立了 multipath 裝置，欲取得相關程序指示，請參閱〈<a href="#">節 4.2. “配置檔案的黑名單”</a>〉。</li> </ul> <p>預設值為 <b>no</b>。然而，從 Red Hat Enterprise Linux 7 起，<code>mpathconf</code> 所建立的預設 <code>multipath.conf</code> 檔案將會啟用 <code>find_multipaths</code>。</p>
<code>verbosity</code>	預設資料詳細等級。愈大的值便會增加資料詳細度的等級。有效的等級範圍為 0 到 6 之間。預設值為 2。
<code>path_selector</code>	<p>指定用來決定下一項 I/O 作業時，該使用何路徑的預設演算法則。有效的值包含了：</p> <p><b>round-robin 0</b>：在路徑群組中的所有路徑之間進行循環，並將相同的 I/O 數量傳送至各個路徑。</p> <p><b>queue-length 0</b>：將下個 I/O 量傳送至含有最少未處理之 I/O 請求數量的路徑。</p> <p><b>service-time 0</b>：將下個 I/O 量傳送至含有最短預估服務時間的路徑，這是透過將未處理的 I/O 總數，除以各個路徑的傳輸量來取得的。</p> <p>預設值為 <b>round-robin 0</b>。</p>

屬性	描述
<b>path_grouping_policy</b>	<p>指定預設路徑群組政策，以套用至未指定的 multipath。可用的值包含了：</p> <p><b>failover</b>：每個優先群組 1 條路徑。</p> <p><b>multibus</b>：1 個優先群組中的所有有效路徑。</p> <p><b>group_by_serial</b>：每個偵測到的序號 1 個優先群組。</p> <p><b>group_by_prio</b>：每個路徑優先值 1 個優先群組。優先權取決於被指定為全域、各別控制器，或各別 multipath 選項的 callout 程式。</p> <p><b>group_by_node_name</b>：一個目標節點名稱 1 個優先群組。目標節點名稱可由 <code>/sys/class/fc_transport/target*/node_name</code> 中取得。</p> <p>預設值為 <b>failover</b>。</p>
<b>prio</b>	<p>指定欲調用的預設函式，以取得路徑優先值。舉例來說，SPC-3 中的 ALUA 位元提供了一組可供利用的 <b>prio</b> 值。可使用的值包含了：</p> <p><b>const</b>：將所有路徑的優先值設為 1。</p> <p><b>emc</b>：為 EMC 陣列產生路徑優先權。</p> <p><b>alua</b>：根據 SCSI-3 ALUA 設定產生路徑優先權。</p> <p><b>ontap</b>：為 NetApp 陣列產生路徑優先權。</p> <p><b>rdac</b>：為 LSI/Engenio RDAC 控制器產生路徑優先權。</p> <p><b>hp_sw</b>：為 active/standby 模式的 Compaq/HP 控制器產生路徑優先權。</p> <p><b>hds</b>：為 Hitachi HDS Modular 儲存裝置陣列產生路徑優先權。</p> <p>預設值為 <b>const</b>。</p>

屬性	描述
features	<p>multipath 裝置的預設額外特性，格式為：<i>"number_of_features_plus_arguments feature1 ..."</i>。</p> <p><b>features</b> 可能的值為：</p> <p>唯一存在的功能為 <b>queue_if_no_path</b>，這與將 <b>no_path_retry</b> 設為 <b>queue</b> 相同。欲查看使用此功能時所可能發生的問題，請參閱〈<a href="#">節 5.7.4 “queue_if_no_path 功能上的問題”</a>〉。</p> <p><b>retain_attached_hw_handler</b>：如果此參數設為 <b>yes</b>，並且 SCSI 層已經設為路徑裝置的硬體處理程式，那麼 multipath 不會強迫裝置使用 <b>multipath.conf</b> 檔案所指定的 <b>hardware_handler</b>。如果 SCSI 層並未連結硬體處理程式，multipath 會繼續如常使用其配置的硬體處理程式。預設值為 <b>no</b>。</p> <p><b>pg_init_retries n</b>：重試路徑群組的初始化過程，最多 <i>n</i> 次，其中 <math>1 \leq n \leq 50</math>。</p> <p><b>pg_init_delay_msecs n</b>：路徑群組初始化重試之間，等待 <i>n</i> 毫秒，其中 <math>0 \leq n \leq 60000</math>。</p>
path_checker	<p>指定用來判斷路徑狀態的預設方式。可用的值包含了：</p> <p><b>readsector0</b>：讀取裝置的第一個磁區。</p> <p><b>tur</b>：發送一項 <b>TEST UNIT READY</b> 指令至裝置。</p> <p><b>emc_clariion</b>：查詢 EMC Clariion 專屬的 EVPD 分頁 0xC0 以判斷路徑。</p> <p><b>hp_sw</b>：檢查含有 Active/Standby 軟體之 HP 儲存裝置陣列的路徑狀態。</p> <p><b>rdac</b>：檢查 LSI/Engenio RDAC 儲存控制器的路徑狀態。</p> <p><b>directio</b>：讀取第一個含有直接 I/O 的磁區。</p> <p>預設值為 <b>directio</b>。</p>
failback	<p>管理路徑群組的容錯回復。</p> <p><b>immediate</b> 值可指定即刻容錯回復至包含了已啟用路徑的最高優先權之路徑群組。</p> <p><b>manual</b> 這個值指定容錯回復不會即刻發生，只有在運算子干預時，容錯回復才會發生。</p> <p><b>followover</b> 值指定當某個路徑群組中的第一條路徑啟用時，自動容錯回復便會進行。這會避免當有個節點請求容錯回復時，另一個節點自動進行容錯回復。</p> <p>超過零的數值表示遞延的容錯回復，單位為秒數。</p> <p>預設值為 <b>manual</b>。</p>

屬性	描述
<b>rr_min_io</b>	指定欲路由轉送至某路徑的 I/O 請求數量（在切換至目前路徑群組中的下個路徑之前）。這項設定僅適用於執行比 2.6.31 還要舊的 kernel 的系統上。較新的系統應使用 <b>rr_min_io_rq</b> 。預設值為 1000。
<b>rr_min_io_rq</b>	使用基於請求的 device-mapper-multipath 來指定欲路由轉送至某路徑的 I/O 請求數量（在切換至目前路徑群組中的下個路徑之前）。這項設定應使用於執行最新 kernel 的系統上。在執行比 2.6.31 還要舊的 kernel 的系統上，請使用 <b>rr_min_io</b> ，預設值為 1。
<b>rr_weight</b>	如果設為 <b>priorities</b> ，那麼與其在調用 <b>path_selector</b> 以選擇下個路徑之前發送 <b>rr_min_io</b> 請求至某個路徑，這項請求數量會相當於 <b>rr_min_io</b> 乘以路徑的優先順序；這乃透過 <b>prio</b> 功能所決定。若設為 <b>uniform</b> ，所有的路徑皆會擁有相等的權重。預設值為 <b>uniform</b> 。
<b>no_path_retry</b>	<p>這個屬性是用數字表示，它指定了系統在停用佇列之前，應試圖使用失效路徑的次數。</p> <p><b>fail</b> 表示立即失效，不進入佇列。</p> <p><b>queue</b> 值代表排程不應停止，直到路徑修正為止。</p> <p>預設值為 0。</p>
<b>user_friendly_names</b>	如果設為 <b>yes</b> ，這表示系統應該使用 <b>/etc/multipath/bindings</b> 檔案來指定一個永續性且獨一無二的別名給 multipath 使用，格式為 <b>mpathn</b> 。若設為 <b>no</b> ，則表示系統應使用 WWID 來作為 multipath 的別名。無論是何種情況，在此所進行的設定，會被您在配置檔案的 <b>multipaths</b> 部份中，所指定的任何裝置特屬別名取代。預設值為 <b>no</b> 。
<b>queue_without_daemon</b>	若設為了 <b>no</b> ，當 <b>multipathd</b> daemon 關閉時，它將會停用所有裝置的排程。預設值為 <b>no</b> 。
<b>flush_on_last_del</b>	若設為 <b>yes</b> ，當連上裝置的最後一條路徑被刪除時， <b>multipathd</b> daemon 便會停用佇列功能。預設值為 <b>no</b> 。
<b>max_fds</b>	設置 multipath 以及 <b>multipathd</b> daemon 所能開啟的最大數量 open file descriptor。這與 <b>ulimit -n</b> 指令相等。由 RHEL 6.3 發行版起，預設值乃 <b>max</b> ，並透過 <b>/proc/sys/fs/nr_open</b> 來將系統設為最大值。在較早的發行版中，若未進行這項設置，open file descriptor 的最大值將會由調用程序取得；一般乃 1024。若該數量大於 1024，為安全起見，應設為路徑的最大值加 32。
<b>checker_timeout</b>	使優先權設定程式與路徑檢查程式發出 SCSI 指令和明確逾時（秒）的逾時功能。預設值乃透過 <b>sys/block/sdx/device/timeout</b> 取得。
<b>fast_io_fail_tmo</b>	當在一個 FC 遠端連接埠上偵測到錯誤時，SCSI 層在傳輸失效 I/O 給該遠端連接埠上的裝置之前，所等待的秒數。此數值應比 <b>dev_loss_tmo</b> 的值還要小。若將此設為 <b>off</b> 將會停用逾時功能。預設值是由 OS 所決定的。
<b>dev_loss_tmo</b>	當在一個 FC 遠端連接埠上偵測到錯誤時，SCSI 層將該連接埠由系統移除之前，所等待的秒數。將此設為無限制，會將其設為 2147483647 秒，或 68 年。預設值乃透過 OS 所決定的。
<b>hwtable_string_match</b>	<p><b>multipath.conf</b> 檔案的 <b>devices</b> 部分中的各項裝置配置皆會建立自己的裝置配置，或是修改內建的裝置配置。若將 <b>hwtable_string_match</b> 設為 <b>yes</b>，而若使用者裝置配置中的供應商、產品以及修訂字串，皆與內建裝置配置的字串完全相符的話，內建配置將由使用者配置中的選項進行修改。否則，使用者的裝置配置將會被視為新的配置。若 <b>hwtable_string_match</b> 設為 <b>no</b>，則會使用一項正規表示式比對來替代字串比對。</p> <p><b>hwtable_string_match</b> 就預設值會設為 <b>no</b>。</p>

屬性	描述
<code>retain_attached_hw_handler</code>	如果此參數設為 <b>yes</b> ，並且 SCSI 層已經設為路徑裝置的硬體處理程式，那麼 <code>multipath</code> 不會強迫裝置使用 <code>multipath.conf</code> 檔案所指定的 <code>hardware_handler</code> 。如果 SCSI 層並未連結硬體處理程式， <code>multipath</code> 會繼續如常使用其配置的硬體處理程式。預設值為 <b>no</b> 。
<code>detect_prio</code>	若設為 <b>yes</b> ， <code>multipath</code> 會先檢查裝置是否支援 ALUA，如果支援的話， <code>multipath</code> 會自動指定裝置為 <b>alua</b> 優先程式。如果裝置不支援 ALUA， <code>multipath</code> 會如常決定優先程式。預設值為 <b>no</b> 。
<code>reload_readwrite</code>	若設為 <b>yes</b> ， <code>multipathd</code> daemon 會監聽路徑裝置的變更事件，如果裝置變成可讀寫，就會重新載入 <code>multipath</code> 裝置。
<code>uid_attribute</code>	提供一項獨特的路徑識別子。預設值為 <b>ID_SERIAL</b> 。
<code>force_sync</code>	(適用 RHEL 7.1 或以上) 如果設為 <b>yes</b> ，這會避免路徑檢查程式在非同步模式下執行。
<code>delay_watch_checks</code>	(Red Hat Enterprise Linux Release 7.2 與更新版本) 若設為了大於零的值， <code>multipathd</code> daemon 將會以指定的檢查次數來監視近期成為有效的路徑。若這些路徑在被監視時再度失效，當它們下次成為有效時，在通過連續的檢測 (透過 <code>delay_wait_checks</code> 指定) 前，都不會再被使用。這能避免信賴度低的路徑在重新啟用後被即刻使用。預設值為 <b>no</b> 。
<code>delay_wait_checks</code>	(Red Hat Enterprise Linux 發行版 7.2 與更新版本) 若設為一個大於 0 的值，當一個近期重新啟用的裝置在指定次數的檢測 (透過 <code>delay_watch_checks</code> 指定) 進行中再次失效時，下次它再重新啟用時便會被標記和延緩，並且在它通過了 <code>delay_wait_checks</code> 中所指定得檢測次數之前，它將不會被使用。預設值為 <b>no</b> 。
<code>ignore_new_boot_devs</code>	(Red Hat Enterprise Linux 發行版 7.2 與更新版本) 若設為了 <b>yes</b> ，當節點早期開機時依然在 <code>initramfs</code> 檔案系統中的時候， <code>multipath</code> 不會建立任何 WWID 不存在 <code>/etc/multipath/wwids</code> 的 <code>initramfs copy</code> 中的裝置。這項功能可用來在安裝時進行啟動，否則 <code>multipath</code> 將會試圖在當時透過 <code>udev</code> 規則顯示、但未宣告的裝置上自行進行設置。這項參數可設為 <b>yes</b> 或 <b>no</b> 。若未設置的話，預設值將會是 <b>no</b> 。
<code>retrigger_tries</code> 、 <code>retrigger_delay</code>	(Red Hat Enterprise Linux 發行版 7.2 與更新版本) 若 <code>udev</code> 無法完整處理原始的 <code>uevent</code> 而使得 <code>multipath</code> 無法使用裝置， <code>retrigger_tries</code> 和 <code>retrigger_delay</code> 參數會被結合使用來令 <code>multipathd</code> 重新啟動這些 <code>uevent</code> 。若裝置未完整設定， <code>retrigger_tries</code> 參數可被用來設置 <code>multipath</code> 嘗試重新啟動一項 <code>uevent</code> 的次數。 <code>retrigger_delay</code> 參數則用來設置重新嘗試之間的時間 (秒)。這兩項選項皆接受大於或等於零的數字。將 <code>retrigger_tries</code> 參數設為零會停用重新嘗試。將 <code>retrigger_delay</code> 參數設為零會使 <code>uevent</code> 在下個路徑檢查的循環中重新發送。若未設置 <code>retrigger_tries</code> 參數，它將會被預設為 3。若未設置 <code>retrigger_delay</code> 參數，它則會被預設為 10。
<code>new_bindings_in_boot</code>	(Red Hat Enterprise Linux 發行版 7.2 與更新版本) <code>new_bindings_in_boot</code> 這個參數主要用來防止 <code>multipath</code> 在 <code>initramfs</code> 檔案系統中提供一組 <code>user_friendly_name</code> (早已由正規檔案系統中的綁定檔案所提供)，這會產生問題因為 <code>initramfs</code> 檔案系統中的 <code>user_friendly_names</code> 綁定僅會在 <code>initramfs</code> 檔案系統重新建立時，才會與正規檔案系統中的綁定同步。當此參數設為 <b>no</b> 時， <code>multipath</code> 將不會在 <code>initramfs</code> 檔案系統中建立新的綁定。若一項裝置在 <code>/etc/multipath/bindings</code> 的 <code>initramfs</code> 副本中不存在綁定， <code>multipath</code> 將會使用其 WWID 作為別名，而非提供一組 <code>user_friendly_name</code> 。在啟動程序後續進行時，當節點掛載了正規檔案系統後， <code>multipath</code> 便會提供一組 <code>user_friendly_name</code> 給裝置。此參數能設為 <b>yes</b> 或 <b>no</b> 。若未設置的話，預設值將會是 <b>no</b> 。

屬性	描述
<code>deferred_remove</code>	若設為 <b>yes</b> ，當最後的路徑裝置被移除後， <code>multipathd</code> 將會進行一項延期的移除，而非正常的移除程序。這可確保當進行正常移除時，若有 <code>multipath</code> 裝置正被使用中而移除程序失敗的情況下，裝置將會在最後一個使用者關閉裝置時被自動移除。預設值為 <b>no</b> 。
<code>log_checker_err</code>	若設為 <b>once</b> ， <code>multipathd</code> 便會以等級 2 的詳細程度紀錄第一項路徑檢查程式錯誤。之後所有的錯誤皆會以等級 3 的詳細程度紀錄，直到裝置復原。若設為 <b>always</b> ， <code>multipathd</code> 總是會以等級 2 的詳細程度紀錄路徑檢查程式錯誤。預設值為 <b>always</b> 。

## 4.4. Multipath 裝置配置的屬性

[表格 4.2, “Multipath 的屬性”](#) 顯示了各個特定 `multipath` 裝置在 `multipath.conf` 配置檔案的 `multipaths` 一節中，您可設定的屬性。這些屬性只會用在該節所指定的 `multipath` 上。這些預設值會由 DM Multipath 所使用，並覆寫設定於 `multipath.conf` 檔案中的 `defaults` 與 `devices` 這兩節裡的屬性。

表格 4.2. Multipath 的屬性

屬性	描述
<code>wwid</code>	指定了 <code>multipath</code> 屬性所會套用的 <code>multipath</code> 裝置 WWID。此參數在 <code>multipath.conf</code> 檔案中的此部份中是必要的。
<code>alias</code>	為 <code>multipath</code> 裝置指定 <code>multipath</code> 屬性適用的符號名稱。若您正在使用 <code>user_friendly_names</code> ，請勿將這個值設為 <code>mpathn</code> ；這可能會造成與自動指定的使用者易記得名稱之間的衝突，並提供您錯誤的裝置節點名稱。
<code>path_grouping_policy</code>	指定預設路徑群組政策，以套用至未指定的 <code>multipath</code> 。可用的值包含了： <ul style="list-style-type: none"> <li><code>failover</code> = 每個優先群組一條路徑</li> <li><code>multibus</code> = 所有合於規定的路徑歸類在一個優先群組</li> <li><code>group_by_serial</code> = 每個偵測到的序號一個優先群組</li> <li><code>group_by_prio</code> = 每個路徑優先值一個優先群組</li> <li><code>group_by_node_name</code> = 每個目標節點名稱一個優先群組</li> </ul>
<code>path_selector</code>	指定用來決定下一項 I/O 作業時，該使用何路徑的預設演算法則。有效的值包含了： <ul style="list-style-type: none"> <li><code>round-robin 0</code>：在路徑群組中的所有路徑之間進行循環，並將相同的 I/O 數量傳送至各個路徑。</li> <li><code>queue-length 0</code>：將下個 I/O 量傳送至含有最少未處理之 I/O 請求數量的路徑。</li> <li><code>service-time 0</code>：將下個 I/O 量傳送至含有最短預估服務時間的路徑，這是透過將未處理的 I/O 總數，除以各個路徑的傳輸量來取得的。</li> </ul>

屬性	描述
<b>failback</b>	<p>管理路徑群組的容錯回復。</p> <p><b>immediate</b> 值可指定即刻容錯回復至包含了已啟用路徑的最高優先權之路徑群組。</p> <p><b>manual</b> 這個值指定容錯回復不會即刻發生，只有在運算子干預時，容錯回復才會發生。</p> <p><b>followover</b> 值指定當某個路徑群組中的第一條路徑啟用時，自動容錯回復便會進行。這會避免當有個節點請求容錯回復時，另一個節點自動進行容錯回復。</p> <p>超過零的數值表示遞延的容錯回復，單位為秒數。</p>
<b>prio</b>	<p>指定欲調用的預設函式，以取得路徑優先值。舉例來說，SPC-3 中的 ALUA 位元提供了一組可供利用的 <b>prio</b> 值。可使用的值包含了：</p> <p><b>const</b>：將所有路徑的優先值設為 1。</p> <p><b>emc</b>：為 EMC 陣列產生路徑優先權。</p> <p><b>alua</b>：根據 SCSI-3 ALUA 設定產生路徑優先權。</p> <p><b>ontap</b>：為 NetApp 陣列產生路徑優先權。</p> <p><b>rdac</b>：為 LSI/Engenio RDAC 控制器產生路徑優先權。</p> <p><b>hp_sw</b>：為 active/standby 模式的 Compaq/HP 控制器產生路徑優先權。</p> <p><b>hds</b>：為 Hitachi HDS Modular 儲存裝置陣列產生路徑優先權。</p>
<b>no_path_retry</b>	<p>這個屬性是用數字表示，它指定了系統在停用佇列之前，應試圖使用失效路徑的次數。</p> <p><b>fail</b> 表示立即失效，不進入佇列。</p> <p><b>queue</b> 值代表排程不應停止，直到路徑修正為止。</p>
<b>rr_min_io</b>	<p>指定欲路由轉送至某路徑的 I/O 請求數量（在切換至目前路徑群組中下個路徑之前）。此設定僅適用於執行比 2.6.31 還要舊的 kernel 的系統上。較新的系統應使用 <b>rr_min_io_rq</b>。預設值為 1000。</p>
<b>rr_min_io_rq</b>	<p>使用基於請求的 device-mapper-multipath 來指定欲路由轉送至某路徑的 I/O 請求數量（在切換至目前路徑群組中的下個路徑之前）。這項設定應使用於執行最新 kernel 的系統上。在執行比 2.6.31 還要舊的 kernel 的系統上，請使用 <b>rr_min_io</b>，預設值為 1。</p>
<b>rr_weight</b>	<p>如果設為 <b>priorities</b>，那麼與其在調用 <b>path_selector</b> 以選擇路徑之前發送 <b>rr_min_io</b> 請求至一條路徑上，此請求數量會取決於 <b>rr_min_io</b> 乘以路徑的優先值，如 <b>prio</b> 函式所決定。若設為了 <b>uniform</b>，所有路徑的權重皆會相等。</p>
<b>flush_on_last_del</b>	<p>若設為 <b>yes</b>，multipath 將會在某個裝置的最後一條路徑被刪除時，停用佇列。</p>

屬性	描述
<b>user_friendly_names</b>	如果設為 <b>yes</b> ，這表示系統應該使用 <code>/etc/multipath/bindings</code> 檔案來指定一個永續性且獨一無二的別名供 multipath 使用，格式為 <code>mpathn</code> 。若設為 <b>no</b> ，則表示系統應使用 WWID 來作為 multipath 的別名。無論是何種情況，在此所進行的設定，會被您在配置檔案的 <b>multipaths</b> 部份中，所指定的任何裝置特屬別名取代。
<b>delay_watch_checks</b>	(Red Hat Enterprise Linux Release 7.2 與更新版本) 若設為了大於零的值， <b>multipathd</b> daemon 將會以指定的檢查次數來監視近期成為有效的路徑。若這些路徑在被監視時再度失效，當它們下次成為有效時，在通過連續的檢測 (透過 <b>delay_wait_checks</b> 指定) 前，都不會再被使用。這能避免信賴度低的路徑在重新啟用後被即刻使用。
<b>delay_wait_checks</b>	(Red Hat Enterprise Linux 發行版 7.2 與更新版本) 若設為一個大於 0 的值，當一個近期重新啟用的裝置在指定次數的檢測 (透過 <b>delay_watch_checks</b> 指定) 進行中再次失效時，下次它再重新啟用時便會被標記和延緩，並且在它通過了 <b>delay_wait_checks</b> 中所指定得檢測次數之前，它將不會被使用。
<b>config_dir</b>	(Red Hat Enterprise Linux 發行版 7.2 與更新版本) 若設為 "" 以外的項目，multipath 將會以字母順序在此目錄中搜尋副檔名為 ".conf" 的檔案，並讀取其配置資訊，如同資訊在 <code>/etc/multipath.conf</code> 檔案中一般。這能讓您在使用多個機器特屬的配置檔案之外，亦可使用一個主要配置以共享於機器之間。 <b>config_dir</b> 這個參數必須是 "" 或是一組完整的目錄名稱。此參數僅能在設置於主要的 <code>/etc/multipath.conf</code> 檔案中，而不可在指定於 <b>config_dir</b> 檔案裡的檔案中設置。預設值為 <code>/etc/multipath/conf.d</code> 。
<b>deferred_remove</b>	若設為 <b>yes</b> ，當最後的路徑裝置被移除後，multipathd 將會進行一項延期的移除，而非正常的移除程序。這可確保當進行正常移除時，若有 multipath 裝置正被使用中而移除程序失敗的情況下，裝置將會在最後一個使用者關閉裝置時被自動移除。

以下範例顯示了 multipath 配置檔案裡面，針對兩個特定 multipath 裝置的屬性。第一個裝置的 WWID 是 **3600508b4000156d70001200000b0000**，符號名稱為 **yellow**。

例子中第二個 multipath 裝置的 WWID 為 **1DEC\_\_\_\_\_321816758474**，符號名稱是 **red**。在這個例子中，**rr\_weight** 這項屬性是設定為 **priorities**。

```

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
  }
}

```

## 4.5. 配置檔案裝置

[表格 4.3. “裝置屬性”](#) 顯示了 `multipath.conf` 配置檔案的 `devices` 部份中，您可為各個儲存裝置所設定的屬性。這些屬性會由 DM Multipath 所使用，除非它們被（包含了裝置的路徑所屬的）`multipath.conf` 配置檔案的 `multipaths` 一節中所指定的屬性覆寫。這些屬性會蓋過設定於 `multipath.conf` 檔案的 `defaults` 一節中的屬性。

就預設值，許多支援 multipath 的裝置皆包含在一個 multipath 配置中。欲取得更多有關於預設配置值（包括受支援的裝置）上的相關資訊，請執行下列指令之一。

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

您也許不用修改這些裝置的值，但如果您打算修改的話，您可覆寫這些預設值，方法是在配置檔案中為一項裝置寫入一條項目，覆蓋過這些值。您可從 `multipathd show config` 指令顯示的輸出複製某個裝置的裝置配置預設值，並覆寫您所希望改變的值。

若要在配置檔案的本節中，新增一項就預設值不會自動配置的裝置，您需要設定 `vendor` 與 `product` 參數。您可以尋找 `/sys/block/device_name/device/vendor` 與 `/sys/block/device_name/device/model`，以取得這些值，其中 `device_name` 代表需 multipath 的裝置，如以下範例所示：

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

需指定的額外參數取決於您的裝置。若裝置為 `active/active`，那麼您一般不需要設置額外的參數。您可能需要將 `path_grouping_policy` 設為 `multibus`。其它您可能會需要設置的參數包括 `no_path_retry` 與 `rr_min_io`，如〈[表格 4.3. “裝置屬性”](#)〉中所述。

若裝置為 `active/passive`，但會自動將 I/O 切換至 `passive` 路徑，那麼您便需要將檢查功能改變為不會發送 I/O 至路徑，以測試它是否能成功運作（否則，您的裝置將會持續進入備援狀態）。這意味著您幾乎總是要將 `path_checker` 設為 `tur`；這對所有支援 Test Unit Ready 指令的 SCSI 裝置（大部分皆支援）來說皆通用。

若裝置需要特殊的指令來切換路徑，那麼為這項裝置配置 multipath，便需要硬體 kernel 處理模組。目前可用的硬體處理程式為 `emc`。若這對於您的裝置來說不夠用，您可能無法為裝置配置 multipath。

表格 4.3. 裝置屬性

屬性	描述
<code>vendor</code>	指定儲存裝置的製造商名稱，供 <code>device</code> 屬性使用；例如 <b>COMPAQ</b> 。
<code>product</code>	指定 <code>device</code> 屬性所使用的儲存裝置之產品名稱，例如 <b>HSV110 (C)COMPAQ</b> 。
<code>revision</code>	指定儲存裝置的產品修訂識別子。
<code>product_blacklist</code>	指定常規表示式，藉由產品名稱將裝置納入黑名單。
<code>hardware_handler</code>	指定用來在切換路徑群組或處理 I/O 錯誤時，進行硬體特屬動作的模組。可能的值包含了： <ul style="list-style-type: none"> <li><b>1 emc</b> : EMC 儲存裝置陣列的硬體處理程式。</li> <li><b>1 alua</b> : SCSI-3 ALUA 陣列的硬體處理程式。</li> <li><b>1 hp_sw</b> : Compaq/HP 控制器的硬體處理程式。</li> <li><b>1 rdac</b> : LSI/Engenio RDAC 控制器的硬體處理程式。</li> </ul>

屬性	描述
<b>path_grouping_policy</b>	<p>指定預設路徑群組政策，以套用至未指定的 multipath。可用的值包含了：</p> <p><b>failover</b> = 每個優先群組一條路徑</p> <p><b>multibus</b> = 所有合於規定的路徑歸類在一個優先群組</p> <p><b>group_by_serial</b> = 每個偵測到的序號一個優先群組</p> <p><b>group_by_prio</b> = 每個路徑優先值一個優先群組</p> <p><b>group_by_node_name</b> = 每個目標節點名稱一個優先群組</p>
<b>path_selector</b>	<p>指定用來決定下一項 I/O 作業時，該使用何路徑的預設演算法則。有效的值包含了：</p> <p><b>round-robin 0</b>：在路徑群組中的所有路徑之間進行循環，並將相同的 I/O 數量傳送至各個路徑。</p> <p><b>queue-length 0</b>：將下個 I/O 量傳送至含有最少未處理之 I/O 請求數量的路徑。</p> <p><b>service-time 0</b>：將下個 I/O 量傳送至含有最短預估服務時間的路徑，這是透過將未處理的 I/O 總數，除以各個路徑的傳輸量來取得的。</p>
<b>path_checker</b>	<p>指定用來判斷路徑狀態的預設方式。可用的值包含了：</p> <p><b>readsector0</b>：讀取裝置的第一個磁區。</p> <p><b>tur</b>：發送一項 <b>TEST UNIT READY</b> 至裝置。</p> <p><b>emc_clariion</b>：查詢 EMC Clariion 專屬的 EVPD 分頁 0xC0 以判斷路徑。</p> <p><b>hp_sw</b>：檢查含有 Active/Standby 軟體之 HP 儲存裝置陣列的路徑狀態。</p> <p><b>rdac</b>：檢查 LSI/Engenio RDAC 儲存控制器的路徑狀態。</p> <p><b>directio</b>：讀取第一個含有直接 I/O 的磁區。</p>

屬性	描述
<b>features</b>	<p>multipath 裝置的預設額外特性，格式為：<i>"number_of_features_plus_arguments feature1 ..."</i>。</p> <p><b>features</b> 可能的值為：</p> <p>唯一存在的功能為 <b>queue_if_no_path</b>，這與將 <b>no_path_retry</b> 設為 <b>queue</b> 相同。欲查看使用此功能時所可能發生的問題，請參閱〈<a href="#">節 5.7, "queue_if_no_path 功能上的問題"</a>〉。</p> <p><b>retain_attached_hw_handler</b>：如果此參數設為 <b>yes</b>，並且 SCSI 層已經設為路徑裝置的硬體處理程式，那麼 multipath 不會強迫裝置使用 <b>multipath.conf</b> 檔案所指定的 <b>hardware_handler</b>。如果 SCSI 層並未連結硬體處理程式，multipath 會繼續如常使用其配置的硬體處理程式。</p> <p><b>pg_init_retries n</b>：重試路徑群組的初始化過程，最多 <i>n</i> 次，其中 <math>1 \leq n \leq 50</math>。</p> <p><b>pg_init_delay_msecs n</b>：路徑群組初始化重試之間，等待 <i>n</i> 毫秒，其中 <math>0 \leq n \leq 60000</math>。</p>
<b>prio</b>	<p>指定欲調用的預設函式，以取得路徑優先值。舉例來說，SPC-3 中的 ALUA 位元提供了一組可供利用的 <b>prio</b> 值。可使用的值包含了：</p> <p><b>const</b>：將所有路徑的優先值設為 1。</p> <p><b>emc</b>：為 EMC 陣列產生路徑優先權。</p> <p><b>alua</b>：根據 SCSI-3 ALUA 設定產生路徑優先權。</p> <p><b>ontap</b>：為 NetApp 陣列產生路徑優先權。</p> <p><b>rdac</b>：為 LSI/Engenio RDAC 控制器產生路徑優先權。</p> <p><b>hp_sw</b>：為 active/standby 模式的 Compaq/HP 控制器產生路徑優先權。</p> <p><b>hds</b>：為 Hitachi HDS Modular 儲存裝置陣列產生路徑優先權。</p>
<b>failback</b>	<p>管理路徑群組的容錯回復。</p> <p><b>immediate</b> 值可指定即刻容錯回復至包含了已啟用路徑的最高優先權之路徑群組。</p> <p><b>manual</b> 這個值指定容錯回復不會即刻發生，只有在運算子干預時，容錯回復才會發生。</p> <p><b>followover</b> 值指定當某個路徑群組中的第一條路徑啟用時，自動容錯回復便會進行。這會避免當有個節點請求容錯回復時，另一個節點自動進行容錯回復。</p> <p>超過零的數值表示遞延的容錯回復，單位為秒數。</p>
<b>rr_weight</b>	<p>如果設為 <b>priorities</b>，那麼與其在調用 <b>path_selector</b> 以選擇路徑之前發送 <b>rr_min_io</b> 請求至一條路徑上，此請求數量會取決於 <b>rr_min_io</b> 乘以路徑的優先值，如 <b>prio</b> 函式所決定。若設為了 <b>uniform</b>，所有路徑的權重皆會相等。</p>

屬性	描述
<b>no_path_retry</b>	<p>這個屬性是用數字表示，它指定了系統在停用佇列之前，應試圖使用失效路徑的次數。</p> <p><b>fail</b> 表示立即失效，不進入佇列。</p> <p><b>queue</b> 值代表排程不應停止，直到路徑修正為止。</p>
<b>rr_min_io</b>	指定欲路由轉送至某路徑的 I/O 請求數量（在切換至目前路徑群組中下個路徑之前）。此設定僅適用於執行比 2.6.31 還要舊的 kernel 的系統上。較新的系統應使用 <b>rr_min_io_rq</b> 。預設值為 1000。
<b>rr_min_io_rq</b>	使用基於請求的 device-mapper-multipath 來指定欲路由轉送至某路徑的 I/O 請求數量（在切換至目前路徑群組中的下個路徑之前）。這項設定應使用於執行最新 kernel 的系統上。在執行比 2.6.31 還要舊的 kernel 的系統上，請使用 <b>rr_min_io</b> ，預設值為 1。
<b>fast_io_fail_tmo</b>	在一個 FC 遠端連接埠上偵測到問題後，SCSI 層在使該遠端連接埠上的裝置 I/O 失效前，所會等待的秒數。這個值應比 <b>dev_loss_tmo</b> 的值還要小。把這設為 <b>off</b> 將會停用逾時。
<b>dev_loss_tmo</b>	當在一個 FC 遠端連接埠上偵測到錯誤時，SCSI 層將該連接埠由系統移除之前，所等待的秒數。將此設為無限制，會將其設為 2147483647 秒，或 68 年。
<b>flush_on_last_del</b>	若設為 <b>yes</b> ，當連上裝置的最後一條路徑被刪除時， <b>multipathd</b> daemon 便會停用佇列功能。
<b>user_friendly_names</b>	如果設為 <b>yes</b> ，這表示系統應該使用 <code>/etc/multipath/bindings</code> 檔案來指定一個永續性且獨一無二的別名供 multipath 使用，格式為 <b>mpathn</b> 。若設為 <b>no</b> ，則表示系統應使用 WWID 來作為 multipath 的別名。無論是何種情況，在此所進行的設定，會被您在配置檔案的 <b>multipaths</b> 部份中，所指定的任何裝置專屬別名取代。預設值為 <b>no</b> 。
<b>retain_attached_hw_handler</b>	如果此參數設為 <b>yes</b> ，並且 SCSI 層已經設為路徑裝置的硬體處理程式，那麼 multipath 不會強迫裝置使用 <b>multipath.conf</b> 檔案所指定的 <b>hardware_handler</b> 。如果 SCSI 層並未連結硬體處理程式，multipath 會繼續如常使用其配置的硬體處理程式。預設值為 <b>no</b> 。
<b>detect_prio</b>	若設為 <b>yes</b> ，multipath 會先檢查裝置是否支援 ALUA，如果支援的話，multipath 會自動指定裝置為 <b>alua</b> 優先程式。如果裝置不支援 ALUA，multipath 會如常決定優先程式。
<b>delay_watch_checks</b>	（Red Hat Enterprise Linux Release 7.2 與更新版本）若設為了大於零的值， <b>multipathd</b> daemon 將會以指定的檢查次數來監視近期成為有效的路徑。若這些路徑在被監視時再度失效，當它們下次成為有效時，在通過連續的檢測（透過 <b>delay_wait_checks</b> 指定）前，都不會再被使用。這能避免信賴度低的路徑在重新啟用後被即刻使用。
<b>delay_wait_checks</b>	（Red Hat Enterprise Linux 發行版 7.2 與更新版本）若設為一個大於 0 的值，當一個近期重新啟用的裝置在指定次數的檢測（透過 <b>delay_watch_checks</b> 指定）進行中再次失效時，下次它再重新啟用時便會被標記和延緩，並且在它通過了 <b>delay_wait_checks</b> 中所指定得檢測次數之前，它將不會被使用。
<b>deferred_remove</b>	若設為 <b>yes</b> ，當最後的路徑裝置被移除後， <b>multipathd</b> 將會進行一項延期的移除，而非正常的移除程序。這可確保當進行正常移除時，若有 multipath 裝置正被使用中而移除程序失敗的情況下，裝置將會在最後一個使用者關閉裝置時被自動移除。

以下範例顯示了 multipath 配置檔案中，**device** 一節。

```
# }
# device {
```

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

## 章 5. DM Multipath 管理與疑難排解

本章提供了有關於在一步運作中的系統上管理 DM Multipath 的相關資訊。

### 5.1. 透過 Multipath Helper 自動產生配置檔案

您可透過 Multipath Helper 應用程式來在 RHEL 上為 multipath 裝置產生一項基本的配置。該應用程式能提供您建立 multipath 配置，並為個別 multipath 裝置進行自訂別名、裝置黑名單與行為上的設定。當完成時，應用程式會產生一個安裝 script，它包含了您所選擇的配置參數，並會提供一個供您檢視的 **multipath.conf** 配置檔案。

Multipath Helper 應用程式能在 <https://access.redhat.com/labsinfo/multipathhelper> 找到。

### 5.2. 重設線上 Multipath 裝置的大小

若您需要重設一個線上 multipath 裝置的大小，請依照下列程序進行。

1. 重設您實體裝置的大小。
2. 執行下列指令來找出 LUN 的路徑：

```
# multipath -l
```

3. 重設您路徑的大小。若是 SCSI 裝置的話，請將 1 寫至造成 SCSI 重新掃描的裝置的 **rescan** 檔案中，如下列指令所示：

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

4. 請藉由執行 **multipathd resize** 指令來重設您 multipath 裝置的大小：

```
# multipathd -k'resize map mpatha'
```

5. 重設檔案系統的大小（假設未使用 LVM 或 DOS 分割區）：

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

### 5.3. 將 root 檔案系統從單路徑裝置移到多路徑裝置

如果您在單一路徑的裝置上安裝系統，之後加入另一個路徑指向 root 檔案系統，您就需要將 root 檔案系統移到多路徑的裝置上。本節記述了從單路徑移到多路徑裝置的步驟。

確定安裝了 **device-mapper-multipath** 套件之後，請進行以下步驟：

1. 執行以下指令以建立 **/etc/multipath.conf** 配置檔、載入 multipath 模組、並使用 **chkconfig** 指令將 **multipathd** 服務設為 on：

```
# mpathconf --enable
```

欲取得更多有關於使用 **multipath** 指令來設定 multipath 的相關資訊，請參閱 [〈節 3.1, “設定 DM Multipath”〉](#)。

- 若 `find_multipaths` 配置參數未設為 `yes`，請編輯 `/etc/multipath.conf` 檔案的 `blacklist` 和 `blacklist_exceptions` 部分，如〈[節 4.2, “配置檔案的黑名單”](#)〉中所述。
- 若要讓 `multipath` 能在發現 `root` 裝置後即刻在該裝置上建立一項 `multipath` 裝置，請執行以下指令。這項指令亦可確保 `find_multipaths` 會允許裝置，儘管它只有一條路徑。

```
# multipath -a root_devname
```

比方說，若 `root` 裝置為 `/dev/sdb`，請執行下列指令。

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

- 若要確認您的配置檔案是否設定正確，您可執行 `multipath` 指令並在輸出中搜尋格式如下的一行。這代表指令並未成功建立 `multipath` 裝置。

```
date wwid: ignoring map
```

比方說，若裝置的 WWID 為 `3600d02300069c9ce09d41c4ac9c53200`，您將會在輸出中看見類似以下的一行：

```
# multipath
Oct 21 09:37:19 | 3600d02300069c9ce09d41c4ac9c53200: ignoring map
```

- 若要以 `multipath` 重建 `initramfs` 檔案系統，請搭配下列選項執行 `dracut` 指令：

```
# dracut --force -H --add multipath
```

- 將電腦關機。
- 配置光纖交換器，讓電腦可以看到其它路徑。
- 將電腦開機。
- 檢查 `root` 檔案系統 (`/`) 是否位於多路徑的裝置上。

## 5.4. 將 swap 檔案系統從單路徑裝置移到多路徑裝置

預設上，`swap` 裝置設於邏輯卷冊上。這並不需要特別的步驟，也不需要將其配置為 `multipath` 裝置；只要您將 `multipath` 設在含有邏輯卷冊群組的那個實體卷冊上即可。然而，如果您的 `swap` 裝置不是 LVM 卷冊，而且是透過裝置名稱掛載，請編輯 `/etc/fstab` 檔案，切換為適當的 `multipath` 裝置名稱。

- 執行 `/sbin/multipath` 指令與 `-v3` 選項，決定 `swap` 裝置的 WWID。指令輸出會在路徑清單中，顯示 `swap` 裝置。

請查看指令輸出中，類似以下格式的一行，尋找 `swap` 裝置：

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

舉例來說，如果您的 `swap` 檔案系統設在 `sda` 或 `sda` 的分割區上，您會看到類似以下一行：

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

- 在 `/etc/multipath.conf` 檔案中，設定 swap 裝置的別名：

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

- 編輯 `/etc/fstab` 檔案，用 root 裝置的 multipath 裝置取代舊的裝置路徑。

舉例來說，如果您的 `/etc/fstab` 檔案有以下項目：

```
/dev/sda2 swap          ext4 defaults    0 0
```

請將項目更改為：

```
/dev/mapper/swapdev swap      ext4 defaults    0 0
```

## 5.5. Multipath Daemon

如果您在實做 multipath 配置時遇到困難，請確認 multipath daemon 運作正常，如 [〈章 3, 設定 DM Multipath〉](#) 所述。若要使用 multipath 裝置，必須執行 `multipathd` daemon。

## 5.6. 大量 LUN 的問題

當大量的 LUN 加入節點時，使用 multipath 裝置會明顯增加 `udev` 裝置管理員為這些 LUN 建立裝置節點的時間。如果您遇到此問題，修正方式是刪除 `/etc/udev/rules.d/40-multipath.rules` 裡面的一行：

```
KERNEL!="dm-[0-9]*", ACTION=="add", PROGRAM==" /bin/bash -c '/sbin/lsmode | /bin/grep
^dm_multipath'", RUN+="/sbin/multipath -v0 %M:%m"
```

這一行會導致每次一個區塊裝置加入節點時，`udev` 裝置管理員就會執行 `multipath`。即使此行被移除，`multipathd` daemon 仍然會自動建立 multipath 裝置，同時 `multipath` 還是會由搭配 multipath 的 root 檔案系統之節點開機程序所執行。唯一不同的是，當 `multipathd` daemon 沒有被執行時，multipath 裝置不會自動建立，這對大部分 multipath 使用者來說，不會造成問題。

## 5.7. queue\_if\_no\_path 功能上的問題

如果一項 multipath 裝置配置了 `features "1 queue_if_no_path"`，那麼任何發出 I/O 的程序都會暫停，直到一或多條路徑回復了為止。若要避免此問題，您可以在 `/etc/multipath.conf` 檔案中設定 `no_path_retry N` 參數（其中 `N` 代表系統應該重新嘗試路徑的次數）。

如果您需要使用 `features "1 queue_if_no_path"` 選項，同時也遇到此處所述的問題，請使用 `dmsetup` 指令，並在 runtime 時為特定的 LUN（也就是針對所有無法使用的路徑）編輯政策。舉例來說，如果您想要將 `multipath` 裝置上的政策由 `"queue_if_no_path"` 改變成 `"fail_if_no_path"`，請執行下列指令。

```
dmsetup message mpathc 0 "fail_if_no_path"
```

請注意，您必須指定 **mpath $n$**  的別名，而不是路徑。

## 5.8. Multipath 指令的輸出

當您建立、修改、或列出 multipath 裝置時，您會看到目前裝置設定的訊息輸出，格式如下。

每個 multipath 裝置都有：

```
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
```

每個路徑群組都有：

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

每個路徑都有：

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

舉例來說，multipath 指令的輸出可能看似：

```
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
```

如果路徑已經啟動、也準備好給 I/O 使用，那麼路徑的狀態會顯示 **ready** 或 **ghost**；反之會顯示 **faulty** 或 **shaky**。路徑狀態會由 **multipathd** daemon 定期更新，更新的頻率是在 **/etc/multipath.conf** 檔案中決定。

dm 狀態跟路徑狀態類似，但是從核心的觀點來看則否。dm 狀態有兩種狀態：**failed**（相當於 **faulty**）與 **active**（包含所有其它的路徑狀態）。偶爾，裝置的路徑狀態與 dm 狀態會暫時不同步。

**online\_status** 可使用的值為 **running** 和 **offline**。**offline** 狀態代表此 SCSI 裝置已停用。



### 注意

當 multipath 裝置被建立或修改時，路徑群組的狀態、dm 裝置名稱、寫入權限和 dm 狀態都是未知的。同時，這些功能並不永遠都是正確的。

## 5.9. 使用 multipath 指令來進行 Multipath 查詢

您可以使用 **multipath** 指令的 **-l** 與 **-ll** 選項來顯示目前的 multipath 設定。**-l** 選項會顯示自 **sysfs** 與 device mapper 蒐集而來的 multipath 拓樸。**-ll** 選項會顯示 **-l** 選項的資訊，加上所有系統的其它元件資訊。

當顯示 multipath 設定時，您可以用 **multipath** 指令的 **-v** 選項，顯示三種訊息等級。使用 **-v0** 表示沒有任何輸出。**-v1** 會列出已建立、以更新的 multipath 名稱，您可以將這些名稱導向到其它程式去，例如 **kpartx**。使用 **-v2** 會列出所有已偵測到的路徑、multipath、以及 device map。

以下範例顯示了 **multipath -l** 指令的輸出。

```
# 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
```

以下範例顯示了 **multipath -ll** 指令的輸出。

```
# 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
```

## 5.10. Multipath 的指令選項

[表格 5.1. “實用的 multipath 指令選項”](#) 詳述了一些您可能會覺得有幫助的 **multipath** 指令選項。

表格 5.1. 實用的 multipath 指令選項

選項	描述
<b>-l</b>	顯示從 <b>sysfs</b> 與 device mapper 蒐集而來的 multipath 設定。
<b>-ll</b>	顯示從 <b>sysfs</b> 、device mapper、與其它系統的可用元件蒐集而來的 multipath 設定。
<b>-f 裝置</b>	移除指定的 multipath 裝置。
<b>-F</b>	移除所有未使用的 multipath 裝置。
<b>-w device</b>	移除 <b>wwids</b> 檔案中，指定裝置的 <b>wwid</b> 。
<b>-W</b>	重新設定 <b>wwids</b> 檔案，以僅納入現有的 multipath 裝置。

部分 **multipathd** 指令包含了一項 **format** 選項，並接著一個萬用字元。您可藉由以下指令顯示一系列可用的萬用字元。

```
# multipathd show wildcards
```

## 5.11. 使用 dmsetup 指令，決定 device mapper 的項目

您可以使用 **dmsetup** 指令，找出哪個 device mapper 項目與 multipath 裝置相符。

以下指令顯示所有 device mapper 裝置與其主要及次要號碼。次要號碼決定了 dm 裝置的名稱。舉例來說，次要號碼「3」指得是 multipath 裝置 **/dev/dm-3**。

```
# 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)
```

## 5.12. 使用 multipathd 的互動式主控台進行疑難排解

**multipathd -k** 指令是 **multipathd** daemon 的互動式介面。執行這個指令會帶出互動式 multipath 主控台。輸入這項指令之後，您可以輸入 **help** 取得可用指令的清單，您可以輸入一項互動式的指令，或按下 **CTRL-D** 離開。

**multipathd** 互動式主控台可以用來為系統進行疑難排解。舉例來說，以下指令會在離開主控台之前，顯示 multipath 設定（包括預設設定）。

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

以下指令會確保 multipath 取得 **multipath.conf** 的任何改變。

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

使用以下指令可以確保路徑檢查程式運作正常。

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

## 附錄 A. 修訂記錄

<b>修訂 0.3-6.1</b>	<b>Fri Jul 7 2017</b>	<b>Terry Chuang</b>
讓翻譯檔案與 XML 來源 0.3-6 同步		
<b>修訂 0.3-6</b>	<b>Mon Nov 9 2015</b>	<b>Steven Levine</b>
準備文件，出版 7.2 GA		
<b>修訂 0.3-3</b>	<b>Wed Aug 19 2015</b>	<b>Steven Levine</b>
準備文件，出版 7.2 Beta		
<b>修訂 0.2-7</b>	<b>Mon Feb 16 2015</b>	<b>Steven Levine</b>
7.1 GA 發行版本		
<b>修訂 0.2-6</b>	<b>Thu Dec 11 2014</b>	<b>Steven Levine</b>
7.1 Beta 發行版本		
<b>修訂 0.2-2</b>	<b>Tue Dec 2 2014</b>	<b>Steven Levine</b>
解決：#1162514、#1085979、#1129897 小幅更新了技術內容修正。		
解決：#1117037 記載了 multipath 指令的 -w 和 -W 選項。		
解決：#1117043 記載了功能配置參數的新選項。		
解決：#1117499 記載了 -force_sync 配置參數。		
<b>修訂 0.2-1</b>	<b>Thu Oct 30 2014</b>	<b>Steven Levine</b>
建置更新程序以移動 7.0 的 root 檔案系統		
<b>修訂 0.2-0</b>	<b>Wed Oct 22 2014</b>	<b>Steven Levine</b>
更新了移動 root 檔案系統的程序		
<b>修訂 0.1-22</b>	<b>Mon Jun 2 2014</b>	<b>Steven Levine</b>
7.0 GA 發行版本		
<b>修訂 0.1-20</b>	<b>Tue May 20 2014</b>	<b>Steven Levine</b>
為風格上的變更而重建文件		
解決：#1069212、#1074858 記載了新的與更新過後的參數		
解決：#1040328 小幅技術修正		
<b>修訂 0.1-10</b>	<b>Wed Apr 9 2014</b>	<b>Steven Levine</b>
7.0 Beta 更新		
<b>修訂 0.1-3</b>	<b>Tue Nov 26 2013</b>	<b>Steven Levine</b>

7.0 pre-Beta 發行版本

修訂 0.1-2	Tue Nov 12 2013	Steven Levine
移除 RHEL 6 的參照。		

修訂 0.1-1	Wed Jan 16 2013	Steven Levine
由本文件的 Red Hat Enterprise Linux 6 版本分支		

## 索引

### 符號

[/etc/multipath.conf 套件, 設定 DM Multipath](#)

使用 multipath 的 root 檔案系統, [將 root 檔案系統從單路徑裝置移到多路徑裝置](#)

使用 multipath 的 swap 檔案系統, [將 swap 檔案系統從單路徑裝置移到多路徑裝置](#)

備援, [DM Multipath 總覽](#)

儲存裝置陣列

- 新增, [配置儲存裝置](#), [配置檔案裝置](#)

儲存陣列的支援, [儲存陣列的支援](#)

新功能與更改的功能, [新功能與更改的功能](#)

本機磁碟, 忽略, [產生 multipath 裝置時, 忽略本機磁碟](#)

總覽

- 新功能與更改的功能, [新功能與更改的功能](#)

裝置

- 新增, [配置儲存裝置](#), [配置檔案裝置](#)

裝置名稱, [Multipath 的裝置識別子](#)

設定

- DM Multipath, [設定 DM Multipath](#)

配置

- DM Multipath, [設定 DM Multipath](#)

配置檔案

- alias 參數, [Multipath 裝置配置的屬性](#)
- blacklist, [配置檔案的黑名單](#)
- checker\_timeout 參數, [配置檔案的預設值](#)
- config\_dir 參數, [Multipath 裝置配置的屬性](#)
- deferred\_remove 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- delay\_wait\_checks 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- delay\_watch\_checks 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- dev\_loss\_tmo 參數, [配置檔案的預設值](#), [配置檔案裝置](#)
- failback 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- fast\_io\_fail\_tmo 參數, [配置檔案的預設值](#), [配置檔案裝置](#)
- features 參數, [配置檔案的預設值](#), [配置檔案裝置](#)
- flush\_on\_last\_del 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- force\_sync 參數, [配置檔案的預設值](#)

- hardware\_handler 參數, [配置檔案裝置](#)
- ignore\_new\_boot\_devs 參數, [配置檔案的預設值](#)
- log\_checker\_err 參數, [配置檔案的預設值](#)
- max\_fds 參數, [配置檔案的預設值](#)
- new\_bindings\_in\_boot 參數, [配置檔案的預設值](#)
- no\_path\_retry 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- path\_checker 參數, [配置檔案的預設值](#), [配置檔案裝置](#)
- path\_grouping\_policy 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- path\_selector 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- polling-interval 參數, [配置檔案的預設值](#)
- prio 參數, [配置檔案的預設值](#), [配置檔案裝置](#)
- product 參數, [配置檔案裝置](#)
- product\_blacklist 參數, [配置檔案裝置](#)
- queue\_without\_daemon 參數, [配置檔案的預設值](#)
- retain\_attached\_hw\_handler 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#)
- retrigger\_delay 參數, [配置檔案的預設值](#)
- retrigger\_tries 參數, [配置檔案的預設值](#)
- revision 參數, [配置檔案裝置](#)
- rr\_min\_io 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#)
- rr\_weight 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- uid\_attribute 參數, [配置檔案的預設值](#)
- user\_friendly\_names 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)
- vendor 參數, [配置檔案裝置](#)
- verbosity 參數, [配置檔案的預設值](#)
- wwid 參數, [Multipath 裝置配置的屬性](#)
- 總覽, [配置檔案總覽](#)

## 重設 multipath 裝置的大小, [重設線上 Multipath 裝置的大小](#)

### A

#### active/active configuration

- illustration, [DM Multipath 總覽](#)

#### active/active 配置

- 定義, [DM Multipath 總覽](#)

#### active/passive 配置

- 定義, [DM Multipath 總覽](#)
- 描述, [DM Multipath 總覽](#)

#### alias 參數, [Multipath 裝置配置的屬性](#)

- 配置檔案, [Multipath 的裝置識別子](#)

### B

#### blacklist

- WWID, [根據 WWID 建立黑名單](#)
- 裝置名稱, [根據裝置名稱來建立黑名單](#)
- 裝置類型, [根據裝置類型來建立黑名單](#)
- 配置檔案, [配置檔案的黑名單](#)
- 預設裝置, [根據裝置名稱來建立黑名單](#)

#### blacklist\_exceptions 部份

- multipath.conf 檔案, [黑名單的例外](#)

### C

**checker\_timeout** 參數, [配置檔案的預設值](#)

**configuration file**

- **detect\_prio** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#)

**config\_dir** 參數, [Multipath 裝置配置的屬性](#)

## D

**defaults** 部份

- **multipath.conf** 檔案, [配置檔案的預設值](#)

**deferred\_remove** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

**delay\_wait\_checks** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

**delay\_watch\_checks** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

**detect\_prio** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#)

**dev/mapper** 目錄, [Multipath 的裝置識別子](#)

**device-mapper-multipath** 套件, [設定 DM Multipath](#)

**devices** 部份

- **multipath.conf** 檔案, [配置檔案裝置](#)

**dev\_loss\_tmo** 參數, [配置檔案的預設值](#), [配置檔案裝置](#)

**DM Multipath**

- 備援, [DM Multipath 總覽](#)
- 元件, [DM Multipath 元件](#)
- 冗餘, [DM Multipath 總覽](#)
- 定義, [Device Mapper Multipathing](#)
- 總覽, [DM Multipath 總覽](#)
- 與 LVM, [邏輯卷冊中的 multipath 裝置](#)
- 裝置, [Multipath 裝置](#)
- 裝置名稱, [Multipath 的裝置識別子](#)
- 設定, [設定 DM Multipath](#)
- 設定, 總覽, [DM Multipath 設定總覽](#)
- 配置, [設定 DM Multipath](#)
- 配置檔案, [DM Multipath 配置檔案](#)

**dm-n** 裝置, [Multipath 的裝置識別子](#)

**dmsetup** 指令, 決定 device mapper 的項目, [使用 dmsetup 指令, 決定 device mapper 的項目](#)

**dm\_multipath** kernel module, [DM Multipath 元件](#)

## F

**failback** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

**fast\_io\_fail\_tmo** 參數, [配置檔案的預設值](#), [配置檔案裝置](#)

**features** 參數, [配置檔案的預設值](#), [配置檔案裝置](#)

**flush\_on\_last\_del** 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

**force\_sync** 參數, [配置檔案的預設值](#)

## H

**hardware\_handler** 參數, [配置檔案裝置](#)

## I

`ignore_new_boot_devs` 參數, [配置檔案的預設值](#)

`initramfs`

- 啟動 `multipath`, [在 `initramfs` 檔案系統中設定 `multipath`](#)

## K

`kpartx` command , [DM Multipath 元件](#)

## L

`log_checker_err` 參數, [配置檔案的預設值](#)

LVM 實體卷冊

- `multipath` 裝置, [邏輯卷冊中的 `multipath` 裝置](#)

`lvm.conf` 檔案, [邏輯卷冊中的 `multipath` 裝置](#)

## M

`max_fds` 參數, [配置檔案的預設值](#)

`mpathconf` command , [DM Multipath 元件](#)

`multipath` command , [DM Multipath 元件](#)

`multipath` daemon (`multipathd`) , [Multipath Daemon](#)

Multipath Helper, [透過 Multipath Helper 自動產生配置檔案](#)

`multipath` 指令

- 查詢, [使用 `multipath` 指令來進行 Multipath 查詢](#)
- 輸出, [Multipath 指令的輸出](#)
- 選項, [Multipath 的指令選項](#)

`multipath` 裝置, [Multipath 裝置](#)

- LVM 實體卷冊, [邏輯卷冊中的 `multipath` 裝置](#)
- 邏輯卷冊, [邏輯卷冊中的 `multipath` 裝置](#)

`multipath.conf` 檔案, [儲存陣列的支援, DM Multipath 配置檔案](#)

- `blacklist_exceptions` 部份, [黑名單的例外](#)
- `defaults` 部份, [配置檔案的預設值](#)
- `devices` 部份, [配置檔案裝置](#)
- `multipaths` 部份, [Multipath 裝置配置的屬性](#)

`multipathd`

- 互動式主控台, [使用 `multipathd` 的互動式主控台進行疑難排解](#)
- 指令, [使用 `multipathd` 的互動式主控台進行疑難排解](#)

`multipathd` daemon , [DM Multipath 元件](#)

`multipathd` start 指令, [設定 DM Multipath](#)

`multipaths` 部份

- `multipath.conf` 檔案, [Multipath 裝置配置的屬性](#)

## N

`new_bindings_in_boot` 參數, [配置檔案的預設值](#)

`no_path_retry` 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

## P

`path_checker` 參數, [配置檔案的預設值](#), [配置檔案裝置](#)

`path_grouping_policy` 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)  
`path_selector` 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)  
`polling_interval` 參數, [配置檔案的預設值](#)  
`prio` 參數, [配置檔案的預設值](#), [配置檔案裝置](#)  
`product` 參數, [配置檔案裝置](#)  
`product_blacklist` 參數, [配置檔案裝置](#)

## Q

`queue_without_daemon` 參數, [配置檔案的預設值](#)  
`quota_quantum_tunable_parameter`, [配置檔案的預設值](#)

## R

`retain_attached_hw_handler` 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#)  
`retrigger_delay` 參數, [配置檔案的預設值](#)  
`retrigger_tries` 參數, [配置檔案的預設值](#)  
`revision` 參數, [配置檔案裝置](#)  
`root` 檔案系統, [將 root 檔案系統從單路徑裝置移到多路徑裝置](#)  
`rr_min_io` 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#)  
`rr_weight` 參數, [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

## S

`swap` 檔案系統, [將 swap 檔案系統從單路徑裝置移到多路徑裝置](#)

## U

`uid_attribute` 參數, [配置檔案的預設值](#)  
`user_friendly_names` 參數, [Multipath 的裝置識別子](#), [配置檔案的預設值](#), [Multipath 裝置配置的屬性](#), [配置檔案裝置](#)

## V

`vendor` 參數, [配置檔案裝置](#)  
`verbosity` 參數, [配置檔案的預設值](#)

## W

World Wide Identifier (WWID) , [Multipath 的裝置識別子](#)  
`wwid` 參數, [Multipath 裝置配置的屬性](#)