



红帽企业版 **Linux 6**

设备映射多路径

设备映射多路径（DM Multipath）配置及管理

红帽 工程内容服务

设备映射多路径 (DM Multipath) 配置及管理

红帽 工程内容服务

docs-need-a-fix@redhat.com

法律通告

Copyright 2013 Red Hat, Inc. and others. 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, MetaMatrix, 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. All other trademarks are the property of their respective owners. 1801 Varsity Drive Raleigh, NC 27606-2072 USA Phone: +1 919 754 3700 Phone: 888 733 4281 Fax: +1 919 754 3701

Keywords

摘要

本数提供有关红帽企业版 Linux 设备映射器多路经功能的信息。

目录

前言	5
1. 读者	5
2. 相关文档	5
3. 我们需要您的反馈！	5
4. 文档约定	5
4.1. 排版约定	5
4.2. 抬升式引用约定	7
4.3. 备注及警告	7
第 1 章 设备映射器多路径	9
1.1. 新的和更改的功能	9
1.1.1. 红帽企业版 Linux 6.0 中新的和更改的功能	9
1.1.2. 红帽企业版 Linux 6.1 中新的和更改的功能	10
1.1.3. 红帽企业版 Linux 6.2 中新的和更改的功能	10
1.1.4. 红帽企业版 Linux 6.3 中新的和更改的功能	10
1.1.5. 红帽企业版 Linux 6.4 中新的和更改的功能	10
1.2. DM-Multipath 概述	10
1.3. 存储阵列支持	12
1.4. DM-Multipath 组件	12
1.5. DM-Multipath 设置概述	13
第 2 章 多路径设备	14
2.1. 多路径设备识别符	14
2.2. 在集群中保持多路径设备名称一致	14
2.3. 多路径设备属性	15
2.4. 逻辑卷中的多路径设备	15
第 3 章 设置 DM-Multipath	16
3.1. 设置 DM-Multipath	16
3.2. 在生成多路径设备时忽略逻辑磁盘	17
3.3. 配置存储设备	19
3.4. 在 initramfs 文件系统中设置启动 Multipath	19
第 4 章 DM-Multipath 配置文件	21
4.1. 配置文件概述	21
4.2. 配置文件黑名单	22
4.2.1. 根据 WWID 将设备列入黑名单	22
4.2.2. 根据设备名称将设备列入黑名单	23
4.2.3. 根据设备类型将其加入黑名单	23
4.2.4. 黑名单之外的设备	23
4.3. 配置文件默认设置	24
4.4. 多路径设备配置属性	29
4.5. 配置文件设备	32
第 5 章 DM-Multipath 管理及故障排除	36
5.1. 重新定义在线多路径设备大小	36
5.2. 将 root 文件系统从单一路径设备移动到多路径设备中	36
5.3. 将 swap 文件系统从单一路径设备移动到多路径设备中	38
5.4. 多路径守护进程	39
5.5. 大量 LUN 造成的问题	39
5.6. 有 queue_if_no_path 功能的问题	39
5.7. 多路径命令输出	40
5.8. 使用多路径命令进行多路径查询	40

5.9. 多路径命令选项	41
5.10. 使用 dmsetup 命令确定设备映射器条目	41
5.11. 使用 multipathd 互动控制台进行故障排除	42
修订记录	43
索引	44
符号	44
A	46
B	46
C	46
D	46
F	47
G	47
H	47
I	47
K	47
L	48
M	48
N	48
P	48
Q	49
R	49
S	49
U	49
V	49
W	49

前言

本书论述了红帽企业版 Linux 6 发行本中红帽企业版 Linux 设备映射器多路径（DM-Multipath）的功能。

1. 读者

本书主要面向管理 Linux 操作系统的系统管理员使用，需要熟悉红帽企业版 Linux。

2. 相关文档

有关使用红帽企业版 Linux 的详情请参考以下资源：

- ▶ 《安装指南》 — 提供与安装红帽企业版 Linux 6 有关的信息。
- ▶ 《部署指南》 — 提供有关部署、配置和管理红帽企业版 Linux 6 的信息。
- ▶ 《存储管理指南》 — 提供有关有效管理红帽企业版 Linux 6 中的存储设备和文件系统的说明。

有关 Red Hat Enterprise Linux 6 Red Hat Cluster Suite 的信息请参考以下资源：

- ▶ 《红帽高可用性附加组件概述》 — 提供红帽集群套件的高级概述。
- ▶ 《集群管理》 — 提供关于安装、配置以及管理高可用性附加组件的信息。
- ▶ 《逻辑卷管理器管理》 — 提供逻辑卷管理器的描述，包括在集群的环境中运行 LVM 的信息。
- ▶ 《全局文件系统 2：配置及管理》 — 提供有关安装、配置和维护红帽 GFS2（红帽全局文件系统 2）的信息。
- ▶ 《负载均衡管理》 — 提供使用负载均衡附加组件配置高性能系统和服务的信息，该组件是一组整合的软件元件，可为在一组真实服务器间平衡 IP 负载提供 Linux 虚拟服务器（LVS）。
- ▶ 《发行注记》 — 提供有关现有红帽产品发行注记的信息。

Red Hat Cluster Suite 文档及其他 Red Hat 文档在 Red Hat Enterprise Linux 文档 CD 中有 HTML、PDF 以及 RPM 版本，其在线地址为 <http://docs.redhat.com/docs/en-US/index.html>。

3. 我们需要您的反馈！

如果您在这本手册中发现任何印刷错误，或者您对本手册有改进意见，我们非常乐于倾听！请在 Bugzilla <http://bugzilla.redhat.com/> 中根据产品 **Red Hat Enterprise Linux 6** 和组件 **doc-DM_Multipath** 提交报告。当提交 bug 报告时，请确定包含了手册识别符：

```
rh-DM_Multipath(EN)-6 (2013-2-15T15:15)
```

如果您有针对文档文档的建议，请尽量具体描述。如果您发现任何错误，请附带章节号以及上下文，以便我们查找。

4. 文档约定

本手册使用几个约定来突出某些用词和短语以及信息的某些片段。

在 PDF 版本以及纸版中，本手册使用在 [Liberation 字体](#) 套件中选出的字体。如果您在您的系统中安装了 Liberation 字体套件，它还可用于 HTML 版本。如果没有安装，则会显示可替换的类似字体。请注意：红帽企业 Linux 5 以及其后的版本默认包含 Liberation 字体套件。

4.1. 排版约定

我们使用四种排版约定突出特定用词和短语。这些约定及其使用环境如下。

单行粗体

用来突出系统输入，其中包括 shell 命令、文件名以及路径。还可用来突出按键以及组合键。例如：

要看到文件您当前工作目录中文件 **my_next_bestselling_novel** 的内容，请在 shell 提示符后输入 **cat my_next_bestselling_novel** 命令并按 **Enter** 键执行该命令。

以上内容包括一个文件名，一个 shell 命令以及一个按键，它们都以固定粗体形式出现，且全部与上下文有所区别。

按键组合与单独按键之间的区别是按键组合是使用加号将各个按键连在一起。例如：

按 **Enter** 执行该命令。

按 **Ctrl+Alt+F2** 切换到虚拟终端。

第一个示例突出的是要按的特定按键。第二个示例突出了按键组合：一组要同时按下的三个按键。

如果讨论的是源码、等级名称、方法、功能、变量名称以及在段落中提到的返回的数值，那么都会以上述形式出现，即**固定粗体**。例如：

与文件相关的等级包括用于文件系统的 **filesystem**、用于文件的 **file** 以及用于目录的 **dir**。每个等级都有其自身相关的权限。

比例粗体

这是指在系统中遇到的文字或者短语，其中包括应用程序名称、对话框文本、标记的按钮、复选框以及单选按钮标签、菜单标题以及子菜单标题。例如：

在主菜单条中选择「系统」→「首选项」→「鼠标」启动 **鼠标首选项**。在「按钮」标签中点击「**惯用左手鼠标**」复选框并点击 **关闭** 切换到主鼠标按钮从左向右（让鼠标适合左手使用）。

要在 **gedit** 文件中插入特殊字符，请在主菜单栏中选择「应用程序」→「附件」→「字符映射表」。接下来选择从 **Character Map** 菜单中选择 **Search** →「查找.....」，在「搜索」字段输入字符名称并点击「下一个」按钮。此时会在「字符映射表」中突出您搜索的字符。双击突出的字符将其放在「**要复制的文本**」字段中，然后点击「**复制**」按钮。现在返回您的文档，并选择 **gedit** 菜单中的「编辑」→「粘贴」。

以上文本包括应用程序名称、系统范围菜单名称及项目、应用程序特定菜单名称以及按钮和 GUI 界面中的文本，所有都以比例粗体出现并与上下文区别。

固定粗斜体 或者 **比例粗斜体**

无论固定粗体或者比例粗体，附加的斜体表示是可替换或者变量文本。斜体表示那些不直接输入的文本或者那些根据环境改变的文本。例如：

要使用 ssh 连接到远程机器，请在 shell 提示符后输入 **ssh *username@domain.name***。如果远程机器是 **example.com** 且您在该其机器中的用户名为 john，请输入 **ssh john@example.com**。

mount -o remount *file-system* 命令会重新挂载命名的文件系统。例如：要重新挂载 **/home** 文件系统，则命令为 **mount -o remount /home**。

要查看目前安装的软件包版本，请使用 **rpm -q *package*** 命令。它会返回以下结果：***package-version-release***。

请注意上述使用黑斜体的文字 -- *username*、*domain.name*、*file-system*、*package*、*version* 和 *release*。每个字都是一个站位符，可用作您执行命令时输入的文本，也可作为该系统显示的文本。

不考虑工作中显示标题的标准用法，斜体表示第一次使用某个新且重要的用语。例如：

Publican 是一个 *DocBook* 发布系统。

4.2. 抬升式引用约定

终端输出和源代码列表要与周围文本明显分开。

将发送到终端的输出设定为 **Mono-spaced Roman** 并显示为：

```
books      Desktop  documentation  drafts  mss    photos  stuff  svn
books_tests Desktop1  downloads      images  notes  scripts svgs
```

源代码列表也设为 **Mono-spaced Roman**，但添加下面突出的语法：

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
    public static void main(String args[])
        throws Exception
    {
        InitialContext iniCtx = new InitialContext();
        Object          ref     = iniCtx.lookup("EchoBean");
        EchoHome        home    = (EchoHome) ref;
        Echo             echo    = home.create();

        System.out.println("Created Echo");

        System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
    }
}
```

4.3. 备注及警告

最后，我们使用三种视觉形式来突出那些可能被忽视的信息。



注意

备注是对手头任务的提示、捷径或者备选的解决方法。忽略提示不会造成负面后果，但您可能会错过一个更省事的诀窍。



重要

重要框中的内容是那些容易错过的事情：配置更改只可用于当前会话，或者在应用更新前要重启的服务。忽略‘重要’框中的内容不会造成数据丢失但可能会让您抓狂。



警告

警告是不应被忽略的。忽略警告信息很可能导致数据丢失。

第 1 章 设备映射器多路径

设备映射器多路径 (DM-Multipath) 可让您将服务器节点和存储阵列间的多个 I/O 路径配置为一个单一设备。这些 I/O 路径是可包含独立电缆、交换机以及控制器的物理 SAN 连接。多路径集合了 I/O 路径，并生成由这些整合路径组成的新设备。

本章提供了红帽企业版 Linux 6 初始发行本中不包括的 DM-Multipath 新功能概述。之后，本章还提供 DM-Multipath 及其组件的高级概述以及 DM-Multipath 设置概述。

1.1. 新的和更改的功能

本节列出红帽企业版 Linux 6 初始及后续版本中 DM-Multipath 新的和更改的功能。

1.1.1. 红帽企业版 Linux 6.0 中新的和更改的功能

红帽企业版 Linux 6.0 包括以下文档和功能更新及更改。

- 在红帽企业版 Linux 6 发行本中，更改了基本故障切换配置的初始 DM-Multipath 设置过程。您现在可以创建 DM-Multipath 配置文件并使用 `mpathconf` 配置工具启用 DM-Multipath，该工具还可载入 `device-mapper-multipath`，启动 `multipathd` 守护进程并将 `chkconfig` 设定为在重启时自动启动该守护进程。

有关新设置过程的详情请参考 [第 3.1 节“设置 DM-Multipath”](#)。有关 `mpathconf` 命令的详情请参考 `mpathconf(5)` man page。

- 红帽企业版 Linux 6 发行本提供设定 multipath 设备的新模式，与 `find_multipaths` 配置文件参数一同使用。在旧版红帽企业版 Linux 中，multipath 总是尝试为每个没有明确放入黑名单的路径创建 multipath 设备。但在红帽企业版 Linux 6 中，如果将 `find_multipaths` 配置参数设定为 `yes`，那么 multipath 将只在满足以下三个条件之一时创建设备：
 - 至少有两个没有列入黑名单的路径使用同一 WWID。
 - 用户使用 `multipath` 命令指定设备强制手动生成该设备。
 - 路径拥有与之前创建的多路径设备相同的 WWID（即使该多路径设备目前不存在）。如果您之前在设有设定 `find_multipaths` 参数的情况下创建了多路径设备，那么有关步骤请参考 [第 4.2 节“配置文件黑名单”](#)。

这个功能可让大多数用户自动选择正确的路径创建多路径设备，而无需编辑黑名单。

有关 `find_multipaths` 配置参数的详情请参考 [第 4.3 节“配置文件默认设置”](#)。

- 红帽企业版 Linux 6 发行本提供两个新的路径选择程序算法，它们可决定下一个 I/O 操作要使用的路径，即 `queue-length` 和 `service-time`。`queue-length` 算法查看该路径未完成的 I/O 数量以便决定下一个要使用的路径。`service-time` 算法查看未完成的 I/O 数量以及该路径的相关流量以便决定下一个要使用的路径。有关配置文件中的路径选择程序参数详情请参考 [第 4 章 DM-Multipath 配置文件](#)。
- 在红帽企业版 Linux 6 发行本中，优先权功能不再能调出程序，而是与路径检查程序功能一样动态共享对象。已使用 `prio` 参数替换 `prio_callout` 参数。有关支持的 `prio` 功能详情请参考 [第 4 章 DM-Multipath 配置文件](#)。
- 在红帽企业版 Linux 6 中，更改了 `multipath` 命令的输出格式。有关 `multipath` 命令输出的详情请参考 [第 5.7 节“多路径命令输出”](#)。
- 在红帽企业版 Linux 6 发行本中，多路径 `bindings` 文件的位置为 `/etc/multipath/bindings`。
- 红帽企业版 Linux 6 发行本在 `multipath.conf` 文件中提供三个新的默认参数：`checker_timeout`、`fast_io_fail_tmo` 和 `dev_loss_tmo`。有关参数详情请参考 [第 4 章 DM-Multipath 配置文件](#)。
- 当将多路径配置文件中的 `user_friendly_names` 选项设定为 `yes` 时，则多路径设备的名称格式为 `mpathn`。在红帽企业版 Linux 6 发行本中，`n` 是字母字符，因此多路径设备的名称就可能是 `mpatha` 或者 `mpathb`。在之前的发行本中，`n` 是整数。

1.1.2. 红帽企业版 Linux 6.1 中新的和更改的功能

红帽企业版 Linux 6.1 中包括以下文档和功能更新和更改。

- ▶ 该文档现在包含新的一章，[第 5.2 节“将 root 文件系统从单一路径设备移动到多路径设备中”](#)。
- ▶ 该文档现在包含新的一章，[第 5.3 节“将 swap 文件系统从单一路径设备移动到多路径设备中”](#)。

1.1.3. 红帽企业版 Linux 6.2 中新的和更改的功能

红帽企业版 Linux 6.2 包括以下文档和功能更新及更改。

- ▶ 红帽企业版 Linux 6.2 发行本提供新的 `multipath.conf` 参数 `rr_min_io_rq`，该参数位于 `multipath.conf` 文件的 `defaults`、`devices` 和 `multipaths` 部分。红帽企业版 Linux 6.2 不再使用 `rr_min_io` 参数。有关 `rr_min_io_rq` 参数的详情请参考 [第 4 章 DM-Multipath 配置文件](#)。
- ▶ 现在可以将 `dev_loss_tmo` 配置文件参数设定为无限大，这样就将实际 `sysfs` 变量设定为 2147483647 秒，或者 68 年。有关这个参数的详情请参考 [第 4 章 DM-Multipath 配置文件](#)。
- ▶ 已更新 [第 5.2 节“将 root 文件系统从单一路径设备移动到多路径设备中”](#) 描述的进程。

1.1.4. 红帽企业版 Linux 6.3 中新的和更改的功能

红帽企业版 Linux 6.3 包括以下文档和功能更新及更改。

- ▶ `queue_without_daemon` 配置文件参数默认值已改为 `no`。
- ▶ `max_fds` 配置文件参数默认值已改为 `no`。
- ▶ 现在可在 `multipath.conf` 配置文件的 `defaults`、`multipaths` 和 `devices` 部分配置 `user_friendly_names` 配置文件参数。
- ▶ `multipath.conf` 配置文件的 `defaults` 部分支持新的 `hwtable_regex_match` 参数。

有关配置文件参数详情请参考 [第 4 章 DM-Multipath 配置文件](#)。

1.1.5. 红帽企业版 Linux 6.4 中新的和更改的功能

红帽企业版 Linux 6.4 包括以下文档和功能更新及更改。

- ▶ `multipath.conf` 配置文件的 `defaults` 部分和 `devices` 部分支持新的 `retain_attached_hardware_handler` 参数和新的 `detect_prio` 参数。有关该配置详情请参考 [第 4 章 DM-Multipath 配置文件](#)。
- ▶ 这个文档现在包括新的一节 [第 3.4 节“在 initramfs 文件系统中设置启动 Multipath”](#)。

1.2. DM-Multipath 概述

可使用 DM-Multipath 提供：

- ▶ 冗余

DM-Multipath 可在主动/被动配置中提供出错冗余。在主动/被动配置中，只有一半的路径在每次 I/O 时都使用。如果 I/O 路径的任意元素（电缆、交换机或者控制器）出现故障，就会将 DM-Multipath 切换到备用路径。
- ▶ 改进的性能

可将 DM-Multipath 配置为主动/主动模式，其中将 I/O 以轮叫调度算法方式分布到所有路径中。在有些配置中，DM-Multipath 可在 I/O 路径中检测负载并动态重新平衡负载。

[图 1.1 “带一个 RAID 设备的主动/被动多路径配置”](#) 演示在服务器和 RAID 设备之间有两个 I/O 路径的主动/被动配置。这里服务器中有两个 HBA，两个 SAN 交换机以及两个 RAID 控制器。

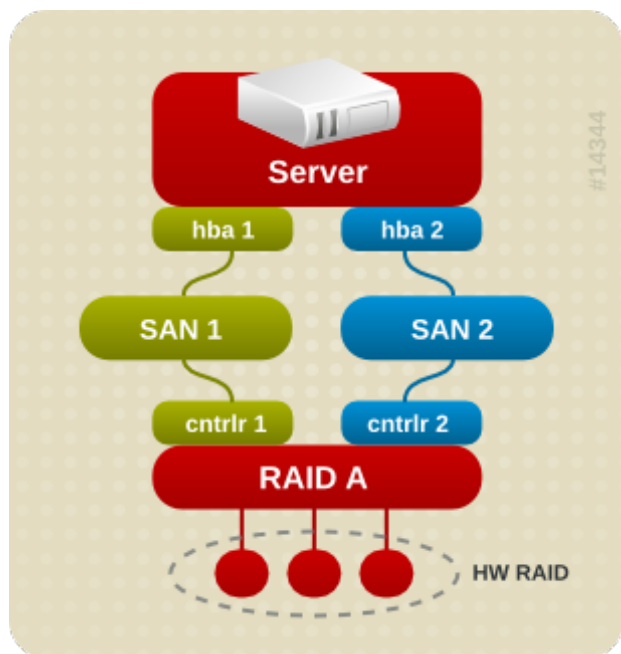


图 1.1. 带一个 RAID 设备的主动/被动多路径配置

在这个配置中，一个 I/O 路径通过 hba1、SAN1 以及控制器 1。另一个 I/O 路径通过 hba2、SAN2 以及控制器 2。在这个配置中有很多地方可能出现故障：

- ▶ HBA 故障
- ▶ FC 电缆故障
- ▶ SAN 交换机故障
- ▶ 阵列控制器端口故障

如果配置了 DM-Multipath，在所有这些地方出现故障时都会使 DM-Multipath 切换到备用 I/O 路径。

[图 1.2 “带两个 RAID 设备的主动/被动多路径配置”](#) 演示更复杂的主动/被动配置，其中服务器中有两个 HBA、两个 SAN 交换机以及两个 RAID 设备，每个带两个 RAID 控制器。

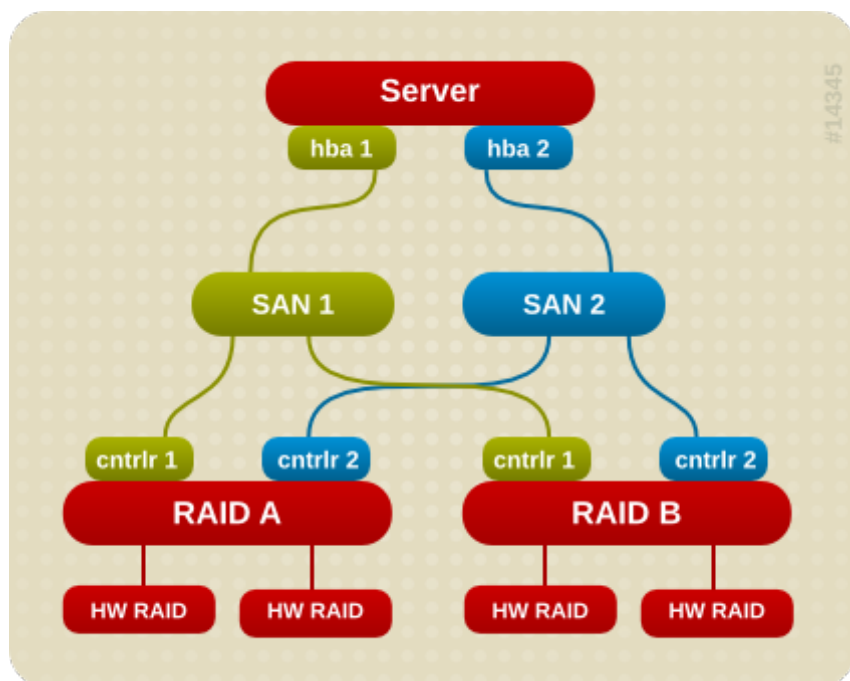


图 1.2. 带两个 RAID 设备的主动/被动多路径配置

在 [图 1.2 “带两个 RAID 设备的主动/被动多路径配置”](#) 演示的示例中，每个 RAID 设备有两个 I/O 路径（与 [图 1.1 “带一个 RAID 设备的主动/被动多路径配置”](#) 中显示的示例相同）配置了 DM-多路径，I/O 路径中任意点到任意 RAID 设备发生错误时，DM-多路径都会为那个设备切换到备用 I/O 路径。

[图 1.3 “带一个 RAID 设备的主动/主动多路径配置”](#) 演示在服务器中有两个 HBA、一个 SAN 交换机以及两个 RAID 控制器的主动/主动配置。在服务器和存储设备间有四条 I/O 路径：

- ▶ hba1 到控制器 1
- ▶ hba1 到控制器 2
- ▶ hba2 到控制器 1
- ▶ hba2 到控制器 2

在这个配置中，可将 I/O 分布到那四条路径中。



图 1.3. 带一个 RAID 设备的主动/主动多路径配置

1.3. 存储阵列支持

默认情况下，DM-Multipath 支持大多数常用的、支持 DM-Multipath 的存储阵列。您可在 `multipath.conf.defaults` 文件中找到这些支持的设备。如果您的存储阵列支持 DM-Multipath 且未在这个文件中默认设置，您可能需要将其添加到 DM-Multipath 配置文件 `multipath.conf` 中。有关 DM-Multipath 配置文件的详情请参考 [第 4 章 DM-Multipath 配置文件](#)。

有些存储阵列需要特殊的 I/O 错误和路径切换处理。这些都需要独立的硬件处理器内核模块。

1.4. DM-Multipath 组件

[表 1.1 “DM-Multipath 组件”](#) 描述了 DM-Multipath 组件。

表 1.1. DM-Multipath 组件

组件	描述
dm-multipath 内核模块	为路径和路径组群重新指定 I/O 并支持出错冗余。
mpathconf 程序	配置并启用设备映射器多路径
multipath 命令	列出并配置 multipath 设备。通常使用 <code>/etc/rc.sysinit</code> 启动，还可以在添加块设备时使用 udev 程序启动。
multipathd 守护进程	监视器路径，如果路径故障并返回，它可能会启动路径组群切换。可为多路径设备提供互动修改。对 <code>/etc/multipath.conf</code> 文件的任何修改都必须启动它。
kpartx 命令	为设备中的分区生成设备映射器设备。这个命令对带 DM-MP 的 DOS 分区是很必要的。 kpartx 在其自身软件包中就存在，但 device-mapper-multipath 软件包要依赖它。

1.5. DM-Multipath 设置概述

DM-Multipath 包含适用于常见多路径配置已编译的默认设置。安装 DM-multipath 通常很简单。

以下是为您的系统配置 DM-multipath 的基本步骤：

1. 安装 **device-mapper-multipath** rpm。
2. 使用 **mpathconf** 命令创建配置文件并启用多路径。如果您不需要编辑该配置文件，您还可以使用这个命令启动多路径守护进程。
3. 如需要，请编辑 **multipath.conf** 配置文件，修改默认值并保存更新的文件：
4. 启动多路径守护进程。

有关多路径配置的安装说明示例请参考 [第 3 章 设置 DM-Multipath](#)。

第 2 章 多路径设备

没有 DM-Multipath，从服务器节点到存储控制器的所有路径都将被系统视为独立的设备，即使连接相同服务器节点到相同存储控制器的 I/O 路径也不例外。DM-Multipath 提供了逻辑管理 I/O 路径的方法，即在基础设备的顶部生成单一多路径设备。

2.1. 多路径设备识别符

每个多路径设备都有一个全球识别符（WWID），它是一个全球唯一的无法更改的号码。默认情况下会将多路径设备的名称设定为它的 WWID。另外，您还可以在多路径配置文件中设置 `user_friendly_names` 选项，该选项可将别名设为格式为 `mpathn` 的节点唯一名称。

例如：当将有两个 HBA 的节点使用一个不分区 FC 切换附加到有两个端口的存储控制器中时，可看到四个设备：`/dev/sda`、`/dev/sdb`、`dev/sdc` 和 `/dev/sdd`。DM-Multipath 生成有唯一 WWID 的单一设备，该设备可将 I/O 根据多路径配置重新路由到那四个基础设备。当将 `user_friendly_names` 配置设定为 `yes`，会将该多路径设备的名称设定为 `mpathn`。

当在 DM-Multipath 中添加新设备时，这些新设备会位于 `/dev` 目录的两个不同位置：`/dev/mapper/mpathn` 和 `/dev/dm-n`。

- ▶ `/dev/mapper` 中的设备是在引导过程中生成的。可使用这些设备访问多路径设备，例如在生成逻辑卷时。
- ▶ 所有 `/dev/dm-n` 格式的设备都只能是作为内部使用，请不要使用它们。

有关多路径配置默认的详情，包括 `user_friendly_names` 配置选项，请参考 [第 4.3 节“配置文件默认设置”](#)。

您还可以使用多路径配置文件 `multipaths` 部分的 `alias` 选项将多路径设备名称设为您选择的名称。有关多路径配置文件 `multipaths` 部分的详情请参考 [第 4.4 节“多路径设备配置属性”](#)。

2.2. 在集群中保持多路径设备名称一致

当将 `user_friendly_names` 配置选项设为 `yes` 时，该多路径设备的名称对于节点来说是唯一的，但不保证对使用多路径设备的所有节点都一致。同样，如果您为 `multipath.conf` 配置文件的 `multipaths` 部分中的设备设定 `alias` 选项，该名称不会自动在集群的所有节点中保持一致。如果您使用 LVM 在多路径设备中创建逻辑设备，这不应是问题。但如果您需要将您的多路径设备名称在集群中的每个节点上都保持一致，请不要将 `user_friendly_names` 选项设定为 `yes`，且不要为那些设备配置别名。默认情况下，如果您不将 `user_friendly_names` 设定为 `yes`，或者为某个设备配置别名，则设备名称将是该设备的 WWID，它是不会变的。

如果您要系统定义的用户友好名称在集群的所有节点中都一致，您可按照以下步骤操作：

1. 在一台机器中设定所有多路径设备。
2. 运行以下命令在其他机器中禁用所有 `multipath` 设备：

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

3. 将第一台机器中的 `/etc/multipath/bindings` 文件复制到集群中的其它所有机器中。
4. 使用以下命令在集群的其他机器中重新 `multipathd` 守护进程：

```
# service mutipathd start
```

如果您添加新设备，您将需要重复这个过程。

同样，如果您为某个设备配置别名以便在集群的节点中使其保持一致，您应确定 `/etc/multipath.conf` 文件对于集群中的每个节点都是一样的，步骤如下：

1. 为机器 `multipath.conf` 文件中的多路基功能设备配置别名。
2. 运行以下命令在其他机器中禁用所有 multipath 设备：

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

3. 将第一台机器中的 `/etc/multipath.conf` 文件复制到集群中的其它所有机器中。
4. 使用以下命令在集群的其他机器中重新 `multipathd` 守护进程：

```
# service mutipathd start
```

当您添加新设备时，您将需要重复这个过程。

2.3. 多路径设备属性

除 `user_friendly_names` 和 `alias` 选项外，multipath 设备有大量属性。您可以为具体 multipath 设备修改这些属性，方法是在 multipath 配置文件的 `multipaths` 部分为那个设备创建一个条目。有关 multipath 配置文件 `multipaths` 部分的详情请参考 [第 4.4 节“多路径设备配置属性”](#)。

2.4. 逻辑卷中的多路径设备

生成多路径设备后，多路径设备的名称可以与您生成一个 LVM 物理卷时使用的物理设备名称相同。例如：如果多路径名称为 `/dev/mapper/mpatha`，以下命令可将 `/dev/mapper/mpatha` 标记为一个物理卷。

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

您可以如在使用其它 LVM 物理设备一样在创建 LVM 卷组时使用得到的 LVM 物理卷。

注意

如果您要在配置了分区的整个设备中创建 LVM 物理卷，`pvcreate` 命令将会失败。请注意：如果您不具体指定每个块设备，Anaconda 和 Kickstart 安装程序会生成空分区表。您可以使用 `kpartx -d` 和 `fdisk` 命令删除现有分区。如果您的系统有大于 2 TB 的块设备，您可使用 `parted` 命令删除分区。

当您创建一个使用主动/被动多路径阵列的 LVM 逻辑卷作为基础物理设备时，您应该在 `lvm.conf` 中添加过滤器，这样可以排除构成多路径设备的磁盘。这是因为如果阵列在接收 I/O 时自动将主动路径改为被动路径，不过滤掉这些设备，无论何时 LVM 扫描被动路径时，多路径将执行出错冗余并进行出错切换。对于需要使用命令激活被动路径的主动/被动阵列，LVM 会在发生这种情况时给出警告信息。

要过滤 LVM 配置文件 (`lvm.conf`) 中的所有 SCSI 设备，请在该文件的 `devices` 部分添加以下过滤器。

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

第 3 章 设置 DM-Multipath

本章提供了配置 DM-Multipath 的分步示例，其中包括以下步骤：

- ▶ 基本 DM-Multipath 设置
- ▶ 忽略本地磁盘
- ▶ 在配置文件中添加更多设备
- ▶ 在 `initramfs` 文件系统中启动 Multipath

3.1. 设置 DM-Multipath

在您的系统中安装 DM-Multipath 前，请确定您的系统已经进行了升级，其中包括对 `device-mapper-multipath` 软件包的升级。

您可使用 `mpathconf` 程序设置多路径，它可创建多路径配置文件 `/etc/multipath.conf`。

- ▶ 如果 `/etc/multipath.conf` 文件已存在，`mpathconf` 程序将会编辑该文件。
- ▶ 如果 `/etc/multipath.conf` 文件不存在，`mpathconf` 程序将使用 `/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf` 文件作为起始文件。
- ▶ 如果 `/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf` 文件不存在，则 `mpathconf` 程序将从头开始创建 `/etc/multipath.conf` 文件。

如果您不需要编辑 `/etc/multipath.conf` 文件，您可以运行以下命令为基本故障切换配置设定 DM-Multipath。这个命令可启用多路径配置文件并启动 `multipathd` 守护进程。

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

如果您需要在启动 `multipathd` 守护进程前编辑 `/etc/multipath.conf` 文件，请使用以下步骤为基本故障切换配置设置 DM-Multipath。

1. 运行带 `--enable` 选项的 `mpathconf` 命令：

```
# mpathconf --enable
```

有关您可能需要的 `mpathconf` 命令的其它选项，请参考 `mpathconf` man page，或者运行带 `--help` 选项的 `mpathconf` 命令。

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

Commands:
Enable: --enable
Disable: --disable
Set user_friendly_names (Default n): --user_friendly_names <y|n>
Set find_multipaths (Default n): --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>
chkconfig on/off multipathd (Default y): --with_chkconfig <y|n>
```

2. 必要时请编辑 `/etc/multipath.conf` 文件。DM-Multipath 的默认设置会编译到系统中，且无需在 `/etc/multipath.conf` 文件中明确指出。

`path_grouping_policy` 的默认值是设为 `failover`，因此在这个示例中您不需要编辑 `/etc/multipath.conf` 文件。有关在配置文件中将默认值改为其它数值的详情请参考 [第 4 章](#)

[DM-Multipath 配置文件。](#)

在您系统配置文件的初始默认部分配置您的系统以便多路径设备的名称格式为 `mpathn`。如果没有这个设置，多路径设备的名称将会是该设备 WWID 别名。

3. 需要时保存配置文件并退出编辑器。
4. 执行以下命令：

```
# service multipathd start
```

因为在配置文件中将 `user_friendly_name` 值设为 `yes`，那么生成的多路径设备将为 `/dev/mapper/mpathn`。有关将设备名称设定为您选择的别名的方法，请参考 [第 4 章 DM-Multipath 配置文件](#)。

如果您不想使用用户友好的名称，您可以运行以下命令：

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

注意

如果您在启动 multipath 守护进程后发现需要编辑 multipath 配置文件，则必须执行 `service multipathd reload` 命令方可使更改生效。

3.2. 在生成多路径设备时忽略逻辑磁盘

有些机器在其内部磁盘中有本地 SCSI 卡。不建议在这些设备中使用 DM-Multipath。如果您将 `find_multipaths` 配置参数设定为 `yes`，则您不一定要将这些设备列入黑名单。有关 `find_multipaths` 配置参数的详情请参考 [第 4.3 节“配置文件默认设置”](#)。

如果您没有将 `find_multipaths` 配置参数设定为 `yes`，则您可以使用以下步骤修改多路径配置文件以便在配置多路径时忽略本地磁盘。

1. 确定哪些磁盘是内部磁盘并将其列入黑名单。

在这个示例中，`/dev/sda` 是内部磁盘。请注意：因为在默认多路径配置文件中已经对其进行了配置，所以执行 `multipath -v2` 会在多路径映射中显示本地磁盘 `/dev/sda`。

有关 `multipath` 命令输出的详情请参考 [第 5.7 节“多路径命令输出”](#)。

```
# 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
```

2. 为防止设备映射器将 `/dev/sda` 与其多路径对应，请编辑 `/etc/multipath.conf` 文件的 `blacklist` 部分，使其包括该设备。虽然您可以使用 `devnode` 类型将 `sda` 设备列入黑名单，但那并不安全，因为重启时无法保证 `/dev/sda` 是同一个名称。要将单独的设备列入黑名单，您可以使用那个设备的 WWID 将其列入黑名单。

请注意：在 `multipath -v2` 命令的输出中，`/dev/sda` 设备的 WWID 是 `SIBM-ESXSST336732LC____F3ET0EP0Q000072428BX1`。要将这个设备列入黑名单，请在 `/etc/multipath.conf` 文件中添加以下内容。

```
blacklist {
    wwid SIBM-ESXSST336732LC____F3ET0EP0Q000072428BX1
}
```

3. 在您更新 `/etc/multipath.conf` 文件后，您必须手动让 `multipathd` 守护进程重新载入该文件。以下命令可重新载入更新的 `/etc/multipath.conf` 文件。

```
# service multipathd reload
```

4. 运行以下命令删除多路径设备：

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

5. 要查看是否删除了该设备，您可以运行 `multipath -ll` 命令显示当前的多路径配置。有关 `multipath -ll` 命令的详情请参考 [第 5.8 节“使用多路径命令进行多路径查询”](#)。

要确定被列入黑名单的设备没有被重新添加回来，您可以运行 `multipath` 命令，如下所示。如果您

没有指定 **-v** 选项，则 **multipath** 命令默认详细等级为 **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. 配置存储设备

默认情况下，DM-Multipath 包括对大多数支持 DM-Multipath 的通用存储阵列的支持。默认配置值请参考 **multipath.conf.defaults** 文件，其中包括支持的设备。

如果您需要添加一个在默认情况下不支持的存储设备作为已知多路径设备，请编辑 **/etc/multipath.conf** 文件并插入正确的设备信息。

例如：要添加有关 HP Open-V 系列的信息，其条目示例如下，其中 **%n** 为设备名称：

```
devices {
    device {
        vendor "HP"
        product "OPEN-V."
        getuid_callout "/lib/udev/scsi_id --whitelisted --device=/dev/%n"
    }
}
```

有关配置文件 **devices** 部分的详情请参考 [第 4.5 节“配置文件设备”](#)。

3.4. 在 **initramfs** 文件系统中设置启动 Multipath

您可以在 **initramfs** 文件系统中设定启动 multipath。配置 multipath 后，您可以使用该 multipath 配置文件重建 **initramfs** 文件系统，方法是执行带以下选项的 **dracut** 命令。

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

如果您在 **initramfs** 文件系统中运行 multipath，并更改 multipath 配置文件，则必须重建 **initramfs**

文件系统方可使更改生效。

第 4 章 DM-Multipath 配置文件

默认情况下，DM-Multipath 为大多数多路径的使用提供配置值。另外，DM-Multipath 支持大多数常见的、支持 DM-Multipath 的存储阵列。默认配置值和支持的设备请参考 `/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf.defaults` 文件。

您可以编辑 `/etc/multipath.conf` 覆盖 DM-Multipath 的默认配置。如果有必要，您还可以默认配置文件不支持的存储阵列。

注意

您可以在 `initramfs` 文件系统中运行 `set up multipathing`。如果您在 `initramfs` 文件系统中运行 `multipath`，并可以更改 `multipath` 配置文件，您必须重建 `initramfs` 文件系统以便使更改生效。有关使用 `multipath` 重建 `initramfs` 文件系统的下个详情请参考 [第 3.4 节 “在 initramfs 文件系统中设置启动 Multipath”](#)。

本章提供了解析和修改 `multipath.conf` 文件的详情。它包含以下小节：

- ▶ 配置文件概述
- ▶ 配置文件黑名单
- ▶ 配置文件默认
- ▶ 配置文件多路径
- ▶ 配置文件设备

在多路径配置文件中，您只需要指定需要的配置，或者您想要修改的在 `multipath.conf.defaults` 文件中指定的默认值。如果文件中有和您的环境不相关的部分或者您不需要更改默认值，您可以将其注释出来，因为它们在文件的起始部分。

配置文件使用正则表达式描述语法。

配置文件的注释版本请参考 `/usr/share/doc/device-mapper-multipathd-0.4.9/multipath.conf.annotated`。

4.1. 配置文件概述

多路径配置文件可分为以下几个部分：

blacklist

不被视为多路径的具体设备列表。

blacklist_exceptions

根据 `blacklist` 部分中的参数列出不在黑名单中的多路径设备。

defaults

DM-Multipath 的常规默认设置。

multipaths

各个独立多路径设备的特性设置。这些数值覆盖了在配置文件的 `defaults` 和 `devices` 部分中指定的数值。

devices

各个存储控制器的设置。这些数值覆盖了在配置文件的 **defaults** 部分指定的数值。如果您要使用不是默认支持的存储阵列，您可能需要为您的阵列创建 **devices** 子部分。

当系统决定多路径设备的属性时，首先它会检查多路径设置，然后是每个设备的设置，然后才是多路径系统默认设置。

4.2. 配置文件黑名单

多路径配置文件的 **blacklist** 部分指定在系统配置多路径设备时不能使用的设备。黑名单中的设备将无法分组到多路径设备中。

在旧版红帽企业版 Linux 中，multipath 总是尝试为每个没有明确放入黑名单的路径创建 multipath 设备。但在红帽企业版 Linux 6 中，如果将 **find_multipaths** 配置参数设定为 **yes**，那么 multipath 将只在满足以下三个条件之一时创建设备：

- ▶ 至少有两个没有列入黑名单的路径使用同一 WWID。
- ▶ 用户可使用 **multipath** 命令手动强制创建该设备。
- ▶ 有与之前创建的多路径设备相同 WWID 的路径（即使那个多路径设备目前不存在）。无论何时，创建多路径设备后，多路径会记住该设备的 WWID，以便在它看到有使用那个 WWID 的路径时即自动再次创建该设备。这可允许您让多路径自动选择正确的路径以便创建多路径设备而无需编辑多路径黑名单。

如果您之前创建了一个 multipath 设备而没有使用 **find_multipaths** 参数，然后您随后将该参数设定为 **yes**，您需要删除所有设备的 WWID，您不想从 **/etc/multipath/wwids** 文件中将其创建为 multipath 设备。以下显示示例 **/etc/multipath/wwids** 文件。WWID 由斜线 (/) 括起来：

```
# 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:
/3600d023000000000000e13955cc3757802/
/3600d023000000000000e13955cc3757801/
/3600d023000000000000e13955cc3757800/
/3600d02300069c9ce09d41c31f29d4c00/
/SWINSYS SF2372 0E13955CC3757802/
/3600d023000000000000e13955cc3757803/
```

如将 **find_multipaths** 参数设定为 **yes**，则您只需要将那些您不想使其具有多路径功能的多路径设备列入黑名单，通常不需要将设备列入黑名单。

如果您需要将设备列入黑名单，您可以根据以下条件进行选择：

- ▶ 根据 WWID，如 [第 4.2.1 节“根据 WWID 将设备列入黑名单”](#) 所述
- ▶ 根据设备名称，如 [第 4.2.2 节“根据设备名称将设备列入黑名单”](#) 所述
- ▶ 根据设备类型，如 [第 4.2.3 节“根据设备类型将其加入黑名单”](#) 所述

默认情况下，各种设备类型都是列在黑名单中的，即使您将配置文件的初始黑名单部分注释出来也是如此。详情请参考 [第 4.2.2 节“根据设备名称将设备列入黑名单”](#)。

4.2.1. 根据 WWID 将设备列入黑名单

您可根据全球识别符将单独的设备列入黑名单，即在配置文件的 **blacklist** 部分加入 **wwid** 条目。

以下示例显示了在配置文件中可将 WWID 为 26353900f02796769 的设备列入黑名单的行。

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

4.2.2. 根据设备名称将设备列入黑名单

您可以根据设备名称将设备类型列入黑名单，以便在配置文件 **blacklist** 部分的 **devnode** 条目中指定不要将它们分组到多路径设备中。

以下实例显示该配置文件中的可以将所有 SCSI 设备放入黑名单的行，因为它将所有 **sd*** 设备放入黑名单。

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

您可以使用配置文件 **blacklist** 部分的 **devnode** 条目指定要列入黑名单的每个设备，而不是指定具体类型的所有设备，我们不推荐使用后一种情况。除非根据 **udev** 规则的静态映射，我们无法保证在重启后每个设备的名称是一样的。例如：重启后，某个设备的名称可能从 **/dev/sda** 变为 **/dev/sdb**。

默认情况下，以下 **devnode** 条目是编译到默认黑名单中的。这些条目中加入黑名单中的设备一般不支持 DM-Multipath。要在这些设备中启用多路径，您可能需要在配置文件的 **blacklist_exceptions** 部分指定这些设备，如 [第 4.2.4 节“黑名单之外的设备”](#) 所述。

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

4.2.3. 根据设备类型将其加入黑名单

您可以在配置文件的 **blacklist** 部分与 **device** 一同指定具体设备类型。以下实例将所有 IBM DS4200 和 HP 设备放入黑名单。

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

4.2.4. 黑名单之外的设备

您可使用配置文件的 **blacklist_exceptions** 部分为被默认加入黑名单的设备启用多路径。

例如：如果您有大量设备，但只有一个需要多路径（WWID 为 3600d0230000000000e13955cc3757803），您不需要将您想要使用多路径的设备之外的每个设备单独加入黑名单，您只需要将所有设备都加入黑名单，然后在 **/etc/multipath.conf** 文件中添加以下行以便只允许您想要使用多路径的设备：

```
blacklist {
    wwid "*"
}

blacklist_exceptions {
    wwid "3600d023000000000000e13955cc3757803"
}
```

当在配置文件的 **blacklist_exceptions** 指定设备时，您必须以指定黑名单的相同方法来指定例外情况。例如：在 **devnode** 黑名单条目中指定的设备无法使用 WWID 将其指定为例外情况，即使列入黑名单的设备和该 WWID 关联也不行。同样，**devnode** 例外也只适用于 **devnode** 条目，而 **device** 例外只适用于 **device** 条目。

4.3. 配置文件默认设置

/etc/multipath.conf 配置文件包括 **defaults** 部分，在该部分中将 **user_friendly_names** 参数设为 **yes**，如下。

```
defaults {
    user_friendly_names yes
}
```

这可覆盖 **user_friendly_names** 参数的默认值。

该配置文件包括配置默认模板。这部分要被注释出来，如下。

```
#defaults {
#    udev_dir                /dev
#    polling_interval        5
#    path_selector            "round-robin 0"
#    path_grouping_policy    failover
#    getuid_callout          "/lib/dev/scsi_id --whitelisted --device=/dev/%n"
#    prio                    const
#    path_checker             directio
#    rr_min_io               1000
#    rr_weight               uniform
#    failback                manual
#    no_path_retry           fail
#    user_friendly_names     no
#}
```

要覆盖任意配置参数的默认值，您可将这个模板中相关的行复制到 **defaults** 部分并取消其注释。例如：要覆盖 **path_grouping_policy** 参数以使用 **multibus** 覆盖默认的 **failover**，请将模板中正确的行复制到配置文件的 **defaults** 部分并取消对它的注释，如下。

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

[表 4.1 “多路径配置默认设置”](#) 描述了在 **multipath.conf** 配置文件 **defaults** 部分设置的属性。这些值会被 DM-Multipath 使用，除非用 **multipath.conf** 配置文件的 **devices** 和 **multipaths** 部分指定的属性将其覆盖。



注意

从红帽企业版 Linux 6.0 开始弃用 **mode**、**uid** 和 **gid** 参数。device-mapper 设备的权限（包括 multipath 映射）由 **udev** 规则设定。**/usr/share/doc/device-mapper-version** 中有一个模板文件，名为 **12-dm-permissions.rules**，您可以使用它并将其放在 **/etc/udev/rules.d** 目录中使其生效。

表 4.1. 多路径配置默认设置

属性	描述
polling_interval	以秒为单位指定两次路径检查之间的间隔。对正常工作的路径，两次检查间的间隔会逐渐增加到 polling_interval 的四倍。默认值为 5。
udev_dir	生成 udev 设备节点的目录，默认值为 /dev 。
multipath_dir	保存动态共享对象的目录。默认值依系统而定，通常为 /lib/multipath 。
find_multipaths	<p>定义设定多路径设备的模式。如果将这个参数设定为 yes，那么多路径将不会尝试为每个不在黑名单中的路径创建设备。反之，多路径将只在满足以下三给条件之一时创建设备：</p> <ul style="list-style-type: none"> - 至少有两个没有列入黑名单的路径使用同一 WWID。 - 用户通过使用 multipath 命令指定设备强制创建该设备。 - 路径拥有与之前创建的多路径设备相同的 WWID。无论何时使用 find_multipaths 组件创建多路径设备后，多路径都会记住该设备的 WWID 以便在再次看到使用那个 WWID 的路径时自动创建设备。这可让您自动选择正确的所路径创建多路径设备而无需编辑多路径黑名单。如果之前您在没有设定 find_multipaths 参数的情况下创建了多路径设备，有关步骤请查看 第 4.2 节“配置文件黑名单”。 <p>默认值为 no。</p>
verbosity	默认详情。数值越高则详细程度越高。有效等级在 0 - 6 之间。默认值为 2。
path_selector	<p>指定用来决定下一个 I/O 操作所使用路径的默认算法。可能的值包括：</p> <p>round-robin 0：在路径组中循环每个路径，向每个路径发送同样数量的 I/O。</p> <p>queue-length 0：将下一组 I/O 发送到具有最少未处理 I/O 请求的路径。</p> <p>service-time 0：将下一组 I/O 发送到具有最短预计服务时间的路径，这是由未处理 I/O 的总量除以每个路径的相对流量决定的。</p> <p>默认值为 round-robin 0。</p>
path_grouping_policy	<p>指定用于未指定路径的默认路径分组策略，可能的值包括：</p> <p>failover：每个优先组群有一个路径。</p> <p>multibus：所有有效路径在一个优先组群中。</p> <p>group_by_serial：每个检测到的系列号有一个优先组群。</p> <p>group_by_prio：每个优先组群有一个路径优先值。优先权根据指定为 global、per-controller 或者 per-multipath 选项的调用程序决</p>

定。

group_by_node_name : 每个目标节点名有一个优先组。目标节点名保存在 `/sys/class/fc_transport/target*/node_name` 文件中。

默认值为 **failover**。

getuid_callout

指定默认程序和参数调用获得唯一路基功能识别程序。需要绝对路径。

默认值为 `/lib/udev/scsi_id --whitelisted --device=/dev/%n`。

prio

指定要获得路径优先值所需调用的默认程序及参数。例如：SPC-3 中的 ALUA 字节提供了一个可改变的 **prio** 值。可能的值包括：

const : 为所有路径设定优先权 1。

emc : 为 EMC 阵列生成路径优先权。

alua : 根据 SCSI-3 ALUA 设置生成路径优先权。

tpg_pref : 根据 SCSI-3 ALUA 设置生成路径优先权，使用首选端口字节。

ontap : 为 NetApp 阵列生成路径优先权。

rdac : 为 LSI/Engenio RDAC 控制程序生成路径优先权。

hp_sw : 为 Compaq/HP 控制程序在激活/等待模式中生成路径优先权。

hds : 为 Hitachi HDS Modular 存储阵列生成路径优先权。

默认值为 **const**。

features

多路径设备的默认额外功能。现有的唯一功能是 **queue_if_no_path**，其功能与将 **no_path_retry** 设为 **queue** 一致。有关使用这个功能可能出现问题的详情，请参考 [第 5.6 节“有 queue if no path 功能的问题”](#)。

path_checker

指定用于决定路径状态的默认方法，可能的值包括：

readsector0 : 读取该设备的第一扇区。

tur : 在该设备中执行 **TEST UNIT READY**。

emc_clariion : 查询 EMC Clariion 具体 EVPD 页面 0xCO 以便决定路径。

hp_sw : 为使用 Active/Standby 固件的 HP 存储阵列检查路径状态。

rdac : 为 LSI/Engenio RDAC 存储控制器检查路径状态。

	<p>directio : 使用直接 I/O 读取第一个扇区。</p> <p>默认值为 directio。</p>
failback	<p>管理路径组群出错切换。</p> <p>immediate 值指定立即恢复到包含活跃路径的最高级别路径组群。</p> <p>manual 值指定不需要立即恢复，只有在操作者干预的情况下会发生恢复。</p> <p>followover 值指定当路径组的第一个路径成为活跃路径时应执行自动恢复。这可让节点在另一个节点请求故障修复时不会自动恢复。</p> <p>大于 0 的数字值指定推迟出错切换，以秒表示。</p> <p>默认值为 manual。</p>
rr_min_io	<p>指定切换到当前路径组的下一个路径前路由到该路径的 I/O 请求数。这个设置值用于运行内核为 2.6.31 之前的系统。使用新版本的系统应使用 rr_min_io_rq。默认值为 1000。</p>
rr_min_io_rq	<p>使用 request-based device-mapper-multipath 指定切换到当前路径组的下一个路径前路由到该路径的 I/O 请求数。这个设置值用于运行当前内核的系统。在使用内核 2.6.31 版本之前的系统应使用 rr_min_io。默认值为 1。</p>
rr_weight	<p>如果将其设为 priorities，就不会在调用 selector 选择下一个路径前向路径发送 rr_min_io 请求，而是由 rr_min_io 乘以路径优先权决定发送的请求数，即由 prio 功能决定。如果将其设定为 uniform，则所有路径都有相同的加权。默认值为 uniform。</p>
no_path_retry	<p>这个属性的数字值指定系统在禁用队列前应该试图使用失败的路径的次数。</p> <p>fail 值说明立即失败，无需排队。</p> <p>queue 表示固定路径前排队不会停止。</p> <p>默认值为 0。</p>
user_friendly_names	<p>如果将其设为 yes，即指定该系统应该使用文件 /etc/multipath/bindings 为该多路径分配一个持久且唯一的别名，格式为 mpathn。如果设定为 no，即指定该系统应使用 WWID 作为该多路径的别名。在这两种情况下，您在这里指定的数值将被您在配置文件 multipaths 部分指定的具体设备别名覆盖。默认值为 no。</p>
queue_without_daemon	<p>如果将其设定为 no，multipathd 守护进程将在其关闭时禁用所有设备排队。默认值为 no。</p>
flush_on_last_del	<p>如果将其设定为 yes，multipathd 守护进程将在删除了最后一个到某个设备的路径时禁用查询。默认值为 no。</p>
max_fds	<p>设定 multipath 可以打开的文件提示符以及 multipathd 守护进程的最大值。这与 ulimit -n 命令效果一致。从红帽企业版 Linux 6.3 开始，默认值为 max，该值将该系统限制到 /proc/sys/fs/nr_open。对其较早的版本，如果没有设定这个</p>

	值，则使用调用进程作为打开文件提示符的最大值，通常为 1024。安全起见，如果该数值大于 1024，应将其设定为路径最大值+32。
checker_timeout	执行带明确超时设定的 SCSI 命令的路径检查程序的超时时间。默认值采用 <code>sys/block/sdx/device/timeout</code> 中指定的值。
fast_io_fail_tmo	在 FC 远程端口发现问题后，无法在那个远程端口设备中执行 I/O 前 SCSI 层要等待的时间。默认值应小于 <code>dev_loss_tmo</code> 值。将其设定为 <code>off</code> 则会禁用超时。默认值由该操作系统决定。
dev_loss_tmo	在 FC 远程端口发现问题后，到从该系统中删除它之前 SCSI 层要等待的时间。将其设定为无限，则会将其设定为 2147483647 秒，或者 68 年。默认值由该操作系统决定。
hwtable_regex_match	<p>(红帽企业版 Linux 6.3 发行本及之后的版本) 控制 multipath 如何使用内嵌设备配置整合该配置文件 <code>devices</code> 部分的设备配置。</p> <p><code>multipath.conf</code> 文件 <code>devices</code> 部分中的每个设备配置都可以创建其自己的设备配置，也可以修改内嵌设备配置。在红帽企业版 Linux 6.3 之前，如果某用户设备配置中的销售商、产品以及修订字符串与内嵌设备配置完全匹配，则该用户配置中的选项就可以修改内嵌配置。否则该用户的设备配置就将被视为一个新配置。</p> <p>如果将 <code>hwtable_regex_match</code> 设定为 <code>yes</code>，则使用正则表达式匹配。销售商、产品以及修订字符串都是正则表达式。该用户的这些选项的设备配置值与内嵌设备配置值匹配。这个匹配还可以同样的方式用于实际设备的销售商、产品以及修订字符串是否与设备配置字符串匹配以便决定为该设备使用哪个配置。如果用户设备配置匹配，那么就使用该用配置中的选项修改内嵌配置。否则该用户的设备配置就将被视为一个新配置。</p>
retain_attached_hw_handler	(红帽企业版 Linux 6.4 及之后的版本) 如果将这个参数设定为 <code>yes</code> ，且 <code>scsi</code> 层已为该路径设备附加可硬件句柄， <code>multipath</code> 将不会强制该设备使用 <code>mutipath.conf</code> 文件指定的 <code>hardware_handler</code> 。如果该 <code>scsi</code> 层没有附加硬件句柄， <code>multipath</code> 将继续使用其配置的硬件句柄。默认值为 <code>no</code> 。
detect_prio	(红帽企业版 Linux 6.4 及之后的版本) 如果将这个参数设定为 <code>yes</code> ， <code>multipath</code> 将首先检查该设备是否支持 ALUA，如果支持，则自动为该设备分配 <code>alua</code> 排序器。如果该设备不支持 ALUA，则按管用方法确定排序器。默认值为 <code>no</code> 。

4.4. 多路径设备配置属性

[表 4.2 “多路径属性”](#) 显示您可以在 `multipath.conf` 配置文件的 `multipaths` 部分为每个具体多路径设备设定的属性。这些属性只可用于指定的多路径。这些默认设置可由 DM-Multipath 使用并覆盖在 `multipath.conf` 文件的 `defaults` 和 `devices` 部分设定的属性。

表 4.2. 多路径属性

属性	描述
wwid	指定 multipath 属性采用的多路径设备的 WWID。这个参数在 multipath.conf 文件的这个部分是必须的。
alias	指定使用 multipath 属性的多路径设备的符号名称。如果您使用的是 user_friendly_names ，请必要将其设定为 mpathn ，这样可能会与自动分配的用户友好名称冲突，进而给出不正确的设备节点名称。
path_grouping_policy	指定用于未指定路径的默认路径分组策略，可能的值包括： failover = 每个优先组群有一个路径 multibus = 所有有效路径在一个优先组群中 group_by_serial = 每个检测到的系列号有一个优先组群 group_by_prio = 每个路径优先值有一个优先组群 group_by_node_name = 每个目标节点名有一个优先组群
path_selector	指定用来决定下一个 I/O 操作所使用路径的默认算法。可能的值包括： round-robin 0 : 在路径组中循环每个路径，向每个路径发送同样数量的 I/O。 queue-length 0 : 将下一组 I/O 发送到具有最少未处理 I/O 请求的路径。 service-time 0 : 将下一组 I/O 发送到具有最短预计服务时间的路径，这是由未处理 I/O 的总量除以每个路径的相对流量决定的。
failback	管理路径组群出错切换。 immediate 值指定立即恢复到包含活跃路径的最高级别路径组群。 manual 值指定不需要立即恢复，只有在操作者干预的情况下会发生恢复。 followover 值指定当路径组的第一个路径成为活跃路径时应执行自动恢复。这可让节点在另一个节点请求故障修复时不会自动恢复。 大于 0 的数字值指定推迟出错切换，以秒表示。
prio	指定要获得路径优先值所需调用的默认程序及参数。例如：SPC-3 中的 ALUA 字节提供了一个可改变的 prio 值。可能的值包括： const : 为所有路径设定优先权 1。 emc : 为 EMC 阵列生成路径优先权。

	<p>alua : 根据 SCSI-3 ALUA 设置生成路径优先权。</p> <p>tpg_pref : 根据 SCSI-3 ALUA 设置生成路径优先权, 使用首选端口字节。</p> <p>ontap : 为 NetApp 阵列生成路径优先权。</p> <p>rdac : 为 LSI/Engenio RDAC 控制程序生成路径优先权。</p> <p>hp_sw : 为 Compaq/HP 控制程序在激活/等待模式中生成路径优先权。</p> <p>hds : 为 Hitachi HDS Modular 存储阵列生成路径优先权。</p>
no_path_retry	<p>这个属性的数字值指定系统在禁用队列前应该试图使用失败的路径的次数。</p> <p>fail 值说明立即失败, 无需排队。</p> <p>queue 表示固定路径前排队不会停止。</p>
rr_min_io	<p>指定切换到当前路径组的下一个路径前路由到该路径的 I/O 请求数。这个设置值用于运行内核为 2.6.31 之前的系统。使用新版本的系统应使用 rr_min_io_rq。默认值为 1000。</p>
rr_min_io_rq	<p>使用 request-based device-mapper-multipath 指定切换到当前路径组的下一个路径前路由到该路径的 I/O 请求数。这个设置值用于运行当前内核的系统。在使用内核 2.6.31 版本之前的系统应使用 rr_min_io。默认值为 1。</p>
rr_weight	<p>如果将其设为 priorities, 就不会在调用 selector 选择下一个路径前向路径发送 rr_min_io 请求, 而是由 rr_min_io 乘以路径优先权决定发送的请求数, 即由 prio 功能决定。如果将其设定为 uniform, 则所有路径都有相同的加权。</p>
flush_on_last_del	<p>如果将其设定为 yes, multipathd 将在删除最后一个到某个设备的路径时禁用排队。</p>
user_friendly_names	<p>如果将其设为 yes, 即指定该系统应该使用文件 /etc/multipath/bindings 为该多路径分配一个持久且唯一的别名, 格式为 mpathn。如果设定为 no, 即指定该系统应使用 WWID 作为该多路径的别名。在这两种情况下, 您在这里指定的数值将被您在配置文件 multipaths 部分指定的具体设备别名覆盖。</p>

以下示例显示在配置文件中为两个特定多路径设备指定的多路径属性。第一个设备的 WWID 为 **3600508b4000156d70001200000b0000**, 符号名称为 **yellow**。

示例中第二个多路径设备的 WWID 为 **1DEC_____321816758474**, 符号名称为 **red**。在这个示例中, **rr_weight** 属性被设为 **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. 配置文件设备

[表 4.3 “设备属性”](#) 显示您可以为 `multipath.conf` 配置文件 `devices` 部分的每个独立存储设备设定的属性。这些属性可由 DM-Multipath 使用，除非使用 `multipath.conf` 文件 `multipaths` 部分为包含该设备的路径指定的属性覆盖这个值。这些属性覆盖了在 `multipath.conf` 文件 `defaults` 部分设定的属性。

默认情况下，在多路径配置中包括很多支持多路径的设备。这些设备默认支持值列在 `multipath.conf.defaults` 文件中。您可能不需要为这些设备修改默认值，但如果您想要修改默认值，您可以通过在想要更改默认值的设备的配置文件中添加一个条目来更改默认值。您可以为该设备复制 `multipath.conf.defaults` 文件中的设备配置默认值，并覆盖那些您想要更改的值。

要在配置文件的这个部分添加没有默认自动配置的设备，您需要设定 `vendor` 和 `product` 参数。您可在 `/sys/block/device_name/device/vendor` 和 `/sys/block/device_name/device/model` 文件中找到这些值，其中 `device_name` 是要进行多路径操作的设备，示例如下：

```

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

```

要指定的附加参数要视具体设备而定。如果设备是主动/主动模式，您通常不需要设置附加参数。您可能想要将 `path_grouping_policy` 设为 `multibus`。其它您可能需要设定的参数有 `no_path_retry` 和 `rr_min_io`，详见 [表 4.3 “设备属性”](#)。

如果设备是主动/被动模式，但它会自动将路径切换到被动路径，那么您需要将检查器功能改为不需向路径发送 I/O 来测试其是否工作（否则，您的设备将一直进行出错冗余）的功能。这几乎意味着您将 `path_checker` 设为 `tur`。这对所有支持 Test Unit Ready 命令的 SCSI 设备都适用。

如果该设备需要一个特殊的命令切换路径，那么为多路径配置此设备需要硬件处理器内核模块。当前的硬件处理器是 `emc`。如果这样还不能满足您的设备，则您可能无法为多路径配置该设备。

表 4.3. 设备属性

属性	描述
vendor	指定 device 采用的存储设备的零售商名称，例如 COMPAQ 。
product	指定 device 属性使用的存储设备产品名，比如 HSV110 (C) COMPAQ 。
revision	指定存储设备的产品修订识别程序。
product_blacklist	根据产品指定用来将设备列入黑名单的正则表达式。
hardware_handler	<p>指定将在切换路径组群或者处理 I/O 错误时用来执行硬件具体动作的模块。可能的值包括：</p> <ul style="list-style-type: none"> 1 emc : EMC 存储阵列的硬件处理程序。 1 alua : SCSI-3 ALUA 阵列的硬件处理程序。 1 hp_sw : Compaq/HP 控制器的硬件处理程序。 1 rdac : LSI/Engenio RDAC 控制器的硬件处理程序。
path_grouping_policy	<p>指定用于未指定路径的默认路径分组策略，可能的值包括：</p> <ul style="list-style-type: none"> failover = 每个优先组群有一个路径 multibus = 所有有效路径在一个优先组群中 group_by_serial = 每个检测到的系列号有一个优先组群 group_by_prio = 每个路径优先值有一个优先组群 group_by_node_name = 每个目标节点名有一个优先组群
getuid_callout	指定默认程序和参数调用获得唯一路基功能识别程序。需要绝对路径。
path_selector	<p>指定用来决定下一个 I/O 操作所使用路径的默认算法。可能的值包括：</p> <ul style="list-style-type: none"> round-robin 0 : 在路径组中循环每个路径，向每个路径发送同样数量的 I/O。 queue-length 0 : 将下一组 I/O 发送到具有最少未处理 I/O 请求的路径。 service-time 0 : 将下一组 I/O 发送到具有最短预计服务时间的路径，这是由未处理 I/O 的总量除以每个路径的相对流量决定的。
path_checker	<p>指定用于决定路径状态的默认方法，可能的值包括：</p> <ul style="list-style-type: none"> readsector0 : 读取该设备的第一扇区。 tur : 在该设备中执行 TEST UNIT READY。 emc_clariion : 查询 EMC Clariion 具体 EVPD 页面 0xCO 以便决

	<p>定路径。</p> <p>hp_sw : 为使用 Active/Standby 固件的 HP 存储阵列检查路径状态。</p> <p>rdac : 为 LSI/Engenio RDAC 存储控制器检查路径状态。</p> <p>directio : 使用直接 I/O 读取第一个扇区。</p>
features	<p>多路径设备的额外功能。现有的唯一功能是 queue_if_no_path, 它与将 no_path_retry 设为 queue 一致。有关使用这个功能可能出现问题的详情, 请参考 第 5.6 节“有 queue_if_no_path 功能的问题”。</p>
prio	<p>指定要获得路径优先值所需调用的默认程序及参数。例如: SPC-3 中的 ALUA 字节提供了一个可改变的 prio 值。可能的值包括:</p> <p>const : 为所有路径设定优先权 1。</p> <p>emc : 为 EMC 阵列生成路径优先权。</p> <p>alua : 根据 SCSI-3 ALUA 设置生成路径优先权。</p> <p>tpg_pref : 根据 SCSI-3 ALUA 设置生成路径优先权, 使用首选端口字节。</p> <p>ontap : 为 NetApp 阵列生成路径优先权。</p> <p>rdac : 为 LSI/Engenio RDAC 控制程序生成路径优先权。</p> <p>hp_sw : 为 Compaq/HP 控制程序在激活/等待模式中生成路径优先权。</p> <p>hds : 为 Hitachi HDS Modular 存储阵列生成路径优先权。</p>
failback	<p>管理路径组群出错切换。</p> <p>immediate 值指定立即恢复到包含活跃路径的最高级别路径组群。</p> <p>manual 值指定不需要立即恢复, 只有在操作者干预的情况下会发生恢复。</p> <p>followover 值指定当路径组的第一个路径成为活跃路径时应执行自动恢复。这可让节点在另一个节点请求故障修复时不会自动恢复。</p> <p>大于 0 的数字值指定推迟出错切换, 以秒表示。</p>
rr_weight	<p>如果将其设为 priorities, 就不会在调用 selector 选择下一个路径前向路径发送 rr_min_io 请求, 而是由 rr_min_io 乘以路径优先权决定发送的请求数, 即由 prio 功能决定。如果将其设定为 uniform, 则所有路径都有相同的加权。</p>
no_path_retry	<p>这个属性的数字值指定系统在禁用队列前应该试图使用失败的路径的次数。</p> <p>fail 值说明立即失败, 无需排队。</p>

	queue 表示固定路径前排队不会停止。
rr_min_io	指定切换到当前路径组的下一个路径前路由到该路径的 I/O 请求数。这个设置值用于运行内核为 2.6.31 之前的系统。使用新版本的系统应使用 rr_min_io_rq 。默认值为 1000。
rr_min_io_rq	使用 request-based device-mapper-multipath 指定切换到当前路径组的下一个路径前路由到该路径的 I/O 请求数。这个设置值用于运行当前内核的系统。在使用内核 2.6.31 版本之前的系统应使用 rr_min_io 。默认值为 1。
fast_io_fail_tmo	在 FC 远程端口发现问题后，无法在那个远程端口设备中执行 I/O 前 SCSI 层要等待的时间。默认值应小于 dev_loss_tmo 值。将其设定为 off 则会禁用超时。
dev_loss_tmo	在 FC 远程端口发现问题后，到从该系统中删除它之前 SCSI 层要等待的时间。将其设定为无限，则会将其设定为 2147483647 秒，或者 68 年。
flush_on_last_del	如果将其设定为 yes ， multipathd 守护进程将在删除了最后一个到某个设备的路径时禁用排队。
user_friendly_names	如果将其设为 yes ，即指定该系统应该使用文件 /etc/multipath/bindings 为该多路径分配一个持久且唯一的别名，格式为 mpathn 。如果设定为 no ，即指定该系统应使用 WWID 作为该多路径的别名。在这两种情况下，您在这里指定的数值将被您在配置文件 multipaths 部分指定的具体设备别名覆盖。默认值为 no 。
retain_attached_hw_handler	（红帽企业版 Linux 6.4 及之后的版本）如果将这个参数设定为 yes ，且 scsi 层已为该路径设备附加可硬件句柄， multipath 将不会强制该设备使用 mutipath.conf 文件指定的 hardware_handler 。如果该 scsi 层没有附加硬件句柄， multipath 将继续使用其配置的硬件句柄。
detect_prio	（红帽企业版 Linux 6.4 及之后的版本）如果将这个参数设定为 yes ， multipath 将首先检查该设备是否支持 ALUA，如果支持，则自动为该设备分配 alua 排序器。如果该设备不支持 ALUA，则按管用方法确定排序器。

以下示例显示了多路径配置文件的 **device** 条目。

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

第 5 章 DM-Multipath 管理及故障排除

本章提供了在运行的系统中管理 DM-Multipath 的信息，其中包括以下部分：

- ▶ 重新定义在线多路径设备大小
- ▶ 将 root 设备动单一路径设备移动到多路径设备中
- ▶ 将 swap 设备动单一路径设备移动到多路径设备中。
- ▶ 多路径守护进程
- ▶ 大量 LUN 造成的问题
- ▶ 有 queue_if_no_path 功能的问题
- ▶ multipath 命令输出
- ▶ 使用 multipath 命令进行多路径查询
- ▶ multipath 命令选项
- ▶ 用 dmsetup 命令进行多路径查询
- ▶ 使用 multipathd 互动控制台进行故障排除

5.1. 重新定义在线多路径设备大小

如果您要重新定义在线多路径设备，请按以下步骤操作。

1. 重新定义您的物理设备大小。
2. 使用以下命令查找 LUN 的路径：

```
# multipath -l
```

3. 重新定义您的路径大小。对于 SCSI 设备，在 rescan 文件中写入 1 以便让 SCSI 驱动器重新扫描，如以下命令：

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

4. 运行重新定义多路径大小的命令重新定义您的多路径设备大小：

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

5. 重新定义文件系统大小（假设没有使用 LVM 或者 DOS 分区）：

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

5.2. 将 root 文件系统从单一路径设备移动到多路径设备中

如果您在单一路径设备中安装您的系统，并后来在 root 文件系统中添加了另一个路径，则需要将您的 root 文件系统移动到多路径设备中。本小节记录了从单一路径移动到多路径设备的过程。

安装 device-mapper-multipath 软件包后执行以下步骤：

1. 执行以下命令创建 /etc/multipath.conf 配置文件，载入多路径模块并为 multipathd 将 chkconfig 设定为 on：

```
# mpathconf --enable
```

有关 mpathconf 命令输出的详情请参考 [第 3.1 节“设置 DM-Multipath”](#)。

2. 编辑 `/etc/multipath.conf` 文件的 `blacklist` 和 `blacklist_exceptions` 部分，如第 4.2 节“配置文件黑名单”所述。
3. 要确定您正确设定了配置文件，您可以运行 `/sbin/multipath` 命令，使用 `-v3` 选项查看该多路径守护进程是否尝试为您的 `root` 设备创建多路径设备。该命令会失败，因为 `root` 设备正在使用中，但该命令的输出应该显示该 `root` 设备所在路径列表。

您应该可在命令输出中看到有以下格式的行：

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

例如：如果在 `sda` 或者它的一个分区中设定您的 `root` 文件系统，您可在输出中看到类似如下的行：

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

```

在之后的输出结果中，您应该可以看到分配给多路径设备的 `root` 设备：

```
time | devname: ownership set to mpathdev
```

例如：输出结果显示如下：

```
Jun 14 06:48:21 | sda: ownership set to mpatha
```

您将看到一条说明告诉您无法使用该命令的以下格式创建多路径设备：

```
time | mpathdev: domap (0) failure for create/reload map
```

如上面的示例所示，您可以在命令输出中看到以下行：

```
Jun 14 06:48:21 | mpatha: domap (0) failure for create/reload map
```

4. 要使用 `multipath` 重建 `initramfs` 文件系统，请使用附带以下选项的 `dracut` 命令：

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

5. 如果您的 `root` 设备不是 LVM 卷，且使用设备名称挂载，则需要边界 `fstab` 文件切换为适当的多路径设备名称。如果您的 `root` 设备是 LVM 设备或者使用 UUID 或者其它属性挂载，就不需要这个步骤。
 - a. 使用第三步中描述的步骤运行代 `-v3` 选项 `/sbin/multipath` 命令确定 `root` 设备的 WWID。
 - b. 在 `/etc/multipath.conf` 文件中为 `root` 设备设定别名。

```

multipaths {
    multipath {
        wwid WWID_of_root_device
        alias rootdev
    }
}

```

- c. 编辑 `/etc/fstab` 并使用带这个多路径设备的 `root` 设备替换旧的设备路径。

例如：如果您在 `/etc/fstab` 文件中有以下条目：


```
/dev/sda1 / ext4 defaults 1 1
```

您可将该条目更改如下：

```
/dev/mapper/rootdev / ext4 defaults 1 1
```

如果您要编辑 `/etc/fstab` 文件，您还需要编辑 `/etc/grub/grub.conf` 文件，并将 `root` 参数从 `root=/dev/sda1` 改为 `root=/dev/mapper/rootdev`。

您编辑 `grub.conf` 文件前条目应类似如下。

```
title Red Hat Enterprise Linux FoundationServer (2.6.32-71.24.1.el6.x86_64)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-71.24.1.el6.x86_64 ro root=/dev/sda1
rd_NO_LUKS
rd_NO_MD rd_NO_DM LANG=en_US.UTF-8 SYSFONT=latarcyrheb-sun16 KEYTABLE=us
console=ttyS0,115200n8 crashkernel=auto
    initrd /initramfs-2.6.32-71.24.1.el6.x86_64.img
```

您编辑 `grub.conf` 文件后条目应类似如下。

```
title Red Hat Enterprise Linux FoundationServer (2.6.32-71.24.1.el6.x86_64)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-71.24.1.el6.x86_64 ro
root=/dev/mapper/rootdev
rd_NO_LUKS rd_NO_MD rd_NO_DM LANG=en_US.UTF-8 SYSFONT=latarcyrheb-sun16
KEYTABLE=us console=ttyS0,115200n8 crashkernel=auto
    initrd /initramfs-2.6.32-71.24.1.el6.x86_64.img
```

6. 关闭机器。
7. 配置 FC 开关以便可在该机器中看到其它路径。
8. 引导机器。
9. 查看 `root` 文件系统（`/`）是否是多路径设备中。

5.3. 将 swap 文件系统从单一路径设备移动到多路径设备中

默认情况下将 `swap` 设备设定为逻辑卷。只要您在由该逻辑卷组组成的物理卷中配置多路径。就不要求在将其配置为多路径设备时有任何特殊配置。如果您的 `swap` 设备不是 LVM 卷，且使用设备名称挂载该卷，您就需要编辑 `fstab` 文件将其转换为适当的多路径设备名称。

1. 运行 `/sbin/multipath` 命令使用 `-v3` 选项确定 `swap` 设备的 WWID 号。该命令的输出应该在路径列表中显示该 `swap` 设备。

您可以在该命令输出中看到有以下格式的行，它显示 `swap` 设备：

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

例如：如果在 `sda` 或者它的分区中设置 `swap` 文件系统，您会在输出中看到类似如下的行：

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

```

2. 在 `/etc/multipath.conf` 文件中为 `swap` 设备配置别名。

```

multipaths {
  multipath {
    wwid WWID_of_swap_device
    alias swapdev
  }
}

```

3. 编辑 `/etc/fstab` 并使用带这个多路径设备的 root 设备替换旧的设备路径。

例如：如果您在 `/etc/fstab` 文件中有以下条目：

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

您可将该条目更改如下：

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

5.4. 多路径守护进程

如果您在进行多路径配置时遇到问题，您应该确定多路径守护进程正在运行，如 [第 3 章 设置 DM-Multipath](#) 所述。必须运行 `multipathd` 守护进程以便使用多路径设备。

5.5. 大量 LUN 造成的问题

当在某个节点中添加大量 LUN 时，使用多路径设备可明显延长 `udev` 设备过滤器为其生成设备节点所用的时间。如果您遇到过这个问题，您可在 `/etc/udev/rules.d/40-multipath.rules` 文件中删除以下行解决这个问题：

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

这行会在每次向该节点中添加块设备时让 `udev` 设备管理器运行 `multipath`。虽然删除了这一行，`multipathd` 守护进程仍将自动生成多路径设备，在引导过程中仍将为带多路径根文件系统的节点调用 `multipath`。唯一的改变就是没有运行 `multipathd` 守护进程不再自动生成多路径设备，对大多数多路径用户来说应该不是个问题。

5.6. 有 `queue_if_no_path` 功能的问题

如果在 `/etc/multipath.conf` 文件中指定 `features "1 queue_if_no_path"`，那么所有采用 I/O 操作的进程都将被挂起直到恢复了一个或者多个路径。要避免这种情况，请在 `/etc/multipath.conf` 文件中设定 `no_path_retry N` 参数（其中 `N` 是该系统应该重试某个路径的次数）。

当您设定 `no_path_retry` 参数时，还要从 `/etc/multipath.conf` 文件中删除 `features "1 queue_if_no_path"` 选项。但如果您使用默认编译为 `features "1 queue_if_no_path"` 选项的多路径设备（很多 SAN 设备都是如此），则您必须添加 `features "0"` 覆盖这个默认设置。您可以将 `/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf.defaults` 中您设备的现有 `devices` 部分复制到 `/etc/multipath.conf` 并进行编辑以便适应您的需要。

如果您需要使用 `features "1 queue_if_no_path"` 选项，且您遇到过这里提到的问题，请使用 `dmsetup` 命令为特定 LUN 编辑运行时策略（就是说对该特定 LUN 来说所有路径都不可用）。例如：如果您想要将多路径设备 `mpath2` 的策略从 `"queue_if_no_path"` 改为 `"fail_if_no_path"`，请执行以下命令。

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

请注意：您必须指定 `mpathn` 别名而不是该路径。

5.7. 多路径命令输出

当您生成、修改或者列出多路径设备时，您会获得一个当前设备组成的打印输出，格式如下。

对于每个多路径设备：

```
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
```

对与每个路径组群：

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

对于每个路径：

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

例如，多路径命令的输出可能是如下形式：

```
3600d023000000000000e13955cc3757800 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
```

如果该路径已经启用并准备好执行 I/O，那么路径的状态就是 **ready** 或者 **ghost**。如果该路径无法使用，则状态为 **faulty** 或者 **shaky**。路径的状态由 `multipathd` 守护进程根据在 `/etc/multipath.conf` 文件中定义的通信间隔进行周期性更新。

dm 状态和路径的状态相似，但从内核来看，dm 有两个状态：**failed**（类似 **faulty**）和 **active**（涵盖所有其它路径状态）。少数情况下，设备的路径状态和 dm 状态会暂时不同。

online_status 的可能值为 **running** 和 **offline**。**offline** 意味着这个 SCSI 设备已被禁用。

注意

当生成或者修改一个多路径设备时，路径组群状态、dm 设备名称、写入权限和 dm 状态是未知的。另外，功能也不一定正确。

5.8. 使用多路径命令进行多路径查询

您可以使用 `multipath` 命令的 `-l` 和 `-ll` 选项来显示现有多路径配置。`-l` 选项显示从 `sysfs` 以及设备映射器中收集的多路径拓补信息。`-ll` 选项显示 `-l` 显示的所有信息以及该系统额外可用的其它组件。

当显示多路径配置时，您可以使用 `multipath` 命令的 `-v` 选项指定三种详细等级。指定为 `-v0` 时没有输出。指定为 `-v1` 则只输出生成或者更新的路径名称。指定 `-v2` 将输出所有检测到的路径、多路径和设备映射。

以下示例显示了 `multipath -l` 命令的输出。

```
# multipath -l
3600d023000000000000e13955cc3757800 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
```

以下示例显示了 `multipath -ll` 命令的输出。

```
# multipath -ll
3600d023000000000000e13955cc3757801 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
3600d023000000000000e13955cc3757803 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
```

5.9. 多路径命令选项

[表 5.1 “有用的 `multipath` 命令选项”](#) 描述了一些您可能会用到的 `multipath` 命令选项。

表 5.1. 有用的 `multipath` 命令选项

选项	描述
<code>-l</code>	显示来自 <code>sysfs</code> 和设备映射器的当前多路径配置。
<code>-ll</code>	显示来自 <code>sysfs</code> 、设备映射器以及系统中的其它可用组件的当前多路径配置。
<code>-f device</code>	删除命名的多路径设备。
<code>-F</code>	删除所有不使用的多路径设备。

5.10. 使用 `dmsetup` 命令确定设备映射器条目

您可以使用 `dmsetup` 找出哪个设备映射器条目与多路径的设备映射。

以下命令显示所有设备映射器设备及其主、副号码。副号码确定 `dm` 设备的名称。例如：副号码 3 与多路径的设备 `/dev/dm-3` 对映。

```
# 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)
```

5.11. 使用 multipathd 互动控制台进行故障排除

multipathd -k 命令是一个 **multipathd** 守护进程的互动界面。执行此命令将进入互动的多路径控制台。执行此命令后，您可以输入 **help** 来获得可用的命令列表。您可以输入互动命令，或者您还可以输入 **CTRL-D** 退出。

multipathd 互动控制台可用于在您的系统有问题时进行故障排除。例如：以下命令会在退出控制台前显示多路径配置，其中包括默认配置。

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

以下命令确定多路径已经识别了所有对 **multipath.conf** 的修改。

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

使用以下命令组确定路径检查器工作正常。

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

修订记录

修订 5.0-9.2 完成翻译、校对	Mon May 6 2013	Leah Liu
修订 5.0-9.1 与 XML 源 5.0-9 版本同步的翻译文件	Wed Apr 17 2013	Chester Cheng
修订 5.0-9 6.4 GA 发行本	Mon Feb 18 2013	Steven Levine
修订 5.0-7 6.4 Beta 发行本	Mon Nov 26 2012	Steven Levine
修订 5.0-6 解决：#856829 编写新的 <code>retain_attached_hardware_handler</code> 和 <code>detect_prio</code> 配置参数。 解决：#815925 编写在 <code>initramfs</code> 文件系统中设定启动 <code>multipath</code> 的步骤。 解决：#836333 修改一些小的排版错误。 解决：#856826 更新 <code>multipath.conf</code> 摘要示例以反映更新的 <code>path_selector</code> 参数。	Tue Nov 13 2012	Steven Levine
修订 4.0-3 发布版本 6.3 GA	Fri Jun 15 2012	Steven Levine
修订 4.0-2 解决：#815925 更新 <code>dmsetup ls</code> 输出格式示例。	Wed Apr 25 2012	Steven Levine
修订 4.0-1 解决：#786399 修改一些小的排版错误。 解决：#771420 编写红帽企业版 Linux 6.3 在中新的 DM-Multipath 功能。	Wed Mar 28 2012	Steven Levine
修订 3.0-3 红帽企业版 Linux 6.2 的 GA 发行版本 解决：#753899 修正 <code>multipath -F</code> 选项描述。	Thu Dec 1 2011	Steven Levine
修订 3.0-2 解决：#743767	Fri Oct 7 2011	Steven Levine

修改一些小的排版错误，并明确一些技术疑点。

修订 3.0-1 **Mon Sep 19 2011** **Steven Levine**

红帽企业版 Linux 6.2 Beta 发行本初始修订

解决：#707638

编写红帽企业版 Linux 6.2 在中新的 DM-Multipath 功能。

解决：#715457

修改 SCSI 设备过滤器示例。

解决：#623450

更新将 root 设备移动到 multipathed 卷的步骤。

解决：#725374, #738051

改正小的排版错误。

修订 2.0-1 **Thu May 19 2011** **Steven Levine**

红帽企业版 Linux 6.1 发行本初始修订

解决：#623450

添加将 root 和 swap 设备从单一路径移动到多路径设备的新过程。

解决：#693948

修改设备属性表格中的一些错误。

解决：#694683

改正排版错误。

解决：#702721

删除 /dev/mpath 中过期的参考。

修订 1.0-1 **Wed Nov 10 2010** **Steven Levine**

红帽企业版 Linux 6 的第一个版本

索引

符号

/etc/multipath.conf 软件包, [设置 DM-Multipath](#)

主动/主动配置

- 图解, [DM-Multipath 概述](#)
- 定义, [DM-Multipath 概述](#)

主动/被动配置

- 图解, [DM-Multipath 概述](#)
- 定义, [DM-Multipath 概述](#)

全球识别符 (WWID) , [多路径设备识别符](#)

功能, 新的和更改的, [新的和更改的功能](#)

反馈

- 本手册的联络信息, [我们需要您的反馈!](#)

多路径 root 文件系统, [将 root 文件系统从单一路径设备移动到多路径设备中](#)

多路径 swap 文件系统, [将 swap 文件系统从单一路径设备移动到多路径设备中](#)

多路径守护进程 (multipathd) , [多路径守护进程](#)

多路径设备, [多路径设备](#)

- LVM 物理卷, [逻辑卷中的多路径设备](#)
- 逻辑卷, [逻辑卷中的多路径设备](#)

存储阵列

- 添加, [配置存储设备](#), [配置文件设备](#)

存储阵列支持, [存储阵列支持](#)

故障切换, [DM-Multipath 概述](#)

本地磁盘, 忽略, [在生成多路径设备时忽略逻辑磁盘](#)

概述

- 功能, 新的和更改的, [新的和更改的功能](#)

设备

- 添加, [配置存储设备](#), [配置文件设备](#)

设备名称, [多路径设备识别符](#)

设置

- DM-Multipath, [设置 DM-Multipath](#)

配置

- DM-Multipath, [设置 DM-Multipath](#)

配置文件

- alias 参数, [多路径设备配置属性](#)
- checker_timeout 参数, [配置文件默认设置](#)
- detect_prio 参数, [配置文件默认设置](#), [多路径设备配置属性](#)
- dev_loss_tmo 参数, [配置文件默认设置](#), [配置文件设备](#)
- failback 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- fast_io_fail_tmo 参数, [配置文件默认设置](#), [配置文件设备](#)
- features 参数, [配置文件默认设置](#), [配置文件设备](#)
- flush_on_last_del 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- getuid_callout 参数, [配置文件默认设置](#), [配置文件设备](#)
- hardware_handler 参数, [配置文件设备](#)
- hwtable_regex_match 参数, [配置文件默认设置](#)

- max_fds 参数, [配置文件默认设置](#)
- no_path_retry 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- path_checker 参数, [配置文件默认设置](#), [配置文件设备](#)
- path_grouping_policy 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- path_selector 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- polling-interval 参数, [配置文件默认设置](#)
- prio 参数, [配置文件默认设置](#), [配置文件设备](#)
- product 参数, [配置文件设备](#)
- product_blacklist 参数, [配置文件设备](#)
- queue_without_daemon 参数, [配置文件默认设置](#)
- retain_attached_hw_handler 参数, [配置文件默认设置](#), [多路径设备配置属性](#)
- revision 参数, [配置文件设备](#)
- rr_min_io 参数, [配置文件默认设置](#), [多路径设备配置属性](#)
- rr_weight 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- udev_dir 参数, [配置文件默认设置](#)
- user_friendly_names 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)
- vendor 参数, [配置文件默认设置](#), [配置文件设备](#)
- wwid 参数, [多路径设备配置属性](#)
- 概述, [配置文件概述](#)
- 黑名单, [配置文件黑名单](#)

重新定义多路径设备大小, [重新定义在线多路径设备大小](#)

黑名单

- WWID, [根据 WWID 将设备列入黑名单](#)
- 设备名称, [根据设备名称将设备列入黑名单](#)
- 设备类型, [根据设备类型将其加入黑名单](#)
- 配置文件, [配置文件黑名单](#)
- 默认设备, [根据设备名称将设备列入黑名单](#)

A

alias 参数, [多路径设备配置属性](#)

- 配置文件, [多路径设备识别符](#)

B

blacklist_exceptions 部分

- multipath.conf 文件, [黑名单之外的设备](#)

C

checker_timeout 参数, [配置文件默认设置](#)

D

defaults 部分

- multipath.conf 文件, [配置文件默认设置](#)

detect_prio 参数, [配置文件默认设置](#), [多路径设备配置属性](#)

dev/mapper 目录, [多路径设备识别符](#)

device-mapper-multipath 软件包, [设置 DM-Multipath](#)

devices 部分

- multipath.conf 文件, [配置文件设备](#)

dev_loss_tmo 参数, [配置文件默认设置](#), [配置文件设备](#)

DM-Multipath

- 冗余, [DM-Multipath 概述](#)
- 和 LVM, [逻辑卷中的多路径设备](#)
- 定义, [设备映射器多路径](#)
- 故障切换, [DM-Multipath 概述](#)
- 概述, [DM-Multipath 概述](#)
- 组件, [DM-Multipath 组件](#)
- 设备, [多路径设备](#)
- 设备名称, [多路径设备识别符](#)
- 设置, [设置 DM-Multipath](#)
- 设置, 概述, [DM-Multipath 设置概述](#)
- 配置, [设置 DM-Multipath](#)
- 配置文件, [DM-Multipath 配置文件](#)

dm-multipath 内核模块, [DM-Multipath 组件](#)

dm-n 设备, [多路径设备识别符](#)

dmsetup 命令, 确定设备映射器条目, [使用 dmsetup 命令确定设备映射器条目](#)

F

failback 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

fast_io_fail_tmo 参数, [配置文件默认设置](#), [配置文件设备](#)

features 参数, [配置文件默认设置](#), [配置文件设备](#)

flush_on_last_del 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

G

getuid_callout 参数, [配置文件默认设置](#), [配置文件设备](#)

H

hardware_handler 参数, [配置文件设备](#)

hwtable_regex_match 参数, [配置文件默认设置](#)

I

initramfs

- 启动 multipath, [在 initramfs 文件系统中设置启动 Multipath](#)

K

kpartx 命令, [DM-Multipath 组件](#)

L

LVM 物理卷

- 多路径设备, [逻辑卷中的多路径设备](#)

lv.conf 文件, [逻辑卷中的多路径设备](#)

M

max_fds 参数, [配置文件默认设置](#)

mpathconf 命令, [DM-Multipath 组件](#)

multipath 命令, [DM-Multipath 组件](#)

- 查询, [使用多路径命令进行多路径查询](#)
- 输出结果, [多路径命令输出](#)
- 选项, [多路径命令选项](#)

multipath.conf 文件, [存储阵列支持](#), [DM-Multipath 配置文件](#)

- blacklist_exceptions 部分, [黑名单之外的设备](#)
- defaults 部分, [配置文件默认设置](#)
- devices 部分, [配置文件设备](#)
- multipaths 部分, [多路径设备配置属性](#)

multipath.conf.annotated 文件, [DM-Multipath 配置文件](#)

multipath.conf.defaults 文件, [存储阵列支持](#), [DM-Multipath 配置文件](#)

multipathd

- 互动控制台, [使用 multipathd 互动控制台进行故障排除](#)
- 命令, [使用 multipathd 互动控制台进行故障排除](#)

multipathd start 命令, [设置 DM-Multipath](#)

multipathd 守护进程, [DM-Multipath 组件](#)

multipaths 部分

- multipath.conf 文件, [多路径设备配置属性](#)

N

no_path_retry 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

P

path_checker 参数, [配置文件默认设置](#), [配置文件设备](#)

path_grouping_policy 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

path_selector 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

polling_interval 参数, [配置文件默认设置](#)

prio 参数, [配置文件默认设置](#), [配置文件设备](#)

product 参数, [配置文件设备](#)

product_blacklist 参数, [配置文件设备](#)

Q

queue_without_daemon 参数, [配置文件默认设置](#)

R

retain_attached_hw_handler 参数, [配置文件默认设置](#), [多路径设备配置属性](#)

revision 参数, [配置文件设备](#)

root 文件系统, [将 root 文件系统从单一路径设备移动到多路径设备中](#)

rr_min_io 参数, [配置文件默认设置](#), [多路径设备配置属性](#)

rr_weight 参数, [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

S

swap 文件系统, [将 swap 文件系统从单一路径设备移动到多路径设备中](#)

U

udev_dir 参数, [配置文件默认设置](#)

user_friendly_names 参数, [多路径设备识别符](#), [配置文件默认设置](#), [多路径设备配置属性](#), [配置文件设备](#)

V

vendor 参数, [配置文件设备](#)

verbosity 参数, [配置文件默认设置](#)

W

wwid 参数, [多路径设备配置属性](#)