SUSE® Linux Enterprise Server

11 SP1 www.novell.com

2010 年 3 月 15 日 | SLES 11 SP1 ：儲存管理指南

SLES 11 SP1 ：儲存管理指南

法律聲明

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

*線上文件：*若要存取本產品及其他 Novell 產品的最新線上文件，請參閱 Novell 文件網頁 [http://www.novell.com/documentation]。

Novell 商標

協力廠商資料

目錄

關於本指南

本指南提供關於如何管理 SUSE® Linux Enterprise Server 11 Support Pack 1 (SP1) 伺服器上之儲存設備的資訊。

使用對象

本指南的適用對象為系統管理員。

意見反應

我們希望得到您對本手冊以及本產品隨附之其他文件的意見和建議。請使用線上文件中每頁底下的「使用者意見」功能，或造訪 www.novell.com/documentation/feedback.html，然後寫下您的意見。

文件更新

請造訪 Novell® 的 SUSE Linux Enterprise Server 11 SP1 文件網站 [http://www.novell.com/documentation/sles11]，獲取最新版本的《*SUSE Linux Enterprise Server 11 SP1 儲存管理指南*》。

其他文件

如需分割和管理設備的詳細資訊，請參閱《*SUSE Linux Enterprise Server 11 SP1 Installation and Administration Guide*》*(SUSE Linux Enterprise Server 11 SP1 安裝與管理指南)* [http://www.novell.com/documentation/sles11]。

文件慣例

在 Novell 文件中，大於符號 (>) 是用來分隔步驟中的動作，以及交互參照路徑中的項目。

商標符號 (®、™ 等) 表示 Novell 的商標。星號 (**) 表示協力廠商的商標。

1 Linux 中檔案系統的綜覽

SUSE® Linux Enterprise Server 隨附有多種不同的檔案系統供選擇，包括 Ext3、Ext2、ReiserFS 和 XFS。每個檔案系統都有其各自的優點和缺點。為滿足高效能叢集案例要求，SUSE Linux Enterprise Server 在高可用性儲存基礎結構 (HASI) 版本中加入了 OCFS2 (Oracle Cluster File System 2)。

1.1 術語

中繼資料
: 資料結構屬於檔案系統的內部結構。它可確保磁碟上的所有資料組織正確並可進行存取。基本上，它是「關於資料的資料」。幾乎每一種檔案系統都有自己的中繼資料結構，這也是檔案系統展現出不同效能特性的原因所在。它對於維護中繼資料的完整極為重要，因為要不是如此，檔案系統上所有資料便無法存取。

inode
: 檔案系統的資料結構包含檔案的各種資訊，包括大小、連結數目、實際儲存檔案內容之磁碟區塊的指標、建立日期和時間、修改和存取權限。

日誌
: 在檔案系統的內容中，日誌是一種磁碟上的結構，包含了檔案系統用於儲存有關檔案系統之中繼資料變更資訊的一種記錄。記錄可大大降低檔案系統的復原時間，因為有了它就不需要在系統啟動時執行檢查整個檔案系統這一冗長的搜尋程序。而是只重複檢查日誌。

1.2 Linux 的主要檔案系統

SUSE Linux Enterprise Server 提供了各種檔案系統可供選擇。本節包含有關這些檔案系統的工作方式及其優點的綜覽。

請記住，沒有一種檔案系統能夠完美適合所有類型的應用程式，這點非常重要。每一種檔案系統都有自己特殊的優、缺點，必須考慮在內。此外，即使是最頂級的檔案系統，也無法取代合理的備份策略。

本節中使用的*資料完整性*和*資料一致性*這兩個詞彙，並不表示使用者空間資料 (應用程式寫入其檔案中的資料) 的一致性。這項資料是否一致必須由應用程式本身控制。

重要

除非在本節中特別指明，否則設定或變更分割區以及檔案系統所需的一切步驟，都可以使用 YaST 來執行。

- 第 1.2.1 節「Ext2」 [3頁]
- 第 1.2.2 節「Ext3」 [3頁]
- 第 1.2.3 節「Oracle Cluster File System 2」 [5頁]
- 第 1.2.4 節「ReiserFS」 [6頁]
- 第 1.2.5 節「XFS」 [7頁]

1.2.1 Ext2

Ext2 的起源要回到 Linux 歷史的古早年代。它的前輩 - 延伸檔案系統，是在 1992 年 4 月落實並整合至 Linux 0.96c。延伸檔案系統已經過多次修改，而到了 Ext2，成為多年來最受歡迎的 Linux 檔案系統。若建立了檔案系統的記錄，因其復原快速，Ext2 就顯得不再那麼重要了。

提供了簡短的 Ext2 功能摘要，可協助您瞭解為什麼它過去是 (目前在某些領域依然是) 很多 Linux 使用者非常喜愛的 Linux 檔案系統。

- 章節「穩固性和速度」[3頁]
- 章節「升級容易」[3頁]

穩固性和速度

Ext2 經過多次改良和密集測試，已經算是「老前輩」了。這可能是為什麼人們通常稱它堅如磐石的原因所在。在檔案系統無法完全取消掛接而導致系統中斷後，e2fsck 會開始分析檔案系統資料。中繼資料會進入一致性狀態，而待處理的檔案或資料區塊會寫入指定的目錄 (稱為 `lost+found`)。與日誌檔案系統相比，e2fsck 會分析整個檔案系統，不只是中繼資料最近修改的位元而已。這比檢查日誌檔案系統的記錄資料，要花費更多時間。按照檔案系統大小，此程序會花半小時或更長的時間。因此，不要為任何需要高可用性的伺服器選擇 Ext2。不過，因為 Ext2 不會維護日誌，而且使用相當少的記憶體，因此有時候比其他檔案系統較快速一些。

升級容易

因為 Ext3 是以 Ext2 程式碼為基礎，而且共用它的磁碟上格式和中繼資料格式，所以從 Ext2 升級至 Ext3 十分容易。

1.2.2 Ext3

Ext3 是由 Stephen Tweedie 設計。不像其他所有下一代檔案系統，Ext3 不依循全新的設計原則。它是以 Ext2 為基礎。這兩個檔案系統彼此關係十分密切。Ext3 檔案系統可以輕易地建立在 Ext2 檔案系統的最上層。Ext2 和 Ext3 最重要的差別是 Ext3 支援日誌處理。簡而言之，Ext3 提供三個主要優點：

- 章節「可輕易從 Ext2 升級，並具有很高的可靠性」[4頁]
- 章節「可靠性和效能」[4頁]
- 章節「將 Ext2 檔案系統轉換成 Ext3」[4頁]

可輕易從 Ext2 升級，並具有很高的可靠性

Ext3 以 Ext2 的程式碼做為強大的基礎，因此可以成為眾人喝采的下一代檔案系統。在 Ext3 中完美融合了 Ext2 的可靠性和穩固性特點，同時具備記錄檔案系統的優點。不像轉換至其他日誌檔案系統 (例如 ReiserFS、JFS 或 XFS) 那麼冗長乏味 (備份整個檔案系統，然後從頭建立)，轉換至 Ext3 只是數分鐘的事。它還非常安全，因為從頭開始重新建立整個檔案系統，並不能確保萬無一失。考慮一下等候升級至記錄檔案系統的現有 Ext2 系統數量，您可以輕易瞭解為什麼 Ext3 對很多系統管理員都具有一定重要性。從 Ext3 降級至 Ext2 就和升級一樣容易。只要乾淨取消掛載 Ext3 檔案系統，然後重新掛載成 Ext2 檔案系統就可以了。

可靠性和效能

其他日誌檔案系統，有些會依照「僅中繼資料」日誌方法。這表示您的中繼資料會始終維持一致的狀態，但這並不能自動保證檔案系統資料本身的一致性。Ext3 的設計是妥善管理中繼資料和資料二者。「管理」的程度可以自定。在 `data=journal` 模式啟用 Ext3，可提供最大的安全性 (資料整合性)，不過因為中繼資料和資料是記錄為日誌，所以系統速度會減慢。較新的方法是使用 `data=ordered` 模式，這樣可以確定資料和中繼資料整合性，不過僅限中繼資料使用日誌處理。檔案系統驅動程式會收集所有對應至某一中繼資料更新的所有資料區塊。更新中繼資料前，這些資料區塊會寫入硬碟。如此一來便可以達到中繼資料和資料的一致性，不會犧牲效能。第三個要使用的選項是 `data=writeback`，允許在其中繼資料已經提交至日誌後，將資料寫入主要檔案系統。一般認為此選項的效能最好。不過，它可以允許在損毀和復原舊資料後，重新顯示舊資料，同時又維護內部檔案系統整合性。Ext3 使用 `data=ordered` 選項做為預設值。

將 Ext2 檔案系統轉換成 Ext3

若要將 Ext2 檔案系統轉換為 Ext3，請執行下列步驟：

1 以 root 使用者身分執行 tune2fs -j 來建立 Ext3 日誌。

這樣會以預設參數建立 Ext3 日誌。

若要指定日誌的大小以及存放它的設備，請執行 tune2fs -J，不要同時使用需要的日誌選項 size= 和 device=。如需 tune2fs 程式的詳細資訊，請參閱 tune2fs 線上文件。

2 以 root 使用者身分編輯檔案 /etc/fstab，將為對應分割區指定的檔案系統類型從 ext2 變更為 ext3，然後儲存變更。

這可確保 Ext3 檔案系統會被識別為 Ext3 檔案系統。完成的變更會在下次啟動時生效。

3 若要開機設定為 Ext3 分割區的根目錄檔案系統，請在 initrd 中加入模組 ext3 和 jbd。

3a 以 root 身分編輯 /etc/sysconfig/kernel，將 ext3 和 jbd 新增至 INITRD_MODULES 變數，然後儲存變更。

3b 執行 mkinitrd 指令。

這樣就可以建置新的 initrd，並準備使用。

4 重新啟動系統。

1.2.3 Oracle Cluster File System 2

OCFS2 為日誌式檔案系統，專為叢集設定量身訂做。與標準的單節點檔案系統 (例如 Ext3) 不同，OCFS2 可管理數個節點。OCFS2 允許透過共用儲存分佈檔案系統，例如 SAN 或多重路徑設定。

OCFS2 設定中的每個節點都可以同時讀取和寫入所有資料。這要求 OCFS2 可以識別業集，也就是說 OCFS2 必須包含一種方法，用於確定叢集的組成節點，及確認這些節點是否實際存在，是否可以使用。為計算叢集的成員，OCFS2 包含了節點管理員。為監看叢集中節點的可用性，OCFS2 包含了簡易的活動訊號實作。為避免各種節點直接存取檔案系統而產生的問題，OCFS2 還包含了分散式鎖定管理員。節點間的通訊由 TCP 訊息系統處理。

OCFS2 包含的主要功能與優點為：

- 中繼資料快取與日誌記錄
- 對資料庫檔案提供非同步且直接 I/O 支援，以加強資料庫效能
- 支援高達 4 KB 的多區塊大小 (各磁碟區可具有不同的區塊大小)，磁碟區的最大大小為 16 TB
- 跨節點資料檔案一致性
- 支援高達 255 個叢集節點

如需有關 OCFS2 更深入詳盡的資訊，請參閱《*高可用性儲存基礎結構管理指南*》。

1.2.4 ReiserFS

正式說來，2.4 核心版本的重要功能之一，ReiserFS，自其 6.4 版以來早已被 2.2.x SUSE 核心當作核心修補程式使用。ReiserFS 是由 Hans Reiser 與 Namesys 開發團隊所設計。ReiserFS 已經證實是 Ext2 的強大替代方案。其重要優點是磁碟空間使用率佳，磁碟存取效能好，當機復原快，以及使用資料記錄確保可靠性。

- 章節「更佳的磁碟空間利用」[6頁]
- 章節「更佳的磁碟存取效能」[7頁]
- 章節「快速損毀復原」[7頁]
- 章節「透過資料日誌的可靠性」[7頁]

更佳的磁碟空間利用

在 ReiserFS 中，所有資料皆按照名為 B*-平衡樹的結構整理。樹狀結構提供更佳的磁碟空間利用，因為小的檔案可以直接儲存在 B* 樹葉節點，而不是儲存在別處，而且只維護實際磁碟位置的指標。此外，未以 1 或 4 kB 的區塊配置儲存體，而是按需要的正確大小。另一項優點則依賴 inode 的動態配置。這樣會使得檔案系統比傳統的檔案系統更有彈性，例如在 Ext2，inode 密度必須在檔案系統建立期間指定。

更佳的磁碟存取效能

至於小的檔案，檔案資料和「stat_data」(inode) 資訊兩者通常儲存在一起。它們可以使用單一磁碟 I/O 作業來讀取，這表示您只需要存取一次磁碟，便能擷取所有需要的資訊。

快速損毀復原

使用日誌來追蹤最新的中繼資料變更，只要幾秒便能檢查檔案系統，即使很大的檔案系統也沒問題。

透過資料日誌的可靠性

ReiserFS 還支援資料記錄和排序資料模式，此模式類似第 1.2.2 節「Ext3」[3頁] 中簡述的概念。預設模式為 `data=ordered`，這個模式可以確保資料和中繼資料的完整性，但是日誌僅適用中繼資料。

1.2.5 XFS

1990 年代早期，SGI 開始對原先要當成 IRIX OS 的檔案系統 XFS 進行研發。XFS 隱含的目標是建立高效能 64 位元記錄檔案系統，以滿足嚴格的計算挑戰。XFS 對於操控大型檔案以及執行高階硬體，具備良好功能。不過，XFS 還是有一個缺點。和 ReiserFS 一樣，XFS 專注於中繼資料完整性，而不重視資料的完整性。

以下是 XFS 主要功能的快速回顧，這些正是為什麼它在高階計算中比其他記錄檔案系統更具競爭力的證明。

- 章節「透過使用配置群組取得的高擴充性」[8頁]
- 章節「透過有效磁碟空間管理取得高效能」[8頁]
- 章節「預先配置來避免檔案系統零散化」[8頁]

透過使用配置群組取得的高擴充性

建立 XFS 檔案系統時，檔案系統所屬的區塊設備，會分割成 8 或更多等同大小的線性區域。這些稱為*配置群組*。每一個配置群組管理自己的 inode 以及可用的磁碟空間。事實上，配置群組可以看成是檔案系統中的檔案系統。因為配置群組彼此各自獨立，所以核心可以同時處理一個以上的配置群組。此功能是 XFS 具備優良延展性的關鍵。當然，獨立配置群組的概念也符合多處理器系統的需求。

透過有效磁碟空間管理取得高效能

可用空間和 inode 是由配置群組裡面的 B^+ 樹處理。使用 B^+ 樹可大大增強 XFS 的效能和延展性。XFS 使用*延遲配置*，透過將程序分為兩個部分來處理配置。待處理的交易會儲存在 RAM 並保留適當的空間。XFS 仍然沒有決定資料儲存的確切位置 (在檔案系統區塊中)。此決策會盡量延緩至最後時刻。部分暫時資料永遠不會儲存至磁碟，因為當 XFS 決定了其實際儲存位置時，它早已過時了。採用這種方式，XFS 將增加寫入效能並減少檔案系統片段。因為比起其他檔案系統，延遲配置會導致較少的寫入事件，這樣在寫入過程中若是發生當機就會導致較嚴重的資料遺失。

預先配置來避免檔案系統零散化

寫入資料至檔案系統前，XFS 會*保留* (預先配置) 檔案需要的可用空間。因此，可大幅降低檔案系統零散化。因為檔案的內容是分佈在檔案系統中，所以效能就會提高。

1.3 其他受支援的檔案系統

表格 1.1 「Linux 的檔案系統類型」 [8頁]彙整 Linux 支援的其他檔案系統。提供其他系統支援主要是確定不同媒體或外來作業系統中，資料交換的相容性。

表格 1.1 *Linux 的檔案系統類型*

檔案系統類型	描述
`cramfs`	壓縮的 ROM 檔案系統：ROM 的一種壓縮唯讀檔案系統。

檔案系統類型	描述
hpfs	高效能檔案系統：IBM* OS/2* 標準檔案系統。僅在唯讀模式下受支援。
iso9660	CD-ROM 的標準檔案系統。
minix	源自作業系統學術研究專案的檔案系統，是 Linux 使用的第一個檔案系統。現在，它可作為磁片檔案系統來使用。
msdos	fat 最早源自 DOS 的檔案系統，現在各種作業系統均使用之。
ncpfs	透過網路掛接 Novell® 磁碟區的檔案系統。
nfs	網路檔案系統：使用這種檔案系統，資料可以儲存在網路中的任何機器上，而且可以經由授權從網路存取。
smbfs	一些產品 (例如 Windows*) 使用伺服器訊息區塊透過網路存取檔案。
sysv	用於 SCO UNIX*、Xenix 和 Coherent (個人電腦的商用 UNIX 系統) 上。
ufs	用於 BSD*、SunOS* 和 NextStep*。僅支援唯讀模式。
umsdos	MS-DOS* 上的 UNIX：適用於標準 fat 檔案系統之上，透過建立特殊檔案來實現 UNIX 功能 (許可權、連結、長檔案名稱)。
vfat	虛擬 FAT：fat 檔案系統的副檔名 (支援長檔名)。
ntfs	Windows NT 檔案系統；唯讀。

1.4 Linux 的大型檔案支援

一開始，Linux 支援的檔案大小最多是 2 GB。在多媒體引爆之前，而且只要沒有人試著在 Linux 操控大型資料庫，這已經夠用了。當應用程式必須使用的一組新介面時，修改核心和 C 程式庫以支援超過 2 GB 的檔案大小，對於伺服器計算變得越來越重要。現在，幾乎所有主要檔案系統都提供 LFS 支援，以執行高階計算。表格 1.2「檔案系統的大小上限 (磁碟上格式)」[10頁] 提供 Linux 檔案和檔案系統目前限制的綜覽。

表格 1.2 *檔案系統的大小上限 (磁碟上格式)*

檔案系統	檔案大小 (位元組)	檔案系統大小 (位元組)
Ext2 或 Ext3 (1 KB 區塊大小)	2^{34} (16 GB)	2^{41} (2 TB)
Ext2 或 Ext3 (2 KB 區塊大小)	2^{38} (256 GB)	2^{43} (8 TB)
Ext2 或 Ext3 (4 KB 區塊大小)	2^{41} (2 TB)	2^{44}-4096 (16 TB-4096 位元組)
Ext2 或 Ext3 (8 KB 區塊大小) (含 8 KB 頁面的系統，例如 Alpha)	2^{46} (64 TB)	2^{45} (32 TB)
ReiserFS v3	2^{46} (64 TB)	2^{45} (32 TB)
XFS	2^{63} (8 EB)	2^{63} (8 EB)
NFSv2 (用戶端)	2^{31} (2 GB)	2^{63} (8 EB)
NFSv3 (用戶端)	2^{63} (8 EB)	2^{63} (8 EB)

重要

表格 1.2「檔案系統的大小上限 (磁碟上格式)」[10頁]會說明磁碟上 (On-Disk) 格式的限制。2.6 Linux 核心會強制其處理的檔案和檔案系統依循自身大小限制。限制如下：

檔案大小
在 32 位元系統上，檔案不能超過 2 TB (2^{41} 位元組)。

檔案系統大小
檔案系統大小最大可達 2^{73} 位元組。不過，此限制仍然跟不上目前可用的硬體。

1.5 其他資訊

Novell 網站上的《*File System Primer*》*(檔案系統入門)* [http://wiki.novell.com/index.php/File_System_Primer] 描述了多種適用於 Linux 的檔案系統。該文件介紹了存在多種檔案系統的原因以及哪些檔案系統最適用於哪些工作負載和資料。

請造訪上述每種檔案系統專案維護的專屬首頁，找出的郵件清單資訊、詳盡文件以及常見問題：

- *E2fsprogs: Ext2/3/4 Filesystem Utilities (Ext2/3/4 檔案系統公用程式)* [http://e2fsprogs.sourceforge.net/]
- *Introducing Ext3 (Ext3 簡介)* [http://www.ibm.com/developerworks/linux/library/l-fs7.html]
- *ReiserFSprogs* [http://chichkin_i.zelnet.ru/namesys/]
- *XFS: A High-Performance Journaling Filesytem (高效能記錄檔案系統)* [http://oss.sgi.com/projects/xfs/]
- *OCFS2 Project (OCFS2 專案)* [http://oss.oracle.com/projects/ocfs2/]

有關Linux檔案系統的多部份全面教學課程，請參閱《*進階檔案系統實作指南*》[http://www-106.ibm.com/developerworks/library/l-fs.html]中的 IBM developerWorks。Wikipedia 計畫上的《*Comparison of File Systems*》*(檔案系統比較)* [http://en.wikipedia.org/wiki/Comparison_of_file_systems#Comparison] 中提供了對幾種檔案系統 (不僅是 Linux 檔案系統) 的深入比較。

2 新增功能

本章所述的是 SUSE® Linux Enterprise Server 11 中的功能與行為變更。

2.1 SLES 11 SP1 中的新增功能

除了錯誤修正之外，本節所述的是 SUSE® Linux Enterprise Server 11 SP1 版本中的功能與行為變更。

- 第 2.1.9 節「MD-SGPIO」[16頁]
- 第 2.1.10 節「調整 LVM 2 鏡像大小」[16頁]
- 第 2.1.11 節「更新適用於 IBM 伺服器上之介面卡的儲存驅動程式」[16頁]

2.1.1 儲存 iSCSI 目標資訊

在「*YaST*」>「*網路服務*」>「*iSCSI 目標*」功能中，新增了「*儲存*」選項，可讓您輸出 iSCSI 目標資訊。這樣，為資源使用者提供資訊便會更為方便。

2.1.2 在 iSCSI 啟動器中修改驗證參數

在「*YaST*」>「*網路服務*」>「*iSCSI 啟動器*」功能中，您可以修改用於連接目標設備的驗證參數。以前，您需要先刪除該項目然後重新建立，才能變更驗證資訊。

2.1.3 允許針對 MPIO 設備進行永久保留

SCSI 啟動器可以對共享的儲存設備施加 SCSI 保留，如此可將其他伺服器上的 SCSI 啟動器鎖定在外，阻止其存取該設備。即使 SCSI 例外處理程序進行 SCSI 重設時，這些保留也依然存在。

以下情況可能需要使用 SCSI 保留：

- 在簡單的 SAN 環境中，若嘗試將某 LUN 新增至一部伺服器，而該 LUN 正被其他伺服器所使用，則可能會導致資料損毀，永久的 SCSI 保留有助於防止出現此類管理員錯誤。SAN 分區通常用於防止出現此類錯誤。
- 在設定了容錯移轉的高可用性環境中，永久的 SCSI 保留有助於防止連錯路徑的伺服器連接到其他伺服器所保留的 SCSI 設備。

2.1.4 MDADM 3.0.2

使用最新版本的多設備管理 (MDADM，`mdadm`)，其中提供了一些錯誤修正並改進了部分功能。

2.1.5 MDRAID 外部中繼資料的開機載入程式支援

新增了該支援以使用 MDADM 公用程式 3.0 版的外部中繼資料功能，從 Intel* 矩陣儲存技術中繼資料格式所定義的 RAID 磁碟區安裝和執行作業系統。這就將該功能從設備對應程式 RAID (DMRAID) 基礎架構移轉到多設備 RAID (MDRAID) 基礎架構，從而提供更為成熟的 RAID 5 實作，以及更豐富的 MD 核心基礎架構功能。如此，在包括 Intel、DDF (常用 RAID 磁碟資料格式) 和原始 MD 中繼資料在內的所有中繼資料格式中都可使用常用 RAID 驅動程式。

2.1.6 MDRAID 外部中繼資料的 YaST 安裝和開機支援

在 YaST® 安裝程式工具中新增了對 RAID 0、1、10、5 和 6 的 MDRAID 外部中繼資料支援。該安裝程式可以偵測 RAID 陣列以及平台是否啟用了 RAID 功能。若適用於 Intel Matrix Storage Manager 的平台 BIOS 中啟用 RAID，安裝程式會提供「DMRAID」、「MDRAID」(建議) 選項，若未啟用，則不提供任何選項。此外還修改了 `initrd`，以支援將基於 BIOS 的 RAID 陣列組合起來的功能。

2.1.7 包含根檔案系統之 MDRAID 陣列的關機已改進

關機程序檔已修改為等待所有 MDRAID 陣列標示為清空。現在，作業系統關機程序會一直等待改動位元的清除，直到所有 MDRAID 磁碟區完成寫入作業為止。

對啟動程序檔、關機程序檔以及 `initrd` 進行了變更，以確定根 (/) 檔案系統 (包含作業系統與應用程式檔案的系統磁碟區) 是否駐留在軟體 RAID 陣列中。陣列的中繼資料處理器在關機程序中較早啟動，以在關機過程中監控最終的根檔案系統環境。該處理器已從一般 `killall` 事件中排除。該程序還允許在關機結束時靜止寫入並清除陣列的中繼資料改動位元(指示陣列是否需要重新同步)。

2.1.8 iSCSI 設備上的 MD

現在，YaST 安裝程式允許經由 iSCSI 設備設定 MD。

若開機時需要 RAID 陣列，則會在 `boot.md` 之前載入 iSCSI 啟動器軟體，以便為系統提供針對 RAID 自動設定的 iSCSI 目標。

若是新的安裝，Libstorage 會建立 `/etc/mdadm.conf` 檔案，並新增 `AUTO -all` 一行。更新期間不會新增該行。若 `/etc/mdadm.conf` 包含該行，

```
AUTO -all
```

則所有 RAID 陣列均不會自動組合，除非 `/etc/mdadm.conf` 中明確列出了某些陣列。

2.1.9 MD-SGPIO

MD-SGPIO 公用程式是一個獨立應用程式，可透過 `sysfs(2)` 監控 RAID 陣列。有些事件會觸發控制 LED 燈閃爍的 LED 變更申請，而這些 LED 燈與機箱或儲存子系統磁碟機槽中的每個插槽相關聯。它支援兩種類型的 LED 系統：

- 雙 LED 系統 (活動 LED 和狀態 LED)
- 三 LED 系統 (活動 LED、位置 LED 和失敗 LED)

2.1.10 調整 LVM 2 鏡像大小

修改了用於調整邏輯磁碟區大小的 `lvresize`、`lvextend` 與 `lvreduce` 指令，以允許調整 LVM 2 鏡像的大小。以前，如果邏輯磁碟區是一個鏡像，這些指令便會報告錯誤。

2.1.11 更新適用於 IBM 伺服器上之介面卡的儲存驅動程式

更新以下儲存驅動程式以使用最新的可用版本支援 IBM 伺服器上的儲存介面卡。

- Adaptec™：`aacraid`、`aic94xx`
- Emulex™：`lpfc`
- LSI™：`mptas`、`megaraid_sas`

  現在，`mptsas` 驅動程式支援原生 EEH (增強型錯誤處理器) 復原，這是進階平台使用者所有 IO 設備的核心功能。
- qLogic™：`qla2xxx`、`qla3xxx`、`qla4xxx`

2.2 SLES 11 中的新增功能

本節所述的是 SUSE® Linux Enterprise Server 11 版本的功能與行為變更。

2.2.1 EVMS2 已廢棄

企業磁碟區管理系統 (EVMS2) 儲存管理解決方案已廢棄。SUSE Linux Enterprise Server 11 套件中已移除了所有 EVMS 管理模組。您在升級系統時，Linux Volume Manager 2 (LVM2) 會自動識別 EVMS 受管設備並加以管理。如需詳細資訊，請參閱《*Evolution of Storage and Volume Management in SUSE Linux Enterprise*》*(SUSE Linux Enterprise 中儲存和磁碟區管理的演進)* [http://www.novell.com/linux/volumemanagement/strategy.html]。

如需有關在 SUSE Linux Enterprise Server 10 中使用 EVMS2 管理儲存的資訊，請參閱《*SUSE Linux Enterprise Server 10 SP3：儲存管理指南*》[http://www.novell.com/documentation/sles10/stor_admin/data/bookinfo.html]。

2.2.2 Ext3 做為預設檔案系統

Ext3 檔案系統已取代 ReiserFS 做為 YaST 工具在安裝和建立檔案系統時建議使用的預設檔案系統。ReiserFS 仍受支援。如需詳細資訊，請參閱 *SUSE Linux Enterprise 10 檔案系統支援*網頁上的《*File System Future Directions*》*(檔案系統未來方向)* [http://www.novell.com/linux/techspecs.html?tab=0]。

2.2.3 JFS 檔案系統已廢棄

不再支援 JFS 檔案系統，已從該套裝作業系統中移除 JFS 公用程式。

2.2.4 OCFS2 檔案系統包含於高可用性版本中

SUSE Linux Enterprise 高可用性延伸完全支援 OCFS2 檔案系統。

2.2.5 /dev/disk/by-name 已廢棄

在 SUSE Linux Enterprise Server 11 套件中，`/dev/disk/by-name` 路徑已廢棄。

2.2.6 /dev/disk/by-id 目錄中的設備名稱永久不變

在 SUSE Linux Enterprise Server 11 中，當啟動多重路徑時，預設多重路徑設定依賴 udev 覆寫 /dev/disk/by-id 目錄中的現存符號連結。在您啟動多重路徑之前，該連結透過使用設備的 scsi-xxx 名稱來指向 SCSI 設備。當多重路徑正在執行時，該符號連結則透過使用設備的 dm-uuid-xxx 名稱來指向設備。這樣可確保不論多重路徑是否啟動，/dev/disk/by-id 路徑中的符號連結始終指向同一設備。由於 lvm.conf 和 md.conf 等組態檔案會自動指向正確的設備，所以無須對其進行修改。

如需關於此行為變更如何影響其他功能的詳細資訊，請參閱以下各節：

- 第 2.2.7 節「多重路徑設備的過濾器」[19頁]
- 第 2.2.8 節「多重路徑設備的使用者易記名稱」[20頁]

2.2.7 多重路徑設備的過濾器

/dev/disk/by-name 目錄的廢棄 (如第 2.2.5 節「/dev/disk/by-name 已廢棄」[18頁] 中所述) 會影響您在組態檔案中設定多重路徑設備過濾器的方式。若在 /etc/lvm/lvm.conf 檔案中，多重路徑設備過濾器使用的是 /dev/disk/by-name 設備名稱路徑，則您需要將該檔案修改為使用 /dev/disk/by-id 路徑。設定使用 by-id 路徑的過濾器時，請注意以下事項：

- /dev/disk/by-id/scsi-* 設備名稱永久不變，且建立它僅是為了該目的。
- 不要在過濾器中使用 /dev/disk/by-id/dm-* 名稱。它們是設備對應程式設備的符號連結，使用這些名稱會導致在回應 pvscan 指令時報告重複的 PV。這些名稱會從 LVM-pvuuid 變更為 dm-uuid，再變回 LVM-pvuuid。

如需關於設定過濾器的資訊，請參閱第 7.2.3 節「在多重路徑設備上使用 LVM2」[57頁]。

2.2.8 多重路徑設備的使用者易記名稱

在 /dev/disk/by-id 目錄中對多重路徑設備名稱處理方式的變更 (如第 2.2.6 節「/dev/disk/by-id 目錄中的設備名稱永久不變」[19頁] 中所述) 會影響您對使用者易記名稱的設定，因為設備的這兩個名稱有所不同。您必須將組態檔案修改為在設定多重路徑後僅掃描設備對應程式名稱。

例如，您需要修改 lvm.conf 檔案以使用多重路徑設備名稱進行掃描，方法是指定 /dev/disk/by-id/dm-uuid-.*-mpath-.* 路徑，而非 /dev/disk/by-id。

2.2.9 多重路徑的進階 I/O 負載平衡選項

除了輪替之外，系統還提供了以下適用於設備對應程式多重路徑的進階 I/O 負載平衡選項：

- Least-pending
- Length-load-balancing
- Service-time

如需更多資訊，請參閱章節「瞭解優先程序群組與屬性」[83頁]。

2.2.10 多重路徑工具 Callout 的位置變更

設備對應程式多重路徑工具的 mpath_* prio_callout 已移至位於 /lib/libmultipath/lib* 中的共享程式庫中。透過使用共享程式庫，callout 會在精靈啟動時載入到記憶體中。這有助於避免在所有路徑失效的情況下出現系統鎖死，例如當需要從磁碟載入程式而磁碟此時不可用的情況下。

2.2.11 mkinitrd -f 的選項從 mpath 變更為 multipath

將設備對應程式多重路徑服務新增到 initrd 的選項已從 -f mpath 變更為 -f multipath。

若要建立新的 initrd，現在指令變更為：

```
mkinitrd -f multipath
```

3 規劃儲存解決方案

請考慮所需儲存及如何才能有效地管理和分割儲存空間以最大限度地滿足您的需要。本章中的資訊可幫助您規劃 SUSE® Linux Enterprise Server 11 伺服器上之檔案系統的儲存部署。

3.1 分割設備

如需使用 YaST 進階磁碟分割程式的資訊，請參閱《*SUSE Linux Enterprise Server 11 安裝與管理指南*》中的「使用 YaST 磁碟分割程式」。

3.2 多重路徑支援

Linux 支援使用多重 I/O 路徑在伺服器及其儲存設備之間建立容錯連接。預設會停用 Linux 多重路徑支援。如果您使用的是儲存子系統廠商提供的多重路徑解決方案，則不必分別設定 Linux 多重路徑。

3.3 軟體 RAID 支援

Linux 支援硬體和軟體 RAID 設備。如果您使用的是硬體 RAID 設備，則無需軟體 RAID 設備。可以在同一伺服器上同時使用硬體和軟體 RAID 設備。

為了最大化軟體RAID設備的效能優點，RAID使用的分割區應來自不同的實體設備。對於軟體 RAID 1 設備，鏡像複製分割區無法共用任何相同的磁碟。

3.4 檔案系統快照

Linux 支援檔案系統快照。

3.5 備份與防毒支援

- 第 3.5.1 節「開放原始碼備份」[24頁]
- 第 3.5.2 節「商業備份與防毒支援」[24頁]

3.5.1 開放原始碼備份

用於在 Linux 上進行資料備份的開放原始碼工具包括 `tar`、`cpio` 和 `rsync`。如需詳細資訊，請參閱這些工具的 man 頁面。

- PAX：POSIX 檔案系統歸檔程式。該歸檔程式支援最常用的兩種標準歸檔(備份) 檔案格式 `cpio` 和 `tar`。若需更多資訊，請參閱 man 頁面。
- Amanda：進階馬里蘭自動網路磁碟歸檔程式。請造訪 www.amanda.org [http://www.amanda.org/]。

3.5.2 商業備份與防毒支援

Novell® Open Enterprise Server (OES) 2 Support Pack 1 for Linux 產品中包含 SUSE Linux Enterprise Server (SLES) 10 Support Pack 2。防毒和備份軟體廠商同時支援

OES 2 SP1 和 SLES 10 SP2。您可以造訪廠商網站瞭解他們排定的 SLES 11 支援。

如需目前可用的備份和防毒軟體廠商清單，請參閱 *Novell Open Enterprise Server 合作夥伴支援：備份和防毒支援* [http://www.novell.com/products/openenterpriseserver/partners_communities.html]。此清單每季度更新一次。

4

LVM 組態

本章概述了邏輯磁碟區管理員 (LVM) 背後的原則，以及可適合多種狀況下使用的基本功能。YaST LVM 組態可透過 YaST 進階磁碟分割程式來完成。這個磁碟分割工具讓您編輯和刪除現有磁碟分割，以及建立應該與LVM一起使用的新磁碟分割。

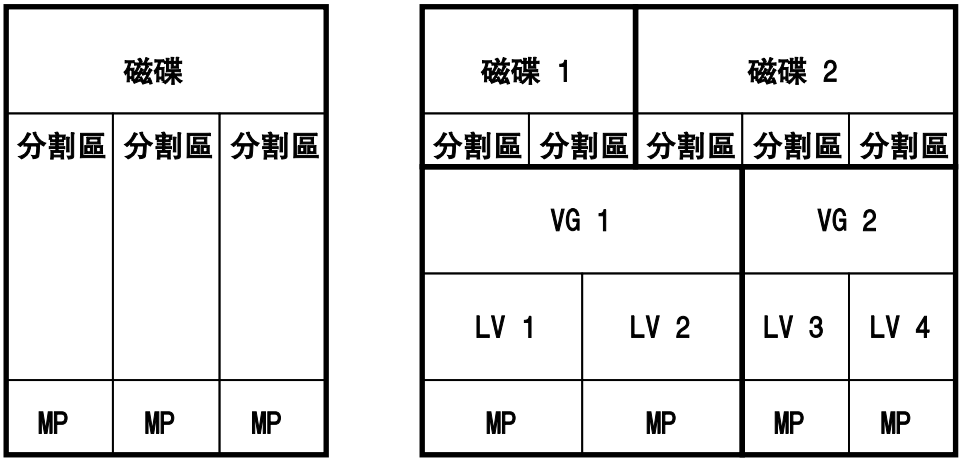
警告

使用 LVM 可能會增加風險，如遺失資料。這些危險也包括應用程式當機、電源中斷和錯誤指令。執行 LVM 或重新設定磁碟區前，請儲存您的資料。決不要在沒有備份的情形下工作。

4.1 瞭解邏輯磁碟區管理員

LVM 可以跨數個檔案系統彈性地分配硬碟空間。在安裝過程中，僅當完成首次磁碟分割時需要變更硬碟空間的分割，由此開發了此工具。因為要修改正在執行之系統上的分割區很困難，LVM 提供了記憶體空間的虛擬集區 (磁碟區群組，或 VG)，需要時，可從虛擬集區建立邏輯磁碟區 (LV)。作業系統可以存取這些 LV，而不是存取實體分割區。磁碟區群組可以延伸至一個以上的磁碟，因此數個磁碟或是數個磁碟的某些部份可能會構成單一的 VG。借此，LVM 提供了一種從實體磁碟空間進行擷取的方法，允許使用比實體磁碟重新分割更為簡單和安全的方式來變更分割。

*圖形 **4.1**　實體分割與 LVM*



圖形 4.1「實體分割與 LVM」[28頁]比較實體分割 (左邊) 與 LVM 分割 (右邊)。在左邊，單一個磁碟已分割為三個實體分割區 (PART)，每一個都會指定掛接點 (MP)，讓作業系統存取它們。在右邊，已經個別將兩個磁碟分割成兩個及三個實體分割區。已經定義兩個 LVM 磁碟區群組 (VG1 與 VG2)。VG1 包含 DISK1 的兩個磁碟區以及 DISK2 的一個磁碟區。VG2 包含 DISK2 其餘的兩個磁碟區。在 LVM 中，在磁碟區群組中合併的實體磁碟分割區稱為實體磁碟區 (PV)。在某些磁碟區群組中，已經定義四個邏輯磁碟區 (LV1 至 LV4)，作業系統可以透過指定的掛接點來使用。在不同的邏輯磁碟區之間的邊緣，不需要對齊任何分割區的邊緣。請參閱此範例中 LV 1 與 LV 2 之間的邊緣。

LVM 功能：

- 數個硬碟或分割區可以在大的邏輯磁碟區結合成一個。
- 如果組態適用，當可用空間耗盡時，可以擴大 LV (如 `/usr`)。
- 使用LVM，就可以在執行的系統中新增硬碟或LV。然而，這種作法需要能執行此動作的熱交換式硬體。
- 可以啟動*分割模式*，將邏輯磁碟區的資料流分散至數個實體磁碟區。如果這些實體磁碟區是在不同的磁碟上，這可改善讀寫效能，就像 RAID 0 一樣。
- 快照功能能夠讓執行系統中的備份 (特別是伺服器) 成為一致。

使用LVM的這些功能，對於使用頻繁的家用個人電腦或小型伺服器而言，在效能上可以看到改善。如果您在資料庫、音樂歸檔或使用者目錄中的資料會一直累積，LVM 就是非常適用的工具。可允許比實體硬碟還大的檔案系統。LVM 的另一個好處是最大可以增加到 256 個 LV。不過，請記住使用 LVM 與使用傳統分割區是不同的。有關設定LVM的指示及詳細資訊，請參閱官方網站的*LVM HOWTO* [`http://tldp.org/HOWTO/LVM-HOWTO/`]。

從核心 2.6 版本開始，即可使用 LVM 2 版本，它可以向下相容之前的 LVM，而且可以繼續管理舊的磁碟區群組。建立新的磁碟區群組時，請決定要使用新的格式或能夠向下相容的版本。LVM 2 不需要任何核心修補程式。這會用到整合於核心 2.6 中的設備對應程式。此核心僅支援 LVM 第 2 版。因此，提到 LVM 時，本節一律指的是 LVM 第 2 版。

4.2 建立 LVM 分割區

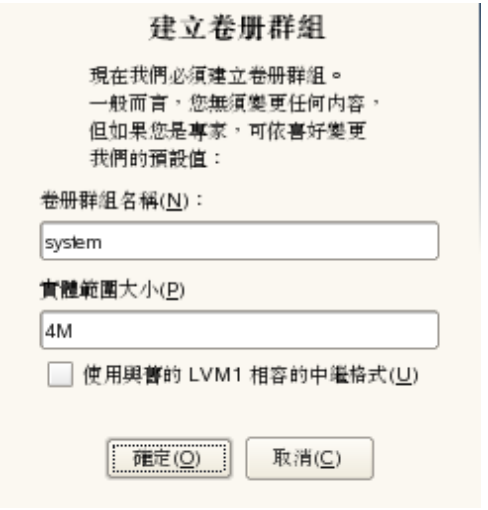
先按一下「*建立*」>「*不格式化*」，再選取「*0x8E Linux LVM*」做為分割區識別碼，可建立 LVM 分割區。在建立所有要與 LVM 一起使用的分割區後，按一下「*LVM*」以啟動 LVM 組態。

4.3 建立磁碟區群組

如果在系統上沒有磁碟區群組，將會提示您新增一個磁碟區群組 (請參閱圖形 4.2「建立磁碟區群組」 [30頁])。可以使用「*新增群組*」來建立其他群組，但通常一個磁碟區群組已經足夠。建議使用 `system` 做為 SUSE® Linux Enterprise Server 系統檔案所在磁碟區群組的名稱。實體擴充大小定義了磁碟區群組中實

體區塊的大小。在磁碟區群組中的所有磁碟空間都會以此大小的區塊來處理。此值通常設為 4 MB，並允許將實體及邏輯磁碟區的最大容量設為 256 GB。只有在需要大於 256 GB 的邏輯磁碟區時，才需要增加實體擴充大小的容量 (例如，設為 8、16 或 32 MB)。

圖形 4.2　建立磁碟區群組

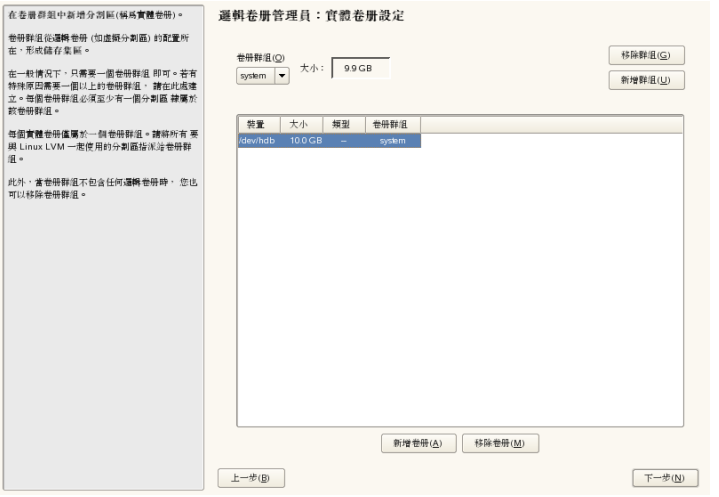


4.4 設定實體磁碟區

一旦建立磁碟區群組，下一個對話方塊 (請參閱圖形 4.3「實體磁碟區設定」[31頁]) 就會列出類型為「Linux LVM」或「Linux native」的所有分割區。不會顯示交換和 DOS 分割區。如果已經指定分割區給磁碟區群組，磁碟區群組的名稱就會顯示在清單中。未指定的分割區以「--」表示。

如果有數個磁碟區群組，請在左上角的選擇方塊中設定目前的磁碟區群組。右上角的按鈕可以建立其他的磁碟區群組以及刪除現有的磁碟區群組。僅能刪除沒有指定分割區的磁碟區群組。所有指定給磁碟區群組的分割區，又稱為實體磁碟區。

圖形 4.3　實體磁碟區設定

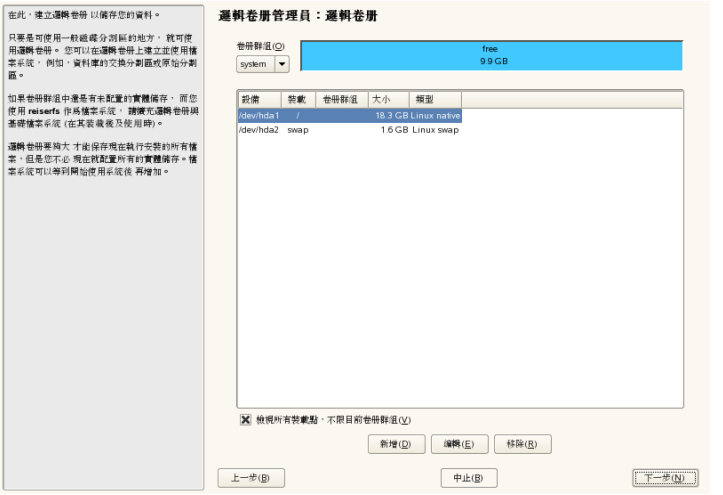


若要將之前未指定的分割區新增至選取的磁碟區群組，請先按一下分割區，再按「*新增磁碟區*」。此時，磁碟區群組的名稱，會輸入到選取分割區的旁邊。指定為LVM保留的所有分割區給磁碟區群組。否則，仍然不會使用分割區上的空間。結束此對話方塊前，每個磁碟區群組必須指定至少一個實體磁碟區。在指定所有的實體磁碟區後，按一下「*下一步*」以繼續邏輯磁碟區的組態。

4.5 設定邏輯磁碟區

磁碟區群組中填入實體磁碟區後，使用下一個對話方塊 (請參閱圖形 4.4「邏輯磁碟區管理」[32頁]) 可定義作業系統應使用的邏輯磁碟區。在左上角的選項方塊中修改目前的磁碟區群組。接下來，會顯示目前磁碟區群組的可用空間。下方的清單包含該磁碟區群組中所有的邏輯磁碟區。指定掛接點的所有標準Linux分割區、所有交換分割區、以及所有已經存在的邏輯磁碟區都列示於此。可以視需要使用「*新增*」、「*編輯*」以及「*移除*」選項管理邏輯磁碟區，直到磁碟區群組中的所有空間都用完為止。至少指定一個邏輯磁碟區給每個磁碟區群組。

圖形 4.4　邏輯磁碟區管理



若要建立新的邏輯磁碟區 (請參閱圖形 4.5「建立邏輯磁碟區」[33頁])，請按一下「*新增*」，然後填寫隨即開啟的快顯視窗。進行磁碟分割時，可指定大小、檔案系統以及掛接點。一般而言，如 Reiserfs 或 Ext2 之類的檔案系統，是先建立於邏輯磁碟區上，接著再指定掛接點。儲存於此邏輯磁碟區上的檔案，可以在已安裝系統的此掛接點上找到。此外也可以將邏輯磁碟區中的資料流分散至數個實體磁碟區 (分割)。如果這些實體磁碟區是在不同的硬碟上，通常可以改善讀寫效能 (像 RAID 0 一樣)。不過，具有 n 個等量磁區的等量 LV，只有在 LV 所需的硬碟空間可以平均分散給 n 個實體磁碟區時，才能正確建立。例如，如果只有兩個可用的實體磁碟區，那麼就不可能建立具有三個等量磁區的邏輯磁碟區。

警告

YaST 在此時沒有機會驗證您所輸入的等量磁區之正確性。在此所犯的錯誤只有稍後在磁碟上執行 LVM 時才會顯示出來。

圖形 4.5　建立邏輯磁碟區

如果已經在系統上設定 LVM，現在就可以指定現有的邏輯磁碟區。在繼續前，請指定適當的掛接點到這些邏輯磁碟區。按一下「*下一步*」返回 YaST 進階磁碟分割程式，然後於該處完成您的工作。

4.6 直接 LVM 管理

如果您已設定 LVM，而且只想變更某些項目，可以使用另一種方法來完成。在 YaST 控制中心選取「*系統*」>「*磁碟分割程式*」。您可以使用前面提及的方法來管理 LVM 系統。

4.7 調整 LVM 分割區大小

`lvresize`、`lvextend` 與 `lvreduce` 指令可用於調整邏輯磁碟區的大小。如需這些指令的語法及選項資訊，請參閱相應指令的 man 頁面。

您還可以使用 YaST 磁碟分割程式增加邏輯磁碟區的大小。YaST 使用 `parted(8)` 來增大分割區。

若要擴充 LV，VG 上必須有足夠多的未配置空間。

您雖然可以在LV正在使用中時對其進行擴充或縮減，但若其上包含檔案系統，則無法如此作業。擴充或縮減 LV 不會自動修改磁碟區中檔案系統的大小。之後必須使用另一個指令來增大檔案系統。如需調整檔案系統大小的相關資訊，請參閱第 5 章「*調整檔案系統大小*」[37頁]。

確保使用正確的順序：

- 若要擴充 LV，則必須先擴充 LV 然後再嘗試增大檔案系統。
- 若要縮減 LV，則必須先縮減檔案系統然後再嘗試縮減 LV。

若要擴充邏輯磁碟區的大小：

1. 開啟終端機主控台，然後以 `root` 使用者身分登入。
2. 如果邏輯磁碟區包含為虛擬機器 (例如 Xen VM) 代管的檔案系統，請關閉該 VM。
3. 卸下邏輯磁碟區上的檔案系統。
4. 在終端機主控台提示符處，輸入以下指令以增加邏輯磁碟區的大小：

   ```
   lvextend -L +size /dev/vgname/lvname
   ```

   對於 *`size`*，請指定您要新增到邏輯磁碟區的空間容量，例如 10GB。以邏輯磁碟區的 Linux 路徑 (例如 `/dev/vg1/v1`) 取代 */dev/vgname/lvname*。例如：

   ```
   lvextend -L +10GB /dev/vg1/v1
   ```

例如，將其上包含 (已掛接並啟用) ReiserFS 的 LV 擴充 10GB：

```
lvextend -L +10G /dev/vgname/lvname
resize_reiserfs -s +10GB -f /dev/vg-name/lv-name
```

例如，將包含 ReiserFS 的 LV 縮減 5GB：

```
umount /mountpoint-of-LV
resize_reiserfs -s -5GB /dev/vgname/lvname
lvreduce /dev/vgname/lvname
mount /dev/vgname/lvname /mountpoint-of-LV
```

5 調整檔案系統大小

當您的資料需要增大磁碟區時，則您可能需要增加分配給其檔案系統的空間容量。

5.1 調整大小準則

調整任何分割區或檔案系統的大小都存在一定的風險，可能會造成資料遺失。

警告

若要避免資料遺失，在開始執行調整大小任務之前，請務必備份資料。

計劃調整檔案系統大小時，請考慮以下準則。

- 第 5.1.3 節「減少檔案系統的大小」[38頁]

5.1.1 支援調整大小的檔案系統

檔案系統必須支援調整大小才能利用為磁碟區增加的可用空間。SUSE® Linux Enterprise Server 11 中提供了適用於檔案系統 Ext2、Ext3 及 ReiserFS 的檔案系統調整大小公用程式。該公用程式支援增加和減少大小，如下所述：

表格 5.1 *檔案系統支援調整大小*

檔案系統	公用程式	增加大小 (增大)	減少大小 (縮減)
Ext2	resize2fs	可以，僅限離線狀態	可以，僅限離線狀態
Ext3	resize2fs	可以，線上或離線狀態均可	可以，線上或離線狀態均可
ReiserFS	resize_reiserfs	可以，線上或離線狀態均可	可以，僅限離線狀態

5.1.2 增加檔案系統的大小

您可以將檔案系統增大到設備的最大可用空間，或指定一個精確值。請務必先增大設備或邏輯磁碟區的大小，然後再嘗試增加檔案系統大小。

為檔案系統指定精確大小時，請確保新大小符合以下條件：

- 新大小必須大於現有資料的大小；否則資料會遺失。
- 新的大小不得超過目前設備的大小，因為檔案系統大小不能超過可用空間大小。

5.1.3 減少檔案系統的大小

當要減少設備上檔案系統的大小時，請確定新大小滿足下列條件：

- 新大小必須大於現有資料的大小；否則資料會遺失。
- 新的大小不得超過目前設備的大小，因為檔案系統大小不能超過可用空間大小。

如果另外還想減少代管檔案系統之邏輯磁碟區的大小，請先減少檔案系統的大小，然後再嘗試減少設備或邏輯磁碟區的大小。

5.2 增大 Ext2 或 Ext3 檔案系統

使用 `resize2fs` 指令掛接或卸載時，可以調整 Ext2 與 Ext3 檔案系統的大小。

1. 開啟終端機主控台，以 `root` 使用者或同等身分登入。
2. 使用下列方法之一增加檔案系統的大小：
   - 若要將檔案系統的大小擴充至名為 `/dev/sda1` 之設備的最大可用大小，請輸入

     ```
     resize2fs /dev/sda1
     ```

     如果未指定大小參數，則預設大小為分割區的大小。
   - 若要將檔案系統擴充至指定大小，請輸入

     ```
     resize2fs /dev/sda1 size
     ```

     `size` 參數可指定所需的檔案系統新大小。如果未指定單位，則大小參數的單位即為檔案系統的區塊大小。也可以選擇透過下列其中一種單位指示項給大小參數加上字尾：s 表示 512 位元組磁區；K 表示 KB (1 KB 為 1024 位元組)；M 表示 MB；G 表示 GB。

   請等候直至完成大小調整，然後再繼續。
3. 如果未掛接檔案系統，請立即掛接。

   例如，若要在掛接點 `/home` 為名為 `/dev/sda1` 的設備掛接 Ext2 檔案系統，請輸入

```
mount -t ext2 /dev/sda1 /home
```

4 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

```
df -h
```

可用磁碟空間 (df) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或 2G。

5.3 增加 Reiser 檔案系統的大小

在掛接或卸載 ReiserFS 檔案系統時可以增加其大小。

1 開啟終端機主控台，以 root 使用者或同等身分登入。

2 使用以下其中一種方法，增加名為 /dev/sda2 之設備上檔案系統的大小：

- 若要將檔案系統的大小擴充至設備的最大可用大小，請輸入

  ```
  resize_reiserfs /dev/sda2
  ```

  若不指定大小，該指令會將磁碟區增加至分割區的總大小。

- 若要將檔案系統擴充至指定大小，請輸入

  ```
  resize_reiserfs -s size /dev/sda2
  ```

  用所需大小 (以位元組計) 取代 *size*。您也可以指定值的單位，例如 50000K (KB)、250M (MB) 或 2G (GB)。也可以使用加號 (+) 為值加上字首，以指定為目前大小增加的值。例如，以下指令可將 /dev/sda2 上的檔案系統的大小增加 500 MB：

  ```
  resize_reiserfs -s +500M /dev/sda2
  ```

請等候直至完成大小調整，然後再繼續。

3 如果未掛接檔案系統，請立即掛接。

例如，若要在掛接點 /home 為名為 /dev/sda2 的設備掛接 ReiserFS 檔案系統，請輸入

```
mount -t reiserfs /dev/sda2 /home
```

4 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

```
df -h
```

可用磁碟空間 (df) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或2G。

5.4 減少 Ext2 或 Ext3 檔案系統的大小

掛接或卸載 Ext2 與 Ext3 檔案系統時，可調整其大小。

1 開啟終端機主控台，以 root 使用者或同等身分登入。

2 輸入以下指令，減少設備 (如 /dev/sda1) 上檔案系統的大小

```
resize2fs /dev/sda1 <size>
```

使用所需大小的整數值 (以 KB 計) 取代 *size*。(1 KB = 1024 B。)

請等候直至完成大小調整，然後再繼續。

3 如果未掛接檔案系統，請立即掛接。例如，若要在掛接點 /home 為名為 /dev/sda1 的設備掛接 Ext2 檔案系統，請輸入

```
mount -t ext2 /dev/md0 /home
```

4 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

```
df -h
```

可用磁碟空間 (df) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或 2G。

5.5 減少 Reiser 檔案系統的大小

只有在卸載磁碟區後才能減少 Reiser 檔案系統的大小。

1. 開啟終端機主控台，以 root 使用者或同等身分登入。
2. 輸入以下指令卸載設備

   ```
   umount /mnt/point
   ```

   如果您嘗試要減少其大小的分割區包含系統檔案 (例如根 (/) 磁碟區)，僅當系統從可開機的 CD 或磁片開機時，才可以進行卸載操作。
3. 輸入以下指令，減少名為 /dev/sda1 的設備上檔案系統的大小

   ```
   resize_reiserfs -s size /dev/sda2
   ```

   用所需大小(以位元組計)取代 *size*。您也可以指定值的單位，例如50000K (KB)、250M (MB) 或 2G (GB)。您也可以使用減號 (-) 為該值加上字首，以指定為目前大小減少的值。例如，以下指令可將 /dev/md0 上檔案系統的大小減少 500 MB：

   ```
   resize_reiserfs -s -500M /dev/sda2
   ```

   請等候直至完成大小調整，然後再繼續。
4. 輸入以下指令掛接檔案系統

   ```
   mount -t reiserfs /dev/sda2 /mnt/point
   ```
5. 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

   ```
   df -h
   ```

可用磁碟空間 (`df`) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或2G。

6 使用 UUID 掛接設備

本章描述可選擇性使用 UUID 而非設備名稱來識別開機載入程式檔案及 /etc/fstab 檔案中的檔案系統設備。

6.1 使用 udev 命名設備

在 Linux 2.6 和更新版本的核心中，udev 透過永久命名設備，為動態 /dev 目錄提供了一個使用者空間解決方案。做為 HotPlug 系統的一部分，會在設備新增至系統或從系統中移除時執行 udev。

規則清單用來比對特定的設備屬性。udev 規則基礎結構 (定義於 /etc/udev/rules.d 目錄) 為所有磁碟設備提供了固定名稱，不會隨辨識順序或設備使用的連線而改變。udev 工具會檢查核心所建立的每個適當區塊設備，根據特定匯流排、磁碟機類型或檔案系統套用命名規則。如需有關如何定義自己的 udev 規則的資訊，請參閱 *寫入 udev 規則* [http://reactivated.net/writing_udev_rules.html]。

udev 會根據動態核心指定設備節點名稱，維護指向 /dev/disk 目錄中設備的永久符號連結類別，該目錄將進一步分類為 by-id、by-label、by-path 及 by-uuid 子目錄。

注意

除了 udev 以外的其他程式，例如 LVM 或 md，也可能會產生 UUID，但不會在 /dev/disk 中列出。

6.2 瞭解 UUID

UUID (通用唯一識別碼) 是 128 位元的檔案系統編號，在本地系統和其他系統中都是唯一的。它根據系統硬體資訊和時戳 (做為其種子的一部分) 隨機產生。UUID 通常用於唯一標記設備。

- 第 6.2.1 節「使用 UUID 組譯或啟動檔案系統設備」[46頁]
- 第 6.2.2 節「尋找檔案系統設備的 UUID」[47頁]

6.2.1 使用 UUID 組譯或啟動檔案系統設備

UUID 對於分割區始終是唯一的，不會隨其顯示順序和掛接位置而變化。特定 SAN 設備連接到伺服器後，系統分割區會被重新命名並上移為上一個設備。例如，如果在安裝期間將根目錄 (/) 指定給 /dev/sda1，它可能會在連接 SAN 後被指定給 /dev/sdg1。解決此問題的一個方法是在開機載入程式和開機設備的 /etc/fstab 檔案中使用 UUID。

無論設備掛接在哪里，廠商為磁碟機指定的設備 ID 都不會改變，所以在開機時始終可以找到該設備。UUID 是檔案系統的內容，在磁碟重新格式化時會發生變更。在開機載入程式檔案中，您通常要在系統開機時指定設備的掛接位置 (例如 /dev/sda1)。開機載入程式還可以透過設備的 UUID 與管理員指定的磁碟區標籤來掛接設備。但是，如果您使用標籤和檔案位置，就不可在掛接分割區後變更標籤名稱。

您可以將 UUID 用做組譯與啟動軟體 RAID 設備的準則。建立 RAID 後，md 驅動程式會為設備產生一個 UUID，並將該值儲存在 md 超級區塊中。

6.2.2 尋找檔案系統設備的 UUID

可以在 /dev/disk/by-uuid 目錄中找到任何區塊設備的 UUID。例如，如下所示的 UUID：

```
e014e482-1c2d-4d09-84ec-61b3aefde77a
```

6.3 使用開機載入程式與 /etc/fstab 檔案中的 UUID (x86)

安裝後，您可以選擇性使用下列程序，在開機載入程式與 /etc/fstab 檔案中為 x86 系統設定系統設備的 UUID。

開始之前，請先複製 /boot/grub/menu.1st 檔案和 /etc/fstab 檔案。

1. 在未連接 SAN 設備的情況下，安裝 SUSE® Linux Enterprise Server for x86。
2. 安裝之後，將系統開機。
3. 以 root 使用者或同等身分開啟終端機主控台。
4. 導覽到 /dev/disk/by-uuid 目錄找到安裝 /boot、/root 及 swap 之設備的 UUID。

   4a 在終端機主控台提示符下，輸入

   ```
   cd /dev/disk/by-uuid
   ```

   4b 輸入以下指令以列出所有分割區

   ```
   ll
   ```

   4c 尋找 UUID，例如

   ```
   e014e482-1c2d-4d09-84ec-61b3aefde77a –> /dev/sda1
   ```

5 使用 YaST2 中的「開機載入程式」選項或使用文字編輯器編輯 /boot/grub/menu.1st 檔案。

例如，將

```
kernel /boot/vmlinuz root=/dev/sda1
```

變更為

```
kernel /boot/vmlinuz
root=/dev/disk/by-uuid/e014e482-1c2d-4d09-84ec-61b3aefde77a
```

重要

如果出錯，您可以在未連接 SAN 的情況下開機伺服器，並根據 /boot/grub/menu.1st 檔案的備份副本來更正錯誤。

如果使用 YaST 中的「開機載入程式」選項，在您變更值時會錯誤地將一些重複的行新增至開機載入程式檔案。使用編輯器移除以下重複的行：

```
color white/blue black/light-gray

default 0

timeout 8

gfxmenu (sd0,1)/boot/message
```

使用 YaST 變更設備掛接到根目錄 (/) 的方式時 (例如使用 UUID 或標籤)，需要再次儲存開機載入程式組態以使變更對於開機載入程式生效。

6 以 root 使用者或同等身分執行以下其中一項操作，將 UUID 存放在 /etc/fstab 檔案中：

- 開啟 YaST 至「*系統*」>「*磁碟分割程式*」，選取所需設備，然後修改「*Fstab 選項*」。
- 編輯 /etc/fstab 檔案將系統設備從位置修改為 UUID。

例如，如果根 (/) 磁碟區中有 /dev/sda1 的設備路徑，且其 UUID 為 e014e482-1c2d-4d09-84ec-61b3aefde77a，則將行項目從

```
/dev/sda1   /            reiserfs   acl,user_xattr        1 1
```

變更為

```
UUID=e014e482-1c2d-4d09-84ec-61b3aefde77a   /  reiserfs
acl,user_xattr        1 1
```

重要

檔案中不要留有零散字元或空格。

6.4 使用開機載入程式與 /etc/fstab 檔案中的 UUID (IA64)

安裝之後，使用以下程序在開機載入程式與 /etc/fstab 檔案中為 IA64 系統設定系統設備的 UUID。IA64 使用 EFI BIOS。其檔案系統組態檔案為 /boot/efi/SuSE/elilo.conf，而非 /etc/fstab。

開始之前，請先複製 /boot/efi/SuSE/elilo.conf 檔案。

1. 在未連接 SAN 設備的情況下，安裝 SUSE Linux Enterprise Server for IA64。
2. 安裝之後，將系統開機。
3. 以 root 使用者或同等身分開啟終端機主控台。
4. 導覽到 /dev/disk/by-uuid 目錄找到安裝 /boot、/root 及 swap 之設備的 UUID。

   4a 在終端機主控台提示符下，輸入

   ```
   cd /dev/disk/by-uuid
   ```

4b 輸入以下指令以列出所有分割區

```
ll
```

4c 尋找 UUID，例如

```
e014e482-1c2d-4d09-84ec-61b3aefde77a –> /dev/sda1
```

5 使用 YaST2 中的「開機載入程式」選項編輯開機載入程式檔案。

例如，將

```
root=/dev/sda1
```

變更為

```
root=/dev/disk/by-uuid/e014e482-1c2d-4d09-84ec-61b3aefde77a
```

6 編輯 `/boot/efi/SuSE/elilo.conf` 檔案將系統設備從位置修改為 UUID。

例如，將

```
/dev/sda1   /   reiserfs   acl,user_xattr        1 1
```

變更為

```
UUID=e014e482-1c2d-4d09-84ec-61b3aefde77a   /  reiserfs   acl,user_xattr
        1 1
```

重要

檔案中不要留有零散字元或空格。

6.5 其他資訊

有關使用 udev(8) 管理設備的詳細資訊，請參閱《*SUSE® Linux Enterprise Server 11 安裝與管理指南*》中的「使用 udev 進行動態核心設備管理」[http://www.novell.com/documentation/sles11/book_sle_admin/data/cha_udev.html]。

有關 udev(8) 指令的詳細資訊，請參閱其線上文件。在終端機主控台提示符下輸入以下指令：

```
man 8 udev
```

7

管理設備的多重路徑 I/O

本章描述如何管理伺服器和區塊儲存設備間之多重路徑的容錯移轉和路徑負載平衡。

- 第 7.14 節「解決擱置的 I/O」 [108頁]
- 第 7.15 節「其他資訊」 [109頁]
- 第 7.16 節「還有什麼功能？」 [110頁]

7.1 瞭解多重路徑

- 第 7.1.1 節「什麼是多重路徑？」 [54頁]
- 第 7.1.2 節「多重路徑的優點」 [54頁]

7.1.1 什麼是多重路徑？

多重路徑是指伺服器與相同實體或邏輯區塊儲存設備在伺服器中的主機匯流排介面卡與設備的儲存控制器之間、跨多重實體路徑進行通訊的能力，通常是在光纖通道 (FC) 或 iSCSI SAN 環境中進行。若有多個通道可用，您還可以與直接連接的儲存建立多個連接。

7.1.2 多重路徑的優點

Linux 多重路徑提供了連線容錯功能並可在各主動連線之間實現負載平衡。設定並執行多重路徑時，會自動隔離並識別設備連線失敗，並將 I/O 重新路由至替代連線。

常見的連線問題包括介面卡、纜線或控制器錯誤。為設備設定多重路徑 I/O 後，多重路徑驅動程式將監控設備間的主動連線。當多重路徑驅動程式偵測到主動路徑的 I/O 錯誤時，會將流量容錯移轉至設備的指定次要路徑。偏好的路徑復原後，控制將返回至該路徑。

7.2 規劃多重路徑

- 第 7.2.1 節「多重路徑準則」 [55頁]
- 第 7.2.2 節「使用多重路徑設備的 By-ID 名稱」 [57頁]

- 第 7.2.3 節「在多重路徑設備上使用 LVM2」[57頁]
- 第 7.2.4 節「將 mdadm 用於多重路徑設備」[59頁]
- 第 7.2.5 節「將 --noflush 用於多重路徑設備」[59頁]
- 第 7.2.6 節「根設備為多重路徑設備時的 SAN 逾時設定」[59頁]
- 第 7.2.7 節「分割多重路徑設備」[60頁]
- 第 7.2.8 節「多重路徑 I/O 的受支援結構」[61頁]
- 第 7.2.9 節「多重路徑的受支援儲存陣列」[61頁]

7.2.1 多重路徑準則

使用本節中的準則規劃多重路徑 I/O 解決方案。

- 章節「先決條件」[55頁]
- 章節「廠商提供的多重路徑解決方案」[56頁]
- 章節「磁碟管理任務」[56頁]
- 章節「軟體 RAID」[56頁]
- 章節「高可用性解決方案」[56頁]
- 章節「磁碟區管理器」[57頁]
- 章節「虛擬化環境」[57頁]

先決條件

- 多重路徑在設備層級進行管理。
- 您用於多重路徑設備的儲存陣列必須支援多重路徑。如需詳細資訊，請參閱 第 7.2.9 節「多重路徑的受支援儲存陣列」[61頁]。

- 只有在伺服器中的主機匯流排配接器和區塊儲存設備的主機匯流排控制器之間存在多重實體路徑時，才需要設定多重路徑。應按伺服器所見為邏輯設備設定多重路徑。

廠商提供的多重路徑解決方案

對於某些儲存陣列，廠商提供了自己的多重路徑軟體來管理陣列之實體和邏輯設備的多重路徑。在這種情況下，您應根據廠商的指示設定那些設備的多重路徑。

磁碟管理任務

請先執行以下磁碟管理任務，再嘗試為具有多路徑的實體或邏輯設備設定多重路徑：

- 使用協力廠商工具將實體磁碟分割為數個較小的邏輯磁碟。
- 使用協力廠商工具分割實體磁碟或邏輯磁碟。如果您變更正在執行系統中的分割區，則設備對應程式多重路徑 (DM-MP) 模組將不會自動偵測並反映這些變更。DM-MPIO 必須重新啟始化，這通常需要重新開機。
- 使用協力廠商 SAN 陣列管理工具建立並設定硬體 RAID 設備。
- 使用協力廠商 SAN 陣列管理工具建立邏輯設備，例如 LUN。指定陣列所支援的邏輯設備類型由陣列廠商決定。

軟體 RAID

Linux 軟體 RAID 管理軟體在多重路徑的最上層執行。對於每個具有多重 I/O 路徑的設備和要用於軟體 RAID 的設備來說，您必須先設定多重路徑的設備，然後才能嘗試建立軟體 RAID 設備。無法自動探查多重路徑設備。軟體 RAID 無法知曉多重路徑管理是否於後台執行。

高可用性解決方案

用於進行叢集的高可用性解決方案通常在多重路徑伺服器的最上層執行。例如，用於透過多重路徑最上層上執行的 LAN 鏡像複製設備的分散式複製區塊設備

(DRBD) 高可用性解決方案。對於每個具有多重 I/O 路徑的設備和要用於 DRDB 解決方案的設備來說，您必須先設定多重路徑的設備，然後才能設定 DRDB。

磁碟區管理器

磁碟區管理 (如 LVM2 和 EVMS) 在多重路徑最上層執行。必須先設定設備的多重路徑，然後才能使用 LVM2 或 EVMS 在其上建立區段管理員和檔案系統。

虛擬化環境

在虛擬化環境中使用多重路徑時，將在主機伺服器環境中控制多重路徑。請先設定設備的多重路徑，然後才能將其指派給虛擬訪客機器。

7.2.2 使用多重路徑設備的 By-ID 名稱

如果您要直接使用整個 LUN (例如，若您使用 SAN 功能分割您的儲存區)，可為 `mkfs`、`fstab`、您的應用程式等使用 `/dev/disk/by-id/xxx` 名稱。

若 `/etc/multipath.conf` 檔案中啟用了使用者易記名稱選項，則您可以使用 `/dev/disk/by-id/dm-uuid-.*-mpath-.*` 設備名稱，因為此名稱是設備 ID 的別名。如需更多資訊，請參閱章節「在 /etc/multipath.conf 中設定易記名稱或別名」 [76頁]。

7.2.3 在多重路徑設備上使用 LVM2

LVM2 預設不辨識多重路徑設備。若要使 LVM2 將多重路徑設備辨識為可用的實體磁碟區，您必須修改 `/etc/lvm/lvm.conf`。重要的是，要修改為使其不掃描與使用實體路徑，而只透過多重路徑 I/O 層存取多重路徑 I/O 儲存區。若您使用的是使用者易記名稱，請務必指定路徑，以便在設定多重路徑之後僅掃描設備 (`/dev/disk/by-id/dm-uuid-.*-mpath-.*`) 的設備對應程式名稱。

若要修改多重路徑的 `/etc/lvm/lvm.conf`，請使用以下方法：

1 在文字編輯器中開啟 `/etc/lvm/lvm.conf` 檔案。

如果 `/etc/lvm/lvm.conf` 不存在，您可以在終端機主控台提示符處輸入以下指令，根據目前的 LVM 組態建立該檔案：

```
lvm dumpconfig > /etc/lvm/lvm.conf
```

2 變更 /etc/lvm/lvm.conf 中的 filter 和 types 項目，如下所示：

```
filter = [ "a|/dev/disk/by-id/.*|", "r|.*|" ]
types = [ "device-mapper", 1 ]
```

這樣可讓 LVM2 只掃描 by-id 路徑，而拒絕其他所有項目。

若您使用的是使用者易記名稱，請按照以下方式指定路徑，以便在設定多重路徑之後僅掃描設備對應程式名稱：

```
filter = [ "a|/dev/disk/by-id/dm-uuid-.*-mpath-.*|", "r|.*|" ]
```

3 如果您也在非多重路徑設備上使用 LVM2，請對 filter 和 types 進行必要的調整以符合您的設定。否則，在修改 lvm.conf 檔案以啟用多重路徑後，使用 pvscan 將無法顯示其他 LVM 設備。

您只想將使用 LVM 設定的設備包含到 LVM 快取中，因此請務必明確指定過濾器應包含哪些其他非多重路徑設備。

例如，如果本地磁碟為 /dev/sda，且所有 SAN 設備都為 /dev/sdb 及以上，則在過濾器中按如下方式指定本地路徑和多重路徑：

```
filter = [ "a|/dev/sda.*|", "a|/dev/disk/by-id/.*|", "r|.*|" ]
types = [ "device-mapper", 253 ]
```

4 儲存檔案。

5 將 dm-multipath 新增至 /etc/sysconfig/kernel:INITRD_MODULES。

6 新增一個 initrd 以確保設備對應程式多重路徑服務已使用變更後的設定載入。在終端機主控台提示符處輸入以下指令：

```
mkinitrd –f multipath
```

7 重新開機伺服器以套用變更。

7.2.4 將 mdadm 用於多重路徑設備

mdadm 工具要求以 ID 而非設備節點路徑存取設備。因此，應使用如下指令設定 /etc/mdadm.conf 中的 DEVICE 項目：

```
DEVICE /dev/disk/by-id/*
```

若您使用的是使用者易記名稱，請按照以下方式指定路徑，以便在設定多重路徑之後僅掃描設備對應程式名稱：

```
DEVICE /dev/disk/by-id/dm-uuid-.*-mpath-.*
```

7.2.5 將 --noflush 用於多重路徑設備

在多重路徑設備上執行時，應始終使用 --noflush 選項。

例如，在執行表格重新載入的程序檔中，應使用 --noflush 選項進行恢復，以確保所有重要 I/O 不會被衝洗，因為您需要多重路徑拓樸資訊。

```
load
resume --noflush
```

7.2.6 根設備為多重路徑設備時的 SAN 逾時設定

所有路徑都已失敗並已從系統移除時，多重路徑設備上包含根目錄 (/) 的系統可能會停止，因為系統會收到儲存子系統 (例如光纖通道儲存陣列) 發出的 dev_loss_tmo 逾時通知。

如果系統設備設定了多個路徑，且多重路徑 no_path_retry 設定處於啟用狀態，您應相應地修改儲存子系統的 dev_loss_tmo 設定，以確保在所有路徑失效的情況下不會移除任何設備。強烈建議您將 dev_loss_tmo 的值設為等於或大於多重路徑中 no_path_retry 設定的值。

建議按如下方式設定儲存子系統的 dev_los_tmo：

```
<dev_loss_tmo> = <no_path_retry> * <polling_interval>
```

其中，以下定義適用於多重路徑值：

- `no_path_retry` 定義多重路徑 I/O 嘗試多少次後路徑視為遺失並停止將 I/O 排入佇列。
- `polling_interval` 是執行路徑檢查的時間間隔 (以秒為單位)。

每個多重路徑值都應在 `/etc/multipath.conf` 組態檔案中設定。如需更多資訊，請參閱第 7.4.5 節「建立並設定 /etc/multipath.conf 檔案」[74頁]。

7.2.7 分割多重路徑設備

如果正在升級，則對多重路徑設備處理方式的行為變更可能會影響您的組態。

- 章節「SUSE Linux Enterprise Server 11」[60頁]
- 章節「SUSE Linux Enterprise Server 10」[60頁]
- 章節「SUSE Linux Enterprise Server 9」[61頁]

SUSE Linux Enterprise Server 11

在 SUSE® Linux Enterprise Server 11 中，當啟動多重路徑時，預設多重路徑設定依賴 `udev` 來覆寫 `/dev/disk/by-id` 目錄中的現存符號連結。在您啟動多重路徑之前，該連結透過使用設備的 `scsi-xxx` 名稱來指向 SCSI 設備。當多重路徑正在執行時，該符號連結則透過使用設備的 `dm-uuid-xxx` 名稱來指向設備。這樣可確保不論多重路徑是否啟動，`/dev/disk/by-id` 路徑中的符號連結始終指向同一設備。由於 `lvm.conf` 和 `md.conf` 等組態檔案會自動指向正確的設備，所以無須對其進行修改。

SUSE Linux Enterprise Server 10

在 SUSE Linux Enterprise Server 10 中，使用 `kpartx` 軟體在 `/etc/init.d/boot.multipath` 中將符號連結新增至所有新建立磁碟分割區之 `multipath.conf` 組態檔案的 `/dev/dm-*` 行中，且不需要重新開機。這樣會觸發 `udevd`

以填寫 /dev/disk/by-* 符號連結。其主要優點是您可以使用新參數呼叫 kpartx，而無需重新開機伺服器。

SUSE Linux Enterprise Server 9

在 SUSE Linux Enterprise Server 9 中，無法分割多重路徑 I/O 設備自身。如果基礎實體設備已進行分割，則多重路徑 I/O 設備將反映這些分割區，且該層會提供 /dev/disk/by-id/<name>p1 ... pN 設備，讓您可透過多重路徑 I/O 層存取分割區。因此，需要在啟用多重路徑 I/O 之前分割設備。如果變更正在執行之系統中的磁碟分割，則 DM-MPIO 不會自動偵測並反映這些變更。必須重新啟始化設備，通常需要重新開機。

7.2.8 多重路徑 I/O 的受支援結構

多重路徑驅動程式與工具支援以下七種受支援的處理器結構：IA32、AMD64/EM64T、IPF/IA64、p-Series (32 位元 與 64 位元)、z-Series (31 位元與 64 位元)。

7.2.9 多重路徑的受支援儲存陣列

多重路徑驅動程式與工具支援大部分儲存陣列。儲存多重路徑設備的儲存陣列必須支援多重路徑，這樣才能使用多重路徑驅動程式與工具。一些儲存陣列廠商會提供其自己的多重路徑管理工具。請參閱廠商的硬體文件以決定所需設定。

- 章節「為多重路徑自動偵測儲存陣列」[61頁]
- 章節「經測試支援多重路徑的儲存陣列」[63頁]
- 章節「需要特定硬體處理器的儲存陣列」[63頁]

為多重路徑自動偵測儲存陣列

multipath-tools 套件會自動偵測以下儲存陣列：

3PARdata VV
Compaq* HSV110
Compaq MSA1000

DDN SAN MultiDirector
DEC* HSG80
EMC* CLARiiON* CX
EMC Symmetrix*
FSC CentricStor*
Hewlett Packard* (HP*) A6189A
HP HSV110
HP HSV210
HP Open
Hitachi* DF400
Hitachi DF500
Hitachi DF600
IBM 3542
IBM ProFibre 4000R
NetApp*
SGI* TP9100
SGI TP9300
SGI TP9400
SGI TP9500
STK OPENstorage DS280
Sun* StorEdge 3510
Sun T4

一般說來，大多數其他儲存陣列應該都可使用。自動偵測到儲存陣列後，會套用多重路徑的預設設定。如果您需要非預設設定，則必須手動建立並設定 `/etc/multipath.conf` 檔案。如需更多資訊，請參閱第 7.4.5 節「建立並設定 /etc/multipath.conf 檔案」[74頁]。

對支援多重路徑的 IBM zSeries* 設備進行的測試表明，應將 dev_loss_tmo 參數設定為 90 秒，將 fast_io_fail_tmo 參數設定為 5 秒。若您使用的是 zSeries 設備，則必須手動建立並設定 `/etc/multipath.conf` 檔案來指定這些值。如需更多資訊，請參閱章節「在 /etc/multipath.conf 中為 zSeries 設定預設設定值」[79頁]。

未自動偵測到的硬體需要在 `/etc/multipath.conf` 檔案的 `DEVICES` 區段中設定適當項目。在此狀況下，您必須手動建立並設定組態檔案。如需更多資訊，請參閱第 7.4.5 節「建立並設定 /etc/multipath.conf 檔案」[74頁]。

請考慮以下警告：

- 並非所有自動偵測到的儲存陣列都已在 SUSE Linux Enterprise Server 上經過測試。如需更多資訊，請參閱章節「經測試支援多重路徑的儲存陣列」[63頁]。
- 某些儲存陣列可能需要特定的硬體處理器。硬體處理器是在切換路徑群組與處理 I/O 錯誤時執行硬體特定動作的核心模組。如需更多資訊，請參閱章節「需要特定硬體處理器的儲存陣列」[63頁]。
- 修改 `/etc/multipath.conf` 檔案後，必須執行 `mkinitrd` 在系統中重新建立 INITRD，然後重新開機以使變更生效。

經測試支援多重路徑的儲存陣列

下列儲存陣列已經過 SUSE Linux Enterprise Server 測試：

EMC
Hitachi
Hewlett-Packard/Compaq
IBM
NetApp
SGI

大多數其他廠商提供的儲存陣列應該也可使用。請參閱廠商文件獲取指導原則。如需 `multipath-tools` 套件可辨識的預設儲存陣列清單，請參閱章節「為多重路徑自動偵測儲存陣列」[61頁]。

需要特定硬體處理器的儲存陣列

從一個路徑到另一個路徑的容錯移轉時需要特殊指令的儲存陣列，或需要特殊非標準錯誤處理的儲存陣列，可能都需要額外的延伸支援。因此，設備對應程式多重路徑服務已配備用於硬體處理器的插入程式。例如，已針對 EMC CLARiiON CX 陣列系列提供一個此類處理器。

重要

請參閱硬體廠商文件，以確定是否必須安裝硬體處理器以用於設備對應程式多重路徑。

multipath -t 指令可顯示需要使用特定硬體處理器進行特殊處理之儲存陣列的內部表格。所顯示的清單不是受支援儲存陣列的完整清單。它列出的僅僅是需要特殊處理的陣列，以及multipath-tools 開發人員在開發該工具期間有存取權限的陣列。

重要

真正支援主動/主動多重路徑的陣列不需要特殊處理，因此 multipath -t 指令不會列出此類陣列。

multipath -t 表格中的清單內容並不表示已在該特定硬體上進行 SUSE Linux Enterprise Server 測試。如需經過測試的儲存陣列清單，請參閱章節「經測試支援多重路徑的儲存陣列」 [63頁]。

7.3 多重路徑管理工具

SUSE Linux Enterprise Server 10 及更高版本中的多重路徑支援以 Linux 2.6 核心的設備對應程式多重路徑模組及multipath-tools 使用者空間套件為基礎。使用多設備管理 (MDADM，mdadm) 公用程式可以檢視多重路徑設備的狀態。

- 第 7.3.1 節「設備對應程式多重路徑模組」 [64頁]
- 第 7.3.2 節「多重路徑 I/O 管理工具」 [67頁]
- 第 7.3.3 節「對多重路徑設備使用 MDADM」 [68頁]
- 第 7.3.4 節「Linux multipath(8) 指令」 [69頁]

7.3.1 設備對應程式多重路徑模組

設備對應程式多重路徑 (DM-MP) 模組為 Linux 提供了多重路徑功能。DM-MPIO 是在 SUSE Linux Enterprise Server 11 中進行多重路徑的較好解決方案。它是該產品提供的唯一一個多重路徑選項，Novell® 與 SUSE 均對其完全支援。

DM-MPIO可自動設定多重路徑子系統的多種設定。每個設備最多可設定8個路徑。模組支援主動/被動 (一個主動路徑，其他均為被動路徑) 或主動/主動 (所有路徑均為主動路徑，使用輪替式負載平衡) 組態。

DM-MPIO 框架可以採用以下兩種方式進行擴充：

- 使用特定硬體處理器。如需更多資訊，請參閱章節「需要特定硬體處理器的儲存陣列」[63頁]。
- 使用更精密的負載平衡演算法取代輪替式演算法

DM-MPIO的使用者空間元件會自動探查路徑並進行分組，還會自動重新測試路徑，以便在先前失敗的路徑恢復正常時，能自動重新啟用該路徑。這樣會將管理員對生產環境的關注需求降到最低。

DM-MPIO防範的是設備路徑中的失敗，而不是設備本身的失敗。如果其中一個主動路徑遺失(例如網路卡損壞或光纖纜線被移開)，則I/O會重新導向到其餘的路徑。如果組態為主動/被動模式，則路徑會容錯移轉到其中一個被動路徑。如果您使用的是輪替式負載平衡組態，則流量會分流到其餘的正常路徑。如果所有主動路徑都失敗，則必須喚醒非主動的次要路徑，因此需經過大約30秒的延遲之後才會開始容錯移轉。

如果磁碟陣列有多個儲存處理器，請確保SAN交換器已連線到您要存取的LUN所屬的儲存處理器。在大多數磁碟陣列中，所有LUN都屬於兩個儲存處理器，因此兩個連線都處於主動模式。

注意

在有些磁碟陣列中，儲存陣列透過儲存處理器來管理流量，因此每次只會顯示一個儲存處理器。一個處理器為主動，另外一個為被動，直到發生了失敗。如果您連接到錯誤的儲存處理器 (路徑為被動的處理器)，則可能找不到所需的 LUN，或雖然找到了 LUN，但在嘗試存取時會發生錯誤。

表格 7.1 *儲存陣列的多重路徑 I/O 功能*

儲存陣列的功能	描述
主動/被動控制器	只有一個控制器處於主動狀態，並服務所有的 LUN。次要控制器處於待機狀態。次要控制器也會將 LUN 提供給多重路徑元件，以便作業系統瞭解備援路徑的情況。如果主要控制器失敗，則由次要控制器接管其工作並服務所有 LUN。

儲存陣列的功能	描述
	在某些陣列中，可將 LUN 指定給不同的控制器。將指定的 LUN 指定給要做為其主動控制器的控制器。每次當一個控制器為任何指定的 LUN 執行磁碟 I/O 時，另一個控制器就為該 LUN 保持待機狀態。另外的那個控制器也會提供路徑，但無法執行磁碟I/O。使用該LUN的伺服器會連接到LUN的指定控制器。如果一組LUN的主要控制器失敗，則由次要控制器接管其工作並服務所有 LUN。
主動/主動控制器	兩個控制器共享所有 LUN 的負載，可以處理任何指定LUN的磁碟I/O。如果一個控制器失敗，另一個控制器便會自動處理所有流量。
載入平衡	設備對應程式多重路徑驅動程式會自動對通過所有主動路徑的流量進行負載平衡控制。
控制器容錯移轉	主動控制器容錯移轉到被動 (或待機) 控制器時，設備對應程式多重路徑驅動程式會自動啟動主機與待機之間的路徑，使其成為主要路徑。
開機/根設備支援	SUSE Linux Enterprise Server 10 及更高版本提供對根 (/) 設備的多重路徑支援。主機伺服器必須連接到開機設備目前的主動控制器與儲存處理器。 SUSE Linux Enterprise Server 11 及更高版本提供了對 `/boot` 設備的多重路徑支援。

設備對應程式多重路徑會偵測每個路徑，查看有無多重路徑設備做為獨立的 SCSI 設備。SCSI 設備名稱採用的格式為 `/dev/sd`*N*，其中 *N* 是為設備自動產生的字母，從 a 開始，並隨著設備的建立依序發佈，例如 `/dev/sda`、`/dev/sdb`，依此類推。如果設備數目超過 26 個，則字母將會重複，這樣，`/dev/sdz` 之後的設備將命名為 `/dev/sdaa`、`/dev/sdab`，依此類推。

如果有多個路徑未自動偵測到，則您可以在 `/etc/multipath.conf` 檔案中手動設定。在您建立並設定 `multipath.conf` 檔案之前，該檔案並不存在。如需更多資訊，請參閱第 7.4.5 節「建立並設定 /etc/multipath.conf 檔案」[74頁]。

7.3.2 多重路徑 I/O 管理工具

multipath-tools 使用者空間套件會自動探查路徑，並對其進行分組。它會週期性地自動測試路徑，這樣，先前失敗的路徑在恢復正常後，便可自動重新啟用。這樣會將管理員對生產環境的關注需求降到最低。

表格 7.2　multipath-tools 套件中的工具

工具	描述
multipath	對系統進行掃描，找出多重路徑設備並進行組合。
multipathd	等待映射事件，然後執行 multipath。
devmap-name	為 udev 提供有意義的設備名稱以進行設備映射 (devmaps)。
kpartx	將線性 devmaps 映射到多重路徑設備中的分割區，這樣能夠建立對設備中各分割區的多重路徑監控。

對於不同的伺服器架構，套件的檔案清單可能會有所不同。有關 multipath-tools 套件中包含的檔案清單，請造訪「*SUSE Linux Enterprise Server Technical Specifications*」>「*Package Descriptions*」網頁 [http://www.novell.com/products/server/techspecs.html]，找到您的架構並選取「*Packages Sorted by Name*」，然後搜尋「multipath-tools」以找到適用於該架構的套件清單。

您也可以透過查詢套件本身來確定 RPM 檔案的檔案清單：使用 rpm -ql 或 rpm -qpl 指令選項。

- 若要查詢已安裝的套件，請輸入

```
rpm -ql <package_name>
```

- 若要查詢未安裝的套件，請輸入

```
rpm -qpl <URL_or_path_to_package>
```

若要檢查是否已安裝 multipath-tools 套件，請執行以下操作：

- 在終端機主控台提示符下輸入以下指令：

  ```
  rpm -q multipath-tools
  ```

  如果已安裝，回應內容會重複該套件名稱並提供版本資訊，例如：

  ```
  multipath-tools-04.7-34.23
  ```

  如果未安裝，則回應內容為：

  ```
  package multipath-tools is not installed
  ```

7.3.3 對多重路徑設備使用 MDADM

Udev 是預設的設備處理器，該系統可透過 Worldwide ID (而不是設備節點名稱) 自動識別設備。這解決了 MDADM 與 LVM 的先前版本中組態檔案 (mdadm.conf 與 lvm.conf) 無法正確辨識多重路徑設備的問題。

與 LVM2 一樣，MDADM 也要求透過 ID 而非設備節點路徑存取設備。因此，應使用如下指令設定 /etc/mdadm.conf 中的 DEVICE 項目：

```
DEVICE /dev/disk/by-id/*
```

若您使用的是使用者易記名稱，請按照以下方式指定路徑，以便在設定多重路徑之後僅掃描設備對應程式名稱：

```
DEVICE /dev/disk/by-id/dm-uuid-.*-mpath-.*
```

若要驗證是否已安裝 MDADM：

- 在終端機主控台提示符處輸入以下指令，確定是否已安裝 mdadm 套件：

  ```
  rpm -q mdadm
  ```

  如果已安裝，回應內容會重複該套件名稱並提供版本資訊。例如：

  ```
  mdadm-2.6-0.11
  ```

如果未安裝，則回應內容為：

```
package mdadm is not installed
```

如需修改 `/etc/lvm/lvm.conf` 檔案的資訊，請參閱第 7.2.3 節「在多重路徑設備上使用 LVM2」[57頁]。

7.3.4 Linux multipath(8) 指令

使用 Linux `multipath(8)` 指令可以設定並管理多重路徑設備。

`multipath(8)` 指令的一般語法為：

```
multipath [-v verbosity] [-d] [-h|-l|-ll|-f|-F] [-p failover | multibus |
group_by_serial | group_by_prio| group_by_node_name ]
```

一般範例

multipath
設定所有多重路徑設備。

multipath *設備名稱*
設定特定的多重路徑設備。

使用 */dev/sdb* (顯示於 udev 的 $DEVNAME 變數中) 等設備節點名稱取代 `devicename`，或採用 major:minor 格式。

multipath -f
選擇性隱藏多重路徑映射及其映射設備的分割區。

multipath -d
顯示潛在的多重路徑設備，但不建立任何設備，也不更新設備映射 (試執行)。

multipath -v2 -d
設定多重路徑設備並顯示其多重路徑映射資訊。multipath -v2 -d 中的 -v2 選項只顯示本地磁碟。

multipath -v2 *設備名稱*
設定某個特定的多重路徑設備並顯示其多重路徑映射資訊。使用-v2選項可僅顯示本地磁碟。

multipath -v3 -d
設定多重路徑設備並顯示其多重路徑映射資訊。使用-v3選項會顯示完整路徑清單。

multipath -v3 *設備名稱*
設定某個特定的多重路徑設備並顯示其多重路徑映射資訊。使用-v3選項會顯示完整路徑清單。

multipath -ll
顯示所有多重路徑設備的狀態。

multipath -ll *設備名稱*
顯示指定多重路徑設備的狀態。

multipath -F
衝洗所有未使用的多重路徑設備映射。這會取消解析多個路徑，但不會刪除設備。

multipath -F *設備名稱*
衝洗指定多重路徑設備之未使用的多重路徑設備映射。這會取消解析多個路徑，但不會刪除該設備。

multipath -p [failover | multibus | group_by_serial | group_by_prio | group_by_node_name]
透過指定表格 7.3「multipath -p 指令的群組規則選項」[70頁] 中所述的其中一個群組規則選項，來設定群組規則：

表格 7.3 *multipath -p 指令的群組規則選項*

規則選項	描述
failover	每個優先程度群組對應一個路徑。每次只能使用一個路徑。
multibus	所有路徑都在一個優先程度群組中。

規則選項	描述
group_by_serial	每個偵測到的 SCSI 序號 (控制器節點全球號碼) 對應一個優先程度群組。
group_by_prio	每個路徑優先程度值對應一個優先程度群組。優先程度相同的路徑位於同一個優先程度群組中。優先程度由註標程式決定，在 `/etc/multipath.conf` 組態檔案中指定為 global、per-controller 或 per-multipath 選項。
group_by_node_name	每個目標節點名稱對應一個優先程度群組。目標節點名稱取自 `/sys/class/fc_transport/target*/node_name` 位置。

7.4 設定系統以進行多重路徑

- 第 7.4.1 節「準備 SAN 設備以進行多重路徑」[71頁]
- 第 7.4.2 節「對多重路徑設備進行磁碟分割」[72頁]
- 第 7.4.3 節「設定伺服器以進行多重路徑」[73頁]
- 第 7.4.4 節「新增 multipathd 至開機順序」[74頁]
- 第 7.4.5 節「建立並設定 /etc/multipath.conf 檔案」[74頁]

7.4.1 準備 SAN 設備以進行多重路徑

在為SAN設備設定多重路徑I/O之前，請視需要執行以下步驟準備SAN設備：

- 使用廠商工具對 SAN 進行設定並分區。
- 使用廠商工具設定儲存陣列中主機 LUN 的許可權。

- 安裝 Linux HBA 驅動程式模組。安裝模組時，驅動程式會自動掃描 HBA 以探查任何擁有存取主機的許可權的 SAN 設備。驅動程式會將這些設備提交至主機以供稍後設定之用。

  注意

  請確保您使用的 HBA 驅動程式沒有啟用本機多重路徑。

  如需更多詳細資料，請參閱廠商的特定指示。

- 載入驅動程式模組後，探查指定給特定陣列 LUN 或分割區的設備節點。
- 如果 SAN 設備將做為伺服器上的根設備使用，請依第 7.2.6 節「根設備為多重路徑設備時的 SAN 逾時設定」[59頁] 中所述修改該設備的逾時設定。

如果 HBA 驅動程式無法探查到 LUN，可使用 `lsscsi` 來檢查作業系統是否能正常探查到 SCSI 設備。如果 HBA 驅動程式無法探查到 LUN，請檢查 SAN 的分區設定。尤其是要檢查 LUN 遮罩是否處於使用中，且 LUN 是否正確地指定給伺服器。

如果 HBA 驅動程式可以探查到 LUN，但不存在相應的區塊設備，則還需要其他核心參數來變更 SCSI 設備的掃描行為，如指出 LUN 沒有連續編號。如需資訊，請參閱 Novell 支援知識庫中的 *Options for SCSI Device Scanning (SCSI 設備掃描的選項)* [http://support.novell.com/techcenter/sdb/en/2005/06/drahn_scsi_scanning.html]。

7.4.2 對多重路徑設備進行磁碟分割

不建議對含有多個路徑的設備進行磁碟分割，但這項操作是受支援的。

- 章節「SUSE Linux Enterprise Server 10」[72頁]
- 章節「SUSE Linux Enterprise Server 9」[73頁]

SUSE Linux Enterprise Server 10

在 SUSE Linux Enterprise Server 10 中，可以使用 `kpartx` 工具在不重新開機的情況下在多重路徑設備中建立分割區。在嘗試使用 YaST2 中的磁碟分割程式功能或使用協力廠商磁碟分割工具來設定多重路徑之前，也可以對設備進行分割。

SUSE Linux Enterprise Server 9

在 SUSE Linux Enterprise Server 9 中，如果要分割設備，應在嘗試使用 YaST2 中的磁碟分割程式功能或使用協力廠商磁碟分割工具來設定多重路徑之前，先設定其分割區。這是必要動作，因為系統不支援對現有多重路徑設備進行分割。若做此嘗試，對多重路徑設備的磁碟分割操作將會失敗。

如果為設備設定了分割區，則 DM-MPIO 會自動辨識這些分割區，並以在設備的 ID 中附加 p1 至 p*n* 的方式來標識它們，例如

`/dev/disk/by-id/26353900f02796769p1`

若要分割多重路徑設備，必須停用DM-MPIO服務，分割一般設備節點(如 `/dev/sdc`)，然後重新開機，讓 DM-MPIO 服務識別新的分割區。

7.4.3 設定伺服器以進行多重路徑

您必須手動設定系統，為多重路徑 IO 設備在 `initrd` 中所連接的控制器自動載入設備驅動程式。您需要將必要的驅動程式模組新增至檔案 `/etc/sysconfig/kernel` 中的變數 INITRD_MODULES。

例如，如果系統包含由 `cciss` 驅動程式存取的 RAID 控制器，且多重路徑設備連接至由 qla2xxx 驅動程式存取的 QLogic* 控制器，則此項目應為：

```
INITRD_MODULES="cciss"
```

由於 QLogic 驅動程式並不會在系統啟動時自動載入，因此請新增在這裡：

```
INITRD_MODULES="cciss qla23xx"
```

變更 `/etc/sysconfig/kernel` 後，必須使用 `mkinitrd` 指令在系統中重新建立 `initrd`，然後重新開機以使變更生效。

若您將 LILO 當作開機管理員使用，請使用 `/sbin/lilo` 指令將其重新安裝。若您使用 GRUB，則無須進一步動作。

7.4.4 新增 multipathd 至開機順序

使用本節所述的方法將多重路徑 I/O 服務 (multipathd) 新增至開機順序。

- 章節「使用 YaST 新增 multipathd」[74頁]
- 章節「使用指令行新增 multipathd」[74頁]

使用 YaST 新增 multipathd

1. 在 YaST® 中，按一下「*系統*」>「*系統服務 (執行層級)*」>「*簡單模式*」。
2. 選取「*multipathd*」，然後按一下「*啟用*」。
3. 按一下「*確定*」確認服務啟動訊息。
4. 按一下「*完成*」，然後按一下「*是*」。

   這些變更只有在伺服器重新啟動後才會生效。

使用指令行新增 multipathd

1. 開啟終端機主控台，以 root 使用者或同等身分登入。
2. 在終端機主控台提示符下，輸入

   ```
   insserv multipathd
   ```

7.4.5 建立並設定 /etc/multipath.conf 檔案

在您建立 /etc/multipath.conf 檔案之前，該檔案並不存在。/usr/share/doc/packages/multipath-tools/multipath.conf.synthetic 檔案包含 /etc/multipath.conf 範例檔案，您可將其當成多重路徑的設定指南。請參閱 /usr/share/doc/packages/multipath-tools/multipath.conf.annotated，瞭解包含每個屬性及其選項之詳細備註的範本。

- 章節「建立 multipath.conf 檔案」[75頁]
- 章節「驗證 etc/multipath.conf 檔案中的設定」[76頁]
- 章節「在 /etc/multipath.conf 中設定易記名稱或別名」[76頁]
- 章節「在 /etc/multipath.conf 中將非多重路徑設備列入黑名單」[78頁]
- 章節「在 /etc/multipath.conf 中設定預設多重路徑行為」[79頁]
- 章節「在 /etc/multipath.conf 中為 zSeries 設定預設設定值」[79頁]
- 章節「套用 /etc/multipath.conf 檔案變更」[80頁]

建立 multipath.conf 檔案

如果 `/etc/multipath.conf` 檔案不存在，請複製範例以建立此檔案：

1. 在終端機主控台中，以 `root` 使用者身分登入。
2. 輸入以下指令 (所有指令必須在一行中) 複製範例：

   ```
   cp /usr/share/doc/packages/multipath-tools/multipath.conf.synthetic
   /etc/multipath.conf
   ```

3. 將 `/usr/share/doc/packages/multipath-tools/multipath.conf.annotated` 檔案作為參考，以決定如何為系統設定多重路徑。
4. 請確保您的 SAN 有相應的 `device` 項目。大多數廠商會提供有關正確設定 `device` 區段的文件。

   在 `/etc/multipath.conf` 檔案中，不同的 SAN 需要不同的 `device` 區段。如果您使用的是自動偵測到的儲存子系統 (請參閱章節「經測試支援多重路徑的儲存陣列」[63頁])，則可以使用該設備的預設項目；而不需要對 `/etc/multipath.conf` 檔案的組態做進一步的設定。
5. 儲存檔案。

驗證 etc/multipath.conf 檔案中的設定

設定組態後，可以輸入以下指令執行試執行 (dry run)

```
multipath -v3 -d
```

此指令會掃描設備，然後顯示設定的內容。輸出類似下方所列：

```
26353900f02796769
[size=127 GB]
[features="0"]
[hwhandler="1    emc"]


\_ round-robin 0 [first]
  \_ 1:0:1:2 sdav 66:240  [ready ]
  \_ 0:0:1:2 sdr  65:16   [ready ]


\_ round-robin 0
  \_ 1:0:0:2 sdag 66:0    [ready ]
  \_ 0:0:0:2 sdc   8:32   [ready ]
```

路徑會分為不同的優先程度群組。每次只有一個優先程度群組處於主動狀態。若要塑造主動/主動組態，所有路徑都必須分在相同的群組。若要塑造主動/被動組態，不應平行作用的路徑則會放置於不同的優先程度群組。這通常會在設備探查時自動進行。

輸出會顯示順序、群組中用來平衡 I/O 的排程規則，以及各優先程度群組的路徑。對於每個路徑，會顯示其實體位址 (host:bus:target:lun)、裝置節點名稱、主要和次要的設備編號和狀態。

在 /etc/multipath.conf 中設定易記名稱或別名

多重路徑設備可透過其 WWID (全球識別碼) 或為其指定的別名來識別。WWID 是多重路徑設備的識別碼，保證其全球唯一，永不變更。多重路徑中使用的預設名稱是邏輯單位的 ID，就是 `/dev/disk/by-id` 目錄中存放的 ID。由於以 `/dev/sdn` 與 `/dev/dm-n` 形式表示的設備節點名稱會在系統重新開機時變更，因此最好使用其 ID 來表示多重路徑設備。

`/dev/mapper` 目錄中的多重路徑設備名稱會參考 LUN 的 ID，且始終保持一致，因為它們使用 `/var/lib/multipath/bindings` 檔案來追蹤其中的關

聯。這些設備名稱是使用者易記名稱，例如 /dev/disk/by-id/dm-uuid-.*-mpath-.*。

您可以在 /etc/multipath.conf 檔案中使用 ALIAS 指令，指定自己的設備名稱。別名會置換 ID 以及 /dev/disk/by-id/dm-uuid-.*-mpath-.* 名稱。

重要

建議您不要對根設備使用別名，因為當該設備名稱變更後您將無法再透過核心指令行無縫關閉多重路徑。

如需 multipath.conf 設定的範例，請參閱 /usr/share/doc/packages/multipath-tools/multipath.conf.synthetic 檔案。

1. 在終端機主控台中，以 root 使用者身分登入。
2. 在文字編輯器中開啟 /etc/multipath.conf 檔案。
3. 取消 Defaults 指令及其結束括號的註解。
4. 取消 user_friendly_names option 的註解，然後將「否」值變更為「是」。

   例如：

   ```
   ## Use user friendly names, instead of using WWIDs as names.
   defaults {
     user_friendly_names yes
   }
   ```

5. 使用 multipath 區段中的 alias 指令，選擇性為設備指定自己的易記名稱。

   例如：

   ```
   multipath {
     wwid 26353900f02796769
     alias sdd4l0
   }
   ```

6 儲存變更，然後關閉檔案。

在 /etc/multipath.conf 中將非多重路徑設備列入黑名單

/etc/multipath.conf 檔案應該包含 blacklist 區段，其中列出了所有非多重路徑設備。例如，本地 IDE 硬碟與軟碟機通常為非多重路徑設備。如果 multipath 會嘗試管理您的單一路徑設備，而您希望 multipath 忽略這些設備，只需將它們加入 blacklist 區段便可解決問題。

注意

關鍵字 devnode_blacklist 已廢棄，被關鍵字黑名單取代。

例如，若要將本地設備與 cciss 驅動程式中的所有陣列列入黑名單，以免受到多重路徑的管理，則 blacklist 區段應為：

```
blacklist {
      wwid 26353900f02796769
      devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st|sda)[0-9]*"
      devnode "^hd[a-z][0-9]*"
      devnode "^cciss!c[0-9]d[0-9].*"
}
```

您也可以只將驅動程式中的分割區 (而不是整個陣列) 列入黑名單。例如，使用下列正規表示式可以只將 cciss 驅動程式中的分割區 (而不是整個陣列) 列入黑名單：

```
^cciss!c[0-9]d[0-9]*[p[0-9]*]
```

修改 /etc/multipath.conf 檔案後，必須執行 mkinitrd 在系統中重新建立 initrd，然後重新開機以使變更生效。

此後，當您發出 multipath -ll 指令時，本地設備將不再列於多重路徑映射中。

在 /etc/multipath.conf 中設定預設多重路徑行為

/etc/multipath.conf 檔案應包含 defaults 區段，您可在其中指定預設行為。如果 device 區段中沒有指定此欄位，系統會為該 SAN 組態套用預設設定。

下面的 defaults 區段指定了一個簡單的容錯移轉規則：

```
defaults {
       multipath_tool  "/sbin/multipath -v0"
       udev_dir        /dev
       polling_interval 10
       default_selector        "round-robin 0"
       default_path_grouping_policy    failover
       default_getuid  "/sbin/scsi_id -g -u -s /block/%n"
       default_prio_callout    "/bin/true"
       default_features        "0"
       rr_min_io               100
       failback                immediate
```

注意

在 default_getuid 指令行中，使用上例所述的路徑 /sbin/scsi_id 代替 /lib/udev/scsi_id 的範例路徑 (位於範例檔案 /usr/share/doc/packages/multipath-tools/multipath.conf.synthetic 以及預設和註記範例檔案中)。在 SLES 11 SP1 及更高版本中，範例檔案應包含 /sbin/scsi_id 的正確路徑。

在 /etc/multipath.conf 中為 zSeries 設定預設設定值

對支援多重路徑的 IBM zSeries 設備進行的測試表明，應將 dev_loss_tmo 參數設定為 90 秒，將 fast_io_fail_tmo 參數設定為 5 秒。若您使用的是 zSeries 設備，請依照以下方式修改 /etc/multipath.conf 檔案以指定這些值：

```
defaults {
       dev_loss_tmo 90
       fast_io_fail_tmo 5
}
```

dev_loss_tmo 參數設定將某個多重路徑連結標示為失敗之前需等待的秒數。若該路徑失敗，則目前在該路徑上的所有 I/O 都會失敗。預設值會因所用的設備

驅動程式而有所不同。值的有效範圍為 0 至 600 秒。若要使用驅動程式的內部逾時，請將該值設定為零 (0) 或任何大於 600 的值。

fast_io_fail_tmo 參數設定偵測到連結問題後將 I/O 確定為失敗之前需等待的時間。到達該驅動程式的 I/O 都會失敗。如果 I/O 排在擁堵的佇列中，則未到 dev_loss_tmo 時間且佇列未疏通之前 I/O 不會失敗。

套用 /etc/multipath.conf 檔案變更

當 `multipathd` 正在執行時，對 `/etc/multipath.conf` 檔案所做的變更無法生效。在進行變更後，儲存並關閉檔案，然後執行以下操作以套用變更：

1. 停止 `multipathd` 服務。
2. 輸入以下指令清除舊的多重路徑繫結

   ```
   /sbin/multipath -F
   ```

3. 輸入以下指令建立新的多重路徑繫結

   ```
   /sbin/multipath -v2 -l
   ```

4. 啟動 `multipathd` 服務。
5. 執行 `mkinitrd` 在系統中重新建立 `initrd`，然後重新開機以使變更生效。

7.5 啟用及啟動多重路徑 I/O 服務

若要啟動多重路徑服務，以及在重新開機時啟用該服務，請執行下列步驟：

1. 開啟終端機主控台，以 `root` 使用者或同等身分登入。
2. 在終端機主控台提示符下，輸入

   ```
   chkconfig multipathd on
   ```

```
chkconfig boot.multipath on
```

如果系統開機時 boot.multipath 服務沒有自動啟動，請執行下列操作手動啟動：

1 開啟終端機主控台，以 `root` 使用者或同等身分登入。

2 輸入

```
/etc/init.d/boot.multipath start
```

```
/etc/init.d/multipathd start
```

7.6 設定路徑容錯移轉規則與優先程度

在 Linux 主機上，若一個儲存控制器有多個路徑，則每個路徑會顯示為獨立的區塊設備，導致單個LUN有多個區塊設備。設備對應程式多重路徑服務會偵測到多個路徑具有同一個 LUN ID，然後會使用該 ID 建立新的多重路徑設備。例如，若主機上有兩個HBA透過未分區的光纖通道交換器連接至具有兩個連接埠的某個儲存控制器，則該主機會探查到四個區塊設備，即 `/dev/sda`、`/dev/sdb`、`/dev/sdc` 與 `/dev/sdd`。設備對應程式多重路徑服務會建立單個區塊設備 `/dev/mpath/mpath1`，透過以上四個基礎區塊設備來重新路由 I/O。

本節說明如何指定容錯移轉的規則及如何設定路徑的優先程度。

- 第 7.6.1 節「設定路徑容錯移轉規則」[81頁]
- 第 7.6.2 節「設定容錯移轉優先程度」[82頁]
- 第 7.6.3 節「使用程序檔設定路徑優先程度」[90頁]
- 第 7.6.4 節「設定 ALUA (mpath_prio_alua)」[91頁]
- 第 7.6.5 節「報告目標路徑群組」[93頁]

7.6.1 設定路徑容錯移轉規則

使用包含 -p 選項的 `multipath` 指令來設定路徑容錯移轉規則：

```
multipath devicename -p policy
```

用以下其中一個規則選項取代 *policy*：

表格 7.4 *multipath -p 指令的群組規則選項*

規則選項	描述
failover	每個優先程度群組對應一個路徑。
multibus	所有路徑都在一個優先程度群組中。
group_by_serial	每個偵測到的序號對應一個優先程度群組。
group_by_prio	每個路徑優先程度值對應一個優先程度群組。優先程度由註標程式決定，在 /etc/multipath.conf 組態檔案中指定為 global、per-controller 或 per-multipath 選項。
group_by_node_name	每個目標節點名稱對應一個優先程度群組。目標節點名稱取自 /sys/class/fc_transport/target*/node_name 位置。

7.6.2 設定容錯移轉優先程度

您必須在 /etc/multipath.conf 檔案中手動輸入設備的容錯移轉優先程度。/usr/share/doc/packages/multipath-tools/multipath.conf.annotated 檔案中提供了所有設定與選項的範例。

- 章節「瞭解優先程序群組與屬性」[83頁]
- 章節「設定輪替式負載平衡」[89頁]
- 章節「設定單一路徑容錯移轉」[89頁]
- 章節「將 I/O 路徑分組以使用輪替式負載平衡」[89頁]

瞭解優先程序群組與屬性

*優先程度群組*是進入相同實體 LUN 的路徑集合。依預設，I/O 以輪替方式發佈到群組的所有路徑。`multipath` 指令會根據 SAN 的 path_grouping_policy 設定，在該 SAN 中自動為每個 LUN 建立優先程度群組。`multipath` 指令會將群組中路徑的數目乘以群組的優先程度，來決定哪個群組為主要群組。計算值最高的群組為主要群組。主要群組中的所有路徑都失敗時，計算值次高的優先程度群組成為主動群組。

*路徑優先程度*是指定給路徑的整數值。值越高，優先程度越高。系統使用了外部程式來指定每個路徑的優先程度。對於指定的設備，優先程度相同的路徑屬於同一個優先程度群組。

表格 7.5 *多重路徑屬性*

多重路徑屬性	描述	值
user_friendly_names	指定是使用 ID 還是使用 `/var/lib/multipath/bindings` 檔案為多重路徑設備指定格式為 `/dev/mapper/mpathN`、永久且唯一的別名。	**yes**： 自動產生易記名稱做為多重路徑設備的別名來取代實際的 ID。 **no**： 預設值。使用 `/dev/disk/by-id/` 位置所顯示的 WWID。
blacklist	指定要做為非多重路徑設備加以忽略的設備名稱清單，例如 cciss、fd、hd、md、dm、sr、scd、st、ram、raw 及 loop 等。	如需取得範例說明，請參閱 章節「在 /etc/multipath.conf 中將非多重路徑設備列入黑名單」[78頁]。
blacklist_exceptions	指定要視為多重路徑設備加以處理的設備名稱清單，即使這些名稱包含於黑名單中。	如需範例，請參閱 `/usr/share/doc/packages/multipath-tools/multipath.conf.annotated` 檔案。

多重路徑屬性	描述	值
failback	指定是否監控失敗路徑的復原狀況，並指出失敗路徑恢復使用後群組進行錯誤回復所花的時間。 失敗路徑復原後，系統會根據此設定將該路徑重新新增到啟用了多重路徑的路徑清單中。多重路徑會評估優先程度群組，並在主要路徑的優先程度高於次要群組時，變更主動優先程度群組。	**immediate**： 某路徑復原後，立即啟用該路徑。 **n (> 0)**： 某路徑復原後，等待 *n* 秒後再啟用此路徑。指定一個大於 0 的整數值。 **manual**： (預設) 不監控失敗路徑的復原狀況。管理員會執行 `multipath` 指令來更新已啟用的路徑與優先程度群組。

getuid	為獲得唯一路徑識別碼而呼叫的預設程式和引數。指定時應使用絕對路徑。	`/sbin/scsi_id -g -u -s`
		
		這是預設位置和引數。
		
		範例：
		
		`getuid "/sbin/scsi_id -g -u -d /dev/%n"`

no_path_retry	指定路徑失敗時要採取的行為。	**n (> 0)**： 指定在 `multipath` 停止佇列且使路徑失敗之前的重試次數。指定一個大於 0 的整數值。 **fail**： 指定的立即失敗 (不排入佇列)。 **queue**： 永不停止佇列 (在路徑恢復正常之前始終排入佇列)。

多重路徑屬性	描述	值
path_grouping_policy	指定由指定控制器代管之多重路徑設備的路徑分組規則。	**failover：** 每個優先程度群組指定一個路徑，以便每次只使用一個路徑。 **multibus：** (預設) 所有有效的路徑均屬於一個優先程度群組。流量透過群組中的所有主動路徑來保持負載平衡。 **group_by_prio：** 每個路徑優先程度值對應一個優先程度群組。優先程度相同的路徑位於同一個優先程度群組中。優先程度由外部程式指定。 **group_by_serial：** 路徑根據 SCSI 目標序號 (控制器節點 WWN) 來分組。 **group_by_node_name：** 每個目標節點名稱指定一個優先程度群組。目標節點名稱取自 `/sys/class/fc_transport/target*/node_name` 中。
path_checker	決定路徑的狀態。	**directio：** (`multipath-tools` 0.4.8 版及更高版本中的預設值) 讀取具備直接 I/O 的第一個磁區，這對 DASD 設備非常有用。在 `/var/log/messages` 中記錄失敗訊息。 **readsector0：** (`multipath-tools` 0.4.7 版及較早版本的預設值) 讀取設備的第一個磁區。在 `/var/log/messages` 中記錄失敗訊息。

多重路徑屬性	描述	值
		tur： 將 SCSI 測試單位就緒指令發送至設備。這是較佳設定(若 LUN 支援)。若指令失敗，它不會在 `/var/log/messages` 中記錄訊息。 有些 SAN 廠商會提供自定 path_checker 選項： • **emc_clariion：** 查詢 EMC Clariion EVPD 的 0xC0 頁以決定路徑狀態。 • **hp_sw：** 檢查包含主動/待機韌體之 HP 儲存陣列的路徑狀態 (Up、Down 或 Ghost)。 • **rdac：** 檢查 LSI/Engenio RDAC 儲存控制器的路徑狀態。
path_selector	指定用於負載平衡的 path-selector 演算法。	**round-robin 0**： (預設值) 負載平衡演算法用於平衡優先程度群組中所有主動路徑中的流量。 從 SUSE Linux Enterprise Server 11 開始，提供了下列額外的 I/O 平衡選項： **least-pending：** 為基於 BIO 的設備對應程式多重路徑提供 least-pending-I/O 動態負載平衡規則。此負載平衡規則會考慮路徑上待處理的未服務申請數，選取待處理服務申請最少的路徑。 此規則在 SAN 環境包含異質元件時尤其有用。例如，當有一個 8GB

多重路徑屬性	描述	值
		HBA 和一個 2GB HBA 連接到同一伺服器時，使用該演算法可將 8GB HBA 發揮出更大的作用。 **length-load-balancing：** 與 least-pending 選項類似，是一個可平衡多個路徑上進行中的 I/O 數量的動態負載平衡器。 **service-time：** 是一個可根據延遲情況來平衡多個路徑上之 I/O 的服務時間導向的負載平衡器。
pg_timeout	指定路徑群組逾時處理。	NONE (內部預設值)
prio_callout 多重路徑 prio_callout 位於 `/lib/libmultipath/lib*` 中的共享程式庫中。透過使用共享程式庫，callout 會在精靈啟動時載入到記憶體中。	指定要用來決定多重路徑映射之配置的程式和引數。 使用 `multipath` 指令查詢時，指定的 mpath_prio_* callout 程式會傳回相對於整個多重路徑配置之指定路徑的優先程度。 該指令與 group_by_prio 的 path_grouping_policy 一起使用時，具有相同優先程度的所有路徑都會分到一個多重路徑群組。總優先程度最高的群組成為主動群組。	如果未使用任何 `prio_callout` 屬性，則所有路徑的優先程度均相等。此為預設選項。 **/bin/true：** 未使用 group_by_priority 時，請使用此值。 使用 `multipath` 指令查詢時，`prioritizer` 程式會產生路徑的優先程度。程式名稱必須以 `mpath_prio_` 開頭，並以設備類型或所使用的平衡方式命名。目前的 prioritizer 程式包括以下幾種： **mpath_prio_alua %n：** 根據 SCSI-3 ALUA 設定產生路徑優先程度。 **mpath_prio_balance_units：** 為所有路徑產生相同的優先程度。

多重路徑屬性	描述	值
	群組中的所有路徑都失敗時，總優先程度次高的群組成為主動群組。此外，系統可能還會將一條容錯移轉指令(由硬體處理器決定)傳送至該目標。 mpath_prio_* 程式也可以是廠商或管理員為指定設定建立的自定程序檔。 指令行中的 %n 會展開至 /dev 目錄中的設備名稱。 %b 會展開至 /dev 目錄中 *major:minor* 格式的設備編號。 %d 會展開至 /dev/disk/by-id 目錄中的設備 ID。 如果設備為熱插式設備，請使用 %d 旗標取代 %n。這能解決從設備可以使用到 udev 建立設備節點之間所經過的一小段時間的問題。	**mpath_prio_emc %n：** 為 EMC 陣列產生路徑優先程度。 **mpath_prio_hds_modular %b：** 為 Hitachi HDS Modular 儲存陣列產生路徑優先程度。 **mpath_prio_hp_sw %n：** 為主動/待機模式下的 Compaq/HP 控制器產生路徑優先程度。 **mpath_prio_netapp %n：** 為 NetApp 陣列產生路徑優先程度。 **mpath_prio_random %n：** 為每個路徑產生隨機優先程度。 **mpath_prio_rdac %n：** 為 LSI/Engenio RDAC 控制器產生路徑優先程度。 **mpath_prio_tpc %n：** 您可以選擇性地使用由廠商或管理員建立的程序檔，從您用來為每個路徑指定要使用的優先程度的檔案中取得優先程度。 **mpath_prio_spec.sh %n：** 提供使用者建立的程序檔的路徑，該程序檔會根據第二個資料檔案中包含的資訊，為多重路徑產生優先程度。(此路徑與檔名僅做為範例。請指定您自己的程序檔位置。)程序檔可由廠商或管理員建立。程序檔的目標檔案會識別所有多重路徑設備的每個路徑，並為每個路徑指定優先程度。如需取得範例說明，請參閱第

多重路徑屬性	描述	值
		7.6.3 節「使用程序檔設定路徑優先程度」 [90頁]。
rr_min_io	指定需要路由至一個路徑之後再切換至同一個路徑群組中下一個路徑的 I/O 異動的數目，這由 `path_selector` 設定中指定的演算法決定。	**n (>0)**： 指定一個大於 0 的整數值。 **1000**： 預設值。
rr_weight	指定用於計算路徑權重的方式。	**uniform**： 預設值。所有路徑都擁有相同的輪替權重。 **priorities**： 每個路徑的權重由路徑的優先程度乘以 rr_min_io 設定來確定。

設定輪替式負載平衡

所有路徑都處於主動狀態。I/O 設定為在移至序列中的下一個開啟路徑之前需經過秒數的時間或數個 I/O 異動。

設定單一路徑容錯移轉

優先程度最高 (設定值最低) 的單一路徑對流量而言是主動路徑。其他路徑可用於容錯移轉，但只有在發生容錯移轉時才會使用。

將 I/O 路徑分組以使用輪替式負載平衡

具有相同優先程度的多個路徑都歸入主動群組。該群組中的所有路徑都失敗時，設備會容錯移轉至優先程度次高的群組。群組中的所有路徑以輪替式負載平衡方式共享流量負載。

7.6.3 使用程序檔設定路徑優先程度

您可以建立一個與設備對應程式多重路徑(DM-MPIO)互動的程序檔，以在LUN設定為 `prio_callout` 設定的資源時，為 LUN 的路徑提供優先程度。

首先，設定文字檔，在其中列出關於每個設備的資訊以及要指定給每個路徑的優先程度值。例如，將檔案命名為 `/usr/local/etc/primary-paths`。以下列格式為每個路徑輸入一行指令：

```
host_wwpn target_wwpn scsi_id priority_value
```

這會為設備中的每個路徑傳回一個優先程度值。請確保變數 FILE_PRIMARY_PATHS 可解析為含有每個設備相應資料 (如主機 wwpn、目標 wwpn、scsi_id 及優先程度值) 的實際檔案。

包含八個路徑的單一 LUN 的 `primary-paths` 檔案中，每個路徑顯示如下：

```
0x10000000c95ebeb4 0x200200a0b8122c6e 2:0:0:0 sdb
3600a0b8000122c6d00000000453174fc 50

0x10000000c95ebeb4 0x200200a0b8122c6e 2:0:0:1 sdc
3600a0b80000fd6320000000045317563 2

0x10000000c95ebeb4 0x200200a0b8122c6e 2:0:0:2 sdd
3600a0b8000122c6d0000000345317524 50

0x10000000c95ebeb4 0x200200a0b8122c6e 2:0:0:3 sde
3600a0b80000fd6320000000245317593 2

0x10000000c95ebeb4 0x200300a0b8122c6e 2:0:1:0 sdi
3600a0b8000122c6d00000000453174fc 5

0x10000000c95ebeb4 0x200300a0b8122c6e 2:0:1:1 sdj
3600a0b80000fd6320000000045317563 51

0x10000000c95ebeb4 0x200300a0b8122c6e 2:0:1:2 sdk
3600a0b8000122c6d0000000345317524 5

0x10000000c95ebeb4 0x200300a0b8122c6e 2:0:1:3 sdl
3600a0b80000fd6320000000245317593 51
```

若要繼續表格 7.5「多重路徑屬性」[83頁] 中提到的範例，請建立名為 /usr/local/sbin/path_prio.sh 的程序檔。您可使用任何路徑和檔名。程序檔會執行下列動作：

- 在多重路徑中查詢時，從 /usr/local/etc/primary-paths 檔案中查找設備及其路徑。
- 將檔案中該項目最後一欄中的優先程度值傳回多重路徑。

7.6.4 設定 ALUA (mpath_prio_alua)

mpath_prio_alua(8) 指令可做為 Linux multipath(8) 指令的優先程度註標。該指令會傳回 DM-MPIO 用於將優先程度相同的 SCSI 設備分在同一組的編號。此路徑優先程度工具以 ALUA (非同步邏輯單位存取) 為基礎。

- 章節「語法」[91頁]
- 章節「必備條件」[91頁]
- 章節「選項」[91頁]
- 章節「傳回值」[92頁]

語法

```
mpath_prio_alua [-d directory] [-h] [-v] [-V] device [device...]
```

必備條件

SCSI 設備。

選項

-d *directory*
指定可在其中找到所列設備節點名稱的 Linux 目錄路徑。預設目錄為 /dev。使用此選項時，請只指定要管理的一或多個設備的設備節點名稱 (如 sda)。

-h
顯示此指令的說明，然後結束。

-v
開啟詳細輸出，以較易理解的形式顯示狀態。輸出包括有關指定設備所處的連接埠群組及其目前狀態的資訊。

-V
顯示此工具的版本號碼，然後結束。

設備 [*設備*...]
指定要管理的 SCSI 設備 (或多個設備)。該設備必須為支援報告目標連接埠群組 (`sg_rtpg(8)`) 指令的 SCSI 設備。為設備節點名稱使用下列其中一種格式：

- 使用完整的 Linux 目錄路徑，如 `/dev/sda`。請勿與 -d 選項一起使用。
- 只使用設備節點名稱，如 `sda`。使用 -d 選項指定目錄路徑。
- 設備的主要和次要編號以冒號 (:) 分隔，不含空格，如 `8:0`。這會在 `/dev` 目錄中建立名稱為 `tmpdev-<major>:<minor>-<pid>` 格式的暫存設備節點。例如，`/dev/tmpdev-8:0-<pid>`。

傳回值

成功後會傳回值 0 和群組的優先程度值。表格 7.6「設備對應程式多重路徑的 ALUA 優先程度」[92頁] 顯示了 `mpath_prio_alua` 指令所傳回的優先程度值。

表格 7.6 *設備對應程式多重路徑的 ALUA 優先程度*

優先程度值	描述
50	設備屬於主動且最佳化的群組。
10	設備屬於主動但非最佳化的群組。
1	設備屬於待機群組。

優先程度值	描述
0	所有其他群組。

由於 multipath 指令對每個值的處理方式不同，因此這些值相差比較大。該指令會將群組中路徑的數目乘以群組的優先程度值，然後選取所得結果最高的群組。例如，如果非最佳化的路徑群組有 6 個路徑 (6 x 10 = 60)，最佳化路徑群組有一個路徑 (1 x 50 = 50)，那麼非最佳化群組的得分最高，因此 multipath 會選擇非最佳化群組。流向設備的流量會以輪替式方式使用該群組中的所有 6 個路徑。

失敗時會傳回指出指令失敗原因的值 (1 到 5)。如需資訊，請參閱 mpath_prio_alua 的線上文件。

7.6.5 報告目標路徑群組

使用 SCSI 報告目標連接埠群組 (sg_rtpg(8)) 指令。如需資訊，請參閱 sg_rtpg(8) 的線上文件。

7.7 為特定主機匯流排配接器微調容錯移轉

使用多重路徑 I/O 時，您希望報告主機匯流排配接器 (HBA) 的任何失敗或纜線失敗的速度比沒有使用多重路徑時更快。設定 HBA 的逾時設定以停用 HBA 層級的容錯移轉，這樣可以最快速度將失敗傳播至多重路徑 I/O 層級，I/O 即可重新導向至另一個正常路徑。

若要停用 HBA 容錯移轉處理，請修改 /etc/modprobe.conf.local 檔案中的驅動程式選項。如需如何停用驅動程式的容錯移轉設定的資訊，請參閱 HBA 廠商文件。

例如，對於主機匯流排配接器的 QLogic qla2xxx 系列，建議做下列設定：

```
options qla2xxx qlport_down_retry=1
```

7.8 設定根設備的多重路徑 I/O

重要

在 SUSE Linux Enterprise Server 10 SP1 初版以及較早版本中，僅當 `/boot` 分割區位於獨立的非多重路徑分割區中時，系統才支援多重路徑中的根分割區 (/)。否則，系統不會寫入任何開機載入程式。

現在，SUSE Linux Enterprise Server 11 中提供了 DM-MPIO 及其對 `/boot` 和 `/root` 的支援。此外，YaST2 安裝程式中的 YaST 磁碟分割程式支援在安裝期間啟用多重路徑。

- 第 7.8.1 節「啟用多重路徑 I/O 以在多重路徑儲存 LUN 上安裝 SLES」[94頁]
- 第 7.8.2 節「啟用多重路徑 I/O 以在主動/被動多重路徑儲存 LUN 上安裝 SLES」[95頁]
- 第 7.8.3 節「對現有根設備啟用多重路徑 I/O」[97頁]
- 第 7.8.4 節「在根設備上停用多重路徑 I/O」[98頁]

7.8.1 啟用多重路徑 I/O 以在多重路徑儲存 LUN 上安裝 SLES

系統安裝過程中，multipathd 精靈不會自動啟動。您可以使用 YaST 磁碟分割程式中的「*設定多重路徑*」選項來啟動。

1. 安裝期間，在 YaST2 的「安裝設定」頁面上按一下「*磁碟分割*」，開啟 YaST 磁碟分割程式。
2. 選取「*自定分割區 (進階)*」。
3. 選取「*硬碟*」主圖示，按一下「*設定*」按鈕，然後選取「*設定多重路徑*」。
4. 啟動多重路徑。

YaST2 即會開始重新掃描磁碟，然後顯示可用的多重路徑設備 (例如 /dev/mapper/3600a0b80000f4593000012ae4ab0ae65)。之後所有的處理步驟都應使用此設備。

5 按「*下一步*」繼續安裝。

7.8.2 啟用多重路徑 I/O 以在主動/被動多重路徑儲存 LUN 上安裝 SLES

系統安裝過程中，multipathd 精靈不會自動啟動。您可以使用 YaST 磁碟分割程式中的「*設定多重路徑*」選項來啟動。

1 安裝期間，在 YaST2 的「安裝設定」頁面上按一下「*磁碟分割*」，開啟 YaST 磁碟分割程式。

2 選取「*自定分割區 (進階)*」。

3 選取「*硬碟*」主圖示，按一下「*設定*」按鈕，然後選取「*設定多重路徑*」。

4 啟動多重路徑。

YaST2 即會開始重新掃描磁碟，然後顯示可用的多重路徑設備 (例如 /dev/mapper/3600a0b80000f4593000012ae4ab0ae65)。之後所有的處理步驟都應使用此設備。記下設備路徑與 UUID，稍後會用到。

5 按「*下一步*」繼續安裝。

6 完成所有設定和安裝後，YaST2 即會開始寫入開機載入程式資訊，並顯示重新啟動系統的倒數計時。按一下「*停止*」按鈕停止計數器，然後按 CTRL+ALT+F5 存取主控台。

7 使用主控台確定是否在 /boot/grub/device.map 檔案中為 hd0 項目輸入了被動路徑。

執行此動作非常必要，因為安裝程序無法區分主動路徑與被動路徑。

7a 輸入以下指令，將根設備掛接至 /mnt

```
mount /dev/mapper/UUID_part2 /mnt
```

例如，輸入

```
mount /dev/mapper/3600a0b80000f4593000012ae4ab0ae65_part2 /mnt
```

7b 輸入以下指令，將開機設備掛接至 /mnt/boot

```
mount /dev/mapper/UUID_part1 /mnt/boot
```

例如，輸入

```
mount /dev/mapper/3600a0b80000f4593000012ae4ab0ae65_part1 /mnt/boot
```

7c 輸入以下指令，開啟 /mnt/boot/grub/device.map 檔案

```
less /mnt/boot/grub/device.map
```

7d 在 /mnt/boot/grub/device.map 檔案中，確定 hd0 項目是否指向被動路徑，然後執行下列其中一項動作：

- **主動路徑：** 不需要執行任何動作，跳過步驟 8 [96頁]，繼續步驟 9 [97頁]。
- **被動路徑：** 必須變更組態並重新安裝開機載入程式。繼續執行步驟 8 [96頁]。

8 如果 hd0 項目指向被動路徑，請變更組態並重新安裝開機載入程式：

8a 在主控台提示符處，輸入下列指令：

```
mount -o bind /dev /mnt/dev
mount -o bind /sys /mnt/sys
mount -o bind /proc /mnt/proc
```

```
chroot
```

8b 在主控台中執行 `multipath -ll`，然後檢查輸出以尋找主動路徑。

被動路徑會有 `ghost` 標記。

8c 在 `/mnt/boot/grub/device.map` 檔案中，將 `hd0` 項目變更為主動路徑並儲存變更，然後關閉檔案。

8d 如果先前選擇從 MBR 開機，`/etc/grub.conf` 內容應如下所示：

```
setup --stage2=/boot/grub/stage2 (hd0) (hd0,0)
quit
```

8e 輸入以下指令，重新安裝開機載入程式

```
grub < /etc/grub.conf
```

8f 輸入下列指令：

```
exit
umount /mnt/*
umount /mnt
```

9 按 CTRL+ALT+F7 返回 YaST 圖形環境。

10 按一下「*確定*」繼續執行安裝的重新開機作業。

7.8.3 對現有根設備啟用多重路徑 I/O

1 僅使用單個主動路徑安裝 Linux，尤其是當磁碟分割程式中列有 `by-id` 符號連結時。

2 使用安裝期間所用的 `/dev/disk/by-id` 路徑來掛接設備。

3 安裝之後，將 dm-multipath 新增至 /etc/sysconfig/kernel: INITRD_MODULES。

4 對於 System Z，請先編輯 /etc/zipl.conf 檔案，以 /etc/fstab 中使用的 by-id 資訊變更 zipl.conf 中的 by-path 資訊，然後再執行 mkinitrd。

5 重新執行 /sbin/mkinitrd 以更新 initrd 影像。

6 對於 System Z，請在執行 mkinitrd 之後執行 zipl。

7 重新載入伺服器。

7.8.4 在根設備上停用多重路徑 I/O

- 將 multipath=off 新增至核心指令行。

  這只會影響根設備，而不會影響所有其他設備。

7.9 設定現有軟體 RAID 的多重路徑 I/O

理想狀況下，您應該先設定設備的多重路徑，然後再將它們當成軟體 RAID 設備的元件使用。如果您在建立任何軟體 RAID 設備後再新增多重路徑，則系統重新開機時可能會先啟動 multipath 服務，然後再啟動 DM-MPIO 服務，導致 RAID 可能會無法使用多重路徑。您可以使用本節所述的程序，讓多重路徑針對多先前存在的軟體 RAID 執行。

例如，在下列情況中，您可能需要設定軟體 RAID 中設備的多重路徑：

- 如果在執行全新安裝或升級期間建立新的軟體 RAID，將其做為磁碟分割設定的一部分。
- 如果將軟體 RAID 中的設備當成成員設備或備品之前未設定設備以進行多重路徑。
- 如果透過將新的 HBA 配接器新增至伺服器，或擴充 SAN 中的儲存子系統來擴展您的系統。

注意

下列指示假設軟體 RAID 設備為 /dev/mapper/mpath0，這是核心可辨識的設備名稱。請務必修改適用於軟體 RAID 的設備名稱的指示。

1 開啟終端機主控台，以 root 使用者或同等身分登入。

請在下列步驟中使用此主控台輸入指令，除非導向到其他位置。

2 如果目前已掛接或正在執行任何軟體RAID設備，請為每個設備輸入以下指令以卸下設備並將其停止。

```
umount /dev/mapper/mpath0
```

```
mdadm --misc --stop /dev/mapper/mpath0
```

3 輸入以下指令停止 boot.md 服務

```
/etc/init.d/boot.md stop
```

4 輸入以下指令啟動 boot.multipath 和 multipathd 服務：

```
/etc/init.d/boot.multipath start
```

```
/etc/init.s/multipathd start
```

5 啟動多重路徑服務之後，驗證軟體 RAID 的元件設備是否列在 /dev/disk/by-id 目錄中。請執行下列其中一個步驟：

- **設備已列出：** 設備名稱現在應該有連至設備對應程式多重路徑設備名稱的符號連結，如 /dev/dm-1。
- **設備未列出：** 透過衝洗和重新探查設備的方式，強制多重路徑服務辨識設備。

  要實現此目的，請輸入以下指令：

  ```
  multipath -F
  ```

```
multipath -v0
```

設備現在應該列於 /dev/disk/by-id 中，並擁有連至其設備對應程式多重路徑設備名稱的符號連結。例如：

```
lrwxrwxrwx 1 root root 10 Jun 15 09:36 scsi-mpath1 -> ../../dm-1
```

6 輸入以下指令重新啟動 boot.md 服務和 RAID 設備

```
/etc/init.d/boot.md start
```

7 輸入以下指令檢查軟體 RAID 的狀態

```
mdadm --detail /dev/mapper/mpath0
```

RAID 的元件設備應與其設備對應程式多重路徑設備名稱相符，這些設備名稱在 /dev/disk/by-id 目錄中列為設備的符號連結。

8 建立新的 initrd，以確保系統重新開機時先載入設備對應程式多重路徑服務，然後再載入 RAID 服務。輸入

```
mkinitrd -f multipath
```

9 將伺服器重新開機，以套用這些安裝後組態設定。

10 檢查 RAID 狀態，以驗證軟體 RAID 陣列是否正確地顯示在多重路徑設備頂部。輸入

```
mdadm --detail /dev/mapper/mpath0
```

例如：

```
Number Major Minor RaidDevice State
0 253 0 0 active sync /dev/dm-0
1 253 1 1 active sync /dev/dm-1
2 253 2 2 active sync /dev/dm-2
```

7.10 掃描新設備而不重新開機

如果您的系統已經設定多重路徑，而您稍後需要將更多儲存體新增到 SAN，則可以使用 `rescan-scsi-bus.sh` 程序檔掃描新的設備。依預設，此程序檔會掃描所有 HBA 的一般 LUN 範圍。

語法

```
rescan-scsi-bus.sh [options] [host [host ...]]
```

您可以透過指令行指定主機 (已廢棄)，或使用 `--hosts=LIST` 選項來指定 (推薦)。

選項

對於大多數儲存子系統，該程序檔都可在不使用任何選項的情況下成功執行。但在某些特殊情況下，可能需要為 `rescan-scsi-bus.sh` 程序檔使用下列一或多個參數：

選項	描述
`-l`	啟動對 LUN 0-7 的掃描。[預設值：0]
`-L NUM`	啟動對 LUN 0-NUM 的掃描。[預設值：0]
`-w`	掃描 ID 為 0 到 15 的目標設備。[預設值：0 到 7]
`-c`	啟用對通道 0 或 1 的掃描。[預設值：0]
`-r` `--remove`	啟用設備的移除。[預設：已停用]

選項	描述
-i --issueLip	對光纖通道 LIP 進行重設。[預設：已停用]
--forcerescan	重新掃描現有設備。
--forceremove	移除所有設備並重新新增。 **警告** 此選項非常危險，請慎用。
--nooptscan	發現 0 之前持續尋找 LUN。
--color	使用彩色的字首 OLD/NEW/DEL。
--hosts=LIST	僅掃描 LIST 中的主機，LIST 是以逗號分隔的單一值和範圍的清單。不允許使用空格。 --hosts=A[-B][,C[-D]]
--channels=LIST	僅掃描 LIST 中的通道，LIST 是以逗號分隔的單一值和範圍的清單。不允許使用空格。 --channels=A[-B][,C[-D]]
--ids=LIST	僅掃描 LIST 中的目標 ID，LIST 是以逗號分隔的單一值和範圍的清單。不允許使用空格。 --ids=A[-B][,C[-D]]
--luns=LIST	僅掃描 LIST 中的 LUN，LIST 是以逗號分隔的單一值和範圍的清單。不允許使用空格。

選項	描述
	`--luns=A[-B][,C[-D]]`

程序

使用以下程序掃描設備，以便在不將系統重新開機的情況下使這些設備適用於多重路徑。

1. 在儲存子系統中，使用廠商的工具來配置設備並更新其存取控制設定，以允許 Linux 系統存取新的儲存。如需詳細資料，請參閱廠商提供的文件。

2. 掃描主機的所有目標，以使 Linux 核心的 SCSI 子系統的中間層級可辨識其新設備。在終端機主控台提示符處輸入

   ```
   rescan-scsi-bus.sh [options]
   ```

3. 檢查系統記錄 (`/var/log/messages` 檔案) 中的掃描進度。在終端機主控台提示符處輸入

   ```
   tail -30 /var/log/messages
   ```

   此指令會顯示記錄的最後 30 行。例如：

   ```
   # tail -30 /var/log/messages
   . . .
   Feb 14 01:03 kernel: SCSI device sde: 81920000
   Feb 14 01:03 kernel: SCSI device sdf: 81920000
   Feb 14 01:03 multipathd: sde: path checker registered
   Feb 14 01:03 multipathd: sdf: path checker registered
   Feb 14 01:03 multipathd: mpath4: event checker started
   Feb 14 01:03 multipathd: mpath5: event checker started
   Feb 14 01:03:multipathd: mpath4: remaining active paths: 1
   Feb 14 01:03 multipathd: mpath5: remaining active paths: 1
   ```

4. 重複步驟 2 [103頁] 到步驟 3 [103頁]，以透過 Linux 系統中連接至新設備的其他 HBA 配接器新增路徑。

5 執行 multipath 指令辨識可設定 DM-MPIO 組態的設備。在終端機主控台提示符處輸入

```
multipath
```

您現在可以設定新設備以進行多重路徑了。

7.11 掃描新分割的設備而不重新開機

使用本節中的範例，可以在不重新開機的情況下偵測新增的多重路徑 LUN。

1 開啟終端機主控台，然後以 root 使用者身分登入。

2 掃描主機的所有目標，以使 Linux 核心的 SCSI 子系統的中間層級可辨識其新設備。在終端機主控台提示符處輸入

```
rescan-scsi-bus.sh [options]
```

如需 rescan-scsi-bus-sh 程序檔的語法和選項資訊，請參閱第 7.10 節「掃描新設備而不重新開機」[101 頁]。

3 輸入以下指令驗證設備是否已探查到 (連結具有一個新的時戳)

```
ls -lrt /dev/dm-*
```

4 輸入以下指令驗證設備的新 WWN 是否顯示在記錄中

```
tail -33 /var/log/messages
```

5 使用文字編輯器在 /etc/multipath.conf 檔案中新增設備的新別名定義，如 oradata3。

6 輸入以下指令建立設備的分割區表

```
fdisk /dev/dm-8
```

7 輸入以下指令觸發 udev

```
echo 'add' > /sys/block/dm-8/uevent
```

這會為 dm-8 上的分割區產生設備對應程式設備。

8 輸入以下指令為新分割區建立檔案系統和標籤

```
mke2fs -j /dev/dm-9
```

```
tune2fs -L oradata3 /dev/dm-9
```

9 輸入以下指令重新啟動 DM-MPIO，使其讀取別名

```
/etc/init.d/multipathd restart
```

10 輸入以下指令驗證 multipathd 是否可辨識設備

```
multipath -ll
```

11 使用文字編輯器在 /etc/fstab 檔案中新增掛接項目。

此時，您在步驟 5 [104頁] 中建立的別名尚不存在於 /dev/disk/by-label 目錄中。在掛接項目中新增 /dev/dm-9 路徑，然後在下次重新開機到以下項目之前變更該項目

```
LABEL=oradata3
```

12 輸入以下指令建立要做為掛接點的目錄，然後掛接設備

```
md /oradata3
```

```
mount /oradata3
```

7.12 檢視多重路徑 I/O 狀態

查詢多重路徑 I/O 狀態會輸出多重路徑映射的目前狀態。

`multipath -l` 選項會顯示上次執行路徑檢查程式後目前的路徑狀態。該選項不會執行路徑檢查程式。

`multipath -ll` 選項會執行路徑檢查程式，更新路徑資訊，然後顯示目前的狀態資訊。此選項會始終顯示路徑狀態的最新資訊。

- 在終端機主控台提示符處輸入

  ```
  multipath -ll
  ```

  此指令會顯示每個多重路徑設備的資訊。例如：

  ```
  3600601607cf30e00184589a37a31d911
  [size=127 GB][features="0"][hwhandler="1 emc"]

  \_ round-robin 0 [active][first]
    \_ 1:0:1:2 sdav 66:240  [ready ][active]
    \_ 0:0:1:2 sdr  65:16   [ready ][active]

  \_ round-robin 0 [enabled]
    \_ 1:0:0:2 sdag 66:0    [ready ][active]
    \_ 0:0:0:2 sdc  8:32    [ready ][active]
  ```

它會針對每個設備顯示設備的 ID、大小、功能和硬體處理器。

在探查設備時，設備的路徑會自動分到不同的優先程度群組。每次只有一個優先程度群組處於主動狀態。對於主動/主動組態，所有路徑都屬於同一個群組。對於主動/被動組態，被動路徑位於另外的優先程度群組中。

指令會顯示每個群組的下列資訊：

- 用於平衡群組內 I/O 的排程規則，如輪替式
- 該群組是處於主動、已停用還是已啟用狀態
- 該群組是否為第一個 (優先程度最高) 群組
- 群組內包含的路徑

指令會顯示每個路徑的下列資訊：

- 實體位址 *host:bus:target:lun*，如 1:0:1:2

- 設備節點名稱，如 sda
- Major:minor 號碼
- 設備的狀態

7.13 在出錯狀況下管理 I/O

如果所有路徑同時失敗，您可能需要啟用 queue_if_no_path 設定多重路徑，以將 I/O 排入佇列。如果不啟用，I/O 便會在所有路徑都失敗時立即失敗。在驅動程式、HBA 或光纖出現假性錯誤，且這類錯誤會導致所有路徑遺失的特定情況下，應將 DM-MPIO 設定為將所有 I/O 排入佇列，且永不向上傳播錯誤。

在叢集中使用多重路徑設備時，您可以選擇停用 queue_if_no_path。如此，系統就不會將 I/O 排入佇列，而是自動使路徑失敗，並會將 I/O 錯誤升級，產生叢集資源容錯移轉。

啟用 queue_if_no_path 會導致 I/O 在有路徑重新啟用之前無限期地排入佇列，因此請確定 multipathd 正在執行，且對您的情況有效。否則，在重新開機或手動返回到容錯移轉以取代佇列之前，I/O 可能會無限期地擱置於受影響的多重路徑設備中。

若要測試案例，請執行下列步驟：

1. 在終端機主控台中，以 root 使用者身分登入。
2. 輸入以下指令啟動設備 I/O 的佇列功能而非容錯移轉：

   ```
   dmsetup message device_ID 0 queue_if_no_path
   ```

   用設備的 ID 取代 *device_ID*。例如，輸入：

   ```
   dmsetup message 3600601607cf30e00184589a37a31d911 0 queue_if_no_path
   ```

3. 輸入以下指令返回到設備 I/O 的容錯移轉：

   ```
   dmsetup message device_ID 0 fail_if_no_path
   ```

   此指令會立即使所有排入佇列的 I/O 失敗。

用設備的 ID 取代 *device_ID*。例如，輸入：

```
dmsetup message 3600601607cf30e00184589a37a31d911 0 fail_if_no_path
```

若要對所有路徑都失敗的情況設定佇列 I/O，請執行下列步驟：

1. 在終端機主控台中，以 `root` 使用者身分登入。
2. 在文字編輯器中開啟 `/etc/multipath.conf` 檔案。
3. 取消預設區段及其結束括號的註解，然後新增 `default_features` 設定，如下所示：

   ```
   defaults {
     default_features "1 queue_if_no_path"
   }
   ```

4. 修改 `/etc/multipath.conf` 檔案後，必須執行 `mkinitrd` 在系統中重新建立 `initrd`，然後重新開機以使變更生效。
5. 當您準備好返回到設備 I/O 的容錯移轉時，請輸入：

   ```
   dmsetup message mapname 0 fail_if_no_path
   ```

   用設備對應的別名或設備 ID 取代 *mapname*。

   此指令會立即使所有排入佇列的 I/O 失敗，並將錯誤傳播到呼叫應用程式。

7.14 解決擱置的 I/O

如果所有路徑同時失敗，且 I/O 雖已排入佇列卻被擱置時，請執行下列步驟：

1. 在終端機主控台提示符處輸入以下指令：

   ```
   dmsetup message mapname 0 fail_if_no_path
   ```

用設備的正確設備 ID 或對應的別名取代 *mapname*。這會使所有排入佇列的 I/O 失敗，並將錯誤傳播到呼叫應用程式。

2 在終端機主控台提示符處輸入以下指令重新啟動佇列：

```
dmsetup message mapname 0 queue_if_no_path
```

7.15 其他資訊

如需有關在 SUSE Linux Enterprise Server 上設定並使用多重路徑 I/O 的詳細資訊，請參閱 Novell 支援知識庫中的下列其他資源：

- *How to Setup/Use Multipathing on SLES (如何在 SLES 中設定/使用多重路徑)* [http://support.novell.com/techcenter/sdb/en/2005/04/sles_multipathing.html]
- *Troubleshooting SLES Multipathing (MPIO) Problems (Technical Information Document 3231766) (SLES 多重路徑 (MPIO) 問題的疑難排解 (技術資訊文件 3231766))* [http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=3231766&sliceId=SAL_Public]
- *Dynamically Adding Storage for Use with Multipath I/O (Technical Information Document 3000817) (動態新增儲存以使用多重路徑 I/O (技術資訊文件 3000817))* [https://secure-support.novell.com/KanisaPlatform/Publishing/911/3000817_f.SAL_Public.html]
- *DM MPIO Device Blacklisting Not Honored in multipath.conf (Technical Information Document 3029706) (multipath.conf 中未遵從 DM MPIO 設備黑名單 (技術資訊文件 3029706))* [http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=3029706&sliceId=SAL_Public&dialogID=57872426&stateId=0%200%2057878058]
- *Static Load Balancing in Device-Mapper Multipathing (DM-MP) (Technical Information Document 3858277) (設備對應程式多重路徑 (DM-MP) 中的靜態負載平衡 (技術資訊文件 3858277))* [http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=3858277

&sliceId=SAL_Public&dialogID=57872426&stateId=0%200%2057878058]

- *Troubleshooting SCSI (LUN) Scanning Issues (Technical Information Document 3955167) (SCSI (LUN) 掃描問題疑難排解 (技術資訊文件 3955167))* [http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=3955167&sliceId=SAL_Public&dialogID=57868704&stateId=0%200%2057878206]

7.16 還有什麼功能?

若要使用軟體RAID，請先建立RAID並進行設定，然後再在設備中建立檔案系統。如需資訊，請參閱以下章節：

- 第 8 章「*軟體 RAID 組態*」[111頁]
- 第 10 章「*使用 mdadm 管理軟體 RAID 6 和 10*」[125頁]

8

軟體 RAID 組態

RAID (獨立磁碟容錯陣列，Redundant Array of Independent Disks) 的用途是將數個硬碟分割區組合成一個大型「虛擬」硬碟，以達最佳化效能、資料安全性或是兩者兼具的功能。大部分 RAID 控制器使用 SCSI 通訊協定，因為它可利用比 IDE 通訊協定更有效的方式處理較大量的硬碟，並且更適合指令的平行處理。有部分的 RAID 控制器支援 IDE 或 SATA 硬碟。軟體 RAID 可提供 RAID 系統的優點，卻不需要硬體 RAID 控制器的額外成本。但是這需要一些 CPU 時間，而且有一些記憶體需求，使它不適用於極高效能的電腦。

重要

OCFS2 等叢集檔案系統不支援軟體 RAID，因為 RAID 不支援同時啟動功能。如果要讓 RAID 適用於 OCFS2，則 RAID 需由儲存子系統處理。

SUSE® Linux Enterprise 可讓您選擇將幾個硬碟組合為一個軟體 RAID 系統。RAID 一詞是表示將數個硬碟結合成 RAID 系統的一些策略，每個都有不同的目標、優點及特色。這些變化通常稱為 *RAID 層級*。

8.1 瞭解 RAID 層級

本節說明通用 RAID 層級 0、1、2、3、4、5 以及巢狀 RAID 層級。

- 第 8.1.1 節「RAID 0」[112頁]
- 第 8.1.2 節「RAID 1」[112頁]
- 第 8.1.3 節「RAID 2 與 RAID 3」[113頁]
- 第 8.1.4 節「RAID 4」[113頁]
- 第 8.1.5 節「RAID 5」[113頁]
- 第 8.1.6 節「巢狀 RAID 層級」[113頁]

8.1.1 RAID 0

此層級將每個檔案的區塊分散於多個磁碟，以提升您的資料存取效能。實際上，它不是真的 RAID，因為它不提供資料備份，但是此類型系統的名稱 *RAID 0* 已經成為標準。使用 RAID 0，就可將兩個以上的磁碟聚集在一起。效能非常好，但是如果其中一個硬碟錯誤，RAID 系統便會損毀而且資料會遺失。

8.1.2 RAID 1

此層級對資料提供足夠的安全性，因為資料是以 1:1 複製到另一個硬碟。這就是所謂的*硬碟鏡射*。如果其中一個磁碟損毀，另一個鏡像複製磁碟上有其內容的複本。如果所有其他磁碟都受到損害，但有一個可用，就不會危害到資料。但是，如果未偵測到損毀的情況，則損毀的資料也可能會鏡射到正確的磁碟，因而造成資料損毀。在複製過程中的寫入效能比使用單一磁碟存取時稍差一些 (慢了百分之十到二十)，但是讀取存取卻較任何一般實體硬碟快得多，因為資料已複製，因此可以平行掃描。一般而言，可以說 RAID 1 比單一磁碟的讀取異動率快了將近兩倍，而且與單一磁碟的寫入異動率幾乎相同。

8.1.3 RAID 2 與 RAID 3

這些都不是一般的RAID實作。「層級 2」在是位元層級分割資料，而不是在區塊層級。「層級3」提供具有專用同位磁碟的位元層級分割，但是無法同時服務多個要求。這兩個層級都很少使用。

8.1.4 RAID 4

「層級4」提供與「層級0」相同的區塊層級分割，並且結合專用的同位磁碟。在資料磁碟失敗時，會使用同位資料建立替代的磁碟。不過，同位磁碟可能造成寫入存取的瓶頸。儘管如此，有時還是會使用「層級 4」。

8.1.5 RAID 5

RAID 5 是在「層級 0」與「層級 1」之間效能和備用方面最佳的折衷方法。硬碟空間等於使用的磁碟數減一。使用RAID 0可將資料分布至各個硬碟。在其中一個分割區上建立的*同位區塊*是基於安全性考量。它們以XOR互相連結，使得系統失敗時，能夠藉由對應的同位區塊重新建構內容。使用RAID 5，不會有一個以上的硬碟同時失敗。如果一個硬碟失敗，必須立即更換以避免資料遺失的風險。

8.1.6 巢狀 RAID 層級

已經開發一些其他的 RAID 層級，如 RAIDn、RAID 10、RAID 0+1、RAID 30 和 RAID 50 等。有些是硬體廠商所建立的專用實作。這些層級並不是很普遍，因此在這裡不做說明。

8.2 使用 YaST 進行軟體 RAID 組態

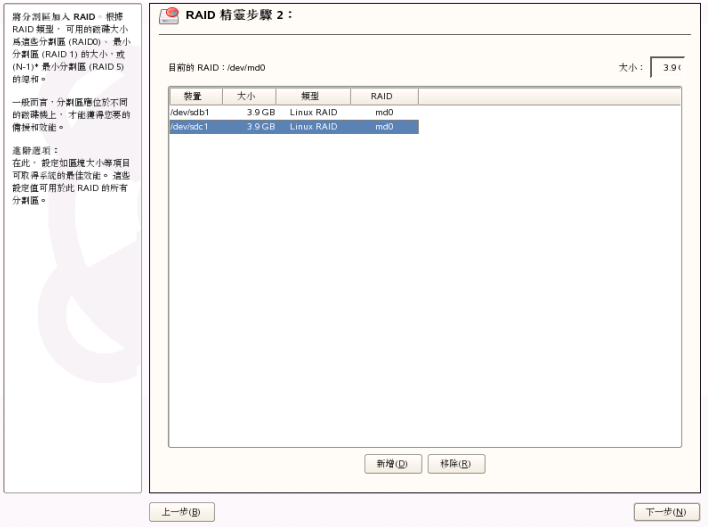
YaST 軟體 RAID 組態可以透過 YaST 進階磁碟分割程式完成。這個磁碟分割工具可讓您編輯和刪除現有磁碟分割，以及建立應該與軟體 RAID 一起使用的新磁碟分割。

您可以按一下「*建立*」>「*不格式化*」，然後選取「*0xFD Linux RAID*」做為分割區識別碼，來建立 RAID 分割區。對於 RAID 0 和 RAID 1，至少需要兩個分

割區。對於 RAID 1，通常剛好兩個而不需更多。如果使用 RAID 5，至少需要三個分割區。建議您僅使用大小相同的分割區，因為每個節區只能提供相同容量的空間做為最小的分割區。RAID 分割區應該儲存在不同的硬碟上，以減少其中一個損壞時 (RAID 1 和 5) 遺失資料的風險，並最佳化 RAID 0 的效能。在建立 RAID 使用的所有分割區後，按一下「*RAID*」>「*建立 RAID*」以啟動 RAID 組態。

在下個對話方塊中，在 RAID 層級 0、1 和 5 之間進行選擇，然後按「*下一步*」。以下對話方塊 (請參閱 圖形 8.1「RAID 分割區」[114頁]) 列出類型為「*Linux LVM*」或「*Linux Native*」的所有分割區。不會顯示交換和 DOS 分割區。如果已經指定分割區給 RAID 磁碟區，RAID 設備的名稱 (例如 `/dev/md0`) 就會顯示在清單中。未指定的分割區以「--」表示。

圖形 8.1 *RAID 分割區*



若要將之前未指定的分割區新增給所選的 RAID 磁碟區，請先選取分割區，然後按一下「*新增*」。此時，RAID 設備的名稱會顯示於所選分割區的旁邊。指定保留給 RAID 的所有分割區。否則，仍然不會使用分割區上的空間。在指定所有的分割區後，按一下「*下一步*」以進入設定對話方塊，在此您可以微調效能 (請參閱 圖形 8.2「檔案系統設定」[115頁]。)

圖形 8.2　檔案系統設定

使用傳統磁碟分割時，會設定要使用的檔案系統、加密以及 RAID 磁碟區的掛接點。在按一下「*完成*」以完成組態之後，請參閱 `/dev/md0` 設備及其他在進階磁碟分割程式中以 *RAID* 表示的設備。

8.3 疑難排解

檢查檔案 `/proc/mdstat` 以確定 RAID 分割區是否已損毀。當系統失敗時，請關閉 Linux 系統並使用以相同方式磁碟分割的新硬碟來更換損壞的硬碟。然後重新啟動系統，並輸入 `mdadm /dev/mdX --add /dev/sdX` 指令。使用特定的設備識別碼取代 `X`。如此可將硬碟自動整合到 RAID 系統並完整地重新建構。

雖然您可以在重建期間存取所有資料，但是在完全重建 RAID 之前，可能會遇到一些效能問題。

8.4 如需更多資訊

可在下列網址的 HOWTO 中找到組態指南及軟體 RAID 的詳細資訊：

- *The Software RAID HOWTO (軟體 RAID HOWTO)* [http://en.tldp.org/HOWTO/Software-RAID-HOWTO.html]
- `/usr/share/doc/packages/mdadm/Software-RAID.HOWTO.html` 檔案中的*The Software RAID HOWTO* (軟體 RAID HOWTO)

也有 Linux RAID 郵寄清單可供參考，如 linux-raid [http://marc.theaimsgroup.com/?l=linux-raid]。

9 設定根分割區的軟體 RAID

在 SUSE® Linux Enterprise Server 11 中，設備對應程式 RAID 工具已整合到 YaST 磁碟分割程式中。您可以在安裝時使用磁碟分割程式，為包含根 (/) 分割區的系統設備建立一個軟體 RAID。

9.1 軟體 RAID 的先決條件

確認您的組態符合下列要求：

- 根據要建立的軟體 RAID 類型的不同，您需要兩個或兩個以上的硬碟。
  - **RAID 0 (分割)：** RAID 0 需要兩台或兩台以上的設備。RAID 0 並不具備容錯的優點，不建議用於系統設備。
  - **RAID 1 (鏡像)：** RAID 1 需要兩台設備。
  - **RAID 5 (備援分割)：** RAID 5 需要三台或三台以上的設備。
- 幾個硬碟的大小應相似。RAID 假設為最小磁碟機的大小。

- 區塊儲存設備可以是本地(機器中或直接連接到機器上)、光纖通道儲存子系統或 iSCSI 儲存子系統的任意組合。
- 如果您使用的是硬體 RAID 設備，請不要嘗試在其上執行軟體 RAID。
- 如果您使用的是 iSCSI 目標設備，請先啟用 iSCSI 啟動器支援，然後再建立 RAID 設備。
- 如果儲存子系統在伺服器與要用於軟體RAID的設備(直接連接的本地設備、光纖通道設備或 iSCSI 設備) 之間提供了多重 I/O 路徑，您必須先啟用多重路徑支援，然後才能建立 RAID 設備。

9.2 安裝時啟用 iSCSI 啟動器支援

如果要將某些 iSCSI 目標設備用於根 (/) 分割區，必須先啟用 iSCSI 啟動器軟體使這些設備可用，然後才能建立軟體 RAID 設備。

1. 繼續進行 SUSE Linux Enterprise 11 的 YaST 安裝，直到進入「安裝設定」頁面。
2. 按一下「*磁碟分割*」開啟「準備硬碟」頁面，按一下「*自定分割區 (進階)*」，然後按「*下一步*」。
3. 在「進階磁碟分割程式」頁面中，展開「*系統檢視*」面板中的「*硬碟*」，以檢視預設建議。
4. 在「*硬碟*」頁面中，選取「*設定*」>「*設定 iSCSI*」，然後在提示您繼續啟始化 iSCSI 啟動器組態時按一下「*繼續*」。

9.3 安裝時啟用多重路徑 I/O 支援

如果要將設備的多重 I/O 路徑用於建立根 (/) 分割區的軟體 RAID 設備，必須先啟用多重路徑支援，然後才能建立軟體 RAID 設備。

1. 繼續進行 SUSE Linux Enterprise 11 的 YaST 安裝，直到進入「安裝設定」頁面。

2 按一下「*磁碟分割*」開啟「準備硬碟」頁面，按一下「*自定分割區 (進階)*」，然後按「*下一步*」。

3 在「進階磁碟分割程式」頁面中，展開「*系統檢視*」面板中的「*硬碟*」，以檢視預設建議。

4 在「*硬碟*」頁面中，選取「*設定*」>「*設定多重路徑*」，然後在提示您啟動多重路徑時按一下「*是*」。

這會重新掃描設備並解析多重路徑，以便每台設備在硬碟清單中只列出一次。

9.4 建立根 (/) 分割區的軟體 RAID 設備

1 繼續進行 SUSE Linux Enterprise 11 的 YaST 安裝，直到進入「安裝設定」頁面。

2 按一下「*磁碟分割*」開啟「準備硬碟」頁面，按一下「*自定分割區 (進階)*」，然後按「*下一步*」。

3 在「進階磁碟分割程式」頁面中，展開「*系統檢視*」面板中的「*硬碟*」以檢視預設建議，選取建議的分割區，然後按一下「*刪除*」。

4 建立交換分割區。

4a 在「進階磁碟分割程式」頁面的「*硬碟*」下，選取要用於交換分割區的設備，然後在「*硬碟分割區*」索引標籤中按一下「*新增*」。

4b 在「*新分割區類型*」下，選取「*主分割區*」，然後按「*下一步*」。

4c 在「*新分割區大小*」下，指定要使用的大小，然後按「*下一步*」。

4d 在「*格式化選項*」下，選取「*格式化分割區*」，然後從下拉式清單中選取「*交換*」。

4e 在「*掛接選項*」下，選取「*掛接分割區*」，然後從下拉式清單中選取「*交換*」。

4f 按一下「*完成*」。

5 為要用於軟體 RAID 的每台設備設定「*0xFD Linux RAID*」格式。

5a 在「進階磁碟分割程式」頁面的「*硬碟*」下，選取要用於 RAID 的設備，然後在「*硬碟分割區*」索引標籤中按一下「*新增*」。

5b 在「*新分割區類型*」下，選取「*主分割區*」，然後按「*下一步*」。

5c 在「*新分割區大小*」下，指定使用最大大小，然後按「*下一步*」。

5d 在「*格式化選項*」下，選取「*不格式化分割區*」，然後從下拉式清單中選取「*0xFD Linux RAID*」。

5e 在「*掛接選項*」下，選取「*不掛接分割區*」。

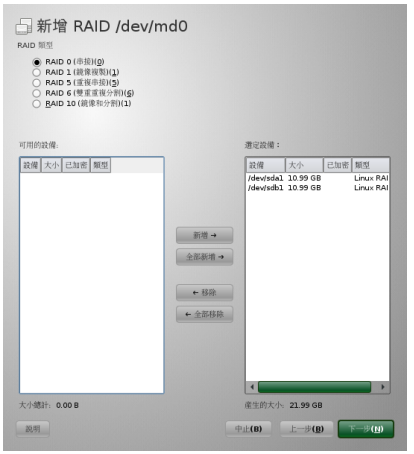
5f 按一下「*完成*」。

5g 對要用於軟體 RAID 的每台設備重複步驟 5a [120頁] 到步驟 5f [120頁]

6 建立 RAID 設備。

6a 在「*系統檢視*」面板中，選取「*RAID*」，然後在「RAID」頁面中按一下「*新增 RAID*」。

「*可用的設備*」中會列出您在步驟 5 [120頁] 中準備的設備。



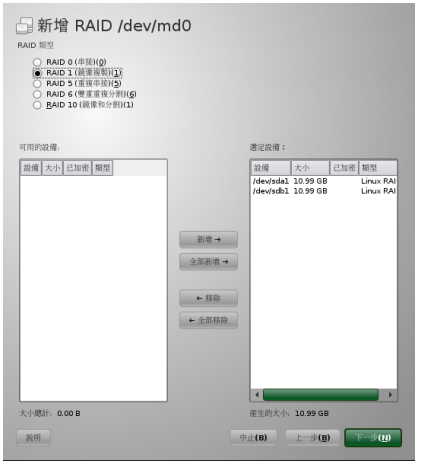
6b 在「*RAID 類型*」下，選取「*RAID 0 (分割)*」、「*RAID 1 (鏡像)*」或「*RAID 5 (備援分割)*」。

例如，選取 RAID 1 (鏡像)。

6c 在「*可用的設備*」面板中，選取要用於RAID的設備，然後按一下「*新增*」將這些設備移至「*選定設備*」面板。

為RAID 1 指定兩台或兩台以上的設備，為RAID 0 指定兩台設備，或為 RAID 5 指定至少三台設備。

若要繼續此範例，請為 RAID 1 選取兩台設備。



6d 點選「*下一步*」。

6e 在「*RAID 選項*」下的下拉式清單中選取區塊大小。

RAID 1 (鏡像) 的預設區塊大小為 4 KB。

RAID 0 (分割) 的預設區塊大小為 32 KB。

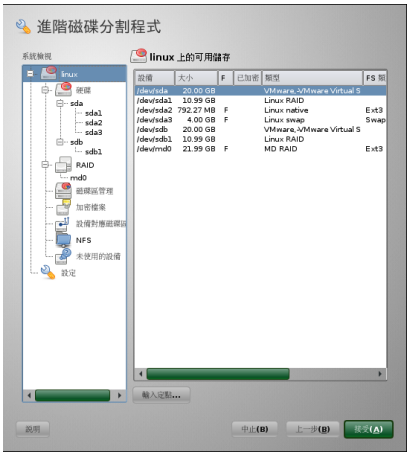
可用的區塊大小有：4 KB、8 KB、16 KB、32 KB、64 KB、128 KB、256 KB、512 KB、1 MB、2 MB 或 4 MB。

6f 在「*格式化選項*」下，選取「*格式化分割區*」，然後從「*檔案系統*」下拉式清單中選取檔案系統類型 (例如 Ext3)。

6g 在「*掛接選項*」下，選取「*掛接分割區*」，然後從「*掛接點*」下拉式清單中選取 /。

6h 按一下「*完成*」。

軟體 RAID 設備便會受設備對應程式的管理，並會在 /dev/md0 路徑下建立一台設備。



7 在「進階磁碟分割程式」頁面中按一下「*接受*」。

新的建議便會出現在「安裝設定」頁面的「*磁碟分割*」下。

例如如下設定

8 繼續安裝。

每次將伺服器重新開機時，設備對應程式都會在開機時啟動，以便讓系統能夠自動辨識軟體 RAID，並啟動根 (/) 分割區上的作業系統。

10 使用 mdadm 管理軟體 RAID 6 和 10

本章說明如何使用多個設備管理 (mdadm(8)) 工具來建立軟體 RAID 6 和 10 設備。您也可使用 mdadm 來建立 RAID 0、1、4 和 5。mdadm 工具可提供舊版程式 mdtools 與 raidtools 的功能。

10.1 建立 RAID 6

10.1.1 瞭解 RAID 6

RAID 6 從本質上說是 RAID 5 的延伸，它透過使用另一獨立的分散式同位規劃 (雙同位) 允許額外容錯。即使在資料復原程序過程中兩個硬碟機都發生故障，系統也能繼續操作，且不遺失資料。

在多個磁碟機同時發生故障的情況下，RAID 6 可提供極高的資料容錯能力。它能夠處理兩個設備的遺失，不會遺失資料。相應地，它需要 N+2 個磁碟機來儲存 N 個磁碟機的重要資料。至少需要 4 個設備。

但是，比較處於正常模式和有一個硬碟機發生故障模式下的 RAID 5，RAID 6 的效能略低。處於雙磁碟機故障模式下時，其速度會變得很慢。

表格 10.1 *比較 RAID 5 與 RAID 6*

特性	RAID 5	RAID 6
設備數目	N+1，最小值為 3	N+2，最小值為 4
同位元	分散式，單一	分散式，兩個
效能	對寫入與重建有中等程度的影響	對連續寫入的影響比 RAID 5 要大
容錯	一個元件設備發生故障	兩個元件設備發生故障

10.1.2 建立 RAID 6

此節中的程序可建立具有四個設備 (`/dev/sda1`、`/dev/sdb1`、`/dev/sdc1` 和 `/dev/sdd1`) 的 RAID 6 設備 `/dev/md0`。請確保修改此程序以使用實際設備節點。

1. 開啟終端機主控台，以 `root` 使用者或同等身分登入。
2. 建立 RAID 6 設備。在指令提示符下，輸入

   ```
   mdadm --create /dev/md0 --run --level=raid6 --chunk=128 --raid-devices=4
    /dev/sdb1 /dev/sdc1 /dev/sdc1 /dev/sdd1
   ```

   預設區塊大小為 64 KB。
3. 在 RAID 6 設備 `/dev/md0` 上建立檔案系統，如 Reiser 檔案系統 (reiserfs)。例如，在指令提示符下輸入

   ```
   mkfs.reiserfs /dev/md0
   ```

如果要使用其他檔案系統，請修改指令。

4 編輯 /etc/mdadm.conf 檔案，以新增元件設備和 RAID 設備 /dev/md0 的項目。

5 編輯 /etc/fstab 檔案以新增 RAID 6 設備 /dev/md0 的項目。

6 重新載入伺服器。

RAID 6 設備已掛接到 /local。

7 (可選) 新增熱備用以用於 RAID 陣列。例如，在指令提示符下輸入：

```
mdadm /dev/md0 –a /dev/sde1
```

10.2 使用 mdadm 建立巢狀 RAID 10 設備

10.2.1 瞭解巢狀 RAID 設備

巢狀 RAID 設備由 RAID 陣列構成，它使用其他 RAID 陣列取代實體磁碟做為其基本元素。此組態的目標是提高 RAID 的效能和容錯能力。

Linux 支援巢狀化 RAID 1 (鏡像) 與 RAID 0 (分割) 陣列。通常，此組合又稱為 RAID 10。為了辨識巢狀的順序，此文件使用下列術語：

- **RAID 1+0：** 首先建立 RAID 1 (鏡像) 陣列，然後再結合以形成 RAID 0 (分割) 陣列。
- **RAID 0+1：** 首先建立 RAID 0 (分割) 陣列，然後再結合以形成 RAID 1 (鏡像) 陣列。

下表描述 RAID 10 巢狀化為 1+0 與 0+1 的優點和缺點。假設使用的儲存物件位於不同的磁碟，且每個物件都有專屬的 I/O 功能。

表格 10.2 *巢狀 RAID 層級*

RAID 層級	描述	效能與容錯
10 (1+0)	使用 RAID 1 (鏡像) 陣列建立 RAID 0 (分割)	RAID 1+0 提供高層級的 I/O 效能、資料備援及磁碟容錯。因為 RAID 0 中的每個成員設備都是個別鏡像的，所以只要發生故障的磁碟處於不同的鏡像複製，這些故障磁碟都可進行容錯，且保持資料可用。 可以選擇性為每個基礎鏡像複製陣列設定備用，或者為服務所有鏡像複製的備用群組設定所用備用。
10 (0+1)	使用 RAID 0 (分割) 陣列建立 RAID 1 (鏡像)	RAID 0+1 提供高層級的 I/O 效能和資料備援，但容錯能力略低於 RAID 1+0。如果同側鏡像複製中的多個磁碟發生故障，則另一側的鏡像複製仍可用。但是，如果兩側鏡像複製中的磁碟同時發生故障，則會遺失所有資料。 此解決方案的磁碟容錯能力低於 1+0 解決方案，但如果要執行維護操作或維護其他位置的鏡像複製，則可使鏡像複製的一側完全處於離線狀態，此時仍可擁有儲存設備的完整功能。同樣地，如果兩個位置之間斷開連接，則每個位置都可獨立進行操作。但如果分割經過鏡像處理的區段，則上述說法不正確，因為鏡像複製的管理處於較低層級。 如果設備發生故障，則位於那側的鏡像複製將失敗，因為 RAID 1 不具容錯能力。建立新 RAID 0 以取代發生故障的那一側，然後重新同步鏡像複製。

10.2.2 使用 mdadm 建立巢狀 RAID 10 (1+0)

建立巢狀 RAID 1+0 的方法是：建立兩個或更多 RAID 1 (鏡像) 設備，然後將其做為元件設備用於 RAID 0。

重要

如果需要管理與設備的多個連線，您必須在設定 RAID 設備之前設定多重路徑 I/O。如需更多資訊，請參閱第 7 章「*管理設備的多重路徑 I/O*」 [53頁]。

此節的程序使用下表顯示的設備名稱。請確保將設備名稱修改為自己設備的名稱。

表格 10.3 *經由巢狀化建立 RAID 10 (1+0) 的案例*

Raw 設備	RAID 1 (鏡像)	RAID 1+0 (分割的鏡像)
/dev/sdb1 /dev/sdc1	/dev/md0	/dev/md2
/dev/sdd1 /dev/sde1	/dev/md1	

1. 開啟終端機主控台，以 `root` 使用者或同等身分登入。

2. 建立兩個軟體 RAID 1 設備，每個 RAID 1 設備使用兩個不同設備。在指令提示符下，輸入下列兩個指令：

   ```
   mdadm --create /dev/md0 --run --level=1 --raid-devices=2 /dev/sdb1
   /dev/sdc1
   ```

   ```
   mdadm --create /dev/md1 --run --level=1 --raid-devices=2 /dev/sdd1
   /dev/sde1
   ```

3. 建立巢狀 RAID 1+0 設備。在指令提示符下，使用您在步驟 2 [129頁] 建立的軟體 RAID 1 設備輸入以下指令：

   ```
   mdadm --create /dev/md2 --run --level=0 --chunk=64 --raid-devices=2
   /dev/md0 /dev/md1
   ```

   預設區塊大小為 64 KB。

4. 在 RAID 1+0 設備 `/dev/md2` 上建立檔案系統，如 Reiser 檔案系統 (reiserfs)。例如，在指令提示符下輸入

```
mkfs.reiserfs /dev/md2
```

如果要使用其他檔案系統，請修改指令。

5 編輯 /etc/mdadm.conf 檔案，以新增元件設備和 RAID 設備 /dev/md2 的項目。

6 編輯 /etc/fstab 檔案以新增 RAID 1+0 設備 /dev/md2 的項目。

7 重新載入伺服器。

RAID 1+0 設備已掛接到 /local。

10.2.3 使用 mdadm 建立巢狀 RAID 10 (0+1)

建立巢狀 RAID 0+1 的方法是：建立兩到四個 RAID 0 (分割) 設備，然後將其進行鏡像處理並做為元件設備用於 RAID 1。

重要

如果需要管理與設備的多個連線，您必須在設定 RAID 設備之前設定多重路徑 I/O。如需更多資訊，請參閱第 7 章「*管理設備的多重路徑 I/O*」[53頁]。

在這一組態中，因為 RAID 0 無法容錯設備遺失，所以無法為基礎 RAID 0 設備指定備用設備。如果鏡像複製某側的設備發生故障，則必須建立取代 RAID 0 設備，然後將其新增至鏡像複製。

此節的程序使用下表顯示的設備名稱。請確保將設備名稱修改為自己設備的名稱。

表格 10.4 *經由巢狀化建立 RAID 10 (0+1) 的案例*

Raw 設備	RAID 0 (分割)	RAID 0+1 (鏡像的分割)
/dev/sdb1 /dev/sdc1	/dev/md0	/dev/md2
/dev/sdd1	/dev/md1	

Raw 設備	RAID 0 (分割)	RAID 0+1 (鏡像的分割)
/dev/sde1		

1 開啟終端機主控台，然後以 root 使用者或同等身分登入。

2 建立兩個軟體 RAID 0 設備，每個 RAID 0 設備使用兩個不同設備。在指令提示符下，輸入下列兩個指令：

```
mdadm --create /dev/md0 --run --level=0 --chunk=64 --raid-devices=2
/dev/sdb1 /dev/sdc1
```

```
mdadm --create /dev/md1 --run --level=0 --chunk=64 --raid-devices=2
/dev/sdd1 /dev/sde1
```

預設區塊大小為 64 KB。

3 建立巢狀 RAID 0+1 設備。在指令提示符下，使用您在步驟 2 [131頁] 建立的軟體 RAID 0 設備輸入以下指令：

```
mdadm --create /dev/md2 --run --level=1 --raid-devices=2 /dev/md0
/dev/md1
```

4 在 RAID 0+1 設備 /dev/md2 上建立檔案系統，如 Reiser 檔案系統 (reiserfs)。例如，在指令提示符下輸入

```
mkfs.reiserfs /dev/md2
```

如果要使用其他檔案系統，請修改指令。

5 編輯 /etc/mdadm.conf 檔案，以新增元件設備和 RAID 設備 /dev/md2 的項目。

6 編輯 /etc/fstab 檔案以新增 RAID 0+1 設備 /dev/md2 的項目。

7 重新載入伺服器。

RAID 0+1 設備已掛接到 /local。

10.3 使用 mdadm 建立複雜 RAID 10

10.3.1 瞭解 mdadm RAID10

在 mdadm 中，RAID10 層級可建立單一複雜軟體 RAID，它結合了 RAID 0 (分割) 與 RAID 1 (鏡像) 的功能。所有資料區塊的多個複本按照分割原則分佈於多個設備之上。元件設備的大小應相同。

比較複雜 RAID10 與巢狀 RAID 10 (1+0)

複雜 RAID 10 與巢狀 RAID 10 (1+0) 的用途相似，但有以下幾處差異：

表格 10.5 *比較複雜與巢狀 RAID 10*

特性	mdadm RAID10 選項	巢狀 RAID 10 (1+0)
設備數目	允許元件設備的數目為偶數或奇數	要求元件設備的數目為偶數
元件設備	視為單一RAID設備進行管理	視為巢狀RAID設備進行管理
等量磁區	在元件設備近配置或遠配置中發生分割。	分割跨元件設備連續發生

特性	mdadm RAID10 選項	巢狀 RAID 10 (1+0)
	遠配置提供根據磁碟機數目 (而非 RAID 1 配對的數目) 調整的連續讀取輸送量。	
多個資料複本	兩個或更多複本，最多為陣列中的設備數	每個鏡像複製區段上的複本
熱備用設備	單一備用可用於所有元件設備	為每個基礎鏡像複製陣列設定備用，或者為服務所有鏡像複製的備用群組設定所用備用。

mdadm RAID10 中的複製本數目

設定 mdadm RAID10 陣列時，必須指定每個所需資料區塊的複製本數目。複製本的預設數目是 2，但該值可以是 2 至陣列中設備數目的任一值。

mdadm RAID10 中的設備數目

必須至少使用與指定的複製本數目相同的元件設備。但是，RAID10 陣列中的元件設備數目不必是每個資料區塊複製本數目的倍數。有效的儲存大小為設備數目除以複製本數目。

例如，如果您為使用 5 個元件設備建立的陣列指定 2 個複製本，每個區塊的複本儲存在兩個不同的設備之上。則所有資料的一個複本的有效儲存大小是元件設備大小的 5/2 或 2.5 倍。

近配置

使用近配置，資料區塊的複本被分割到不同元件設備上鄰近位置。即一個資料區塊的多個複本位於不同設備中偏移值類似的位置。近配置是 RAID10 的預設配置。例如，如果使用奇數數目的元件設備和兩個資料複本，某些複本可能是設備中的一個區塊。

mdadm RAID10 之近配置的讀與寫效能與磁碟機數目超過其半數的 RAID 0 相似。

具有偶數數目的磁碟和兩個複製本的近配置：

```
sda1 sdb1 sdc1 sde1
  0    0    1    1
  2    2    3    3
  4    4    5    5
  6    6    7    7
  8    8    9    9
```

具有奇數數目的磁碟和兩個複製本的近配置：

```
sda1 sdb1 sdc1 sde1 sdf1
  0    0    1    1    2
  2    3    3    4    4
  5    5    6    6    7
  7    8    8    9    9
  10   10   11   11   12
```

遠配置

遠配置將資料分割到所有磁碟機的較前部分，然後將資料的另一複本分割到所有磁碟機的較後部分，確保區塊的所有複本位於不同的磁碟機。第二組值啟始於元件磁碟機的中間部分。

使用遠配置，mdadm RAID10 的讀取效能與超出其全部磁碟機數目的 RAID 0 相似，但寫入效能大大低於 RAID 0，因為它會更多搜尋磁碟機開頭部分。遠配置最適用於讀密集型操作，如唯讀檔案伺服器。

raid10 的寫入速度與其他鏡像 RAID 類型相似 (例如 raid1 和使用近配置的 raid10)，因為該檔案系統的升級程式會以一種比 raw 寫入更佳的方式排程寫入操作。使用遠配置的 raid10 最適合鏡像寫入應用程式。

具有偶數數目的磁碟和兩個複製本的遠配置：

```
sda1 sdb1 sdc1 sde1
  0    1    2    3
  3    5    6    7
  . . .
  3    1    2    3
  7    4    5    6
```

具有奇數數目的磁碟和兩個複製本的遠配置：

```
sda1 sdb1 sdc1 sde1 sdf1
  0    1    2    3    4
  5    6    7    8    9
  . . .
  4    0    1    2    3
  9    5    6    7    8
```

10.3.2 使用 mdadm 建立 RAID 10

mdadm 的 RAID10 選項可建立未經巢狀化的 RAID 10 設備。如需有關 RAID10-的資訊，請參閱第 10.3 節「使用 mdadm 建立複雜 RAID 10」[132頁]。

此節的程序使用下表顯示的設備名稱。請確保將設備名稱修改為自己設備的名稱。

表格 10.6 *使用 mdadm RAID10 選項建立 RAID 10 的案例*

Raw 設備	RAID10 (近或遠分割規劃)
/dev/sdf1	/dev/md3
/dev/sdg1	
/dev/sdh1	
/dev/sdi1	

1. 在 YaST 中，於要在 RAID 中使用的設備上建立 0xFD Linux RAID 分割區，如 /dev/sdf1、/dev/sdg1、/dev/sdh1 和 /dev/sdi1。
2. 開啟終端機主控台，然後以 root 使用者或同等身分登入。
3. 建立 RAID 10 指令。在指令提示符下，輸入 (在同一行)：

   ```
   mdadm --create /dev/md3 --run --level=10 --chunk=4 --raid-devices=4
   /dev/sdf1 /dev/sdg1 /dev/sdh1 /dev/sdi1
   ```

4 在 RAID 10 設備 /dev/md3 上建立 Reiser 檔案系統。在指令提示符下，輸入

```
mkfs.reiserfs /dev/md3
```

5 編輯 /etc/mdadm.conf 檔案，以新增元件設備和 RAID 設備 /dev/md3 的項目。例如：

```
DEVICE /dev/md3
```

6 編輯 /etc/fstab 檔案以新增 RAID 10 設備 /dev/md3 的項目。

7 重新載入伺服器。

RAID 10 設備已掛接到 /raid10。

10.4 建立降級 RAID 陣列

降級陣列是指其中某些設備遺失的陣列。只有 RAID 1、RAID 4、RAID 5 和 RAID 6 支援降級陣列。這些 RAID 類型具有容錯功能，可容許遺失某些設備。降級陣列通常發生在設備故障時。也可能會出於某種目的而建立降級陣列。

RAID 類型	允許遺失插槽數目
RAID 1	只要有一個設備未遺失即可
RAID 4	一個插槽
RAID 5	一個插槽
RAID 6	一或兩個插槽

若要建立其中某些設備遺失的降級陣列，只需在設備名稱處標記遺失一詞。這會導致 mdadm 將陣列中的相應插槽保留空白。

建立 RAID 5 陣列時，mdadm 會自動建立具有額外備用磁碟機的降級陣列。這是因為在降級陣列中建立備用磁碟機通常比重新同步非降級但不乾淨的陣列上的同位要快得多。可以使用 --force 選項覆寫此功能。

如果您要建立 RAID，但要使用的其中一個設備上已有資料，則可以建立降級陣列。在這種情況下，建立具有其他設備的降級陣列，將資料從使用中的設備複製到在降級模式下執行的 RAID，再將該設備新增至 RAID，然後等候 RAID 重建，如此操作資料就會分佈到所有設備。下列程序是該處理程序的一個範例：

1 使用一個單一磁碟機 /dev/sd 1 建立降級 RAID 1 設備 /dev/md0，然後在指令提示符下輸入：

```
mdadm --create /dev/md0 -l 1 -n 2 /dev/sda1 missing
```

該設備大小應不小於您計劃新增的設備。

2 如果您要新增至鏡像複製的設備包含要移至 RAID 陣列的資料，請將該資料立即複製到在降級模式下執行的 RAID 陣列。

3 將設備新增至鏡像複製。例如，若要將 /dev/sdb1 新增至 RAID，請在指令提示符下輸入：

```
mdadm /dev/md0 -a /dev/sdb1
```

您一次只能新增一個設備。請耐心等候核心建立鏡像複製並將其完全發佈到線上，然後才能新增其他鏡像複製。

4 在指令提示符下輸入以下指令以監控建立進度：

```
cat /proc/mdstat
```

若要查看每秒重新整理一次的重建進度，請輸入

```
watch -n 1 cat /proc/mdstat
```

11 使用 mdadm 調整軟體 RAID 陣列的大小

本章描述如何使用「多設備管理」(mdadm(8)) 工具增加或減少軟體 RAID 1、4、5 或 6 設備的大小。

警告

在開始執行本節描述的任務之前，請確保您已有效備份所有資料。

11.1 瞭解調整大小處理程序

調整現有軟體 RAID 設備的大小包括增加或減少每個元件分割區所佔空間。

11.1.1 調整軟體 RAID 大小的準則

mdadm(8) 工具僅支援調整軟體 RAID 層級 1、4、5 和 6 的大小。這些 RAID 層級提供磁碟容錯功能，因此調整大小時一次可以移除一個元件分割區。原則上，可以即時調整 RAID 分割區的大小，但在執行此操作時要格外留意您的資料避免遺失。

還必須能夠調整 RAID 上之檔案系統的大小，以便充分利用設備上可用空間的變更。SUSE® Linux Enterprise Server 11 中提供了適用於檔案系統 Ext2、Ext3 及 ReiserFS 的檔案系統調整大小公用程式。該公用程式支援增加和減少大小，如下所述：

表格 11.1 *檔案系統支援調整大小*

檔案系統	公用程式	增加大小	減少大小
Ext2 或 Ext3	resize2fs	可以，僅限離線狀態	可以，僅限離線狀態
ReiserFS	resize_reiserfs	可以，線上或離線狀態均可	可以，僅限離線狀態

調整任何分割區或檔案系統的大小都存在一定的風險，可能會造成資料遺失。

警告

若要避免資料遺失，在開始執行調整大小任務之前，請務必備份資料。

11.1.2 任務綜覽

調整 RAID 大小包括下列任務。執行這些任務的順序取決於是增加還是減少大小。

表格 11.2　調整 RAID 大小的相關任務

任務	描述	增加大小時採用的順序	減少大小時採用的順序
調整每個元件分割區的大小。	增加或減少每個元件分割區的使用中大小。一次僅可移除一個元件分割區，修改其大小，然後將其傳回 RAID。	1	2
調整軟體RAID 自身大小。	RAID 無法自動知曉您對基礎元件分割區大小所做的調整 (增加或減少)。您必須通知它新大小。	2	3
調整檔案系統的大小。	您必須調整RAID上檔案系統的大小。這可能僅適用於提供調整大小工具的檔案系統，如 Ext2、Ext3、 及 ReiserFS。	3	1

11.2 增加軟體 RAID 的大小

開始之前，請參閱第 11.1 節「瞭解調整大小處理程序」[139頁] 中的準則。

- 第 11.2.1 節「增加元件分割區的大小」[141頁]
- 第 11.2.2 節「增加 RAID 陣列的大小」[143頁]
- 第 11.2.3 節「增加檔案系統的大小」[144頁]

11.2.1 增加元件分割區的大小

套用此節中的程序以增加 RAID 1、4、5 或 6 的大小。對於 RAID 中的每個元件分割區，請先將它從 RAID 移除，修改其大小，然後將它傳回 RAID，RAID 需要一定的穩定時間，隨後就可以繼續。移除分割區時，RAID 在降級模式下操

作，此時不具備磁碟容錯功能或會降低此功能。即便對於能夠容許多個磁碟同時發生故障的 RAID，也不要一次移除多個元件分割區。

警告

如果 RAID 不具備磁碟容錯功能或只是不一致，則移除任何分割區都會導致資料遺失。移除分割區時要非常小心，並確保已備份可用資料。

此節的程序使用下表顯示的設備名稱。確保修改名稱以使用自己設備的名稱。

表格 11.3 *增加元件分割區大小的案例*

RAID 設備	元件分割區
/dev/md0	/dev/sda1
	/dev/sdb1
	/dev/sdc1

若要增加 RAID 元件分割區的大小，請執行下列步驟：

1. 開啟終端機主控台，以 `root` 使用者或同等身分登入。
2. 輸入以下指令，以確保 RAID 陣列具有一致性且經過同步

   ```
   cat /proc/mdstat
   ```

   如果 RAID 陣列仍在根據指令的輸出進行同步，您必須等候同步完成，然後才能繼續。
3. 從 RAID 陣列移除一個元件分割區。例如，若要移除 `/dev/sda1`，請輸入

   ```
   mdadm /dev/md0 --fail /dev/sda1 --remove /dev/sda1
   ```

   為了操作成功，必須執行容錯和移除動作。
4. 執行下列操作之一，增加在步驟 3 [142頁] 中移除之分割區的大小：

- 使用磁碟分割程式，如 `fdisk(8)`、`cfdisk(8)` 或 `parted(8)`，增加分割區的大小。通常選擇此選項。
- 用更大容量的設備取代分割區所在的磁碟。

  僅當原始磁碟上的其他檔案系統沒有被系統存取時，該選項才可用。當取代設備被新增回 RAID 時，它需要更久的時間來同步資料，因為此時必須重建原始設備上的所有資料。

5 再次將分割區新增至 RAID 陣列。例如，若要新增 `/dev/sda1`，請輸入

```
mdadm -a /dev/md0 /dev/sda1
```

請等候直至 RAID 實現同步和一致性，然後再繼續下一分割區。

6 對陣列中的每個剩餘元件設備重複執行步驟 2 [142頁] 到步驟 5 [143頁] 的操作。確保為正確的元件分割區修改指令。

7 如果系統發出訊息告知您核心無法重新讀取 RAID 的分割區表，則必須在調整所有分割區大小之後重新開機電腦，以強制更新分割區表。

8 請繼續執行第 11.2.2 節「增加 RAID 陣列的大小」[143頁]。

11.2.2 增加 RAID 陣列的大小

在調整 RAID 中每個元件分割區的大小後 (請參閱第 11.2.1 節「增加元件分割區的大小」[141頁])，RAID 陣列組態會繼續使用原始陣列大小，直至您強制其知曉新的可用空間。您可以指定 RAID 的大小，或使用最大可用空間。

在本節的程序中，使用 RAID 設備的設備名稱 `/dev/md0`。請確保修改名稱以使用自己設備的名稱。

1 開啟終端機主控台，以 `root` 使用者或同等身分登入。

2 請輸入以下指令，以檢查陣列的大小及陣列可識別的設備大小

```
mdadm -D /dev/md0 | grep -e "Array Size" -e "Device Size"
```

3 請執行下列其中一個步驟：

- 請輸入以下指令，以將陣列的大小增加到可用的最大大小

  ```
  mdadm --grow /dev/md0 -z max
  ```

- 請輸入以下指令，以將陣列的大小增加到指定值

  ```
  mdadm --grow /dev/md0 -z size
  ```

  用適當的大小 (以 KB 為單位的整數值，1 KB 為 1024 位元組) 取代 *size*。

4 請輸入以下指令，以重新檢查陣列的大小及陣列可識別的設備大小

```
mdadm -D /dev/md0 | grep -e "Array Size" -e "Device Size"
```

5 請執行下列其中一個步驟：

- 如果已成功調整好陣列的大小，請繼續第 11.2.3 節「增加檔案系統的大小」[144頁]。
- 如果陣列未調整到預期大小，您必須重新開機，然後再次嘗試執行此程序。

11.2.3 增加檔案系統的大小

在增加陣列的大小後 (請參閱第 11.2.2 節「增加 RAID 陣列的大小」[143頁])，您就可以調整檔案系統的大小。

可以將檔案系統的大小增加到最大可用空間大小，或指定一個精確的值。為檔案系統指定精確大小時，請確保新大小符合以下條件：

- 新大小必須大於現有資料的大小；否則資料會遺失。
- 新大小不得超過目前RAID的大小，因為檔案系統大小無法超過可用空間大小。

Ext2 或 Ext3

使用 `resize2fs` 指令掛接或卸載時，可以調整 Ext2 與 Ext3 檔案系統的大小。

1. 開啟終端機主控台，以 `root` 使用者或同等身分登入。
2. 使用下列方法之一增加檔案系統的大小：

   - 若要將檔案系統的大小擴充至名為 `/dev/md0` 的軟體 RAID 設備的最大可用大小，請輸入

     ```
     resize2fs /dev/md0
     ```

     如果未指定大小參數，則預設大小為分割區的大小。

   - 若要將檔案系統擴充至指定大小，請輸入

     ```
     resize2fs /dev/md0 size
     ```

     `size` 參數可指定所需的檔案系統新大小。如果未指定單位，則大小參數的單位即為檔案系統的區塊大小。也可以選擇透過下列其中一種單位指示項給大小參數加上字尾：s 表示 512 位元組磁區；K 表示 KB (1 KB 為 1024 位元組)；M 表示 MB；G 表示 GB。

   請等候直至完成大小調整，然後再繼續。

3. 如果未掛接檔案系統，請立即掛接。

   例如，若要在掛接點 `/raid` 為名為 `/dev/md0` 的 RAID 掛接 Ext2 檔案系統，請輸入

   ```
   mount -t ext2 /dev/md0 /raid
   ```

4. 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

   ```
   df -h
   ```

   可用磁碟空間 (`df`) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或 2G。

ReiserFS

對於 Ext2 與 Ext3，掛接或卸載 ReiserFS 檔案系統時可增加其大小。調整大小的操作是在 RAID 陣列的區塊設備上完成的。

1 開啟終端機主控台，以 `root` 使用者或同等身分登入。

2 使用以下其中一種方法，增加名為 `/dev/md0` 的軟體 RAID 設備上檔案系統的大小：

- 若要將檔案系統的大小擴充至設備的最大可用大小，請輸入

  ```
  resize_reiserfs /dev/md0
  ```

  若不指定大小，該指令會將磁碟區增加至分割區的總大小。

- 若要將檔案系統擴充至指定大小，請輸入

  ```
  resize_reiserfs -s size /dev/md0
  ```

  用所需大小 (以位元組計) 取代 *`size`*。您也可以指定值的單位，例如 50000K (KB)、250M (MB) 或 2G (GB)。也可以使用加號 (+) 為值加上字首，以指定為目前大小增加的值。例如，以下指令可將 `/dev/md0` 上的檔案系統的大小增加 500 MB：

  ```
  resize_reiserfs -s +500M /dev/md0
  ```

請等候直至完成大小調整，然後再繼續。

3 如果未掛接檔案系統，請立即掛接。

例如，若要為位於掛接點 `/raid` 處名稱為 `/dev/md0` 的 RAID 掛接 ReiserFS 檔案系統，請輸入

```
mount -t reiserfs /dev/md0 /raid
```

4 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

```
df -h
```

可用磁碟空間 (df) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或 2G。

11.3 減少軟體 RAID 的大小

開始之前，請參閱第 11.1 節「瞭解調整大小處理程序」[139頁] 中的準則。

- 第 11.3.1 節「減少檔案系統的大小」[147頁]
- 第 11.3.2 節「減少元件分割區的大小」[149頁]
- 第 11.3.3 節「減少 RAID 陣列的大小」[151頁]

11.3.1 減少檔案系統的大小

當要減少 RAID 設備上檔案系統的大小時，請確定新大小滿足下列條件：

- 新大小必須大於現有資料的大小；否則資料會遺失。
- 新大小不得超過目前RAID的大小，因為檔案系統大小無法超過可用空間大小。

在 SUSE Linux Enterprise Server SP1 中，只有 Ext2、Ext3 及 ReiserFS 提供減少檔案系統大小的公用程式。請使用下文中相應的程序減少檔案系統的大小。

本節所述的程序使用的 RAID 設備名稱為 /dev/md0。請務必修改指令，以使用您自己的設備名稱。

Ext2 或 Ext3

掛接或卸載 Ext2 與 Ext3 檔案系統時，可調整其大小。

1. 開啟終端機主控台，以 root 使用者或同等身分登入。
2. 輸入以下指令，減少 RAID 上檔案系統的大小

```
resize2fs /dev/md0 <size>
```

使用所需大小的整數值 (以 KB 計) 取代 *size*。(1 KB = 1024 B。)

請等候直至完成大小調整，然後再繼續。

3 如果未掛接檔案系統，請立即掛接。例如，若要在掛接點 /raid 為名為 /dev/md0 的 RAID 掛接 Ext2 檔案系統，請輸入

```
mount -t ext2 /dev/md0 /raid
```

4 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

```
df -h
```

可用磁碟空間 (df) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或 2G。

ReiserFS

僅當卸載磁碟區時才能減少 ReiserFS 檔案系統的大小。

1 開啟終端機主控台，以 root 使用者或同等身分登入。

2 輸入以下指令卸載設備

```
umount /mnt/point
```

如果您嘗試要減少其大小的分割區包含系統檔案 (例如根 (/) 磁碟區)，僅當系統從可開機的 CD 或磁片開機時，才可以進行卸載操作。

3 輸入以下指令，減少位於軟體 RAID 設備 /dev/md0 上檔案系統的大小

```
resize_reiserfs -s size /dev/md0
```

用所需大小(以位元組計)取代 *size*。您也可以指定值的單位，例如50000K (KB)、250M (MB) 或 2G (GB)。您也可以使用減號 (-) 為該值加上字首，以指定為目前大小減少的值。例如，以下指令可將 /dev/md0 上檔案系統的大小減少 500 MB：

```
resize_reiserfs -s -500M /dev/md0
```

請等候直至完成大小調整，然後再繼續。

4 輸入以下指令掛接檔案系統

```
mount -t reiserfs /dev/md0 /mnt/point
```

5 請輸入以下指令，以檢查在已掛接檔案系統上調整大小的效果

```
df -h
```

可用磁碟空間 (df) 指令可顯示磁碟的總大小、使用的區塊數以及檔案系統上可用的區塊數。-h選項會以較易理解的格式列印大小，如1K、234M或2G。

11.3.2 減少元件分割區的大小

每次調整 RAID 的一個元件分割區大小。對於每個元件分割區，您可以將其從 RAID 移除，修改其分割區的大小，將分割區返回至 RAID，然後等待 RAID 穩定下來。移除分割區時，RAID在降級模式下操作，此時不具備磁碟容錯功能或會降低此功能。即使對於那些可以容許多個磁碟失敗同時發生的 RAID 而言，您也決不能一次即移除一個以上的元件分割區。

警告

如果 RAID 不具備磁碟容錯功能或只是不一致，則移除任何分割區都會導致資料遺失。移除分割區時要非常小心，並確保已備份可用資料。

此節的程序使用下表顯示的設備名稱。請務必修改指令，以使用您自己的設備名稱。

表格 11.4 *增加元件分割區大小的案例*

RAID 設備	元件分割區
/dev/md0	/dev/sda1

RAID 設備	元件分割區
	/dev/sdb1
	/dev/sdc1

若要調整 RAID 元件分割區的大小，請執行下列步驟：

1. 開啟終端機主控台，以 root 使用者或同等身分登入。
2. 輸入以下指令，以確保 RAID 陣列具有一致性且經過同步

   ```
   cat /proc/mdstat
   ```

   如果 RAID 陣列仍在根據指令的輸出進行同步，您必須等候同步完成，然後才能繼續。
3. 從 RAID 陣列移除一個元件分割區。例如，若要移除 /dev/sda1，請輸入

   ```
   mdadm /dev/md0 --fail /dev/sda1 --remove /dev/sda1
   ```

   為了操作成功，必須執行容錯和移除動作。
4. 執行下列操作之一，增加在步驟 3 [142頁] 中移除之分割區的大小：

   - 使用 fdisk、cfdisk 或 parted 等磁碟分割程式增加分割區的大小。
   - 用其他設備取代分割區所在的磁碟。

     僅當原始磁碟上的其他檔案系統沒有被系統存取時，該選項才可用。被取代的設備重新新增到 RAID 時，同步化資料所需的時間會更長。
5. 再次將分割區新增至 RAID 陣列。例如，若要新增 /dev/sda1，請輸入

   ```
   mdadm -a /dev/md0 /dev/sda1
   ```

   請等候直至 RAID 實現同步和一致性，然後再繼續下一分割區。

6 對陣列中的每個剩餘元件設備重複執行步驟 2 [142頁] 到步驟 5 [143頁] 的操作。確保為正確的元件分割區修改指令。

7 如果您收到一條訊息，告知核心無法重新讀取 RAID 的分割區表，則您必須在調整所有元件分割區的大小後將電腦重新開機。

8 請繼續執行第 11.3.3 節「減少 RAID 陣列的大小」[151頁]。

11.3.3 減少 RAID 陣列的大小

當您調整了 RAID 中各元件分割區的大小後，RAID 陣列組態會繼續使用原始的陣列大小，除非您強制讓其瞭解到可用的新空間。使用 --grow 選項可強制 RAID 讀取可用磁碟大小的變更。您可以指定 RAID 的大小，或使用最大可用空間。

在本節的程序中，使用 RAID 設備的設備名稱 `/dev/md0`。請務必修改指令，以使用您自己的設備名稱。

1 開啟終端機主控台，以 `root` 使用者或同等身分登入。

2 請輸入以下指令，以檢查陣列的大小及陣列可識別的設備大小

```
mdadm -D /dev/md0 | grep -e "Array Size" -e "Device Size"
```

3 請執行下列其中一個步驟：

- 輸入以下指令將陣列大小減少至最大可用大小

  ```
  mdadm --grow /dev/md0 -z max
  ```

- 輸入以下指令將陣列大小減少至指定值

  ```
  mdadm --grow /dev/md0 -z size
  ```

  使用所需大小的整數值 (以 KB 計) 取代 *size*。(1 KB = 1024 B。)

4 請輸入以下指令，以重新檢查陣列的大小及陣列可識別的設備大小

```
mdadm -D /dev/md0 | grep -e "Array Size" -e "Device Size"
```

5 請執行下列其中一個步驟：

- 如果陣列的大小成功調整，那麼您即完成了全部操作。
- 如果陣列未調整到預期大小，您必須重新開機，然後再次嘗試執行此程序。

12 iSNS for Linux

儲存區域網路 (SAN) 可包含許多在複雜網路中散佈的磁碟機。這可能會使探查及擁有設備變得困難。iSCSI 啟動程式必須可識別 SAN 中的儲存資源，並確定這些資源是否已進行存取。

網際網路儲存名稱服務 (iSNS) 是一項標準服務，從 SUSE® Linux Enterprise Server (SLES) 10 SP 2 即開始支援。iSNS 有助於自動探查、管理及設定 TCP/IP 網路上的 iSCSI 設備。iSNS 提供可與光纖通道媲美的智能儲存探查與管理服務。

重要

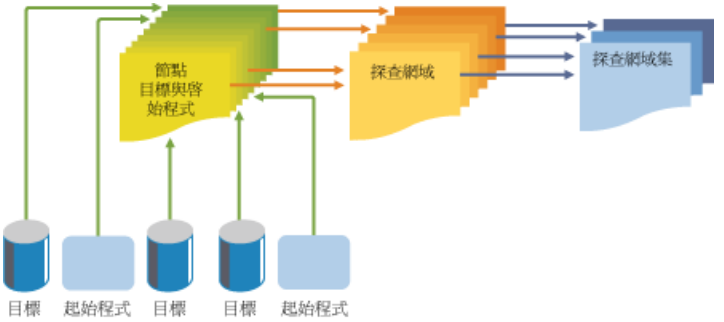
iSNS 只可用於安全的內部網路中。

12.1 iSNS 的工作原理

若要讓iSCSI啟動程式探查iSCSI目標，則需要識別網路中屬於儲存資源的設備及需要存取的 IP 位址。對於 iSNS 伺服器的查詢會傳回應用程式有權存取的 iSCSI 目標與 IP 位址。

透過使用 iSNS，您就可以建立 iSNS 探查網域與探查集。然後將 iSCSI 目標與啟動程式分組或組織到探查網域中，並將探查網域分組到探查網域集中。透過將儲存節點劃分為網域，您就可以將每台主機的探查程序限定為使用iSNS註冊的目標之最合適的子集，這樣就可透過減少不必要的探查數量並限制每台主機用於建立探查關係所耗費的時間讓儲存網路進行縮放。此操作可讓您控制並簡化必須進行探查的目標與啟動程式的數量。

圖形 12.1　iSNS 探查網域與探查網域集



iSCSI 目標與 iSCSI 啟動程式都是使用 iSNS 用戶端透過 iSNS 通訊協定啟動與 iSNS 伺服器的異動。然後在常見探查網域中註冊設備屬性資訊，下載其他註冊用戶端相關的資訊，並接收發生在探查網域中的事件之非同步通知。

iSNS 伺服器會回應 iSNS 用戶端使用 iSNS 通訊協定作出的 iSNS 通訊協定查詢與申請。iSNS 伺服器會啟動 iSNS 通訊協定狀態變更通知，並將註冊申請提交的經適當驗證的資訊儲存到 iSNS 資料庫中。

iSNS for Linux 提供的部分利益包括：

- 為註冊、探查與管理網路內的儲存資產帶來資訊便利。
- 與 DNS 基礎結構相整合
- 合併 iSCSI 儲存的註冊、探查與管理。

- 簡化了儲存管理實作。
- 與其他探查方法相比，提高了擴充性。

透過以下的情況可以讓您更了解 iSNS 所能提供的利益。

假設您擁有一個包含 100 個 iSCSI 啟動程式與 100 個 iSCSI 目標的公司。根據您的配置，所有 iSCSI 啟動程式可能會嘗試探查並連接到 100 個 iSCSI 目標中的任一一個。這樣可能會造成探查與連線方面的問題。透過將啟動程式與目標分組到探查網域中，您就可以阻止一個部門中的 iSCSI 啟動程式探查另一個部門中的 iSCSI 目標。導致特定部門中的 iSCSI 啟動程式僅會探查屬於該部門探查網域的 iSCSI 目標。

12.2 安裝 iSNS Server for Linux

iSNS Server for Linux 隨附於 SLES 10 SP2 及更高版本中，但預設不會對其加以安裝或設定。您必須安裝 iSNS 套件模組 (`isns` 與 `yast2-isns` 模組) 並設定 iSNS 服務。

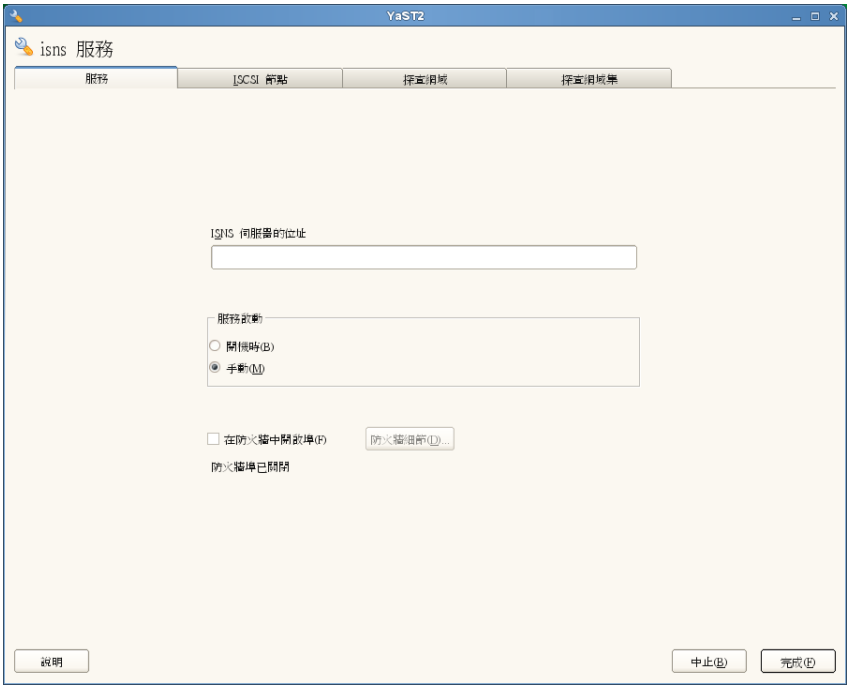
注意

可將 iSNS 安裝在 iSCSI 目標或 iSCSI 起始程式軟體所在的伺服器上。不能將 iSCSI 目標軟體與 iSCSI 起始程式軟體安裝在同一部伺服器上。

安裝 iSNS for Linux：

1. 啟動 YaST 並選取「*網路服務*」>「*iSNS 伺服器*」。
2. 提示安裝 `isns` 套件時按一下「*安裝*」。
3. 依照安裝對話方塊的指示提供 SUSE Linux Enterprise Server 11 安裝磁碟。

   安裝完成後，iSNS 服務組態對話方塊會自動開啟至「*服務*」索引標籤。



4 在「*iSNS 伺服器位址*」中，指定 iSNS 伺服器的 DNS 名稱或 IP 位址。

5 在「*服務啟動*」中選取下列其中一項：

- **開機時：** iSNS 服務在伺服器啟動時自動啟動。
- **手動 (預設)：** 必須在用於安裝 iSNS 服務之伺服器的主控台中手動輸入 `rcisns start` 或 `/etc/init.d/isns start`，從而啟動 iSNS 服務。

6 指定下列防火牆設定：

- **在防火牆中開啟埠：** 選取核取方塊開啟防火牆，並允許從遠端電腦存取服務。預設關閉防火牆連接埠。

- **防火牆詳細資料：** 如果開啟防火牆連接埠，則依預設會在所有網路介面上開啟連接埠。按一下「*防火牆詳細資料*」選取要在其上開啟連接埠的介面，並選取要使用的網路，然後按一下「*確定*」。

7 按一下「*完成*」可套用組態設定並完成安裝。

8 請繼續執行第 12.3 節「設定 iSNS 探查網域」 [157頁]。

12.3 設定 iSNS 探查網域

若要讓 iSCSI 啟動器與 iSCSI 目標使用 iSNS 服務，則它們必須屬於探查網域。

重要

必須已安裝並執行 SNS 服務，才能設定 iSNS 探查網域。如需更多資訊，請參閱第 12.4 節「啟動 iSNS」 [162頁]。

- 第 12.3.1 節「建立 iSNS 探查網域」 [157頁]
- 第 12.3.2 節「建立 iSNS 探查網域集」 [159頁]
- 第 12.3.3 節「將 iSCSI 節點新增至探查網域」 [160頁]
- 第 12.3.4 節「將探查網域新增至探查網域集」 [162頁]

12.3.1 建立 iSNS 探查網域

安裝 iSNS 服務時，會自動建立名稱為「*default DD*」的預設探查網域。設定使用 iSNS 的現有 iSCSI 目標與啟動程式會自動新增至預設探查網域。

建立新的探查網域：

1 啟動 YaST，然後在「*網路服務*」之下選取「*iSNS 伺服器*」。

2 按一下「*探查網域*」索引標籤。

「*探查網域*」區域列出了所有探查網域。您可以建立新的探查網域，也可以刪除現有探查網域。刪除網域會移除其成員，但不會刪除 iSCSI 節點成員。

「*探查網域成員*」區域列出了為所選探查網域指定的所有 iSCSI 節點。如果選取其他探查網域，則會重新整理清單，並列出該探查網域的成員。您可以對選取的探查網域新增或刪除 iSCSI 節點。如果刪除 iSCSI 節點，則會將其從網域中移除，但不會刪除該 iSCSI 節點。

如果建立 iSCSI 節點，即可將尚未註冊的節點新增為探查網域成員。該節點經 iSCSI 啟動器或目標註冊後，就會成為此網域的一部分。

iSCSI啟動器執行探查申請時，iSNS服務會傳回屬於同一探查網域的所有 iSCSI 節點目標。

3 按一下「*建立探查網域*」按鈕。

您也可以選取現有的探查網域，然後按一下「*刪除*」按鈕移除該探查網域。

4 指定您正在建立的探查網域之名稱，然後按一下「*確定*」。

5 請繼續執行第 12.3.2 節「建立 iSNS 探查網域集」[159頁]。

12.3.2 建立 iSNS 探查網域集

探查網域必須屬於探查網域集。您可以建立探查網域並將節點新增至該探查網域，但是不起作用，並且iSNS服務也不起作用，除非您將探查網域新增至探查網域集。安裝 iSNS 時會自動建立名稱為「*default DDS*」的預設探查網域集，並且預設探查網域會自動新增至該網域集。

建立探查網域集：

1 啟動 YaST，然後在「*網路服務*」之下選取「*iSNS 伺服器*」。

2 按一下「*探查網域集*」索引標籤。

「*探查網域集*」區域列出了所有探查網域集。探查網域必須是探查網域集的成員，否則無法使用。

在 iSNS 資料庫中，探查網域集包含多個探查網域，而後者又包含多個 iSCSI 節點成員。

「*探查網域集成員*」區域列出了為所選探查網域集指定的所有探查網域。如果選取其他探查網域集，則會重新整理清單，並列出該探查網域集的成員。您可以對選取的探查網域集新增或刪除探查網域。如果移除探查網域，會將其從網域集中移除，但不會刪除該探查網域。

如果向網域集新增探查網域，即可將尚未註冊的 iSNS 探查網域新增為探查網域集的成員。

3 按一下「*建立探查網域集*」按鈕。

您也可以選取現有的探查網域集，然後按一下「*刪除*」按鈕移除該探查網域集。

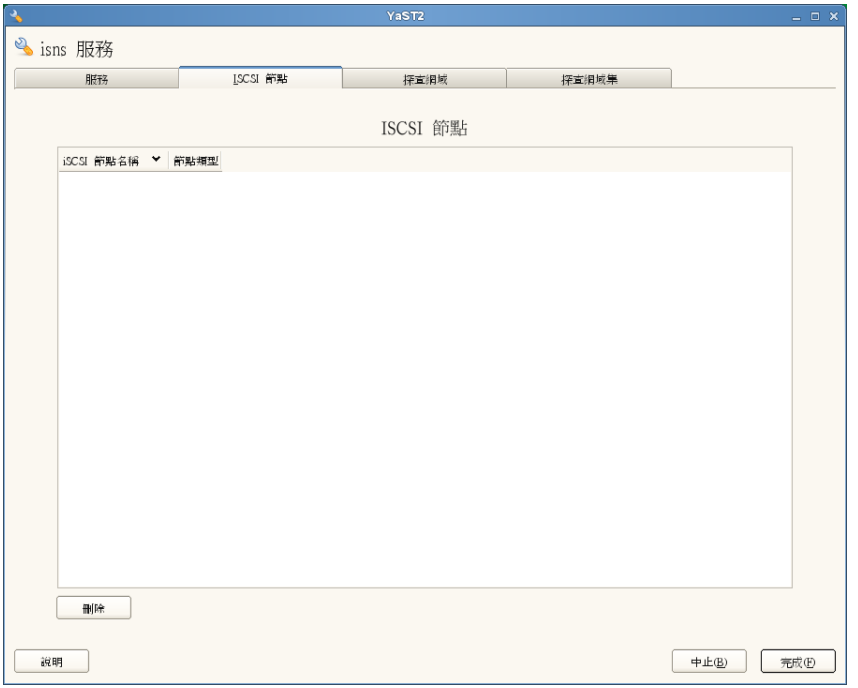
4 指定您正在建立的探查網域集之名稱，然後按一下「*確定*」。

5 請繼續執行第 12.3.3 節「將 iSCSI 節點新增至探查網域」[160頁]。

12.3.3 將 iSCSI 節點新增至探查網域

1 啟動 YaST，然後在「*網路服務*」之下選取「*iSNS 伺服器*」。

2 按一下「*iSCSI 節點*」索引標籤。



3 檢閱節點清單，確保已列出要使用 iSNS 服務的 iSCSI 目標和啟動器。

若未列出 iSCSI 目標或啟動程式，您可能需要重新啟動節點上的 iSCSI 服務。您可以透過執行 `rcopen-iscsi restart` 指令重新啟動啟動程式或 `rciscsitarget restart` 指令重新啟動目標執行此作業。

您可以選取 iSCSI 節點，然後按一下「*刪除*」按鈕將該節點從 iSNS 資料庫移除。若您不再使用 iSCSI 節點或已對該節點進行重新命名，這會帶來幫助。

除非您移除 iSCSI 組態檔案的 iSNS 部分或將其注解化，否則在重新啟動 SCSI 服務或伺服器時，iSCSI 節點會自動再次新增至清單 (iSNS 資料庫)。

4 按一下「*探查網域*」索引標籤，選取所需的探查網域，然後再按「*顯示成員*」按鈕。

5 按一下「*新增現有的 iSCSI 節點*」，選取您要新增至網域的節點，然後再按「*新增節點*」。

6 對要新增至探查網域的所有節點重複步驟 5 [162頁]，完成新增後按一下「*完成*」。

一個 iSCSI 節點可屬於多個探查網域。

7 請繼續執行第 12.3.4 節「將探查網域新增至探查網域集」[162頁]。

12.3.4 將探查網域新增至探查網域集

1 啟動 YaST，然後在「*網路服務*」之下選取「*iSNS 伺服器*」。

2 按一下「*探查網域集*」索引標籤。

3 選取「*建立探查網域集*」可將新集新增至探查網域集清單。

4 選取要修改的探查網域集。

5 按一下「*新增探查網域*」，選取要新增至探查網域集的探查網域，再按「*新增探查網域*」。

6 為要新增至探查網域集的探查網域重複最後一步，然後按一下「*完成*」。

一個探查網域可屬於多個探查網域集。

12.4 啟動 iSNS

iSNS 必須在所安裝到的伺服器上啟動。以 `root` 使用者身分在終端機主控台中輸入以下其中一則指令：

```
rcisns start
```

```
/etc/init.d/isns start
```

您也可以使用 iSNS 的 `stop`、`status` 與 `restart` 選項。

iSNS 也可設定為在每次伺服器重新開機時自動啟動：

1 啟動 YaST，然後在「*網路服務*」之下選取「*iSNS 伺服器*」。

2 選取了「*服務*」索引標籤之後，指定 iSNS 伺服器的 IP 位址，然後按一下「*儲存位址*」。

3 在螢幕的「*服務啟動*」區段中，選取「*開機時*」。

您也可以選擇手動啟動 iSNS 伺服器。每次伺服器重新啟動時，您必須使用 `rcisns start` 指令啟動服務

12.5 停止 iSNS

iSNS 必須在執行其所在的伺服器上停止。以 `root` 使用者身分在終端機主控台中輸入以下其中一則指令：

```
rcisns stop
```

```
/etc/init.d/isns stop
```

12.6 有關更多資訊

如需相關資訊，請參閱 *Linux iSNS for iSCSI project* [http://sourceforge.net/projects/linuxisns/]。這個專案的電子郵件清單為 *Linux iSNS - Discussion (Linux iSNS - 討論)* [http://sourceforge.net/mailarchive/forum.php?forum_name=linuxisns-discussion]。

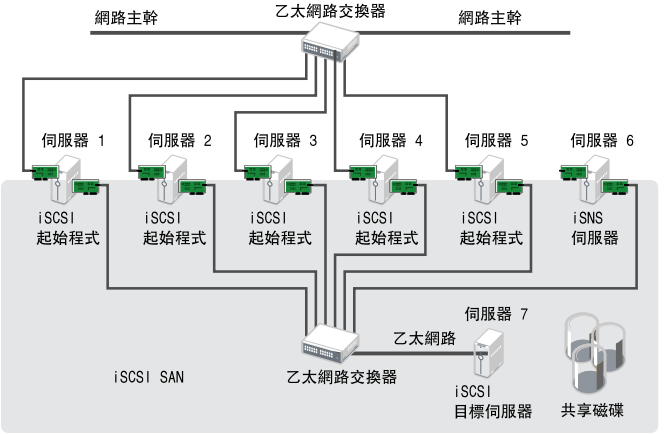
如需關於 iSNS 的一般資訊，請參閱 *RFC 4171：網際網路儲存設備名稱服務* [http://www.ietf.org/rfc/rfc4171]。

IP 網路上的大型儲存設備：iSCSI

13

如何為伺服器系統提供硬碟容量，是電腦中心以及操作伺服器時的一項關鍵任務。為此通常使用光纖通道。iSCSI (網際網路 SCSI) 解決方案的實施成本較低，可用來替代光纖通道，並能充分發揮商用伺服器與乙太網路設備的價值。Linux iSCSI 提供 iSCSI 啟動器和目標軟體，用於將 Linux 伺服器連接至中央儲存系統。

圖形 13.1　配有 iSNS 伺服器的 iSCSI SAN



iSCSI 是一種儲存網路通訊協定，可以透過 TCP/IP 網路在區塊儲存設備與伺服器之間傳輸 SCSI 封包的資料。iSCSI 目標軟體在目標伺服器上執行，並將邏輯單元定義為 iSCSI 目標設備。iSCSI 啟動器軟體在不同伺服器上執行，然後會連接到目標設備，以使該伺服器上的儲存設備可用。

重要

在線上環境中，不能在同一部伺服器上執行 iSCSI 目標軟體與 iSCSI 啟動器軟體。

iSCSI 目標與啟動器伺服器之間透過在 LAN 的 IP 層級上傳送 SCSI 封包這種方式進行通訊。如果在啟動器伺服器上執行的應用程式發出iSCSI目標設備查詢，作業系統會產生必要的 SCSI 指令。接著，通常稱為 *iSCSI 啟動器*的軟體會根據需要將 SCSI 指令嵌入 IP 封包並加密。這些封包隨後會透過內部 IP 網路傳輸到對應的 iSCSI 遠端工作站，稱為 *iSCSI 目標*。

許多儲存解決方案提供透過 iSCSI 的存取方式，但還另一種可能就是執行提供 iSCSI 目標的 Linux 伺服器。在這種情況下，針對檔案系統服務設定最佳化的 Linux 伺服器是很重要的。iSCSI 目標會存取 Linux 中的區塊設備，因此，您可以使用 RAID 解決方案來增加磁碟空間及大量記憶體來提高資料快取效率。如需關於 RAID 的詳細資訊，請參閱第 8 章「*軟體 RAID 組態*」[111頁]。

- 第 13.1 節「安裝 iSCSI」[166頁]
- 第 13.2 節「設定 iSCSI 目標」[168頁]
- 第 13.3 節「設定 iSCSI 啟動程式」[176頁]

13.1 安裝 iSCSI

YaST 包含 iSCSI 目標和 iSCSI 啟動器軟體的項目，但預設不會安裝這些套件。

重要

在線上環境中，不能在同一部伺服器上執行 iSCSI 目標軟體與 iSCSI 啟動器軟體。

- 第 13.1.1 節「安裝 iSCSI 目標軟體」[167頁]
- 第 13.1.2 節「安裝 iSCSI 啟動器軟體」[167頁]

13.1.1 安裝 iSCSI 目標軟體

將 iSCSI 目標軟體安裝到要建立 iSCSI 目標設備的伺服器上。

1. 開啟 YaST，並以 `root` 使用者身分登入。
2. 選取「*網路服務*」>「*iSNS 目標*」。
3. 當系統提示您安裝 `iscsitarget` 套件時按一下「*安裝*」。
4. 依照畫面上的安裝指示操作，並在需要時提供安裝媒體。

   安裝完成後，YaST 會開啟至「iSCSI 目標綜覽」頁面的「*服務*」索引標籤中。
5. 請繼續執行第 13.2 節「設定 iSCSI 目標」[168頁]。

13.1.2 安裝 iSCSI 啟動器軟體

將iSCSI啟動器軟體安裝到要存取iSCSI目標伺服器上設定之目標設備的每部伺服器。

1. 開啟 YaST，並以 `root` 使用者身分登入。
2. 選取「*網路服務*」>「*iSCSI 啟動器*」。
3. 當系統提示您安裝 `open-iscsi` 套件時按一下「*安裝*」。
4. 依照畫面上的安裝指示操作，並在需要時提供安裝媒體。

   安裝完成後，YaST 會開啟至「iSCSI 啟動器綜覽」頁面的「*服務*」索引標籤中。
5. 請繼續執行第 13.3 節「設定 iSCSI 啟動程式」[176頁]。

13.2 設定 iSCSI 目標

SUSE® Linux Enterprise Server 隨附一個由 Ardis iSCSI 目標演進而來的開放原始碼 iSCSI 目標解決方案。使用 YaST 即可完成基本設定，但如果要充分利用 iSCSI 的優點，就必須用手動設定。

- 第 13.2.1 節「準備儲存空間」[168頁]
- 第 13.2.2 節「使用 YaST 建立 iSCSI 目標」[170頁]
- 第 13.2.3 節「手動設定 iSCSI 目標」[173頁]
- 第 13.2.4 節「使用 ietadm 設定線上目標」[174頁]

13.2.1 準備儲存空間

iSCSI 目標組態會將現有區塊設備輸出到 iSCSI 啟動器中。您必須準備要用於目標設備的儲存空間，方法是使用 YaST 中的磁碟分割程式來設定未格式化的分割區或設備，或從指令行對設備進行磁碟分割。

重要

將某個設備或分割區設定為 iSCSI 目標後，您將無法再透過其本地路徑直接存取它。請勿在建立時為其指定掛接點。

- 章節「分割設備」[168頁]
- 章節「對虛擬環境中的設備進行磁碟分割」[169頁]

分割設備

1. 以 `root` 使用者身分登入，然後開啟 YaST。
2. 選取「*系統*」>「*磁碟分割程式*」。
3. 按一下「*是*」繼續顯示關於使用磁碟分割程式的警告。
4. 按一下「*新增*」建立分割區，但不對其進行格式化，也不進行掛接。

iSCSI 目標能夠以 Linux、Linux LVM 或 Linux RAID 檔案系統 ID 來使用未格式化的分割區。

4a 選取「*主分割區*」，然後按「*下一步*」。

4b 指定要使用的空間大小，然後按「*下一步*」。

4c 選取「*不格式化*」，然後指定檔案系統 ID 類型。

4d 選取「*不掛接*」。

4e 按一下「*完成*」。

5 對稍後要用做 iSCSI LUN 的所有區域重複步驟 4 [168頁]。

6 按一下「*接受*」保留變更，然後關閉 YaST。

對虛擬環境中的設備進行磁碟分割

您可以使用 Xen 訪客伺服器做為 iSCSI 目標伺服器。您必須為客體虛擬機器指定要用於 iSCSI 儲存設備的儲存空間，然後在客體環境內以虛擬磁碟的方式存取該空間。每個虛擬磁碟可以是實體區塊設備，如整個磁碟、分割區或磁碟區，也可以是檔案備份磁碟影像，其中虛擬磁碟是 Xen 主機伺服器中較大實體磁碟上的一個影像檔案。為獲得最佳效能，請從實體磁碟或分割區建立所有虛擬磁碟。為訪客虛擬機器設定虛擬磁碟後，啟動訪客伺服器，然後依照與實體伺服器相同的程序將新的空白虛擬磁碟設定為 iSCSI 目標設備。

檔案備份磁碟影像建立在 Xen 主機伺服器上，然後會指定給 Xen 訪客伺服器。依預設，Xen 會將檔案備份磁碟影像儲存到 `/var/lib/xen/images/`*`vm 名稱`*目錄中，其中 *`vm 名稱`*為虛擬機器的名稱。

例如，若要建立大小為 4GB 的磁碟影像 `/var/lib/xen/images/vm_one/xen-0`，請先確認該目錄已存在，然後再建立影像本身。

1 以 `root` 使用者的身分登入主機伺服器。

2 在終端機主控台提示符處，輸入下列指令

```
mkdir -p /var/lib/xen/images/vm_one
dd if=/dev/zero of=/var/lib/xen/images/vm_one/xen-0 seek=1M bs=4096
count=1
```

3 將檔案系統影像指定給 Xen 組態檔案中的訪客虛擬機器。

4 以 root 使用者身分登入訪客伺服器，然後依照章節「分割設備」[168頁]中的程序使用 YaST 設定虛擬區塊設備。

13.2.2 使用 YaST 建立 iSCSI 目標

1 開啟 YaST，並以 root 使用者身分登入。

2 選取「*網路服務*」>「*iSNS 目標*」。

YaST 會開啟至「iSCSI 目標綜覽」頁面的「*服務*」索引標籤中

3 在「*服務啟動*」區域中選取下列其中一項：

- **開機時：** 以後伺服器重新開機時自動啟動啟動器服務。
- **手動 (預設)：** 手動啟動服務。

4 如果您是使用 iSNS 通告目標，請選取「*iSNS 存取控制*」核取方塊，然後輸入 IP 位址。

5 若需要，可開啟防火牆埠以允許從遠端電腦存取伺服器。

5a 選取「*在防火牆中開啟埠*」核取方塊。

5b 指定要開啟連接埠的網路介面，方法是按一下「*防火牆細節*」，選取該網路介面旁的核取方塊將其啟用，然後按一下「*確定*」接受設定。

6 若需要驗證才能連接到此伺服器上設定的目標設備，請選取「*全域*」索引標籤，取消選取「*無驗證*」以啟用驗證，然後指定內送和外送驗證所需的身分證明。

系統預設會啟用「*無驗證*」選項。您可以指定內送驗證、外送驗證，或針對內送和外送兩個方向的驗證。您也可以將多組使用者名稱與密碼新增至「*內送驗證*」下的清單中，為內送驗證指定多組身分證明。

7 設定 iSCSI 目標設備。

7a 選取「*目標*」索引標籤。

7b 若尚未清除 iSCSI 範例目標，請從清單中將其選取並刪除，然後按一下「*繼續*」確認刪除。

7c 按一下「*新增*」以新增新的 iSCSI 目標。

iSCSI 目標會自動顯示為未格式化的分割區或區塊設備，並完成「目標」和「識別碼」字段。

7d 您可以接受此設定，也可以瀏覽並選取另一個空間。

您也可以分割空間以在設備上建立 LUN，只需按一下「*新增*」並指定分配給該 LUN 的磁區。若需要為這些 LUN 設定其他選項，請選取「*進階設定*」。

7e 按「*下一步*」

7f 對要建立的每個 iSCSI 目標設備重複步驟 7c [171頁] 至步驟 7e [171頁]。

7g (選擇性) 在「*服務*」索引標籤中，按一下「*儲存*」將設定之 iSCSI 目標的相關資訊輸出到檔案。

這樣，稍後為資源使用者提供此資訊便會更為方便。

7h 按一下「*完成*」建立設備，然後按一下「*是*」重新啟動 iSCSI 軟體堆疊。

若要設定 iSCSI 目標，請在 YaST 中執行「*iSCSI 目標*」模組。組態分為三個索引標籤：在「*服務*」索引標籤中，選取啟動模式和防火牆設定。如果要從遠端機器存取 iSCSI 目標，請選取「*開啟防火牆中的連接埠*」。若 iSNS 伺服器應管理探查與存取控制，請啟用「*iSNS 存取控制*」，然後輸入 iSNS 伺服器的 IP 位址。您不能使用主機名稱，而是必須使用 IP 位址。如需 iSNS 相關的詳細資訊，請閱讀第 12 章「*iSNS for Linux*」 [153頁]。

「*全域*」索引標籤提供 iSCSI 伺服器的設定。此處所設定的驗證將用來探查服務，而不是用於存取目標。如果不想將存取僅限於搜索，請使用「*無驗證*」。

如果需要驗證，就必須考慮兩種可能性。一種是啟動程式必須證明它有許可權，可以在 iSCSI 目標上執行探查。這是藉由「*內送驗證*」來完成。另一種可能性是 iSCSI 目標必須向啟動程式證明它就是預期的目標。因此，iSCSI 目標也可以提供使用者名稱和密碼。這是藉由「*外送驗證*」來完成。*RFC 3720* [http://www.ietf.org/rfc/rfc3720.txt] 提供了有關驗證的詳細資訊。

目標是在「*目標*」索引標籤中定義。使用「*新增*」可建立新的 iSCSI 目標。第一個對話方塊會詢問要輸出的設備相關資訊。

目標
: 「*目標*」行有類似下列固定語法：

```
iqn.yyyy-mm.<reversed domain name>
```

開頭一定是 iqn。yyyy-mm 則採用目標啟用時的日期格式。如需更多有關命名慣例的資訊，請參閱 *RFC 3722* [http://www.ietf.org/rfc/rfc3722.txt]。

Identifier
: 「*識別碼*」可自由選取。它應該遵循某些機制，使系統結構更為一致。

LUN
: 數個 LUN 可以指定給一個目標。若要執行此操作，請在「*目標*」索引標籤中選取目標，然後按一下「*編輯*」。然後，向現有的目標新增新的 LUN。

路徑
: 新增要輸出的區塊設備或檔案系統影像的路徑。

下一個功能表可設定目標的存取限制。組態非常類似探查驗證的組態。在這裡，您至少必須設定內送驗證。

「*下一步*」會完成新目標的組態，讓您回到「*目標*」索引標籤的綜覽頁面。最後按一下「*完成*」啟動變更。

13.2.3 手動設定 iSCSI 目標

在 /etc/ietd.conf 中設定 iSCSI 目標。這個檔案在 *Target* 宣告之前的所有參數都是供檔案全域使用。這部分的驗證資訊具有特殊意義—它不是全域的，而只用於探查 iSCSI 目標。

如果您可以存取 iSNS 伺服器，則應先設定檔案以告知目標有關此伺服器的資訊。iSNS 伺服器的位址必須始終以 IP 位址提供。您無法指定伺服器的 DNS 名稱。此功能的組態如下：

```
iSNSServer 192.168.1.111
iSNSAccessControl no
```

此組態可確保 iSCSI 目標使用 iSNS 伺服器進行註冊，這樣就可為啟動程式提供探查。如需有關 iSNS 的詳細資訊，請參閱第 12 章「*iSNS for Linux*」[153頁]。iSNS 探查的存取控制不受支援。只需保持「無 iSNS 存取控制」。

所有直接的 iSCSI 驗證都可以在兩個方向上完成。iSCSI 目標可要求 iSCSI 啟動程式使用 IncomingUser 進行驗證，這可以新增許多次。iSCSI 啟動器也可以要求 iSCSI 目標進行驗證。這時應使用 OutgoingUser。兩者語法相同：

```
IncomingUser <username> <password>
OutgoingUser <username> <password>
```

驗證後面接著一或多個目標定義。請為每個目標新增 Target 區段。此區段的開頭固定是 Target 識別碼，後面接著邏輯單位編號的定義：

```
Target iqn.yyyy-mm.<reversed domain name>[:identifier]
          Lun 0 Path=/dev/mapper/system-v3
          Lun 1 Path=/dev/hda4
          Lun 2 Path=/var/lib/xen/images/xen-1,Type=fileio
```

在 Target 行中，yyyy-mm 是目標啟用時的日期，而且 identifier 可以自由選取。如需更多有關命名慣例的資訊，請參閱 *RFC 3722* [http://www.ietf.org/rfc/rfc3722.txt]。本例中輸出三個不同的區塊設備。第一個區塊設備是邏輯磁碟區 (請參閱第 4 章「*LVM 組態*」[27頁])，第二個是 IDE 分割區，第三個是本地檔案系統中可用的影像。這些對 iSCSI 啟動程式而言都像是區塊設備。

啟用 iSCSI 目標前，請在 Lun 定義後至少新增一個 IncomingUser。它會執行此目標所用的驗證。

若要啟用所有變更，請用 rcopen-iscsi restart 重新啟動 iscsitarget 精靈。檢查 /proc 檔案系統中的組態：

```
cat /proc/net/iet/volume
tid:1 name:iqn.2006-02.com.example.iserv:systems
        lun:0 state:0 iotype:fileio path:/dev/mapper/system-v3
        lun:1 state:0 iotype:fileio path:/dev/hda4
        lun:2 state:0 iotype:fileio path:/var/lib/xen/images/xen-1
```

還有許多其他選項可控制 iSCSI 目標的行為。如需詳細資訊，請參閱 ietd.conf 線上文件。

/proc 檔案系統中也會顯示作用中工作階段。針對每個連接的啟動程式，/proc/net/iet/session 中會新增一個額外的項目：

```
cat /proc/net/iet/session
tid:1 name:iqn.2006-02.com.example.iserv:system-v3
   sid:562949957419520
initiator:iqn.2005-11.de.suse:cn=rome.example.com,01.9ff842f5645
      cid:0 ip:192.168.178.42 state:active hd:none dd:none
   sid:281474980708864 initiator:iqn.2006-02.de.suse:01.6f7259c88b70
      cid:0 ip:192.168.178.72 state:active hd:none dd:none
```

13.2.4 使用 ietadm 設定線上目標

如有必要變更 iSCSI 目標組態，您必須重新啟動目標，才能使在組態檔案中所做的變更生效。可惜的是，在這個過程中，所有作用中工作階段都會被中斷。若要維持不受干擾的操作，除了在主要組態檔案 /etc/ietd.conf 中進行變更之外，您還要使用 ietadm 管理公用程式手動變更目前組態。

若要建立擁有 LUN 的 iSCSI 目標，請先更新您的組態檔案。增加的項目可為：

```
Target iqn.2006-02.com.example.iserv:system2
          Lun 0 Path=/dev/mapper/system-swap2
          IncomingUser joe secret
```

若要手動設定這個組態，請執行下列步驟：

1. 使用 `ietadm --op new --tid=2 --params Name=iqn.2006-02.com.example.iserv:system2` 指令建立新目標。

2. 使用 `ietadm --op new --tid=2 --lun=0 --params Path=/dev/mapper/system-swap2` 建立邏輯單位。

3. 使用 `ietadm --op new --tid=2 --user --params=IncomingUser=joe,Password=secret` 設定這個目標上的使用者名稱和密碼組合。

4. 使用 `cat /proc/net/iet/volume` 檢查組態。

您也可以刪除作用中連線。首先，使用 `cat /proc/net/iet/session` 指令檢查所有作用中連線。如下所示：

```
cat /proc/net/iet/session
tid:1 name:iqn.2006-03.com.example.iserv:system
        sid:281474980708864 initiator:iqn.1996-04.com.example:01.82725735af5
                cid:0 ip:192.168.178.72 state:active hd:none dd:none
```

若要刪除工作階段 ID 為 281474980708864 的工作階段，請使用 `ietadm --op delete --tid=1 --sid=281474980708864 --cid=0` 指令。請注意，這樣會使用戶端系統無法存取設備，而且存取這個設備的程序可能會暫停。

ietadm 也可用來變更各種組態參數。使用 `ietadm --op show --tid=1 --sid=0` 可取得全域變數清單。輸出會類似以下資訊：

```
InitialR2T=Yes
ImmediateData=Yes
MaxConnections=1
MaxRecvDataSegmentLength=8192
MaxXmitDataSegmentLength=8192
MaxBurstLength=262144
FirstBurstLength=65536
DefaultTime2Wait=2
DefaultTime2Retain=20
MaxOutstandingR2T=1
DataPDUInOrder=Yes
DataSequenceInOrder=Yes
ErrorRecoveryLevel=0
HeaderDigest=None
DataDigest=None
OFMarker=No
```

```
IFMarker=No
OFMarkInt=Reject
IFMarkInt=Reject
```

所有這些參數均可輕鬆變更。例如，如果要將最大連線數變更為 2，請使用

```
ietadm --op update --tid=1 --params=MaxConnections=2.
```

在 `/etc/ietd.conf` 檔案中，關聯行應該類似 `MaxConnections 2`。

警告

透過 `ietadm` 公用程式所做的變更對系統並非永久有效。這些變更如果不加入 `/etc/ietd.conf` 組態檔案，則會在下次重新開機時丟失。根據您網路的 iSCSI 使用情況，這可能會導致嚴重的問題。

`ietadm`公用程式還有其他許多選項可供使用。使用`ietadm –h`可找到綜覽。該處的縮寫為目標 ID (tid)、工作階段 ID (sid) 和連線 ID (cid)。您也可以在 `/proc/net/iet/session` 找到這些資訊。

13.3 設定 iSCSI 啟動程式

iSCSI 啟動器也稱為 iSCSI 用戶端，可用來連接任何 iSCSI 目標。這不僅限於第 13.2 節「設定 iSCSI 目標」[168頁] 中說明的 iSCSI 目標解決方案。iSCSI 啟動器的組態涉及兩個主要步驟：探查可用的 iSCSI 目標和設定 iSCSI 工作階段。這兩個步驟都可以使用 YaST 來完成。

- 第 13.3.1 節「使用 YaST 設定 iSCSI 啟動程式的組態」[176頁]
- 第 13.3.2 節「手動設定 iSCSI 啟動程式」[180頁]
- 第 13.3.3 節「iSCSI 用戶端資料庫」[181頁]
- 第 13.3.4 節「如需更多資訊」[182頁]

13.3.1 使用 YaST 設定 iSCSI 啟動程式的組態

YaST 中的 iSCSI 啟動器綜覽包含三個索引標籤：

- **服務：** 「*服務*」索引標籤可用來在開機時啟用 iSCSI 啟動器。同時會提供設定用於該探查的唯一「*啟動程式名稱*」及 iSNS 伺服器。iSNS 的預設連接埠為 3205。
- **已連線目標：** 「*連接的目標*」索引標籤會提供目前已連接 iSCSI 目標的綜覽。它與「*探查的目標*」索引標籤一樣，也提供為系統新增新目標的選項。

  在此頁面中，您可以選取目標設備，然後切換每個 iSCSI 目標設備的啟動設定：

  - **自動：** 此選項用於 iSCSI 服務自身啟動時要連接的 iSCSI 目標。這是一般組態。
  - **開機時：** 此選項用於開機時要連接的 iSCSI 目標；也就是說，當根目錄 (/) 位於 iSCSI 上時。因此，伺服器開機時，iSCSI 目標設備會從 initrd 進行評估。
- **探查的目標：** 「*探查的目標*」提供手動探查網路中 iSCSI 目標的途徑。
- 章節「設定 iSCSI 啟動器」[177頁]
- 章節「使用 iSNS 探查 iSCSI 目標」[178頁]
- 章節「手動探查 iSCSI 目標」[179頁]
- 章節「設定 iSCSI 目標設備的啟動優先設定」[179頁]

設定 iSCSI 啟動器

1. 開啟 YaST，並以 `root` 使用者身分登入。
2. 選取「*網路服務*」>「*iSCSI 啟動器*」。

   YaST 會開啟至「iSCSI 啟動器綜覽」頁面的「*服務*」索引標籤中。
3. 在「*服務啟動*」區域中選取下列其中一項：
   - **開機時：** 以後伺服器重新開機時自動啟動啟動器服務。
   - **手動 (預設)：** 手動啟動服務。

4 指定或驗證「*啟動器名稱*」。

為此伺服器上的iSCSI啟動器指定一個格式正確的啟動器合格名稱(IQN)。此啟動器名稱在網路上必須是全域唯一的。IQN 的一般格式如下：

```
iqn.yyyy-mm.com.mycompany:n1:n2
```

其中，n1 與 n2 為英數字元。例如：

```
iqn.1996-04.de.suse:01:9c83a3e15f64
```

「*啟動器名稱*」中會自動填上伺服器上 `/etc/iscsi/initiatorname.iscsi` 檔案中的對應值。

如果伺服器支援 iBFT (iSCSI 開機韌體表)，「*啟動器名稱*」中會填上 IBFT 中的對應值，並且該名稱無法在此介面上變更，不過您可以使用 BIOS 設定來修改。iBFT 是指包含各種對 iSCSI 開機程序有用之參數的資訊區塊，包括伺服器的 iSCSI 目標與啟動器描述。

5 使用下列方法之一探查網路上的 iSCSI 目標。

- **iSNS：** 若要使用 iSNS (網際網路儲存名稱服務) 來探查 iSCSI 目標，請繼續章節「使用 iSNS 探查 iSCSI 目標」[178頁]。
- **探查的目標：** 若要手動探查 iSCSI 目標設備，請繼續章節「手動探查 iSCSI 目標」[179頁]。

使用 iSNS 探查 iSCSI 目標

只有在您的環境中安裝並設定了 iSNS 伺服器後，才可以使用此選項。如需更多資訊，請參閱第 12 章「*iSNS for Linux*」[153頁]。

1 在 YaST 中，選取「*iSCSI 啟動器*」，然後選取「*服務*」索引標籤。

2 指定 iSNS 伺服器與連接埠的 IP 位址。

預設埠為 3205。

3 在「iSCSI啟動器綜覽」頁面中，按一下「*完成*」以儲存並套用您的變更。

手動探查 iSCSI 目標

對要從您目前正設定iSCSI啟動器的伺服器存取的所有iSCSI目標伺服器重複下列程序。

1. 在YaST中，選取「*iSCSI啟動器*」，然後選取「*探查的目標*」索引標籤。
2. 按一下「*探查*」開啟「*iSCSI 啟動器探查*」對話方塊。
3. 輸入 IP 位址，並視需要變更連接埠。

   預設埠為 3260。
4. 如果需要驗證，請取消選取「*無驗證*」，然後指定「*內送*」或「*外送*」驗證的身分證明。
5. 按「*下一步*」開始探查並連接到 iSCSI 目標伺服器。
6. 如果需要身分證明，則在探查成功後使用「*登入*」啟動目標。

   系統會提示您提供驗證身分證明以使用所選的 iSCSI 目標。
7. 按一下「*下一步*」完成組態。

   如果一切順利，現在目標就會出現在「*連接的目標*」中。

   接著，就可以使用虛擬 iSCSI 設備。
8. 在「iSCSI啟動器綜覽」頁面中，按一下「*完成*」以儲存並套用您的變更。
9. 您可以使用 `lsscsi` 指令尋找 iSCSI 目標設備的本地設備路徑：

```
lsscsi
[1:0:0:0]   disk    IET      VIRTUAL-DISK     0     /dev/sda
```

設定 iSCSI 目標設備的啟動優先設定

1. 在YaST中，選取「*iSCSI啟動器*」，然後選取「*已連線目標*」索引標籤，以檢視目前連接到伺服器的 iSCSI 目標設備清單。

2 選取要管理的 iSCSI 目標設備。

3 按一下「*切換啟動*」修改設定：

- **自動：** 此選項用於 iSCSI 服務自身啟動時要連接的 iSCSI 目標。這是一般組態。
- **開機時：** 此選項用於開機時要連接的 iSCSI 目標；也就是說，當根目錄 (/) 位於 iSCSI 上時。因此，伺服器開機時，iSCSI 目標設備會從 initrd 進行評估。

4 按一下「*完成*」以儲存並套用您的變更。

13.3.2 手動設定 iSCSI 啟動程式

iSCSI 連線的探查和組態都需要執行中的 iscsid。第一次執行搜索時，會在 `/var/lib/open-iscsi` 目錄中建立 iSCSI 啟動程式的內部資料庫。

如果您的探查受到密碼保護，請提供驗證資訊給 iscsid。因為執行第一次探查時，內部資料庫還不存在，所以這時無法使用該資料庫，而必須編輯 `/etc/iscsid.conf` 組態檔案來提供資訊。若要新增您的搜索密碼資訊，請將下列幾行加到 `/etc/iscsid.conf` 結束處：

```
discovery.sendtargets.auth.authmethod = CHAP
discovery.sendtargets.auth.username = <username>
discovery.sendtargets.auth.password = <password>
```

探查會將收到的所有值儲存在永久的內部資料庫中。此外，它會顯示所有偵測到的目標。請使用 `iscsiadm -m discovery --type=st --portal=<targetip>` 執行這個探查。輸出應該類似以下資訊：

```
149.44.171.99:3260,1 iqn.2006-02.com.example.iserv:systems
```

若要探查 `iSNS` 伺服器上可使用的目標，請使用 `iscsiadm --mode discovery --type isns --portal <targetip>` 指令

針對 iSCSI 目標上定義的每個目標，會各出現一行。如需已儲存資料的詳細資訊，請參閱第 13.3.3 節「iSCSI 用戶端資料庫」[181頁]。

iscsiadm 特殊的 --login 選項會建立所有需要的設備：

```
iscsiadm -m node -n iqn.2006-02.com.example.iserv:systems --login
```

新產生的設備會顯示在 lsscsi 的輸出中，而且現在可以用 mount 來存取。

13.3.3 iSCSI 用戶端資料庫

iSCSI 啟動程式探查到的所有資訊都儲存在位於 /var/lib/open-iscsi 的兩個資料庫檔案中。一個資料庫用來探查目標，一個資料庫用於已探查到的節點。存取資料庫時，您必須先選取要從探查資料庫或從節點資料庫中取得資料。使用 iscsiadm 的 -m discovery 和 -m node 參數就可以做到這一點。使用 iscsiadm 而且只搭配其中一個參數，可提供儲存記錄的綜覽：

```
iscsiadm -m discovery
149.44.171.99:3260,1 iqn.2006-02.com.example.iserv:systems
```

這個範例中的目標名稱為 iqn.2006-02.com.example.iserv:systems。與這個特殊資料集相關的所有動作都需要這個名稱。若要檢查 ID iqn.2006-02.com.example.iserv:systems 的資料記錄內容，請使用下列指令：

```
iscsiadm -m node --targetname iqn.2006-02.com.example.iserv:systems
node.name = iqn.2006-02.com.example.iserv:systems
node.transport_name = tcp
node.tpgt = 1
node.active_conn = 1
node.startup = manual
node.session.initial_cmdsn = 0
node.session.reopen_max = 32
node.session.auth.authmethod = CHAP
node.session.auth.username = joe
node.session.auth.password = ********
node.session.auth.username_in = <empty>
node.session.auth.password_in = <empty>
node.session.timeo.replacement_timeout = 0
node.session.err_timeo.abort_timeout = 10
node.session.err_timeo.reset_timeout = 30
node.session.iscsi.InitialR2T = No
node.session.iscsi.ImmediateData = Yes
....
```

若要編輯這其中一個變數的值，請使用 iscsiadm 指令搭配 update 作業。例如，如果希望 iscsid 在初始化時登入 iSCSI 目標，請將 node.startup 變數設定為 automatic 值：

```
iscsiadm -m node -n iqn.2006-02.com.example.iserv:systems --op=update
--name=node.startup --value=automatic
```

使用 delete 操作移除過時的資料集。如果目標 iqn.2006-02.com.example.iserv:systems 不再是有效的記錄，則使用以下指令加以刪除：

```
iscsiadm -m node -n iqn.2006-02.com.example.iserv:systems --op=delete
```

重要

請謹慎地使用此選項，因為該選項會刪除記錄，而不提供其他確認提示。

若要取得所有探查目標的清單，則執行 iscsiadm -m node 指令。

13.3.4 如需更多資訊

iSCSI 通訊協定已存在多年，所以有許多評鑑報告和其他文件，將 iSCSI 與 SAN 解決方案做比較、測試其效能基準或僅僅說明各種硬體解決方案。以下是 open-iscsi 相關詳細資訊的重要網頁：

- *Open-iSCSI Project (開放式 iSCSI 專案)* [http://www.open-iscsi.org/]
- *AppNote: iFolder on Open Enterprise Server Linux Cluster using iSCSI (AppNote：使用 iSCSI 之 Open Enterprise Server Linux Cluster 上的 iFolder)* [http://www.novell.com/coolsolutions/appnote/15394.html]

此外也有一些線上文件。請參閱 iscsiadm、iscsid、ietd.conf 和 ietd 的線上文件，以及範例組態檔案 /etc/iscsid.conf。

14 磁碟區快照

檔案系統快照是一種寫入時複製技術，它會監控現有磁碟區資料區塊的變更，以便在對其中一個區塊執行寫入操作時，將進行快照時區塊的值複製到快照磁碟區。使用這種方式便會保留一份時間點資料，直到快照磁碟區被刪除。

14.1 瞭解磁碟區快照

檔案系統快照包含在進行快照後已變更之原始磁碟區的相關中繼資料及資料區塊。透過快照存取資料時，您會看到複製原始磁碟區的時間點。您無需從備份媒體還原資料，或是覆寫已變更的資料。

在 Xen 主機環境中，虛擬機器必須將 LVM 邏輯磁碟區用做其儲存後端，以免使用虛擬磁碟檔案。

Linux快照可用於從檔案系統的時間點檢視窗建立備份。您可以實時建立快照，建立後它會一直保留，直到您將它刪除。您可以從快照備份檔案系統，而磁碟區本身仍可繼續供使用者使用。快照最初包含自身相關的一些中繼資料，但不包含原始磁碟區的實際資料。快照會使用寫入時複製技術偵測原始資料區塊中發生的資料變更。當對快照磁碟區中的區塊捕獲快照時，它會複製所包含的值，

然後允許在原始區塊中儲存新的資料。隨著區塊從其原始值開始不斷變更，快照大小也在不斷增加。

調整快照大小時，請考慮原始磁碟區中要變更的資料量，以及要保留快照的時間。為快照磁碟區配置的空間量可變更，具體取決於原始磁碟區大小、快照的預期保留時間長度，以及在快照存留期間預期變更的資料區塊數。快照磁碟區一經建立，就無法調整大小。建議在建立快照磁碟區時，將其大小設定成約為原始邏輯磁碟區的 10%。如果您預測在刪除快照前，原始磁碟區中的每個區塊都會至少變更一次，則快照的容量至少應相當於原始磁碟區的容量加上部份額外空間，其中後者用於儲存快照磁碟區的相關中繼資料。如果資料變更並不頻繁，或者預計的存留期足夠簡短，所需的空間就會減少。

重要

在快照的存留期間，必須先掛接快照，然後才能掛接其原始磁碟區。

對快照的操作完成後，請務必將其從系統中移除。隨著原始磁碟區上資料區塊的不斷變更，快照終將完全填滿。快照填滿時就會處於停用狀態，導致您無法重新掛接原始磁碟區。

快照移除的順序為越晚建立則越早刪除。

14.2 使用 LVM 建立 Linux 快照

邏輯磁碟區管理員 (LVM) 可用於建立檔案系統的快照。

- 開啟終端機主控台，以 `root` 使用者身分登入，然後輸入

  ```
  lvcreate -s -L 1G -n snap_volume source_volume_path
  ```

  例如：

  ```
  lvcreate -s -L 1G -n linux01-snap /dev/lvm/linux01
  ```

  則快照建立為 `/dev/lvm/linux01-snap` 磁碟區。

14.3 監控快照

- 開啟終端機主控台，以 root 使用者身分登入，然後輸入

```
lvdisplay snap_volume
```

例如：

```
lvdisplay /dev/vg01/linux01-snap

--- Logical volume ---
  LV Name                /dev/lvm/linux01
  VG Name                vg01
  LV UUID                QHVJYh-PR3s-A4SG-s4Aa-MyWN-Ra7a-HL47KL
  LV Write Access        read/write
  LV snapshot status     active destination for /dev/lvm/linux01
  LV Status              available
  # open                 0
  LV Size                80.00 GB
  Current LE             1024
  COW-table size         8.00 GB
  COW-table LE           512
  Allocated to snapshot  30%
  Snapshot chunk size    8.00 KB
  Segments               1
  Allocation             inherit
  Read ahead sectors     0
  Block device           254:5
```

14.4 刪除 Linux 快照

- 開啟終端機主控台，以 root 使用者身分登入，然後輸入

```
lvremove snap_volume_path
```

例如：

```
lvremove /dev/lvm/linux01-snap
```

15 儲存問題疑難排解

本章描述如何解決設備、軟體 RAID、多重路徑 I/O 及磁碟區的已知問題。

15.1 開機分割區可以使用 DM-MPIO 嗎？

自 SUSE® Linux Enterprise Server 10 Support Pack 1 起，即提供了對開機分割區的設備對應程式多重路徑 I/O (DM-MPIO) 支援。

文件更新

A

自 SUSE® Linux Enterprise Server 11 最初發行之後，此《*SUSE Linux Enterprise Server 儲存管理指南*》中的內容做了一些變更，本章將介紹這些變更的相關資訊。如果您之前已用過該產品，請檢閱變更項目，瞭解修改的內容。如果您是新使用者，只需閱讀目前版本的指南。

本文件在以下日期進行了更新：

A.1 2010 年 5 月 (SLES 11 SP1)

對以下小節進行了更新。變更說明如下。

- 第 A.1.1 節「管理設備的多重路徑 I/O」 [190頁]
- 第 A.1.2 節「IP 網路上的大型儲存設備：iSCSI」 [191頁]
- 第 A.1.3 節「新增功能」 [191頁]

A.1.1 管理設備的多重路徑 I/O

位置	改變
第 7.2.3 節「在多重路徑設備上使用 LVM2」 [57頁]	修正了步驟 3 [58頁] 中的範例。
第 7.2.6 節「根設備為多重路徑設備時的 SAN 逾時設定」 [59頁]	本節為新增內容。
第 7.3.2 節「多重路徑 I/O 管理工具」 [67頁]	對於不同的伺服器架構，套件的檔案清單可能會有所不同。有關 multipath-tools 套件中包含的檔案清單，請造訪「*SUSE Linux Enterprise Server Technical Specifications*」>「*Package Descriptions*」網頁 [http://www.novell.com/products/server/techspecs.html]，找到您的架構並選取「*Packages Sorted by Name*」，然後搜尋「multipath-tools」以找到該架構的套件清單。
第 7.4.1 節「準備 SAN 設備以進行多重路徑」 [71頁]	如果 SAN 設備將做為伺服器上的根設備使用，請依第 7.2.6 節「根設備為多重路徑設備時的 SAN 逾時設定」 [59頁] 中所述修改該設備的逾時設定。
第 7.8.1 節「啟用多重路徑 I/O 以在多重路徑儲存 LUN 上安裝 SLES」 [94頁]	本節為新增內容。

位置	改變
第 7.8.2 節「啟用多重路徑 I/O 以在主動/被動多重路徑儲存 LUN 上安裝 SLES」[95頁]	本節為新增內容。

A.1.2 IP 網路上的大型儲存設備：iSCSI

位置	改變
第 13.2.2 節「使用 YaST 建立 iSCSI 目標」[170頁] 中的步驟 7g [171頁]	在「*YaST*」>「*網路服務*」>「*iSCSI 目標*」功能中，「*儲存*」選項可讓您輸出 iSCSI 目標資訊，更方便地向資源使用者提供此資訊。

A.1.3 新增功能

位置	改變
第 2.1 節「SLES 11 SP1 中的新增功能」[13頁]	本節為新增內容。

A.2 2010 年 2 月 23 日

對以下小節進行了更新。變更說明如下。

- 第 A.2.1 節「設定根分割區的軟體 RAID」[192頁]
- 第 A.2.2 節「管理多重路徑 I/O」[192頁]

A.2.1 設定根分割區的軟體 RAID

位置	改變
第 9.1 節「軟體 RAID 的先決條件」[117頁]	修正了 RAID 0 定義中的一個錯誤。

A.2.2 管理多重路徑 I/O

位置	改變
第 7.10 節「掃描新設備而不重新開機」[101頁]	新增了關於使用 `rescan-scsi-bus.sh` 程序檔在不重新開機的情況下掃描設備的資訊。
第 7.11 節「掃描新分割的設備而不重新開機」[104頁]	新增了關於使用 `rescan-scsi-bus.sh` 程序檔在不重新開機的情況下掃描設備的資訊。

A.3 2010 年 1 月 20 日

以下小節進行了更新。變更說明如下。

- 第 A.3.1 節「管理多重路徑 I/O」[192頁]

A.3.1 管理多重路徑 I/O

位置	改變
章節「在 /etc/multipath.conf 中設定預設多重路徑行為」[79頁]	在 `default_getuid` 指令行中，使用上例所述的路徑 `/sbin/scsi_id` 代替 `/lib/udev/scsi_id` 的範例路徑 (位於範例檔案 `/usr/share/doc/packages/`

位置	改變
	multipath-tools/multipath.conf .synthetic以及預設和註記範例檔案中)。
表格 7.5「多重路徑屬性」[83頁] 中的 getuid	使用路徑 /sbin/scsi_id，取代了範例檔案 /usr/share/doc/packages/multipath-tools/multipath.conf .synthetic 以及預設與註記範例檔案中的路徑 /lib/udev/sbin_id。

A.4 2009 年 12 月 1 日

對以下小節進行了更新。變更說明如下。

- 第 A.4.1 節「管理設備的多重路徑 I/O」[193頁]
- 第 A.4.2 節「調整檔案系統大小」[194頁]
- 第 A.4.3 節「新增功能」[194頁]

A.4.1 管理設備的多重路徑 I/O

位置	改變
第 7.2.3 節「在多重路徑設備上使用 LVM2」[57頁] 第 7.9 節「設定現有軟體RAID的多重路徑 I/O」[98頁]	-f mpath 選項已變更為 -f multipath： mkinitrd -f multipath
表格 7.5「多重路徑屬性」[83頁] 中的 prio_callout	多重路徑 prio_callout 位於 /lib/libmultipath/lib* 中的共享程式庫中。透過使用共享程式庫，callout 會在精靈啟動時載入到記憶體中。

位置	改變

A.4.2 調整檔案系統大小

位置	改變
第 5.1.1 節「支援調整大小的檔案系統」[38頁]	resize2fs 公用程式支援線上或離線調整 ext3 檔案系統的大小。

A.4.3 新增功能

位置	改變
第 2.2.10 節「多重路徑工具 Callout 的位置變更」[20頁]	本節為新增內容。
第 2.2.11 節「mkinitrd -f 的選項從 mpath 變更為 multipath」[21頁]	本節為新增內容。

A.5 2009 年 10 月 20 日

對以下小節進行了更新。變更說明如下。

- 第 A.5.1 節「LVM 組態」[195頁]

- 第 A.5.2 節「管理設備的多重路徑 I/O」[195頁]
- 第 A.5.3 節「新增功能」[196頁]

A.5.1 LVM 組態

位置	改變
第 4.6 節「直接 LVM 管理」[33頁]	在 YaST 控制中心選取「*系統*」>「*磁碟分割程式*」。

A.5.2 管理設備的多重路徑 I/O

位置	改變
章節「在 /etc/multipath.conf 中將非多重路徑設備列入黑名單」[78頁]	關鍵字 devnode_blacklist 已廢棄，被關鍵字黑名單取代。
章節「在 /etc/multipath.conf 中設定預設多重路徑行為」[79頁]	getuid_callout 已變更為 getuid。
章節「瞭解優先程序群組與屬性」[83頁]	getuid_callout 已變更為 getuid。
章節「瞭解優先程序群組與屬性」[83頁]	新增了對 least-pending、length-load-balancing 和 service-time 選項的描述。

A.5.3 新增功能

位置	改變
第 2.2.9 節「多重路徑的進階 I/O 負載平衡選項」[20頁]	本節為新增內容。

A.6 2009 年 8 月 3 日

ïi以下小節進行了更新。變更說明如下。

- 第 A.6.1 節「管理多重路徑 I/O」[196頁]

A.6.1 管理多重路徑 I/O

位置	改變
第 7.2.5 節「將 --noflush 用於多重路徑設備」[59頁]	本節為新增內容。

A.7 2009 年 6 月 22 日

對以下小節進行了更新。變更說明如下。

- 第 A.7.1 節「管理多重路徑 I/O」[197頁]
- 第 A.7.2 節「使用 mdadm 管理軟體 RAID 6 和 10」[197頁]
- 第 A.7.3 節「IP 網路上的大型儲存設備：iSCSI」[197頁]

A.7.1 管理多重路徑 I/O

位置	改變
第 7.8 節「設定根設備的多重路徑 I/O」[94頁]	針對 System Z 新增了步驟 4 [98頁] 與步驟 6 [98頁]。
第 7.11 節「掃描新分割的設備而不重新開機」[104頁]	修正了步驟 2 中指令行的語法。
第 7.11 節「掃描新分割的設備而不重新開機」[104頁]	步驟 7 [104頁] 取代了原來的步驟 7 和步驟 8。

A.7.2 使用 mdadm 管理軟體 RAID 6 和 10

位置	改變
第 10.4 節「建立降級 RAID 陣列」[136頁]	若要查看每秒重新整理一次的重建進度，請輸入 `watch -n 1 cat /proc/mdstat`

A.7.3 IP 網路上的大型儲存設備：iSCSI

位置	改變
第 13.3.1 節「使用 YaST 設定 iSCSI 啟動程式的組態」[176頁]	為求清楚，資料已經過重新編排。 新增了關於如何使用 iSCSI 目標設備啟動選項之設定的資訊：

位置	改變
	• **自動：** 此選項用於 iSCSI 服務自身啟動時要連接的 iSCSI 目標。這是一般組態。 • **開機時：** 此選項用於開機時要連接的 iSCSI 目標；也就是說，當根目錄 (/) 位於 iSCSI 上時。因此，伺服器開機時，iSCSI 目標設備會從 initrd 進行評估。

A.8 2009 年 5 月 21 日

以下小節進行了更新。變更說明如下。

- 第 A.8.1 節「管理多重路徑 I/O」[198頁]

A.8.1 管理多重路徑 I/O

位置	改變
章節「為多重路徑自動偵測儲存陣列」[61頁]	對支援多重路徑的 IBM zSeries 設備進行的測試表明，應將 dev_loss_tmo 參數設定為 90 秒，將 fast_io_fail_tmo 參數設定為 5 秒。若您使用的是 zSeries 設備，則必須手動建立並設定 `/etc/multipath.conf` 檔案來指定這些值。如需更多資訊，請參閱章節「在 /etc/multipath.conf 中為 zSeries 設定預設設定值」[79頁]。
第 7.3.1 節「設備對應程式多重路徑模組」[64頁]	SUSE Linux Enterprise Server 11 及更高版本提供了對 `/boot` 設備的多重路徑支援。

位置	改變
章節「在 /etc/multipath.conf 中為 zSeries 設定預設設定值」[79頁]	本節為新增內容。
第 7.8 節「設定根設備的多重路徑 I/O」[94頁]	現在，SUSE Linux Enterprise Server 11 中提供了 DM-MP 及其對 `/boot` 和 `/root` 的支援。