SUSE Linux Enterprise Server

10 SP3

www.novell.com

2009 年8 月28 日

安裝與管理

安裝與管理

法律事項

目錄

關於本指南

本指南的主要對象為實際規劃、部署、設定和操作 SUSE Linux Enterprise® 期間的專業網路和系統管理員。因此，本指南的重點只在於確保 SUSE Linux Enterprise 的設定正確，而且網路所需服務都已備妥，讓它可以像初始安裝一樣正常運作。至於如何確保 SUSE Linux Enterprise 能夠與您企業的應用程式軟體正確相容，其核心功能能否符合您的需求，則不在本指南的討論範圍。本文假設您可符合完整需求，且需要安裝或檢測是否符合完整需求的測試安裝。

本指南包含下列內容：

部署
: 安裝 SUSE Linux Enterprise 之前，請先選擇最適合您案例的部署策略和磁碟設定。瞭解如何手動安裝系統、如何使用網路安裝設定及如何執行自動安裝。使用 YaST 設定已安裝的系統，以符合您的需求。

管理
: SUSE Linux Enterprise 提供範圍廣泛的工具，可自定系統的各個面向。這個部分介紹其中幾項。

系統
: 學習這部份，進一步瞭解作業系統基礎。SUSE Linux Enterprise 支援多種硬體結構，您可以此自行打造可執行於 SUSE Linux Enterprise 上的應用程式。開機載入程式和開機程序資訊會協助您瞭解 Linux 系統的運作方式，以及您個人的自定程序檔和應用程式如何與作業系統融合。

服務
: SUSE Linux Enterprise 的設計目的是要做為網路作業系統。它提供範圍廣泛的網路服務 (如 DNS、DHCP、網頁、Proxy 和驗證服務)，並充分整合到異質的環境中，包括 MS Windows 用戶端和伺服器。

安全性
: 這個版本的 SUSE Linux Enterprise 包含數個安全性相關功能。它隨附了 Novell® AppArmor，可讓您藉由限制權限來保護您的應用程式。本指南也將討論安全登入、防火牆和檔案系統加密。

疑難排解
當您遇到問題時，SUSE Linux Enterprise 備有豐富的應用程式、工具和文件供您參考。本指南將詳細討論 SUSE Linux Enterprise 最常見的一些問題及其解決方案。

1 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。請使用線上文件每頁下方的使用備註功能，並在其中輸入您的意見。

2 文件更新

如需本文件的最新版本，請參閱 SUSE Linux Enterprise Server 網站 [http://www.novell.com/documentation/sles10/index.html]。

3 其他文件

如需本產品的其他相關文件，請參閱 http://www.novell.com/documentation/sles10/index.html：

啟動指南
有關安裝類型和工作流程的基本資訊。

Architecture-Specific Information
SUSE Linux Enterprise Server 目標進行安裝所需準備的結構特定資訊。

Novell AppArmor Administration Guide
Novell AppArmor 的深入管理指南，為您介紹應用程式限制，以加強您環境的安全性。

儲存管理指南
在 SUSE Linux Enterprise 上管理多種儲存設備的簡介。

Heartbeat Guide
以 Heartbeat 設定高度可用性方式的深入管理指南。

Novell Virtualization Technology User Guide
以 SUSE Linux Enterprise 與 Xen* 虛擬化技術為基礎的虛擬化解決方案簡介。

如需 SUSE® Linux Enterprise Desktop 產品的文件綜覽，請參閱 http://www.novell.com/documentation/sled10/index.html。以下手冊是專為 SUSE Linux Enterprise Desktop 提供的：

GNOME User Guide
GNOME 桌面及其最重要應用程式的完整指南。

KDE User Guide
KDE 桌面及其最重要應用程式的完整指南。

Deployment Guide
為負責部署和管理 SUSE Linux Enterprise Desktop 的管理員提供的詳細指南。

Novell AppArmor Administration Guide
Novell AppArmor 的深入管理指南，為您介紹應用程式限制，以加強您環境的安全性。

本手冊的許多章節包含連到其他文件資源的連結。這包括系統和網際網路上所提供的其他文件。

4 文件慣例

本手冊使用下列印刷慣例：

- `/etc/passwd`：檔案名稱和目錄名稱
- *`placeholder`*：以實際的值來取代 *`placeholder`*
- `PATH`：環境變數 PATH
- `ls`, `--help`：指令、選項和參數
- `user`：使用者或群組
- Alt, Alt + F1：供人按下的按鍵或案件組合；顯示的按鍵與鍵盤上一樣為大寫

- 「*檔案*」,「*檔案*」>「*另存新檔*」：功能表項目、按鈕
- ►**amd64 ipf:** 本段僅與指定的結構有關。箭頭標示了文字區塊的開頭與結尾。◄

  ►**ipseries s390 zseries:** 本段僅與指定的架構有關。箭頭標示了文字區塊的開頭與結尾。◄
- *Dancing Penguins* (章節 *Penguins*, ↑其他手冊)：這是對其他手冊某章節的參考。

I. 部署

1 SUSE Linux Enterprise 規劃

無論是對現有 IT 環境還是全新的案例，作業系統的實作都必須仔細準備。SUSE Linux Enterprise 10 提供了多種新功能。我們不可能在此詳述所有新功能。下面僅列出一些有趣的主要增強功能。

Xen 3.0 虛擬系統
: 在單一伺服器上執行多部虛擬機器，每部都有其各自的作業系統例項。如需有關此技術的詳細資訊，請參閱 http://www.novell.com/documentation/sles10/index.html 中的虛擬化手冊。

YaST
: YaST 中開發了許多新組態選項。一般會在介紹該技術的章節中說明。

以 openWBEM 進行 CIM 管理
: 公用資訊模型物件管理員 (CIMON) 是一種網路式企業管理公用程式。它可以提供一種成熟的管理結構。並請參閱 第 11 章「*OpenWBEM*」 [219頁]。

SPident
: 管理公用程式 SPident 提供了已安裝軟體的綜覽，並釐清系統的目前 service pack 層級。

目錄服務
: 有多種 LDAP 相容的目錄服務可使用：

  - Microsoft 現用目錄
  - OpenLDAP

Novell AppArmor
: 使用 Novell AppArmor 技術強化系統。在*Novell AppArmor Administration Guide* (↑*Novell AppArmor Administration Guide*)中有此服務的深入說明。

iSCSI
: iSCSI 是一套連接 Linux 電腦與中央儲存系統的解決方案，易於使用，價格低廉。如需 iSCSI 的詳細資訊，請參閱 第 12 章「*IP 網路 — iSCSI 上的大型存放設備* 」 [247頁]。

網路檔案系統 v4
: SUSE Linux Enterprise 從版本 10 開始也支援 NFS 版本 4。如此可向您提供改進的效能、增強的安全性和「可設定狀態的」通訊協定。並請參閱 第 38 章「*使用 NFS 共享檔案系統*」 [651頁]。

Oracle Cluster File System 2
: OCFS2 是一般用途的日誌式檔案系統，與 Linux 2.6 和更新版本的核心完全整合。請在第 14 章「*Oracle Cluster File System 2*」 [263頁]檢視 OCFS2 的綜覽。

Heartbeat 2
: Heartbeat 2 提供業集成員資格與傳訊基礎結構。有關設定此類業集的方法，詳述於 *Heartbeat Guide*。

多路徑 I/O
: 設備映射多路徑 IO 功能會自動設定多種設定的子系統。如需詳細資訊，請參閱《*儲存管理指南*》中的多重路徑 I/O 相關章節。

Linux 核心損毀傾印
: 使用 Kexec 和 Kdump，現可更輕鬆的進行核心相關問題的除錯。此技術可用於 x86、AMD64、Intel 64 和 POWER 平台。

1.1 SUSE Linux Enterprise 部署考量

在計劃程序初期，您應嘗試定義專案目標以及所需功能。每個專案都必須個別執行此動作，但可同時考量下列問題：

- 要完成的安裝數量為何？最佳部署方法根據安裝數量而異。並請參閱 第 2 章「*部署策略*」 [7頁]。

- 系統是否將部署於惡意的環境中？請參閱第 49 章「*安全性與機密性*」[801 頁]以了解其後果。
- 您要如何進行定期更新？所有註冊的使用者都可從線上取得所有修補程式。請在http://support.novell.com/patches.html找到註冊與修補程式支援資料庫。
- 需要本地安裝的協助嗎？Novell 為 SUSE Linux Enterprise 所有相關主題提供訓練、支援與諮詢。若需詳細資訊，請至http://www.novell.com/products/server/。
- 您需要協力廠商產品嗎？請確認所需產品在預期的平台上受支援。若需要的話，Novell 亦可提供將軟體轉用至不同平台的服務。

1.2 部署 SUSE Linux Enterprise

為了確保系統能夠完美運作，請務必嘗試使用經認證的硬體。硬體認證程序是持續進行的程序，且認證硬體資料庫會定期更新。請到http://developer.novell.com/yessearch/Search.jsp搜尋認證的硬體。

根據所需的安裝數量，使用安裝伺服器甚至完全自動化安裝可能會很有幫助。請參閱第 2 章「*部署策略*」[7 頁]以獲得更多資訊。使用 Xen 虛擬化技術時，應考量網路根檔案系統或 iSCSI 這類的網路儲存解決方案。並請參閱 第 12 章「*IP 網路 — iSCSI 上的大型存放設備* 」[247 頁]。

SUSE Linux Enterprise 提供的服務數量眾多。請在本書中的關於本指南 [xv 頁]檢視文件綜覽。大部分的所需組態都可透過 SUSE 的組態公用程式 YaST 來完成。除此之外，相關章節內也說明了許多手動組態方式。

除了純粹的軟體安裝工作之外，您也應該考量對系統終端使用者與服務台職員的訓練。

1.3 執行 SUSE Linux Enterprise

SUSE Linux Enterprise 作業系統是經過充分測試的穩定系統。然而，這並不能防止硬體故障或其他原因所造成的停機或資料遺失。對於資料遺失會影響的所有重要運算任務，都應執行定期備份。

為了獲得最佳安全性並保證工作的安全，您應定期更新所有運作的機器。若您具有關鍵業務伺服器，則可能需要執行第二部相同機器，以在實際系統上運作之前先套用變更以進行測試。這也可以讓您在遇到硬體故障的時候能夠切換電腦。

2 部署策略

SUSE® Linux Enterprise 有幾種不同的部署方式。從使用實體媒體或網路安裝伺服器進行本地安裝，到使用遠端控制、高度自定和自動安裝技術進行大量部署，有各式各樣的作法供您選擇。請選擇最符合您需要的方法。

2.1 部署最多 10 台工作站

如果 SUSE Linux Enterprise 的部署只涉及 1 到 10 台工作站，最簡單的 SUSE Linux Enterprise 部署方式是第 3 章「*使用 YaST安裝*」 [17頁]中所描述的簡易手動安裝。手動安裝可依您的需求用幾種不同方式進行：

從 SUSE Linux Enterprise 媒體安裝 [7頁]
如果要安裝單一、未連接的工作站，請考慮這種作法。

從使用 SLP 的網路伺服器安裝 [8頁]
如果只有一台或少量工作站，而且可以使用透過 SLP 宣告的網路安裝伺服器，請考慮這種作法。

從網路伺服器安裝 [8頁]
如果只有一台或少量工作站，而且可以使用網路安裝伺服器，請考慮這種作法。

表格 2.1 *從 SUSE Linux Enterprise 媒體安裝*

安裝來源	SUSE Linux Enterprise 媒體套件

需要手動互動的任務	• 插入安裝媒體 • 啟動安裝目標 • 變更媒體 • 決定 YaST 安裝範圍 • 使用 YaST 系統設定系統
遠端控制的任務	無
詳細資料	第 3.3.2 節「從 SUSE Linux Enterprise 媒體安裝」[19頁]

表格 2.2 *從使用 SLP 的網路伺服器安裝*

安裝來源	存放 SUSE Linux Enterprise 安裝媒體的網路安裝伺服器
需要手動互動的任務	• 插入開機磁片 • 啟動安裝目標 • 決定 YaST 安裝範圍 • 使用 YaST 設定系統
遠端控制的任務	無，但這種方法可以跟 VNC 合併使用
詳細資料	第 3.3.3 節「從使用 SLP 的網路伺服器安裝」[19頁]

表格 2.3 *從網路伺服器安裝*

安裝來源	存放 SUSE Linux Enterprise 安裝媒體的網路安裝伺服器

需要手動互動的任務	• 插入開機磁片 • 提供開機選項 • 啟動安裝目標 • 決定 YaST 安裝範圍 • 使用 YaST 設定系統
遠端控制的任務	無，但這種方法可以跟 VNC 合併使用
詳細資料	第 3.3.4 節「從沒有 SLP 的網路來源安裝」[19頁]

2.2 部署最多 100 台工作站

因為要安裝的工作站數量增多，您當然不想手動安裝和設定每一台工作站。這時有許多自動或半自動方法，以及多種選項，可以在不需要使用者互動或實際互動極少的情況下執行安裝。

考慮採用全自動的方法之前，請考慮到情況越複雜，設定的時間就越長。如果部署工作有時間限制，最好選擇執行速度較快而且比較不複雜的方法。自動化適用於大量部署以及必須從遠端執行的部署。

請從下列選項中選擇：

透過 VNC 進行的簡易遠端安裝—靜態網路組態 [10頁]
: 如果是採用靜態網路設定的小到中型案例，請考慮這種作法。這種作法需要網路、網路安裝伺服器和 VNC 檢視器應用程式。

透過 VNC 進行的簡易遠端安裝—動態網路組態 [11頁]
: 如果是採用透過 DHCP 的動態網路設定的小到中型案例，請考慮這種作法。這種作法需要網路、網路安裝伺服器和 VNC 檢視器應用程式。

透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能 [11頁]
: 如果是必須透過網路安裝而且與安裝目標沒有實際互動的小到中型案例，請考慮這種作法。這種作法需要網路、網路安裝伺服器、網路開機影像、網路可開機目標硬體和 VNC 檢視器應用程式。

透過 SSH 進行的簡易遠端安裝—靜態網路組態 [12頁]
如果是採用靜態網路設定的小到中型案例，請考慮這種作法。這種作法需要網路、網路安裝伺服器和 SSH 用戶端應用程式。

透過 SSH 進行的簡易遠端安裝—動態網路組態 [12頁]
如果是採用透過 DHCP 的動態網路設定的小到中型案例，請考慮這種作法。這種作法需要網路、網路安裝伺服器和 SSH 用戶端應用程式。

透過 SSH 進行的遠端安裝—PXE 開機和網路喚醒功能 [13頁]
如果是必須透過網路安裝而且與安裝目標沒有實際互動的小到中型案例，請考慮這種作法。這種作法需要網路、網路安裝伺服器、網路開機影像、網路可開機目標硬體和 SSH 用戶端應用程式。

簡易大量安裝 [14頁]
如果要大量部署到完全相同的機器，請考慮這種作法。如果設定為使用網路開機，就完全不必與目標系統有任何實際互動。這種作法需要網路、網路安裝伺服器、遠端控制應用程式 (如 VNC 檢視器或 SSH 用戶端) 和 AutoYaST 組態設定檔。如果使用網路開機，也需要網路開機影像和網路可開機硬體。

以規則為基礎的自動安裝 [14頁]
如果要大量部署到各種不同類型的硬體，請考慮這種作法。如果設定為使用網路開機，就完全不必與目標系統有任何實際互動。這種作法需要網路、網路安裝伺服器、遠端控制應用程式 (如 VNC 檢視器或 SSH 用戶端)、數個 AutoYaST 組態設定檔和 AutoYaST 的規則設定。如果使用網路開機，也需要網路開機影像和網路可開機硬體。

表格 2.4 *透過 VNC 進行的簡易遠端安裝—靜態網路組態*

安裝來源	網路
準備	• 設定安裝來源 • 從安裝媒體開機
控制和監控	遠端：VNC
最適合	包含各種不同硬體的小到中型案例
缺點	• 必須個別設定每台機器

	• 需要實際存取才能開機
詳細資料	第 4.1.1 節「透過 VNC 進行的簡易遠端安裝—靜態網路組態」[44頁]

表格 2.5 *透過 VNC 進行的簡易遠端安裝—動態網路組態*

安裝來源	網路
準備	• 設定安裝來源 • 從安裝媒體開機
控制和監控	遠端：VNC
最適合	包含各種不同硬體的小到中型案例
缺點	• 必須個別設定每台機器 • 需要實際存取才能開機
詳細資料	第 4.1.2 節「透過 VNC 進行的簡易遠端安裝—動態網路組態」[45頁]

表格 2.6 *透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能*

安裝來源	網路
準備	• 設定安裝來源 • 設定 DHCP、TFTP、PXE 開機和 WOL • 從網路開機
控制和監控	遠端：VNC

最適合	• 包含各種不同硬體的小到中型案例 • 完全遠端安裝；跨網站部署
缺點	必須手動設定每台機器
詳細資料	第 4.1.3 節「透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能」[46頁]

表格 2.7 *透過 SSH 進行的簡易遠端安裝—靜態網路組態*

安裝來源	網路
準備	• 設定安裝來源 • 從安裝媒體開機
控制和監控	遠端：SSH
最適合	• 包含各種不同硬體的小到中型案例 • 與目標之間採用低頻寬連線
缺點	• 必須個別設定每台機器 • 需要實際存取才能開機
詳細資料	第 4.1.4 節「透過 SSH 進行的簡易遠端安裝—靜態網路組態」[47頁]

表格 2.8 *透過 SSH 進行的簡易遠端安裝—動態網路組態*

安裝來源	網路
準備	• 設定安裝來源

	• 從安裝媒體開機
控制和監控	遠端：SSH
最適合	• 包含各種不同硬體的小到中型案例 • 與目標之間採用低頻寬連線
缺點	• 必須個別設定每台機器 • 需要實際存取才能開機
詳細資料	第 4.1.5 節「透過 SSH 進行的簡易遠端安裝—動態網路組態」[48頁]

表格 2.9 *透過 SSH 進行的遠端安裝—PXE 開機和網路喚醒功能*

安裝來源	網路
準備	• 設定安裝來源 • 設定 DHCP、TFTP、PXE 開機和 WOL • 從網路開機
控制和監控	遠端：SSH
最適合	• 包含各種不同硬體的小到中型案例 • 完全遠端安裝；跨網站部署 • 與目標之間採用低頻寬連線
缺點	必須個別設定每台機器

詳細資料	第 4.1.6 節「透過 SSH 進行的遠端安裝—PXE 開機和網路喚醒功能」[49頁]

表格 2.10 *簡易大量安裝*

安裝來源	慣用的網路
準備	• 取得硬體資訊 • 建立 AutoYaST 設定檔 • 設定安裝伺服器 • 配送設定檔 • 設定網路開機 (DHCP、TFTP、PXE、WOL) *或* 從安裝媒體啟動目標
控制和監控	透過 VNC 或 SSH 於本地或遠端進行
最適合	• 大型案例 • 完全一樣的硬體 • 不存取系統 (網路開機)
缺點	只適用於硬體完全相同的機器
詳細資料	第 5.1 節「簡易大量安裝」[77頁]

表格 2.11 *以規則為基礎的自動安裝*

安裝來源	慣用的網路
準備	• 取得硬體資訊

	• 建立 AutoYaST 設定檔 • 建立 AutoYaST 規則 • 設定安裝伺服器 • 配送設定檔 • 設定網路開機 (DHCP、TFTP、PXE、WOL) *或* 從安裝媒體啟動目標
控制和監控	透過 SSH 或 VNC 於本地或遠端進行
最適合	• 各種不同硬體 • 跨網站部署
缺點	複雜的規則設定
詳細資料	第 5.2 節「以規則為基礎的自動安裝」 [87頁]

2.3 部署超過 100 台工作站

第 2.1 節「部署最多 10 台工作站」 [7頁]中大部分的中型安裝案例考慮事項，也適用於大型部署案例。但是，隨著安裝目標的數量增多，完全自動安裝方法的優點勝過其缺點。

為了符合大量部署網站的需求，的確值得花可觀的時間，在 AutoYaST 中建立複雜的規則和類別結構，因為不必個別接觸每個目標，可以為您節省大量的時間，不過這仍要視安裝專案的範圍而定。

3 使用 YaST安裝

當您已依 *Architecture-Specific Information* 手冊所述做好安裝 SUSE Linux Enterprise® 的硬體準備，且已建立與安裝系統的連線之後，畫面上會出現 SUSE Linux Enterprise 系統助手 YaST 的介面。YaST 將導引您完成安裝與組態設定程序。

3.1 IBM POWER：啟動系統以進行網路安裝

對於 IBM POWER 平台，系統會依 *Architecture-Specific Information* 手冊所述啟始化 (IPL)。在網路安裝時，SUSE Linux Enterprise Server 不會在這些系統中顯示開頭顯示畫面或開機載入程式指令行。請在安裝時手動載入核心。透過 VNC、X 或 SSH 與安裝系統建立連線後，YaST 便會立即啟動安裝畫面。由於不顯示開頭顯示畫面或開機載入程式指令行，因此無法在畫面上輸入核心或開機參數，而必須使用 `mkzimage_cmdline` 公用程式將其含入核心影像。如需相關描述，請參閱 *Architecture-Specific Information* 手冊中的「準備」章節。

提示：IBM POWER：後續步驟

請遵循 第 3.6 節「語言」[23頁] 提供的安裝步驟說明，使用 YaST 進行安裝。

3.2 IBM System z：啟動系統進行安裝

對於 IBM System z 平台，系統會依 *Architecture-Specific Information* 手冊所述啟始化 (IPL)。SUSE Linux Enterprise 不會在這些系統中顯示開頭顯示畫面。安裝期間，請手動載入核心、initrd 和 parmfile。透過 VNC、X 或 SSH 與安裝系統建立連線後，YaST 便會立即啟動安裝畫面。由於不顯示開頭顯示畫面，因此無法在畫面上輸入核心或開機參數，而必須用 parmfile 指定 (請參閱附錄 A *Appendix* (↑*Architecture-Specific Information*) 中的 parmfile 資訊)。

提示：IBM System z：後續步驟

請遵循第 3.6 節「語言」[23頁]提供的安裝步驟說明，使用 YaST 進行安裝。

3.3 系統啟動進行安裝

您可以從本地安裝來源 (如 SUSE Linux Enterprise CD 或 DVD)，或從 FTP、HTTP、SLP 或 NFS 伺服器等網路來源來安裝 &product。每一種作法都必須實際存取要安裝的系統，而且安裝過程中需要使用者互動。安裝程序基本上相同，無論其安裝來源為何。

3.3.1 開機選項

如果從 CD 或 DVD 開機發生問題，可以使用其他開機選項。如需這些選項的詳細資訊，請參閱表格 3.1「開機選項」[18頁]。

表格 3.1 *開機選項*

開機選項	描述
DVD/CD-ROM	這是最簡單的開機選項。如果系統上的本地 CD/DVD-ROM 光碟機受 Linux 支援，便可以使用這個選項。
磁片	用來產生開機磁片的影像位於 CD/DVD 1 的 `/boot/` 目錄中。同一目錄中還有一個 `README`。

開機選項	描述
PXE 或 BOOTP	只有在系統的BIOS或韌體支援而且網路上有可用的開機伺服器的情況，才支援這個選項。這個作業也可以由其他 SUSE Linux Enterprise 系統處理。
硬碟	SUSE Linux Enterprise 也可以從硬碟開機。要這樣做，請從 CD/DVD 1 的 `/boot/loader` 目錄將核心 (`linux`) 和安裝系統 (`initrd`) 複製到硬碟，並在開機載入程式中新增適當的項目。

3.3.2 從 SUSE Linux Enterprise 媒體安裝

若要從媒體安裝，請將第一張 CD 或 DVD 插入系統中適當的磁碟機中以進行安裝。從媒體重新啟動系統，並開啟開機畫面。

3.3.3 從使用 SLP 的網路伺服器安裝

如果網路設定支援 OpenSLP，且網路安裝來源已設定為透過 OpenSLP 來宣告自己 (如 第 4.2 節「安裝保存安裝來源的伺服器」 [51頁] 中所述)，請從媒體或使用其他開機選項來啟動系統。在開機畫面中，選取想要的安裝選項。按 F4，然後選取「*SLP*」。

安裝程式會擷取使用 OpenSLP 的網路安裝來源位置，並使用 DHCP 設定網路連線。如果 DHCP 網路組態失敗，將會出現提示要您手動輸入適當的參數。安裝程序將依以下的說明繼續。

3.3.4 從沒有 SLP 的網路來源安裝

如果您的網路設定不支援 OpenSLP 來取回網路安裝來源，請從媒體或其他開機選項啟動系統。在開機畫面中，選取想要的安裝選項。按 F4，然後選取所需的網路通訊協定 (NFS、HTTP、FTP 或 SMB)。提供伺服器位址及安裝媒體的路徑。

安裝程式會擷取使用 OpenSLP 的網路安裝來源位置，並使用 DHCP 設定網路連線。如果 DHCP 網路組態失敗，將會出現提示要您手動輸入適當的參數。安裝程序將依以下的說明繼續。

3.4 安裝工作流程

SUSE Linux Enterprise 安裝過程分為三個主要部分：準備、安裝和組態設定。在準備階段，您可以設定一些基本參數，例如語言、時間和桌面類型。在安裝階段，您可以決定要安裝的軟體、在何處安裝以及如何啟動安裝的系統。安裝完成後，機器將重新開機進入新安裝的系統，並開始組態設定。在此階段，您可以設定使用者和密碼、網路和網際網路存取權，以及硬體元件 (如印表機)。

3.5 開機畫面

開機畫面顯示一些安裝程序的選項。「*從硬碟開機*」會從已安裝的系統開機，且預設為選取該選項，因為 CD/DVD 通常還留在光碟機中。若要安裝系統，請以方向鍵選取一個安裝選項。這些選項包含：

安裝
: 一般安裝模式。將啟用所有先進的硬體功能。將啟用所有先進的硬體功能。

「*安裝 — ACPI 已停用*」
: 如果一般安裝失敗，可能是因為系統硬體不支援 ACPI (進階組態及電源介面)。如果是這種情況，請使用這個選項安裝但不要 ACPI 支援。

「*安裝 — 本地 APIC 已停用*」
: 如果正常安裝失敗，可能是因為系統硬碟不支援本地 APIC (進階可程式中斷控制卡)。如果發生這種情形，請在不使用本地 APIC 支援情況下使用這個選項進行安裝。

  如果您不確定，請先嘗試以下其中一個選項：「*安裝—ACPI 已停用*」或「*安裝—安全設定*」。

「*安裝 — 安全設定*」
: 使用 DMA 模式 (用於光碟機) 開機並關閉電源管理的功能。

「*救援系統*」
啟動不包含圖形使用者介面的精簡型Linux系統。若需要更多的資訊，請參閱章節「使用救援系統」[853頁]。

「*記憶體測試*」
以重複讀取及寫入週期的方式來測試您系統的RAM。透過重新開機來終止測試。若需要更多的資訊，請參閱第 51.2.5 節「無法開機」[831頁]。

選單中的安裝選項只會停用最有問題的功能。如果您必須停用或設定其他功能，請使用「*開機選項*」提示。在`http://en.opensuse.org/Linuxrc`中可找到核心參數的詳細資訊。

使用畫面底端的列中指出的功能鍵來變更語言、顯示器解析度或安裝來源，或新增您硬體廠商的其他驅動程式：

F1「*說明*」
取得開機畫面中使用中元素的內容感應式說明。

F2「*語言*」
選取安裝的顯示語言。預設語言為「英語」。

F3「*視訊模式*」
選取安裝的各種圖形顯示模式。如果圖形安裝會造成問題，請選取「*文字模式*」。

F4「*來源*」
通常是從插入的安裝媒體執行安裝。請在此處選取其他來源，如FTP或NFS伺服器。如果安裝是在具有 SLP 伺服器的網路上執行，請選取伺服器上有這個選項可用的其中一個安裝來源。如需有關 SLP 的詳細資訊，請參閱 第 31 章「*網路中的 SLP 服務*」[549頁]。

F5「*驅動程式*」
按下此鍵可以告訴系統您備有選用光碟，其中含有SUSE Linux Enterprise的驅動程式更新。使用「*檔案*」，可以在開始安裝之前直接從CD載入驅動程式。如果選取「*是*」，則安裝過程中會在適當時機提示您插入該更新光碟。預設選項是「*否*」— 不載入驅動程式更新。

開始安裝後，SUSE Linux Enterprise 會載入並設定精簡型 Linux 系統來執行安裝程序。若要在此過程中檢視開機訊息和著作權聲明，請按 Esc。完成此程序後，YaST 安裝程式將啟動並顯示圖形安裝程式。

提示：不使用滑鼠安裝

如果安裝程式未正確偵測到您的滑鼠，請使用 Tab 來導覽，使用箭頭按鍵來捲動，使用 Enter 確認選擇。

3.5.1 提供存取 SMT 伺服器所需的資料

如果您的網路有提供當成本地更新來源的SMT伺服器，則需要在用戶端設定該伺服器的 URL。用戶端和伺服器只能經由 HTTPS 通訊協定進行通訊，因此，在證書管理中心尚未發佈證書的情況下，您還需要輸入伺服器證書的路徑。這些資訊必須在開機提示處輸入。

smturl
: SMT 伺服器的 URL。URL 使用固定的格式：
`https://`*`FQN`*`/center/regsvc/` *`FQN`* 必須是 SMT 伺服器完全合法的主機名稱。範例：

```
smturl=https://smt.example.com/center/regsvc/
```

smtcert
: SMT 伺服器證書的位置。指定下列其中一個位置：

  URL
  : 可從中下載證書的遠端位置 (http、https 或 ftp)。範例：

    ```
    smtcert=http://smt.example.com/smt-ca.crt
    ```

  磁片
  : 指定磁片中的某個位置。開機時必須插入磁片，如果找不到磁片，系統不會提示您插入。該值必須以字串 `floppy` 開頭，後面是證書的路徑。範例：

    ```
    smtcert=floppy/smt/smt-ca.crt
    ```

  本地路徑
  : 本地機器上證書的絕對路徑。範例：

    ```
    smtcert=/data/inst/smt/smt-ca.cert
    ```

  互動
  : 在安裝期間可以使用`ask` 開啓一個快顯功能表，您可在其中指定證書的路徑。對於 AutoYaST 請勿使用此選項。範例

```
smtcert=ask
```

停用證書安裝
: 如果證書將由附加產品安裝，或者您使用的是官方證書管理中心發佈的證書，請使用 `done`。範例：

```
smtcert=done
```

警告：請小心避免輸入錯誤

確保輸入值正確。如果未正確指定 `smturl`，則更新來源的註冊作業將會失敗。如果為 smtcert 輸入的值不正確，系統會提示您提供證書的本地路徑。

如果不指定 smtcert，則它的值將預設為 `http://FQN/smt.crt`，其中 `FQN` 表示 SMT 伺服器的名稱。

3.6 語言

在一般情況下，您可以依需求將 YaST 和 SUSE Linux Enterprise 設成使用不同語言。此處選取的語言也會用於鍵盤配置。另外，YaST 將使用語言設定來猜測系統時鐘的時區。這些設定可於稍後選擇將次要語言安裝至系統時進行修改。

在安裝期間，您稍後可以依照 第 3.12 節「安裝設定」 [28頁] 中的說明變更語言。如需已安裝系統中語言設定的相關資訊，請參閱 第 8.1 節「YaST 語言」 [118頁]。

3.7 IBM System z：硬碟組態

在 IBM System z 平台進行安裝時，語言選取對話方塊之後會顯示一個用於對連接的硬碟進行設定的對話方塊。請選取 DASD、光纖通道附加 SCSI 磁碟 (zFCP) 或 iSCSI 來安裝 SUSE Linux Enterprise。

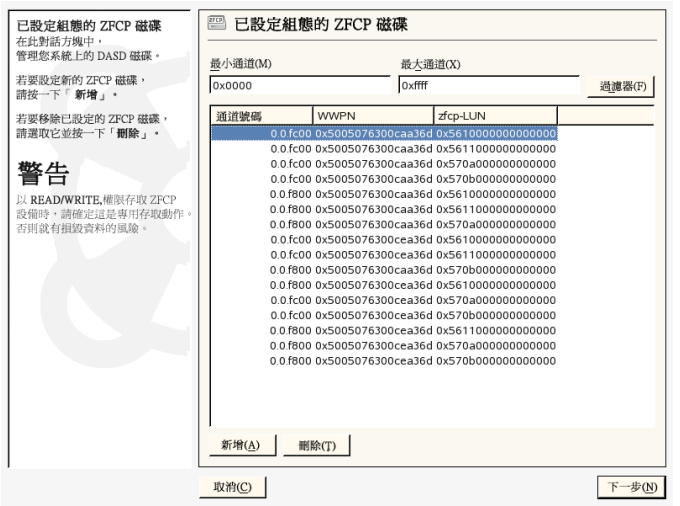
選取「*設定 DASD 磁碟*」後，會顯示列出所有可用 DASD 的綜覽。要取得更詳細的可用設備資料，請使用清單上方的輸入欄位來指定要顯示的通道範圍。要依據這樣的範圍過濾清單，請選取「*過濾器*」。請參閱圖形 3.1「IBM System z：選取 DASD」 [24頁]。

圖形 3.1　IBM System z：選取 DASD

現在，請在清單中選取對應的項目然後按一下「*選取或取消選取*」，指定安裝使用的 DASD。然後，請選取「*執行動作*」>「*啟用*」來啟用並使用 DASD 進行安裝。請參閱圖形 3.2「IBM System z：啟用 DASD」[24頁]。若要格式化 DASD，請立即選取「*執行動作*」>「*格式化*」，或在稍後使用 YaST 磁碟分割程式 (詳細步驟說明請參閱第 8.5.7 節「使用 YaST 磁碟分割程式」[141頁])。

圖形 3.2　IBM System z：啟用 DASD

圖形 3.3　IBM System z：可用 zFCP 磁碟綜覽



若要使用 zFCP 磁碟進行 SUSE Linux Enterprise 安裝，請在選擇對話方塊中選取「*設定 zFCP 磁碟*」。這將會開啟一個對話方塊，列出系統上可用 zFCP 磁碟的清單。在此對話方塊中，選取「*新增*」開啟另一個用於輸入 zFCP 參數的對話方塊。請參閱 圖形 3.3「IBM System z：可用 zFCP 磁碟綜覽」 [25頁]。

若要建立一個可用來安裝 SUSE Linux Enterprise 的 zFCP 磁碟，請從下拉式清單中選擇可用的「*通道號碼*」。「*取得 WWPN*」(全球埠號碼) 和「*取得 LUN*」(邏輯單元編號) 會分別傳回可用的 WWPN 與 FCP-LUN 清單，供您選擇。完成後，按「*下一步*」離開 zFCP 對話方塊，按一下「*完成*」離開一般硬碟組態對話方塊，以繼續其餘的組態設定。

提示：在以後新增 DASD 或 ZFCP 磁碟

在安裝工作流程期間與在顯示安裝建議時，都可以新增 DASD 或 ZFCP 磁碟。若要在那時新增磁碟，請按一下「*進階*」並向下捲動。DASD 與 ZFCP 項目顯示在最底部。

新增磁碟後，請重新讀取分割區表。返回安裝建議畫面並選擇「*製作分割*」，然後選取「*重新讀取分割區表*」。這樣可以讀取新的分割區表並重設先前輸入的資訊。

3.8 媒體檢查

僅當您透過基於已下載ISO建立的媒體進行安裝時，才顯示媒體檢查對話方塊。如果您透過原始媒體集安裝，則會跳過此對話方塊。

媒體檢查功能會檢驗媒體的完整性。若要開始媒體檢查，請選取包含安裝媒體的磁碟機，然後按一下「*開始檢查*」。檢查可能會持續一段時間。

若要測試多個媒體，請等到對話方塊中出現結果訊息後再更換媒體。如果最後檢查的媒體不是您啟動安裝時使用的媒體，YaST將提示您提供適當的媒體，然後才可繼續安裝。

警告：媒體檢查失敗

如果媒體檢查失敗，則表明您的媒體已損壞。此時請勿繼續安裝，因為安裝可能會失敗，或者您會遺失資料。請更換受損的媒體，然後重新開始安裝程序。

如果媒體檢查的結果為通過，請按「*下一步*」繼續安裝。

3.9 授權書

詳細閱讀畫面上顯示的授權書。如果同意這些授權條款，請選擇「*是，我同意「授權書」的內容*」，然後按一下「*下一步*」，確認您的選擇。如果不同意授權書，則不能安裝 SUSE Linux Enterprise，而且安裝將會結束。

3.10 安裝模式

在系統分析 YaST 嘗試在您的機器上尋找其他已安裝系統或現有 SUSE Linux Enterprise 系統，YaST 會顯示可用的安裝模式：

「*新的安裝*」
選取此選項從頭開始新安裝。

「*更新現有系統*」
選取此選項以更新為較新版本。如需系統的詳細資訊，請參閱第 10 章「*更新 SUSE Linux Enterprise*」[191頁]。

「*其他選項*」
此選項讓使用者有機會中止安裝，並改為開機或修復已安裝系統。若要啟動已安裝的 SUSE Linux Enterprise，請選取「*啟動已安裝系統*」。如果您在啟動安裝的 SUSE Linux Enterprise 時遇到問題，請參閱第 51.3 節「開機問題」[835頁]。

若要修復啟動失敗的已安裝系統，請選取「*修復已安裝系統*」。如需系統修復選項的說明，請參閱章節「使用 YaST 系統修復」[849頁]。

注：更新安裝的系統

只有已安裝舊 SUSE Linux Enterprise 系統時才會更新。如果沒有已安裝的 SUSE Linux Enterprise 系統，您只能執行全新安裝。

您可以選擇在首次安裝期間一起安裝 SUSE Linux Enterprise 系統與附加產品，或在將來隨時安裝附加產品，如第 8.3.2 節「安裝附加產品」[126頁]所述。附加產品 (Add-on) 就是的延伸程式 SUSE Linux Enterprise。附加產品可包含專屬協力廠商產品，或您系統的額外軟體。

若要在 SUSE Linux Enterprise 安裝期間包含附加產品，請選取「*包含不同媒體的附加產品*」，並按一下「*下一步*」。在下一個對話方塊中，請按一下「*新增*」以選取所安裝附加產品的來源。有許多可用的來源類型，例如 CD、FTP 或本機目錄。新增附加媒體後，您可能要同意協力廠商產品的其他授權合約。

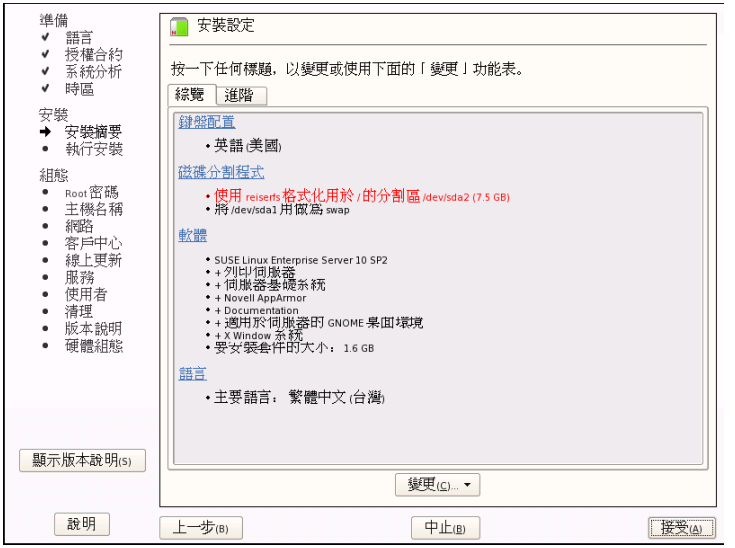
3.11 時鐘和時區

在此對話方塊中，從清單中選擇地區和時區。安裝期間，會根據選取的安裝語言，預先選取這兩個設定。在「*硬體時鐘設為*」下，選擇「*本地時間*」或「*UTC*」(GMT)。選擇由您機器上的 BIOS 硬體時鐘的設定決定。如果已經設成 UTC 對應的 GMT，則您的系統可由 SUSE Linux Enterprise 自動切換標準時間和日光節約時間。按一下「*變更*」來設定目前的日期和時間。完成時，按一下「*下一步*」繼續安裝。

3.12 安裝設定

完整的系統分析後，YaST 會提供合理的安裝設定建議。可在「*綜覽*」索引標籤中變更基本設定，而「*進階*」索引標籤中提供了進階選項。若要修改建議，請按一下「*變更*」，然後選取要變更的類別；或者按一下標題之一。設定好在這些對話方塊中顯示的項目後，一定要回到已經顯示變更的摘要視窗。

圖形 3.4 *安裝設定*



提示：將變更重設為預設值

您也可以按一下「*變更*」 > 「*重設為預設*」將所有變更重設為預設值。YaST 接著會再次顯示原始建議。

3.12.1 綜覽

在一般安裝情況下有時需要手動設定的選項會出現在「*概觀*」索引標籤。在此處修改「磁碟分割」、「軟體選項」和「地區設定」的設定值。

鍵盤配置

若要變更鍵盤配置，請選取「*鍵盤配置*」。依預設，鍵盤配置與安裝時選擇的語言相對應。從清單中選取鍵盤配置。使用對話方塊底部的「*測試*」檢查您是否可以正確輸入該配置的特殊字元。有關變更鍵盤配置的詳細資訊，請參閱第 8.4.10 節「鍵盤配置」[136頁]。完成後，按一下「*接受*」回到安裝摘要。

► **zseries:** 在 IBM System z 平台上，安裝程序是從遠端終端機執行。這些主機沒有直接連接的鍵盤或滑鼠。 ◄

磁碟分割

大部分情況下，YaST 會建議合理的磁碟分割綱要，您可以直接接受使用而不用進行任何變更。YaST 也可用於自定磁碟分割，但應只有經驗豐富的使用者才能變更磁碟分割。

當您第一次選取「*磁碟分割*」時，YaST 磁碟分割對話方塊會顯示建議的分割區設定。若要接受這些設定，請按一下「*接受建議*」。

若要在建議中進行微幅變更，請選取「*此建議上的基礎分割區設定*」，並在下一個對話方塊中調整磁碟分割。如需完全不同的磁碟分割，請選取「*建立自定的分割區設定*」。在下一個對話方塊中，選擇一個要分割的特定磁碟；如果希望存取所有磁碟，請選擇「*自定分割區*」。如需關於自定分割區的詳細資訊，請參閱 第 8.5.7 節「使用 YaST 磁碟分割程式」[141頁] SUSE Linux Enterprise Server 文件。YaST 分割程式也提供了一個 LVM 建立工具。若要建立 LVM 建議，請選取「*建立 LVM 基礎建議*」。有關 LVM 的詳細資訊，請參閱第 7.1 節「LVM 組態」[103頁]。

注：在 z/VM 中使用迷你磁碟

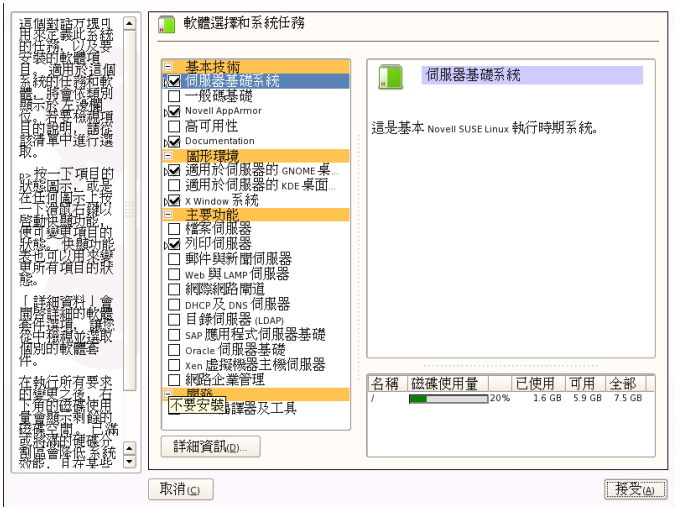
若 SUSE Linux Enterprise Server 安裝在 z/VM 中位於相同實體磁碟的幾個迷你磁碟上，則這些迷你磁碟的存取路徑 (/dev/disk/by-id/) 並不是唯一的，而是實體磁碟的 ID。因此，若兩個或多個迷你磁碟位於相同的實體磁碟上，則所有迷你磁碟都擁有相同的 ID。

為避免在裝載迷你磁碟時出現問題，請一律「依路徑」或「依 UUID」進行裝載。

軟體

SUSE Linux Enterprise 含有許多滿足各種應用程式目的所需的軟體套件。按一下建議視窗中的「*軟體*」來啟動軟體選擇，並依據個人需要來修改安裝範圍。從中間的清單中選取您的模式，並閱讀視窗右側的說明。每個模式都包含特定功能所需的眾多軟體套件 (例如多媒體或 Office 軟體)。如需要安裝之軟體套件的詳細選項，請選取「*詳細資料*」，以切換到 YaST 軟體管理員。請參閱圖形 3.5「使用 YaST 軟體管理員安裝和移除軟體」[30頁]。

圖形 3.5 *使用 YaST 軟體管理員安裝和移除軟體*



日後您也可以隨時安裝額外的軟體套件，或將軟體套件從您的系統中移除。若需更多資訊，請參考 第 8.3.1 節「安裝和移除軟體」[119頁].

注：預設桌面

SUSE Linux Enterprise 的預設桌面是 GNOME。若要安裝 KDE，請按一下「*軟體*」，並在「*圖形環境*」中選取「*KDE 桌面環境*」。

語言

若要變更系統語言或設定支援次要語言，請選取「*語言*」。從清單中選取一種語言。主要語言會做為系統語言。選擇次要語言，以便將來隨時都可切換到其

中的一種語言，而不必安裝附加的套件。有關更詳細的資訊，請參閱 第 8.5.15 節「語言選擇」 [149頁].

3.12.2 進階

如果您是進階使用者，並要設定開機或變更時區或預設執行層級，並選取「*專家*」索引標籤。它會顯示以下「*綜覽*」索引標籤中未包含的額外項目：

「*系統*」
這個對話方塊顯示 YaST 可取得有關電腦的所有硬體資訊。選擇清單中任一項目，然後按一下「*詳細資訊*」，檢視選取項目的詳細資訊。進階使用者還可以透過選擇「*系統設定*」來變更 PCI ID 設定和內核設定。

「*附加產品*」
在綜覽中會出現附加媒體的新增來源。在您開始安裝 SUSE Linux Enterprise 之前，請先視需要新增、移除或修改附加產品。

「*開機*」

► **zseries:** 此模組不能用來在 IBM System z 平台上設定開機載入程式 (zipl)。 ◄

YaST 會建議您系統的開機組態。通常，您不需要變更這些設定。不過，如果您需要自定的設定，請修改針對您的系統提供的建議。如需更多資訊，請參閱 第 21.3 節「使用 YaST 設定開機載入程式」 [380頁]。

「*時區*」
這與前面第 3.11 節「時鐘和時區」 [27頁]中顯示的組態相同。

預設的 Runlevel
SUSE Linux Enterprise 可以開機到不同的 Runlevel。通常不需要在此處變更任何設定，但如果需要，請使用此對話方塊設定預設的 runlevel。如需有關 Runlevel 設定的資訊，請參閱第 20.2.3 節「使用 YaST 設定系統服務 (Runlevel)」 [367頁]。

3.13 執行安裝

完成所有安裝設定後，請按一下建議視窗中的「*接受*」，開始安裝。確認「*安裝*」。某些軟體可能需要授權確認。如果您的軟體選項包含這類軟體，則會顯示授權確認對話方塊。按一下「*接受*」開始安裝軟體。如果您不同意授權合約，請按一下「*我不同意*」，這樣將不會安裝軟體。

視系統效能與所選軟體而定，安裝通常需要 15 至 30 分鐘。 在此程序中，將出現一份介紹 SUSE Linux Enterprise 的功能。選擇「*詳細資料*」可切換到安裝記錄。所有套件安裝完成後，YaST 將開機進入新的 Linux 系統，您接著可以設定硬體和系統服務。

3.13.1 IBM System z：對安裝的系統執行 IPL

在大多數情況下，YaST 會自動重新開機至 IBM System z 平台上的已安裝系統。例外情況是在機器上 LPAR 版本早於 z9 或 z/VM 版本早於 5.3 的環境中進行的安裝，其中的開機載入程式位於 FCP 設備上。開機載入程式會寫入至擁有 `/boot` 目錄的設備。若 `/boot` 不是位於單獨的分割區中，則會位於與根檔案系統 / 相同的分割區上。

在無法自動重新開機的情況下，YaST 會顯示一個對話方塊，其中包含要從何設備執行 IPL 的相關資訊。接受關機選項，並在關機後執行 IPL。該程序會因安裝類型而有所不同：

LPAR 安裝
: 在 IBM System z HMC 中，選取「*載入*」，再選取「*清除*」，然後輸入載入位址 (擁有包含開機載入程式之 `/boot` 目錄的設備的設備位址)。如果使用 ZFCP 磁碟做為開機設備，請選擇「*從 SCSI 載入*」並指定 FCP 介面卡的載入位址，以及開機設備的 WWPN 和 LUN。現在，請啟動載入程序。

z/VM 安裝
: 以 `LINUX1` 的身分登入 VM 客體 (有關組態，請參閱範例 2.2「Configuration of a z/VM Directory」 (↑*Architecture-Specific Information*))，並繼續對已安裝系統執行 IPL：

```
IPL 151 CLEAR
```

`151` 是 DASD 開機設備的一個範例位址，請使用正確的位址取代此值。

如果使用 ZFCP 磁碟作為開機設備，請在啟始化 IPL 之前，指定開機設備的 ZFCP WWPN 和 LUN。參數長度限制為 8 個字元。較長的數字必須用空格分隔：

```
SET LOADDEV PORT 50050763 00C590A9 LUN 50010000 00000000
```

最後，請起始 IPL：

```
IPL FC00
```

`FC00` 是 ZFCP 介面卡的一個範例位址，請使用正確的位址取代此值。

3.13.2 IBM System z：連接安裝的系統

安裝的系統完成 IPL 後，請建立連線來完成安裝。此處使用的步驟會因開始使用的連線類型而不同。

使用 VNC 進行連線

3270 終端機將顯示訊息，要求您使用 VNC 用戶端來連接 Linux 系統。不過，此訊息容易被遺漏，因為它會夾在核心訊息中，而且終端機處理程序可能在您看到訊息之前就已經結束。如果 5 分鐘內未發生任何情況，請嘗試使用 VNC 檢視程式來啟動與 Linux 系統的連線。

如果您使用具有 Java 功能的瀏覽器，請以下列格式輸入由已安裝系統的 IP 位址以及埠號碼組成的完整 URL：

```
http://<IP of installed system>:5801/
```

使用 X 進行連線

當您對已安裝系統執行 IPL 時，請確定安裝的第一個階段所使用的 X 伺服器處於開啟狀態且在從 DASD 開機之前仍然可用。YaST 會在這部 X 伺服器上開啟以完成安裝。如果系統已開機，但無法及時連接至 X 伺服器，則會發生問題。

使用 SSH 進行連線

重要： IBM System z：從 Linux 或 UNIX 系統進行連接

在 xterm 啟動 SSH。其他終端機模擬器未對 YaST 的文字介面提供完整的支援。

3270 終端機將出現訊息，要求您使用 SSH 用戶端來連接 Linux 系統。不過，這個訊息容易被遺漏，因為它會夾在核心訊息中，而且在您發覺之前終端機處理程序可能已經結束。

一旦出現訊息，請使用 SSH 做為 `root` 登入到 Linux 系統。如果連線被拒或超過時限，請稍等幾分鐘之後再試一次。

建立連線後，執行 `/usr/lib/YaST2/startup/YaST2.ssh` 指令。在這種情況下，yast 並未完成。

然後，YaST 將啟動以完成剩餘套件的安裝並建立初始的系統組態。

3.14 已安裝系統的組態

系統現在已安裝，但不會設定為使用。尚未設定使用者、硬體或服務。如果組態設定在進行到此階段的其中一個步驟時失敗，它會重新開始，並從最後一個成功的步驟繼續。

首先，提供系統管理員 (`root` 使用者) 的帳戶密碼。設定網際網路存取和網路連線。透過實際可用的網際網路連線，您可以執行系統更新並當作安裝的一部份。 您還可以連接至驗證伺服器，以集中管理本地網路中的使用者。最後，請設定與機器連接的硬體設備組態。

3.14.1 系統管理員 「root」 的密碼

`root` 是超級使用者，系統管理員的名稱。一般使用者在系統上可能有或可能沒有執行特定動作的權限，而 `root` 使用者則不同，她/他擁有不受限制的權力，可執行任何動作，包括變更系統組態、安裝各種程式、設定新硬體。如果使用者忘記密碼或遇到其他系統問題，`root` (根使用者) 可以提供協助。`root` (根)

帳戶應該僅用於系統的管理、維護和修復。以 root 身份登入來進行每日工作是相當危險的，因為一個錯誤就可能導致系統檔案遺失且無法恢復。

root (根) 密碼必須輸入兩次以進行確認。請不要忘記 root (根) 密碼。一旦輸入後，便無法查詢這個密碼。

輸入密碼時，字元將以點號取代，因此您看不到所輸入的字串。如果您不確定是否輸入了正確的字串，則可以使用「*測試鍵盤配置*」欄位來進行測試。

SUSE Linux Enterprise 可為密碼使用 DES、MD5 或 Blowfish 加密演算法。預設的加密類型為 Blowfish。若要變更加密類型，請按一下「*專家選項*」>「*加密類型*」，並選取新類型。

以後可以隨時在安裝的系統中變更 root。要執行此動作，請執行 YaST，然後啟動「*安全性與使用者*」>「*使用者管理*」。

3.14.2 主機名稱與網域名稱

主機名稱是電腦在網路中的名稱。網域名稱是網路名稱。預設會建議主機名稱和網域。如果您的系統是網路的組成部分，則主機名稱在此網路中必須是唯一的，而網域名稱必須由網路中所有主機共用。

在許多網路中，系統透過 DHCP 獲得其名稱。在這種情況下，需要修改主機名稱和網域名稱。請改為選取「*透過 DHCP 變更主機名稱*」。若希望可以使用此主機名稱存取系統 (即使系統未連接至網路)，請選取「*將主機名稱寫入 /etc/hosts*」。如果您經常需要在不重新啟動桌面環境的情況下變更網路 (例如，需要在不同的 WLAN 之間切換)，請勿啟用此選項，因為 /etc/hosts 中的主機名稱發生變更後，桌面系統可能會混淆。

若要在安裝後隨時變更主機名稱設定，請使用 YaST「*網路設備*」>「*網路卡*」。有關更詳細的資訊，請參閱第 30.4.1 節「使用 YaST 設定網路卡」[512頁].

3.14.3 網路組態

提示：IBM System z：網路組態

在 IBM System z 平台上，進行安裝時需要實際可用的網路連線，以便連接目標系統、安裝來源以及控制安裝程序的 YaST 終端機。網路設定步驟在

Architecture-Specific Information 手冊的網路組態章節中討論 (第 2 章「*Preparing for Installation*」(↑*Architecture-Specific Information*))。IBM System z 平台僅支援該章節中提及的網路介面類型 (OSA Token Ring、OSA Ethernet、OSA Gigabit Ethernet、OSA Express Fast Ethernet、Escon、IUCV 以及 OSA Express High-Speed Token Ring)。YaST 對話方塊僅會顯示已經設定設定值的介面。請在這個對話方塊進行確認並繼續。

「*不使用 NetworkManager Applet 的傳統方法*」預設為啟用。若需要，您也可使用NetworkManager來管理所有網路設備。但傳統方法是伺服器解決方案的優先選項。如需有關 NetworkManager 的詳細資訊，請參閱 第 30.6 節「使用 NetworkManager 管理網路連線」[529頁]。

此組態設定步驟還可讓您設定系統的網路設備，並進行防火牆或代理等安全性設定。若要日後設定您的網路連線，請選取「*略過組態*」，並按一下「*下一步*」。網路硬體也可在系統安裝完成後設定。如果您略過網路設備組態，您的系統會離線，且無法取回任何可用的更新。

除了設備組態設定外，還可以藉由此步驟完成下列網路設定：

「*網路模式*」
依上面的說明啟用或禁用 NetworkManager。

「*防火牆*」
依預設，SuSEfirewall2 已在所有設定的網路介面上啟用。若要為此電腦全面禁用防火牆，請按一下「*禁用*」。如果防火牆已啟用，您可以「*開啓*」SSH 連接埠，以便使用者可以透過安全外圍程序建立遠端連接。若要開啓詳細的防火牆組態設定對話方塊，請按一下「*防火牆*」。請參閱第 43.4.1 節「以 YaST 設定防火牆」[750頁]以獲得詳細資訊。

「*IPv6*」
依預設，已啟用 IPv6 支援。若要停用，按一下「*停用 IPv6*」。 欲知 IPv6 的更多資訊，請參閱 第 30.2 節「IPv6—下一代的網際網路」[503頁]。

「*VNC 遠端管理*」
若要透過 VNC 從遠端管理您的機器，請按一下「*變更*」>「*VNC 遠端管理*」啟用遠端管理，並在防火牆中開啟該連接埠。如果您有多個網路設備，而且想選取要在其上開啟連接埠的設備，請按一下「*防火牆詳細資料*」，並選取網路設備。您也可以使用比較安全的選項 SSH 進行遠端管理。

「*代理*」
如果您使用了代理伺服器來控制網路中的網際網路存取，則可以在此對話方塊中設定代理 URL 和驗證詳細資料。

提示：將網路組態設定為預設

按一下「*變更*」>「*重設為預設*」，將網路設定重設回原始建議值。這會放棄任何所做的變更。

測試網際網路連線

設定網路連線後，可對其進行測試。有基於此，YaST 會建立與 SUSE Linux Enterprise 伺服器的連線，並下載最新的版本說明。您會在安裝過程的最後看到。測試成功也是成功進行線上註冊和更新的前提。

如果有多個網路介面，請確認是否使用了所需的介面卡來連接到網際網路。如果不是，請按一下「*變更設備*」。

若要開始測試，請按一下「*是，測試與網際網路的連接*」，並按一下「*下一步*」。在下一個對話方塊中，檢視測試進度和測試結果。如需關於測試程序的詳細資訊，請選取「*檢視記錄*」。如果測試失敗，請按一下「*上一步*」返回到網路組態設定，並修正您的輸入。

如果現在不想測試連線，請選取「*否，略過此測試*」，接著按一下「*下一步*」。這也會跳過下載版本說明、設定客戶中心和線上更新。完成系統初始設定後，可以隨時執行這些步驟。

3.14.4 Novell Customer Center 組態

若要取得技術支援和產品更新，請先註冊並啟用您的產品。「*Novell 客戶中心組態*」提供助手可幫您完成這項作業。

如果您處於離線狀態，或想略過這個步驟，請選取「*稍後再設定*」。這也會略過 SUSE Linux Enterprise 線上更新。

在「*包含以方便使用*」中，選擇註冊時是否傳送未經請求的附加資訊。這樣可以簡化註冊程序。按一下「*詳細資料*」可取得關於資料隱私權和所收集資料的詳盡資訊，

該模組除了會啟用及註冊您的產品外，還可在組態中新增官方更新目錄。該目錄針對已知錯誤或安全性問題提供修正程式，您可以透過線上更新安裝這些程式。

若要使您的目錄維持有效，請選取「*定期與客戶中心同步化*」。此選項會檢查您的目錄，然後新增最新提供的目錄或移除過時的目錄。但它不涉及手動新增的目錄。

提示：技術支援

如需有關技術支援的詳細資訊，請造訪 http://www.novell.com/support/products/linuxenterpriseserver/。

3.14.5 線上更新

如果「*Novell Customer Center 組態*」設定成功，請選擇是否執行 YaST 線上更新。如果伺服器上有適用的修補套件，請立即下載並安裝以修復已知錯誤或安全性問題。第 8.3.5 節「YaST 線上更新」[128頁]上提供了如何在已安裝系統中執行線上更新的說明。

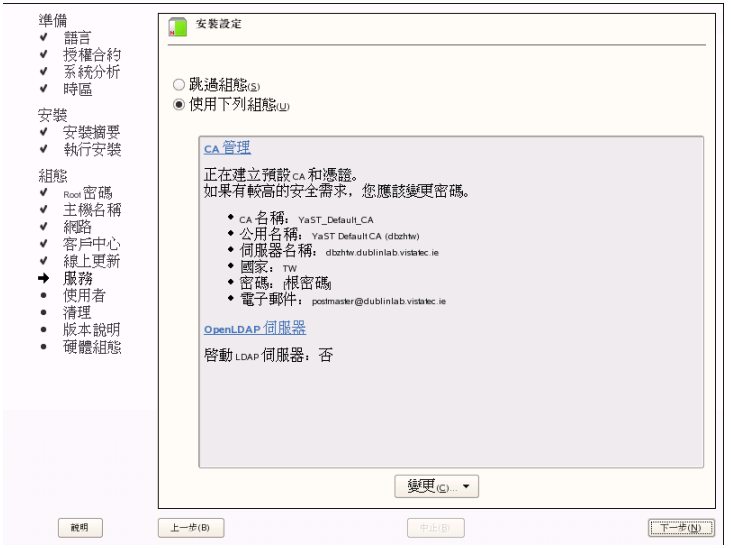
重要：下載軟體更新

下載更新內容可能需要較長時間，視網際網路連線的頻寬和更新檔案的大小而定。如果修補系統自身已經更新，則線上更新將會重新啟動，然後下載更多的修補程式。如果內核已更新，則系統會重新開機，然後完成組態設定。

3.14.6 網路服務

設定網路後會開啟一個對話方塊，您可在此啟用並設定兩個重要的網路服務：認證機構與 OpenLDAP 伺服器。如果您願意的話，現在可以跳過此組態建議。在安裝完成後，請透過 YaST 的協助來設定並啟動這兩個服務。

圖形 3.6　網路服務的建議設定



CA 管理
: CA (認證機構) 的目的在於保證所有網路服務之間的通訊都是信任關係。如果沒有 CA，您可以對每個服務分別使用 SSL 與 TLS 來保護伺服器通訊的安全。依預設，安裝期間將建立並啟用 CA。有關使用 YaST 建立 CA 的詳細信息，請參閱第 42 章「*管理 X.509 憑證*」[731頁]。

OpenLDAP 伺服器
: 您可以在主機上執行 LDAP 服務，來取得集中管理某個範圍的組態檔的工具。LDAP 伺服器通常用來管理使用者帳戶資料，但是在 SUSE Linux Enterprise 中，它還可以用於郵件、DHCP 和 DNS 資料的管理。如需有關 LDAP 以及使用 YaST 進行組態設定的詳細資訊，請參閱第 36 章「*LDAP—一種目錄服務*」[605頁]。

提示：將服務組態重設為預設值

透過按一下「*變更*」>「*重設成預設值*」即可還原預設值。這會放棄任何所做的變更。

3.14.7 使用者

如果已透過前幾個安裝步驟成功設定網路存取，則您現在可以在多個使用者管理選項中進行選擇。如果尚未設定網路連線，請建立本地使用者帳戶。有關使用者管理的詳細資訊，請參閱 第 8.9.1 節「使用者管理」[157頁]SUSE Linux Enterprise Server 文件。

本地 (/etc/passwd)
: 在安裝主機上本地管理使用者。這是單機工作站適用的選項。使用者資料由本地檔案 `/etc/passwd` 管理。輸入到這個檔案中的所有使用者都可以登入系統，即使沒有可用的網路。

  如果 YaST 發現早期版本的 SUSE Linux Enterprise，或者有另一個系統正在使用 `/etc/passwd`，則會提示輸入本地使用者。若要如此做，請勾選「*讀取上一版安裝的使用者資料*」，並按一下「*選擇*」。在下一個對話方塊中，選取要輸入的使用者，並按一下「*確定*」。

LDAP
: 在 LDAP 伺服器上集中管理網路中所有系統的使用者。如需詳細資訊，請參閱第 36.6 節「使用 YaST 設定 LDAP 用戶端」[623頁]。

NIS
: 在 NIS 伺服器上集中管理網路中所有系統的使用者。請參閱第 35.2 節「設定 NIS 用戶端」[603頁] 以取得詳細資訊。

Windows 領域
: Linux 與 Windows 混合網路中常使用 SMB 驗證。如需詳細資訊，請參閱第 37.6 節「含 Active Directory 的網路中之 Samba 伺服器」[645頁]。

注：驗證功能表的內容

如果您使用自定套件選項，且選單中缺少一或多個驗證方法，則或許不會安裝必要的套件。

您可以使用 Kerberos 驗證以及所選的使用者驗證方法。您的 SUSE Linux Enterprise 必須整合至 Active Directory 網域，如 第 37.6 節「含 Active Directory 的網路中之 Samba 伺服器」[645頁] 所述。若要使用 Kerberos 驗證，請選取「*設定 Kerberos 驗證*」。

3.14.8 版本說明

完成使用者驗證設定後，YaST 將顯示版本說明。建議您詳加閱讀，因為其中可能含有未列印在手冊中的最新資訊。如果您已經安裝更新套件，請閱讀從 SUSE Linux Enterprise 伺服器擷取的最新版本版本說明。使用「*其他*」>「*版本說明*」來檢視版本說明。

3.14.9 硬體組態

在安裝程序的最後，YaST 會開啟對話方塊以設定圖形卡以及連接至系統之其他硬體元件的組態。請按一下個別元件來啟動硬體組態設定。在大部分情況，YaST 會自動偵測設備並進行組態。

提示：IBM System z：硬體組態

在 IBM System z 上，沒有任何 XFree 支援的顯示器。因此，這些系統中找不到「*圖形介面卡*」項目。

您可以跳過任何周邊設備並在以後進行設定，如第 8.4 節「硬體」[133頁]所述。若要略過組態，請選取「*略過組態*」，並按一下「*下一步*」。

不過，您最好現在設定圖形卡。儘管 YaST 自動設定的顯示器設定值通常都可以被接受，但大部分的使用者對於解析度、色彩深度以及其他圖形功能卻有強烈的個人偏好。若要變更這些設定，請選取相關的項目並設定想要的值。若要測試新組態，請按一下「*測試組態*」。

提示：將硬體組態重設回預設值

您可以按一下「*變更*」>「*重設回預設值*」來取消變更。YaST 接著會再次顯示原始建議。

3.14.10 完成安裝

成功完成安裝後，YaST 會顯示「*安裝完成*」對話方塊。請在此對話方塊中選擇是否要為 AutoYaST 複製新安裝的系統。要執行此動作，請選取「*為 AutoYaST 複製這個系統*」。目前系統的設定檔儲存於 `/root/autoyast.xml`。預設會選取複製。

AutoYaST 這套系統可在不需使用者互動的情況下自動安裝一或多台 SUSE Linux Enterprise 系統。AutoYaST 安裝是使用包含安裝和組態資料的控制檔案來執行。如需詳細資訊，請參閱第 5 章「*自動安裝*」[77頁]。在最後一個對話方塊中，按一下「*完成*」，完成 SUSE Linux Enterprise 的安裝。

3.15 圖形登入

提示：IBM System z：沒有圖形登入

IBM System z 平台不提供圖形登入。

SUSE Linux Enterprise 至此已安裝並已設定。除非您啟用自動登入功能或自定預設 runlevel，您應該在您的畫面上看到圖形登入，在此您可輸入登入系統的使用者名稱和密碼。如果已啟用自動登入，則桌面會自動啟動。

4 遠端安裝

SUSE Linux Enterprise® 有幾種不同的安裝方法。您可以依照第 3 章「*使用 YaST 安裝*」[17頁]中的說明，從媒體進行一般安裝，或者您也可以選擇使用不同網路結構方式來安裝 SUSE Linux Enterprise，甚至採用完全自動的安裝方式。

每一個方法的簡介都透過兩個核對清單來進行：一個列出該方法的先決條件，另一個則說明基本程序。接著會介紹用於每個安裝方式的所有技術的詳細資訊。

注

以下章節中，將安裝新 SUSE Linux Enterprise 的系統稱做*目標系統*或*安裝目標*。*安裝來源*這個名詞則是用來表示所有安裝資料的來源，其中包括實體媒體 (例如 CD 和 DVD) 以及您網路中發佈安裝資料的網路伺服器。

4.1 遠端安裝的安裝方式

此章節將介紹最常用於遠端安裝的安裝方式。不論使用那個方式，都必須仔細檢查先決條件清單，並依照該安裝方式中說明的程序進行安裝。如果需要某個特定步驟的詳細說明，請連接至每個步驟中提供的連結以取得說明。

重要

X Window System 的組態不屬於任何遠端安裝程序中的一個部分。完成安裝後，請以 `root` 身份登入目標系統、輸入 `telinit 3`，接著啟動 SaX2 來設定圖形硬體。

4.1.1 透過 VNC 進行的簡易遠端安裝—靜態網路組態

此安裝類型仍需要某個程度的實體存取權限，以便存取並啟動目標系統來進行安裝。安裝本身完全受控於遠端工作站，遠端工作站會使用VNC連接至安裝程式來進行安裝。這時也必須執行像第 3 章「*使用 YaST安裝*」[17頁]所述手動安裝時的使用者互動。

使用此安裝類型時，請確定已符合以下需求：

- 遠端安裝來源：NFS、HTTP、FTP 或執行網路連線的 SMB
- 執行網路連線的目標系統
- 執行網路連線的控制系統和 VNC 檢視器軟體，或已啟用 Java 的瀏覽器 (Firefox、Konqueror、Internet Explorer 或 Opera)
- 用來啟動目標系統的實體開機媒體 (CD 或 DVD)
- 已指定用於安裝來源和控制系統的有效靜態 IP 位址
- 已指定到目標系統的有效靜態 IP 位址

若要執行此安裝類型，請按照下列步驟進行：

1. 依照第 4.2 節「安裝保存安裝來源的伺服器」[51頁] 中的說明安裝安裝來源。選擇 NFS、HTTP 或 FTP 網路伺服器。若要使用 SMB 安裝來源，請參閱第 4.2.5 節「管理 SMB 安裝來源」[57頁]。
2. 使用 SUSE Linux Enterprise 媒體套件的第一張 CD 或 DVD 啟動目標系統。
3. 出現目標系統的開機畫面時，請使用開機選項提示來設定適當的 VNC 選項和安裝來源的位址。如需詳細說明，請參閱第 4.4 節「啟動要安裝的目標系統」[69頁]。

   目標系統會啟動到以文字模式為基礎的環境，並提供網路位址和顯示編號，而 VNC 檢視器應用程式或瀏覽器必須使用這些資訊才可在圖形安裝環境下進行安裝。VNC 安裝會透過 OpenSLP 自我宣告，您可以使用 Konqueror 在 `service:/` 或 `slp:/` 模式中找到 VNC 安裝。

4 接著，請在控制工作站上開啟 VNC 檢視應用程式或網頁瀏覽器，並依照第 4.5.1 節「安裝 VNC」[74頁] 中的說明連接到目標系統。

5 依照第 3 章「*使用 YaST安裝*」[17頁]所述步驟執行安裝。目標系統重新開機後，重新連接至目標系統，以完成安裝的最後一個部分。

6 完成安裝。

4.1.2 透過 VNC 進行的簡易遠端安裝—動態網路組態

此安裝類型仍需要某個程度的實體存取權限，以便存取並啟動目標系統來進行安裝。請在 DHCP 定義網路組態。安裝本身完全受控於遠端工作站，遠端工作站會使用 VNC 連接至安裝程式，但使用者仍必須與安裝程式進行互動，以實際設定組態。

使用此安裝類型時，請確定已符合以下需求：

- 遠端安裝來源：NFS、HTTP、FTP 或執行網路連線的 SMB
- 執行網路連線的目標系統
- 執行網路連線的控制系統和 VNC 檢視器軟體，或已啟用 Java 的瀏覽器 (Firefox、Konqueror、Internet Explorer 或 Opera)
- 用來啟動目標系統的實體開機媒體 (CD、DVD 或自定的開機磁片)
- 提供 IP 位址的執行中 DHCP 伺服器

若要執行此安裝類型，請按照下列步驟進行：

1 依照 第 4.2 節「安裝保存安裝來源的伺服器」[51頁] 中的說明安裝安裝來源。選擇 NFS、HTTP 或 FTP 網路伺服器。若要使用 SMB 安裝來源，請參閱第 4.2.5 節「管理 SMB 安裝來源」[57頁]。

2 使用 SUSE Linux Enterprise 媒體套件的第一張 CD 或 DVD 啟動目標系統。

3 出現目標系統的開機畫面時，請使用開機選項提示來設定適當的 VNC 選項和安裝來源的位址。如需詳細說明，請參閱第 4.4 節「啟動要安裝的目標系統」[69頁]。

目標系統會啟動到以文字模式為基礎的環境，並提供網路位址和顯示編號，而 VNC 檢視器應用程式或瀏覽器必須使用這些資訊才可在圖形安裝環境下進行安裝。VNC 安裝會透過 OpenSLP 自我宣告，您可以使用 Konqueror 在 `service:/` 或 `slp:/` 模式中找到 VNC 安裝。

4 接著，請在控制工作站上開啟 VNC 檢視應用程式或網頁瀏覽器，並依照第 4.5.1 節「安裝 VNC」[74頁] 中的說明連接到目標系統。

5 依照第 3 章「*使用 YaST安裝*」[17頁]所述步驟執行安裝。目標系統重新開機後，重新連接至目標系統，以完成安裝的最後一個部分。

6 完成安裝。

4.1.3 透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能

此安裝類型為完全自動安裝。目標機器將經由遠端啟動。使用者只需在實際安裝時才需要與安裝程式進行互動。此方法適用於跨網站部署。

若要執行此安裝類型，請確定已符合以下需求：

- 遠端安裝來源：NFS、HTTP、FTP 或執行網路連線的 SMB
- TFTP 伺服器
- 執行用於您網路的 DHCP 伺服器
- 可以使用 PXE 啟動、網路喚醒的目標系統、插入和連接到網路
- 執行網路連線的控制系統和 VNC 檢視器軟體，或已啟用 Java 的瀏覽器 (Firefox、Konqueror、Internet Explorer 或 Opera)

若要執行此安裝類型，請按照下列步驟進行：

1 依照 第 4.2 節「安裝保存安裝來源的伺服器」[51頁] 中的說明安裝安裝來源。選擇一個 NFS、HTTP、FTP 網路伺服器，或設定 SMB 安裝來源，詳細步驟說明請參閱第 4.2.5 節「管理 SMB 安裝來源」[57頁]。

2 設定 TFTP 伺服器以存放目標系統所需的開機影像。如需詳細資訊，請參閱第 4.3.2 節「設定 TFTP 伺服器」[61頁]。

3 設定 DHCP 伺服器以向所有機器提供 IP 位址，並向目標系統顯示 TFTP 伺服器的位置。如需詳細資訊，請參閱第 4.3.1 節「設定 DHCP 伺服器」[59頁]。

4 準備用於 PXE 啟動的目標系統。如需進一步詳細說明，請參閱第 4.3.5 節「準備用於 PXE 啟動的目標系統」[67頁]。

5 使用網路喚醒功能啟動目標系統的開機程序。如需詳細資訊，請參閱第 4.3.7 節「區域網路喚醒」[68頁]。

6 接著，請在控制工作站上開啟 VNC 檢視應用程式或網頁瀏覽器，並依照第 4.5.1 節「安裝 VNC」[74頁] 中的說明連接到目標系統。

7 依照第 3 章「*使用 YaST安裝*」[17頁]所述步驟執行安裝。目標系統重新開機後，重新連接至目標系統，以完成安裝的最後一個部分。

8 完成安裝。

4.1.4 透過 SSH 進行的簡易遠端安裝—靜態網路組態

此安裝類型仍需要某個程度的實體存取權限，以便存取並啟動目標系統並決定安裝目標的 IP 位址。安裝本身完全受控於遠端工作站，遠端工作站會使用 SSH 連接至安裝程式。這時也必須執行像第 3 章「*使用 YaST安裝*」[17頁]所述一般安裝時的使用者互動。

使用此安裝類型時，請確定已符合以下需求：

- 遠端安裝來源：NFS、HTTP、FTP 或執行網路連線的 SMB
- 執行網路連線的目標系統

- 執行網路連線和 SSH 用戶端軟體的控制系統
- 用於目標系統的實體開機媒體 (CD、DVD 或自定的開機磁片)
- 已指定用於安裝來源和控制系統的有效靜態 IP 位址
- 已指定到目標系統的有效靜態 IP 位址

若要執行此安裝類型，請按照下列步驟進行：

1. 依照 第 4.2 節「安裝保存安裝來源的伺服器」 [51頁] 中的說明安裝安裝來源。選擇 NFS、HTTP 或 FTP 網路伺服器。若要使用 SMB 安裝來源，請參閱第 4.2.5 節「管理 SMB 安裝來源」 [57頁]。
2. 使用 SUSE Linux Enterprise 媒體套件的第一張 CD 或 DVD 啟動目標系統。
3. 出現目標系統的開機畫面時，請使用開機選項提示來設定適當的網路連線參數和安裝來源位址，並啟用 SSH。如需詳細說明，請參閱 第 4.4.3 節「使用自定開機選項」 [71頁]。

   目標系統會啟動到以文字模式為基礎的環境，並提供網路位址，而所有 SSH 用戶端必須使用這些資訊才可在圖形安裝環境下進行安裝。
4. 接著，請在控制工作站上開啟終端機視窗，並依照 章節「連接到安裝程式」 [76頁] 中的說明連接到目標系統。
5. 依照第 3 章「*使用 YaST安裝*」 [17頁]所述步驟執行安裝。目標系統重新開機後，重新連接至目標系統，以完成安裝的最後一個部分。
6. 完成安裝。

4.1.5 透過 SSH 進行的簡易遠端安裝—動態網路組態

此安裝類型仍需要某個程度的實體存取權限，以便存取並啟動目標系統並決定安裝目標的 IP 位址。安裝本身完全受控於遠端工作站，遠端工作站會使用 VNC 連接至安裝程式，但使用者仍必須與安裝程式進行互動，以實際設定組態。

使用此安裝類型時，請確定已符合以下需求：

- 遠端安裝來源：NFS、HTTP、FTP 或執行網路連線的 SMB
- 執行網路連線的目標系統
- 執行網路連線和 SSH 用戶端軟體的控制系統
- 用來啟動目標系統的實體開機媒體 (CD 或 DVD)
- 提供 IP 位址的執行中 DHCP 伺服器

若要執行此安裝類型，請按照下列步驟進行：

1. 依照 第 4.2 節「安裝保存安裝來源的伺服器」 [51頁] 中的說明安裝安裝來源。選擇 NFS、HTTP 或 FTP 網路伺服器。若要使用 SMB 安裝來源，請參閱第 4.2.5 節「管理 SMB 安裝來源」 [57頁]。
2. 使用 SUSE Linux Enterprise 媒體套件的第一張 CD 或 DVD 啟動目標系統。
3. 出現目標系統的開機畫面時，請使用開機選項提示來輸入適當的網路連線參數和安裝來源位置，並啟用 SSH。請參閱 第 4.4.3 節「使用自定開機選項」 [71頁]，以取得使用這些參數的詳細說明。

   目標系統會啟動到以文字模式為基礎的環境，並提供網路位址，而所有 SSH 用戶端必須使用這些資訊才可在圖形安裝環境下進行安裝。
4. 接著，請在控制工作站上開啟終端機視窗，並依照 章節「連接到安裝程式」 [76頁] 中的說明連接到目標系統。
5. 依照第 3 章「*使用 YaST安裝*」 [17頁]所述步驟執行安裝。目標系統重新開機後，重新連接至目標系統，以完成安裝的最後一個部分。
6. 完成安裝。

4.1.6 透過 SSH 進行的遠端安裝—PXE 開機和網路喚醒功能

此安裝類型為完全自動安裝。目標機器將經由遠端啟動。

若要執行此安裝類型，請確定已符合以下需求：

- 遠端安裝來源：NFS、HTTP、FTP 或執行網路連線的 SMB
- TFTP 伺服器
- 網路中目前提供靜態 IP 給將要安裝主機的執行中 DHCP 伺服器
- 可以使用 PXE 啟動、網路喚醒的目標系統、插入和連接到網路
- 執行網路連線和 SSH 用戶端軟體的控制系統

若要執行此安裝類型，請按照下列步驟進行：

1. 依照第 4.2 節「安裝保存安裝來源的伺服器」[51頁]中的說明安裝安裝來源。選擇 NFS、HTTP 或 FTP 網路伺服器。如需 SMB 安裝來源的組態資訊，請參閱第 4.2.5 節「管理 SMB 安裝來源」[57頁]。
2. 設定 TFTP 伺服器以存放目標系統所需的開機影像。如需詳細資訊，請參閱第 4.3.2 節「設定 TFTP 伺服器」[61頁]。
3. 設定 DHCP 伺服器以向所有機器提供 IP 位址，並向目標系統顯示 TFTP 伺服器的位置。如需詳細資訊，請參閱第 4.3.1 節「設定 DHCP 伺服器」[59頁]。
4. 準備用於 PXE 啟動的目標系統。如需進一步詳細說明，請參閱第 4.3.5 節「準備用於 PXE 啟動的目標系統」[67頁]。
5. 使用網路喚醒功能啟動目標系統的開機程序。如需詳細資訊，請參閱第 4.3.7 節「區域網路喚醒」[68頁]。
6. 接著，請在控制工作站上啟動 SSH 用戶端，並連接到目標系統，詳細步驟說明請參閱第 4.5.2 節「安裝 SSH」[75頁]。
7. 依照第 3 章「*使用 YaST安裝*」[17頁]所述步驟執行安裝。目標系統重新開機後，重新連接至目標系統，以完成安裝的最後一個部分。
8. 完成安裝。

4.2 安裝保存安裝來源的伺服器

根據做為 SUSE Linux Enterprise 網路安裝來源機器上所執行的作業系統的不同，有幾種伺服器組態選項可供使用。設定安裝伺服器最簡單的方法就是使用 SUSE Linux Enterprise Server 9 或 10 或 SUSE Linux 9.3 (和更新版本) 上的 YaST。若是使用其他版本的 SUSE Linux Enterprise Server 或 SUSE Linux Enterprise，請手動設定安裝來源。

提示

您甚至可以將安裝 Microsoft Windows 機器做為 Linux 部署的安裝伺服器。請參閱第 4.2.5 節「管理 SMB 安裝來源」 [57頁]以獲得詳細資料。

4.2.1 使用 YaST 設定安裝伺服器

YaST 會提供圖形工具，方便您建立網路安裝來源。它支援 HTTP、FTP 和 NFS 網路安裝伺服器。

1. 請以 `root` 身份登入要做為安裝伺服器的機器。
2. 依序啟動「*YaST*」>「*其他*」>「*安裝伺服器*」。
3. 選取伺服器類型 (HTTP、FTP 或 NFS)。之後，每次啟動系統時都會自動啟動選取的伺服器服務。如果您的系統上已在執行選取的服務類型，且您要手動設定該伺服器，請選取「*不要設定任何網路服務*」選項，停用伺服器服務的自動組態功能。在這兩種情況下，都必須定義安裝資料可用於伺服器上的目錄。
4. 設定所需的伺服器類型。這個步驟與伺服器服務的自動組態功能相關。若您停用自動組態功能，則請略過此步驟。

   為可找到安裝資料所在的 FTP 或 HTTP 伺服器根目錄定義別名。之後，您即可經由 `ftp://Server-Ip/Alias/Name` (FTP) 或 `http://Server-IP/Alias/NAME` (Http) 找到安裝來源。*Name* 代表安裝來源的名稱，可按照以下步驟定義。如果您在上一個步驟中選取 NFS，請定義萬用字元和輸出選項。您可以經由 `nfs://伺服器 IP/名稱`存取 NFS 伺服器。

提示：防火牆設定

請確定您伺服器系統的防火牆設定允許 HTTP、NFS 和 FTP 等連接埠上的傳輸。如果不允許，請啟動 YaST 防火牆模組並開啟個別的連接埠。

5 設定安裝來源。將安裝媒體複製到目的地前，請先定義安裝來源的名稱(最好是以產品縮寫和版本做為名稱，以方便記憶)。YaST 允許提供媒體的 ISO 影像，而不需使用安裝 CD 的副本。如果您要製作 ISO 影像，請啟用相關的核取方塊並指定本地存放 ISP 檔案的目錄路徑。根據要使用此安裝伺服器來發佈的產品而定，可能需要更多的附加產品 CD 或 Service Pack CD，需做為額外的安裝來源予以加入。若要透過 OpenSLP 宣告網路中的安裝伺服器，請啟用適當的選項。

提示

但是，請考慮您的網路設定是否支援該選項，再決定是否要透過 OpenSLP 宣告您的安裝來源。如此才可以確保每台目標機器皆可進入該網路安裝路徑。使用 SLP 開機選項啟動的目標系統將會尋找網路安裝來源，且您不需進一步設定任何組態。如需此選項的詳細資訊，請參閱第 4.4 節「啟動要安裝的目標系統」 [69頁]。

6 上傳安裝資料。設定安裝伺服器最長的步驟就是複製實際安裝 CD。請依照 YaST 要求的順序插入媒體，接著等待複製程式結束。來源複製完成後，請選取「*完成*」返回現有資訊來源的綜覽頁面，並關閉組態。

現在，您已完成安裝伺服器的設定，並可開始提供服務。之後每當啟動系統時，將會自動啟動該伺服器。您將不需再進行任何操作。如果您一開始就使用 YaST 停用選定網路服務的自動組態功能，那麼您只需正確地設定和啟動此服務即可。

若要停用安裝來源，請選取要移除的安裝來源，在選取「*刪除*」。安裝資料會從系統移除。若要停用網路服務，請使用個別的 YaST 模組。

如果您的安裝伺服器提供一個以上的產品版本，那麼請啟動 YaST 安裝伺服器模組，並在現有安裝來源的綜覽頁面中選取「*新增*」來設定新的安裝來源。

4.2.2 手動設定 NFS 安裝來源

基本上，設定 NFS 安裝來源只需兩個步驟。第一個步驟是，建立保存安裝資料的目錄結構，並將安裝媒體複製到此結構中。第二個步驟是，將包存安裝資料的目錄輸出至網路。

若要建立保存安裝資料的目錄，請執行下列步驟：

1. 以 `root` 的身份登入。
2. 建立一個之後要保存所有安裝資料的目錄，並將資料放入該目錄。例如：

   ```
   mkdir install/product/productversion
   cd install/product/productversion
   ```

   使用產品名稱的縮寫取代 *`product`*，並使用包含產品名稱和版本的字串來取代 *`productversion`*。
3. 請在插入媒體套件中的每張 CD 時執行以下指令：

   3a 將安裝 CD 中的所有內容複製到安裝伺服器目錄：

   ```
   cp -a /media/path_to_your_CD-ROM_drive .
   ```

   您的 CD 或 DVD 光碟機所在位置的實際路徑將會取代 *`path_to_your_CD-ROM_drive`*。根據您系統所使用光碟機類型的不同，可能會是 `cdrom`、`cdrecorder`、`dvd` 或 `dvdrecorder`。

   3b 重新命名 CD 編號的目錄：

   ```
   mv path_to_your_CD-ROM_drive CDx
   ```

   CD 的實際編號將會取代 *`x`*。

在 SUSE Linux Enterprise Server 上，您可以使用 YaST 透過 NFS 輸出安裝來源。請執行下列步驟：

1. 以 `root` 的身份登入。
2. 依序啟動「*YaST*」>「*網路服務*」>「*NFS 伺服器*」。

3 選取「*啟動*」和「*開啟防火牆中的連接埠*」，接著按一下「*下一步*」。

4 選取「*新增目錄*」並瀏覽至含有安裝來源的目錄，而這個範例中的是 *productversion*。

5 選取「*新增主機*」，並輸入安裝資料要輸出的目標機器主機名稱。此處除了可以指定主機名稱外，您還可以使用萬用字元、網路位址範圍或只要指定您網路的網域名稱即可。您可以選擇輸入適當的輸出選項，或者保留預設值 (在大多數設定中皆可正常運作)。如需更多有關輸出 NFS 共享所使用語法的詳細資訊，請參閱 `exports` man 頁面。

6 按一下「*完成*」。如此一來，存放 SUSE Linux Enterprise 安裝來源的 NFS 伺服器就會自動啟動，並將啟動該伺服器的步驟整合到開機程序中。

如果您想透過 NFS 手動輸出安裝來源，而不使用 YaST NFS 伺服器模組輸出安裝來源，請執行下列步驟：

1 以 `root` 的身份登入。

2 開啟檔案 `/etc/exports`，並輸入以下內容：

```
/productversion *(ro,root_squash,sync)
```

此操作可將目錄 */productversion* 輸出至網路中的任何一台主機，或任何可連接到這部伺服器的主機。若要限制存取此伺服器，請使用網路遮罩或網域名稱，而不要使用一般萬用字元 `*`。如需詳細資訊，請參閱 `export` 線上文件。儲存並結束此組態檔案。

3 若要新增 NFS 服務至系統啟動時要啟動的伺服器列表，請執行以下指令：

```
insserv /etc/init.d/nfsserver
insserv /etc/init.d/portmap
```

4 使用 `rcnfsserver start` 啟動 NFS 伺服器。如果您之後必須變更 NFS 伺服器的組態，請修改組態檔案並使用 `rcnfsserver restart` 重新啟動 NFS 精靈。

透過 OpenSLP 宣告 NFS 伺服器，可以讓您網路中的所有用戶端都知道該伺服器的位址。

1 以 `root` 的身份登入。

2 輸入目錄 `/etc/slp.reg.d/`。

3 建立名為 `install.suse.nfs.reg` 的組態檔案，檔案中必須包含以下內容：

```
# Register the NFS Installation Server
service:install.suse:nfs://$HOSTNAME/path_to_instsource/CD1,en,65535
description=NFS Installation Source
```

將 *`path_to_instsource`* 取代成您伺服器上安裝來源的實際路徑。

4 儲存此組態檔案，並使用 `rcslpd start` 啟動 OpenSLP 精靈。

如需 OpenSLP 的更多資訊，請參閱 `/usr/share/doc/packages/openslp/` 中的套件文件，或是參閱第 31 章「*網路中的 SLP 服務*」[549頁]。

4.2.3 手動設定 FTP 安裝來源

建立 FTP 安裝來源與建立 NFS 安裝來源的方式非常相似。您也可以使用 OpenSLP，透過網路來宣告 FTP 安裝來源。

1 依照第 4.2.2 節「手動設定 NFS 安裝來源」[53頁] 中的說明，建立一個保存安裝來源的目錄。

2 設定要發佈安裝目錄內容的 FTP 伺服器：

2a 以 `root` 身份登入，接著使用 YaST 套件管理員來安裝 `vsftpd` 套件。

2b 輸入 FTP 伺服器根目錄：

```
cd /srv/ftp
```

2c 在 FTP 根目錄中建立保存安裝來源的子目錄：

```
mkdir instsource
```

將 *`instsource`* 取代成產品名稱。

2d 請將安裝儲存庫中的內容裝載至 FTP 伺服器的變更根目錄環境中：

```
mount --bind path_to_instsource /srv/ftp/instsource
```

將 *path_to_instsource* 和 *instsource* 取代成符合設定的值。如果您必須讓此組態永久生效，請將此組態新增至 `/etc/fstab`。

2e 以 `vsftpd` 啟動 vsftpd。

3 如果您的網路設定支援 OpenSLP，請透過 OpenSLP 來宣告安裝來源：

3a 在 `/etc/slp.reg.d/` 中建立名為 `install.suse.ftp.reg` 的組態檔案，檔案中必須包含以下內容：

```
# Register the FTP Installation Server
service:install.suse:ftp://$HOSTNAME/instsource/CD1,en,65535
description=FTP Installation Source
```

將 *instsource* 取代成您伺服器上安裝來源目錄的實際名稱。`service:` 行的輸入內容必須維持連續一行。

3b 儲存此組態檔案，並使用 `rcslpd start` 啟動 OpenSLP 精靈。

4.2.4 手動設定 HTTP 安裝來源

建立 HTTP 安裝來源與建立 NFS 安裝來源的方式非常相似。您也可以使用 OpenSLP 透過網路宣告 HTTP 安裝來源。

1 依照 第 4.2.2 節「手動設定 NFS 安裝來源」 [53頁] 中的說明，建立一個保存安裝來源的目錄。

2 設定要發佈安裝目錄內容的 HTTP 伺服器：

2a 安裝網頁伺服器 Apache，詳細步驟說明請參閱第 40.1.2 節「安裝」 [674頁]。

2b 輸入 HTTP 伺服器的根目錄 (`/srv/www/htdocs`)，並建立要保存安裝來源的子目錄。

```
mkdir instsource
```

將 *instsource* 取代成產品名稱。

2c 建立一個從安裝來源位置連結到Web伺服器根目錄的符號連結(`/srv/www/htdocs`):

```
ln -s /path_instsource /srv/www/htdocs/instsource
```

2d 接著，請修改 HTTP 伺服器的組態檔案 (`/etc/apache2/default-server.conf`)，使其遵循符號連結。將下列一行：

```
Options None
```

與

```
Options Indexes FollowSymLinks
```

2e 使用 `rcapache2 reload` 重新載入 HTTP 伺服器組態。

3 如果您的網路設定支援 OpenSLP，請透過 OpenSLP 來宣告安裝來源：

3a 在 `/etc/slp/reg.d/` 中建立名為 `install.suse.http.reg` 的組態檔案，檔案中必須包含以下內容：

```
# Register the HTTP Installation Server
service:install.suse:http://$HOSTNAME/instsource/CD1/,en,65535
description=HTTP Installation Source
```

將 *`instsource`* 取代成您伺服器上安裝來源的實際路徑。`service:` 行的輸入內容必須維持連續一行。

3b 儲存此組態檔案，並使用 `rcslpd restart` 啟動 OpenSLP 精靈。

4.2.5 管理 SMB 安裝來源

使用 SMB，您可以從 Microsoft Windows 伺服器輸入安裝來源，並在沒有 Linux 機器的情況下啟動您的 Linux 部署。

若要設定存放 SUSE Linux Enterprise 安裝來源的輸出 Windows 共用，請執行下列步驟：

1 登入您的 Windows 機器。

2 啟動「檔案總管」來建立一個保存完整安裝樹狀結構的資料夾，並將資料夾命名為 INSTALL 或其他名稱。

3 根據您 Windows 文件中概述的程序輸出共享。

4 輸入此共享，並建立名稱為 *product* 的子資料夾。以實際產品名稱取代 *product*。

5 輸入 INSTALL/*product* 資料夾並複製所有 CD 或 DVD 到個別的資料夾，例如 CD1 和 CD2。

若要使用裝載的 SMB 共享作為安裝來源，請如下執行：

1 啟動安裝目標。

2 選取「*安裝*」。

3 按 F4 以選取安裝來源。

4 選擇 SMB，並輸入 Windows 的機器名稱或 IP 位址、共享名稱 (在此範例中為INSTALL/*product*/CD1)、使用者名稱和密碼。

YaST 會在您按下 Enter 之後啟動，這時您就可以開始執行安裝。

4.2.6 在伺服器上使用安裝媒體的 ISO 影像

您也可以將安裝媒體的 ISO 影像裝載至安裝伺服器做為安裝來源，而不需手動將實體媒體複製到伺服器目錄中。若要設定 HTTP、NFS 或 FTP 伺服器使用 ISO 影像 (而不使用媒體副本)，請按照下列步驟進行：

1 下載 ISO 影像並將其儲存至機器，做為安裝伺服器使用。

2 以 root 的身份登入。

3 請為安裝資料選擇並建立適當的位置，如第 4.2.2 節「手動設定 NFS 安裝來源」[53頁]、第 4.2.3 節「手動設定 FTP 安裝來源」[55頁] 或 第 4.2.4 節「手動設定 HTTP 安裝來源」[56頁]所述。

4 針對各 CD 或 DVD 建立子目錄。

5 若要將各個 ISO 影像裝載並解壓縮到最後的位置，請執行下列指令：

```
mount -o loop path_to_iso path_to_instsource/product/mediumx
```

以 ISO 影像的本地副本路徑取代 *path_to_iso*，以伺服器的來源目錄取代 *path_to_instsource*，以產品名稱取代 *product*，以您使用的媒體類型 (CD 或 DVD) 和號碼取代 *mediumx*。

6 請重複前面的步驟來為您的產品裝載所有必須的 ISO 影像。

7 請以您平常的方式啟動安裝伺服器，如第 4.2.2 節「手動設定 NFS 安裝來源」[53頁]、第 4.2.3 節「手動設定 FTP 安裝來源」[55頁]或第 4.2.4 節「手動設定 HTTP 安裝來源」[56頁]所述。

4.3 準備啟動目標系統

本節內容會介紹各種複雜開機方式的組態任務。其中包含 DHCP、PXE 啟動、TFTP 和網路喚醒等準備應用的組態。

4.3.1 設定 DHCP 伺服器

有兩種方法可設定 DHCP 伺服器。YaST 為 SUSE Linux Enterprise Server 9 和更新版本提供圖形介面，來處理這項工作。至於其他任何 SUSE Linux 產品的使用者和非 SUSE Linux 使用者，則應該手動編輯組態檔案，或使用作業系統廠商所提供的前端工具。

使用 YaST 設定 DHCP 伺服器

若要對網路用戶端宣告 TFTP 伺服器的位置，並指定安裝目標應使用的開機影像檔案，請在您的 DHCP 伺服器組態中新增兩個宣告。

1 以 `root` 身份登入代管 DHCP 伺服器的機器。

2 啟動「*YaST*」>「*網路服務*」>「*DHCP 伺服器*」。

3 完成基本 DHCP 伺服器設定的設定精靈。

4 選取「*進階設定*」，並在出現即將離開啟動對話方塊的警告時，選取「*是*」。

5 在「*設定的宣告*」對話方塊中，選取新系統所在位置的子網路，並按一下「*編輯*」。

6 在「*子網路組態*」對話方塊中，選取「*新增*」將新選項新增到子網路的組態中。

7 選取 `filename`，並輸入 `pxelinux.0` 做為值。

8 新增另一個選項 (`next-server`)，將它的值設為 TFTP 伺服器的位址。

9 選取「*確定*」和「*完成*」以完成 DHCP 伺服器組態。

若要將 DHCP 設定成為特定主機提供靜態 IP 位址，請進入 DHCP 伺服器組態模組 (步驟 4 [60頁]) 的「*進階設定*」，並新增主機類型的新宣告。在這個主機宣告中新增 `hardware` 和 `fixed-address` 選項，並提供適當的值。

手動設定 DHCP 伺服器

除了提供您網路用戶端自動位址配置，DHCP 伺服器需要執行的所有工作就是宣告 TFTP 伺服器的 IP 位址，以及宣告目標機器上安裝常式應使用的檔案。

1 以 `root` 身份登入代管 DHCP 伺服器的機器。

2 將下面幾行附加到 `/etc/dhcpd.conf` 下的 DHCP 伺服器組態檔案：

```
group {
  # PXE related stuff
  #
  # "next server" defines the tftp server that will be used
  next server ip_tftp_server:
  #
  # "filename" specifies the pxelinux image on the tftp server
  # the server runs in chroot under /srv/tftpboot
  filename  "pxelinux.0";
}
```

將 *`ip_of_the_tftp_server`* 取代成 TFTP 伺服器的實際 IP 位置。如需更多 `dhcpd.conf` 中可用選項的詳細資訊，請參閱 `dhcpd.conf` man 頁面。

3 執行 `rcdhcpd restart` 重新啟動 DHCP 伺服器。

如果您計畫使用 SSH 來遠端控制 PXE 和網路喚醒功能安裝，請明確指定 DHCP 應提供給安裝目標的 IP 位址。若要完成這項工作，請根據以下範例修改上述 DHCP 組態：

```
group {
  # PXE related stuff
  #
  # "next server" defines the tftp server that will be used
  next server ip_tftp_server:
  #
  # "filename" specifies the pxelinux image on the tftp server
  # the server runs in chroot under /srv/tftpboot
  filename "pxelinux.0";
  host test { hardware ethernet mac_address;
              fixed-address some_ip_address; }
      }
```

這項主機陳述式將引入安裝目標的主機名稱。若要繫結特定主機的主機名稱和 IP 位址，您必須瞭解並指定系統的硬體 (MAC) 位址。將此範例中的所有變數取代成符合您環境的實際值。

重新啟動 DHCP 伺服器之後，它將向所指定的主機提供一個靜態 IP，從而使您能夠透過 SSH 連接到該系統。

4.3.2 設定 TFTP 伺服器

請使用 YaST 在 SUSE Linux Enterprise Server 和 SUSE Linux Enterprise 上設定 TFTP 伺服器，或以手動方式在任何支援 xinetd 和 tftp 的 Linux 作業系統上設定 TFTP 伺服器。每當目標系統開機時，TFTP 伺服器即會傳送開機影像至目標系統，並傳送對目標系統的要求。

使用 YaST 設定 TFTP 伺服器

1 以 `root` 的身份登入。

2 啟動「*YaST*」>「*網路服務*」>「*TFTP 伺服器*」，並安裝所需套件。

3 按一下「*啟用*」，確定伺服器已經啟動，並已包含至開機常式。為安全起見，xinetd 在開機時啟動 tftpd 時，您不需執行任何動作。

4 按一下「*開啟防火牆中的連接埠*」，開啟在您機器上運作中防火牆的適當連接埠。如果您的伺服器上沒有執行任何防火牆，就無法使用此選項。

5 按一下「*瀏覽*」，瀏覽開機影像目錄。這時系統會建立預設目錄為 `/tftpboot`，而且會自動選取。

6 按一下「*完成*」便可套用您的設定，並啟動伺服器。

手動設定 TFTP 伺服器

1 以 `root` 身份登入，並安裝 `tftp` 和 `xinetd` 套件。

2 若無法安裝，請建立 `/srv/tftpboot` 和 `/srv/tftpboot/pxelinux.cfg` 目錄。

3 接著，請加入開機影像需要的正確檔案，詳細步驟說明請參閱第 4.3.3 節「使用 PXE 開機」[63頁]。

4 修改位於 `/etc/xinetd.d/` 下的 xinetd 組態，以確保開機時會啟動 TFTP 伺服器：

4a 如果 xinetd 不存在，請使用 `touch tftp`，在此目錄下建立一個名為 `tftp` 的檔案。接著執行 `chmod 755 tftp`。

4b 開啟檔案 `tftp`，並新增以下內容：

```
service tftp
{
        socket_type             = dgram
        protocol                = udp
        wait                    = yes
        user                    = root
        server                  = /usr/sbin/in.tftpd
        server_args             = -s /srv/tftpboot
        disable                 = no
}
```

4c 儲存檔案，並使用 `rcxinetd restart` 重新啟動 xinetd。

4.3.3 使用 PXE 開機

如需取得一些技術背景資訊和 PXE 的完整規格，請參閱「開機前執行環境 (Preboot Execution Environment, PXE) 規格」(http://www.pix.net/software/pxeboot/archive/pxespec.pdf)。

1 輸入以下指令，以變更安裝儲存庫的目錄，並將 `linux`、`initrd`、`message` 和 `memtest` 檔案複製到 `/srv/tftpboot` 目錄：

```
cp -a boot/loader/linux boot/loader/initrd
      boot/loader/message boot/loader/memtest /srv/tftpboot
```

2 請使用 YaST 直接從您的安裝 CD 或 DVD 安裝 `syslinux` 套件。

3 輸入以下指令，將 `/usr/share/syslinux/pxelinux.0` 檔案複製到 `/srv/tftpboot` 目錄：

```
cp -a /usr/share/syslinux/pxelinux.0 /srv/tftpboot
```

4 輸入以下指令，切換至安裝儲存庫的目錄，並將 `isolinux.cfg` 檔案複製到 `/srv/tftpboot/pxelinux.cfg/default`：

```
cp -a boot/loader/isolinux.cfg /srv/tftpboot/pxelinux.cfg/default
```

5 編輯 `/srv/tftpboot/pxelinux.cfg/default` 檔案，並移除開頭為 `gfxboot`、`readinfo` 和 `framebuffer` 等行文字。

6 接著，在預設的 `failsafe` 和 `apic` 標籤附加行中，插入下面項目：

`insmod=`*`kernel module`*
插入這個項目之後，便可輸入PXE用戶端上支援網路安裝所需的網路核心模組。以網路設備適當的模組名稱取代 *`kernel module`*。

`netdevice=`*`interface`*
這個項目可定義網路安裝時必須使用的用戶端網路介面。只有當用戶端配有數張網路卡時才必須插入這個項目，而且必須視情況修改項目內容。如果只有使用一張網路卡，就可以省略這個項目。

install=nfs://*ip_instserver*/*path_instsource*/CD1
: 這個項目可定義用戶端安裝時所適用的 NFS 伺服器和安裝來源。將 *ip_instserver* 取代成安裝伺服器的實際 IP 位址；而 *path_instsource* 應該要取代成安裝來源的實際路徑。HTTP、FTP 或 SMB 來源皆以類似方法指定位址，除了其通訊協定字首 `http`、`ftp` 或 `smb` 等差異。

重要

如果您需要傳送其他開機選項給安裝來源，例如 SSH 或 VNC 開機參數，那麼請將這些參數附加到 `install` 項目。您可以參考 第 4.4 節「啟動要安裝的目標系統」[69頁]，以取得參數綜覽和一些範例。

以下將介紹 `/srv/tftpboot/pxelinux.cfg/default` 範例檔案。調整安裝來源的通訊協定字首以符合您的網路設定，接著將 `vnc` 和 `vncpassword` 或 `usessh` 和 `sshpassword` 選項新增到 `install` 輸入內容，以指定連接到安裝程式的方式。這幾行內容會以 \ 分隔，而且應該是連續一行，中間不可間斷，也不能使用 \。

```
default linux

# default
label linux
  kernel linux
     append initrd=initrd ramdisk_size=65536 insmod=e100 \
     install=nfs://ip_instserver/path_instsource/product/CD1

# failsafe
label failsafe
  kernel linux
  append initrd=initrd ramdisk_size=65536 ide=nodma apm=off acpi=off \
  insmod=e100 install=nfs://ip_instserver/path_instsource/product/CD1

# apic
label apic
  kernel linux
  append initrd=initrd ramdisk_size=65536 apic insmod=e100 \
  install=nfs://ip_instserver/path_instsource/product/CD1

# manual
label manual
  kernel linux
  append initrd=initrd ramdisk_size=65536 manual=1
```

```
# rescue
label rescue
  kernel linux
  append initrd=initrd ramdisk_size=65536 rescue=1

#  memory test
label memtest
  kernel memtest

# hard disk
label harddisk
  localboot 0

implicit     0
display      message
prompt       1
timeout      100
```

將 *ip_instserver* 和 *path_instsource* 取代成在您設定中所使用的值。

下節內容可作為此設定中所使用的 PXELINUX 選項的簡短參考。如需更多可用選項的資訊，請參閱 /usr/share/doc/packages/syslinux/ 中的 syslinux 套件說明文件。

4.3.4 PXELINUX 組態選項

此處列出的選項為 PXELINUX 組態檔案中所有可用選項的子集合。

DEFAULT *kernel options...*
設定預設的核心指令行。當 PXELINUX 是自動啟動，則該選項的作用相當於在啟動提示處輸入了在 DEFAULT 後輸入的所有內容 (表示自動啟動的 auto 選項除外，它是自動新增的)。

如果目前沒有任何組態檔案，或是組態檔案中沒有任何 DEFAULT 項目，那麼預設值會是不包含任何選項的核心名稱「linux」。

APPEND *options...*
新增一個或多個選項至核心指令行。這些新增選項是用於自動和手動開機。新增選項會加在每個核心指令行的開頭位置，通常您只要明確輸入核心選項，便可覆寫這些選項。

LABEL *label* KERNEL *image* APPEND *options*...

指示是否要將輸入的 *label* 當作開機核心，或是 PXELINUX 應該要開機 *image*；以及是否要使用指定 APPEND 選項，而不是使用該檔案全域區段中的指定選項(在第一個 LABEL 指令前)。*image* 的預設值與 *label* 相同，而且如果沒有指定 APPEND，就會預設使用全域項目(如果有的話)。您最多可輸入 128 個 LABEL 項目。

請注意，GRUB 將使用以下語法：

```
title mytitle
  kernel my_kernel my_kernel_options
  initrd myinitrd
```

PXELINUX 使用以下語法：

```
label mylabel
  kernel mykernel
  append myoptions
```

標籤會像檔名一樣進行拆分(Mangle)，因此它們在拆分之後一定會是唯一專屬名稱。舉例來說，「v2.1.30」和「v2.1.31」這兩個標籤在 PXELINUX 中將無法辨識，因為它們都會拆分成相同的 DOS 檔案名稱。

這時的核心不一定要是 Linux 核心，它可以是開機磁區或 COMBOOT AE 案。

APPEND –

不附加任何項目。在 LABEL 區段中做為引數且包含單一連字號的 APPEND，可以用來覆寫全域的 APPEND。

LOCALBOOT *type*

在 PXELINUX 上，指定 LOCALBOOT 0 而不指定 KERNEL 選項，表示要呼叫此特定標籤；而且最後要使用本地磁碟開機，而不使用核心開機。

引數	描述
0	執行正常開機

引數	描述
4	使用仍常駐在記憶體中的「通用網路驅動程式介面」(Universal Network Driver Interface, UNDI) 執行本地開機。
5	使用仍常駐在記憶體中完整的 PXE 堆疊 (包括 UNDI 驅動程式) 執行本地開機。

其他所有值都未定義。如果您對 UNDI 或 PXE 堆疊不甚瞭解，請指定 0。

TIMEOUT *time-out*

指示在開機提示等待自動開機的時間長度，單位為 1/10 秒。使用者只要在鍵盤上按下任何一個鍵，就會取消逾時，這是假設使用者會完成已開始的指令。如果逾時時間設為 0，則會完全停用逾時功能 (此設定值亦為預設值)。可能的最大逾時值為 35996 (小於一小時)。

PROMPT *flag_val*

如果 flag_val 為 0，只有在按下 Shift 或 Alt，或者已設定 Caps Lock 或 Scroll Lock 時才會顯示開機提示 (此設定值為預設值)。如果 flag_val 為 1，則永遠會顯示開機提示。

```
F2  filename
F1  filename
..etc...
F9  filename
F10 filename
```

可在開機提示下按下功能鍵時，於螢幕畫面上顯示指定的檔案。這可用來執行開機前線上說明 (可能是針對核心指令行選項)。如果是使用較早版本的反向相容，F10 也可以當作 F0 輸入。請注意，目前還無法將檔案名稱繫結至 F11 和 F12。

4.3.5 準備用於 PXE 啟動的目標系統

請在 BIOS 開機順序中包含 PXE 選項，以便準備 PXE 開機的系統 BIOS。

警告：BIOS 開機順序

請勿將 PXE 選項置於 BIOS 硬碟開機選項之前。否則，此系統會在您每次重新開機時嘗試重新安裝系統。

4.3.6 準備用於網路喚醒功能的目標系統

您需要使用適當的 BIOS 選項才可在安裝前啟用網路喚醒 (WOL) 功能。另外，請記下目標系統的 MAC 位址。啟動網路喚醒功能時將會用到此資料。

4.3.7 區域網路喚醒

「網路喚醒功能」可藉由包含機器MAC位址的特定網路封包來開啟機器。因為世界上的每台機器都有一個獨一無二的MAC識別碼，所以您不必擔心會不小心開啟錯誤的機器。

重要：跨越不同網路區段的網路喚醒功能

如果控制機器與喚醒安裝目標的位置不在同一個網路區段，請將要傳送的WOL要求設定為多重廣播，或者遠端控制網路區段上傳送這些要求的機器。

SUSE Linux Enterprise Server 9 和更新版本的使用者可以使用名稱為 WOL 的 YaST 模組輕鬆設定網路喚醒功能。其他 SUSE Linux 版本作業系統的使用者則可以使用指令行工具。

4.3.8 使用 YaST 設定網路喚醒功能

1. 以 `root` 的身份登入。
2. 啟動「*YaST*」>「*網路服務*」>「*WOL*」。
3. 按一下「*新增*」，並輸入目標系統的主機名稱和 MAC 位址。
4. 若要開啟此機器，請選取適當的項目，並按一下「*喚醒*」。

4.3.9 網路喚醒功能

1 以 `root` 的身份登入。

2 啟動「*YaST*」>「*軟體管理*」，然後安裝 `netdiag` 套件。

3 開啟終端機，並以 `root` 身份輸入以下指令來喚醒目標：

```
ether-wake  mac_of_target
```

將 *`mac_of_target`* 取代成目標的實際 MAC 位址。

4.4 啟動要安裝的目標系統

基本上，除了第 4.3.7 節「區域網路喚醒」[68頁] 和第 4.3.3 節「使用 PXE 開機」[63頁] 中提到的方法，還有兩種不同的方法可以自定安裝的開機程序。您可以使用預設開機選項和功能鍵，或使用安裝開機畫面的開機選項提示，來傳送安裝核心可能需要用於此特定硬體的任何開機選項。

4.4.1 使用預設開機選項

開機選項已在第 3 章「*使用 YaST安裝*」[17頁]中詳細介紹過。一般而言，只要選擇「*安裝*」就可以開始安裝開機程序。

如果發生問題，請使用「*安裝—關閉 ACPI*」或「*安裝—安全設定*」。若需更多安裝程序疑難排解的資訊，請參閱第 51.2 節「安裝問題」[827頁]。

4.4.2 使用 F 鍵

畫面下方的功能表列會一些提供部分設定所需的進階功能。在不清楚參數詳細語法的情況下，您可以使用 F 鍵指定要傳送到安裝常式的其他選項 (請參閱第 4.4.3 節「使用自定開機選項」[71頁])。

請參閱下表瞭解可用選項的完整清單。

表格 4.1 *安裝時使用 F 鍵*

按鍵	目的	可用選項	預設值
F1	提供說明	無	無
F2	選取安裝語言	所有支援的語言	英文
F3	變更安裝的畫面解析度	• 文字模式 • VESA • 解析度 #1 • 解析度 #2 • ...	• 預設值會因您的圖形硬體而異
F4	選取安裝來源	• CD-ROM 或 DVD • SLP • FTP • HTTP • NFS • SMB • 硬碟	CD-ROM 或 DVD
F5	套用驅動程式更新磁片	驅動策	無

4.4.3 使用自定開機選項

使用適當的開機選項設定可以協助您進行安裝程序。您之後也可以使用 linuxrc 常式設定許多參數，但是使用開機選項則更方便。在某些自動化設定中，`initrd` 和 `info` 檔案會提供一些開機選項。

以下表格列出本章節中提及的所有安裝方式，其中包括開機所需的參數和對應的開機選項。您只要依照順序將此表格中出現的內容附加到檔案中，即可開機選項字串送到安裝常式中。例如 (全部在一行上)：

```
install=... netdevice=... hostip=...netmask=... vnc=... vncpassword=...
```

請使用適用於您設定的值來取代所有 值 (...)。

表格 4.2 *本章使用的安裝 (開機) 案例*

安裝方式	開機時的必要參數	開機選項
第 3 章「*使用 YaST安裝*」[17頁]	無：系統自動開機	不需要
第 4.1.1 節「透過 VNC 進行的簡易遠端安裝—靜態網路組態」[44頁]	• 安裝伺服器的位置 • 網路設備 • IP 位址 • 網路遮罩 • 閘道 • 啟用 VNC • VNC 密碼	• `install=(nfs,http, ftp,smb):///`*`path_to_instmedia`* • `netdevice=`*`some_netdevice`* (只有當存在數個網路設備時才需要) • `hostip=`*`some_ip`* • `netmask=`*`some_netmask`* • `gateway=`*`ip_gateway`* • `vnc=1` • `vncpassword=`*`some_password`*

安裝方式	開機時的必要參數	開機選項
第 4.1.2 節「透過 VNC 進行的簡易遠端安裝—動態網路組態」[45頁]	• 安裝伺服器的位置 • 啟用 VNC • VNC 密碼	• `install=(nfs,http, ftp,smb):///`*`path_to_instmedia`* • `vnc=1` • `vncpassword=`*`some_password`*
第 4.1.3 節「透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能」[46頁]	• 安裝伺服器的位置 • TFTP 伺服器的位置 • 啟用 VNC • VNC 密碼	不適用；透過 PXE 和 DHCP 管理的程序
第 4.1.4 節「透過 SSH 進行的簡易遠端安裝—靜態網路組態」[47頁]	• 安裝伺服器的位置 • 網路設備 • IP 位址 • 網路遮罩 • 閘道 • 啟用 SSH • SSH 密碼	• `install=(nfs,http, ftp,smb):///`*`path_to_instmedia`* • `netdevice=`*`some_netdevice`* (只有當存在數個網路設備時才需要) • `hostip=`*`some_ip`* • `netmask=`*`some_netmask`* • `gateway=`*`ip_gateway`* • `usessh=1` • `sshpassword=`*`some_password`*

安裝方式	開機時的必要參數	開機選項
第 4.1.5 節「透過 SSH 進行的簡易遠端安裝—動態網路組態」[48頁]	• 安裝伺服器的位置 • 啟用 SSH • SSH 密碼	• `install=(nfs,http,ftp,smb):///`*`path_to_instmedia`* • `usessh=1` • `sshpassword=`*`some_password`*
第 4.1.6 節「透過 SSH 進行的遠端安裝—PXE 開機和網路喚醒功能」[49頁]	• 安裝伺服器的位置 • TFTP 伺服器的位置 • 啟用 SSH • SSH 密碼	不適用；透過 PXE 和 DHCP 管理的程序

提示：linuxrc 開機選項的詳細資訊

如需更多用來啟動 Linux 系統的 linuxrc 開機選項資訊，請參閱 `/usr/share/doc/packages/linuxrc/linuxrc.html`。

4.5 監控安裝程序

有數種選項可以用來遠端監控安裝程式。如果在開機時已指定用於安裝的正確開機選項，將會使用 VNC 或 SSH 從遠端工作站來控制安裝和系統組態。

4.5.1 安裝 VNC

您可以使用 VNC 檢視器軟體，在幾乎任何作業系統上遠端控制 SUSE Linux Enterprise 的安裝。本節將介紹如何使用 VNC 檢視器應用程式或網頁瀏覽器來進行安裝。

準備安裝 VNC

準備安裝VNC時，您在安裝目標上必須執行的動作就是在第一次開始安裝時，提供適當的開機選項 (請參閱 第 4.4.3 節「使用自定開機選項」[71頁])。目標系統將會啟動到以文字為基礎的環境，接著，請等待 VNC 用戶端連接到安裝程式。

安裝程式將會宣告進行連接安裝時所需要的 IP 位址和顯示編號。如果您有實體存取目標系統的權限，系統開機後會隨即提供用於安裝的資訊。當 VNC 用戶端軟體提示輸入資料時，請輸入資料並提供您的 VNC 密碼。

因為安裝目標本身會透過 OpenSLP 進行宣告，所以您可以透過 SLP 瀏覽器擷取安裝目標的位址資訊，且不需實體連接到您網路設定提供的安裝本身。另外，所有機器都必須支援 OpenSLP。

1. 啟動 KDE 檔案和網頁瀏覽器 Konqueror。

2. 在位置列中輸入 `service://yast.installation.suse`。接著，Konqueror 畫面中會出現一個代表目標系統的圖示。按一下此圖示可啟動要執行安裝的 KDE VNC 檢視器。另外，請使用提供的 IP 位址執行您的 VNC 檢視器軟體，並在 IP 位置結尾加上 `:1`，以顯示安裝正在執行中。

連接到安裝程式

基本上，有兩種方法可以連接到 VNC 伺服器 (即本範例中的安裝目標)。您可以在任何作業系統上啟動獨立的 VNC 檢視器應用程式，或是使用已啟用 Java 的網頁瀏覽器進行連接。

您可以使用VNC，從任何其他作業系統 (包括其他 Linux 版本、Windows 或 Mac 作業系統) 控制 Linux 系統的安裝。

若是使用 Linux 機器，請確定已安裝 `tightvnc` 套件。若是使用 Windows 機器，請安裝此應用程式 (可在 TightVNC 首頁 http://www.tightvnc.com/download.html 取得) 的 Windows 連接埠。

若要連接到在目標機器上執行的安裝程式，請執行下列步驟：

1. 啟動 VNC 檢視器。
2. 輸入由 SLP 瀏覽器或安裝程式本身提供的安裝目標 IP 位址和顯示編號：

   ip_address:*display_number*

   接著會在您的桌面上開啟一個視窗，並顯示做為一般本地安裝的 YaST 畫面。

使用網頁瀏覽器連接到安裝程式，可讓您完全不需理會 VNC 軟體或舊版的作業系統。只要瀏覽器應用程式已啟用 Java 支援，您就可以使用任何瀏覽器 (Fireofx、Internet Explorer、Konqueror、Opera 等) 來執行 Linux 系統安裝。

若要進行 VNC 安裝，請依照下列步驟執行：

1. 啟動您要使用的網頁瀏覽器。
2. 在位址提示中輸入以下內容：

   http://*ip_address_of_target*:5801

3. 接著，系統會提示您輸入 VNC 密碼。然後，瀏覽器視窗會顯示做為一般本地安裝的 YaST 畫面。

4.5.2 安裝 SSH

您可以使用 SSH，透過任何 SSH 用戶端軟體遠端控制 Linux 機器的安裝。

準備安裝 SSH

除了安裝適當的軟體套件 (用於 Linux 的 OpenSSH 和用於 Windows 的 PuTTY) 外，您只需要傳送適當的開機選項即可開始安裝 SSH。如需詳細資料，請參閱第 4.4.3 節「使用自定開機選項」[71頁]。依預設，OpenSSH 會安裝在任何 SUSE Linux 作業系統上。

連接到安裝程式

1. 擷取安裝目標的 IP 位址。如果您有實際存取目標機器的權限，您就只要取用初始開機後由安裝常式在控制台提供的 IP 位址。或者，也可以採用在 DHCP 伺服器組態中指定給此特定主機的 IP 位址。

2. 在指令行，輸入以下指令：

   ```
   ssh -X root@ip_address_of_target
   ```

   將 *ip_address_of_target* 取代成安裝目標的實際 IP 位址。

3. 系統會提示輸入使用者名稱，接著請輸入 `root`。

4. 當提示輸入密碼時，請輸入使用 SSH 開機選項所設定的密碼。完成驗證後，會出現一個指令行，提示您輸入安裝目標。

5. 請輸入 `yast` 啟動安裝程式。出現一個視窗，顯示第 3 章「*使用 YaST安裝*」[17頁]所述的一般 YaST 畫面。

5 自動安裝

AutoYaST 可讓您同時在大量機器上安裝 SUSE® Linux Enterprise。AutoYaST 技術提供充足的彈性，可以針對異質硬體調整部署方式。本章說明如何準備簡易的自動安裝，並展示一個涉及不同硬體類型和安裝用途的複雜案例。

5.1 簡易大量安裝

重要：完全一樣的硬體

本案例假設您要將 SUSE Linux Enterprise 部署到硬體組態完全一樣的一組機器上。

若要為 AutoYaST 大量安裝做準備，請執行下列步驟：

1. 建立 AutoYaST 設定檔，以包含部署所需的安裝詳細資料，詳細步驟說明請參閱第 5.1.1 節「建立 AutoYaST 設定檔」[78頁]。
2. 決定 AutoYaST 設定檔的來源，以及要傳給安裝常式的參數，詳細步驟說明請參閱第 5.1.2 節「配送設定檔和決定 AutoYaST 參數」[79頁]。
3. 決定 SUSE Linux Enterprise 安裝資料的來源，詳細步驟說明請參閱第 5.1.3 節「提供安裝資料」[82頁]。
4. 決定和設定自動安裝的開機方式，詳細步驟說明請參閱第 5.1.4 節「設定開機方式」[82頁]。

5 手動新增參數或建立 `info` 檔案，以傳送指令行給安裝常式，詳細步驟說明請參閱第 5.1.5 節「建立 `info` 檔案」[84頁]。

6 開始自動安裝程序，詳細步驟說明請參閱第 5.1.6 節「啟始和監控自動安裝」[87頁]。

5.1.1 建立 AutoYaST 設定檔

AutoYaST 設定檔會告訴 AutoYaST 要安裝什麼，及如何設定安裝系統，以便在最後獲得完全可用的系統。有多種不同方法可建立 AutoYaST 設定檔：

- 從參考機器將全新安裝複製到一組完全相同的機器
- 使用 AutoYaST GUI 建立和修改設定檔，以符合您的需要
- 使用 XML 編輯器從頭開始建立設定檔

若要複製全新參考安裝，請執行下列步驟：

1 執行正常安裝。

2 完成硬體組態並閱讀版本說明後，勾選「*複製此安裝供 AutoYaST 使用*」(如果尚未預設勾選這一項)，就會建立名稱為 `/root/autoinst.xml` 的現成設定檔，可用來建立這次安裝的複製檔案。

若要使用 AutoYaST GUI 根據現有系統組態建立設定檔，再依您的需要修改，請執行下列步驟：

1 以 `root` 身份啟動 YaST。

2 選取「*其他*」>「*自動安裝*」，以啟動 AutoYaST 的圖形前端工具。

3 選取「*工具*」>「*建立參考控制檔案*」，讓 AutoYaST 準備為目前的系統組態建立 AutoYaST 設定檔鏡像。

4 除了預設資源 (如開機載入程式、磁碟分割和軟體選擇) 以外，您也可以勾選「*建立參考控制檔案*」中的清單項目，以在設定檔中新增系統的其他各種不同面向。

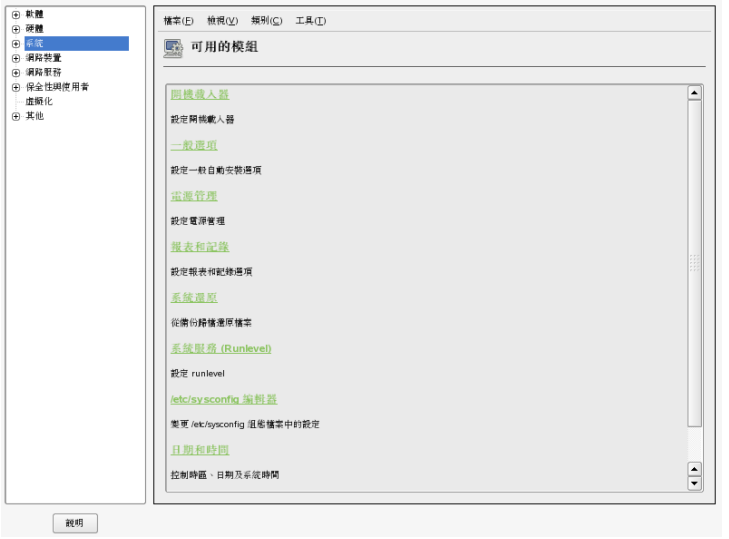
5 按一下「*建立*」，讓 YaST 收集所有系統資訊，並寫入新的設定檔。

6 若要繼續進行，請選擇下列其中之一：

- 如果設定檔已完成，而且符合您的需要，請選取「*檔案*」>「*另存新檔*」，並輸入設定檔的檔案，如 `autoinst.xml`。
- 在左側的樹狀檢視中選取適當的組態面向(如「硬體/印表機」)，並按一下「*設定*」，以修改參考設定檔，就會啟動個別的 YaST 模組，但您的設定會寫入 AutoYaST 設定檔，而不是套用到您的系統。完成後，選取「*檔案*」>「*另存新檔*」，並為設定檔輸入合適的名稱。

7 選取「*檔案*」>「*結束*」，以離開 AutoYaST 模組。

圖形 5.1 *以 AutoYaST 前端編輯 AutoYaST 模組*



5.1.2 配送設定檔和決定 AutoYaST 參數

有多種不同方法可以配送 AutoYaST 設定檔。依用來配送設定檔資料的通訊協定而定，我們會使用不同的 AutoYaST 參數，讓用戶端上的安裝常式知道設定檔的位置在哪裡。設定檔的位置是利用開機提示或開機時載入的 `info` 檔案傳給安裝常式。可用的選項如下：

設定檔位置	參數	描述
檔案	`autoyast=file://`*`/path`*	使安裝常式在指定路徑中尋找控制檔案 (相對於來源根目錄—如果是在光碟的最上層目錄，則是 `file:///autoinst.xml`)。
設備	`autoyast=device://`*`/path`*	使安裝常式在儲存設備上尋找控制檔案。僅須設備名稱—`/dev/sda1` 是錯的，應該用 `sda1`。
磁片	`autoyast=floppy://`*`/path`*	使安裝常式在軟碟機的磁片上尋找控制檔案。如果要從光碟機開機，這個選項特別有用。 若無法從磁片擷取控制檔案，AutoYaST 會自動掃描您機器上連接的所有 USB 設備。
USB (Flash) 磁碟	`autoyast=usb://`*`/path`*	此選項會啟動搜尋，在連接的所有 USB 設備中尋找控制檔案。
NFS	`autoyast=nfs://`*`/server/path`*	讓安裝常式從 NFS 伺服器取得控制檔案。
HTTP	`autoyast=http://`*`/server/path`*	讓安裝常式從 HTTP 伺服器取得控制檔案。
HTTPS	`autoyast=https://`*`/server/path`*	讓安裝常式從 HTTPS 伺服器取得控制檔案。
TFTP	`autoyast=tftp://`*`/server/path`*	讓安裝常式從 TFTP 伺服器取得控制檔案。
FTP	`autoyast=ftp://`*`/server/path`*	讓安裝常式從 FTP 伺服器取得控制檔案。

以符合實際設定的值取代 *`server`* 和 *`path`* 預留位置。

AutoYaST 包含一項功能，可將若干設定檔繫結到用戶端的 MAC 位址。您不必改變 `autoyast=` 參數，就可以讓同一個設定使用不同設定檔安裝幾個不同例項。

若要使用這項功能，請繼續執行下列步驟：

1. 以用戶端的 MAC 位址做為檔名建立不同的設定檔，將它們放在存放您的 AutoYaST 設定檔的 HTTP 伺服器上。
2. 建立 `autoyast=` 參數時應省略包含檔名的真實路徑，例如：

   ```
   autoyast=http://192.0.2.91/
   ```

3. 啟動自動安裝。

YaST 會嘗試以下列方式決定設定檔的位置：

1. YaST 用它自己大寫的十六進位 IP 位址搜尋設定檔，例如 `192.0.2.91` 是 `C000025B`。
2. 如果找不到這個檔案，YaST 會移去一個十六進位數，然後再試一次。這個動作會重複八次，直到找到正確名稱的檔案為止。
3. 如果仍然找不到，它會嘗試以用戶端的 MAC 位址做為檔名來尋找檔案。用戶端範例的 MAC 位址是 `0080C8F6484C`。
4. 如果找不到以 MAC 位址命名的檔案，YaST 會搜尋名稱為 `default` (小寫字母) 的檔案。YaST 搜尋 AutoYaST 設定檔的位址順序範例如下：

   ```
   C000025B
   C000025
   C00002
   C0000
   C000
   C00
   C0
   C
   0080C8F6484C
   default
   ```

5.1.3 提供安裝資料

安裝資料可利用產品 CD 或 DVD 或使用網路安裝來源的方式提供。如果使用產品 CD 做為安裝來源，就必須能夠實際存取要安裝的用戶端，因為開機程序必須以手動方式啟始，而且必須更換 CD。

若要透過網路提供安裝來源，請依第 4.2.1 節「使用 YaST 設定安裝伺服器」[51頁]所述設定網路安裝伺服器 (HTTP、NFS、FTP)。使用 `info` 檔案可傳送伺服器的位置給安裝常式。

5.1.4 設定開機方式

用戶端有數種不同開機方法：

網路開機
: 對於標準遠端安裝，可以用網路喚醒功能和 PXE 啟始自動安裝、可以透過 TFTP 引進開機影像和控制檔案，而且安裝來源可以來自任何網路安裝伺服器。

可開機 CD-ROM
: 您可以使用原始的 SUSE Linux Enterprise 媒體啟動要自動安裝的系統，再從網路位置或磁片引進控制檔案。或者，您可以建立自定的 CD-ROM，來存放安裝來源和 AutoYaST 設定檔。

下列章節提供網路開機或從光碟機開機的基本程序大綱。

準備網路開機

使用網路喚醒功能、PXE 和 TFTP 進行網路開機的方式於第 4.1.3 節「透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能」[46頁]中說明。若要使該處介紹的安裝方式適用於自動安裝，請修改 PXE 功能的 Linux 組態檔案 (`/srv/tftp/pxelinux.cfg/default`)，以包含指到 AutoYaST 設定檔位置的 `autoyast` 參數。標準安裝的項目範例如下：

```
default linux

# default label linux
    kernel linux append initrd=initrd ramdisk_size=65536 insmod=e100 \
    install=http://192.168.0.22/install/suse-enterprise/
```

自動安裝的相同範例如下：

```
default linux

# default label linux
    kernel linux append initrd=initrd ramdisk_size=65536 insmod=e100 \
    install=http://192.168.0.22/install/suse-enterprise/ \
    autoyast=nfs://192.168.0.23/profiles/autoinst.xml
```

以您的設定中所用的資料取代 IP 位址和路徑範例。

準備從光碟機開機

在 AutoYaST 安裝中，有幾種不同方式可以從光碟機開機。請在下列方式之中做選擇：

從 SUSE Linux Enterprise 媒體開機，透過網路取得設定檔
: 如果完全以網路為基礎的方式不可行 (例如，硬體不支援 PXE)，而且您在大部分過程中可以實際存取要安裝的系統，請使用這種作法。

  您需要：

  - SUSE Linux Enterprise 媒體
  - 提供設定檔資料的網路伺服器 (詳細資料請參閱第 5.1.2 節「配送設定檔和決定 AutoYaST 參數」[79頁])
  - 包含 `info` 檔案的磁片，以告訴安裝常式要到哪裡找到設定檔

    或

    存取要安裝系統的開機提示，讓您手動輸入 `autoyast=` 參數

從 SUSE Linux Enterprise 媒體開機並進行安裝，從磁片取得設定檔
: 如果完全以網路為基礎的安裝方式不可行，請使用這種作法。這種作法必須實際存取要安裝的系統，以調整目標機器，或在開機提示時輸入設定檔的位置 (這是第二種情況)。不管是哪一種情況，您都必須根據安裝範圍更換媒體。

您需要：

- SUSE Linux Enterprise 媒體
- 存放設定檔和 `info` 檔案的磁片

  或

  存取目標的開機提示以輸入 `autoyast=` 參數

從自定媒體開機並進行安裝，從媒體取得設定檔
: 如果您只須安裝有限的軟體套件，而且目標數量相當少，理想的作法是建立您自己的自定 CD，來存放安裝資料和設定檔，尤其是在您的設定中無法使用網路時。

5.1.5 建立 info 檔案

位於目標的安裝常式必須知道 AutoYaST 結構的所有不同元件在哪裡，所以我們必須建立指令行，內含找到 AutoYaST 元件、安裝來源和控制安裝程序所需參數的所有參數。

若要這樣做，請在安裝的開機提示時手動傳送這些參數，或提供名稱為 `info` 的檔案 (安裝常式 (linuxrc) 會讀取這個檔案)。前者必須實際存取要安裝的任何用戶端，使這種作法不適合用於大型部署。後者可讓您在安裝前於預先準備的媒體上提供 `info` 檔案，並插入用戶端的磁碟機。或者，使用 PXE 開機，並依章節「準備網路開機」 [82頁]所述在 `pxelinux.cfg/default` 檔案中包含 linuxrc 參數。

下列是常用的 linuxrc 參數。如需詳細資訊，請參閱 `/usr/share/doc/packages/autoyast` 中的 AutoYaST 套件文件。

重要：分隔參數和值

於開機提示中傳送參數給 linuxrc 時，請使用 = 分隔參數和值。使用 `info` 檔案時，則用 : 分隔參數和值。

關鍵字	數值
netdevice	網路設定所要用的網路設備 (用於 BOOTP/DHCP 要求)。只有在有數個網路設備可用時才需要。
hostip	如果空白，用戶端會傳送 BOOTP 要求；否則，使用指定的資料設定用戶端。
netmask	網路遮罩。
gateway	閘道。
nameserver	名稱伺服器。
autoyast	自動安裝所用控制檔案的位置，例如 autoyast=http://192.168.2.1/profiles/。
install	安裝來源的位置，例如 install=nfs://192.168.2.1/CDs/。
vnc	如果設為 1，即允許 VNC 遠端控制的安裝。
vncpassword	VNC 的密碼。
usessh	如果設為 1，即允許 SSH 遠端控制的安裝。

如果您的自動安裝方式涉及透過 DCHP 和網路安裝來源來設定用戶端，而且您希望使用 VNC 監控安裝程序，您的 info 應該類似：

```
autoyast:profile_source install:install_source vnc:1 vncpassword:some_password
```

如果您習慣在安裝時使用靜態網路設定，您的 info 檔案應該類似：

```
autoyast:profile_source \
install:install_source \
hostip:some_ip \
netmask:some_netmask \
gateway:some_gateway
```

\ 表示是為了方便閱讀才多加了幾個分行符號。所有選項必須以一個連續的字串輸入。

`info` 資料可以用幾種不同方式提供給 linuxrc：

- 做為安裝時位於用戶端軟碟機的磁片根目錄中的一個檔案。
- 做為系統開機所用初始 RAM 磁碟機的根目錄中的一個檔案，由自定安裝媒體或透過 PXE 開機提供。
- 做為 AutoYaST 設定檔的一部分。在這種情況下，AutoYaST 檔案必須命名為 `info`，這樣 linuxrc 才能剖析它。這種作法的範例如下。

linuxrc 會在設定檔中尋找代表檔案開頭的字串 (`start_linuxrc_conf`)。如果找到，就從該字串開始剖析，並在找到 `end_linuxrc_conf` 字串時結束。儲存在設定檔中的選項如下：

```
....
  <install>
....
    <init>
      <info_file>
<![CDATA[
#
# Don't remove the following line:
# start_linuxrc_conf
#
install: nfs:server/path
vnc: 1
vncpassword: test
autoyast: file:///info

# end_linuxrc_conf
# Do not remove the above comment
#
]]>

      </info_file>
    </init>
......
  </install>
....
```

linuxrc 會載入包含開機參數的設定檔，而不是傳統的 `info` 檔案。`install:` 參數會指示安裝來源的位置。`vnc` 與 `vncpassword` 指示使用 VNC 監控安裝。`autoyast` 參數則告訴 linuxrc 將 `info` 視為 AutoYaST 設定檔。

5.1.6 啟始和監控自動安裝

提供上述所有基礎結構後 (設定檔、安裝來源和 `info` 檔案)，您可以繼續啟動自動安裝。依選擇的開機和監控程序方式而定，您可能必須與用戶端有實際的互動：

- 如果用戶端系統從任何實體媒體 (產品媒體或自定光碟) 開機，您必須將這些媒體插入用戶端的磁碟機。
- 如果用戶端不是透過網路喚醒功能開啟，您至少必須開啟用戶端機器。
- 如果您未選擇遠端控制的自動安裝，就會傳送 AutoYaST 的圖形回應給連接用戶端的監視器，或 (如果您使用無人操作的用戶端) 序列主控台。

若要啟用遠端控制的自動安裝，請依第 5.1.5 節「建立 `info` 檔案」[84頁]所述使用 VNC 或 SSH 參數，並依第 4.5 節「監控安裝程序」[73頁]所述從其他機器連接到用戶端。

5.2 以規則為基礎的自動安裝

下列章節介紹使用 AutoYaST 以規則為基礎的安裝的基本概念，並提供案例範例讓您能夠建立您自己的自定安裝設定。

5.2.1 瞭解以規則為基礎的自動安裝

以規則為基礎的 AutoYaST 安裝讓您能夠處理異質的硬體環境：

- 您的網站包含不同廠商的硬體嗎？
- 您的網站上的機器採用不同硬體組態嗎 (例如，使用不同設備，或配備不同記憶體和磁碟大小)？
- 您打算跨越網域進行安裝，而且必須加以區別嗎？

基本上，以規則為基礎的自動安裝會將數個設定檔合併成一個，來產生符合異質案例的自定設定檔。每個規則會描述設定中的一個特色 (例如磁碟大小)，並告訴 AutoYaST 當規則符合時要使用哪個設定檔。描述設定中不同特色的數個規則合併在一個 AutoYaST `rules.xml` 檔案中。然後 AutoYaST 會處理規則堆

疊，並將符合 AutoYaST 規則的不同設定檔合併成一個，來產生最後的設定檔。如需這個程序的示範，請參閱第 5.2.2 節「以規則為基礎的自動安裝案例範例」[89頁]。

在規劃和執行 SUSE Linux Enterprise 部署時，以規則為基礎的 AutoYaST 為您提供充足的彈性。您可以：

- 在 AutoYaST中建立符合任何預先定義系統屬性的規則
- 使用邏輯運算子將多個系統屬性 (如磁碟大小和核心結構) 合併成一個規則
- 執行外圍程序檔並將輸出傳到 AutoYaST 結構以建立自定規則。自定規則數不得超過 5 個。

注

如需有關 AutoYaST的規則建立和用法的更多資訊，請參閱 `/usr/share/doc/packages/autoyast2/html/index.html` 中套件文件的「*Rules and Classes*」一章。

若要準備進行以規則為基礎的 AutoYaST 大量安裝，請執行下列步驟：

1. 建立數個 AutoYaST 設定檔，以包含異質設定所需的安裝詳細資料，詳細步驟說明請參閱第 5.1.1 節「建立 AutoYaST 設定檔」[78頁]。
2. 定義符合硬體設定的系統屬性的規則，詳細步驟說明請參閱第 5.2.2 節「以規則為基礎的自動安裝案例範例」[89頁]。
3. 決定 AutoYaST 設定檔的來源，以及要傳給安裝常式的參數，詳細步驟說明請參閱第 5.1.2 節「配送設定檔和決定 AutoYaST 參數」[79頁]。
4. 決定 SUSE Linux Enterprise 安裝資料的來源，詳細步驟說明請參閱第 5.1.3 節「提供安裝資料」[82頁]。
5. 手動新增參數或建立 `info` 檔案，以傳送指令行給安裝常式，詳細步驟說明請參閱第 5.1.5 節「建立 `info` 檔案」[84頁]。
6. 決定和設定自動安裝的開機方式，詳細步驟說明請參閱第 5.1.4 節「設定開機方式」[82頁]。

7 開始自動安裝程序，詳細步驟說明請參閱第 5.1.6 節「啟始和監控自動安裝」[87頁]。

5.2.2 以規則為基礎的自動安裝案例範例

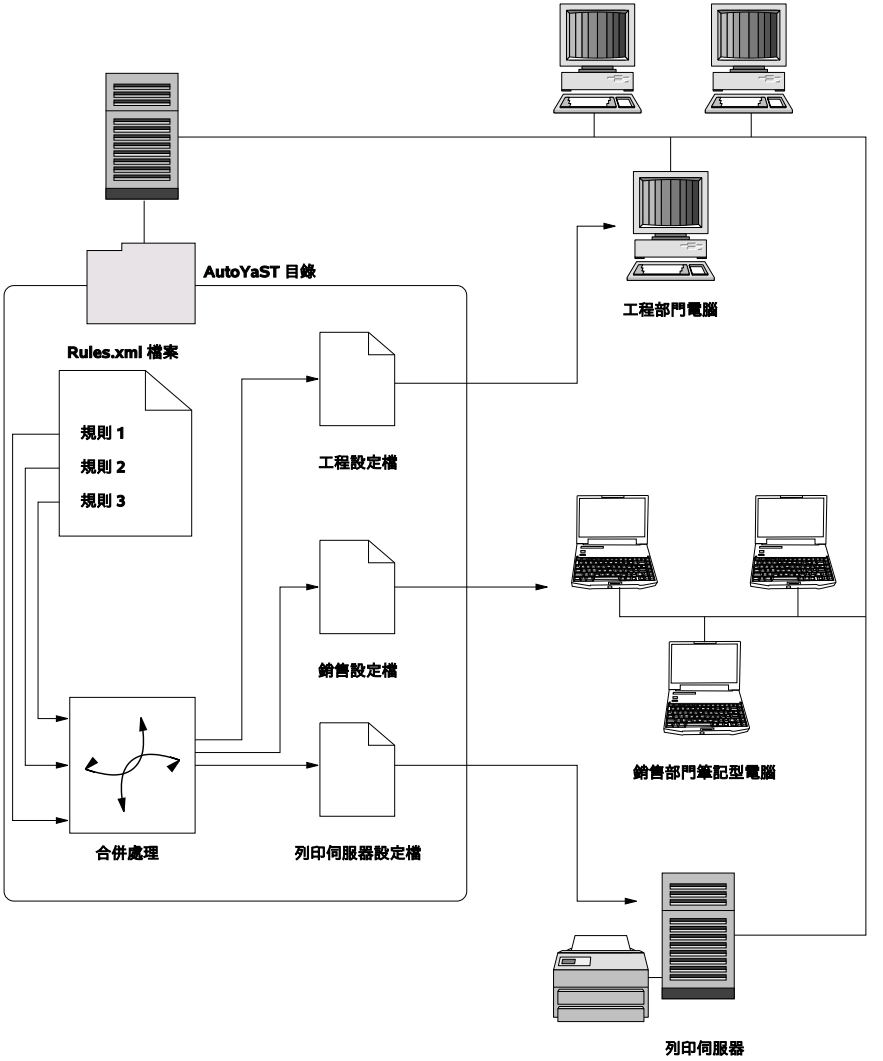
若要對規則的建立方式有基本的瞭解，請參閱圖形 5.2「AutoYaST 規則」[90頁]中描述的下列範例。執行 AutoYaST 會安裝下列設定：

列印伺服器
: 這台機器不需要桌面環境，只需要最基本的安裝，和有限的軟體套件。

工程部門的工作站
: 這些機器需要桌面環境，和大量開發工具。

銷售部門的筆記型電腦
: 這些機器需要桌面環境，和有限的專用應用程式，如辦公室和行事曆軟體。

圖形 5.2　AutoYaST 規則



在第一個步驟中，使用第 5.1.1 節「建立 AutoYaST 設定檔」 [78頁]中說明的一種方法針對每一種案例建立設定檔。在此範例中，您會建立 `print.xml`、`engineering.xml` 和 `sales.xml`。

在第二個步驟中，建立規則來區分三種硬體類型，並告訴 AutoYaST 要使用哪個設定檔。使用類似下列演算法來設定規則：

1. 機器有「*192.168.27.11*」這個 IP 嗎？有的話，將它設為列印伺服器。
2. 機器有 PCMCIA 硬體而且使用 Intel 晶片組嗎？有的話，將它視為 Intel 筆記型電腦，並安裝銷售部門軟體選擇。
3. 如果上述條件都不成立，則將該機器視為開發工作站，並據此進行安裝。

這可以大致轉換成包含下列內容的 `rules.xml` 檔案：

```
<?xml version="1.0"?>
<!DOCTYPE autoinstall SYSTEM "/usr/share/autoinstall/dtd/rules.dtd">
<autoinstall xmlns="http://www.suse.com/1.0/yast2ns"
xmlns:config="http://www.suse.com/1.0/configns">
  <rules config:type="list">
    <rule>
       <hostaddress>
     <match>192.168.27.11</match>
            <match_type>exact</match_type>
       </hostaddress>
       <result>
            <profile>print.xml</profile>
            <continue config:type="boolean">false</continue>
        </result>
    </rule>
    <rule>
       <haspcmcia>
            <match>1</match>
            <match_type>exact</match_type>
       </haspcmcia>
       <custom1>
           <script>
if grep -i intel /proc/cpuinfo > /dev/null; then
echo -n "intel"
else
echo -n "non_intel"
fi;
           </script>
           <match>*</match>
           <match_type>exact</match_type>
       </custom1>
       <result>
            <profile>sales.xml</profile>
            <continue config:type="boolean">false</continue>
        </result>
        <operator>and</operator>
    </rule>
```

```
    <rule>
        <haspcmcia>
            <match>0</match>
            <match_type>exact</match_type>
        </haspcmcia>
  <result>
            <profile>engineering.xml</profile>
            <continue config:type="boolean">false</continue>
        </result>
    </rule>
  </rules>
</autoinstall>
```

配送規則檔案時，必須確定 `rules` 目錄位於 `autoyast=`*`protocol`*`:`*`serverip`*`/profiles/` URL 中指定的 `profiles` 目錄下。AutoYaST 會尋找包含 `rules.xml` 檔案的 `rules` 子目錄，然後載入並合併規則檔案中指定的設定檔。

其餘自動安裝程序依正常方式執行。

5.3 如需更多資訊

如需有關 AutoYaST 技術的詳細資訊，請參閱隨軟體一起安裝的文件。檔案位於 `/usr/share/doc/packages/autoyast2`。如需本文件的最新版本，請造訪 http://www.suse.de/~ug/autoyast_doc/index.html。

6 部署自定的預先安裝

將 SUSE Linux Enterprise 的自定預先安裝轉出到大量的相同機器上，讓您不需要個別安裝，並為使用者提供標準畫的安裝體驗。透過 YaST 的第一次開機，建立自定化預先安裝影像，並決定需要與最終使用者互動之最終個人化步驟的工作流程。這與 AutoYaST 不同，後者允許完全自動的安裝；如需詳細資訊，請參閱第 5 章「*自動安裝*」[77頁]。

建立自定安裝、轉出至您的硬體並個人化最終產品，包括下列步驟：

1. 備妥要將磁碟複製到用戶端機器的主要機器若需更多資訊，請參閱第 6.1 節「備妥主要機器」[94頁]。
2. 自定化第一次開機工作流程。若需更多資訊，請參閱第 6.2 節「自定第一次開機安裝」[94頁]。
3. 複製主要機器的磁碟，並將影像轉出到用戶端磁碟。若需更多資訊，請參閱第 6.3 節「複製主要安裝」[102頁]。
4. 讓使用者個人化 SUSE Linux Enterprise 例項。若需更多資訊，請參閱第 6.4 節「個人化安裝」[102頁]。

6.1 備妥主要機器

若要備妥主要機器進行第一次開機工作流程，請遵循下列步驟：

1. 將安裝媒體插入主要機器中。
2. 開機。
3. 執行包括所有必要組態步驟的一般安裝，並等待安裝的機器開機。同時安裝 `yast2-firstboot` 套件。
4. 若要為最終使用者定義您自己的 YaST 組態步驟工作流程，或將您自己的 YaST 模組新增到此工作流程，請繼續第 6.2 節「自定第一次開機安裝」[94頁]。否則請直接執行步驟 5 [94頁]。
5. 以 `root` 身份啟動第一次開機。

   5a 建立空白檔案 `/etc/reconfig_system` 以觸發第一次開機的執行。成功完成第一次開機組態設定後，將會刪除此檔案。使用下列指令建立此檔案：

   ```
   touch /etc/reconfig_system
   ```

   5b 透過 YaST Runlevel 編輯器啟用第一次開機服務。

6. 繼續進行第 6.3 節「複製主要安裝」[102頁]。

6.2 自定第一次開機安裝

自定第一次開機安裝可以包含許多不同元件。自定是選擇性的。若您不希望進行變更，第一次開機會使用預設設定進行安裝。可用的選項如下：

- 如第 6.2.1 節「自定 YaST 訊息」[95頁]所述，自定給使用者的訊息。
- 如第 6.2.2 節「自定授權條例」[96頁]所述，自定授權與授權條例。
- 如第 6.2.3 節「自定版本說明」[96頁]所述，自定要顯示的版本說明。

- 如第 6.2.4 節「自定工作流程」[97頁]所述，自定安裝所包含的元件數目與順序。
- 如第 6.2.5 節「設定其他程序檔」[102頁]所述，設定其他選用的程序檔。

若要自定這些元件，請調整下列組態檔：

`/etc/sysconfig/firstboot`
設定第一次開機的各個層面，如版本說明、程序檔與授權條例。

`/etc/YaST2/firstboot.xml`
啟用或停用元件，或新增自定元件，設定安裝工作流程。

6.2.1 自定 YaST 訊息

依照預設，SUSE Linux Enterprise 的安裝包含數種已當地語系化、以及顯示於各安裝過程特定階段的預設訊息。包括歡迎訊息、授權訊息，以及安裝完畢的恭喜訊息。您可以您自己的版本取代任一個訊息，並將當地語系化版本包含於安裝中。若要包含您個人的歡迎訊息，請如下執行：

1. 以 `root` 身份登入。
2. 開啟 `/etc/sysconfig/firstboot` 組態檔並套用下列變更：
   - **2a** 將 `FIRSTBOOT_WELCOME_DIR` 設定為要在其中儲存含有歡迎訊息和當地語系化版本之檔案的目錄路徑，例如：

     ```
     FIRSTBOOT_WELCOME_DIR="/usr/share/firstboot/"
     ```

   - **2b** 如果歡迎訊息使用的檔名不是 `welcome.txt` 和 `welcome_`*`locale`*`.txt`(其中 *`locale`* 與 ISO 639 語言代碼相符，例如「cs」、「de」)，請在 `FIRSTBOOT_WELCOME_PATTERNS` 中指定檔名模式。例如：

     ```
     FIRSTBOOT_WELCOME_PATTERNS="mywelcome.txt"
     ```

     若未設定的話，會假設使用預設值 `welcome.txt`。
3. 建立歡迎檔案與當地語系化版本，並將這些檔案放到 `/etc/sysconfig/firstboot` 組態檔中指定的目錄。

以類似方式設定自定授權與結束訊息。變數為 FIRSTBOOT_LICENSE_DIR 與 FIRSTBOOT_FINISH_FILE。

6.2.2 自定授權條例

您可自定當使用者不接受授權合約時，安裝系統的反應。若使用者不接受授權合約，系統可有三種不同方式的反應：

中斷
　第一次開機安裝中止，整個系統關閉。此為預設值。

繼續
　第一次開機繼續安裝。

中止
　第一次開機安裝中止，但系統嘗試開機。

進行選擇，並將 LICENSE_REFUSAL_ACTION 設定為適當值。

6.2.3 自定版本說明

根據您是否變更您以第一次開機部署的 SUSE Linux Enterprise 例項而定，您可能需要教育您的使用者，以使用此新作業系統的重要功能。標準安裝使用者版本說明，會在安裝完成步驟之一顯示，將重要訊息提供給使用者。若要在第一次開機安裝顯示您修改過的版本說明，請如下操作：

1. 建立您自己的版本說明檔案。依照 /usr/share/doc/release-notes 中的範例檔案使用 RTF 格式，並將結果儲存為 RELEASE-NOTES.en.rtf (對於英文)。

2. 將選擇性的當地語系化版本儲存於原始版本旁，並將檔名的 en 部份取代為實際的 ISO 639 語言代碼，如德文就用 de。

3. 從 /etc/sysconfig/firstboot 開啟第一次開機 組態檔，並將 FIRSTBOOT_RELEASE_NOTES_PATH 設定為儲存版本說明檔案的實際目錄。

6.2.4 自定工作流程

依照預設，標準的第一次開機工作流程包含下列元件：

- 語言選擇
- 歡迎
- 授權書
- 主機名稱
- 網路
- 時間和日期
- 桌面
- root 密碼
- 使用者驗證方式
- 使用者管理
- 硬體組態
- 完成安裝

這個第一次開機安裝工作流程的標準配置不是強制性的。您可啟用或停用特定元件，或將您自己的模組插入工作流程中。若要修改第一次開機工作流程，請手動編輯第一次開機組態檔 `/etc/YaST2/firstboot.xml`。此 XML 檔案為標準 `control.xml` 檔案的子集，該檔案是 YaST 用來控制安裝工作流程的。

工作流程綜覽提供您足夠的背景，以修改第一次開機安裝工作流程。您可在其中看到第一次開機組態檔的基本語法，以及關鍵元件是如何設定的。

範例 6.1　設定提議畫面

```
…
<proposals config:type="list">❶
    <proposal>❷
        <name>firstboot_hardware</name>❸
        <mode>installation</mode>❹
        <stage>firstboot</stage>❺
        <label>Hardware Configuration</label>❻
        <proposal_modules config:type="list">❼
            <proposal_module>printer</proposal_module>❽
        </proposal_modules>
    </proposal>
    <proposal>
    …
    </proposal>
</proposals>
```

❶　應為第一次開機工作流程的所有提議之容器。

❷　個別提議之容器。

❸　提議的內部名稱。

❹　此提議的模式。在此不進行任何變更。對於第一次開機安裝，這裡一定要設為 `installation`。

❺　此提議所呼叫的安裝程序階段。在此不進行任何變更。對於第一次開機安裝，這裡一定要設為 `firstboot`。

❻　要顯示在提議上的標籤。

❼　身為提議畫面一部分的所有模組之容器。

❽　一或多模組為提議畫面的一部分。

下一段的第一次開機組態檔包含工作流程定義。應為第一次開機安裝工作流程一部分的所有模組，都應該列於此。

範例 6.2 *設定工作流程部份*

```
<workflows  config:type="list">
    <workflow>
        <defaults>
            <enable_back>yes</enable_back>
            <enable_next>yes</enable_next>
            <archs>all</archs>
        </defaults>
        <stage>firstboot</stage>
        <label>Configuration</label>
        <mode>installation</mode>
        … <!-- list of modules  -->
        </modules>
    </workflow>
</workflows>
…
```

工作流程部份的整體結構與提議部份非常相似。容器會包含工作流程元件，而工作流程元件均含有階段、標籤與模式資訊，如同範例 6.1「設定提議畫面」[98頁]介紹的提議部份。最大的差別在於 `defaults` 部份，其中包含工作流程元件的基本設計資訊：

`enable_back`
: 所有對話方塊都包含包含「*上一步*」。

`enable_next`
: 所有對話方塊都包含「*下一步*」。

`archs`
: 指定會使用此工作流程的硬體結構。

範例 6.3 *設定工作流程元件清單*

```
<modules  config:type="list">❶
    <module>❷
        <label>Language</label>❸
        <enabled config:type="boolean">false</enabled>❹
        <name>firstboot_language</name>❺
    </module>
<modules>
```

❶ 工作流程所有元件的容器。

❷ 模組定義。

❸ 模組顯示的標籤。

❹ 在此工作流程中啟用或停用此元件的開關。

❺ 模組名稱。模組本身必須位於 /usr/share/YaST2/clients 下，且檔案名稱結尾必須為 .ycp。

若要變更第一次開機安裝過程中的提議畫面數目與順序，請如下操作：

1 開啟位於 /etc/YaST2/firstboot.xml 的第一次開機組態檔。

2 刪除或新增提議畫面，或變更現有畫面的順序：

- 若要刪除整個提議，請從 proposal 部份移除 proposals 元素，包含其所有子元素，並從工作流程中個別移除模組元素 (與子元素)。
- 若要新增提議，請建立新 proposals 元素，並填妥所有必須的子元素。確認提議以 YaST 模組形式，存在於 /usr/share/YaST2/clients 中。
- 若要變更提議順序，請在工作流程中移動包含提議畫面的個別模組元件。請注意，其他安裝步驟可能對特定提議順序或工作流程元件具有相依性。

3 套用您的變更並關閉組態檔。

您一律可在預設值不符合您需求時，變更組態步驟的工作流程。在工作流程中啟用或停用特定模組，或新增您自定的模組。

若要切換第一次開機工作流程中的模組狀態，請如下操作：

1 開啟 /etc/YaST2/firstboot.xml 組態檔。

2 將 enabled 元件的值從 true 變更至 false，以停用模組，或從 false 變更為 true，以再次啟用。

```
<module>
    <label>Time and Date</label>
    <enabled config:type="boolean">true</enabled>
```

```
    <name>firstboot_timezone</name>
</module>
```

3 套用您的變更並關閉組態檔。

若要將自定模組新增到工作流程，請如下操作：

1 建立您個人的 YaST 模組，並將模組檔案 *module_name*.ycp 儲存於 /usr/share/YaST2/clients。

2 開啟 /etc/YaST2/firstboot.xml 組態檔。

3 決定新模組要執行於工作流程的哪個點。這麼做的時候，請確認您已考量工作流程中其他步驟可能的相依性。

4 在 modules 容器中建立新的 module 元素，並新增適當的子元素：

```
<modules config:type="list">
     …
     <module>
        <label>my_module</label>
        <enabled config:type="boolean">true</enabled>
        <name>filename_my_module</name>
     </module>
</modules>
```

4a 在 label 元素中輸入您模組要顯示的標籤。

4b 確認 enabled 已設為 true，讓您的模組包含於工作流程中。

4c 在 name 元素中輸入您模組的檔案名稱。省略完整路徑與 .ycp 字尾。

5 套用您的設定並關閉組態檔。

提示：如需更多資訊

若需有關 YaST 開發的詳細資訊，請參閱 http://developer.novell.com/wiki/index.php/YaST。

6.2.5 設定其他程序檔

您可設定第一次開機在第一次開機工作流程完成後，執行其他程序檔。若要將其他程序檔新增至第一次開機序列，請如下操作：

1. 開啟 `/etc/sysconfig/firstboot` 組態檔，並確認 `SCRIPT_DIR` 指定的路徑正確。預設值為 `/usr/share/firstboot/scripts`。
2. 建立您的外圍程序檔、儲存於指定目錄中，並套用適當的檔案許可權。

6.3 複製主要安裝

複製使用您可用的所有影像機制的主要機器磁碟，並將這些影像轉出到目標機器。

6.4 個人化安裝

複製的磁碟影像開機之後，第一次開機就會如同第 6.2.4 節「自定工作流程」[97頁]，一模一樣的啟動安裝程序。僅會啟動包含於第一次開機工作流程組態中的元件。而略過其他安裝步驟。使用者可調整語言、鍵盤、網路與密碼設定，以個人化工做站。程序完成之後，第一次開機所安裝的系統行為，會與其他 SUSE Linux Enterprise 例項完全相同。

7 進階磁碟安裝

複雜的系統組態必須在特定磁碟上進行安裝。所有一般的磁碟分割任務都可以藉由 YaST 完成。若要以區塊設備取得永久設備命名，請使用以下區塊設備 `/dev/disk/by-id/`。LVM (Logical Volume Management，邏輯磁碟區管理) 是一個磁碟分割結構，這項設計比標準安裝中使用的實體分割方式更為靈活。其快照功能可讓您輕鬆建立資料備份。另外，獨立磁碟容錯陣列 (RAID) 可提高資料的完整性、效能和容錯。SUSE® Linux Enterprise Server 也支援多路徑的 I/O。如需詳細資訊，請參閱《*儲存管理指南*》中有關多路徑 I/O 的章節。此外，從 SUSE Linux Enterprise 10 起還新增了使用 iSCSI 做為網路磁碟的選項。若要深入瞭解 iSCSI，請參閱第 12 章「*IP 網路 — iSCSI 上的大型存放設備*」[247頁]。

7.1 LVM 組態

本小節簡短地說明在 LVM 背後的原則，以及它在許多狀況下好用的基本功能。在第 7.1.2 節「使用 YaST 設定 LVM 組態」[105頁]中，學習如何使用 YaST 設定 LVM。

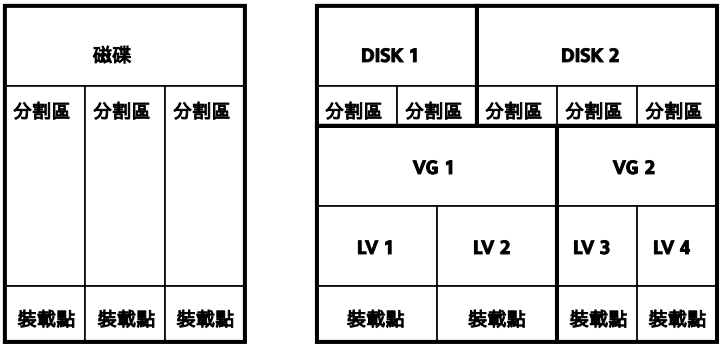
警告

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。

7.1.1 邏輯磁碟區管理

邏輯磁碟區管理 (Logical Volume Manager，LVM) 可以在數個檔案系統上彈性地散佈硬碟空間。在安裝過程中的啟始分割已經完成後，有時需要變更硬碟空間的分割，因此開發此工具。因為要修改執行系統上的分割區很困難，LVM 提供記憶體空間的虛擬集區 (磁碟區群組，簡稱 VG)。如有必要，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，因此數個磁碟或是磁碟的某些部份可能會構成單一的 VG。LVM 這種方法提供從實體磁碟空間擷取的方法，允許使用比實體重新分割更為簡單和安全的方式來變更分割。您可以在章節「分割區類型」[143頁]與第 8.5.7 節「使用 YaST 磁碟分割程式」[141頁]中找到有關實體分割的背景資訊。

圖形 7.1　實體分割與 LVM



圖形 7.1「實體分割與 LVM」[104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)，每一個都會指定定點 (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裝載點來使用。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。
- 可以啟用「等量分割模式」，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。如果您的資料會一直累積，如資料庫、音樂歸檔或使用者目錄等，LVM就是適合您的工具。這樣能夠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的說明及詳細資訊，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。

除了 LVM 2，您還可以使用 EVMS (企業磁碟區管理系統，Enterprise Volume Management System)，它提供統一的邏輯磁碟區和RAID磁碟區介面。與LVM 2類似，EVMS 利用整合於核心 2.6 中的設備映射程式。

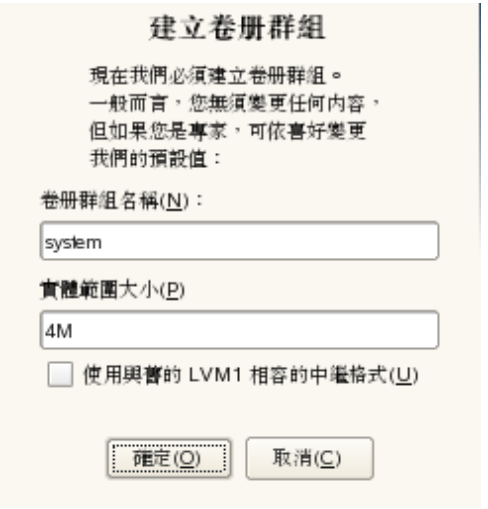
7.1.2 使用 YaST 設定 LVM 組態

YaST LVM 組態可以從 YaST 專家分割程式完成 (請參閱第 8.5.7 節「使用 YaST 磁碟分割程式」[141頁])。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。接著，先按一下「*建立*」>「*不要格式化*」，以建立 LVM 分割區，再選取「*0x8E Linux LVM*」做為分割區的識別碼。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。

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 7.2「建立磁碟區群組」 [106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® 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 7.2　建立磁碟區群組



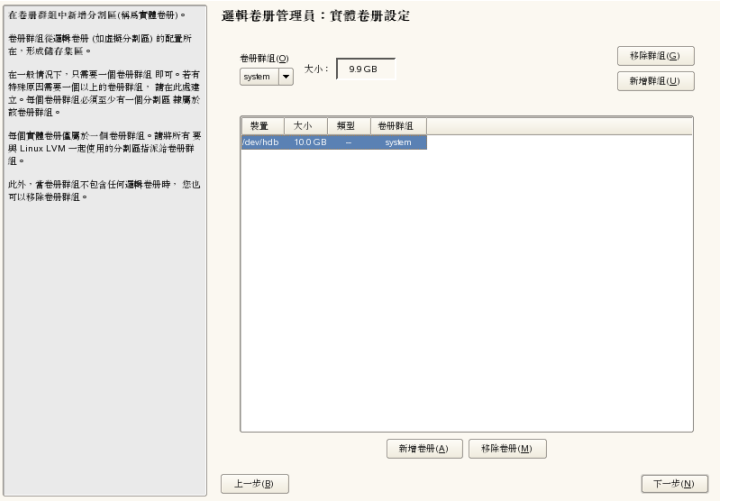
設定實體磁碟區

一旦建立磁碟區群組，以下對話方塊就會列出具有「Linux LVM」或「Linux native」類型的所有分割區。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 (PV)。

圖形 7.3　實體磁碟區設定

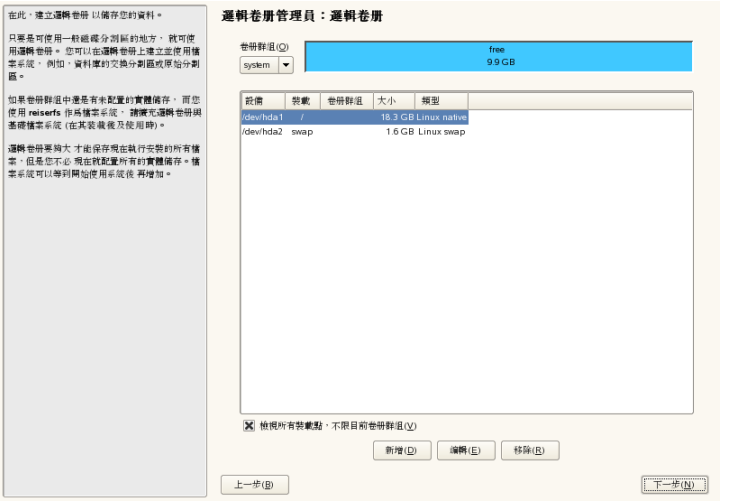


若要新增之前未指定的分割區給選取的磁碟區群組，請先按一下分割區，再按「*新增磁碟區*」。此時，磁碟區群組的名稱，會輸入到選取分割區的旁邊。指定為LVM保留的所有分割區給磁碟區群組。否則，仍然不會使用分割區上的空間。結束此對話方塊前，每個磁碟區群組必須指定至少一個實體磁碟區。在指定所有的實體磁碟區後，按一下「*下一步*」以繼續邏輯磁碟區的組態。

設定邏輯磁碟區

在磁碟區群組已經使用實體磁碟區填滿後，在下一個對話方塊中定義作業系統應該使用的邏輯磁碟區。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。指定裝載點的所有標準 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。「*視需要*」新增「、」編輯「*以及*」移除邏輯磁碟區，直到在磁碟區群組中的所有空間都使用完畢。至少指定一個邏輯磁碟區給每個磁碟區群組。

圖形 7.4　邏輯磁碟區管理



若要建立新的邏輯磁碟區，請按一下「*新增*」，然後填寫開啟的快顯視窗。可以輸入磁碟分割、大小、檔案系統以及裝載點。一般而言，如 reiserfs 或 ext2 之類的檔案系統，是建立於邏輯磁碟區上，接著再指定裝載點。儲存於此邏輯磁碟區上的檔案，可以在已安裝系統的此裝載點上找到。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 (等量分割)。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：等量磁區

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 7.5　建立邏輯磁碟區

如果已經在系統上設定 LVM，現在就可以輸入現有的邏輯磁碟區。在繼續前，請指定適當的裝載點到這些邏輯磁碟區。使用「*下一步*」，返回「YaST 專家分割程式」，然後在那完成您的工作。

直接 LVM 管理

如果您已設定 LVM，而且只想變更某些項目，可以使用另一種方式來完成。在「YaST 控制中心」選取「*系統*」>「*LVM*」。基本上此對話方塊允許如上方所描述的相同動作，除了實體分割以外。它以兩個清單顯示現有的實體磁碟區及邏輯磁碟區，而且您可以使用已經描述的方法來管理 LVM 系統。

7.1.3 以 EVMS 進行儲存管理

「企業磁碟區管理系統 2」(EVMS2) 是一個功能強大的可擴充磁碟區管理員，內建叢集感知。其外掛程式結構可讓外掛程式新增任何分割區類型的支援與知識功能。由於有叢集感知，因此 EVMS2 保證所管理的裝置在叢集中的每個節點都有獨一的名稱，以易於管理。

EVMS2 提供統一的介面 (`evmsgui` 與指令行)，讓您管理下列儲存資源：

- 本地媒體與 SAN 式媒體上的實體磁碟與邏輯裝置，包括 iSCSI
- 提供高度可用性的軟體 RAID 0、1、4 和 5
- 用於容錯的叢集感知多路徑 I/O
- 叢集儲存物件與叢集區段管理 (CSM) 外掛程式
- 具備 EVMS2 檔案系統介面模組 (FSIM) 的所有檔案系統磁區
- 磁區快照

在 SUSE Linux Enterprise Server 10中，有下列新功能：

- EVMS2 與 CLVM2 (Cluster Linux Volume Manager 2) 在核心中使用相同的多磁碟 (MD) 驅動程式與裝置對應程式 (DM) 驅動程式。
- Heartbeat 2 Cluster Manager 和 Oracle Cluster File System 2 中可使用檔案系統外掛程式。

EVMS 裝置

「EVMS 管理公用程式」區分五種層級的裝置：

磁碟機
: 這是最低層級的設備。可以實體磁碟方式存取的所有裝置都會被視為磁碟。

區段
: 區段包含分割區與磁碟上的其他記憶體區域，如主要開機記錄 (MBR)。

容器
: 這是 LVM 中磁區群組的對應項目。

區域
: 可用裝置會在此被群組至 LVM2 和 RAID。

磁碟區
: 所有裝置，無論是否具備真實分割區、邏輯磁區，或對應裝載點是否可使用 RAID 裝置的。

若您選擇使用 EVMS，您必須將設備名稱取代為 EVMS 設備名稱。簡單分割區位於 `/dev/evms/`，邏輯磁碟區位於 `/dev/evms/lvm/`，而 RAID 設備位於 `/dev/evms/md`。若要在開機時啟用 EVMS，請在 YaST runlevel 編輯器中將 `boot.evms` 新增至開機程序檔。並請參閱 第 20.2.3 節「使用 YaST 設定系統服務 (Runlevel)」 [367頁]。

如需更多資訊

如需使用 EVMS 管理儲存資源的相關資訊，請在安裝 `sles-stor_evms_en` 套件後參閱 `/usr/share/doc/manual/sles-stor_evms_en` 中的《*儲存管理指南*》。有關 EVMS 的更多通用資訊，也可參閱 SourceForge* 上代管之 EVMS 專案 [`http://evms.sourceforge.net/`]中的 EVMS 使用者指南 [`http://evms.sourceforge.net/users_guide/`]。

7.2 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。

7.2.1 RAID 層級

SUSE® Linux Enterprise 提供結合數個硬碟至一個軟體 RAID 系統的選項，結合 YaST 的使用，對硬體 RAID 是非常好的替代方案。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*。

常用 RAID 層級為：

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。

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*。如果其中一個硬碟損毀，另一個硬碟上有其內容的複本。只有其中一個會受到損害，但不會危害到資料。然而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較差 (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「層級 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。

RAID 2 與 RAID 3

這些都不是一般的 RAID 實作。「層級 2」在是位元層級等量分割資料，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等量分割，但是無法同時服務多個要求。這兩個層級都很少使用。

RAID 4

「層級 4」提供與「層級 0」相同的區塊層級等量分割，並且結合專用的同位磁碟。在資料磁碟失敗時，會使用同位資料以建立替代的磁碟。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」。

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。

其他的 RAID 層級

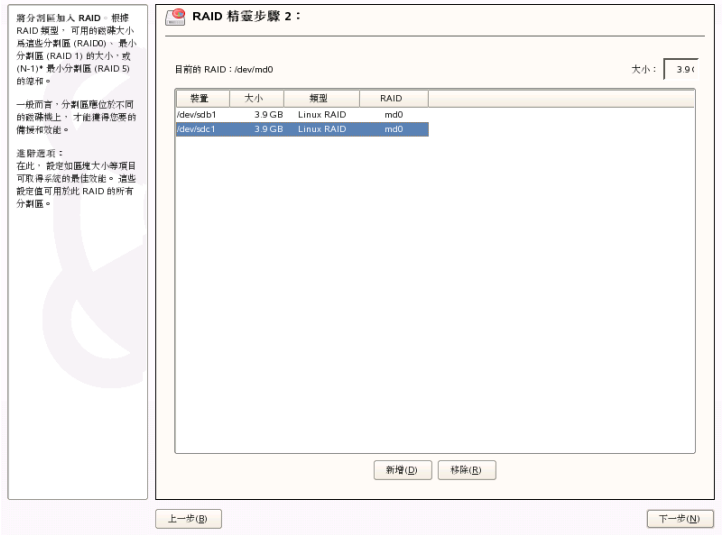
已經開發一些其他的 RAID 層級 (RAIDn、RAID 10、RAID 0+1、RAID 30、RAID 50 等等)，有些是硬體廠商所建立的專用實作。這些層級並不是很普遍，因此在這裏並不做說明。

7.2.2 使用 YaST 進行軟體 RAID 組態

YaST 軟體 RAID 組態可以從 YaST 進階分割程式完成 (請參閱 第 8.5.7 節「使用 YaST 磁碟分割程式」[141頁])。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。接著，先按一下「*建立*」>「*不要格式化*」，以建立 RAID 分割區，再選取「 *0xFD Linux RAID*」做為分割區的識別碼。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。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。

在下一個對話方塊中，在 RAID 層級 0、1 及 5 之間選擇 (請參閱第 7.2.1 節「RAID 層級」[111頁]以取得詳細資訊)。按一下「*下一步*」後，下列對話方塊會列出具有「Linux RAID」或「Linux native」類型的所有分割區 (請參閱圖形 7.6「RAID 分割區」[113頁])。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 7.6　RAID 分割區



若要新增之前未指定的分割區給選取的 RAID 磁碟區，請先按一下分割區，再按「*新增*」。此時，RAID 設備的名稱，會輸入到選取分割區的旁邊。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 7.7「檔案系統設定」[114頁]。)

圖形 7.7　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 磁碟區的裝載點。在按一下「*完成*」以完成組態之後，請參閱 `/dev/md0` 設備及其他在專家分割程式中以 *RAID* 表示的設備。

7.2.3 疑難排解

檢查檔案 `/proc/mdstats` 以找出 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。

7.2.4 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：

- http://www.novell.com/documentation/sles10/stor_evms/data/bookinfo.html
- /usr/share/doc/packages/mdadm/Software-RAID.HOWTO.html
- http://en.tldp.org/HOWTO/Software-RAID-HOWTO.html

也有 Linux RAID 郵寄清單可供參考，如 http://marc.theaimsgroup.com/?l=linux-raid&r=1&w=2。

8 使用 YaST 的系統組態

在 SUSE Linux Enterprise 中，YaST 可處理系統的安裝與組態。本章說明系統元件(硬體)的組態、網路存取以及安全性設定與使用者管理。如需文字模式 YaST 介面的簡介，請參閱第 8.12 節「文字模式的 YaST」[169頁]。如需有關手動設定系統組態的說明，請參閱第 20.3 節「透過 /etc/sysconfig 設定系統」[368頁]。

使用各種 YaST 模組來設定包含 YaST 的系統。視硬體平台與安裝的軟體而定，共有三種方式可在安裝的系統中存取 YaST。

在 KDE 或 GNOME 中，從主功能表啟動「YaST 控制中心」。YaST 啟動之前，會提示您輸入 `root` 密碼，因為 YaST 需具備系統管理員權限才能變更系統檔案。

若要從指令行啟動 YaST，請輸入指令 `su` (以變更為使用者 `root`) 與 `yast2`。若要啟動文字版本，請輸入 `yast` 而不是 `yast2`。您也可以使用指令 `yast` 從其中一個虛擬主控台來啟動程式。

對於無法支援自有顯示設備的硬體平台，以及要在其他主機進行遠端管理，就要從遠端執行 YaST。首先，在要顯示 YaST 的主機上開啟主控台，然後輸入指令 `ssh -x root@<system-to-configure>` 以 `root` 登入系統來進行設定，並將 X 伺服器輸出重新導向至您的終端機。在成功登入 SSH 之後，輸入 `yast2` 以圖形模式啟動 YaST。

若要從另一個系統以文字模式啟動 YaST，請使用 `ssh root@<system-to-configure>` 來開啟連線。然後利用 `yast` 來啟動 YaST。

如果要節省時間，您可以直接啟動個別的 YaST 模組。若要啟動模組，請輸入 `yast2` *`module_name`*。使用 `yast2 -l` 或 `yast2 --list`，則可以檢視一

個清單，其中包含您系統中所有可用的模組。例如，使用 `yast2 lan` 可啟動網路模組。

8.1 YaST 語言

若要變更 YaST 的語言，請在「YaST 控制中心」中選取「*系統*」>「*語言選擇*」。選擇語言之後，便可結束「YaST控制中心」，登出系統後再次登入。下次啟動 YaST 時，就會使用新的語言設定。這也會變更整個系統的語言。

如果您必須使用其他語言卻不想變更系統的語言設定，則請在執行 YaST 時將 `LANG` 變數設定為偏好的語言。請使用 *`langcode_statecode`* 格式的長語言代碼。例如，若是美式英文，請輸入 `LANG="en_US" yast2`。

這個指令會以指定的語言執行 YaST。此語言只在此 YaST 工作階段中有效。終端機、其他使用者以及其他工作階段的語言設定，都維持不變。

如果是透過 SSh 遠端執行 YaST，YaST 會使用本地系統的語言設定。

8.2 YaST 控制中心

在圖形模式啟動 YaST 時，會開啟「YaST 控制中心」，如圖形 8.1「YaST 控制中心」[119頁]所示。左邊的框架包含可用的類別。當您按一下類別時，就會在右框架列出內容。然後選取想要的模組。例如，如果選取「*硬體*」，再按一下右框架的「*音效*」，就會開啟音效卡的組態對話方塊。個別項目的組態通常由數個步驟組成。按「*下一步*」，繼續進行下面的步驟。

大部分模組的左框架都會顯示說明文字，提供組態建議並解釋必要的項目。若要在沒有說明框架的模組中取得說明，請按 F1 或選擇「*說明*」。選取需要的設定之後，在組態對話方塊的最後一頁中按下「*接受*」，就能完成琵序。這時組態便完成儲存。

圖形 8.1　YaST 控制中心

注：YaST 軟體管理 Gtk 和 Qt 前端

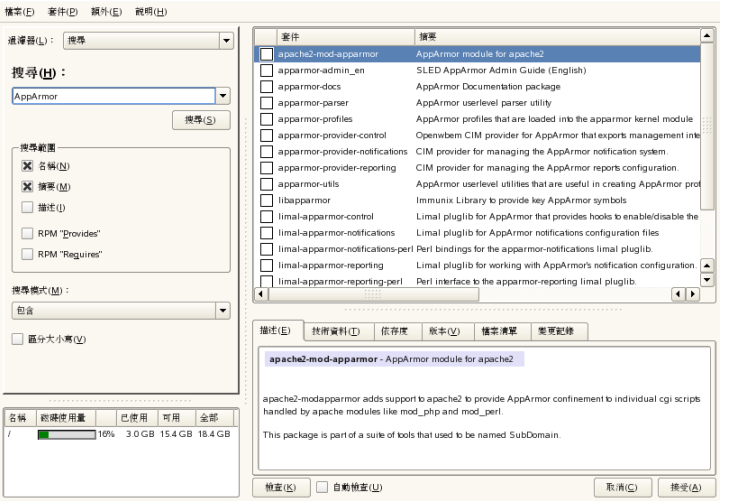
YaST 配備了兩個前端，具體視您系統上安裝的桌面而定。依預設，YaST gtk 前端在 GNOME 桌面上執行，而 YaST qt 前端在其他桌面上執行。該設定是透過 `/sbin/yast2` 程序檔中的 `WANT_UI` 變數定義的。從功能上講，gtk 前端與手冊中說明的 qt 前端極其相似，但有一項例外，gtk 軟體管理模組與 qt 連接埠有相當大的差別。

8.3 軟體

8.3.1 安裝和移除軟體

若要安裝、解除安裝和更新機器中的軟體，請使用「*軟體*」>「*軟體管理*」。這會開啟套件管理員對話方塊，如圖形 8.2「YaST 套件管理員」[120頁]所示。

圖形 8.2　YaST 套件管理員



在 SUSE® Linux Enterprise 中，是以套件的形式取得軟體。一般來說，套件中會包含程式所需的所有元件：程式本身、組態檔案以及所有說明文件。個別套件的清單會顯示在個別套件視窗的右側。此清單的內容，視目前選取的過濾器而定。例如，如果選取「*模式*」過濾器，個別套件視窗就會顯示目前選擇的所有套件。

在套件管理員中，每個套件的狀態可決定要如何處理該套件，例如「安裝」或「刪除」。在狀態方塊中，此狀態會在每行的開頭以符號顯示。在項目上按一下滑鼠右鍵，從功能表按一下或選擇想要的狀態，就可以切換狀態。根據目前的情況，可能無法選擇部分狀態旗標。例如，無法將尚未安裝的套件設為「刪除」。利用「*說明*」>「*符號*」檢視可用的狀態旗標。

個別套件視窗中，不同套件所使用的字型顏色能提供其他資訊。安裝媒體上若有較新的版本可供已安裝的套件使用，就會顯示為藍色。已安裝的套件版本號碼若比安裝媒體上的版本更高，就會顯示為紅色。不過，由於套件的版本編號並非一直都是線性增加，因此資訊可能不夠完美，但也應該足以指出有問題的套件。如有需要，請檢查版本號碼。

安裝套件

若要安裝套件，請選擇要安裝的套件，然後按一下「*接受*」。選擇的套件必須是「*安裝*」狀態圖示。套件管理員會自動檢查其依存度，並選擇其他所需的套

件 (依存度解決方案)。若要在按一下「*接受*」之前檢視其他安裝所需的套件，請從主功能表選擇「*額外*」>「*顯示自動套件變更*」。在安裝套件之後，請按一下「*安裝更多套件*」繼續使用套件管理員，或按一下「*完成*」將其關閉。

套件管理員會提供預先選取的群組以進行安裝。您可以選擇整個群組，而不要選取單一套件。若要檢視這些群組，請使用左側框架中的「*過濾器*」。

提示：所有可用套件的清單

若要顯示安裝媒體中的所有套件，請使用「*套件群組*」過濾器，並在樹狀目錄下方選擇「*zzz 全部*」。由於 SUSE Linux Enterprise 包含大量的套件，因此要顯示此份冗長的清單可能需要花費一些時間。

安裝和移除模式

「*模式*」過濾器會根據應用程式用途 (如檔案或列印伺服器) 對程式套件進行分組。這時會列出不同的「*模式*」過濾器群組，其中包含預先選取的已安裝套件。

在行的開頭處按一下狀態方塊，來安裝或解除安裝此模式。直接以滑鼠右鍵在選項按一下並使用內容功能表，選擇模式。從右邊顯示目前模式所包含套件的個別套件綜覽中，選取或取消選取個別套件。

安裝和移除語言支援

若要尋找特定語言套件，例如程式使用者介面、文件和字型的翻譯文字，請使用「*語言*」過濾器。此過濾器會顯示 SUSE Linux Enterprise 所支援的所有語言清單。如果您選取其中一種語言，右邊框架就會顯示該種語言適用的所有套件。其中套用至您目前軟體選項的所有套件，都會自動加上標籤以進行安裝。

若要讓某個語言從您的系統解除安裝，請在語言清單中選取語言，並取消勾選位於一行開頭的狀態方塊。

注

因為特定語言套件可能需要其他套件，因此套件管理員可能會加選其他套件來安裝。

套件和安裝來源

如果您只要尋找特定來源的套件，請使用「*安裝來源*」過濾器。在預設設定中，這個過濾器會顯示選定來源的所有套件清單。若要限制這份清單，請使用次要過濾器。

若要檢視選定安裝來源之所有已安裝套件的清單，請選擇「*安裝來源*」過濾器，然後在「*次要過濾器*」中選擇「*安裝摘要*」，然後取消勾選除了「*保留*」以外的所有核取方塊。

可以像平常一樣，變更個別套件視窗中的套件狀態。不過，變更的套件可能不再符合搜尋準則。若要移除清單中的這類套件，請使用「*更新清單*」來更新清單。

安裝來源套件

您通常可以取得包含程式來源檔案的套件。執行程式並不需要來源檔案，但您可能想要安裝來源以編譯程式的自定版本。

若要安裝所選程式的來源，請標示「*來源*」欄位中的核取方塊。如果您看不到核取方塊，表示安裝來源並不包含套件來源。

儲存套件選擇

如果您要在數個電腦中安裝相同的套件，您可以將組態儲存至檔案以供其他系統使用。若要儲存您對套件的選擇，請在功能表中選擇「*檔案*」>「*輸出*」。若要輸入已備妥的選擇，請依序使用「*檔案*」>「*輸入*」。

重要：硬體相容性

因為這項功能會儲存確切的套件清單，所以只有當來源和目標系統上的硬體相同時才可靠。對於更為複雜的情形，則 AutoYaST 會是較好的選擇，如第 5 章「*自動安裝*」 [77頁]所述。

移除套件

若要移除套件，請指派移除套件的正確狀態，然後按一下「*接受*」。選擇的套件必須是「*刪除*」狀態。如果已經標示要刪除其他已安裝套件所需的套件，套件管理員就會發出具有詳細資訊與替代方案的警示。

重新安裝套件

如果您發現套件中有損毀的檔案，或者要從安裝媒體中重新安裝套件的原始版本，請重新安裝套件。若要重新安裝套件，請選擇要重新安裝的套件，然後按一下「*接受*」。選擇的套件必須是「*更新*」狀態。如果安裝的套件發生任何依存度問題，套件管理員就會發出具有詳細資訊與替代方案的警示。

搜尋套件、應用程式和檔案

若要尋找特定套件，請使用「*搜尋*」過濾器。輸入搜尋字串，然後按一下「*搜尋*」。您可以指定各種搜尋準則來限制搜尋範圍，以顯示較少或甚至一個套件。您也可以在「*搜尋模式*」中使用萬用字元與正規表示式來定義特殊搜尋型式。

提示：快速搜尋

除了「*搜尋*」過濾器，套件管理員的所有清單都有快速搜尋的功能。只要輸入字母，就可以將游標移動到清單中以該字母為開頭名稱的第一個套件。游標必須位於清單中 (按一下該清單即可)。

若要按名稱尋找套件，請選擇「*名稱*」，在搜尋欄位中輸入要尋找的套件名稱，然後按一下「*搜尋*」。若要按說明中的文字尋找套件，請選擇「*摘要*」和「*說明*」，輸入搜尋字串，然後按一下「*搜尋*」。

若要搜尋包含特定檔案的套件，請輸入檔案名稱，選擇「*RPM提供*」，然後按一下「*搜尋*」。若要尋找依賴特殊套件的所有套件，請選擇「*RPM 要求*」，輸入套件名稱，然後按一下「*搜尋*」。

如果您熟悉 SUSE Linux Enterprise 的套件結構，便可以使用「*套件群組*」過濾器來按主題尋找套件。此過濾器會在左邊的樹狀結構中，依主題排序程式套件，例如，應用程式、開發與硬體。展開的分支越多，選擇就越明確。這表示在個別套件視窗顯示的套件也越少。

安裝摘要

在選擇要安裝、更新或刪除的套件後，使用「*安裝摘要*」來檢視安裝摘要。摘要會顯示當您按一下「*接受*」時，會對套件造成什麼影響。使用左邊的核取方塊來過濾套件，在個別的套件視窗中檢視。例如，若要檢查已經安裝哪些套件，只選取「*保留*」並取消選取其他核取方塊。

可以像平常一樣，變更個別套件視窗中的套件狀態。不過，對應的套件可能會不再符合搜尋準則。若要移除清單中的這類套件，請使用「*更新清單*」來更新清單。

套件的相關資訊

您可以透過框架右下方的索引標籤取得所選套件的相關資訊。如果有其他版本的套件，您就會取得兩種版本的資訊。

提供選取套件的「*說明*」索引標籤會自動啟用。若要檢視套件大小、版本、安裝媒體等相關資訊和其他的技術性詳細資訊，請選擇「*技術資料*」。關於已提供或所需檔案的資訊都位於「*依存度*」中。若要檢視包含安裝來源的可用版本，請按一下「*版本*」。

磁碟使用

選擇軟體時，模組左下方資源視窗會顯示所有已裝載檔案系統的預期磁碟使用量。每增加一個選擇，就會使色條的圖形增長。只要圖形仍是綠色，即代表有足夠的空間。隨著磁碟空間逐漸用盡，圖形的顏色會慢慢變成紅色。如果您選擇要安裝的套件過多，就會顯示警示。

檢查依存度

部分套件依存於其他套件。這代表必須安裝另一個套件，才能正常運作該套件軟體。某些套件會具有相同或相似的功能。如果這些套件會使用到相同的系統資源，就不應該同時進行安裝 (套件衝突)。

套件管理員啟動時，就會檢查系統並顯示已安裝的套件。如果您選擇安裝和移除其他套件，套件管理員就會自動檢查其依存度，並選擇其他所需的套件(依存度解決方案)。如果選取或取消選取相互衝突的套件，套件管理員就會加以指出，並提交可解決問題的建議 (衝突解決方案)。

若要啟用自動依存度檢查，請選取資訊視窗之下的「*自動檢查*」。啟用「*自動檢查*」以後，則套件狀態若有任何變更，就會觸發自動檢查程序。因為套件選擇的一致性永遠會受到監視，所以這個功能很實用。不過，此程序耗費資源，而且會使套件管理員的速度減緩。基於此原因，預設並不會啟用自動檢查功能。不論「*自動檢查*」的狀態為何，當您以「*接受*」來確認自己的選擇時，都會執行一致性檢查程序。

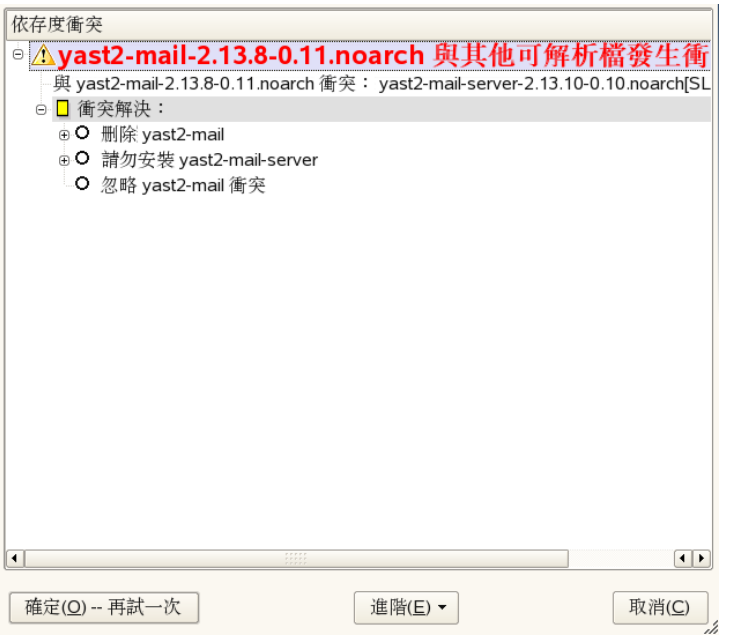
當您按一下資訊視窗之下的「*檢查*」時，套件管理員就會檢查目前的套件選擇是否會產生無法解決的套件依存度或衝突。如果有無法解決的依存度存在，就會自動選取所需的其他套件。若為套件衝突，套件管理員就會開啟對話方塊，顯示該衝突並提供解決問題的不同選項。

例如，可能無法同時安裝 `sendmail` 與 `postfix`。圖形 8.3「套件管理員的衝突管理」[126頁]顯示會提示您做決定的衝突訊息。`postfix` 已經安裝。因此，您應該避免安裝 `sendmail`、移除 `postfix` 或承擔忽略該衝突的風險。

警告：處理套件衝突

除非您有豐富的經驗，建議您遵循 YaST 的建議，否則存在的衝突可能會危及系統的穩定性與功能。

圖形 8.3　套件管理員的衝突管理



安裝 -devel 套件

套件管理員可提供 devel 和 debug 套件的快速及簡易安裝功能。若要為已安裝的系統安裝所有的 devel 套件，請依序選擇「*額外*」>「*安裝所有符合的 -devel 套件*」。若要為已安裝的系統安裝所有的 debug 套件，請依序選擇「*額外*」>「*安裝所有符合的 -debuginfo 套件*」。

8.3.2 安裝附加產品

附加產品 (Add-on) 就是系統的延伸程式。您可以安裝協力廠商的附加產品或是 SUSE Linux Enterprise 的特殊延伸程式，例如 SDK 附加程式或是包含二進位驅動程式的 CD。若要安裝新的附加程式，請使用「*軟體*」>「*附加產品*」。您可以選取各種類型的產品媒體，例如 CD、FTP 或是本地目錄。您也可以直接執行 ISO 檔案。若要以 ISO 檔案媒體方式來新增附加程式，請選取「*本地目錄*」，然後選擇「*ISO 映像*」。

成功新增附加媒體之後，將會顯示套件管理員視窗。如果此附加程式有提供新的「模式」，請使用「*模式*」過濾器來檢視新項目。若要檢視選定安裝來源之所有套件的清單，請選取「*安裝來源*」過濾器，然後選擇要檢視的安裝來源。若要依套件群組來檢視選定附加程式的套件，請選取「*套件群組*」為次要過濾器。

提示：建立自定的附加產品

您可以使用YaST附加產品建立程式來建立自己的附加產品。請參閱中http://developer.novell.com/wiki/index.php/Creating_Add-On_Media_with_YaSTの YaST 附加產品建立程式相關資訊。您可以在http://developer.novell.com/wiki/index.php/Creating_Add-Ons中找到技術背景資訊。

8.3.3 選擇安裝來源

您可以使用幾種多重安裝來源。請選擇來源，並使用「*軟體*」>「*安裝來源*」讓它們進行安裝或更新。例如，您可以將 SUSE Software Development Kit 指定為安裝來源。啟動時，畫面會顯示所有之前註冊的來源清單。從 CD 進行一般安裝之後，只會列出該安裝CD。按一下「*新增*」，可以讓清單包含其他來源。來源可以是 CD、DVD 或 NFS 及 FTP 伺服器等網路來源。甚至可以選擇本地硬碟上的目錄，來作為安裝媒體。如需更多詳細資訊，請參閱詳細的 YaST 說明內容。

所有已註冊的來源，在清單的第一欄都有啟用狀態。您可以按一下「*啟用或停用*」，啟用或停用個別的安裝來源。在安裝軟體套件或更新的期間，YaST 會從啟用的安裝來源清單中選擇適用的項目。當您使用「*關閉*」結束模組時，目前設定就會進行儲存，並套用到「*軟體管理*」與「*系統更新*」組態模組。

8.3.4 註冊 SUSE Linux Enterprise

如需取得技術支援和產品更新，您的系統必須先註冊並啟用。如果您想在安裝程序中略過註冊的步驟，請透過「*軟體*」中的「*Novell 客戶中心組態*」模組取得註冊的協助。這個對話方塊與第 3.14.4 節「Novell Customer Center 組態」[37頁]中所述的一樣。

8.3.5 YaST 線上更新

使用 YaST 線上更新安裝重要的更新與改良。包含修補程式的產品專用更新目錄中提供了適用於 SUSE Linux Enterprise 的最新更新。若要新增或是移除目錄，請依照第 8.3.3 節「選擇安裝來源」[127頁]所述，使用「*軟體*」>「*安裝來源*」模組。

注：存取更新目錄時發生錯誤

如果您無法存取更新目錄，可能表示訂閱已過期。SUSE Linux Enterprise 通常提供一年或三年的訂閱，您只能在這個時間段內存取更新目錄。一旦訂閱期結束，此存取權將被拒絕。

存取更新目錄遭拒絕時，會顯示一則警告訊息，建議您造訪 Novell Customer Center 以檢查您的訂閱。Novell Customer Center 的網址為 `http://www.novell.com/center/`。

若要以 YaST 安裝更新與改良，請執行「*軟體*」>「*線上更新*」。目前您系統可用的所有新修補程式 (選購程式除外) 均已標示為即將安裝。按一下「*接受*」自動安裝這些修補程式。安裝完成之後，請按「*結束*」確認。您的系統現在已是最新狀態。

術語定義

套件
: 套件是 rpm 格式的壓縮檔案，其中包含特定程式所需的檔案。

修補程式
: 修補程式包括一個或多個套件—可能是完整的套件，也可能是 patchrpm 或 deltarpm 套件—還可能會引入對尚未安裝之套件的相依性。

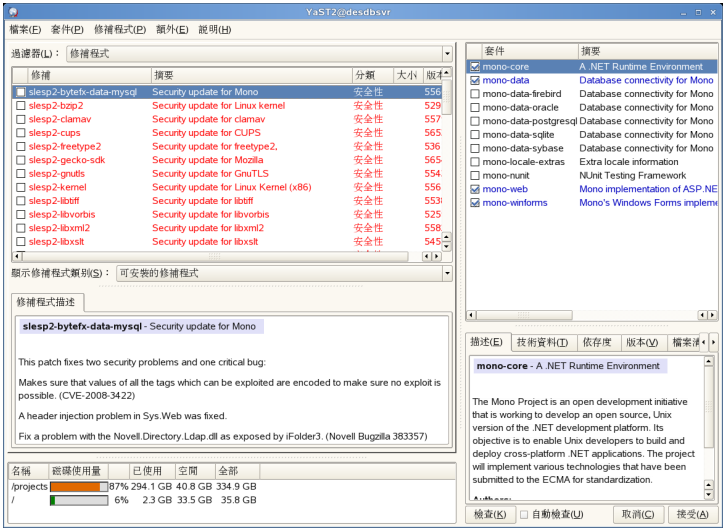
patchrpm
: patchrpm 只包括自從為 SUSE Linux Enterprise 10 首次發行以來，已有更新的檔案。其下載大小通常明顯小於套件的大小。

deltarpm
: deltarpm 只包括某套件之兩個定義版本之間的二進位差異，因此，它的下載大小最小。在安裝之前，必須在本地機器上重建 rpm 套件。

手動安裝修補程式

「*線上更新*」視窗包含五個部份。左邊是所有可用的修補程式清單。修補程式清單下方會顯示所選修補程式的描述。左欄下方顯示磁碟使用率。右欄列出所選修補程式所包含的套件 (修補程式可包含多組套件)，且下方會列出所選套件的詳細描述。

圖形 8.4 *YaST 線上更新*



修補程式畫面會列出 SUSE Linux Enterprise 的可用修補程式。修補程式根據安全性關聯程度排序。修補程式名稱的顏色以及滑鼠游標下方的快顯視窗可指出該修補程式的安全性狀態：安全性 (紅色)、建議 (藍色) 或選擇性 (黑色)。修補程式存在三種不同的檢視窗。使用「*顯示修補程式類別*」可以切換不同的檢視窗：

「*可安裝的修補程式*」(預設檢視窗)
適用於系統上安裝的套件但目前尚未安裝的修補程式。

「*可安裝的和已安裝的修補程式*」
所有適用於系統上安裝之套件的修補程式。

「*所有修補程式*」
可用於 SUSE Linux Enterprise 的所有修補程式。

清單項目包含符號與修補程式名稱。如需可能符號清單，請按 Shift + F1。「安全性」和「建議」狀態的修補程式所要求的動作已自動預設。這些動作是「*自動安裝*」、「*自動更新*」或「*自動刪除*」。「選擇性」修補程式的動作未預設—用滑鼠右鍵按一下修補程式，然後從清單中選擇一個動作。

如果從更新目錄以外的其他目錄安裝最新的套件，則可以使用此安裝來達到此套件修補程式的需求。在此情況下，會有一個核取標記顯示在修補程式摘要的前面。僅當您將修補程式標示為已安裝時，該修補程式才會顯示在清單中。事實上這並沒有安裝修補程式 (因為套件已經是最新的)，而是將修補程式標示為已安裝。

大部分的修補程式都會包含多套件的更新。若要變更對單一套件所執行的動作，請在套件視窗中的套件上按一下滑鼠右鍵，並選擇動作。當您標示完所有要處理的修補程式與套件後，請按一下「*接受*」繼續。

提示：禁用 deltarpm

由於從 deltarpm 重建 rpm 套件需要佔用大量的記憶體和 CPU 資源，因此，出於效能的考量，某些設定或硬體組態可能要求禁用 deltarpm。若要禁用 deltarpm，請編輯檔案 `/etc/zypp/zypp.conf`，並將 `download.use_deltarpm` 設定為 `false`。

另一個更新軟體的工具，就是 KDE 和 GNOME 的 ZENworks 更新程式 Applet。ZENworks 更新程式可協助監控新的修補程式，同時也提供快速更新功能。若需更多資訊，請參閱第 9.2 節「使用 ZEN 工具管理套件」[185頁]。

8.3.6 自動線上更新

YaST 也可讓您設定自動更新。請選取「*軟體*」>「*自動線上更新*」。設定「*每日*」或「*每週*」更新。某些修補程式 (例如核心更新) 需要使用者介入，而這可能會讓自動更新程序停止下來。請勾選「*略過互動性質的修補程式*」，讓更新程序自動進行。在此情況下，請偶爾手動執行「*線上更新*」來安裝互動性質的修補程式。

當勾選「*只下載修補程式*」時，修補程式就會在指定的時間獲得下載，但不安裝。您必須手動安裝它們。這些修補程式會下載到 rug 快取目錄，預設為 `/var/cache/zmd/web`。請使用 `rug get-prefs cache-directory` 指令來取得

目前的 rug 快取目錄。若需有關 rug 的詳細資訊，請參閱第 9.1 節「使用 rug 透過指令行更新」[182頁]。

8.3.7 從修補程式光碟進行更新

注

在 s390 系統中，修補程式 CD 更新選項不可用。

「*軟體*」區段的「*修補程式光碟更新*」模組會從光碟安裝修補程式，而不是從 FTP 伺服器進行安裝。使用光碟的優點是能更迅速進行更新。一旦插入修補程式光碟之後，光碟上的所有修補程式都會顯示在對話方塊中。在修補程式清單中選擇想要安裝的套件。如果修補程式光碟不存在，該模組就會發出錯誤訊息。請插入修補程式光碟，然後重新啟動該模組。

8.3.8 更新系統

使用「*軟體*」>「*系統更新*」來更新安裝在系統上的 SUSE Linux Enterprise 版本。操作期間只能更新應用程式軟體，不能更新基本系統。若要更新基本系統，請從安裝媒體將電腦開機，例如光碟。在 YaST 中選擇安裝模式時，請選取「*更新*」。

更新系統的程序與全新安裝類似。一開始，YaST 會檢查系統，決定適當的更新策略，然後在建議對話方塊中顯示結果。按一下「*變更*」或個別項目，以變更任何詳細資料。

更新選項

設定系統的更新方法。有兩個選項可用。

使用依選項安裝新軟體和功能進行更新
: 若要將整個系統更新至最新的軟體版本，請選擇其中一個預先定義的選項。這些選項會確定先前不存在的套件也能進行安裝。

只更新已安裝套件
: 此選項只會更新系統上已存在的套件。不會安裝任何新功能。

除此之外，您可以使用「*刪除過時套件*」，移除新版本中不存在的套件。根據預設，會事先選取此選項，以避免過時的套件佔據硬碟空間。

套件

按一下「*套件*」啟動套件管理員，以及選取或取消選取要進行更新的個別套件。任何套件衝突應該利用一致性檢查來解決。如需使用套件管理員的詳細資訊，請參閱第 8.3.1 節「安裝和移除軟體」 [119頁]。

備份

更新期間，部分套件的組態檔案可能會以新版本取代。因為您可能已修改過目前系統中的部分檔案，所以套件管理員通常會針對被取代檔案進行備份。請利用此對話方塊來決定備份的範圍。

重要：備份範圍

此備份並不包括軟體。僅包含組態檔案。

語言

這裡會列出系統上目前安裝的主要與其他語言。您可以在顯示的組態中按一下「*語言*」，或依序選擇「*變更*」>「*語言*」進行變更。您可選擇性將鍵盤配置與時區調整為使用主要語言的區域。如需這些語言選項的詳細資訊，請參閱第 8.5.15 節「語言選擇」 [149頁]。

更新的重要資訊

系統更新是非常複雜的程序。對於每個程式套件而言，YaST 必須先檢查電腦上安裝的版本，然後決定需要進行的操作，才能以新的版本正確取代舊版套件。YaST 也會嘗試套用已安裝套件的任何個人設定。

大部分情況下，YaST 以新版套件取代舊版套件時不會發生問題。在進行更新之前，備份現有的系統可以確保現有組態不會在更新期間遺失。更新完成之後，就可以手動解決衝突問題。

8.3.9 安裝到目錄中

這個 YaST 模組可將套件安裝到您指定的目錄中。選取要將根目錄放置在哪裡、如何命名目錄，以及系統類型和要安裝的軟體。輸入此模組後，YaST 會判斷系統設定，並列出預設目錄、安裝指示，以及要安裝的軟體。按一下「*變更*」來編輯設定值。按一下「*接受*」之後，才會確認所有變更。完成所有變更之後，請按一下「*下一步*」，直到收到安裝已完成通知為止。按一下「*完成*」，離開此對話方塊。

8.3.10 檢查媒體

如果您使用 SUSE Linux Enterprise 安裝媒體時遇到任何問題，您可以使用「*軟體*」>「*媒體檢查*」來檢查 CD 或 DVD。媒體問題比較可能發生在您自己燒錄的媒體中。若要檢查 SUSE Linux Enterprise CD 或 DVD 是否有錯誤，請將該媒體放入設備中並執行此模組。按一下「*啟動*」，然後 YaST 會檢查媒體的 MD5 檢查總數。這可能會花費幾分鐘。如果偵測到錯誤，您就不應該使用此媒體來進行安裝。

8.4 硬體

新硬體必須先依照廠商指定的方式來安裝或連接。開啟外部設備，然後啟動適當的 YaST 模組。YaST 能自動偵測到大部分的設備，並顯示技術資料。如果自動偵測失敗，YaST 就會提供設備清單 (型號、廠商等)，讓使用者由此選擇適當的設備。如需詳細資訊，請參閱硬體隨附的文件。

重要：指定型號

如果設備清單中沒有您的型號，請嘗試選擇名稱類似的型號。不過，由於相似的名稱不一定代表具備相容性，所以在某些情況下必須與型號完全相符。

8.4.1 紅外線設備

使用「*硬體*」>「*紅外線設備*」，設定紅外線設備。按一下「*啟動 IrDa*」開始設定。您可以在這裡設定「*埠*」和「*限制鮑率*」。

8.4.2 圖形卡和顯示器

使用「*硬體*」>「*圖形卡和顯示器*」來設定圖形卡和顯示器。它會使用SaX2介面，如第 8.14 節「SaX2」 [175頁]所述。

8.4.3 印表機

使用「*硬體*」>「*印表機*」來設定印表機。若印表機已正確連接到系統，應該就會自動被偵測出來。如需更多關於使用 YaST 設定印表機的說明，請參閱第 23.4 節「設定印表機」 [404頁]。

8.4.4 硬碟控制器

系統的硬碟控制器通常會在安裝期間完成設定。如果要新增控制器，請使用「*硬體*」>「*磁碟控制器*」整合到系統。您也可以修改現有的組態，但一般而言不需要這樣做。

對話方塊會以清單顯示所偵測到的硬碟控制器，可讓使用者利用特定的參數指定適合的核心模組。在系統中設定永久性儲存之前，請使用「*測試模組載入*」，檢查目前的設定是否能夠運作。

警告：硬碟控制器的組態

建議您先測試該設定，然後才將設定永久儲存在系統。不正確的設定會使系統無法開機。

8.4.5 硬體資訊

使用「*硬體*」>「*硬體資訊*」顯示偵測到的硬碟和技術資料。按一下樹狀結構的任何節點，取得關於設備的更多資訊。例如當您需要有關硬體的資訊而希望提交支援要求時，這個模組就會特別實用。

按一下「*儲存至檔案*」將顯示的硬體資訊儲存至檔案。選取想要的目錄和檔案名稱，然後按一下「*儲存*」來建立檔案。

8.4.6 IDE DMA 模式

使用「*硬體*」>「*IDE DMA 模式*」，為已安裝系統的 IDE 硬碟以及 IDE CD 與 DVD 光碟機啟用和停用 DMA 模式。此模組對 SCSI 設備沒有任何作用。DMA 模式能夠大幅提昇系統的效能與資料傳輸速度。

在安裝期間，目前的 SUSE Linux Enterprise 核心會自動啟用硬碟的 DMA，但不會啟用 CD 光碟機的 DMA，因為如果預設啟用所有磁碟機的 DMA，經常會造成 CD 光碟機方面的問題。請使用 DMA 模組來啟用磁碟機的 DMA。如果磁碟機可支援 DMA 模式而不會發生問題，啟用 DMA 將可提昇磁碟機的資料傳輸速度。

注

DMA (直接記憶體存取) 表示您的資料可以跳過處理器控制，直接傳輸到 RAM。

8.4.7 IBM System z：DASD 設備

您可以使用兩種可能方式，將 DASD 新增到已安裝的系統中：

YaST
: 若要將 DASD 新增到已安裝的系統，請使用 YaST DASD 模組 (「*硬體*」>「*DASD*」)。在第一個畫面中，選擇要讓您的 Linux 安裝使用的磁碟，然後按一下「*執行動作*」。選擇「*啟動*」，然後按「*下一步*」，離開對話方塊。

指令行
: 發出下列指令：

```
dasd_configure 0.0.0150 1 0
```

將 *0.0.0150* 取代成 DASD 實際連接的通道號碼。如果 DASD 應該以 DIAG 模式存取，則指令行最後一個 0 的位置應該是 1。

注

在任何一種情況下，您都必須執行下列指令

```
mkinitrd
zipl
```

以使變更永久生效。

8.4.8 IBM System z：ZFCP

若要將更多附加 FCP 的 SCSI 設備新增到已安裝的系統，請使用 YaST ZFCP 模組 (「*硬體*」>「*ZFCP*」)。選擇「*新增*」，新增其他設備。從清單選擇「*通道編號*」(介面卡)，然後指定「*WWPN*」與「*FCP-LUN*」。選擇「*下一步*」與「*關閉*」，便可完成設定。請檢查 `cat /proc/scsi/scsi` 的輸出，確認已新增設備。

注

若要透過開機使變更永久生效，請執行下列指令：

```
mkinitrd
zipl
```

8.4.9 搖桿

使用「*硬體*」>「*搖桿*」，設定連接到音效卡的搖桿。在提供的清單中選取搖桿類型。若未列出您的搖桿，請選取「*一般類比搖桿*」。選取您的搖桿之後，確認搖桿已連接至電腦，並按一下「*測試*」測試其功能。按一下「*繼續*」，接著 YaST 就會安裝需要的檔案。出現「*搖桿測試*」視窗後，請移動各方向並試按所有按鈕以測試搖桿。每個動作都應顯示於視窗中。若您滿意此設定，請按一下「*確定*」返回模組，再按一下「*結束*」完成設定。

若您有 USB 設備，就不需要此組態。插入搖桿後就可開始使用。

8.4.10 鍵盤配置

若要設定控台的鍵盤，請在文字模式中執行 YaST，然後使用「*硬體*」>「*鍵盤配置*」。按一下此模組之後，就會顯示目前配置。若要選擇其他鍵盤配置，請從提供的清單中選擇希望的配置。按下鍵盤上的按鍵，在「*測試*」中測試配置。

按一下「*進階設定*」微調此設定。請調整按鍵重複速率、延遲，並在「*啟動狀態*」中選擇希望的設定，設定啟動狀態。對於「*要鎖定的設備*」，請以空格為間隔輸入要套用 Scroll Lock、Num Lock 和 Caps Lock 設定的設備清單。按一下「*確定*」完成微調。最後，在完成所有選擇之後，請按一下「*接受*」讓變更生效。

若要設定圖形環境的鍵盤，請執行圖形 YaST，然後選取「*鍵盤配置*」。如需關於圖形組態的資訊，請參閱第 8.14.3 節「鍵盤內容」 [179頁]。

8.4.11 滑鼠型號

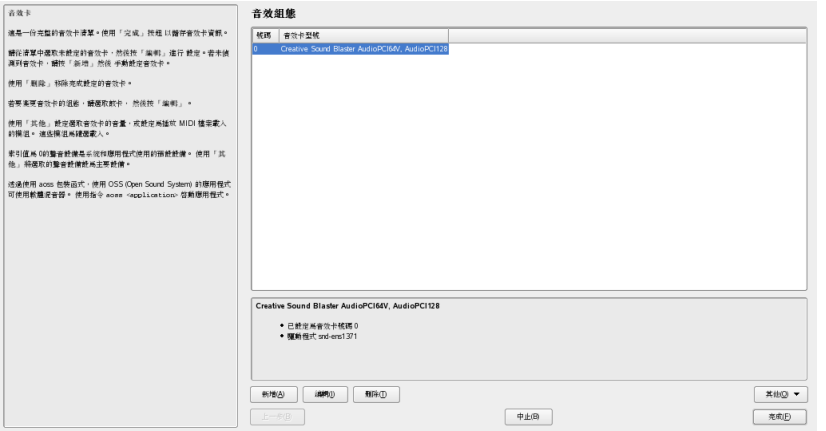
在設定圖形環境中的滑鼠時，請按一下「*滑鼠型號*」來存取 SaX2 滑鼠組態。如需詳細資訊，請參閱第 8.14.2 節「滑鼠內容」 [178頁]。

若要設定文字環境的滑鼠，請在文字模式中使用 YaST。進入文字模式並選取「*硬體*」>「*滑鼠型號*」後，使用方向鍵從提供的清單中選擇您的滑鼠。再按一下「*接受*」儲存設定，然後離開模組。

8.4.12 音效

起始安裝時，大多數的音效卡都會自動被偵測到，並且會以合理的值進行設定。若要稍後想安裝新介面卡或修改設定，請使用「*硬體*」>「*音效*」。您也可以切換介面卡的順序。

圖形 8.5　音效組態



如果 YaST 無法自動偵測您的音效卡，請執行下列步驟：

1. 按一下「*新增*」來開啟對話方塊，並在其中選取音效卡廠商與型號。請參閱音效卡文件，取得所需的資訊。您可以在 `/usr/share/doc/packages/alsa/cards.txt` 與 http://www.alsa-project.org/alsa-doc/ 中找到 ALSA 支援的音效卡對應清單，以及對應的音效模組。在完成選擇之後，請按一下「*下一步*」。

2. 在「*音效卡組態*」的第一個設定畫面中選擇組態層級：

   「*快速自動設定*」
   　您無須通過任何進一步的組態步驟，而且不會執行音效測試。可自動完成設定音效卡。

   「*一般設定*」
   　請調整輸出音量並播放測試音效。

   「*進階設定 (可變更選項)*」
   　請手動自定所有設定。

   您也可以在此對話方塊中，找到搖桿組態的捷徑。請按一下「*搖桿組態*」並在下列的對話方塊中選取搖桿類型，以進行搖桿的設定。按一下「*下一步*」繼續。

3 在「*音效卡音量*」中，測試音效卡組態並調整音量。為了避免損壞您的喇叭或聽覺，請從百分之十的音量開始。當您按一下「*測試*」時，應該要能夠聽見測試聲音。如果聽不見，請增加音量。請按「*下一步*」>「*完成*」來完成音效組態。

若要變更音效卡的組態，請前往「*音效組態*」對話方塊，選取顯示的「*介面卡型號*」後按一下「*編輯*」。使用「*刪除*」來完整移除音效卡。

請按一下「*其他*」來手動自定下列選項：

「*磁碟區*」
請使用這個對話方塊設定音量。

「*啟動編曲程式*」
如需播放 MIDI 檔案，請勾選此選項。

「*設定主要介面卡*」
請按一下「*設定主要介面卡*」來調整音效卡的順序。索引為0的音效設備將做為系統和應用程式預設使用的設備。

當您按一下 YaST 音效模組中的「*完成*」時，就會儲存所有已安裝的音效卡音量與組態。混音器設定會儲存於檔案 `/etc/asound.conf`，而 ALSA 組態資料 會附加於檔案 `/etc/modprobe.d/sound` 和 `/etc/sysconfig/hardware` 的結尾。

8.5 系統

此模組群組旨在協助您管理您的系統。群組中所有模組均與系統和伺服器相關，是可確認您系統正確執行且有效管理資料的珍貴工具。

提示：IBM System z：繼續

若為 IBM System z，請繼續第 8.5.3 節「開機載入程式組態」 [140頁]。

8.5.1 備份

您可以使用「*系統*」>「*系統備份*」來建立系統和資料的備份。然而，由此模組建立的備份並不包含整個系統。系統會儲存硬碟上重要儲存區域以備份，該

區域在嘗試還原系統時不可或缺，例如，分割區表或主要開機磁區 (MBR)。而且也會包含安裝系統時所需的 XML 組態，這項組態會用於 AutoYaST。儲存安裝媒體上可存取套件中已變更的檔案、無法存取的整個套件 (如線上更新)，以及例如許多位於 `/etc` 中或 `/home` 之組態檔案等不屬於套件的檔案，便可完成備份資料。

8.5.2 還原

使用「*系統*」>「*系統還原*」，便可讓系統從「*系統還原*」所建立的備份歸檔進行還原。首先，指定歸檔的位置 (抽取式媒體、本地硬碟或網路檔案系統)。按一下「*下一步*」來檢視個別歸檔的說明與內容，並決定要從哪個歸檔來進行還原。

您也可以解除安裝在上次備份之後新增的套件，並重新安裝在上次備份之後刪除的套件。這兩個步驟可讓您將系統確實還原到上次備份時的狀態。

警告：系統還原

因為此模組通常會安裝、取代或解除安裝許多套件與檔案，若您具有備份的經驗，才使用此模組。否則可能會遺失資料。

8.5.3 開機載入程式組態

若要為您電腦所安裝的系統設定開機，請使用「*系統*」>「*開機載入程式*」模組。如何使用 YaST 來設定開機載入程式的詳細資訊，請參閱第 21.3 節「使用 YaST 設定開機載入程式」[380頁]。

8.5.4 叢集

如需 YaST 的 Heartbeat 和高可用性組態的相關資訊，請參閱 *Heartbeat Guide*。

8.5.5 LVM

Logical Volume Manager (LVM) 這項工具能利用邏輯磁碟機自定硬碟的磁碟分割。如需更多關於 LVM 的詳細資訊，請參閱第 7.1 節「LVM 組態」[103頁]。

8.5.6 EVMS

企業磁碟區管理系統 (EVMS，Enterprise Volume Management System) 是類似 LVM 的工具，可自定磁碟分割區並可將硬碟分組成虛擬磁碟區。此工具相當有彈性、具擴充性，而且可以針對不同的磁碟區管理系統需求，使用插件模型量身訂做。

EVMS 可與現有的記憶體與磁碟區管理系統相容，例如 DOS、Linux LVM、GPT (GUID 分割區表)、IBM System z、Macintosh 與 BSD 分割區。如需詳細資訊，請參閱http://evms.sourceforge.net/。

8.5.7 使用 YaST 磁碟分割程式

您可使用如圖形 8.6「YaST 磁碟分割程式」[142頁] 所示的進階磁碟分割程式，手動修正一個或多個硬碟分割區。分割區可以新增、刪除、調整大小或編輯。從此 YaST 模組還可以存取軟體式 RAID、EVMS 與 LVM 組態。

警告：重新分割執行中系統

儘管可以在系統執行時對其重新分割，但是這樣極有可能會造成資料遺失的風險。儘量避免重裝分割安裝的系統，若要這樣做，請始終在開始之前對資料執行完整的備份。

*圖形 **8.6*** *YaST 磁碟分割程式*

提示：IBM System z：設備名稱

IBM System z 僅識別 DASD 與 SCSI 硬碟，不支援 IDE 硬碟。所以這些設備會在分割區表上顯示為 `dasda` 或 `sda` 做為第一個識別的設備。

所有連線硬碟上的現有或建議分割區都顯示在 YaST「「*進階磁碟分割程式*」對話方塊的清單中。全部硬碟都列為不含編號的設備，例如 `/dev/hda` 或 `/dev/sda` (或 `/dev/dasda`)。各分割區則列示為這些設備的一部分，例如 `/dev/hda1` 或 `/dev/sda1` (或 `/dev/dasda1`)。硬碟的大小、類型、檔案系統和定點以及他們的分割區也都會顯示在其中。裝載點描述分割區出現在 Linux 檔案系統樹狀結構上的位置。

安裝期間若執行專家對話方塊，同時會自動選取並列出所有可用的硬碟空間。若要提供更多磁碟空間給 SUSE Linux Enterprise®，請在清單中由下而上 (從硬碟的最後一個分割區開始往上) 釋放所需空間。例如，如果您有三個分割區，您不可以把第二個供給 SUSE Linux Enterprise 專用，而將第三和第一個保留給其他作業系統。

分割區類型

提示：IBM System z：硬碟

在 IBM System z 平台上，SUSE Linux Enterprise Server 支援 SCSI 硬碟和 DASD (直接存取儲存設備)。SCSI 磁碟分割描述如下，DASD 分割表不可以含有超過三個的分割登錄。

每一個硬碟都有一個分割區表，可以儲存四筆登錄。分割區表中的一筆登錄可以對應一個主分割區或延伸分割區。不過，只能出現一個延伸分割區項目。

主分割區僅由指派給特定作業系統之連續範圍的磁柱 (實體磁碟區) 組成。如果只有主分割區，每一個硬碟將受限於四個分割區，因為無法再容納分割區表。這就是使用延伸分割區的原因。延伸分割區也是由連續範圍的磁柱組成，但延伸分割區又可以再劃分為*邏輯分割區*。邏輯分割區不需要在分割區表中登錄。換句話說，延伸分割區是邏輯分割區的容器。

如果您需要四個以上的分割區，請建立延伸分割區當作第二至第四個分割區。這個延伸分割區應該包含所有剩餘可用的整個磁柱範圍。接著在延伸分割區中建立多個邏輯分割區即可。對於 SCSI、SATA 和 Firewire 磁碟，邏輯分割區的最大數目為 15；對於 (E)IDE 磁碟，最大數目為 63。Linux 對於使用的分割區類型沒有限制。主分割區與邏輯分割區都可以正常運作。

提示：具有 GPT 磁碟標籤的硬碟

對使用 GPT 磁碟標籤的結構來說，主要分割區的數量並無任何限制。因此，就沒有任何邏輯分割區。

建立分割區

若要從頭建立分割區，請如下執行：

1. 選擇「*建立*」。如果有數個連接的硬碟，會出現一個選擇對話方塊，您可在其中為新分割區選擇一個硬碟。

2. 指定分割區類型(主分割區或延伸分割區)。最多可建立四個主分割區，或是三個主分割區和一個延伸分割區。在延伸分割區內建立數個邏輯分割區 (請參閱章節「分割區類型」 [143頁])。

3 選擇要使用的檔案系統與裝載點。YaST 建議為每個建立的分割區都準備一個裝載點。如需各種檔案系統的詳細資訊，請參閱第 25 章「*Linux 的檔案系統*」[431頁]。

4 依您的安裝需求，指定其他檔案系統選項。例如，如果您需要持續性設備名稱，則必須這樣做。關於可用選項的詳細資料，請參閱 章節「編輯分割」[144頁]。

5 按一下「*確定*」>「*套用*」至您的磁碟分割設定，並離開磁碟分割模組。

若您在安裝過程中建立分割區，會回到安裝綜覽畫面。

編輯分割

如果您建立一個新的分割區或修正一個現有分割區，可設定多項參數。對於新的分割區，YaST 會設定適當的參數，所以通常並不需要任何修正。若要手動編輯您的分割設定，請如下操作：

1 選擇分割區。

2 按一下「*編輯*」，編輯分割區並設定參數：

檔案系統 ID
: 即使您在此階段不想要格式化分割區，也請指派一個檔案系統 ID 給它，以確保分割區可以正確註冊。可能的值包括「*Linux*」、「*Linux swap*」、「*Linux LVM*」、「*Linux EVMS*」和「*Linux RAID*」。如需關於 LVM 和 RAID 的詳細資訊，請參閱第 7.1 節「LVM 組態」[103頁]和第 7.2 節「軟體 RAID 組態」[111頁]。

檔案系統
: 在此變更檔案系統或格式化分割區。變更檔案系統或者以不可還原的方式重新格式化分割區均會刪除分割區中的所有資料。關於各種檔案系統的詳細資料，請參閱第 25 章「*Linux 的檔案系統*」[431頁]。

檔案系統選項
: 在此為選擇的檔案系統設定各種參數。大部分情況都可接受預設值。

加密檔案系統
: 如果您啟用加密，所有資料都會以加密格式寫入硬碟。這可增加敏感資料的安全性，但會略為降低系統速度，因為加密處理需要一點時

間。如需檔案系統加密的詳細資訊，請參閱第 47 章「*加密分割區和檔案*」[785頁]。

Fstab 選項
: 指定全域檔案系統管理檔案 (`/etc/fstab`) 中包含的各個參數。預設值應足以滿足大多數安裝需求。例如，您可以將檔案系統的識別資訊由設備名稱變更為磁碟區標籤。在磁碟區標籤中，您可以使用除 / 和空格以外的所有字元。

定點
: 指定在檔案系統樹狀結構中，要裝載分割區的目錄。由各種 YaST 提案選取，或指定任何其他名稱。

3 選取「*確認*」>「*套用*」啟用分割區。

進階使用者選項

「*進階使用者*」會開啟包含以下指令的功能表：

重新讀取分割區表
: 重新從磁碟讀取磁碟分割。例如，在文字主控台中手動磁碟分割時需要使用此選項。

刪除分割區表和磁碟標籤
: 這樣會完全覆寫舊的分割區表。例如，如果使用不常見的磁碟標籤時出現問題，這個指令很有用。使用此方法，會遺失硬碟上所有資料。

更多磁碟分割提示

下一節包含一些關於磁碟分割的提示和祕訣，當您安裝系統時，這些資訊可以協助您做出正確的決定。

提示：磁柱編號

請注意，不同的磁碟分割工具會從 `0` 或 `1` 開始統計分割區的磁柱數目。在計算磁柱的數目時，應該總是將最後一個磁柱號與第一個磁柱號相減再加一。

如果分割是由 YaST 執行，而且系統中偵測到其他分割區，這些分割區也會新增至檔案 /etc/fstab，以方便存取這項資料。這個檔案包含系統中的所有分割區及其屬性，像是檔案系統、裝載點和使用者權限。

範例 8.1 */etc/fstab：分割區資料*

```
/dev/sda1     /data1     auto       noauto,user 0 0
/dev/sda5     /data2     auto       noauto,user 0 0
/dev/sda6     /data3     auto       noauto,user 0 0
```

無論是 Linux 或 FAT 分割區都帶有指定選項 noauto 和 user。這讓任何使用者都可依需要裝載或解除裝載這些分割區。為了安全理由，YaST 不會自動輸入 exec 選項，該選項是執行程式時所必需。不過，若要由此執行程式，您可以手動輸入這個選項。如果您遇到系統訊息，例如「解譯器不良」或「權限遭拒」，就需要使用這個方法。

磁碟分割與 LVM

在進階分割程式中，使用「*LVM*」存取 LVM 組態 (請參閱第 7.1 節「LVM 組態」[103頁])。不過，如果使用的 LVM 組態已經存在於系統上，只要您第一次在此區段中輸入 LVM 組態，就會自動啟用。在此情況下，磁碟的分割區若是屬於啟用中的磁碟區群組，就無法重新分割，因為當磁碟分割區正在使用時，Linux 核心就無法重新讀取這個硬碟的修改分割區表格。不過，如果系統已經有可以運作的 LVM 組態，就不需要進行實體重新分割。而是變更邏輯磁碟區的組態即可。

在實體磁碟區 (PV) 的開頭，有關磁碟區的資訊會寫入分割區。若要為了其他非 LVM 的使用而要重新使用這類分割區，3\u204 I|n±N|\u185 1\u168 ··÷¥Ua\u186 o\u182 ?\AY刪豹u163 αu161。例如，在 VG system 與 PV /dev/sda2 中，使用 ddif=/dev/zero of=/dev/sda2 bs=512 count=1 指令，即可完成這個動作。

警告：開機的檔案系統

開機的檔案系統 (root 檔案系統或 /boot) 不得儲存在 LVM 邏輯磁碟區中。請另外儲存在一般的實體分割區中。

8.5.8 PCI 設備驅動程式

提示：IBM System z：繼續

若為 IBM System z，請繼續第 8.5.12 節「系統服務 (Runlevel)」[148頁]。

每個核心驅動程式都包含有其支援的設備ID清單。如果新設備沒有出現在任何驅動程式資料庫中，該設備就不會受到支援，即使它可以搭配現有驅動程式使用。您可以使用「*系統*」區段的 YaST 模組來新增 PCI ID。只有進階使用者才應該嘗試使用這個 YaST 模組。

圖形 8.7 *新增 PCI ID*

若要新增 ID，請按一下「*新增*」並選取其指定方式：在清單中選取 PCI 設備，或手動輸入 PCI 值。使用第一個選項時，從提供清單中選取 PCI 設備，然後輸入驅動程式或目錄的名稱。如果沒有填入任何目錄，驅動程式名稱就會當作目錄名稱使用。手動指定 PCI ID 值時，請輸入適當的資料來設定 PCI ID。按一下「*確定*」儲存變更。

若要編輯 PCI ID，請從清單中選擇設備驅動程式，然後按一下「*編輯*」。編輯其中資訊，然後按一下「*確定*」，儲存此項變更。若要刪除 ID，請選取該驅動程式，然後按一下「*刪除*」。此ID就會立刻從清單中消失。當完成時，按一下「*確定*」。

8.5.9 電源管理

您可以使用「*系統*」>「*電源管理*」模組來使用省電技術。這項技術對延長筆記型電腦的作業時間來說，更是特別重要。如需取得有關使用這項模組的詳細資訊，請參閱第 28.6 節「YaST電源管理模組」[479頁]。

8.5.10 Powertweak 組態

Powertweak 是 SUSE Linux 公用程式，可用來將您的系統調整到最佳效能，其方式為微調核心與硬體組態。只有進階使用者才能使用此公用程式。在使用「*系統*」>「*Powertweak*」進行啟動之後，它會偵測您的系統設定，並在模組左窗格中以樹狀形式列出。您也可以使用「*搜尋*」來尋找組態變數。選取要調整的選項並將其顯示在螢幕上，其中會包含其目錄和設定。若要儲存設定，請按一下「*完成*」，然後按一下「*確定*」進行確認。

8.5.11 設定檔管理員

使用「*系統*」>「*設定檔管理*」和 YaST 系統組態設定檔管理 (SCPM) 模組，來建立、管理和切換系統組態。這對於可在不同位置 (不同網路中)，並可由不同使用者使用的行動電腦特別有幫助。然而，即使是固定不動的電腦，此功能也有所幫助，因為它可以使用不同的硬體元件或測試組態。

8.5.12 系統服務 (Runlevel)

使用「*系統*」>「*系統服務 (runlevel)*」，設定 runlevel 和在其中啟動的服務。如需 SUSE Linux Enterprise 中 runlevel 的詳細資訊與 YaST runlevel 編輯器的說明，請參閱第 20.2.3 節「使用 YaST 設定系統服務 (Runlevel)」[367頁]。

8.5.13 /etc/sysconfig 編輯器

`/etc/sysconfig` 目錄包含的檔案具有 SUSE Linux Enterprise 最重要的設定。使用「*系統*」>「*/etc/sysconfig 編輯器*」來修改這些值，並將其儲存到個別的組態檔案中。一般而言，不需要進行手動編輯，因為在安裝套件或設定服務時，就會自動調整檔案。如需 `/etc/sysconfig` 與 YaST sysconfig 編輯器的詳細資訊，請參閱第 20.3.1 節「使用 YaST Sysconfig 編輯器變更系統組態」[368頁]。

8.5.14 日期和時間設定

安裝過程中已完成時區的初始設定，不過您可以使用「*系統*」>「*日期和時間*」來進行變更。這也可以用來變更目前的系統日期和時間。

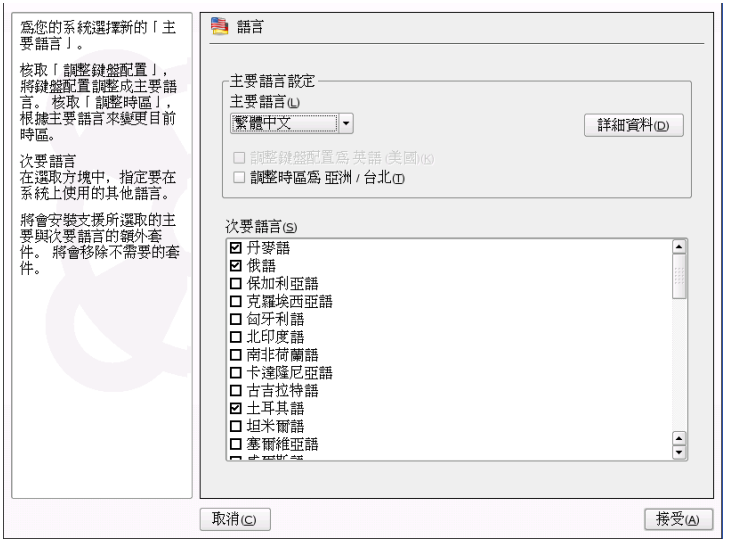
若要變更時區，請在左欄中選取地區，並在右欄中選取位置或時區。使用「*硬體時鐘設為*」，設定系統時鐘要使用「*本地時間*」或「*UTC*」(國際標準時間)。「*UTC*」通常用於 Linux 系統中，而其他作業系統如 Microsoft Windows 等，大部分使用本地時間。

使用「*變更*」，設定目前的系統時間和日期。在開啟的對話方塊中輸入新值來修改時間和日期，或是使用箭頭按鈕進行調整。按「*套用*」儲存變更。

8.5.15 語言選擇

系統的主要與次要語言會在安裝時設定。然而，您可隨時使用「*系統*」>「*語言*」進行變更。在 YaST 中設定的主要語言會套用到整個系統，包括 YaST 與桌面環境。這是您希望在大部分時間使用的語言。次要語言是使用者偶而因其他目的而需要用到的語言，例如桌面語言或文字處理。

圖形 8.8　設定語言



在「*主要語言*」中選擇系統所使用的主要語言。若要將鍵盤或時區調整成此設定，請啟用「*調整鍵盤配置*」或「*調整時區*」。

使用「*詳細資料*」設定 `root` 使用者之地區設定變數的設定方式。您也可以使用「*詳細資料*」，將主要語言設定為不在主要清單上的方言。此設定會寫入 `/etc/sysconfig/language` 檔案中。

8.6 網路設備

系統連接的所有網路設備必須在服務使用它們之前起始。這些設備的偵測和設定作業會在模組群組「*網路設備*」中完成。

8.6.1 DSL、ISDN、數據機或網路卡

若要設定 DSL、ISDN 或者是數據機或網路卡，請從「*網路設備*」畫面選取適用的模組。請在清單中選取自動偵測的設備，然後按一下「*編輯*」。如果未偵測到您的設備，請按一下「*新增*」，並以手動方式選取。若要編輯現有設備，請選取它再按一下「*編輯*」。如需更詳細資訊，請參閱第 30.4 節「使用 YaST 手動設定網路連線」[512頁]。如需關於無線網路介面的資訊，請參閱第 29 章「*無線通訊*」[485頁]。

提示：CDMA 和 GPRS 數據機

您可以將支援的 CDMA 和 GPRS 數據機設定為 YaST 數據機模組中的一般數據機。

8.7 網路服務

此群組包含一些工具，用來設定網路中所有類型的服務。這些工具包含名稱解析、使用者驗證以及檔案服務。

8.7.1 郵件轉送代辦程式

如果您是使用 sendmail、Postfix 或提供者的 SMTP 伺服器來傳送電子郵件，請在「*網路服務*」>「*郵件轉送代辦程式*」設定您的郵件設定。您可以透過 fetchmail 程式取得郵件，也可以輸入提供者之 POP3 或 IMAP 伺服器的詳細資訊。或是，使用您選擇的郵件程式，例如 KMail 或 Evolution，來設定存取資料。在此狀況下，您不需要此模組。

若要使用 YaST 設定您的郵件，請在第一個對話方塊中指定用來連接到網際網路的連線類型。選擇下列其中一個選項：

「*永久*」
: 如果您已有連到網際網路的專屬線路，請選擇此選項。您的電腦永久與網際網路連線，因此不需要撥號連接。如果您的系統位於使用中央電子郵件伺服器的本地網路中，請選擇此選項以確保可永久存取您的電子郵件。

「*撥號連接*」
: 有電腦在家中、電腦不在網路中，而且偶爾才會連接到網際網路的使用者才適合使用此項目。

「*無連線*」
: 如果您無法存取網際網路，而且電腦不在網路中，您就無法傳送或接收電子郵件。

選取該選項，以透過 AMaViS 在內送與外送的電子郵件中執行病毒掃描功能。該套件會在您啟用郵件過濾功能之後，立即自動安裝。請在下列對話方塊中，指定外送郵件伺服器 (通常是提供者的 SMTP 伺服器)，以及內送郵件的參數。根據不同的使用者接收郵件情形，指定不同的 POP 或 IMAP 伺服器。利用此對話方塊，您也可以指定別名、使用偽裝，或設定虛擬領域。按一下「*完成*」來結束郵件組態。

8.7.2 郵件伺服器

重要：以 LDAP 為基礎的郵件伺服器組態

只有當使用者、群組以及 DNS 與 DHCP 服務都利用 LDAP 進行管理時，SUSE Linux Enterprise 的郵件伺服器模組才有作用。

您可以使用郵件伺服器模組，將 SUSE Linux Enterprise 設定成郵件伺服器。YaST 會協助您完成下列設定程序的步驟：

全域設定
: 設定本地郵件伺服器的識別和內送或外送郵件大小上限，以及郵件傳輸類型。

本地傳送
: 設定本地郵件傳送類型。

郵件傳輸
: 根據郵件的目標位址來設定郵件的特殊傳輸路由。

預防垃圾郵件
: 設定郵件伺服器的垃圾郵件保護設定。如此將啟用 AMaViS 工具。設定 SPAM 檢查的類型和嚴格程度。

郵件伺服器轉送
: 決定郵件伺服器不可用來傳送來自哪個網路的非本地郵件。

取得郵件
: 透過不同的通訊協定，從外部郵件帳號收取郵件的設定。

郵件伺服器領域
: 這可決定郵件伺服器所應負責的領域。如果伺服器不應該當作專門用來傳送而不接收郵件的 Null 用戶端，則至少必須設定一個主要領域。

  區分三種領域類型：

  主要
  : 本地郵件伺服器的主要領域

  本地
  : 在主要領域中可接收郵件的所有使用者也可以在本地領域中接收郵件。就本地領域中的郵件而言，只會評估 @ 之前的部分。

  虛擬
  : 在虛擬領域中，只有擁有明確位址的使用者才能夠接收郵件。虛擬郵件位址可在 YaST 的使用者管理模組中設定。

8.7.3 其他可用的服務

YaST「*網路服務*」中有很多其他的網路模組。

DHCP 伺服器
: 您只需要幾個步驟，就能夠完成設定自訂的 DHCP 伺服器。第 34 章「*DHCP*」[581頁]會提供該主題的基本知識，還會逐步說明組態程序。

DNS 伺服器
: 建議您在大型網路上設定負責進行名稱解析的 DNS 伺服器組態。您可以在其中使用「*DNS 伺服器*」，詳細資訊請參閱第 33.2 節「使用 YaST 進行設定」[560頁]。第 33 章「*網域名稱系統*」[559頁]會提供關於 DNS 的背景資訊。

DNS 和主機名稱
: 如果設定網路設備時沒有進行這些設定，請使用此模組來設定主機名稱和 DNS。此模組也可以用來變更主機名稱與領域名稱。如果提供者已經正確設定 DSL、數據機或 ISDN 存取，從提供者資料擷取出的項目，就會包含在名稱服務清單上。如果您處於本地網路中，可能就會透過 DHCP 收到您的主機名稱，若為此狀況，則不應修改該名稱。

HTTP 伺服器
: 若要執行您自己的網頁伺服器，請在「*HTTP 伺服器*」中設定 Apache。如需更多詳細資訊，請參閱第 40 章「*Apache HTTP 伺服器*」[673頁]。

主機名稱
: 當您在小型網路中進行開機時，請使用「*主機名稱*」(而不是 DNS) 來進行主機名稱解析。在此模組中的項目會反映 `/etc/hosts` 檔案的資料。如需更多資訊，請參閱章節「 `/etc/hosts` 」[536頁]。

Kerberos 用戶端
: 如果您的網路是用 Kerberos 伺服器來進行網路驗證，請使用「*Kerberos 用戶端*」。如需使用 YaST 處理用戶端組態的詳細說明，請參閱第 46.6 節「以 YaST 設定 Kerberos 用戶端」[774頁]。

LDAP 用戶端
: 如果要在網路中使用 LDAP 進行使用者驗證，可在「*LDAP 用戶端*」中設定用戶端。如需 LDAP 的資訊以及使用 YaST 來設定用戶端的詳細資訊，請參閱第 36.6 節「使用 YaST 設定 LDAP 用戶端」[623頁]。

LDAP 伺服器
: LDAP 伺服器可先將各種資料都儲存到中央目錄中，再從此處將資料配送給網路中的所有用戶端。它多半是用來儲存共用的聯絡人資訊，但是所提供的功能不止於此。LDAP 伺服器也可用於執行驗證。如需 LDAP 的資訊以及使用 YaST 設定伺服器的詳細資訊，請參閱第 36 章「*LDAP—一種目錄服務*」[605頁]。

NFS 用戶端
: 使用 NFS 用戶端，將 NFS 伺服器提供的目錄裝載到您自己的檔案目錄樹中。使用「*NFS 用戶端*」，將您的系統設定成可在網路中存取 NFS 伺服器。

NFS 伺服器
: 利用 NFS，您可以執行網路上的所有成員皆能存取的檔案伺服器。此檔案伺服器可讓使用者能夠使用特定應用程式、檔案與儲存空間。在「*NFS 伺服器*」中，您可以將主機設定為 NFS 伺服器，並決定要輸出的目錄，以供網路使用者用於一般用途。所有具有適當權限的使用者，都可以在他們自己的檔案樹中裝載這些目錄。如需 YaST 模組的說明以及有關 NFS 的背景資訊，請參閱第 38 章「*使用 NFS 共享檔案系統*」[651頁]。

NIS 用戶端
: 如果您要執行 NIS 伺服器，從中央位置管理使用者資料，並將資料配送給用戶端，請在這裡設定該用戶端。如需有關 NIS 用戶端與使用 YaST 設定其組態的詳細資訊，請參閱第 35.2 節「設定 NIS 用戶端」[603頁]。

NIS 伺服器
: 如果有一個以上的系統，由本地使用者進行管理 (使用 `/etc/passwd` 與 `/etc/shadow` 檔案) 就不夠現實，需要花費許多心力維護。在此情況中，使用者資料會在中央伺服器上進行管理，再從中配送給用戶端。NIS 是其中一種選項。如需有關 NIS 與使用 YaST 設定其組態的詳細資訊，請參閱第 35.1.1 節「設定主要 NIS 伺服器」[598頁]。

NTP 用戶端
: NTP (網路時間協定) 這項協定可用來同步化網路上的硬體時鐘。如需 NTP 的相關資訊以及使用 YaST 設定其組態的說明，請參閱第 32 章「*使用 NTP 進行時間同步化*」[553頁]。

網路服務 (xinetd)
: 設定當 SUSE Linux Enterprise 使用「*網路服務*」開機時所要啟動的網路服務 (例如 finger、talk 與 ftp)。這些服務可讓外部主機連接到您的電腦。每個服

務都可以設定不同的參數。根據預設，並不會啟動管理個別服務 (inetd 或 xinetd) 的主要服務。

此模組啟動之後，請選擇要啟動 inetd 或 xinetd。選擇的精靈會以標準的服務選擇來啟動。或者，您可以利用「*新增*」、「*刪除*」與「*編輯*」來編撰您自己的服務選擇。

警告：設定網路服務 (xinetd)

在系統上撰寫與調整網路服務的程序複雜，必須要對 Linux 服務的概念有廣泛的瞭解。預設設定通常都已足夠。

代理

在「*Proxy*」中設定網際網路代理用戶端設定。按一下「*啟用 Proxy*」，然後輸入想要的代理設定。您可按一下「*測試代理設定*」，測試這些設定。這時會出現一個小視窗，通知您的代理設定是否正常運作。輸入且測試設定之後，請按一下「*接受*」儲存設定。

遠端管理

若要從其他機器遠端管理您的機器，請使用「*遠端管理*」。若要遠端維護您的系統，請使用 VNC 用戶端，如 krdc 或有 Java 功能的瀏覽器。雖然使用 VNC 進行遠端管理非常簡單，但卻比使用 SSH 不安全，使用 VNC 伺服器時必須隨時將此謹記在心。如需關於使用 VNC 用戶端安裝的詳細資訊，請參閱第 4.1.1 節「透過 VNC 進行的簡易遠端安裝—靜態網路組態」[44 頁]。

您可以在「*遠端管理設定*」中選取「*允許遠端管理*」允許遠端管理。選取「*不允許遠端管理*」會停用此功能。按一下「*開啟防火牆中的連接埠*」，允許存取您的電腦。按一下「*防火牆詳細資訊*」，會以防火牆中開啟的連接埠顯示網路介面。選擇希望的介面，並按一下「*確定*」返回主對話方塊。請按一下「*接受*」完成設定。

強烈建議您使用 YaST「*遠端管理*」模組在機器上設定 VNC。雖然 YaST 介面亦允許您設定遠端存取內容，但這並無法取代 YaST。這只能讓您將 X 伺服器設定為 VNC 工作階段的主機。

路由

使用「*路由*」來設定管理網路的路徑資料。在大部分的情況下，只需輸入系統 IP 位址，再透過此位址從「*預設閘道*」中傳送所有資料。若要建立更複雜的組態，請使用「*進階組態*」。

Samba 伺服器
: 在 Linux 與 Windows 主機所組成的異質網路中，Samba 可控制兩方之間的通訊。 如需有關 Samba 以及設定這些用戶端的詳細資訊，請參閱第 37 章「*Samba*」[635頁]。

SLP 伺服器
: 使用*服務位址通訊協定*(SLP)時，您可以在未事先知道伺服器名稱以及這些伺服器提供的服務情況下，在網路中設定用戶端。如需有關 SLP 伺服器和使用 YaST 設定其組態的詳細資訊，請參閱第 31 章「*網路中的 SLP 服務*」[549頁]。

TFTP 伺服器
: TFTP 伺服器並不是 FTP 伺服器。FTP 伺服器使用的是檔案傳輸通訊協定(FTP)，而 TFTP 伺服器則使用更為簡單但不提供安全性功能的簡單式檔案傳輸通訊協定 (TFTP)。TFTP 伺服器通常用來開機無磁片的工作站、X 終端機以及路由器。如需有關 TFTP 伺服器和使用 YaST 設定其組態的詳細資訊，請參閱第 4.3.2 節「設定 TFTP 伺服器」[61頁]。

WOL
: WOL(網路喚醒功能)表示可使用特殊套件透過網路將電腦從待命模式喚醒。它只可與 BIOS 中支援此功能的主機板搭配使用。如需有關使用 YaST 設定 WOL 的詳細資訊，請參閱第 4.3.7 節「區域網路喚醒」[68頁]。

Windows 網域成員
: 在 Linux 與 Windows 主機所組成的異質網路中，Samba 可控制兩方之間的通訊。使用「*Samba 用戶端*」模組時，您可以將電腦設定成 Windows 領域的成員。如需有關 Samba 與其用戶端設定的詳細資訊，請參閱第 37 章「*Samba*」[635頁]。

iSCSI 目標
: iSCSI 技術提供了一種將 Linux 電腦與中央儲存系統相連接的簡單且低廉的解決方案。若要設定伺服器端，請使用「*其他*」>「*iSCSI 目標*」。如需有關使用 YaST 來設定 iSCSI 的詳細資訊，請參閱第 12 章「*IP 網路 — iSCSI 上的大型存放設備* 」[247頁]。

iSCSI 啟動器
: 若要設定與中央儲存系統的連接，請使用「*其他*」>「*iSCSI 啟動器*」。如需有關使用 YaST 來設定 iSCSI 的詳細資訊，請參閱第 12 章「*IP 網路 — iSCSI 上的大型存放設備* 」[247頁]。

8.8 AppArmor

Novell AppArmor 可為伺服器和工作站提供容易使用的應用程式安全性。Novell AppArmor是一種存取控制系統，可讓您指定每個程式可讀取、寫入和執行哪些檔案。若要啟用或停用系統上的 Novell AppArmor，請使用「*AppArmor 控制台*」。如需 Novell AppArmor 的資訊以及使用 YaST 來設定用戶端的詳細資訊，請參閱*Novell AppArmor Administration Guide* (↑*Novell AppArmor Administration Guide*)。

8.9 安全性與使用者

多重使用者是Linux的基本功能。因此，多位使用者可以在相同的Linux系統上獨立進行工作。登入名稱與個人密碼可用來識別每位使用者的使用者帳戶，以供登入系統使用。所有使用者都會擁有自己的主目錄，用以儲存個人檔案與組態。

8.9.1 使用者管理

使用「*安全性與使用者*」>「*使用者管理*」來建立和編輯使用者。它提供系統中所有使用者的綜覽，包括 NIS、LDAP、Samba 和 Kerberos 使用者 (若有需要)。如果您屬於大型網路中的一部分，請按一下「*設定過濾器*」來依照類別列出所有的使用者。您也可按一下「*自定過濾器*」來自定過濾器設定。

提示：不關閉模組而套用組態變更

如果您需要進行多項組態變更，但您不想在每一次變更時都重新啟動使用者和群組的組態模組，則使用「*立即寫入變更*」可在不離開組態模組的情況下儲存您的變更。

新增使用者

若要新增使用者，請按照下列步驟進行：

1 按一下「*新增*」。

2 請輸入「*使用者資料*」的必要資料。如果您已不需為這個新使用者調整任何詳細設定，則請繼續前往步驟 5 [158頁]。

3 若要變更使用者的 ID、主目錄名稱、預設主目錄、群組、群組成員、目錄許可權或登入外圍程序，請開啟「*詳細資料*」索引標籤並變更預設的值。

4 若要調整使用者的密碼期限、長度和過期警告，請使用「*密碼設定*」索引標籤。

5 請按一下「*接受*」來撰寫使用者帳戶組態。

新使用者可使用新建立的登入名稱和密碼立即登入。

刪除使用者

若要刪除使用者，請按照下列步驟進行：

1 在清單中選擇使用者。

2 按一下「*刪除*」。

3 請決定是否要刪除使用者的主目錄，或先予以保留而在日後刪除。

4 按一下「*是*」套用設定。

變更登入組態

若要變更登入組態，請按照下列步驟進行：

1 在清單中選擇使用者。

2 按一下「*編輯*」。

3 調整「*使用者資料*」、「*詳細資料*」和「*密碼設定*」之下的設定。

4 請按一下「*接受*」來儲存使用者帳戶組態。

管理加密的主目錄

在使用者帳戶的建立程序中，您可以建立加密的主目錄。若要為使用者建立加密的主目錄，請按照下列步驟進行：

1. 按一下「*新增*」。
2. 請輸入「*使用者資料*」的必要資料。
3. 啟用「*詳細資料*」索引標籤中的「*使用加密的主目錄*」。
4. 選擇「*接受*」套用您的設定。

若要為現有的使用者建立加密的主目錄，請按照下列步驟進行：

1. 請在清單中選取使用者，然後按一下「*編輯*」。
2. 啟用「*詳細資料*」索引標籤中的「*使用加密的主目錄*」。
3. 輸入選定使用者的密碼。
4. 選擇「*接受*」套用您的設定。

若要停用主目錄的加密，請按照下列步驟進行：

1. 請在清單中選取使用者，然後按一下「*編輯*」。
2. 停用「*詳細資料*」索引標籤中的「*使用加密的主目錄*」。
3. 輸入選定使用者的密碼。
4. 選擇「*接受*」套用您的設定。

如需更多關於加密主目錄的資訊，請參閱第 47.2 節「使用加密主目錄」[788頁]。

自動登入

警告：使用自動登入功能

若在可讓多人實際存取的系統上使用自動登入功能，對安全性有潛在的風險。所有存取此系統的使用者都能操縱上面的資料。如果您的系統含有機密資料，則請勿使用自動登入功能。

如果您是系統的唯一使用者，則可以使用自動登入功能。此功能可讓使用者在系統啟動後自動登入。只有一個選定的使用者可以使用自動登入功能。自動登入功能只適用於 KDM 和 GDM。

若要啟用自動登入功能，請在使用者清單中選取使用者，然後按一下「*專家選項*」>「*登入設定*」。接著請選擇「*自動登入*」，再按一下「*確定*」。

若要停用此功能，請選取使用者並按一下「*專家選項*」>「*登入設定*」。接著請取消勾選「*自動登入*」，再按一下「*確定*」。

不以密碼登入

警告：允許不以密碼登入

若在可讓多人實際存取的系統上使用不以密碼登入的功能，對安全性有潛在的風險。所有存取此系統的使用者都能操縱上面的資料。如果您的系統含有機密資料，則請勿使用此功能。

當使用者在登入管理員中輸入使用者名稱時，這個不以密碼登入的功能會自動將進入系統的使用者記錄下來。這個功能可讓系統上的多個使用者使用，只適用於 KDM 或 GDM。

若要啟用此功能，請在使用者清單中選取使用者，然後按一下「*專家選項*」>「*登入設定*」。接著請選擇「*不以密碼登入*」，再按一下「*確定*」。

若要停用此功能，請在使用者清單中選取要停用此功能的使用者，然後按一下「*專家選項*」>「*登入設定*」。接著請取消勾選「*不以密碼登入*」，再按一下「*確定*」。

停用使用者登入

若想建立一個無法登入系統但可管理某些系統相關任務的系統使用者身份，請在建立使用者帳戶時停用使用者登入功能。請執行下列步驟：

1. 按一下「*新增*」。
2. 請輸入「*使用者資料*」的必要資料。
3. 請勾選「*停用使用者登入*」。
4. 選擇「*接受*」套用您的設定。

若要停用現有使用者的登入，請按照下列步驟進行：

1. 請在清單中選取使用者，然後按一下「*編輯*」。
2. 請勾選「*使用者資料*」中的「*停用使用者登入*」。
3. 選擇「*接受*」套用您的設定。

強制執行密碼規則

在任何擁有多個使用者的系統上，若能至少強制執行基本的密碼安全性規則，是個不錯的作法。使用者應定期變更密碼，並應使用增強式密碼，因為此種密碼無法輕易破解。如需有關如何強制執行較嚴厲的密碼規則，請參閱第 8.9.3 節「本地安全性」[164頁]。若要強制執行密碼的更換，請建立密碼過期規則。

若要為新使用者設定密碼過期規則，請按照下列步驟進行：

1. 按一下「*新增*」。
2. 請在「*使用者資料*」輸入必要資料。
3. 請調整「*密碼設定*」中的值。
4. 選擇「*接受*」套用您的設定。

若要為現有的使用者變更密碼過期規則，請按照下列步驟進行：

1 請在清單中選取使用者，然後按一下「*編輯*」。

2 請調整「*密碼設定*」中的值。

3 選擇「*接受*」套用您的設定。

您可以為特定的使用者帳戶指定一個過期日，藉以限制該帳戶的生命期。請以 *YYYY-MM-DD* 的格式指定「*過期日*」並填入使用者的組態資訊。若未提供「*過期日*」，使用者帳戶就不會過期。

變更新使用者的預設設定

建立新的本地使用者時，YaST 會使用幾個預設設定。您可以變更這些預設設定來符合自身需求。

1 請選取「*專家選項*」>「*新使用者的預設設定*」。

2 請對以下的任何項目套用您的變更：

- 「*預設群組*」
- 「*次要群組*」
- 「*預設登入外圍程序*」
- 「*主目錄的路徑字首*」
- 「*主目錄的基本架構*」
- 「*主目錄的 umask*」
- 「*預設過期日*」
- 「*密碼過期後仍可登入的天數*」

3 選擇「*接受*」套用您的變更。

有幾個其他與安全性有關的預設設定也可以使用「*本地安全性*」進行變更。請參閱第 8.9.3 節「本地安全性」 [164頁]以取得更多資訊。

變更密碼的加密

注

密碼加密的變更只適用於本地使用者。

SUSE Linux Enterprise 可以使用 DES、MD5 或 Blowfish 來進行密碼的加密，而 Blowfish 是預設的密碼加密方法。加密方法是在系統安裝期間所設定的，如第 3.14.1 節「系統管理員「root」的密碼」[34頁]所述。若要變更已安裝系統的密碼加密方法，請選取「*專家選項*」>「*密碼加密*」。

變更認證和使用者來源

使用者管理方法 (例如 NIS、LDAP、Kerberos 或 Samba) 是在安裝期間設定的，如第 3.14.7 節「使用者」[40頁]所述。若要變更已安裝系統的使用者認證法，請選取「*專家選項*」>「*認證和使用者來源*」。此模組可提供組態綜覽以及用戶端的設定選項。您也可使用此模組來進行進階用戶端設定。

8.9.2 群組管理

若要建立和編輯群組，請選取「*安全性與使用者*」>「*群組管理*」，或在使用者管理模組中按一下「*群組*」。這兩個對話方塊擁有相同的功能，都可讓您建立、編輯或刪除群組。

模組可提供所有群組的綜覽。就像使用者管理對話方塊，只要按一下「*設定過濾器*」就可以變更過濾器設定。

若要新增群組，請按一下「*新增*」，並輸入適當的資料。您可選取對應方塊，從清單中選取群組成員。按一下「*接受*」建立群組。若要編輯群組，請從清單中選擇要編輯的群組，並按一下「*編輯*」。進行所有必要的變更，並使用「*接受*」儲存變更。若要刪除群組，只需從清單中選擇，然後按一下「*刪除*」。

按一下「*進階選項*」，進行進階的群組管理。如需這些選項的詳細資訊，請參閱第 8.9.1 節「使用者管理」[157頁]。

8.9.3 本地安全性

若要在整個系統中套用一組安全性設定，請使用「*安全性與使用者*」>「*本地安全性*」。這些設定包含開機、登入、密碼、使用者建立和檔案許可權的安全性。SUSE Linux Enterprise 提供三種預先設定的安全性組合：「*主工作站*」、「*網路工作站*」和「*網路伺服器*」。使用「*詳細資料*」來修改預設值。若要建立您自己的配置，請使用「*自定設定*」。

詳細或自定設定包含如下：

「*密碼設定*」
若要在接受新的密碼之前讓系統檢查密碼安全性，請按一下「*檢查新密碼*」與「*測試複雜密碼*」。設定新建使用者的密碼長度下限。定義密碼的有效期間、以及應該在到期前幾天內，於該使用者登入文字主控台時就發出警示。

「*開機設定*」
設定選擇所需的動作來定義按鍵組合 Ctrl + Alt + Del。通常在文字主控台中輸入此組合，就會讓系統重新開機。除非您的電腦或伺服器可供公用存取，而且您擔心會有人未經過授權就執行此動作，否則請不要修改此設定。如果選擇「*停止*」，此按鍵組合就會使系統關機。使用「*忽略*」，則會忽略此按鍵組合。

如果您使用 KDE 登入管理員 (KDM)，請在「*KDM 的關機行為*」中設定關閉系統的權限。可將許可權授予「*只有 root*」(系統管理員)、「*所有使用者*」、「*無人*」或「*本地使用者*」。如果選擇「*無人*」，系統就只能透過文字主控台來關閉。

「*登入設定*」
一般情況下，登入失敗之後會先等待數秒，然後才能進行另一次登入。如此可讓密碼監聽程式 (sniffer) 不容易登入。可選擇性啟用「*記錄成功登入次數*」。如果您懷疑有人試圖要探查您的密碼時，請在 `/var/log` 中檢查系統記錄檔中的項目。啟用「*允許遠端圖形登入*」，允許其他使用者透過網路存取圖形登入畫面。因為此存取方式有潛在的安全性風險，因此預設會關閉該功能。

「*使用者新增*」
每位使用者都擁有數值與字母混合的使用者 ID。這些資料之間的關聯是使用 `/etc/passwd` 檔案建立，而且應該是唯一專屬資訊。使用此畫面中的資料，可在新增使用者時，針對要指定給使用者 ID 的數值部分來定義數字

範圍。使用者適用的下限為 500。自動產生的系統使用者會從 1000 開始。請以群組 ID 設定的相同步驟繼續。

「*其他設定*」
若要使用預先定義的檔案權限設定，請選取「*簡易*」、「*安全*」或「*Paranoid*」。「*簡易*」選項對於大部分使用者而言應已足夠。「*Paranoid*」設定相當嚴格，而且可作為自定設定的作業基礎。請記得，如果選取「*Paranoid*」，有些程式可能就無法運作或無法正確運作，因為使用者可能已經不具備存取特定檔案的權限。

您也可以定義哪些使用者可以在安裝後啟動 `updatedb` 程式。此程式每天都會自動執行，或是在開機後執行，您電腦上每個檔案的儲存位置都會包含在所產生的資料庫 (locatedb) 內。如果選擇「*無人*」，則使用者都只能夠在資料庫中，找到其他 (未經授權) 使用者都能看到的路徑。如果選擇 `root`，則會製作所有本地檔案的索引，因為 `root` 使用者是超級使用者，可以存取所有目錄。確認已停用「*根路徑中目前的目錄*」和「*一般使用者路徑中目前的目錄*」。只有進階使用者才應考慮使用這些選項，因為若使用錯誤的話，這些設定可能導致明顯的安全性風險。即使系統損毀後仍想擁有系統的部分控制權時，請按一下「*開啟魔術 SysRq 鑰匙*」。

按一下「*完成*」，完成安全性組態。

8.9.4 憑證管理

憑證可供通訊使用，也可應用在公司ID卡等設備中。若要管理憑證或是輸入一般伺服器憑證，請使用「*安全性與使用者*」>「*CA 管理*」。如需有關憑證、憑證使用技術以及使用 YaST 進行管理的詳細資訊，請參閱第 42 章「*管理 X.509 憑證*」[731頁]。

8.9.5 防火牆

SUSEfirewall2 可保護您的電腦不會受到來自網際網路的攻擊。使用「*安全性與使用者*」>「*防火牆*」來設定。如需關於 SuSEfirewall2 的詳細資訊，請參閱第 43 章「*偽裝與防火牆*」[745頁]。

提示：自動啟用防火牆

YaST 會根據每個已設定的網路介面，以合適的設定自動啟動防火牆。如果想要以自定設定重新設定防火牆，或是停用防火牆功能，請僅啟動此模組。

8.10 虛擬化

虛擬化可讓您在一部實體電腦上執行多個作業系統。不同系統的硬體會以虛擬模式提供。虛擬化 YaST 模組可設定 Xen 虛擬化系統。如需此技術的詳細資訊，請參閱http://www.novell.com/documentation/sles10/index.html中的虛擬化手冊。

「*虛擬化*」區段提供了下列模組：

安裝虛擬機器管理者和工具
: 在使用 Xen 之前，請您先安裝附有 Xen 支援和相關工具的核心。若要安裝它們，請使用「*虛擬化*」>「*安裝虛擬機器管理者和工具*」。安裝之後，請重新啟動您的系統，進而使用 Xen 核心。

建立虛擬機器
: 成功安裝 Xen 虛擬機器管理者和工具之後，您就可以在虛擬伺服器上安裝虛擬機器。若要安裝虛擬機器，請使用「*虛擬化*」>「*建立虛擬機器*」。

8.11 其他

「YaST 控制中心」有數種無法簡單歸類到前六種模組群組中的模組。當您檢視記錄檔、以及從廠商提供光碟安裝驅動程式時，就可以使用這些模組。

8.11.1 建立自訂安裝 CD

使用「*其他*」>「*CD 建立程式*」，您可以從原始安裝設定建立自定安裝 CD。若要開始建立動作，請按一下「*新增*」。使用套件管理員來選擇套件或是 AutoYaST 控制檔案，以便使用預先設定的 AutoYaST 設定檔進行建立。

8.11.2 設定安裝伺服器

進行網路安裝時必須使用安裝伺服器。若要設定這類伺服器，請使用「*其他*」>「*安裝伺服器*」。如需有關使用 YaST 來設定安裝伺服器的詳細資訊，請參閱第 4.2.1 節「使用 YaST 設定安裝伺服器」[51頁]。

8.11.3 自動安裝

AutoYaST 工具的設計有利於自動安裝。在「*其他*」>「*自動安裝*」中，準備這項工具要使用的設定檔。如需有關使用 AutoYaST 進行自動安裝的詳細資訊，請參閱第 5 章「*自動安裝*」[77頁]。如需有關使用「*自動安裝*」模組的詳細資訊，請參閱第 5.1.1 節「建立 AutoYaST 設定檔」[78頁]。

8.11.4 支援查詢

「*其他*」>「*支援查詢*」可用來收集支援小組所需要的所有系統資訊，以便小組盡快找出您的問題，提供可解決問題的協助。請依據您的查詢，從隨後出現視窗中選取問題類別。在收集到所有資訊之後，將此份資訊連結到您的支援要求。

8.11.5 版本說明

版本說明為安裝、更新、組態和技術性議題的相關重要來源。這些版本說明會持續透過線上更新進行更新和發行。使用「*其他*」>「*版本說明*」來檢視版本說明。

8.11.6 開機記錄

您可以在「*其他*」>「*開機記錄*」中檢視電腦開機的相關資訊。當系統遇到問題、或在進行疑難排解時，您可能首先希望檢視這個記錄。它會顯示 `/var/log/boot.msg` 開機記錄，其中會包含電腦啟動時所顯示的畫面訊息。檢視這項記錄可協助您判斷電腦是否正確啟動，以及是否所有的服務和功能都正確啟動。

8.11.7 系統記錄

使用「*其他*」>「*系統記錄*」來檢視系統記錄，這項記錄會追蹤電腦操作、並記錄於 `var/log/messages`。核心訊息會根據日期與時間排序而記錄在此。使用上方的方塊來檢視特定系統元件的狀態。下列為系統記錄和開機記錄模組中的可能選項：

/var/log/messages
: 這是一般的系統記錄檔案。您可在此檢視核心訊息、使用者登入為 `root` 的情形，以及其他有用的資訊。

/proc/cpuinfo
: 這裡顯示處理器資訊，包括其類型、廠商、型號與效能。

/proc/dma
: 這裡顯示目前使用的 DMA 頻道。

/proc/interrupts
: 這裡顯示使用中的岔斷為何，以及使用中的數量。

/proc/iomem
: 這裡顯示輸入/輸出記憶體的狀態。

/proc/ioports
: 這裡顯示此時使用中的 I/O 埠。

/proc/meminfo
: 這裡顯示記憶體狀態。

/proc/modules
: 這裡顯示個別模組。

/proc/mounts
: 這裡顯示目前裝載的設備。

/proc/partitions
: 這裡顯示所有硬碟的分割區。

/proc/version
: 這裡顯示目前的 Linux 版本。

/var/log/YaST2/y2log
這裡顯示所有的 YaST 記錄訊息。

/var/log/boot.msg
這裡顯示關於啟動系統的資訊。

/var/log/faillog
這裡顯示登入失敗。

/var/log/warn
這裡顯示所有系統警告。

8.11.8 廠商驅動程式光碟

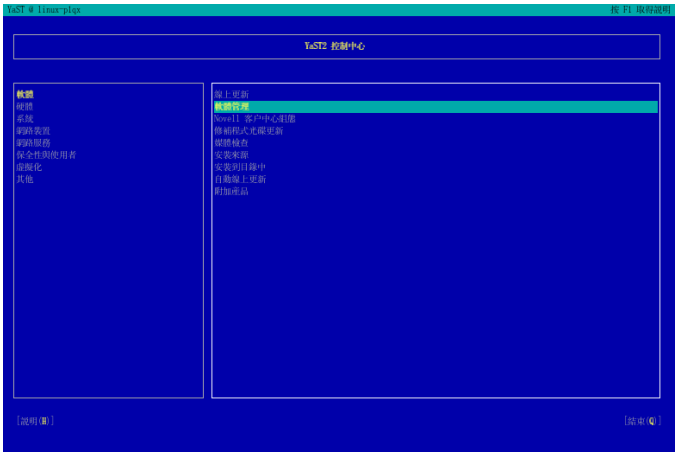
使用「*其他*」>「*廠商驅動程式光碟*」，從包含 SUSE Linux Enterprise 驅動程式的 Linux 驅動程式光碟安裝設備驅動程式。從頭開始安裝 SUSE Linux Enterprise 時，可在安裝之後，使用此 YaST 模組從廠商光碟載入所需的驅動程式。

8.12 文字模式的 YaST

本章節對象為未在其系統上執行 X 伺服器，且依賴以文字為基礎的安裝工具的系統管理員及進階使用者。這裡提供了一些基本資訊，說明如何在文字模式中啟動與操作 YaST。

在文字模式中啟動 YaST，會先出現 YaST 控制中心。請參閱圖形 8.9「文字模式中的 YaST 主視窗」[170頁]。主要視窗包含 3 個區域。由粗的白色框線所圍繞的左框架，內含一些不同模組所屬的類別。會以彩色背景來表示作用中的類別。由細的白色框線所圍繞的右框架，內含作用中類別的可用模組的綜覽。下方框架中有「*說明*」按鈕與「*結束*」按鈕。

圖形 8.9　文字模式中的 YaST 主視窗



啟動 YaST 控制中心時，會自動選取「*軟體*」類別。您可以使用 ↓ 與 ↑ 來變更類別。若要啟動所選取的類別中的某個模組，請按 →。此模組選項會加上一個粗的框線。您可以使用 ↓ 與 ↑ 以選取想要的模組。您可以按住方向鍵不放來捲動可用模組清單。選取模組後，模組標題會出現彩色背景，並在下方框架中顯示一段簡要說明。

您可以按 Enter 以啟動想要的模組。模組中的各個按鈕或選項欄位中，都有一個不同顏色的字母 (預設為黃色)。您可以使用 Alt + yellow_letter 組合鍵來直接選取按鈕，而毋須使用 Tab 來到達想到的地方。按 Alt + Q 或選取「*結束*」並按 Enter 來結束 YaST 控制中心。

8.12.1 在模組中瀏覽

以下對 YaST 模組的控制元件所做的說明，假設所有的功能鍵及 Alt 組合鍵都能作用，且未被指定不同的全域功能。如需有關可能的例外狀況的資訊，請參閱第 8.12.2 節「組合鍵的限制」 [172頁]。

在按鈕與選項清單中瀏覽
: 使用 Tab 與 Alt + Tab 或 Shift + Tab 以便在含有選項清單的按鈕與框架中進行瀏覽。

在選項清單中瀏覽

在作用中且包含選項清單的框架中，您可以使用方向鍵 (↑ 和 ↓) 以便在其中的個別元件之間進行瀏覽。如果框架中個別項目超出其寬度，您可以使用 Shift + →或Shift + ← 以水平方式向右捲動或向左捲動。或者使用 Ctrl + E 或 Ctrl + A。如果使用 → 或 ←，會導致作用中的框架或目前的選項清單變更，如同在控制中心內一樣，則您也可以使用此組合鍵。

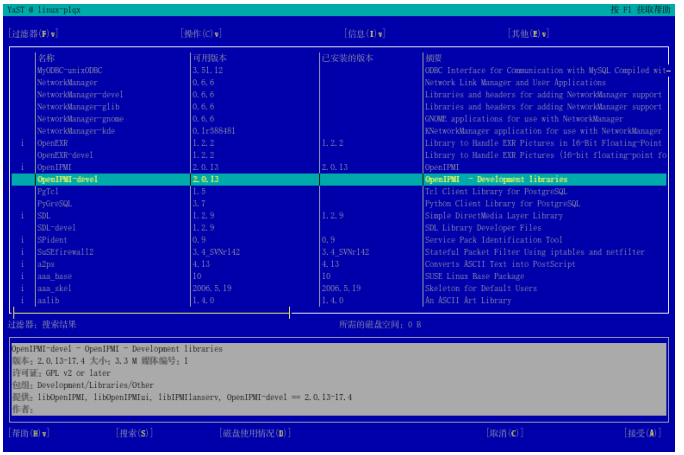
按鈕、圓形按鈕以及核取方塊

如果要選取有空白方括號 (核取方塊) 的按鈕，或是有空白括號 (選項圓鈕) 的按鈕，請按空格鍵或 Enter 鍵。或者，直接使用 Alt + yellow_letter 來選取選項圓鈕與核取方塊。在此狀況中，您不需按 Enter 來做確認。如果您使用 Tab 瀏覽至某個項目，按 Enter 即可執行所選取的動作或啟用個別的功能表項目。

功能鍵

F 鍵 (F1 到 F12) 可用來快速存取不同的按鈕。因為不同的模組提供不同的按鈕設定 (詳細資訊、資訊、新增、刪除等等)，所以各個功能鍵所實際對應的按鈕是依照作用中的 YaST 模組而定。您可以使用 F10 當作「*確定*」、「*下一步*」以及「*完成*」來使用。您可以按 F1 來存取 YaST 說明，該說明會顯示個別 F 鍵所對應的功能。

圖形 8.10 *軟體安裝模組*



8.12.2 組合鍵的限制

如果您的視窗管理員使用了全部的 Alt 組合，則 YaST 中的 Alt 組合可能無效。像是 Alt 或 Shift 等鍵也可能事先已由終端機的設定所佔用。

以 Esc 取代 Alt
: 您可以使用 Esc 來取代 Alt，以執行 Alt 捷徑。例如，Esc – H 可取代 Alt + H。(先按 Esc，*然後*按 H。)

您可以使用 Ctrl + F 與 Ctrl + B 來往前瀏覽和往後瀏覽。
: 如果 Alt 和 Shift 組合已先由視窗管理員或終端機所佔用，則請使用 Ctrl + F 組合 (往前) 與 Ctrl + B 組合 (往後) 來代替。

功能鍵的限制
: 有些功能會使用 F 鍵。有些特定的功能鍵可能已由終端機所佔用，無法供 YaST 使用。不過，在純文字主控台中，應該都可以使用各種的 Alt 組合鍵與功能鍵。

8.13 從指令行管理 YaST

當某任務只需進行一次時，圖形或 ncurses 介面通常是最好的解決方案。如果某任務必須重複進行，使用 YaST 指令行介面可能就比較簡單。自定程序檔也能使用此介面來進行任務的自動化。

使用 `yast -l` 或 `yast --list`，則可以檢視一個清單，其中包含您系統中所有可用的模組。若要顯示某模組的可用選項，請輸入 `yast` *`module_name`* `help`。如果某模組沒有指令行模式，就會發出訊息通知您。

若要顯示某模組的指令選項說明，請輸入 `yast` *`module_name command`* `help`。若要設定選項值，請輸入 `yast` *`module_name command option=value`*。

某些模組並不支援指令行模式，因為已經有指令行工具擁有相同的功能。相關的模組以及可用的指令行工具包括：

sw_single
: `sw_single` 可提供套件管理和系統更新功能。使用 `rug`，而不使用程序檔中的 YaST。請參閱 第 9.1 節「使用 rug 透過指令行更新」[182頁]。

online_update_setup
: `online_update_setup` 會為您的系統設定自動更新。這可以使用 `cron` 設定。

inst_suse_register
: 請利用 `inst_suse_register` 註冊您的 SUSE Linux Enterprise。如需關於註冊的詳細資訊，請參閱第 8.3.4 節「註冊 SUSE Linux Enterprise」[127頁]。

hwinfo
: `hwinfo` 可提供您系統硬體的相關資訊。指令 `hwinfo` 也有相同作用。

GenProf、LogProf、SD_AddProfile、SD_DeleteProfile、SD_EditProfile、SD_Report 和子領域
: 這些模組可控制或設定 AppArmor。 AppArmor 擁有自己的指令行工具。

8.13.1 管理使用者

與傳統指令不同，YaST 用於使用者管理的指令在建立、修改或移除使用者時，會將您系統的組態驗證方法和預設使用者管理設定納入考量。舉例來說，您在使用者新增前後都不需要建立主目錄或複製 `skel` 檔案。如果您輸入使用者名稱和密碼，所有其他設定都會根據預設組態而自動產生。指令行提供的功能與圖形介面的相同。

YaST 模組 `users` 可用於使用者管理。若要顯示指令選項，請輸入 `yast users help`。

若要新增多個使用者，請建立 `/tmp/users.txt` 檔案，並在其中列出要新增的使用者。請在每一行輸入一個使用者名稱，並使用下列程序檔：

範例 8.2 *新增多個使用者*

```
#!/bin/bash
#
# adds new user, the password is same as username
#

for i in `cat /tmp/users.txt`;
do
 yast users add username=$i password=$i
done
```

與新增操作類似，您可以刪除 /tmp/users.txt 中定義的使用者：

範例 8.3 *移除多個使用者*

```
#!/bin/bash
#
# the home will be not deleted
# to delete homes, use option delete_home
#

for i in `cat /tmp/users.txt`;
do
yast users delete username=$i
done
```

8.13.2 設定網路和防火牆

程序檔中通常沒有網路和防火牆組態指令。請針對網路組態使用 yast lan 和 yast firewall。

若要顯示 YaST 網路卡組態選項，請輸入 yast lan help。若要顯示 YaST 防火牆介面卡組態選項，請輸入 yast firewall help。YaST 之網路和防火牆組態的效果會保持一致。重新開機以後，並沒有必要再次執行程序檔。

若要顯示網路的組態摘要，請使用 yast lan list。範例 8.4「yast lan list 的輸出範例」[174頁]輸出的第一個項目是設備 ID。如需設備組態的詳細資訊，請使用 yast lan show id=<number>。此範例中的正確指令為 yast lan show id=0。

範例 8.4 *yast lan list 的輸出範例*

```
0       Digital DECchip 21142/43, DHCP
```

YaST 防火牆組態的指令行介面可快速且輕易地啟用或停用各種服務、連接埠或通訊協定。若要顯示允許的服務、連接埠和通訊協定，請使用 yast firewall services show。如需啟用服務或連接埠的範例，請使用 yast firewall services help。如需啟用偽裝，請輸入 yast firewall masquerade enable。

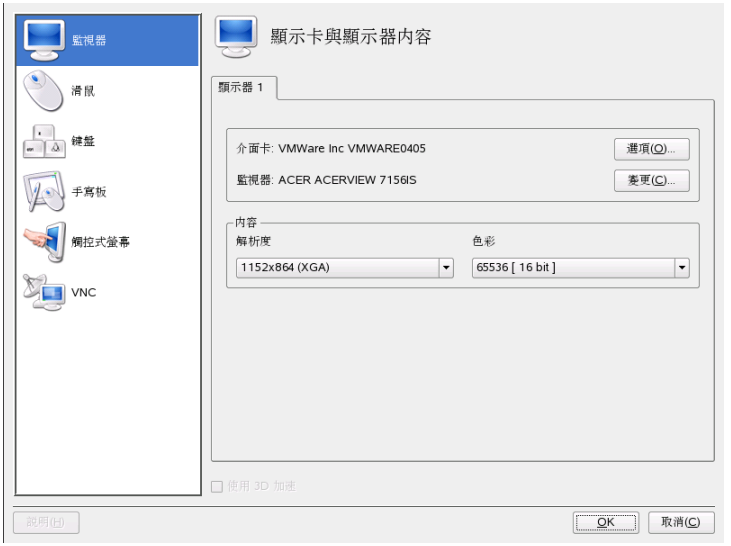
8.14 SaX2

使用「*硬體*」>「*圖形卡和監視器*」來設定系統的圖形環境。這時將會開啟「SUSE 進階 X11 設定」介面 (SaX2)，供您在其中設定如滑鼠、鍵盤和顯示設備等設備。此介面也可以從 GNOME 主功能表 (「*電腦*」>「*其他應用程式*」>「*系統*」>「*Sax2*」) 或 KDE 主功能表 (「*系統*」>「*設定*」>「*SaX2*」) 進行存取。

8.14.1 圖形卡與顯示器內容

請在「*圖形卡與顯示器內容*」調整圖形卡與顯示設備的設定。若您安裝了多張圖形卡，每項設備都會顯示在可由索引標籤讀取的個別對話方塊中。在對話方塊的上方，您可看到所選的圖形卡以及所連接的顯示器的目前設定。若您在介面卡 (雙螢幕顯示) 上連接了不止一個螢幕，此時會顯示主要輸出上的顯示器。一般而言，系統會在安裝期間自動偵測介面卡和顯示器設備。然而，您也可手動調整許多參數，甚至完全變更顯示設備。

圖形 8.11 *圖形卡與顯示器內容*



提示：自動偵測新的顯示硬體

如果您在安裝後變更顯示硬體，請使用指令行的 `sax2 -r` 來讓 SaX2 偵測硬體。您必須以 `root` 身份登入才能從指令行執行。

圖形卡

您無法變更圖形卡，因為系統僅支援已知型號，而這些是自動偵測的。然而，您可變更許多影響圖形卡行為的選項。一般而言，您並不需要這樣做，因為系統已在安裝過程中正確設定這些選項了。若您是進階使用者，且希望調整一些選項，請按一下圖形卡旁邊的「*選項*」，並選取要變更的選項。若要將所需的值指派至特定選項，請在選擇選項後出現的對話方塊中輸入數值。完成後，按一下「*確定*」，關閉選項對話方塊。

顯示器

若要變更顯示器的目前設定，請按一下顯示器旁邊的「*變更*」。這時會出現新的對話方塊，讓您調整多種顯示器特殊設定。對話方塊會針對不同的顯示器操作提供多個索引標籤。請選取第一個索引標籤，在兩份清單中手動選擇顯示設備的廠商與型號。若您的顯示器未列於上面，可選擇適合您需求的 VESA 或 LCD 模式之一，或若您有廠商驅動程式磁片或 CD 的話，請按一下「*磁碟公用程式*」，並遵循螢幕指示使用。選取「*啟用 DPMS*」，使用顯示器電源管理訊號。包含顯示器幾何內容的「*顯示器大小*」、以及包含顯示器水平與垂直同步頻率範圍的「*同步頻率*」，通常都是由系統進行正確的設定，不過您可以手動修改這些值。完成所有調整之後，請按一下「*確定*」，關閉此對話方塊。

警告：變更顯示器頻率

雖然有安全機制，但手動變更允許的顯示器頻率時，還是應該小心謹慎。不正確的值可能會損壞您的顯示器。變更頻率前請務必參考顯示器的手冊。

解析度與色彩深度

您可直接從對話方塊中間的兩個清單中，選擇解析度和色彩深度。您在此選取的解析度會標示出要使用的最高解析度。所有低至 640x480 的常見解析度也都會自動加入組態中。根據所使用的圖形桌面，您可稍後切換任何解析度，而無需重新設定。

雙螢幕顯示

若您的圖形卡在電腦上安裝了兩個輸出，則您可在系統上連接兩個螢幕。當兩個螢幕連接到相同圖形卡時，就稱為*雙螢幕顯示*。SaX2 會自動偵測系統中的多顯示設備，並據此備妥組態。若要使用圖形卡的雙螢幕顯示模式，請選取對話方塊底部的「*啟用雙螢幕顯示模式*」，並按一下「*設定*」來設定雙螢幕顯示選項，並在雙螢幕顯示對話方塊中設定其排列方式。

位於對話方塊頂端列中的每個索引標籤，都會對應到系統中的圖形卡。選取要配置的圖形卡，並在下方的對話方塊中設定其多重顯示器選項。在多重顯示器對話方塊的上方，按一下「*變更*」來設定其他螢幕。可能的選項與第一個螢幕相同。在清單中選擇此螢幕所使用的解析度。選取可能的三種多重顯示器模式之一。

複製多重顯示器
: 在這個模式中，所有顯示器都會顯示相同的內容。只有在主要螢幕上才看得到滑鼠。

Xinerama 多重顯示器
: 所有螢幕會組合成單一的大畫面。程式視窗可以隨意放置在所有螢幕上，或是縮放大小讓視窗填滿一台以上的顯示器。

注

Linux 目前不支援 Xinerama 多重顯示器的 3D 環境。在此狀況下，SaX2 會停用 3D 支援。

雙螢幕顯示環境的排列方式會描述各個螢幕的順序。根據預設，SaX2 會遵循偵測到的圖形卡順序設定標準配置，由左自右安排所有螢幕。在對話方塊的「*排列方式*」部分，選取順序按鈕之一，決定螢幕的排列方式。完成後，按一下「*確定*」，關閉對話方塊。

提示：透過筆記型電腦使用視訊設備

若要將視訊設備連接到筆記型電腦，請啟用雙螢幕顯示模式。在此情況中，SaX2 會將外部輸出的解析度設定為 1024x768，更新速率則為 60 Hz。這些值最適用於視訊設備。

多重顯示器

如果已在電腦中安裝多個圖形卡，您的系統就可以連接一台以上的螢幕。在 不同 圖形卡上連接兩個螢幕，請參閱 *多重顯示器*。SaX2 會自動偵測系統中的多圖形卡，並據此備妥組態。根據預設，SaX2 會遵循偵測到的圖形卡順序設定標準配置，由左自右安排所有螢幕。另一個「*排列方式*」索引標籤可讓您手動變更此配置。在格線中拖曳代表個別螢幕的圖示，並按一下「*確定*」，關閉對話方塊。

測試與組態

完成顯示器與圖形卡的設定之後，在主視窗中按一下「*確定*」，然後測試您的設定。這個動作可以確保該設定是否適用於您的設備。如果影像不穩定，按 Ctrl+Alt+Backspace 立即終止測試，然後降低更新速率或色彩深度。

注

無論您是否執行測試，只有在重新啟動 X 伺服器之後才會啟用所有修改。

8.14.2 滑鼠內容

在「*滑鼠內容*」中調整滑鼠的設定。若您以不同驅動程式安裝了一個以上的滑鼠，則各驅動程式會顯示於不同的索引標籤中。由相同驅動程式操作的多個設備會顯示為單一設備。您可透過對話方塊上方的核取方塊，來啟用或停用目前選擇的滑鼠。在核取方塊下可看到該滑鼠的目前設定。一般而言，會自動偵測到滑鼠，但若自動偵測有誤的話，您亦可手動變更。如需型號的描述，請參閱您的滑鼠文件。按一下「*變更*」，從兩個下拉式清單中選擇廠商與型號，並按一下「*確定*」確認您的選擇。在對話方塊的選項部分，設定操作滑鼠的多種選項。

「*啟用 3 鍵模擬*」
若您的滑鼠只有兩個按鍵，則可在您同時按下兩鍵時模擬第三鍵。

「*啟用滑鼠滾輪*」
選取此方塊可使用捲動滾輪。

「*反轉 X 軸*」和「*反轉 Y 軸*」
若選取其中一個選項，滑鼠游標會以相反方向移動。此功能有時對觸控版非常實用。

「*以滑鼠按鍵模擬滾輪*」
若您的滑鼠沒有捲動滾輪，但您希望使用類似功能的話，您可指派其他的按鍵執行此功能。選擇要使用的按鈕。按下此按鈕後，滑鼠的任何動作都會被解譯為捲動滾輪的指令。此功能對於軌跡球而言特別有用。

滿意您的設定之後，請按一下「*確定*」確認您的變更。

注

在此執行的任何變更，都會在重新啟動 X 伺服器後生效。

8.14.3 鍵盤內容

使用此對話方塊，在圖形環境中調整操作您鍵盤的設定。在對話方塊上半部中選取類型、語言配置，和自設機甲。使用對話方塊下方的測試區域，檢查特定的字元是否能夠正確顯示。從中間的清單中選取要使用的其他配置與自設機甲。根據您桌面的類型，這些設定可能會在執行中系統中直接切換，而無需重新設定。按一下「*確定*」之後，就會立即套用變更。

8.14.4 圖形板內容

使用此對話方塊以設定您系統所連接的圖形板。按一下「*圖形板*」索引標籤，從清單中選擇廠商與型號。目前僅支援有限的圖形板。若要啟用圖形板，請在對話上方選取「*啟用此圖形板*」。

在「*連接埠與模式*」對話方塊中，設定圖形板的連線。SaX2 可讓您設定連接於 USB 埠或序列埠的圖形板。如果圖形板已連接到序列埠，請確認該埠。`/dev/ttyS0` 參照到第一序列埠。`/dev/ttyS1` 會參照到第二序列埠。其他埠則使用類似的表示法。從清單中選擇適當的「*選項*」，並選擇符合您需求的「*主要圖形板模式*」。

若您的圖形板支援電子感應筆，請在「*電子感應筆*」中設定它們。新增橡皮擦與筆，並按一下「*內容*」設定其內容。

滿意您的設定之後，請按一下「*確定*」，確認您的變更。

8.14.5 觸碰式螢幕內容

使用此對話方塊來設定系統所連接的觸碰式螢幕。若您安裝了多個觸碰式螢幕，則各設備會顯示於不同索引標籤中的對話方塊。若要啟用目前選擇的觸碰式螢幕，請選取對話上方的「*指定觸碰式螢幕以顯示*」。從下方清單中選擇廠商與型號，並在底部設定適當的「*連接埠*」。您可以設定連接至 USB 埠或序列埠的觸碰式螢幕。如果觸碰式螢幕已連接到序列埠，請確認該埠。`/dev/ttyS0` 參照到第一序列埠。`/dev/ttyS1` 會參照到第二序列埠。其他埠則使用類似的表示法。滿意您的設定之後，請按一下「*確定*」確認您的變更。

8.15 疑難排解

所有錯誤訊息和警示會記錄在 `/var/log/YaST2` 目錄中。可找出 YaST 問題的最重要檔案為 `y2log`。

8.16 如需更多資訊

您可以在下列網站和目錄中找到更多有關 YaST 的資訊：

- `/usr/share/doc/packages/yast2`—本地 YaST 開發文件
- `http://www.opensuse.org/YaST_Development`—openSUSE wiki 中的 YaST 專案頁面
- `http://forge.novell.com/modules/xfmod/project/?yast`—其他 YaST 專案頁面

9 用 ZENworks 管理軟體

SUSE Linux Enterprise 可立即整合到由 Novell ZENworks Linux Management 管理的環境中。其中包含開放原始碼 ZENworks 管理代辦、後端精靈和使用者空間軟體管理工具。Novell ZENworks 套件管理工具使用 ZENworks Linux Management 伺服器來下載套件和更新。如果本地網路上沒有可用的 ZENworks Linux Management 伺服器，您的系統可從 Novell 客戶中心取得更新，如第 3.14.4 節「Novell Customer Center 組態」[37頁] 中所述。

Novell ZENworks Linux Management 代辦的後端精靈是 ZENworks 管理精靈 (ZMD)。ZMD 可執行軟體管理功能。開機時會自動啟動精靈。

使用 `rczmd status` 可檢查精靈的狀態。若要啟動精靈，請輸入 `rczmd start`。若要重新啟動精靈，請使用 `rczmd restart`。使用 `rczmd stop` 可停用精靈。

也可使用特殊選項啟動精靈，以控制其行為。若要永久使用一些特殊選項啟動 ZMD，請在 `/etc/sysconfig/zmd` 中設定 `ZMD_OPTIONS`，然後執行 `SuSEconfig`。可用的選項為：

-n, --no-daemon
: 不在背景中執行精靈。

-m, --no-modules
: 不載入任何模組。

-s, --no-services
: 不載入初始服務。

-r, --no-remote
不啟動遠端服務。

ZMD 組態儲存在 /etc/zmd/zmd.conf 中。您可以手動或使用 rug 變更組態。ZMD 在首此啟動時使用之 ZENworks 服務的 URL 和註冊碼儲存在 /var/lib/zmd 中。系統會將更新下載到 /var/cache/zmd 中的 ZMD 快取記憶體。

ZMD 僅做為後端。透過指令行工具 rug 或圖形 軟體更新程式 applet 可啟動軟體管理任務。

9.1 使用 rug 透過指令行更新

rug 可根據指定指令，使用 zmd 精靈來安裝、更新和移除軟體。它可以從本地檔案或是從伺服器安裝軟體。您可以使用一個或多個遠端伺服器(稱為服務)。支援的服務有適用於本地檔案的 mount 和適用於伺服器的 yum 或 ZENworks。

rug 會將服務的軟體歸入各個目錄 (又稱為通道)，與相似軟體的群組相對應。例如，一個目錄可能包含來自更新伺服器的軟體，以及來自協力軟體廠商的軟體。您可以訂閱各個目錄，以控制所顯示的可用套件，以免意外安裝不需要的軟體。更新作業通常都只會對所訂閱目錄中的軟體進行。

9.1.1 取得 rug 的資訊

rug 可提供廣泛的可用資訊。它可讓您檢查 zmd 的狀態、檢視已註冊的服務和目錄，或者查看關於可用修補程式的資訊。

如果一段時間內沒有使用 zmd，它便會切換到睡眠模式。若要檢查 zmd 狀態或重新啟用精靈，請使用 rug ping。此指令會喚醒 zmd 並記錄其狀態資訊。

若要查看您已註冊的服務，請使用 rug sl，若要查看您的系統上支援哪些服務，請使用 rug st。

若要檢查有無新的修補程式，請使用 rug pch。若要獲取某個修補程式的相關資訊，請輸入 rug patch-info *patch*。

9.1.2 訂閱 rug 服務

依預設，新安裝的系統會訂閱數個服務。若要新增服務，請使用 `rug sa URI service_name`。以具有意義且獨一無二的字串取代 *service_name*，該字串用於辨識新服務。

注：存取更新目錄時發生錯誤

如果您無法存取更新目錄，可能表示訂閱已過期。SUSE Linux Enterprise 通常提供一年或三年的訂閱，您只能在這個時間段內存取更新目錄。一旦訂閱期結束，此存取權將被拒絕。

存取更新目錄遭拒絕時，會顯示一則警告訊息，建議您造訪 Novell Customer Center 以檢查您的訂閱。Novell Customer Center 的網址為 http://www.novell.com/center/。

9.1.3 以 rug 安裝和移除軟體

若要從已訂閱的目錄安裝套件，請使用 `rug in package_name`。若要僅從選取的目錄進行安裝，請使用 `-c catalog name`。若要獲取某套件的相關資訊，請使用 `rug if package_name`。

若要移除套件，請使用 `rug rm package_name`。如果有其他套件依賴這個套件，`rug` 會顯示它們的名稱、版本和類型。確認是否確實要移除套件。

9.1.4 rug 使用者管理

`rug` 的一項主要優點是它的使用者管理功能。通常情況下，只有 `root` 可以更新或安裝新套件。使用 `rug` 時，您可以將更新系統的權限指定給其他使用者，並設定一些限制，例如，只能更新而不能移除軟體。您可以授予下列權限：

安裝
: 可以安裝新軟體

鎖定
: 可以設定套件鎖定

移除
　可以移除軟體

訂閱 (subscribe)
　可以變更通道訂閱

受信任 (trusted)
　認為使用者是可信的，因此他能夠在沒有套件簽章的情況下安裝套件

升級
　可以更新軟體套件

檢視
　這個權限可讓使用者檢視機器上已安裝哪些軟體，以及可用通道中有哪些軟體。這個選項只和遠端使用者有關，通常本地使用者已允許檢視已安裝和可用的套件。

超級使用者 (superuser)
　允許執行所有 rug 指令，但是使用者管理和設定除外，這些工作只能在本地進行。

若要指定更新系統的使用者許可權，請使用 `rug ua` *`username`* `upgrade` 指令。以該使用者的名稱取代 *`username`*。若要撤銷使用者的權限，請使用指令 `rug ud` *`username`*。若要列出使用者及其權限，請使用 `rug ul`。

若要變更使用者目前的權限，請使用 `rug ue` *`username`* 並使用所需使用者的名稱取代 *`username`*。您會獲取選定使用者的權限清單。`edit` 指令屬於互動性質。請使用加號 (+) 或減號 (-) 來增加或移除使用者的權限，然後按 Enter。例如，若要允許使用者刪除軟體，請輸入 `+remove`。若要儲存和結束，請在空白提示中按下 Enter。

9.1.5 編程更新

使用 `rug`，就可以自動更新系統 (例如，利用程序檔)。最簡單的範例就是全自動更新。若要這麼做，請以 `root` 身份設定 cron 工作，執行 `rug up -y`。`up -y` 選項會下載並安裝目錄中的修補程式，不需您的確認。

但是，您可能不希望自動安裝修補程式，而希望擷取它們並在以後選取修補程式以進行安裝。若只下載修補程式，請使用 `rug up -dy` 指令。`up -dy` 選項

無需您的確認即會下載目錄中的修補程式，並將其儲存到 rug 快取記憶體。rug 快取記憶體的預設位置是 /var/cache/zmd。

9.1.6 設定 rug

rug 可讓您透過一組優先設定來自定其安裝。而其中有一些設定是在安裝期間預先設定的。使用 rug get 指令可獲取可用優先設定的清單。若要編輯優先設定，請輸入 rug set *preference*。例如，若您必須透過代理來更新系統，則請調整設定。下載更新之前，把您的使用者名稱和密碼送到代理伺服器。若要執行此作業，請使用以下指令：

```
rug set proxy-url url_path
rug set proxy-username name
rug set proxy-password password
```

以代理伺服器的名稱取代 *url_path*。以您的使用者名稱取代 *name*。以您的密碼取代 *password*。

9.1.7 如需更多資訊

如需有關使用指令行進行更新的詳細資訊，請輸入 rug --help，或參閱 rug(1) 手冊頁。所有的 rug 指令也可以使用 --help 選項。例如，假設您需要 rug update 的說明，請輸入 rug update --help。

9.2 使用 ZEN 工具管理套件

ZEN 工具可作為 ZENworks Management Daemon (zmd) 的圖形前端，可讓您輕鬆安裝或移除軟體、套用安全性更新，而且只要按幾下即可管理服務和類別。

9.2.1 取得權限

若要管理 Linux 系統上的套件，您需要「根部」權限。ZEN 工具和 rug 擁有專屬的使用者管理系統，讓使用者安裝軟體更新。當使用者在 ZEN 工具中進行需要特殊權限的動作時，就會出現提示要求輸入「根部」密碼。密碼經過驗證之後，ZEN 工具便會自動將使用者帳戶新增到具備更新權限的使用者管理系統。

若要檢閱或變更這些設定，請使用 `rug` 使用者管理指令 (請參閱第 9.1.4 節「rug 使用者管理」[183頁] 以取得相關資訊)。

9.2.2 取得和安裝軟體更新

軟體更新程式 駐留在通知區域 (GNOME) 或面板的系統匣 (KDE)，作用是描繪地球的圖示。會根據網路連結和新更新的可用性，來變更色彩和外觀。軟體更新程式 每天會自動檢查一次，是否有系統適用的更新 (以滑鼠右鍵按一下應用程式圖示並選擇「*重新整理*」強制執行立即檢查)。如果有新的更新，面板中的軟體更新程式 Applet 就會從地球變更為橘色背景上的驚嘆號。

注：存取更新目錄時發生錯誤

如果您無法存取更新目錄，可能表示訂閱已過期。SUSE Linux Enterprise 通常提供一年或三年的訂閱，您只能在這個時間段內存取更新目錄。一旦訂閱期結束，此存取權將被拒絕。

存取更新目錄遭拒絕時，會顯示一則警告訊息，建議您造訪 Novell Customer Center 以檢查您的訂閱。Novell Customer Center 的網址為 `http://www.novell.com/center/`。

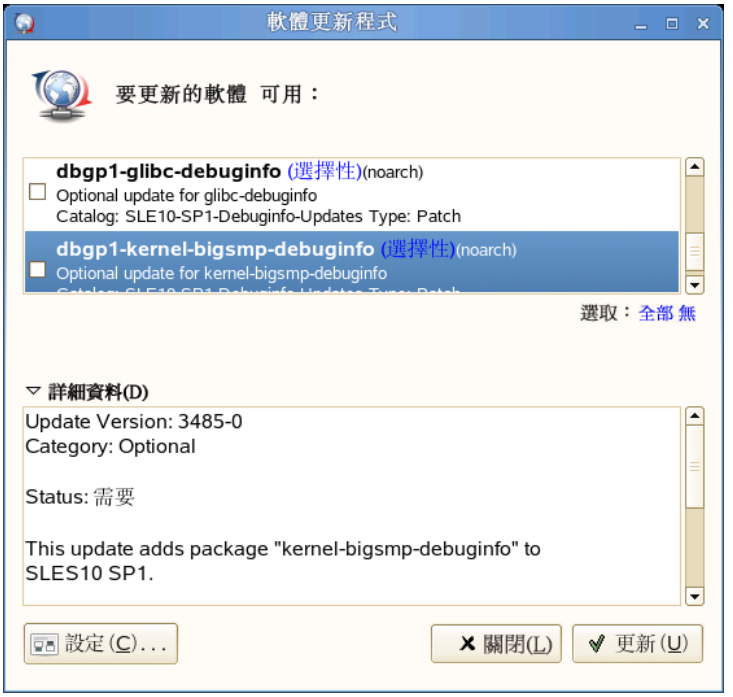
以滑鼠左鍵按一下面板圖示以開啟更新器視窗。會顯示可用的修補程式和新套件版本的清單。每個項目都會有簡短說明和類別圖示 (若有的話)：安全性修補程式會以黃色屏蔽標示。選用的修補程式會以淡藍色的圓圈標示。建議使用的修補程式不會以圖示標示。會先列出安全性修補程式，然後依次列出建議使用的修補程式、選用修補程式，最後則會列出新套件版本。使用連結「*全部*」、「*套件*」和「*修補程式*」以過濾所顯示的套件清單。

注：套件和修補程式

Novell 正式出版的更新會以「*修補程式*」的形式呈現。其他來源的新套件版本則會以「*套件*」的形式呈現。

若要取得特定項目的詳細資料，請以滑鼠標示並按一下清單視窗下的「*詳細資料*」連結。若要選取要安裝的項目，請勾選項目的核取方塊。使用連結「*全部*」和「*無*」以選取或取消選取所有修補程式。按一下「*更新*」即會安裝所選的程式。

圖形 9.1　選取軟體更新



9.2.3 安裝軟體

若要安裝軟體套件，請從功能表啟動「*安裝軟體*」或執行 `zen-installer`。介面幾乎與 軟體更新程式 完全相同 (請參閱第 9.2.2 節「取得和安裝軟體更新」[186頁])。唯一不同點在於您可用於搜尋套件或過濾清單的搜尋面板。將應該要安裝之套件的核取方塊標示出來，然後按「*安裝*」以開始安裝套件。安裝程式會自動解析與其他套件的可能相依性。

9.2.4 移除軟體

從功能表啟動「*移除軟體*」或執行 `zen-remover` 以取消安裝軟體套件。可使用連結「*產品*」(取消安裝完整產品)、「*樣式*」(請參閱章節「安裝和移除模式」[121頁]以取得樣式的詳細資料)、「*套件*」和「*修補程式*」來減少列出的套件清單。將應該要移除之清單項目的核取方塊標示出來，然後按「*移除*」以開始取

消安裝套件。如果其他套件與您標示的套件有依存關係，則也會遭到移除。您必須確認已移除其他套件。如果您在確認對話方塊中按一下「*取消*」，則不會取消安裝任何套件。

9.2.5 設定 軟體更新程式

若要設定 ZEN 工具，請在應用程式視窗中按一下「*設定*」。包含三個索引標籤的視窗便會開啟：「*服務*」、「*類別*」和「*優先設定*」。

服務及類別

一般而言，服務是提供軟體套件和套件相關資訊的來源。每個服務都可提供一個以上的類別。

服務索引標籤會列出所有可用的服務以及類型和狀態資訊(如果您沒有看到後兩項資訊，請調整視窗大小)。使用「*移除服務*」或「*新增服務*」以新增或移除服務。下列是可用的服務類型：

YUM
: 針對套件資料使用 RPM-MD 格式的 HTTP、HTTPS 或 FTP 伺服器。

ZYPP
: ZYPP 服務指的是在 YaST 中透過「*軟體*」>「*安裝來源*」新增的 YaST 安裝來源。使用 軟體更新程式 或 YaST 新增安裝來源。會預先安裝您最初安裝的來源 (在大多數的情況下是 DVD 或 CD-ROM)。如果您變更或刪除此來源，請以其他有效的安裝來源 (ZYPP 服務) 加以取代，否則，您就無法安裝新軟體。

注：術語

YaST 安裝來源、YaST 套件儲存機制和 ZYPP 服務等辭彙都是您可安裝軟體之來源的相同名稱。

設備
: 使用「*裝載*」，將目錄內嵌到您的機器。例如，在定期鏡像 Novell YUM 伺服器並將內容輸出到本地網路的網路中，便可以使用此功能。若要新增目錄，請在「*服務 URI*」提供目錄的完整路徑。

NU
: Novell 更新的 NU 標準。Novell 會提供專供 NU 服務使用 SUSE Linux Enterprise 的更新。如果您在安裝期間設定更新，正式的 Novell NU 伺服器就會出現在清單中。

  如果您在安裝期間略過更新組態，請執行指令行上的 `suse_register` 或做為根部使用者的 YaST 模組「*軟體*」>「*產品註冊*」。Novell 更新伺服器會自動新增到 軟體更新程式。

RCE 和 ZENworks
: 只有當您的公司或組織已經在內部網路中安裝這些服務，Opencarpet、Red Carpet Enterprise 或 ZENworks 服務才可以使用。舉例來說，您的組織可能使用的是第三方軟體，而會在單一伺服器上部署更新。

安裝 SUSE Linux Enterprise 之後，系統會預先設定兩個服務：做為 ZYPP 服務的安裝來源 (DVD、CD-ROM 或網路資源)，以及做為 NU 服務的 SUSE Linux Enterprise 更新伺服器 (在產品註冊期間進行新增)。通常不需要變更這些設定。如果您沒有看到 NUYUM 服務，請開啟 `root` 外圍程序並執行指令 `suse_register`。會自動新增服務。

類別

服務可以提供軟體中不同部分或不同軟體版本的套件 (通常是 RCE 或 ZENworks 服務才會這麼做)。會以名為「*類別*」的不同類別加以組織整理。標示或取消標示類別前方的核取方塊即可訂閱或取消訂閱類別。

目前，SUSE Linux 服務 (YUM 和 ZYPP) 不會提供不同的類別。每個服務只會有一個類別。如果 軟體更新程式 在安裝期間已經設定好或具備 `suse_register`，則會自動訂閱 YUM 和 ZYPP 類別。如果您手動新增服務，則必須訂閱其類別。

優先設定

在「*優先設定*」索引標籤上指定 軟體更新程式 是否要在啟動時加以啟動。做為「根部」使用者，您也可以修改 軟體更新程式 設定。因為您是沒有特殊權限的使用者，所以您只可以檢視設定。請參閱 `rug` 主頁面以探索這些設定。

9.3 如需更多資訊

如需有關 ZENworks Linux Management 和 ZMD 的詳細資訊，請參閱 http://www.novell.com/products/zenworks/linuxmanagement/index.html。

10 更新 SUSE Linux Enterprise

SUSE® Linux Enterprise 不必完全重新安裝即可將現有系統更新成新版本。不需要任何新安裝。類似主目錄和系統組態等舊資料仍保持不變。在產品的生命週期期間，您可以套用 Service Pack 來提高系統安全性，以及修正軟體問題。請從本地 CD 或 DVD 光碟機或從中央網路安裝來源進行安裝。

10.1 更新 SUSE Linux Enterprise

如果要從 SUSE Linux Enterprise Server 9 更新到 SUSE Linux Enterprise Server 10，請依照此節中說明的步驟進行。如果您要從 SUSE Linux Enterprise 10 SP1 更新為 SUSE Linux Enterprise 10 SP2，也請依照這些步驟進行。

軟體通常會隨著版本更新而「擴增」。 因此在更新之前，先使用 `df` 來檢視可用的分割空間。如果您認為您的磁碟空間可能不夠，請在更新前先確保您資料的安全，然後再進行分割。每個分割區應該多大並沒有常規可循。空間需求將依特定的磁碟分割設定檔和選取軟體而有差異。

10.1.1 準備

更新之前，請先將舊的組態檔案複製到個別媒體，例如磁帶設備、抽取式硬碟、USB 晶片組或 ZIP 磁碟，以確保資料的安全。此作業主要適用於儲存在 `/etc` 中的檔案、一些目錄及 `/var` 和 `/opt` 中的檔案。您最好也將 `/home` (即 `HOME` 目錄) 中的使用者資料複製到備份媒體。將此資料備份為 `root`。只有 `root` 擁有所有本地檔案的讀取許可權。

開始更新前，請記住這個root分割區。`df /`指令會列出根分割區的設備名稱。在 範例 10.1「使用 `df -h` 來列示」[192頁] 中，要記下的root分割區為 `/dev/hda3` (裝載位置是 /)。

範例 10.1 *使用 df -h 來列示*

```
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda3        74G   22G   53G  29% /
tmpfs           506M     0  506M   0% /dev/shm
/dev/hda5       116G  5.8G  111G   5% /home
/dev/hda1        44G    4G   40G   9% /data
```

10.1.2 可能的問題

如果您將預設系統從上一版更新到這個版本，YaST 會執行必要變更並予以執行。根據您的自定項目，有些步驟或整個更新程序可能會失敗，導致您必須求訴於複製還原備份資料。請先在開始系統更新之前檢查下列問題。

檢查 /etc 中的密碼和群組

更新系統之前，請先確定 `/etc/passwd` 和 `/etc/group` 沒有任何語法錯誤。若要完成這個目的，請以 `root` 身份來啟動驗證公用程式 `pwck` 和 `grpck`，並排除任何回報的錯誤。

PostgreSQL

在更新 PostgreSQL (`postgres`)之前，先傾印資料庫。請參閱手冊中的 `pg_dump`。您只需要在更新前有使用PostgreSQL的情況下，才需要執行這個動作。

10.1.3 使用 YaST 更新

依照第 10.1.1 節「準備」[191頁]所述步驟執行準備程序之後，您就可以開始更新系統了：

1 您可以選擇性地準備安裝伺服器。如需背景詳細資訊，請參閱第 4.2.1 節「使用 YaST 設定安裝伺服器」[51頁]。

2 依照 第 3.3 節「系統啟動進行安裝」[18頁] 中說明的安裝方式將您的系統開機。在 YaST 中，選擇語言，並選取「*安裝模式*」對話方塊中的「*更新*」。請不要選取「*全新安裝*」。

3 YaST 決定是否有多個 root 分割區。如果只有一個，繼續進行下一個步驟。若有多個的話，請選擇正確的分割區，然後按一下「*下一步*」，確認 (的範例是選擇/dev/hda3第 10.1.1 節「準備」[191頁])。YaST 會在此分割區上讀取舊的 `fstab`，以分析並裝載此處列出的檔案系統。

4 在「*安裝設定*」對話方塊中，根據您的需求調整設定。一般而言，您可以保留預設設定，但若您希望加強系統，請在「*軟體選擇*」子功能表中選取提供的套件，或是新增其他的語言支援。

4a 請按一下「*更新選項*」，只更新已安裝的軟體(「*只更新已安裝套件*」)或根據選取的模式來新增軟體或功能到系統中。建議接受所建議的元件。您可以稍後使用 YaST 對其進行調整。

4b 您也可以製作不同系統元件的備份 (「*備份*」)。選取備份會使更新程序變慢。如果您沒有最新的系統備份，請使用這個選項。

5 按一下「*接受*」並確認「*開始更新*」，開始軟體安裝程序。

安裝結束後，請閱讀版本說明，然後按一下「*完成*」以重新啟動電腦並登入。

10.2 安裝 Service Pack

使用 Service Pack 來更新 SUSE Linux Enterprise 安裝。您可以用幾種不同的方式來套用 Service Pack。您可以使用 Service Pack 媒體來更新現有安裝，或是啟動全新安裝。這裡將介紹更新系統和設定中心網路安裝來源的可能情形。

提示：安裝變更

請讀取 Service Pack 媒體的安裝指示，取得關於變更的詳細資訊。

10.2.1 設定 Service Pack 媒體的網路安裝來源

處理 SUSE Linux Enterprise 初始安裝時，比起使用一組實體媒體為所有用戶端個別進行安裝，從網路中的中心安裝來源為所有用戶端提供安裝服務的方式，可以獲得更高效率。

使用 YaST 設定 SUSE Linux Enterprise 的網路安裝來源

基本上，請依據第4.2節「安裝保存安裝來源的伺服器」[51頁]所列程序執行。您只要加入另一個名為 SLE-10-SP-*x*-*arch*、SLES-10-SP-*x*-*arch* 或 SLED-10-SP-*x*-*arch* 的安裝來源 (其中 *x* 是 Service Pack 編號，而 *arch* 是硬體結構的名稱)，並設定可經由 NFS、HTTP 或 FTP 進行。

10.2.2 安裝 Service Pack

注

若要將現有 SUSE Linux Enterprise 10 系統更新成 SUSE Linux Enterprise 10 Service Pack (SP)，請參閱第 10.2.3 節「更新 Service Pack」[196頁]。

安裝 SUSE Linux Enterprise Service Pack 的程序非常類似安裝原始 SUSE Linux Enterprise 媒體。處理原始安裝時，您可以選擇從本地 CD 或 DVD 光碟機或是中央網路安裝來源進行安裝。

從本地 CD 或 DVD 光碟機

在開始 SUSE Linux Enterprise SP 的新安裝程序前，請確定所有 Service Pack 安裝媒體 (CD 或 DVD) 都已備妥。

過程 10.1 *從 Service Pack 媒體開機*

1. 插入第一份 SUSE Linux Enterprise SP 媒體 (CD 1 或 DVD 1)，然後開機。這時會出現類似 SUSE Linux Enterprise 10 原始安裝的開機畫面。

2 請選取「安裝」並參照第 3 章「*使用 YaST安裝*」[17頁]中的 YaST 安裝指示繼續進行。

網路安裝

在開始進行 SUSE Linux Enterprise SP 網路安裝之前，請先確定確實符合下列需求：

- 網路安裝來源已依據第 10.2.1 節「設定 Service Pack 媒體的網路安裝來源」[194頁]所述完成設定。
- 安裝伺服器以及包含名稱服務、DHCP (選擇性，但是 PXE 開機時必須用到) 和 OpenSLP (選擇性) 的目標機器，都已正常連接網路。
- 要用來為目標系統開機的 SUSE Linux Enterprise SP CD 1 或 DVD 1，或依照第 4.3.5 節「準備用於 PXE 啟動的目標系統」[67頁]所述步驟設定 PXE 開機時所要使用的目標系統。

網路安裝—使用 CD 或 DVD 開機

若要使用 SP CD 或 DVD 當作開機媒體來進行網路安裝，請依照下列步驟執行：

1 插入 SUSE Linux Enterprise SP CD 1 或 DVD 1，然後開機。這時會出現類似 SUSE Linux Enterprise 10 原始安裝的開機畫面。

2 選取「安裝」來為 SP 核心開機，然後使用 F3，選取網路安裝來源類型 (FTP、HTTP、NFS 或 SMB)。

3 提供適當的路徑資訊，或是選取「*SLP*」作為安裝來源。

4 從所提供伺服器中選擇適當的安裝伺服器，或是依照第 3.3.4 節「從沒有 SLP 的網路來源安裝」[19頁]所述步驟，使用開機選項提示字串提供安裝來源類型和確實位置。YaST 於是啟動。

請依照第 3 章「*使用 YaST安裝*」[17頁]所述完成安裝。

網路安裝—PXE 開機

若要執行 SUSE Linux Enterprise 的網路安裝，請依照下列步驟執行：

1 依據第 4.3.5 節「準備用於 PXE 啟動的目標系統」[67頁]所述，調整 DHCP 伺服器的設定，提供 PXE 開機時所需要的位址資訊。

2 設定 TFTP 伺服器存放 PXE 開機時所需要的開機影像。

使用 SUSE Linux Enterprise Service Pack 的第一張 CD 或 DVD 進行這項開機，或者依照第 4.3.2 節「設定 TFTP 伺服器」[61頁]的指示執行安裝。

3 在目標機器上準備 PXE 開機和網路喚醒功能。

4 啟始目標系統開機，並使用VNC遠端連接到這部機器所執行的安裝常式。如需相關資訊，請參閱第 4.5.1 節「安裝 VNC」[74頁]。

5 接受授權書，然後選取語言、預設桌面和其他的安裝設定。

6 按一下「*是，請安裝*」，開始安裝。

7 繼續依一般方式進行安裝 (輸入 `root` 的密碼、完成網路設定、測試網路連線、啟用「線上更新服務」、選擇使用者驗證方法，並輸入使用者名稱和密碼)。

如需安裝 SUSE Linux Enterprise 的詳細指示說明，請參閱第 3 章「*使用 YaST安裝*」[17頁]。

10.2.3 更新 Service Pack

有兩種首選方法可以將系統更新為 Service Pack (SP) 功能層級。一種是以 SP 媒體開機，備選方法是執行 YaST 線上更新或 zen-updater，然後選取「*更新到 Service Pack X*」修補程式。更新為新的功能層級後，您的系統就可以使用諸如新驅動程式或軟體增強功能之類的額外功能。

警告：請勿遺漏「*更新到 Service Pack*」修補程式

如果不選取「*更新到 Service Pack*」修補程式，系統將仍然停留在先前的功能層級，而您只能取得限定時間內的錯誤修正和安全性更新 (對於 SUSE Linux Enterprise 10 SP2，此時間段現在延長為六個月)。因此，為了確保持續的系統完整性，建議儘早變更為新的功能層級。

另一種更新方法是使用修補程式CD(請參閱第 8.3.7 節「從修補程式光碟進行更新」[131頁]) 或使用本地安裝的 SMT 系統，手動執行 `rug` 指令。

注

在 s390 系統中，修補程式 CD 更新選項不可用。

更新時使用 SP 媒體開機

請使用 SP 媒體開機，並在 YaST 中選擇「*更新*」做為安裝模式。如需詳細資訊並完成更新程序，請參閱第 10.1.3 節「使用 YaST 更新」[192頁]。

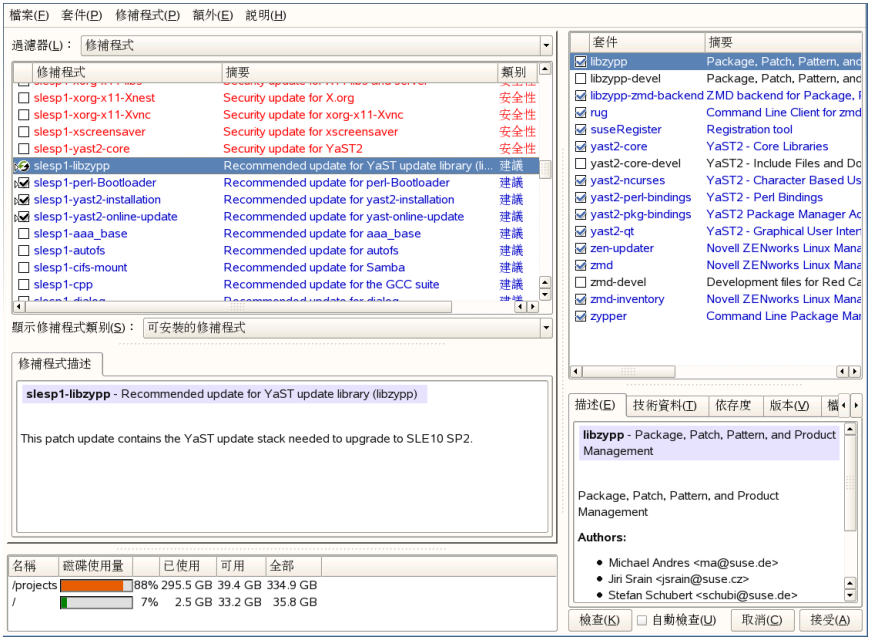
開始 YaST 線上更新

要起始 YaST 線上更新來更新至 SP 功能等級之前，請先確定是否符合下列需求：

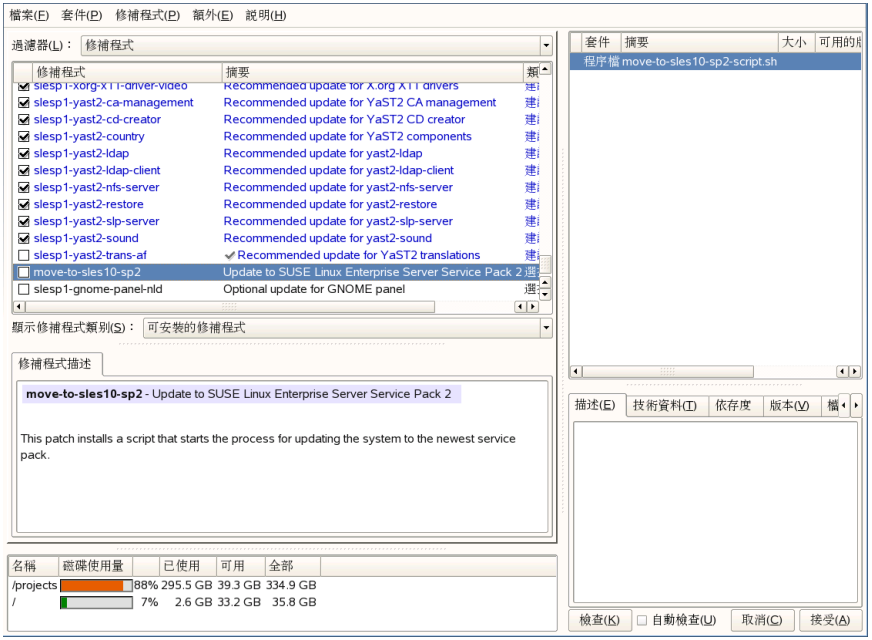
- 在整個更新程序中，系統都必須保持在線上的狀態，因為此程序需要存取 Novell Customer Center。
- 如果您的安裝中有協力廠商軟體或附加軟體，請在其他機器上測試此程序，以確定這些相依性並未遭到更新程序的破壞。
- 請確定整個程序都成功完成，否則系統會不一致。

如果先前安裝了完整的 Service Pack 1，則只能更新到 Service Pack 2。如果情況並非如此，請依照章節「SUSE Linux Enterprise GA 到 SP1 和 SP2」[203頁]中的說明先更新到 Service Pack 1。

圖形 10.1　Service Pack 1套件管理更新



圖形 10.2　更新至 Service Pack 2



注

在使用 YaST 線上更新執行更新移轉期間，將會更新 ZMD 堆疊，ZMD 精靈也會重新啟動。因此，建議不要使用任何其他軟體管理工具，例如 `rug`、`zen-updater`、`zen-installer` 和 `zen-remover`。建議在移轉期間結束 `zen-updater`。

1 在正在執行的 SUSE Linux Enterprise 系統中選取「*電腦*」>「*YaST*」>「*軟體*」>「*線上更新*」。

如果您不是以 `root` 身份登入，請在收到提示時輸入 `root` 密碼。

2 這時會開啟「*線上更新*」對話方塊。有幾個修補程式已預先選取。向下捲動修補程式清單並驗證是否真正預先選取了套件管理相關的修補程式和 SUSE Linux Enterprise 10 SP2 維護堆疊更新 (`slesp1u-libzypp`)。按一下「*接受*」以套用選取的更新。

3 「*修補程式下載和安裝*」對話方塊會追蹤進度記錄。當「*總進度*」達到「*100%*」時，請按一下「*關閉*」。「*線上更新*」隨即會自動重新啟動。

4 重新啟動後，按一下「*接受*」套用所有可用的更新以及新核心。安裝後，必須重新啟動系統。

5 在重新啟動的「*線上更新*」中，現在可以向下捲動修補程式清單，並選取「*更新至 Service Pack 2*」(`move-to-sles10-sp2`)，如圖形 10.2「更新至 Service Pack 2」[199頁]中所示。在快顯視窗中按一下「*接受*」，以確認開始更新程序，進而更新為 Service Pack 功能等級。

`move-to-sles10-sp2` 修補程式標示為 `optional`。如果不選取該項，系統將仍然停留在 SP1 功能層級，而您只能取得限定時間內 (SP2 可用後的六個月) 的錯誤修正和安全性更新。

6 「*下載並安裝修補程式*」對話方塊可以追蹤移轉修補程式安裝程序的進度記錄。當「*總進度*」達到「*100%*」時，請按一下「*完成*」。

7 再次啟動 YaST 線上更新。套用 `product-sles10-sp2` 和 `slesp2o-sp2_online` 修補程式，將系統升級至 SP2。如果您在先前的步驟中安裝了 `move-to-sles10-sp2`，系統會預先選取這兩個修補程式，這是強制性的。

8 按一下「*關閉*」以完成到 SUSE Linux Enterprise 10 SP2 的更新並重新開機。

使用 zen-updater 啟動

如需 ZENworks 的背景資訊，請參閱第 9 章「*用 ZENworks 管理軟體*」[181頁]。

在使用 zen-updater 啟動線上更新來提升到 SP 功能層級之前，請確定符合章節「開始 YaST 線上更新」[197頁]中列出的需求。

*圖形 **10.3**　套用 SLE10 SP2 維護堆疊更新*

1 在執行 SUSE Linux Enterprise 的系統中，按一下底部的更新程式圖示啟動 zen-updater。

提示：喚醒 ZMD

如果看到「*ZMD 未在執行中*」訊息，不論 ZMD 是否在執行，都請使用 `rczmd status` 以 `root` 身份歸還某個終端機。如果發生問題，請輸入 `rug restart --clean` 強制重新啟動並清理 ZMD 及其資料庫。

如果您不是以 `root` 身分登入，請在收到提示時輸入 `root` 密碼。

2 套用可用於您系統的所有維護更新。

3 套用 SLE10 SP2 維護堆疊更新 (slesp1u-libzypp)。這些項目應已預先選取，並且按一下「*更新*」應該可以啟動此步驟。解決所有相依問題後，請按一下「*套用*」。完成時按一下「*關閉*」以確認快顯訊息。

4 在重新啟動的「軟體更新程式」中，向下翻頁然後選取選擇性的 move-to-sles10-sp2 修補程式並套用。如果不選取該項，系統將仍然停留在 SP1 功能層級，而您只能取得限定時間內 (SP2 可用後的六個月) 的錯誤修正和安全性更新。

5 在「軟體更新程式」中，套用 product-sles10-sp2 和 slesp2o-sp2_online 修補程式，將系統升級至 SP2。如果您在先前的步驟中安裝了 move-to-sles10-sp2，系統會預先選取這兩個修補程式，這是強制性的。

6 按一下「*關閉*」以完成到 SUSE Linux Enterprise 10 SP2 的更新並重新開機。

使用 rug

如需關於 rug 指令行工具的背景資訊，請參閱第 9.1 節「使用 rug 透過指令行更新」[182頁]。如需此更新的可編寫指令解決方案，請使用 rug。

在使用 rug 啟動線上更新以提升到 SP 功能層級之前，請確定符合章節「開始 YaST 線上更新」[197頁]中列出的需求。

若要將系統移轉到 SP2 修補程式層級，至少需要執行以下指令序列：

```
rug in -t patch slesp1u-libzypp && rug ping -a
rug in -t patch move-to-sles10-sp2 && rug ping -a
rug refresh && rug ping -a
rug up -t patch -g recommended && rug ping -a
reboot
```

注

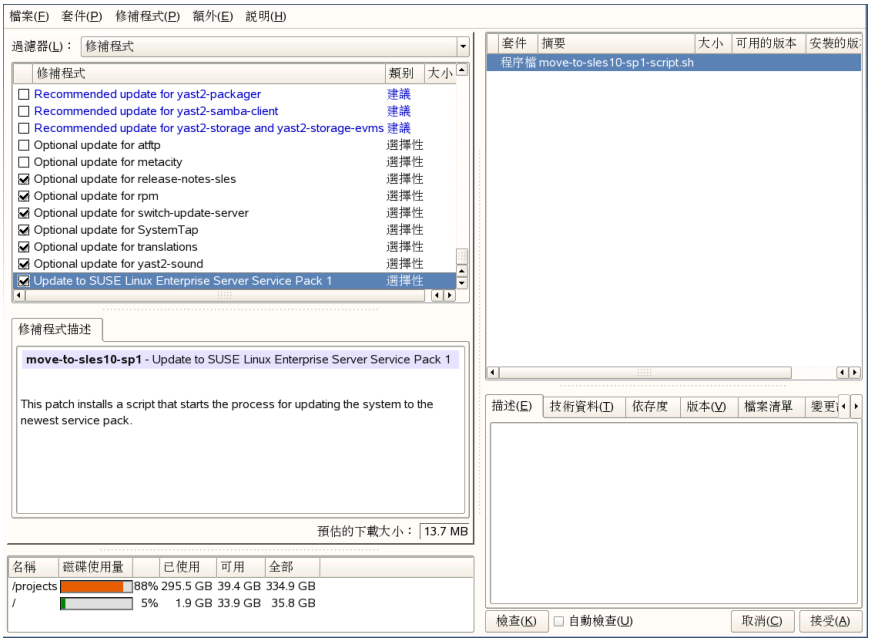
rug ping -a，以確保在執行先前的 rug 指令之後，ZMD 啟始化已完成。

SUSE Linux Enterprise GA 到 SP1 和 SP2

注

僅當您的系統依然以 GA 修補程式層級執行時，下列步驟才與主題相關。

圖形 10.4　更新至 Service Pack 1



1 在正在執行的 SUSE Linux Enterprise 系統 (GA) 中，選取「*電腦*」>「*YaST*」>「*軟體*」>「*線上更新*」。

如果您不是以 `root` 身份登入，請在收到提示時輸入 `root` 密碼。

2 這時會開啟「*線上更新*」對話方塊。向下捲動修補程式清單，選取「*更新至 Service Pack 1*」，如圖形 10.4「更新至 Service Pack 1」[203頁]所示。在快顯視窗中按一下「*接受*」，以確認開始更新程序，進而更新為 Service Pack 功能等級。

3 「*下載並安裝修補程式*」對話方塊可以追蹤移轉修補程式安裝程序的進度記錄。當「*總進度*」達到「*100%*」時，請按一下「*完成*」。

4 請再次執行線上更新。完成之後，請在「*下載並安裝修補程式*」中按一下「*關閉*」。第二次執行更新時，YaST 會安裝核心以及所有其他的軟體。

5 當您看到進度記錄最後區域附近出現「*安裝完成*」時，按一下「*完成*」。

6 為了完成更新，請手動將系統重新開機，以啟用新核心。

現在，SUSE Linux Enterprise 將以 SP1 修補程式層級執行。繼續章節「開始 YaST 線上更新」[197頁]，以將系統提升到 SP2 修補程式層級。

10.3 版本 9 至版本 10 的軟體變更

下面將詳細介紹版本 9 演進至版本 10 的個別變更項目。如摘要所述，是否已完全重新設定基本設定、是否已將設定檔移至他處，或者是否已大幅變更一般應用程式。這裡也會介紹影響到使用者階層、或管理者階層之日常系統使用的大幅度修改。

注：從 SLES 10 到 SLES 10 SP 1 的軟體變更

如需從 SUSE Linux Enterprise Server 10 到 SUSE Linux Enterprise Server 10 SP1 的軟體與組態變化詳細清單，請參閱 service pack 的版本說明。請使用 YaST 版本說明模組，在已安裝系統上進行檢閱。

10.3.1 多核心

的確可以安裝多個核心。這個功能是指，允許管理員透過安裝新核心來從一個核心升級到另一個核心，接著驗證新核心可以如預期般正常運作，然後再解除安裝舊核心。在 YaST 不支援這項功能時，使用 `rpm -i package.rpm` 指令便可輕易地從外圍程序安裝核心或解除安裝核心。

預設的開機載入程式功能表包含一個核心項目。在安裝多個核心之前，您可以為額外的核心新增一個項目，以方便您選取這些核心。在安裝新核心之前啟動的核心，可以依 `vmlinuz.previous` 及 `initrd.previous` 方式存取。透過建立與預設項目相似的開機載入程式項目，並讓這個項目從參照 `vmlinuz` 與 `initrd`，改成參照 `vmlinuz.previous` 與 `initrd.previous`，就可以存取之前啟動的核心。此外，GRUB 與 LILO 支援萬用字元開機載入程式項目。

請參閱 GRUB 資訊頁面 (`info grub`) 以及 `lilo.conf` (5) 手冊頁，以取得詳細資訊。

10.3.2 核心模組的變更

不再提供下列核心模組：

- `km_fcdsl`—AVM Fritz!Card DSL
- `km_fritzcapi`—AVM FRITZ! ISDN 介面卡

下列核心模組已進行內部變更：

- `km_wlan`—適用於無線 LAN 卡的各種驅動程式。來自 `km_wlan`、用於 Atheros WLAN 卡的 `madwifi` 驅動程式已被移除。

基於技術原因，必須停止支援 Ralink WLAN 卡。下列模組不包含在配送中，且未來也不會加入配送：

- `ati-fglrx`—ATI FireGL 圖形卡
- `nvidia-gfx`—NVIDIA gfx 驅動程式
- `km_smartlink-softmodem`—Smart Link 軟體數據機

10.3.3 主控台編號變更及序列設備

如同 2.6.10 所述，ia64 上序列設備的命名依據為 ACPI 及 PCI 列舉順序。ACPI 名稱空間中的第一個設備 (如果有的話) 為 `/dev/ttyS0`，第二個為 `/dev/ttyS1`，依此類推，而 PCI 設備會在 ACPI 設備之後依序命名。

在 HP 系統上，您必須重新設定 EFI 主控台，然後才能從核心開機指令刪除主控台參數。若要解決這個問題，您可以嘗試使用 `console=ttyS1...` 來取代 `console=ttyS0...` 作為開機參數。

詳細資訊請參閱 `kernel-source` 軟體套件中的 /usr/src/linux/Documentation/ia64/serial.txt。

10.3.4 LD_ASSUME_KERNEL 環境變數

LD_ASSUME_KERNEL 環境變數不再需要設定。過去可使用此變數來加強 LinuxThreads 支援，但 LinuxThreads 支援已被移除。如果在 SUSE Linux Enterprise 10 中設定LD_ASSUME_KERNEL=2.4.x，ld.so 將會在不存在的路徑中尋找 glibc 及相關工具，進而造成損壞。

10.3.5 較嚴格的 tar 語法

新的 `tar` 使用語法較為嚴格。`tar` 選項必須放在指定檔案或目錄的前面。將選項 (例如 `--atime-preserve` 或 `--numeric-owner`) 附加在指定的檔案或目錄之後，則 `tar` 指令會失敗。請檢查您的備份程序檔。下列指令已經無法運作：

```
tar czf etc.tar.gz /etc --atime-preserve
```

如需詳細資訊，請參閱 `tar` info 頁面。

10.3.6 以 Apache 2.2 取代 Apache 2

Apache 網頁伺服器 (版本 2) 已取代為版本 2.2。針對 Apache 版本 2.2，第 40 章「*Apache HTTP 伺服器*」[673頁] 進行了重新安排。此外，如需一般升級資訊，請造訪 `http://httpd.apache.org/docs/2.2/upgrading.html`；如需新功能的說明，請造訪 `http://httpd.apache.org/docs/2.2/new_features_2_2.html`。

10.3.7 網路驗證的 Kerberos

`Kerberos` 取代 `heimdal`，成為預設的網路驗證。無法自動轉換現有的 heimdal 組態。系統更新組態檔案的備份時，會建立 表格 10.1「備份檔案」[207頁] 中顯示的項目。

表格 10.1　備份檔案

舊的檔案	備份檔案
/etc/krb5.conf	/etc/krb5.conf.heimdal
/etc/krb5.keytab	/etc/krb5.keytab.heimdal

用戶端組態 (/etc/krb5.conf) 和 heimdal 的組態十分相似。若無特別設定，用 admin_server 來替換參數 kpasswd_server 就可以了。

您無法複製伺服器相關 (kdc 和 kadmind) 資料。系統更新之後，仍舊可以在 /var/heimdal 中使用舊的 heimdal 資料庫；MIT kerberos 仍會維護 /var/lib/kerberos/krb5kdc 下的資料庫。如需詳細資訊，請參閱第 45 章「*網路驗證—Kerberos*」[761頁] 與第 46 章「*安裝與管理 Kerberos*」[767頁]。

10.3.8 udev 精靈所處理的熱插拔事件

熱插拔事件現在已全部由 udev 精靈所處理 (udevd)。/etc/hotplug.d 和 /etc/dev.d 中的事件多工器系統已停止使用。現在是由 udevd 根據其規則直接呼叫所有熱插拔協助工具。udev 規則和協助工具是由 udev 和其他套件所提供。

10.3.9 安裝期間啟用防火牆

為了提高安全性，系統會在安裝結束時，以提議對話方塊啟用附帶的防火牆解決方案 SuSEFirewall2。這表示一開始時所有連接埠都已關閉，而且可以需要時從提議對話方塊開啟。依照預設，您無法從遠端系統登入。這也會干擾網路瀏覽及多重廣播應用程式，像是 SLP 及 (網路上的芳鄰)，以及一些遊戲。您可以使用 YaST 來微調防火牆的設定。

在安裝服務與設定服務期間，如果需要網路存取，個別 YaST 模組會開啟所有內部與外部模組中必要的 TCP 與 UDP 埠。不需要時，請關閉 YaST 模組中的連接埠或指定其他詳細的防火牆設定。

10.3.10 KDE 和 IPv6 支援

依照預設，KDE 並未啟用 IPv6 支援。您可以使用 YaST 的 `/etc/sysconfig` 編輯器來啟用它。停用這個功能的原因，是因為不是所有的網際網路服務提供者都支援 IPv6 位址，所以可能會造成瀏覽網頁時出現錯誤訊息，以及網頁的顯示出現延遲現象。

10.3.11 線上更新與 Delta 套件

線上更新現在支援特別的 RPM 套件，此套件僅會儲存指定基本套件的二進位變更。這項技術大幅地降低了套件大小，以及最後重新組合需要用到較多 CPU 的下載時間。如需技術詳細資訊，請參閱 `/usr/share/doc/packages/deltarpm/README`。

10.3.12 列印系統組態

在安裝結束前 (提議對話方塊) 必須在防火牆組態中打開列印系統所需的連接埠。CUPS 需要連接埠 631/TCP 和連接埠 631/UDP，而且不應將其關閉，以維持正常作業。連接埠 515/TCP (用於舊的 LPD 協定) 和 Samba 所使用的連接埠也都必須開啟，以透過 LPD 或 SMB 來列印。

10.3.13 變更為 X.Org

從 XFree86 到 X.Org 的變更是由相容連結來進行，透過該連結可使用舊名稱來存取重要的檔案和指令。

表格 10.2 *指令*

XFree86	X.Org
`XFree86`	`Xorg`
`xf86config`	`xorgconfig`
`xf86cfg`	`xorgcfg`

表格 10.3　/var/log 中的記錄檔

XFree86	X.Org
XFree86.0.log	Xorg.0.log
XFree86.0.log.old	Xorg.0.log.old

在變更為 X.Org 的過程中，XFree86* 套件會被重新命名為 xorg-x11*。

10.3.14 X.Org 組態檔案

組態工具 SaX2 可將 X.Org 組態設定寫入 /etc/X11/xorg.conf。在重頭安裝時，並不會建立任何從 XF86Config 到 xorg.conf 的相容連結。

10.3.15 刪除 XView 和 OpenLook 支援

捨棄套件 xview、xview-devel、xview-devel-examples、olvwm 和 xtoolpl。過去僅提供 XView (OpenLook) 基本系統。系統更新之後，我們不再提供 XView 程式庫。更重要的，無法再使用 OLVWM (OpenLook Virtual Window Manager，OpenLook 虛擬視窗管理員)。

10.3.16 X11 的終端機模擬器

由於某些終端機模擬器已停止維護或無法在預設環境中運作，尤其是不支援 UTF-8 的關係，這些終端機模擬器已被移除。SUSE Linux Enterprise Server 提供標準終端機，像是 xterm、KDE 及 GNOME 終端機，以及 mlterm (X 的多語系終端機模擬器)，這些終端機可取代 aterm 與 eterm。

10.3.17 OpenOffice.org (OOo)

目錄

OOo 現在安裝於 /usr/lib/ooo-2.0 而非 /opt/OpenOffice.org。使用者設定的預設目錄現在為 ~/.ooo-2.0 而非 ~/OpenOffice.org1.1。

包裝程式

有一些新的包裝程式可用來啟動OOo元件。表格 10.4「包裝程式」[210頁]會列出這些新名稱。

表格 10.4 *包裝程式*

舊的	新增
/usr/X11R6/bin/OOo-calc	/usr/bin/oocalc
/usr/X11R6/bin/OOo-draw	/usr/bin/oodraw
/usr/X11R6/bin/OOo-impress	/usr/bin/ooimpress
/usr/X11R6/bin/OOo-math	/usr/bin/oomath
/usr/X11R6/bin/OOo-padmin	/usr/sbin/oopadmin
/usr/X11R6/bin/OOo-setup	–
/usr/X11R6/bin/OOo-template	/usr/bin/oofromtemplate
/usr/X11R6/bin/OOo-web	/usr/bin/ooweb
/usr/X11R6/bin/OOo-writer	/usr/bin/oowriter
/usr/X11R6/bin/OOo	/usr/bin/ooffice
/usr/X11R6/bin/OOo-wrapper	/usr/bin/ooo-wrapper

包裝程式現在支援選項 `--icons-set`，因此可在 KDE 和 GNOME 圖示之間切換。不再支援下列選項：`--default-configuration`、`--gui`、`--java-path`、`--skip-check`、`--lang` (目前由地區設定判斷語言)、`--messages-in-window`，與 `--quiet`。

KDE 和 GNOME 支援

可在 `OpenOffice_org-kde` 和 `OpenOffice_org-gnome` 套件中找到 KDE 和 GNOME 的副檔名。

10.3.18 混音器 kmix

已預先將混音器 kmix 設定為預設項目。高階硬體有其他混音器，像是 QAMix/KAMix、envy24control (限 ICE1712) 或 hdspmixer (限 RME Hammerfall)。

10.3.19 DVD 燒錄

過去我們會從 `cdrecord` 套件將修補程式套用到 `cdrecord` 二進位檔案，以便支援燒錄 DVD。現在，新安裝的二進位檔案 `cdrecord-dvd` 本身會包含這個修補程式。

`dvd+rw-tools` 套件的 `growisofs` 程式現在可以燒錄所有的 DVD 媒體 (DVD+R、DVD-R、DVD+RW、DVD-RW、DVD+RL)。建議您使用此程式，取代套用修補程式的 `cdrecord-dvd`。

10.3.20 在核心提示啟動手動安裝

開機載入程式畫面已不再提供「*手動安裝*」模式。您仍然能夠在開機提示中使用 `manual=1`，讓 linuxrc 進入手動模式。通常這不是必要的，因為您可以直接在核心提示中設定安裝選項，像是 `textmode=1`，或是設定一個 URL 做為安裝來源。

10.3.21 JFS：不再支援

由於 JFS 的技術問題，所以不再支援 JFS。核心檔案系統驅動程式還在，只是 YaST 不提供使用 JFS 進行磁碟分割。

10.3.22 AIDE 做為 Tripwire 替代品

如需入侵偵測系統，請使用 GPL 所發行的 AIDE (套件名稱為 `aide`)。SUSE Linux 上無法再使用 Tripwire。

10.3.23 PAM 組態

新的組態檔案 (包含註解提供更多資訊)

`common-auth`
Auth 區段的預設 PAM 組態

`common-account`
帳戶區段的預設 PAM 組態

`common-password`
密碼變更的預設 PAM 組態

`common-session`
工作階段管理的預設 PAM 組態

您應該在應用程式特殊組態檔案中加入這些預設組態檔案，因為與修改、維護將近四十個在系統中常見的檔案相比，僅修改和維護一個組態檔是要容易多了。若您之後再安裝一個應用程式，此應用程式會繼承已套用的變更，管理員不需要費事去調整其組態。

這個變更很簡單。若您有下列組態檔 (大多數的應用程式預設會有這些檔案)：

```
#%PAM-1.0
auth     required       pam_unix2.so
account  required       pam_unix2.so
password required       pam_pwcheck.so
password required       pam_unix2.so    use_first_pass use_authtok
#password required      pam_make.so     /var/yp
session required        pam_unix2.so
```

您可以將它變更成：

```
#%PAM-1.0
auth     include        common-auth
account  include        common-account
password include        common-password
session  include        common-session
```

10.3.24 使用 su 指令成為超級使用者

依照預設，呼叫 su 以成為 root 使用者，並不會設定 root 的 PATH。請呼叫 su -，以 root 的完整環境啟動登入外圍程序，或若您希望變更 su 的預設行為的話，請在 /etc/default/su 中將 ALWAYS_SET_PATH 設定為 是。

10.3.25 powersave 套件中的變更

/etc/sysconfig/powersave 中的組態檔案已變更：

表格 10.5 *分割 /etc/sysconfig/powersave 中的組態檔案*

舊的	現在已分割為
/etc/sysconfig/powersave/common	common
	cpufreq
	events
	battery
	sleep
	thermal

/etc/powersave.conf已經過時。現有的變數已移至 表格 10.5「分割 /etc/sysconfig/powersave 中的組態檔案」[213頁] 中所列的檔案。如果您變更了 /etc/powersave.conf 中的「event」變數，則現在必須在 /etc/sysconfig/powersave/events 中進行相同的變更。

下列睡眠狀態名稱已經變更：

- 暫停 (ACPI S4, APM 暫停)
- 待命 (ACPI S3、APM 待命)

收件者：

- 暫停寫入到磁碟 (ACPI S4, APM 暫停)
- 暫停寫入到 RAM (ACPI S3, APM 暫停)
- 待命 (ACPI S1、APM 待命)

10.3.26 省電組態變數

省電組態變數的名稱已變更以達一致性，但 sysconfig 檔案仍然相同。如需更多詳細資訊，請參閱第 28.5.1 節「設定 powersave 套件」[472頁]。

10.3.27 PCMCIA

`cardmgr` 已不再管理 PC 卡。而是由核心模組管理 Cardbus 卡和其他子系統。所有必要的動作均由 `hotplug` 所執行。`pcmcia` 啟動程序檔已被移除，而且 `cardctl` 已取代成 `pccardctl`。若需詳細資訊，請檢視 `/usr/share/doc/packages/pcmciautils/README.SUSE`。

10.3.28 設定 .xinitrc 中程序間通訊的 D-BUS

許多應用程式現在必須依靠 D-BUS 來進行程序間通訊(IPC)。呼叫 `dbus-launch` 可啟動 `dbus-daemon`。全系統 `/etc/X11/xinit/xinitrc` 會使用 `dbus-launch` 來啟動視窗管理員。

如果您有本地 `~/.xinitrc` 檔案，您就必須跟著進行變更。否則如 f-spot、banshee、tomboy 或網路管理員 banshee 等應用程式都可能會失敗。請儲存舊的 `~/.xinitrc`。然後使用以下指令，將新的範本檔案複製到主目錄：

```
cp /etc/skel/.xinitrc.template ~/.xinitrc
```

最後，從儲存的 `.xinitrc` 新增自定。

10.3.29 重新命名的 NTP 相關檔案

為了顧及與 LSB (Linux Standard Base) 的相容性，大部分的組態檔和 init 程序檔都必須從 `xntp` 重新命名為 `ntp`。新的檔案名稱為：

- `/etc/slp.reg.d/ntp.reg`
- `/etc/init.d/ntp`
- `/etc/logrotate.d/ntp`
- `/usr/sbin/rcntp`
- `/etc/sysconfig/ntp`

10.3.30 GNOME 應用程式的檔案系統變更通知

GNOME 應用程式必須依靠檔案系統變更通知的支援，才能正常運作。如果是使用僅限本地的檔案系統，請安裝 gamin 套件 (優先設定) 或執行 FAM 精靈。如果是使用遠端檔案系統，請在伺服器和用戶端上執行 FAM，並為 FAM 啟動的 PRC 呼叫開啟防火牆。

GNOME (gnome-vfs2 和 libgda) 包含了包裝程式，其會選擇 gamin 或 fam 來提供檔案系統變更通知：

- 若 FAM 精靈未執行，則偏好使用 gamin (原理：只有 gamin 支援 Inotify，且其對本地檔案系統而言較有效率)。
- 若執行 FAM 精靈，則偏好使用 FAM (原理：若執行 FAM，您可能希望得到遠端通知，而只有 FAM 支援此功能)。

10.3.31 啟動 FTP 伺服器 (vsftpd)

根據預設，`xinetd` 將不再啟動 `vsftpd` FTP 伺服器。它目前是獨立執行的精靈，因此您必須使用 YaST 執行期間編輯器來進行設定。

10.3.32 Firefox 1.5：URL 開啟指令

在 Firefox 1.5 中，應用程式開啟 Firefox 例項或視窗的方法已改變。這項新方法已有部分運用於舊版本，當時是透過包裝程式程序檔來實作這些行為。

如果您的應用程式不使用`mozilla-xremote-client`或`firefox -remote`，您就不需要進行任何變更。另外，開啟 URL 的新指令是`firefox` *`url`*，而且這個指令可以隨時執行，無論 Firefox 是否正在執行。如果 Firefox 正在執行，它會依照「*從其他應用程式開啟連結*」已設定的優先設定來執行動作。

透過指令行，您可以使用`firefox -new-window` *`url`* 或 `firefox -new-tab` *`url`* 來影響這項行為。

II. 管理

11 OpenWBEM

Novell® 已接納分散式管理任務推動小組 (Distributed Management Task Force, DMTF) [http://www.dmtf.org/home] 所擬定之網路企業管理 (Web-Based Enterprise Management, WBEM) 開放標準策略。應用這些策略可以大幅降低與管理網路中異質系統相關的複雜性程度。

以下內容將介紹根據 DMTF 標準所擬定之部分元件。認識這些元件和它們彼此之間的關聯，有助於您了解 OpenWBEM 是什麼，以及您要如何最有效地在網路中應用這項工具。

- 網路企業管理 (Web-Based Enterprise Management, WBEM) 是針對統整企業運算環境管理工作，而開發的一組管理暨網際網路標準技術。WBEM 的引進，使得業界能夠提供遵循標準的、利用新興 Web 技術的、充分整合的管理工具組合。DMTF 已經制訂組成 WBEM 的一組核心標準：
  - 資料模型：標準的通用資訊模型 (CIM)
  - 編碼規格：CIM-XML 編碼規格
  - 輸送機制：透過 HTTP 的 CIM 操作
- 通用資訊模型 (CIM) 是一種描述管理的概念性資訊模型，而不專指特定實作方式。管理系統和應用程式之間，可以運用這個模型來交換管理資訊。這個模型可以是適用於分散式系統管理使用的 Agent-to-Manager 或是 Manager-to-Manager 通訊模式。CIM 由兩個部分組成：CIM 規格與 CIM 綱要。

  CIM 規格描述語言、命名和中繼綱要 (Meta Schema)。中繼綱要是正式的模型定義。它定義了用來表達此模型的詞彙，以及這些詞彙的使用和語意。中

繼綱要的元素包括類別、內容和方法。這個中繼綱要也支援「類別」類型的指示 (Indications) 和關聯 (Associations)，以及「屬性」類型的參考 (References)。

CIM 綱要則提供實際的模型描述。CIM 綱要提供了一組包含可提供完整解讀概念框架之屬性和關聯的類別，使用者可運用此框架來組織關於受管理環境的可用資訊。

- 共用資訊模型物件管理員 (Common Information Model Object Manager, CIMOM) 就是 CIM 物件管理員，更詳細說來，就是指根據 CIM 標準管理物件的應用程式。
- CIMOM 提供者，是指透過 CIMOM 來執行用戶端應用程式所要求之特定任務的軟體。每個提供者都會執行 CIMOM 綱要的一項或多項範疇工作。

SUSE® Linux Enterprise Server 包含 OpenWBEM 計劃 [http://openwbem.org] 的開放原始碼 CIMOM。

網路企業管理 (Web-Based Enterprise Management) 軟體選項包括了一組包含基本 Novell 提供者的套件，其中包括一些範例提供者，以及附隨 Novell 綱要的基礎組合。

由於 Novell 隨時與 OpenWBEM 和特定提供者開發同時演進，因此，我們會開發可提供下列重要功能的工具：

- 高效率的網路系統監控
- 記錄現有管理組態的變更情形
- 硬體清查和資產管理

認識如何安裝和設定 OpenWBEM CIMOM 的方式，有助於您更有自信、更輕鬆地監控和管理網路中的異質系統。

11.1 安裝 OpenWBEM

若要安裝 OpenWBEM，請在安裝 SUSE Linux Enterprise Server 時選取 YaST 中的 WBEM 軟體選項或模式，或者選取它做為要安裝在已執行 SUSE Linux Enterprise Server 伺服器上的元件。這個軟體選項包含了下列套件：

cim-schema，也就是共用資訊模型 (CIM) 綱要：
此套件包含通用資訊模型 (CIM)。CIM 是描述網路或企業環境內所有管理資訊的模型。CIM 由規格和綱要所組成。其中的規格定義了與其他管理模型整合的詳細資訊。而綱要則會提供實際的模型描述。

openwbem，即網路企業管理 (WBEM) 實作：
此套件包含 OpenWBEM 的實作。OpenWBEM是指一組可協助部署分散式管理任務推動小組 (DMTF) CIM 和 WBEM 技術的軟體元件。如果您不清楚 DMTF 及其技術，請造訪 DMTF 網站 [http://www.dmtf.org]。

openwbem-base-providers：
這個套件包含針對 OpenWBEM CIMOM 所使用基本作業系統元件的 Novell Linux 測試工具，其中包括電腦、系統、作業系統和處理器等元件。

openwbem-smash-providers：
這個套件包含針對 OpenWBEM CIMOM 所使用伺服器硬體系統管理結構 (Systems Management Architecture for Server Hardware, SMASH) 提供者的 Novell Linux 測試工具。

yast2-cim，即 YaST2 - CIM 繫結功能：
此套件會將 CIM 繫結新增到 YaST2 (YaST2 是 SUSE 系統工具管理員的圖形使用者介面)。這些繫結功能會提供存取共用資訊模型物件管理員 (CIMOM) 的用戶端介面。

本節包含以下資訊：

- 第 11.1.1 節「啟動、停止 owcimomd，或是檢查其狀態」[221頁]
- 第 11.1.2 節「確保安全的存取」[222頁]
- 第 11.1.3 節「設定記錄」[225頁]

11.1.1 啟動、停止 owcimomd，或是檢查其狀態

網路企業管理軟體完成安裝後，系統會預設啟動精靈 owcimomd。下表將說明如何啟動、停止和檢查 owcimomd 的狀態。

表格 11.1　管理 owcimomd 的指令

任務	Linux 指令
啟動 owcimomd	以 root 身份，在主控台外圍程序中輸入 `rcowcimomd start`。
停止 owcimomd	以 root 身份，在主控台外圍程序中輸入 `rcowcimomd stop`。
檢查 owcimomd 狀態	以 root 身份，在主控台外圍程序中輸入 `rcowcimomd status`。

11.1.2 確保安全的存取

OpenWBEM 的預設設定相當安全。然而，您可以檢閱下面項目，確保 OpenWBEM 元件的存取是否符合您組織的安全性需求。

- 章節「憑證」[222頁]
- 章節「埠」[223頁]
- 章節「驗證」[224頁]

憑證

安全通訊端層 (SSL) 傳輸必須使用憑證，才能執行安全通訊服務。當 OES 安裝完成時，OpenWBEM 便已產生自己的自簽憑證。

如有必要，您可以將預設憑證的路徑取代成購買的商用憑證路徑，或是取代成在 `/etc/openwbem/openwbem.conf` 檔案的 `http_server.SSL_cert = path_filename` 設定中所產生的其他憑證路徑。

預設產生的憑證會存放在下列位置：

```
/etc/openwbem/servercert.pem
```

如果要產生新憑證，請使用下列指令。執行這個指令便可取代目前的憑證，所以 Novell 會建議在產生新憑證之前先複製舊憑證作為備份。

以 root 的身份，在主控台外圍程序中輸入

```
sh/etc/openwbem/owgencert
```

如果您要變更 OpenWBEM 所使用的憑證，請參閱第 11.2.2 節「變更憑證設定」[233頁]。

埠

OpenWBEM 預設接受來自安全連接埠 5989 的所有通俊。下表說明連接埠設定與建議組態。

表格 11.2 *埠通訊設定和建議組態*

埠	類型	備註和建議
5989	安全	OpenWBEM 通訊使用的安全埠會經過 HTTPS 服務。 此為預設的組態。 如果使用這項設定，當透過網際網路在伺服器和工作站之間傳送時，所有 CIMOM 和用戶端應用程式之間的通訊都會進行加密。使用者必須通過用戶端應用程式驗證，才可以檢視這份資訊。 Novell 建議維護在組態檔案中的這項設定。 為了使 OpenWBEM CIMOM 可以與必要的應用程式進行通訊，如果路由器和防火牆是位在用戶端應用程式和受監控節點之間，則其中的這個埠必須為開啟狀態。
5988	不安全	OpenWBEM 通訊使用的不安全埠會經過 HTTP 服務。 這項設定已預設為停用。 使用這項設定時，所有 CIMOM 和用戶端應用程式之間的通訊，在透過網際網路在伺服器和工作站之間傳送時，都會開放提供任何人檢視，而不用任何驗證。

埠	類型	備註和建議
		Novell 建議只在嘗試為 CIMOM 相關問題除錯時使用這項設定。一旦問題獲得解決，請將不安全埠選項設回「停用」。 為了使 OpenWBEM CIMOM 可以與要求不安全存取權限之必要應用程式進行通訊，如果路由器和防火牆是位在用戶端應用程式和受監控節點之間，則其中的這個埠必須為開啟狀態。

如果您要變更預設的埠指定，請參閱第 11.2.3 節「變更埠設定」 [234頁]。

驗證

SUSE Linux Enterprise Server 中的 OpenWBEM 已經預設並啟用下列驗證設定：

您可以變更其中任何一項預設設定。請參閱第 11.2.1 節「變更驗證設定」 [226頁]。

- `http_server.allow_local_authentication = true`
- `http_server.ssl_client_verification = disabled`
- `http_server.use_digest = false`
- `owcimomd.allow_anonymous = false`
- `owcimomd.allowed_users = root`
- `owcimomd.authentication_module = /usr/lib/openwbem/authentication/libpamauthentication.so`

此 OpenWBEM CIMOM 已經預設啟用 PAM 功能；因此，本地 root 使用者可以使用本地 root 使用者認證向 OpenWBEM CIMOM 進行驗證。

11.1.3 設定記錄

您可以變更其中任何一項預設設定。若需要更多的資訊，請參閱第 11.2.4 節「變更預設的記錄設定」[235頁]。

根據預設，這時您要依照下面項目來設定 OpenWBEM 記錄功能。

- `log.main.components = *`
- `log.main.level = ERROR`
- `log.main.type = syslog`

這表示 owcimomd 記錄已設定成移到 `/var/log/messages` 檔案或其他檔案，這將由 `syslogd` 的組態來決定。它會記錄全部元件 (owcimomd) 的所有錯誤。

11.2 變更 OpenWBEM CIMOM 組態

當 OpenWBEM CIMOM (owcimomd) 啟動時，它會從 `openwbem.conf` 檔案讀取其執行時間設定。`openwbem.conf` 檔案存放在 `/etc/openwbem` 目錄中。

任何包含以分號 (;) 或是井字號 (#) 標記註解之選項的設定，都會使用此項預設設定。

您可以使用可將檔案儲存成使用中平台之原始格式的任何文字編輯器，來變更此檔案的內容。

您可以變更 `openwbem.conf` 檔案中的任何一項設定。本節將討論下列組態設定：

- 第 11.2.1 節「變更驗證設定」[226頁]
- 第 11.2.2 節「變更憑證設定」[233頁]
- 第 11.2.3 節「變更埠設定」[234頁]
- 第 11.2.4 節「變更預設的記錄設定」[235頁]
- 第 11.2.5 節「設定除錯記錄」[243頁]

- 第 11.2.6 節「設定其他記錄」[244頁]

11.2.1 變更驗證設定

您可以在變更驗證設定時控制下面幾個項目：

- 誰可以存取此 CIMOM
- 使用哪項驗證模組

請檢視下面設定：

- 章節「http_server.allow_local_authentication 」[226頁]
- 章節「http_server.digest_password_file 」[227頁]
- 章節「http_server.ssl_client_verification 」[228頁]
- 章節「http_server.ssl_trust_store 」[229頁]
- 章節「http_server.use_digest」[229頁]
- 章節「owcimomd.ACL_superuser」[230頁]
- 章節「owcimomd.allow_anonymous」[230頁]
- 章節「owcimomd.allowed_users」[231頁]
- 章節「owcimomd.authentication_module」[232頁]
- 章節「simple_auth.password_file」[232頁]

http_server.allow_local_authentication

目的、用途

依據本地系統檔案許可權的不同，指示 http_server 允許不用提供密碼的本地驗證。

您可以配合「基本」或「摘要」設定來使用這項設定。

語法

```
http_server.allow_local_authentication = option
```

選項	描述
true	啟用本地驗證。 此為預設值。
false	停用本地驗證。

範例

```
http_server.allow_local_authentication = true
```

http_server.digest_password_file

目的、用途

指定密碼檔案的位置。在 http_server.use_digest 設定已經啟用的情況下，必須設定這個項目。

語法

```
http_server.digest_password_file = path_filename
```

下面是摘要密碼檔案的預設路徑和檔名：

```
/etc/openwbem/digest_auth.passwd
```

範例

```
http_server.digest_password_file =
/etc/openwbem/digest_auth.passwd
```

http_server.ssl_client_verification

目的、用途

決定伺服器是否要嘗試以 SSL 用戶端憑證確認來驗證用戶端。

這項設定已預設為停用。

語法:

```
http_server.ssl_client_verification = option
```

選項	描述
autoupdate	指定此同一個功能為「*選擇性*」選項；但是，之前通過 HTTP 驗證的不明用戶端憑證會加入到信任儲存區 (truststore)，因此，後來使用相同憑證的用戶端連線就不需要進行 HTTP 驗證。
disabled	停用用戶端憑證檢查。 此為預設值。
optional	允許受信任憑證進行驗證 (不用進行任何 HTTP 驗證)。 也可以允許不受信任憑證在用戶端通過 HTTP 驗證時通過 SSL 信號交換。
required	要求 SSL 信號交換的受信任憑證成功通過。

範例

```
http_server.ssl_client_verification = disabled
```

http_server.ssl_trust_store

目的、用途

指定包含此 OpenSSL 信任儲存區的目錄。

語法

```
http_server.ssl_trust_store = path
```

下面是此信任儲存區檔案的預設路徑。

```
/etc/openwbem/truststore
```

範例

```
http_server.ssl_trust_store = /etc/openwbem/truststore
```

http_server.use_digest

目的、用途

指示 HTTP 伺服器使用會略過「基本」驗證機制的「摘要」驗證。若要使用摘要機制，您必須使用 `owdigestgenpass` 來設定摘要密碼。

摘要機制不會使用 owcimomd.authentication_module 組態設定所指定的驗證模組。

語法

```
http_server.use_digest = option
```

選項	描述
false	啟用基本驗證機制。 此為預設值。

選項	描述
true	停用基本驗證機制。

範例

```
http_server.use_digest = false
```

owcimomd.ACL_superuser

目的、用途

指定可以存取 owcimomd 維護的全部名稱空間中所有 CIM 資料的使用者名稱。這個使用者可以用來管理 `/root/security` 名稱空間，也就是所有 ACL 使用者權限的儲存位置。

ACL 處理要在 `OpenWBEM_Acl1.0.mof` 檔案輸入之後才會啟用。

語法

```
owcimomd.ACL_superuser = username
```

範例

```
owcimomd.ACL_superuser = root
```

owcimomd.allow_anonymous

目的、用途

啟用或是停用 owcimomd 的匿名登入。

語法

```
owcimomd.allow_anonymous = option
```

選項	描述
false	要求透過使用者名稱和密碼來存取 owcimomd 資料。 這是預設以及建議使用的設定。
true	允許匿名登入 owcimomd。 這個選項會停用驗證。存取 owcimomd 資料時不需使用任何使用者名稱或是密碼。

範例

```
owcimomd.allowed_anonymous = false
```

owcimomd.allowed_users

目的、用途

指定允許存取 owcimomd 資料的使用者清單。

語法

```
owcimomd.allowed_users = option
```

選項	描述
使用者名稱	指定允許存取 owcimomd 資料的一位或是多位使用者。 每個使用者名稱之間要以空格隔開。 根使用者是預設值。
*	允許所有使用者進行驗證 (例如，當您選擇改用 ACL 來控制存取的情況下)。 所有驗證方法都會強制使用這個選項，除非 owcimomd.allow_anonymous 是設定成 `True`。

範例

```
owcimomd.allowed_users = bcwhitely jkcarey jlanderson
```

owcimomd.authentication_module

目的、用途

指定 owcimomd 所使用的驗證模組。這個設定應該是包含此驗證模組之共用程式庫的絕對路徑。

語法

```
owcimomd.authentication_module = path_filename
```

下面是驗證模組的預設路徑和檔名：

```
/usr/lib/openwbem/authentication/libpamauthentication.so
```

範例

```
owcimomd.authentication_module =
/usr/lib/openwbem/authentication/libpamauthentication.so
```

simple_auth.password_file

目的、用途

指定在使用簡單驗證模組時的密碼檔案路徑。

這項設定已預設為停用。

語法

```
simple_auth.password_file = path_filename
```

範例

```
simple_auth.password_file =
/etc/openwbem/simple_auth.passwd
```

11.2.2 變更憑證設定

http_server.SSL_cert 和 http_server.SSL_key 設定會指定包含主機私密金鑰和 OpenSSL 用來進行 HTTPS 通訊之憑證的檔案 (或多個檔案) 位置。

此 .pem 檔案會存放在下面的預設位置：

```
/etc/openwbem/servercert.pem
```

```
/etc/openwbem/serverkey.pem
```

語法

```
http_server.SSL_cert = path_filename
```

或

```
http_server.SSL_key = path_filename
```

注

金鑰和憑證可以存放在相同檔案。在這種情況下，http_server.SSL_cert 和 http_server.SSL_key 會使用相同值。

範例

```
http_server.SSL_cert = /etc/openwbem/servercert.pem
```

```
http_server.SSL_key = /etc/openwbem/servercert.pem
```

```
http_server.SSL_key = /etc/openwbem/serverkey.pem
```

11.2.3 變更埠設定

http_server.http_port 和 server.https_port 設定會指定 owcimomd 要在其上傾聽所有 HTTP 和 HTTPS 通訊的埠號碼。

語法

```
http_server.http_port = option
```

或

```
http_server.https_port = option
```

選項	描述
Specific_port_number	指定 HTTP 或是 HTTPS 通訊的特定埠。 對於 HTTP 通訊，預設埠是 5988。 對於 HTTPS 通訊，預設埠是 5989。
-1	停用 HTTP 或 HTTPS 連接 (例如，當您只希望支援 HTTPS 連接時)。
0	在執行時期動態指定埠號碼。

範例

這些設定會停用 HTTP 埠，以及啟用提供 HTTPS 通訊使用的埠 5989。

```
http_server.http_port = -1

http_server.https_port = 5989
```

11.2.4 變更預設的記錄設定

下面在 `owcimomd.conf` 檔案中的記錄設定可以讓您指定記錄位置和進行幾次記錄，以及記錄的錯誤類型、記錄大小、檔名和格式：

- 章節「log.main.categories」[235頁]
- 章節「log.main.components」[236頁]
- 章節「log.main.format」[237頁]
- 章節「log.main.level」[239頁]
- 章節「log.main.location」[240頁]
- 章節「log.main.max_backup_index」[241頁]
- 章節「log.main.max_file_size」[241頁]
- 章節「log.main.type」[242頁]

如果要設定除錯記錄功能，請參閱第 11.2.5 節「設定除錯記錄」[243頁]。

如果要設定其他記錄功能，請參閱第 11.2.6 節「設定其他記錄」[244頁]。

log.main.categories

目的、用途

指定記錄輸出的類別。

語法

`log.main.categories = `*`選項`*

選項	描述
`category_name`	指定使用空格分隔清單記錄的類別。

選項	描述
	下面是 owcimomd 所使用的類別： • DEBUG • ERROR • FATAL • INFO 如需關於這些選項的詳細資訊，請參閱章節「log.main.level」[239頁]。 如果在這個選項中進行了指定，系統就不會依據層級來處理預定義的類別，而是當作獨立類別來處理。沒有可用的預設設定；而且如果沒有設定類別，系統就不會記錄任何類別，同時會使用 log.main.level 設定。
*	記錄全部的類別。 此為預設值。

範例

```
log.main.categories = FATAL ERROR INFO
```

log.main.components

目的、用途

指定記錄輸出的元件。

語法

```
log.main.components = 選項
```

選項	描述
component_name	使用空格分隔清單指定要記錄的元件 (例如 owcimomd)。 提供者可以使用自己的元件。
*	指定記錄全部的元件。 此為預設值。

範例

```
log.main.components = owcimomd nssd
```

log.main.format

目的、用途

指定記錄訊息的格式 (混用 printf() 樣式轉換規範的文字)。

語法

```
log.main.format = conversion_specifier
```

選項	指定
%%	%
%c	元件 (例如 owcimomd)
%d	日期 後面可以接上包含在括號中的日期格式規範。例如，%d{%H:%M:%S} or %d{%d %b %Y %H:%M:%S}。如果沒有提供日期格式規範，系統會假設使用 ISO 8601 格式。

選項	指定
	唯一新增的是 %Q，它代表毫秒數目。 如需有關日期格式規範的詳細資訊，請參閱可在 <ctime> 標頭中找到之 strftime() 函式的說明文件。
%e	以 XML CDATA 格式記錄訊息。這種格式包括「<![CDATA[“ and ending “]]>」
%F	檔名
%l	檔名和行號。例如，file.cpp(100)
%L	行號
%M	發出記錄請求的方法名稱 (僅在支援 __PRETTY_FUNCTION__ 或 C99 的 __func__ 的 C++ 編譯器中有效)。
%m	Message
%n	平台相關行分隔字元 (\n) 或字元 (\r\n)。
%p	類別，又稱為層級或是優先程度。
%r	從應用程式啟動到記錄事件建立之間所經過的毫秒數目。
%t	線串 ID
\n	新線
\t	索引標籤
\r	換行字元
\\	\
\x<hexDigits>	以十六進位代表的字元

您可以變更最小欄位寬度、最大欄位寬度和對齊。選擇性格式修飾詞可以放在百分比符號 (%) 和轉換字元之間。第一個選擇性格式修飾詞為靠左對齊旗標，是一個減號 (-) 字元。跟在後面的為選擇性的最小欄位寬度修飾詞，是代表要輸出之最小字元數目的整數。如果此資料項目要求使用較少的字元，它就會在左邊或是右邊塞滿空格，這將由對齊旗標來決定。如果此資料項目大於最小欄位寬度，此欄位就會進行擴充來符合資料使用。

最大欄位寬度修飾詞的指定方式，是在英文句號 (.) 後面加上十進位常數。如果此資料項目比最大欄位還要長，系統就會從資料項目的開頭 (根據預設) 或是從結束位置 (當指定靠左對齊旗標時) 移除多出的字元。

範例

Log4j TTCC 配置：

```
"%r [%t] %-5p %c - %m"
```

類似 TTCC，但是還包含一些固定大小的欄位：

```
"%-6r [%15.15t] %-5p %30.30c - %m"
```

符合 log4j.dtd 1.2 的 XML 輸出，可由 Chainsaw 進行處理 (如果使用這種配置，這項資料必定是出現在單行中；它會配合可讀性而在此加以分割)：

```
"<log4j:event logger="%c" timestamp="%d{%s%Q}" level="%p"
thread="%t"> <log4j:message>%e</log4j:message>
<log4j:locationInfo class="" method="" file="%F"
line="%L"/></log4j:event>"
```

下面是預設設定：

```
log.main.format = [%t]%m
```

log.main.level

目的、用途

指定記錄輸出的層級。如果已設定層級，此記錄就會輸出指定層級和更高層級的所有預定義類別。

語法

```
log.main.level = 選項
```

選項	描述
DEBUG	記錄所有 Debug、Info、Error 和 Fatal 類別的錯誤訊息。
ERROR	記錄所有 Error 和 Fatal 類別的錯誤訊息。 此為預設值。
FATAL	僅記錄 Fatal 類別的錯誤訊息。
INFO	記錄所有 Info、Error 和 Fatal 類別的錯誤訊息。

範例

```
log.main. level = ERROR
```

log.main.location

目的、用途

指定當 log.main.type 設定選項已指定記錄要送到檔案時，owcimomd 所要使用的記錄檔案位置。

語法

```
log.main.location = path_filename
```

範例

```
log.main.location = /system/cimom/var/owcimomd.log
```

log.main.max_backup_index

目的、用途

指定在最舊記錄去除之前要保留的備份記錄數量。

語法

```
log.main.backup_index = option
```

選項	描述
unsigned_integer_above_0	指定要保留的備份記錄數量。 預設設定是 1 個記錄檔案。
0	不備份任何記錄，而且當記錄到達最大檔案大小時就截斷該記錄。

範例

```
log.main.max_backup_index = 1
```

log.main.max_file_size

目的、用途

指定 owcimomd 記錄功能可以增大成為的最大大小 (單位是 KB)。

語法

```
log.main.max_file_size = option
```

選項	描述
unsigned _integer_in_KB	限制記錄為特定大小 (單位是 KB)。

選項	描述
0	不限制記錄的增大大小。 此為預設值。

範例

```
log.main.max_file_size = 0
```

log.main.type

目的、用途

指定 owcimomd 使用的主要記錄類型。

語法

```
log.main.type = 選項
```

選項	描述
file	將所有訊息傳送到 log.main.location 組態設定所指定的檔案。
null	停用記錄。
syslog	將所有訊息傳送到 syslog 介面。 此為預設值。

範例

```
log.main.type = syslog
```

11.2.5 設定除錯記錄

如果 owcimomd 正在執行除錯模式，此除錯記錄功能就會執行下列設定：

- `log.debug.categories = *`
- `log.debug.components = *`
- `log.debug.format = [%t] %m`
- `log.debug.level = *`
- `log.debug.type = stderr`

標示顏色的除錯記錄

如果要使用彩色版本的除錯記錄功能，請使用下列 ASCII 逸出程式碼：

```
log.debug.format =
\x1b[1;37;40m[\x1b[1;31;40m%-.6t\x1b[1;37;40m]\x1b[1;32;40m
%m\x1b[0;37;40m
```

如果要使用其他顏色，請透過 `log.debug.format` 指令來使用下列色碼：

表格 11.3 *用於 log.debug.format 指令的其他色碼*

色彩	代碼
紅色	\x1b[1;31;40m
深紅色	\x1b[0;31;40m
綠色	\x1b[1;32;40m
深綠色	\x1b[0;32;40m
黃色	\x1b[1;33;40m
深黃色	\x1b[0;33;40m

色彩	代碼
藍色	\x1b[1;34;40m
深藍色	\x1b[0;34;40m
紫色	\x1b[1;35;40m
深紫色	\x1b[0;35;40m
青色	\x1b[1;36;40m
深青色	\x1b[0;36;40m
白色	\x1b[1;37;40m
灰白色	\x1b[0;37;40m
灰色	\x1b[0;37;40m
重設色彩	\x1b[0;37;40m

11.2.6 設定其他記錄

如果要建立其他記錄，請在這個設定下面列出記錄名稱：

`owcimomd.additional_logs = logname`

分隔多個記錄名稱空間。

語法

`owcimomd.additional_logs = logname`

針對每個記錄套用下列設定：

- `log.log_name.categories`

- `log.log_name.components`
- `log.log_name.format`
- `log.log_name.level`
- `log.log_name.location`
- `log.log_name.max_backup_index`
- `log.log_name.max_file_size`

範例

```
owcimomd.additional_logs = errorlog1 errorlog2 errorlog3
```

11.3 如需更多資訊

如需有關 OpenWBEM 的詳細資訊，請參閱下列資訊：

- 本地伺服器檔案系統之 `usr/share/doc/packages/openwbem` 中的文件：
  - `readme`
  - `openwbem-faq.html`
- Novell Cool Solutions 文章：介紹 SUSE Linux 中的 WBEM 與 OpenWBEM [http://www.novell.com/coolsolutions/feature/14625.html]
- OpenWBEM 網站 [http://www.openwbem.org]
- DMTF 網站 [http://www.dmtf.org]

12 IP 網路 — iSCSI 上的大型存放設備

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。在主機磁區中為了此目的通常會使用光纖通道。到目前為止，UNIX 電腦及大部分的伺服器尚未連接到中央儲存解決方案。

linux-iSCSI 是將 Linux 電腦與中央儲存系統連接在一起的一個簡單且低廉的解決方案。原則上，iSCSI 代表在 IP 層級上傳輸 SCSI 指令。如果程式開始查詢這類設備，作業系統會產生必要的 SCSI 指令。接著，系統會依照通稱為 *iSCSI 啟動程式* (iSCSI Initiator) 的軟體所需，將這些指令嵌入 IP 封包並加密，然後將這些封包傳送到對應的 iSCSI 遠端工作站，也稱為 *iSCSI 目標* (iSCSI Target)。

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，設定針對檔案系統服務最佳化的 Linux 伺服器是很重要的。iSCSI 目標只會存取 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，並使用大量記憶體來提高資料快取。如需關於 RAID 的詳細資訊，請參閱第 7.2 節「軟體 RAID 組態」[111 頁]。

12.1 設定 iSCSI 目標

SUSE® Linux Enterprise Server 隨附由 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。

12.1.1 使用 YaST 建立 iSCSI 目標

iSCSI 目標組態會將現有區塊設備或檔案系統影像輸出到 iSCSI 啟動程式。請先使用 YaST 建立需要的區塊設備，或建立檔案系統影像。如需磁碟分割的綜覽，請參閱第 8.5.7 節「使用 YaST 磁碟分割程式」[141 頁]。檔案系統影像必須以手動方式建立。例如，如果要建立容量 4GB 的 `/var/lib/xen/images/xen-0` 影像，請先確認該目錄已存在，然後建立影像本身：

```
mkdir -p /var/lib/xen/images
dd if=/dev/zero of=/var/lib/xen/images/xen-0 seek=1M bs=4096 count=1
```

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。注意：您不能使用有效的主機名稱，但必須使用 IP 位址。如需 `iSNS` 相關的詳細資訊，請閱讀第 13 章「*iSNS for Linux 綜覽*」[257 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 提供更多有關驗證的詳細資訊 (請參閱 `http://www.ietf.org/rfc/rfc3720.txt`)。

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。RFC 3722 提供更多有關命名慣例的詳細資訊 (請參閱 `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*」。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。請按一下「*完成*」啟用變更。

12.1.2 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。

若您已存取了 `iSNS` 伺服器，首先需要設定將此伺服器告知該目標。注意：`iSNS` 伺服器的位址必須始終以 IP 位址提供。正常的網域名稱不夠。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 `iSNS` 相關的詳細資訊，請閱讀 第 13 章「*iSNS for Linux 綜覽*」[257頁]。注意：`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 驗證可以朝兩個方向來完成。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 啟動程式也可以要求 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。這個區段的開頭固定是 `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，而且識別碼可以自由選取。RFC 3722 提供更多有關命名慣例的詳細資訊 (請參閱 http://www.ietf.org/rfc/rfc3722.txt)。本例中輸出三個不同的區塊設備。第一個是邏輯磁碟區 (請參閱第 7.1 節「LVM 組態」 [103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。請參閱 ietd.conf 的手冊頁，以瞭解詳細資料。

/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
```

12.1.3 使用 ietadm 設定線上目標

如有必要變更 iSCSI 目標組態，您一定要重新啟動目標，才能啟用在組態檔案中所做的變更。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。這會提供類似以下資訊：

```
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。要知道，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
```

這些參數全都可以輕易變更。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：使用 ietadm 根據變更更新 ietd.conf

您用 `ietadm` 指令進行的變更對系統不具有永久效力。如果不加到 `/etc/ietd.conf` 檔案中，下次重新開機時，這些變更都會消失不見。依您網路的 iSCSI 使用方式而定，這可能導致嚴重的問題。

ietadm 指令還有許多其他選項可供使用。如需綜覽，請使用 `ietadm -h`。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。

12.2 設定 iSCSI 啟動程式

iSCSI 啟動程式也稱為用戶端，它可用來連接任何 iSCSI 目標。這不僅僅限於上述 iSCSI 目標解決方案。iSCSI 啟動程式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。

12.2.1 使用 YaST 設定 iSCSI 啟動程式的組態

組態分為三個索引標籤。「*服務*」索引標籤可用來在開機時啟用 iSCSI 啟動程式。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`。「*連接的目標*」索引標籤會提供目前已連接 iSCSI 目標的綜覽。它和「*探查的目標*」索引標籤一樣，提供為系統新增新目標的選項。

最開始請從「*探查的目標*」索引標籤開始。它提供在網路上探查 iSCSI 目標的可能性。

1. 使用「*探查*」開啟探查對話方塊。
2. 輸入 IP 位址，並視需要變更連接埠。
3. 若有需要，請新增「*內送*」或「*外送*」驗證。
4. 按一下「*下一步*」開始探查。

探查成功後，使用「*登入*」啟用目標。系統會詢問使用所選 iSCSI 目標的驗證資訊。「*按一下「*」下一步*」以完成組態。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。請用 `lsscsi` 尋找實際設備：

```
lsscsi
[1:0:0:0]   disk    IET      VIRTUAL-DISK     0     /dev/sda
```

12.2.2 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。請參閱第 12.2.4 節「iSCSI 用戶端資料庫」[255頁]，了解如何取得更多有關儲存資料的詳細資訊。

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。

12.2.3 在 iSCSI 設備上設定 LVM 自動組件

由於 udev 支援 LVM 啟動，因此當偵測到所有需要的實體磁碟區之後，各 LVM 磁碟區群組就會透過 udev 自動啟動。

udev 中的 LVM 自動組件會使用 udev 輔助程式 `collect`。此程式會將要檢查的抽象 `ID` 做為第一個引數，後跟元件 ID 的清單。如果呼叫此程式時，每個元件 ID 都是做為第一個引數，則程式會傳回 0。

因此，對於自動組件，指定磁碟區群組的實體磁碟區 UUID 會註冊為 `collect` 的引數清單。udev (或 vol_id) 能夠偵測出設備上的實體磁碟區 UUID，因此可做為第一個引數傳遞至 `collect`。

當已使用所有實體磁碟區 UUID 呼叫 `collect` 後 (即 udev 已收到所有元件設備的事件)，下一個規則就會觸發，呼叫 `vgchange -a y <vgname>`，該磁碟區群組即會啟動。

設定方法

使用程序檔 `/usr/share/doc/packages/lvm2/lvm-vg-to-udev-rules.sh`。該程序檔會將您希望自動啟動的磁碟區群組做為引數，並會產生所需的 udev 規則。現在，重新啟動 iSCSI 即可啟動磁碟區群組。若要在開機時自動啟動陣列，您必須將 iSCSI 元件設備切換為 `automatic`，以便啟動器在開機時自動登入目標。

12.2.4 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 作業可移除過時的資料集。如果目標 iqn.2006-02.com.example.iserv:systems 不再是有效記錄，請使用 iscsiadm -m node -n iqn.2006-02.com.example.iserv:systems

`--op=delete` 指令刪除這筆記錄。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，請執行 `iscsiadm -m node` 指令。

12.2.5 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：

- http://www.open-iscsi.org/
- http://www.open-iscsi.org/cgi-bin/wiki.pl
- 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` 的手冊頁，以及 `/etc/iscsid.conf` 組態檔案範例。

13 iSNS for Linux 綜覽

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。

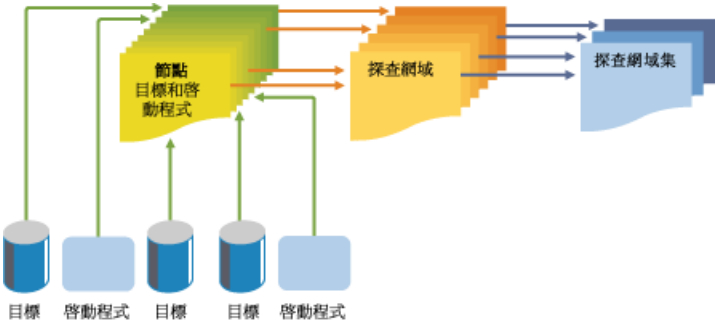
網際網路儲存名稱服務 (iSNS) 是一項可透過 SUSE Linux Enterprise Server (SLES) 10 SP 2 取得的標準服務。iSNS 有助於自動探查、管理及設定 TCP/IP 網路上的 iSCSI 設備。iSNS 提供可與光纖通道媲美的智能儲存探查與管理服務。

13.1 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 13.1　iSNS 探查網域與探查網域集



iSCSI 目標與啟動程式均使用 iSNS 用戶端啟動使用 iSNS 通訊協定與 iSNS 伺服器達成的交易。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。這樣可能會讓探查與連接成為一場惡。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。結果是特定部門中的 iSCSI 啟動程式僅探查作為部門探查網域組成部分的 iSCSI 目標。

13.2 iSNS for Linux 安裝與設定

iSNS for Linux 已包含在 SLES 10 SP2 中，但預設不會進行安裝或設定。您必須安裝 iSNS 套件模組 (isns 與 yast2-isns 模組) 並設定 iSNS 以使用。

注

iSNS 可作為 iSCSI 目標或驅動程式安裝在相同的伺服器上。在同一台機器上使用 iSCSI 目標與驅動程式不受支援。

安裝 iSNS for Linux：

1. 啟動 YaST，然後選取「*軟體管理*」。
2. 在「*搜尋*」欄位中，輸入 `isns`。
3. 選取 isns 與 yast2-isns 套件，然後按一下「*接受*」。

13.3 設定 iSNS

iSNS必須在伺服器啟動。您可以透過在安裝伺服器之伺服器主控台輸入`rcisns start`或`/etc/init.d/isns start`執行此作業。您也可以使用 iSNS 的停止、狀態與重新啟動選項。

iSNS 也可設定為自動在每次伺服器重新啟動時啟動。執行的操作

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. 在畫面的服務啟動部分中，選取「*啟動時*」。

   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

13.3.1 建立 iSNS 探查網域

若要讓 iSCSI 啟動程式與 iSCSI 目標使用 iSNS 服務，則它們必須屬於探查網域。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，然後再按「*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。
3. 指定您正在建立的探查網域之名稱，然後按一下「*確定*」。

13.3.2 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，然後再按「*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。
3. 指定您正在建立的探查網域集之名稱，然後按一下「*確定*」。

13.3.3 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 N節點*」索引標籤，並確定已列出要使用iSNS服務的iSCSI目標與啟動程式。

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。

重新啟動 SCSI 服務或重新啟動伺服器時，iSCSI 節點會再次自動新增至清單 (iSNS 資料庫)，除非您移除 iSCSI 組態檔的 iSNS 部分或提供注解。

3 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。

4 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*」。

5 為新增至探查網域的節點重複最後一步，當您完成新增節點時，再按一下「*完成*」。

一個 iSCSI 節點可屬於多個探查網域。

13.3.4 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。

13.4 如需更多資訊

linuxisns項目的代管位置為 http://sourceforge.net/projects/linuxisns/。關於這個項目的郵件清單，請造訪 http://sourceforge.net/mailarchive/forum.php?forum_name=linuxisns-discussion。iSNS 相關的一般資訊在 rfc4171 中寫入。並請參閱 http://www.ietf.org/rfc/rfc4171。

14 Oracle Cluster File System 2

Oracle Cluster File System 2 (OCFS2) 是一般用途的日誌式檔案系統，與 Linux 2.6 和更新版本的核心完全整合。OCFS2 可讓您將設備上的應用程式二進位檔案、資料檔案和資料庫儲存於 SAN。叢集中所有節點均同時具有檔案系統的讀取與寫入權限。分散式鎖定管理員可幫助避免檔案存取權衝突。OCFS2 的每個目錄最多可支援 32,000 個子目錄，以及數百萬個檔案。O2CB 叢集服務 (驅動程式) 會在各節點上執行，以管理叢集。

OCFS2 新增至 SUSE Linux Enterprise Server 9 以支援 Oracle Real Application Cluster (RAC) 資料庫及其應用程式檔案 Oracle Home。在 SUSE Linux Enterprise Server 10 與更新版本，可使用 OCFS2 做為下列任一儲存解決方案之一：

- Oracle RAC 與其他資料庫
- 一般應用程式與負荷
- 叢集中的 XEN 影像儲存

  XEN 虛擬機器與虛擬伺服器可儲存於由叢集伺服器裝載的 OCFS2 磁碟區，以在伺服器間提供快速而便捷的可攜式 XEN 虛擬機器。

- LAMP (Linux、Apache、MySQL 和 PHP | PERL | Python) 堆疊

此外，其亦與 Heartbeat 2 完全整合。

OCFS2 身為高效能、對稱、平行叢集檔案系統，支援下列功能：

- 叢集上的所有節點都可使用應用程式的檔案。使用者只需在叢集上的 OCFS2 磁碟區安裝一次即可。

- 所有節點可直接透過標準檔案系統介面同時讀取與寫入儲存區，讓執行於叢集上的應用程式易於管理。
- 「分散式鎖定管理員 (DLM)」會協調檔案存取權。

  DLM 控制對於大部分狀況都非常實用，但應用程式的設計與 DLM 競爭檔案存取權協調能力的話，其擴充性可能會受限。
- 所有後端儲存區均可使用儲存區備份功能。您可輕鬆建立共享應用程式檔案的複本，以利於提供有效的災難復原。

OCFS2 亦提供下列功能：

- 中繼資料快取
- 中繼資料日誌
- 跨節點資料檔案一致性
- 透過 `ocfs2console` 公用程式進行 GTK GUI 式管理
- 以共享根檔案系統形式操作
- 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
- 支援高達 255 個叢集節點
- 對指定節點的本機檔案提供內容相關符號連結 (CDSL) 支援
- 對資料庫檔案提供非同步的直接 I/O 支援，以加強資料庫效能

14.1 O2CB 叢集服務

O2CB 業集服務是一組模組以及管理 OCFS2 服務與磁碟區所需的記憶體內檔案系統。您可以設定在系統開機時載入並裝載這些模組。如需指示，請參閱第 14.6.2 節「設定 OCFS2 服務」[269頁]。

表格 14.1　O2CB 叢集服務堆疊

服務	描述
節點管理 (NM)	追蹤 `/etc/ocfs2/cluster.conf` 檔案中的所有節點
Heartbeat (HB)	當節點加入或離開業集時，發出上/下通知
TCP	以 TCP 通訊協定處理節點間的通訊
分散式鎖定管理員 (DLM)	追蹤所有鎖定、其擁有者，及狀態
CONFIGFS	使用者空間組態檔案系統。如需詳細資料，請參閱第 14.3 節「記憶體內檔案系統」[266頁]。
DLMFS	核心空間 DLM 的使用者空間介面。如需詳細資料，請參閱第 14.3 節「記憶體內檔案系統」[266頁]。

14.2 磁碟心跳

OCFS2 需要網路上節點均在作用中。O2CB 業集服務會定期傳送 keepalive 套件，以確定節點均在作用中。其在節點間使用的是私人連線而不是 LAN，以避免網路延遲而被解讀成節點消失，導致節點自我防護。

OC2B 業集服務會透過磁碟心跳傳遞節點狀態。心跳系統檔案位於 Storage Area Network (SAN) 中，業集內所有節點均可使用。檔案中的區塊指定後續會與各節點的插槽指定相關聯。

各節點會讀取檔案，並以兩秒間隔寫入檔案中所指派的區塊。節點時戳的變更表示節點在作用中。若節點在指定的連續間隔數內 (稱為心跳臨界值) 均未寫入心跳檔案，表示節點死亡。即使只有單一節點存活，O2CB 叢集也必須執行此檢查，因為隨時都可能有其他節點動態加入。

您可在 `/etc/sysconfig/o2cb` 檔案中使用 `O2CB_HEARTBEAT_THRESHOLD` 參數修改磁碟心跳臨界值。等待時間的計算方式如下：

```
(O2CB_HEARTBEAT_THRESHOLD value - 1) * 2 = threshold in seconds
```

例如，若 O2CB_HEARTBEAT_THRESHOLD 值設為預設值 7，則等待時間為 12 秒 ((7 - 1) * 2 = 12)。

14.3 記憶體內檔案系統

OCFS2 使用兩個記憶體內檔案系統進行通訊：

表格 14.2 OCFS2 使用的記憶體內檔案系統

記憶體內檔案系統	描述	裝載點
configfs	將叢集內節點清單傳遞至核心內節點管理員，並將心跳所使用的資源傳遞至核心內心跳線串	/config
ocfs2_dlmfs	將全叢集內對資源的鎖定與解除鎖定傳遞至核心內分散式鎖定管理員，其會追蹤所有鎖定及其擁有者與狀態	/dlm

14.4 管理公用程式與指令

OCFS2 會在節點上儲存節點特定的參數檔案。業集組態檔案 (/etc/ocfs2/cluster.conf) 位於指派至業集的各節點上。

ocfs2console 公用程式是管理業集中 OCFS2 服務組態的 GTK GUI 式介面。使用此公用程式可以設定並儲存業集所有成員節點的 /etc/ocfs2/cluster.conf 檔案。此外，您亦可使用此公用程式以格式化、微調、裝載與解除裝載 OCFS2 磁碟區。

重要

ocfs2console 公用程式中的檔案瀏覽器欄透過叢集的速度很慢並且不一致。建議您使用 ls(1) 指令來列出檔案。

下表將說明其他 OCFS2 公用程式。如需關於這些指令的語法資訊，請參閱其線上文件。

表格 14.3 *OCFS2 共用程式*

OCFS2 共用程式	描述
debugfs.ocfs2	以偵錯為目的，檢驗 OCFS 檔案系統。
fsck.ocfs2	檢查檔案系統是否有錯誤，並選擇性修復錯誤。
mkfs.ocfs2	在設備上建立 OCFS2 檔案系統，通常是共用實體或邏輯磁碟上的分割區。此工具需要啟動 O2CB 叢集服務。
mounted.ocfs2	偵測並列出業集系統上的所有 OCFS2 磁碟區。偵測並列出裝載 OCFS2 裝置的系統上之所有節點，或列出所有 OCFS2 裝置。
ocfs2cdsl	為節點的指定檔案名稱 (檔案或目錄) 建立內容相關符號連結 (CDSL)。CDSL 檔案名稱具有其自身對特定節點的複本，但在 OCFS2 中具有一般名稱。
tune.ocfs2	變更 OCFS2 檔案系統參數，包括磁碟區標籤、節點插槽數目、所有節點插槽的日至大小，以及磁碟區大小。

請使用下列指令管理 O2CB 服務。如需關於 `o2cb` 指令語法的更多資訊，請參閱其線上文件。

表格 14.4 *O2CB 指令*

指令	描述
/etc/init.d/o2cb status	報告是否已載入或裝載 o2cb 服務
/etc/init.d/o2cb load	載入 O2CB 模組與記憶體內檔案系統
/etc/init.d/o2cb online *ocfs2*	稱為 ocfs2 的叢集上線 此叢集中至少需有一節點作用中，該叢集才能上線。

指令	描述
/etc/init.d/o2cb offline *ocfs2*	稱為 ocfs2 的叢集離線
/etc/init.d/o2cb unload	卸載 O2CB 模組與記憶體內檔案系統
/etc/init.d/o2cb start *ocfs2*	若叢集設定為開機時載入，則藉由載入 `o2cb` 並將叢集連上線，啟動名為 ocfs2 的叢集 此叢集中至少需有一節點作用中，該叢集才能上線。
/etc/init.d/o2cb stop *ocfs2*	若叢集設定為開機時載入，則藉由使叢集離線並卸載 O2CB 模組與記憶體內檔案系統，停止名為 ocfs2 的叢集

14.5 OCFS2 套件

SUSE Linux Enterprise Server 10 和更新版本會自動安裝 OCFS2 核心模組 (`ocfs2`)。若要使用 OCFS2，請使用 YaST (或指令行) 在叢集內各節點安裝 `ocfs2-tools` 和 `ocfs2console` 套件。

1. 以 `root` 使用者身份或同等地位登入，然後開啟 YaST 控制中心。
2. 選取「*軟體*」>「*軟體管理*」。
3. 在「*搜尋*」欄位中，輸入 `ocfs2`。

   右邊窗格中將列出軟體套件 `ocfs2-tools` 和 `ocfs2console`。若已選擇這些套件，則已安裝這些套件。
4. 若您需要安裝套件，請選擇套件，再按「*安裝*」並遵循畫面上的指示。

14.6 建立 OCFS2 磁碟區

遵循本節中的程序，設定您的系統使用 OCFS2 並建立 OCFS2 磁碟區。

14.6.1 先決條件

開始之前，請執行下列步驟：

- 在SAN磁碟上依需要啟始化、分割或設定RAID(獨立磁碟容錯陣列)，以準備計劃為 OCFS2 磁碟區使用的設備。將裝置留為可用空間。

  建議您將應用程式檔案與資料檔案儲存於不同的 OCFS2 磁碟區，但唯有在應用程式磁碟區與資料磁碟區具有不同的裝載需求時，才需強制執行此動作。例如，Oracle RAC 資料庫磁碟區需要 `datavolume` 與 `nointr` 裝載選項，但 Oracle Home 磁碟區不得使用這些選項。

- 請確認已安裝 `ocfs2console` 和 `ocfs2-tools` 套件。若尚未安裝的話，請使用 YaST 或指令行方法進行安裝。如需 YaST 說明，請參閱第 14.5 節「OCFS2 套件」[268頁]。

14.6.2 設定 OCFS2 服務

建立 OCFS2 磁碟區之前，必須先設定 OCFS2 服務。透過下列程序，您可以產生 `/etc/ocfs2/cluster.conf` 檔案，將 `cluster.conf` 檔案儲存於所有節點上，並建立與啟動 O2CB 業集服務 (`o2cb`)。

請遵循此節程序設定叢集中的一節點。

1. 開啟終端機視窗，並以 `root` 使用者身份登入。
2. 如果還未啟用 `o2cb` 叢集服務，請輸入 `chkconfig --add o2cb`。

   新增新服務時，`chkconfig` 會確認該服務在每個 run level 中均具有 start 或 kill 項目。
3. 如果還未啟用 `ocfs2` 服務，請輸入 `chkconfig --add ocfs2`。
4. 設定 `o2cb` 叢集服務驅動程式在開機時載入。

   4a 輸入 `/etc/init.d/o2cb configure`

   4b 在 `Load O2CB driver on boot (y/n) [n]` 提示中，輸入 `y`(是)以在開機時載入。

4c 在 `Cluster to start on boot (Enter "none" to clear) [ocfs2]` 提示中，輸入 `none`。此選項假定您是第一次設定 OCFS2 或重新設定服務。設定 `/etc/ocfs2/cluster.conf` 檔案時，會在下個步驟指定業集名稱。

5 使用 `ocfs2console` 公用程式設定並儲存 `/etc/ocfs2/cluster.conf` 檔案到業集所有成員節點上。

業集所有節點中的此檔案均應相同。請使用下列步驟設定第一個節點。稍候您可使用 `ocfs2console`，動態新增新節點，並將修改過的 `cluster.conf` 檔案傳播到所有節點。

然而，若您變更其他設定，如叢集名稱或 IP 位址，則必須重新啟動叢集讓變更生效，如步驟 6 [270頁]中所述。

5a 輸入 `ocfs2console` 以開啟 `ocfs2console` GUI。

5b 在 `ocfs2console` 中，選取「*業集*」>「*業集節點*」。

若 `cluster.conf` 不存在，則主控台會以預設業集名稱 `ocfs2` 建立一個。依喜好修改叢集名稱。

5c 在「節點組態」對話方塊中，按一下「*新增*」以開啟「新增節點」對話方塊。

5d 在「新增節點」對話方塊中，指定您主要節點的唯一名稱、唯一 IP 位址 (如 *`192.168.1.1`*)，以及連接埠號碼 (選用，預設為 7777)，再按一下「*確定*」。

`ocfs2console` 主控台會依序指派 0 到 254 的節點插槽號碼。

5e 在「節點組態」對話方塊中，按一下「*套用*」，再按一下「*關閉*」離開「新增節點」對話方塊。

5f 按一下「*業集*」>「*傳播組態*」，將 `cluster.conf` 檔案儲存至所有節點。

6 若您需要重新啟動 OCFS2 叢集讓變更生效，請輸入下列行，等待程序傳回「*確定*」狀態。

```
/etc/init.d/o2cb stop
/etc/init.d/o2cb start
```

14.6.3 建立 OCFS2 磁碟區

建立 OCFS2 檔案系統並將新節點增加至叢集，只能在叢集中的一節點執行。

1. 開啟終端機視窗，並以 `root` 使用者身份登入。
2. 若 O2CB 業集服務離線，請輸入下列指令啟動服務，並等待程序傳回「*確定*」狀態。

   ```
   /etc/init.d/o2cb online ocfs2
   ```

   使用 OCFS2 業集上的實際業集名稱取代 *`ocfs2`*。

   OCFS2 叢集必須為上線狀態，因為格式化作業首先必須確定磁碟區未裝載於叢集中的任一節點。
3. 使用下列方法之一建立並格式化磁碟區：
   - 在 EVMSGUI 中，前往「*磁碟區*」頁面，選取「*製作檔案系統*」>「*OCFS2*」，然後指定組態設定。
   - 使用 `mkfs.ocfs2` 公用程式。如需此指令的語法資訊，請參閱 `mkfs.ocfs2` 線上文件。
   - 在 `ocfs2console` 中，按一下「*任務*」>「*格式化*」，在「可用設備」清單中選取您希望用於 OCFS2 磁碟區的設備，並為磁碟區指定組態，然後按一下「*確定*」格式化磁碟區。

   請參閱下列表格以得知建議設定。

OCFS2 參數	描述與建議
磁碟區標籤	磁碟區的描述性名稱可讓其裝載於不同節點時易於辨識。

OCFS2 參數	描述與建議
	使用 `tunefs.ocfs2` 公用程式依需要修改標籤。
叢集大小	叢集大小是配置給持有資料的檔案之空間最小單位。 選項有 4、8、16、32、64、128、256、512 和 1024 KB。格式化磁碟區之後就無法修改叢集大小。 Oracle 建議資料庫磁碟區使用 128 KB 或更大的業集。Oracle 亦建議 Oracle Home 使用 32 或 64 KB 的叢集大小。
節點插槽名稱	可同時裝載磁碟區的最大節點數目。裝載時，OCFS2 會為各節點建立單獨的系統檔案，如日誌。存取磁碟區的節點可以是小 endian 架構 (如 x86、x86-64 和 ia64) 和大 endian 架構 (如 ppc64 和 s390x) 的組合。 節點特定的檔案會被視為本機檔案。節點插槽號碼會附加至本機檔案。例如：journal:0000 隸屬於指派至插槽 0 的任一節點。 建立時請根據您希望同時裝載磁碟區的節點數量，設定節點插槽的最大數目。使用 `tunefs.ocfs2` 公用程式依需要增加節點插槽數目；該數值無法減少。
區塊大小	檔案系統可定址的空間最小單位。請在建立磁碟區時指定區塊大小。 選項有 512 位元 (不建議)、1 KB、2 KB 或 4 KB (建議大部分磁碟區使用)。格式化磁碟區之後就無法修改區塊大小。

14.7 裝載 OCFS2 磁碟區

1 開啟終端機視窗，並以 `root` 使用者身份登入。

2 若O2CB叢集服務離線，請輸入下列指令啟動服務，並等待程序傳回「*確定*」狀態。

```
/etc/init.d/o2cb online ocfs2
```

使用 OCFS2 業集上的實際業集名稱取代 *ocfs2*。

OCFS2 叢集必須為上線狀態，因為格式化作業必須確定磁碟區未裝載於叢集中的任一節點。

3 使用下列方法之一裝載磁碟區。

- 在ocfs2console中，選取「可用設備」清單中的設備並按一下「*裝載*」。選擇性地指定目錄裝載點和裝載選項，然後按一下「*確定*」。
- 從指令行裝載磁碟區，請使用 mount 指令。
- 系統開機時從 /etc/fstab 檔案裝載券冊。

裝載一個 OCFS2 磁碟區約需 5 秒，視心跳線串穩定下來所需花費的時間而定。成功裝載後，ocfs2console中的裝置清單會顯示裝載點與裝置。

提示：新增節點

當新節點嘗試連接叢集時，它們無法合併，因為節點未將這些節點新增至連線清單。若要解決此問題，請手動移至每個節點並發出以下指令以更新各自的連線清單：

```
o2cb_ctl -H -n ocfs2 -t cluster -a online=yes
```

如需使用這三種方法裝載 OCFS2 磁碟區的更多資訊，請參閱「Oracle 的 OCFS2 專案 [http://oss.oracle.com/projects/ocfs2/]」上的 *OCFS2 使用者指南 [http://oss.oracle.com/projects/ocfs2/documentation/]*。

執行 Oracle RAC 時，若 OCFS 磁碟區包含 Voting diskfile (CRS)、Cluster registry (OCR)、Data file、Redo log、Archive log 和 Control file 時，請務必使用 datavolume 和 nointr 裝載選項。裝載 Oracle Home 磁碟區時請勿使用這些選項。

選項	描述
datavolume	確認 Oracle 程序以 o_direct 旗標開啟檔案。
nointr	無岔斷。確認 IO 未被訊號岔斷。

14.8 其他資訊

如需使用 OCFS2 的更多資訊，請參閱「Oracle 的 OCFS2 專案 [http://oss.oracle.com/projects/ocfs2/]」上的*OCFS2 使用者指南 [http://oss.oracle.com/projects/ocfs2/documentation/]*。

15 Linux 存取控制清單

POSIX ACL (存取控制清單) 可以看做是檔案系統物件之傳統許可權概念的延伸。使用 ACL，定義許可權比傳統許可權概念更有彈性。

POSIX ACL 這個詞彙表示這是真正的 POSIX (*可攜式作業系統介面*) 標準。草稿標準 POSIX 1003.1e 和 POSIX 1003.2c 已各自因為某些原因而遭到撤銷。然而，UNIX 家族中許多系統的 ACL 都是以這些草稿為基礎，而且本章實作的檔案系統 ACL，也是依循這兩個標準。要查閱它們，請到 http://wt.xpilot.org/publications/posix.1e/。

15.1 傳統檔案許可權

有關傳統 Linux 檔案許可權的基本說明，請參閱 第 18.2 節「使用者和存取許可權」[330頁]。還有 setuid、setgid 以及黏貼位元的更多進階功能。

15.1.1 setuid 位元

在某些狀況下，存取許可權的限制可能太多。因此，Linux 擁有其他的設定，可讓目前使用者與群組的身份針對特定的動作而進行暫時的變更。例如，`passwd` 程式通常需要根許可權來存取 `/etc/passwd`。此檔案包含一些重要的資訊，例如，使用者的主目錄，以及使用者與群組的 ID。因此，一般使用者將無法變更 `passwd`，因為授權讓所有使用者直接存取此檔案是非常危險的。此問題的可行解決方案是 *setuid* 機制。setuid (設定使用者 ID) 是一種特殊檔案屬性，可以指示系統根據特定使用者 ID 的標示來執行程式。請考慮使用 `passwd` 指令：

```
-rwsr-xr-x  1 root shadow 80036 2004-10-02 11:08 /usr/bin/passwd
```

您會看到 s，表示已針對使用者許可權設定 setuid 位元。藉由 setuid 位元，所有啟動 passwd 指令的使用者，皆能以 root 身份來執行。

15.1.2 setgid 位元

setuid 位元適用於使用者。不過，群組也有對等的內容：*setgid*。設定此位元的程式會以儲存時指定的群組ID來執行，無論啟動該程式的使用者為何。因此，在附帶 setgid 位元的目錄中，所有新建的檔案和子目錄會指定至目錄所屬的群組。請思考以下範例目錄：

```
drwxrws--- 2 tux archive 48 Nov 19 17:12  backup
```

您會看到 s，表示已針對群組許可權設定 setgid 位元。目錄的擁有者與 archive 群組的成員可存取此目錄。非此群組成員的使用者會「對應」到個別群組。所有已寫入檔案的有效群組 ID 將會是 archive。例如，以 archive 群組 ID 執行的備份程式，即使沒有 root 權限，仍然能夠存取此目錄。

15.1.3 黏貼位元

另外還有*黏貼位元*。視它屬於可執行程式或目錄而定，結果是有差別的。如果屬於程式，用此方法標示的檔案會載入RAM，每次使用時就不需從硬碟取得該程式。這個屬性很少用，因為現在的硬碟速度都夠快了。如果此位元指定至目錄，就會禁止使用者刪除其他人的檔案。典型範例包括 /tmp 和 /var/tmp 目錄：

```
drwxrwxrwt 2 root root 1160 2002-11-19 17:15 /tmp
```

15.2 ACL 的優點

傳統上，會為 Linux 系統的每一個檔案物件，定義三組的許可權。這三組包括讀取 (r)、寫入 (w) 以及執行 (x) 許可權，用於三種使用者類型 (檔案擁有者、群組和其他使用者) 的每一個。此外，也可以設定*設定使用者 ID*、*設定群組 ID* 以及*黏貼*位元。這個小概念就足以應付大部分的實際情況。不過，對於更複雜的

案例或進階應用程式，系統管理員以前必須使用一些訣竅才能克服傳統許可權概念的限制。

ACL 可以看做是傳統檔案許可權概念的延伸，可指定許可權給個別使用者或群組，即使這些許可權並不對應至原始擁有者或所屬群組。存取控制清單是Linux核心的特性之一，目前ReiserFS、Ext2、Ext3、JFS和XFS都能支援。使用ACL可以理解複雜的情況，無需在應用程式層級上實行複雜的許可權模型。

在以Linux伺服器取代Windows伺服器時，您可以明顯發現ACL的優點。部分連接的工作站還可以在移轉之後，繼續在Winodws下執行。Linux系統可提供檔案和列印服務給含Samba的Winodws用戶端。若Samba可支援存取控制清單，使用者許可權就可以同時在Linux伺服器和使用圖形使用者介面的Windows(限Winodws NT和更新的版本)上設定。透過`winbindd`，部分的samba suite還可以指定許可權給只存取Windows領域中而且沒有Linux伺服器任何帳戶的使用者。

15.3 定義

使用者類別
: 傳統的POSIX許可權概念在指定檔案系統的許可權會使用三種*類別*的使用者：擁有者、所屬群組和其他使用者。三種許可權位元可以設定給每一個使用者類別，提供許可權來讀取(`r`)、寫入(`w`)以及執行(`x`)。

存取ACL
: 所有檔案系統物件(檔案和目錄)的使用者和群組存取許可權，都透過存取ACL的方法來決定。

預設ACL
: 預設的ACL只能套用至目錄。它們在檔案系統物件建立時，會決定從其上層目錄繼承的許可權。

ACL項目
: 每一個ACL都包含一組的ACL項目。ACL項目包含了類型、項目參照的使用者或群組修飾詞，以及一組許可權。對於部份項目類型，群組或使用者的修飾詞尚未定義。

15.4 處理 ACL

表格 15.1「ACL 項目類型」[278頁]概述了 6 種可能的 ACL 項目類型，每個類型皆針對使用者或使用者群組的許可權提出定義。*擁有者*項目定義擁有檔案或目錄之使用者的許可權。*所屬群組*項目定義檔案的所屬群組許可權。超級使用者可以使用 `chown` 或 `chgrp` 變更擁有者或所屬群組，在此情況下，擁有者和所屬群組項目將參照新的擁有者和所屬群組。每一個*命名的使用者*項目都可定義使用者許可權，而該許可權將於項目的修飾詞欄位中指定。每一個*命名的群組*項目都可定義群組許可權，而該許可權將於項目的修飾詞欄位中指定。只有命名的使用者和命名的群組的修飾詞欄位不是空的。*其他*項目定義所有其他使用者許可權。

*遮罩*項目可進一步限制由命名的使用者、命名的群組以及所屬群組項目所授予的許可權，方法是定義那些項目中的哪些許可權為有效以及哪些為遮罩。如果許可權存在於之前提及的任一項目或遮罩中，那麼就是有效的。只有包含於遮罩或實際項目的許可權是無效的—這表示並未授予許可權。定義在擁有者和所屬群組項目的所有許可權皆永遠有效。表格 15.2「遮罩存取許可權」[279頁]中的範例會示範此機制。

ACL 有兩種基本類別：*最小值* ACL 只包含擁有者、所屬群組和其他類別的項目，這對應於檔案和目錄的傳統許可權位元。*延伸* ACL 則不僅如此。它必須包含遮罩項目而且可以包含命名使用者和命名群組類型的多個項目。

表格 15.1 *ACL 項目類型*

類型	文字形式
擁有者	`user::rwx`
命名使用者	`user:name:rwx`
所屬群組	`group::rwx`
命名群組	`group:name:rwx`
遮罩	`mask::rwx`
其他	`other::rwx`

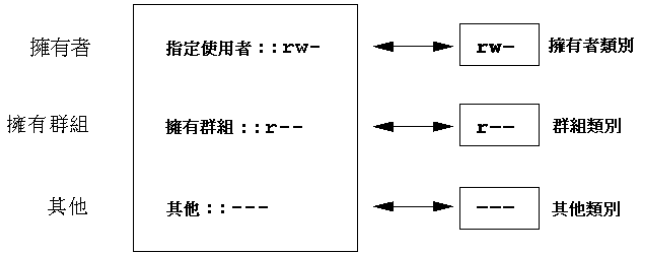
表格 15.2　遮罩存取許可權

項目類型	文字形式	許可權
命名使用者	`user:geeko:r-x`	`r-x`
遮罩	`mask::rw-`	`rw-`
	有效許可權：	`r--`

15.4.1 ACL 項目和檔案模式許可權位元

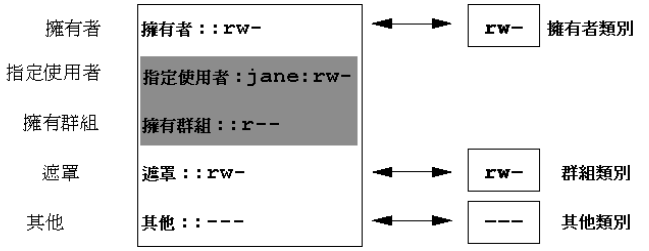
圖形 15.1「最小值 ACL：ACL 項目與許可權位元的比較」[279頁]和圖形 15.2「延伸 ACL：ACL 項目與許可權位元的比較」[280頁]將分別說明最小值 ACL 和延伸 ACL。圖的結構分成三個區塊—左方區塊顯示 ACL 項目的類型格式，中間區塊顯示範例 ACL，而右方區塊則顯示傳統許可權概念的許可權位元，例如以 `ls -l` 顯示。在這兩種狀況中，*擁有者類別*許可權都會對應至 ACL 項目擁有者。*其他類別*許可權會對應至個別的 ACL 項目。不過，*群組類別*許可權的對應與這兩個案例不同。

圖形 15.1　最小值 ACL：ACL 項目與許可權位元的比較



在最小值 ACL 中 (不含遮罩)，群組類別許可權會對應至 ACL 項目所屬群組，如圖形 15.1「最小值 ACL：ACL 項目與許可權位元的比較」[279頁]所示。在延伸 ACL 中 (含遮罩)，群組類別許可權是對應至遮罩項目。這會顯示於 圖形 15.2「延伸 ACL：ACL 項目與許可權位元的比較」[280頁]。

圖形 15.2　延伸 ACL：ACL 項目與許可權位元的比較



此對應方法可以確保應用程式互動順暢，無論其是否支援 ACL。透過許可權位元方法所指定的存取許可權，代表藉由 ACL 所做的所有其他「微調上限」。對許可權位元所做的變更會由 ACL 反映，反之亦然。

15.4.2 含存取 ACL 的目錄

在指令行上使用 `getfacl` 和 `setfacl` 這兩個指令，您就可以存取 ACL。使用這兩個指令的方式如以下範例所示範。

建立目錄之前，使用 `umask` 指令來定義每次建立檔案物件時，要遮罩的存取許可權。指令 `umask 027` 會指定完整範圍的許可權 (0) 給擁有者，拒絕群組寫入存取 (2) 以及不提供其他使用者任何許可權 (7)，以設定預設許可權。`umask` 實際上會遮罩相應的許可權位元或關閉它們。如需詳細資訊，請參閱 `umask` 線上文件。

`mkdir mydir` 會以 `umask` 設定的預設許可權建立 `mydir` 目錄。使用 `ls -dl mydir` 檢查所有許可權是否指定正確。範例的輸出如下：

```
drwxr-x--- ... tux project3 ... mydir
```

使用 `getfacl mydir` 檢查 ACL 的啟始狀態。會提供類似以下的資訊：

```
# file: mydir
# owner: tux
# group: project3
user::rwx
group::r-x
other::---
```

前三個輸出行顯示目錄的名稱、擁有者和所屬群組。接下來三行包含三個 ACL 項目擁有者、所屬群組和其他。實際上，在此最小 ACL 案例中，getfacl 指令不會產生任何您以 ls 指令無法取得的資訊。

以下列指令修改 ACL 以指定讀取、寫入和執行許可權給另一個使用者 geeko 以及另一個群組 mascots：

```
setfacl -m user:geeko:rwx,group:mascots:rwx mydir
```

選項 -m 提示 setfacl 修改現有的 ACL。以下引數指示要修改的 ACL 項目 (多個項目由逗點分開)。最後的部分指定這些修改要套用至什麼目錄名稱。使用 getfacl 指令，查看產生的 ACL。

```
# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other::---
```

除了為使用者 geeko 和群組 mascots 實施的項目之外，已經產生一個遮罩項目。會自動設定遮罩項目讓所有的許可權生效。除非使用 -n 停用此功能，否則 setfacl 會將現有遮罩項目調整為修改過的設定值。遮罩會定義群組類別中所有項目的最大有效存取許可權。這包括命名的使用者、命名的群組以及所屬群組。ls -dl mydir 顯示的類別群組許可權位元，現在會對應至遮罩項目。

```
drwxrwx---+ ... tux project3 ... mydir
```

輸出的第一欄包含額外的 +，指示此項目有一個*延伸* ACL。

根據 ls 指令的輸出，遮罩項目的許可權包括寫入存取。傳統上，此類的許可權位元表示所屬群組 (在此為 project3) 也擁有目錄 mydir 的寫入存取。然而，所屬群組的有效存取許可權會對應至針對所屬群組和遮罩而定義之許可權的重疊部分—在本範例中是 r-x (請參閱表格 15.2「遮罩存取許可權」 [279頁])。就範例中所屬群組的有效許可權而言，即使新增 ACL 項目之後也不會變更有效所有權。

使用 setfacl 或 chmod，編輯遮罩項目。例如，使用 chmod g-w mydir。接著 ls -dl mydir 顯示如下：

```
drwxr-x---+ ... tux project3 ... mydir
```

getfacl mydir 提供下列輸出：

```
# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx          # effective: r-x
group::r-x
group:mascots:rwx       # effective: r-x
mask::r-x
other::---
```

執行 chmod 指令，從群組類別位元移除寫入許可權之後，ls 指令的輸出就可以知道遮罩位元必須同時變更：寫入許可權將再次限於 mydir 的擁有者。getfacl 的輸出可確認此項結果。此輸出針對有效許可權未對應至原始許可權的所有項目包含一個註解，因為它們是根據遮罩項目過濾的。可以隨時使用 chmod g+w mydir 復原原始許可權。

15.4.3 含預設 ACL 的目錄

目錄可以有一個預設的 ACL，此為特別的 ACL，可在建立物件時定義目錄中物件繼承的存取許可權。預設 ACL 會影響子目錄和檔案。

預設 ACL 的作用

子目錄的預設 ACL，其許可權傳送至檔案以及子目錄的方法有兩種：

- 子目錄繼承上層目錄的預設 ACL，當成預設 ACL，也當成存取 ACL。
- 檔案會將預設 ACL 繼承為存取 ACL。

建立檔案系統的所有系統呼叫，會使用 mode 參數，它會為新建立的檔案系統物件，定義存取許可權。如果上層目錄沒有預設 ACL，umask 定義的許可權位元會從 mode 參數傳送的許可權中去除，將結果指定給新物件。如果上層目錄的預設 ACL 存在，指定給新物件的許可權位元會對應至 mode 參數的許可權重疊部份，以及定義在預設 ACL 的許可權。umask 在此案例中不予處理。

預設 ACL 的應用

以下三個範例顯示目錄和預設 ACL 的主要作業：

1. 新增預設 ACL 至現有的目錄 mydir，透過：

```
setfacl -d -m group:mascots:r-x mydir
```

setfacl 指令的 -d 選項，提示 setfacl 在預設 ACL 執行以下修改 (-m 選項)。

仔細查看指令的結果：

```
getfacl mydir

# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other::---
default:user::rwx
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other::---
```

getfacl 傳回存取 ACL 和預設 ACL 二者。預設 ACL 是由開頭為 default 的所有行形成的。雖然您只是在預設 ACL 的 mascots 群組的一個項目執行 setfacl 指令，不過 setfacl 會自動從存取 ACL 複製所有其他項目來建立有效的預設 ACL。預設 ACL 對於存取許可權沒有立即的作用。它們只會在建立檔案系統物件時才有作用。這些新物件只會從其上層目錄的預設 ACL 繼承許可權。

2. 在下一個範例中，使用 mkdir，在 mydir 目錄中建立子目錄，它會繼承預設 ACL。

```
mkdir mydir/mysubdir

getfacl mydir/mysubdir

# file: mydir/mysubdir
# owner: tux
# group: project3
user::rwx
group::r-x
group:mascots:r-x
mask::r-x
other::---
default:user::rwx
```

```
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other::---
```

如預期般，新建立的子目錄 mysubdir 擁有上層目錄預設 ACL 的許可權。mysubdir的存取 ACL 與mydir的預設 ACL 完全相同。目錄傳給其從屬物件的預設 ACL，也是如此。

3. 使用 touch 於 mydir 目錄建立檔案，例如，touch mydir/myfile。接著 ls -l mydir/myfile 會顯示如下：

```
-rw-r-----+ ... tux project3 ... mydir/myfile
```

getfacl mydir/myfile 的輸出如下：

```
# file: mydir/myfile
# owner: tux
# group: project3
user::rw-
group::r-x              # effective:r--
group:mascots:r-x       # effective:r--
mask::r--
other::---
```

建立新檔案時，touch 會使用值為 0666 的 模式。這表示檔案建立後，所有使用者類別都有讀取和寫入的許可權，而且在 umask 或預設 ACL 中沒有其他限制 (請參閱章節「預設 ACL 的作用」[282頁])。實際上，這表示所有未包含在 mode 值中的存取許可權，會從各自的 ACL 項目移除。雖然未從群組類別的 ACL 項目移除許可權，將遮罩項目修改為遮罩許可權，不在此模式中設定。

此方法可以確定應用程式 (例如編譯器) 與 ACL 的互動平順。您可以建立存取許可權受限的檔案，然後標示成可執行。mask 機制可以保證正確的使用者和群組可以執行它們。

15.4.4 ACL 檢查演算法

任何程序或應用程式授予存取 ACL 保護的檔案系統物件前，會先套用檢查演算法。做為基本的規則，ACL 項目會依下列順序接受檢查：擁有者、命名的使用者、所屬群組或命名的群組，以及其他。處理的存取合乎最符合程序的項目。許可權不會累積。

如果程序屬於一個以上的群組而且可能符合多個群組項目，就會變得更複雜。項目是從符合必要許可權的合適項目中隨機選取的。它與觸發最後的結果「access granted」的項目無關。同樣地，如果適合的群組項目不包含需要的許可權，隨機選取的項目會觸發最後的結果「拒絕存取」。

15.5 應用程式的 ACL 支援

ACL 可以用來實行很複雜的許可權情況，符合現在的應用程式需求。傳統許可權概念和 ACL 可以使用聰明的方法結合。基本檔案指令 (`cp`、`mv`、`ls` 等等) 都支援 ACL，Samba 和 Konqueror 也支援 ACL。

可惜的是，許多編輯器和檔案管理員都不支援 ACL。例如，使用 Emacs 複製檔案時，這些檔案的 ACL 會遺失。使用編輯器修改檔案時，檔案的 ACL 有時候會保留，有時候則不保留，由編輯器的備份模式決定。如果編輯器將變更寫入原始檔案，會保留存取 ACL。如果編輯器將更新的內容儲存至新檔案，後來重新命名成舊檔案名稱，則除非編輯器支援 ACL，否則 ACL 可能會遺失。除了 star 歸檔設備之外，目前沒有備份應用程式可以保存 ACL。

15.6 如需更多資訊

如需關於 ACL 的詳細資訊，請參閱 http://acl.bestbits.at/。請參閱 `getfacl(1)`、`acl(5)` 和 `setfacl(1)` 的 man 頁面。

16 RPM — 套件管理員

RPM (RPM 套件管理員) 用於管理軟體套件。主要指令為 `rpm` 及 `rpmbuild`。使用者、系統管理員和套件建立者可在威力強大的RPM資料庫中查詢已安裝軟體的詳細資訊。

`rpm`主要包括五種模式：安裝、解除安裝或更新軟體套件；重建RPM資料庫；查詢 RPM 基礎或個別的 RPM 歸檔；套件完整性檢查；簽署套件。`rpmbuild` 可用於建立初始來源的可安裝套件。

可安裝的RPM歸檔以特殊二進位格式包裝封裝。這些歸檔由要安裝的程式檔和`rpm`在安裝期間用來設定軟體套件或儲存在RPM資料庫中供記錄之用的特定中繼資訊所組成。RPM 歸檔的副檔名通常為 `.rpm`。

提示：軟體開發套件

對於許多套件，軟體開發所需的元件 (程式庫、標頭、Include 檔案等) 分別封裝在單獨的套件中。只有在您想要自行編譯軟體 (例如最新的 GNOME 套件) 時，才需要這些開發套件。由副檔名 `-devel` 即可識別出開發套件，例如 `alsa-devel`、`gimp-devel` 和 `kdelibs3-devel` 套件。

16.1 確認套件驗證性

RPM 套件具有 GnuPG 簽章。包含指紋的金鑰為：

```
1024D/9C800ACA 2000-10-19 SuSE Package Signing Key <build@suse.de>
Key fingerprint = 79C1 79B2 E1C8 20C1 890F  9994 A84E DAE8 9C80 0ACA
```

`rpm --checksig package-1.2.3.rpm` 指令可用來驗證 RPM 套件的簽章，以判斷它確實來自 SUSE，還是來自其他可信任的設備。特別建議在從網際網路更新套件時使用此指令。SUSE 公用套件簽章金鑰通常位於 `/root/.gnupg/` 中。在 `/usr/lib/rpm/gnupg/` 目錄中也額外放置了金鑰，以便讓一般使用者確認 RPM 套件的簽名。

16.2 管理套件：安裝、更新和解除安裝

一般而言，安裝 RPM 歸檔很簡單：`rpm -i package.rpm`。使用此指令可安裝套件，但是必須滿足套件的相依性，而且套件不可與其他套件衝突。如果 `rpm` 要求要安裝的套件必須符合相依性要求，會顯示錯誤訊息。RPM 資料庫會在背景中確認沒有產生衝突 — 特定的檔案僅可屬於一個套件。藉由選擇不同選項，您可以強迫 `rpm` 忽略這些預設，但只有進階使用者才可以這樣做。否則，會有危害系統完整性的風險，而且可能會危害更新系統的能力。

選項 `-U` 或 `--upgrade` 和 `-F` 或 `--freshen` 可用來更新套件，例如，`rpm -F package.rpm`。此指令會移除舊版的檔案，並立刻安裝新檔案。兩個版本之間的不同在於 `-U` 會安裝先前系統中沒有的套件，而 `-F` 僅更新先前安裝的套件。在更新時，`rpm` 會使用下列策略小心地更新組態檔：

- 如果系統管理員未變更組態檔，`rpm` 會安裝新版本的相應檔案。系統管理員不需要做任何動作。

- 如果系統管理員在更新之前變更了組態檔，`rpm` 會將變更的檔案以副檔名 `.rpmorig` 或 `.rpmsave` (備份檔案) 儲存，並安裝新套件的版本，但此操作僅在原始安裝的檔案與新版本不同時才會發生。在這種情況下，請比較備份檔案 (`.rpmorig` 或 `.rpmsave`) 與新安裝的檔案，然後再對新檔案做一次變更。之後，請確定刪除所有 `.rpmorig` 和 `.rpmsave` 檔案以避免未來更新的問題。

- 如果組態檔已存在，*且*如果在 `.spec` 檔案中指定了 `noreplace` 標籤，便會出現 `.rpmnew` 檔案。

在更新之後，應該在比較完 `.rpmsave` 和 `.rpmnew` 之後將它們移除，才不會妨礙未來的更新。如果 RPM 資料庫之前無法辨識檔案，會指定 `.rpmorig` 副檔名。

否則，會使用 .rpmsave。換言之，.rpmorig 是在將外來格式更新為 RPM 後產生的。.rpmsave 是在將舊版 RPM 更新為新版 RPM 後產生的。.rpmnew 不會透露任何關於系統管理員是否曾對組態檔做過任何變更的資訊。可在 /var/adm/rpmconfigcheck 找到這些檔案的清單。部分組態檔 (如 /etc/httpd/httpd.conf) 不會覆寫以允許後續操作。

-U 切換參數的功能不不完全等同於使用 -e 選項進行解除安裝以及使用 -i 選項進行安裝。如果可能，請使用 -U。

若要移除套件，請輸入 rpm -e *package*。如果沒有無法解析的相依性，rpm 僅會刪除套件。只要其他應用程式還需要它，理論上無法刪除 Tcl/Tk，。即使是這種情況下，RPM 還是可從資料庫呼叫以得到協助。如果此種刪除是 (不論出於何種原因或處於不尋常的狀況下) 不可行的 — 即使不存在額外的相依性，使用選項 --rebuilddb 來重建 RPM 資料庫可能有幫助。

16.3 RPM 與修補程式

為了確保系統的操作安全性，必須經常在系統上安裝更新套件。以前，要除去套件中的錯誤，只能夠更換整個套件。在大型套件中，如果有包含錯誤的小檔案就很容易產生大量的資料。但是，SUSE RPM 提供在套件中安裝修補程式的功能。

最重要的考量可用 pine 當作範例：

修補程式 RPM 是否適用於我的系統?

若要進行檢查，首先請查詢安裝的套件版本。以 pine 為例，可使用指令

```
rpm -q pine
pine-4.44-188
```

然後檢查修補程式 RPM 是否適用於此版本的 pine：

```
rpm -qp --basedon pine-4.44-224.i586.patch.rpm
pine = 4.44-188
pine = 4.44-195
pine = 4.44-207
```

此修補程式適用於三種不同版本的 Pine。範例中也列出安裝的版本，因此可安裝此修補程式。

修補程式會更換哪些檔案?

受到修補程式影響的檔案可在修補程式 RPM 中輕易地看出。rpm 參數 -P 可讓您選擇特殊的修補程式功能。可使用以下指令顯示檔案清單：

```
rpm -qpPl pine-4.44-224.i586.patch.rpm
/etc/pine.conf
/etc/pine.conf.fixed
/usr/bin/pine
```

或者，如果已經安裝修補程式，可使用以下指令：

```
rpm -qPl pine
/etc/pine.conf
/etc/pine.conf.fixed
/usr/bin/pine
```

修補程式 RPM 如何安裝於系統中?

修補程式 RPM 可做為一般 RPM 使用。唯一的不同是必須已經安裝適合的 RPM。

系統中已經安裝哪個修補程式，是用於哪個套件版本?

使用指令 rpm -qPa 可顯示系統中已經安裝的所有修補程式清單。如果新系統中僅安裝一個修補程式 (如本範例)，則清單顯示如下：

```
rpm -qPa
pine-4.44-224
```

如果在日後，您想要知道原始安裝的套件版本，可在 RPM 資料庫中找到此資訊。以 pine 為例，可使用以下指令顯示此資訊：

```
rpm -q --basedon pine
pine = 4.44-188
```

可在 rpm 以及 rpmbuild 的 man 頁面中找到包括 RPM 修補程式功能的相關資訊。

16.4 Delta RPM 套件

Delta RPM 套件包含舊版與新版 RPM 套件之間的差異。在舊版 RPM 上套用 delta RPM 會產生完整的新版 RPM。但是您不需要取得舊版的 RPM，因為 delta RPM 也可以和已安裝的 RPM 搭配使用。delta RPM 套件的大小比修補程式 RPM 還小，這一特點有利於透過網際網路傳送更新套件。缺點是使用 delta RPM 的更新作業會比一般或修補程式 RPM 消耗更多的 CPU 週期。

prepdeltarpm、writedeltarpm 以及 applydeltarpm 二進位檔屬於 delta RPM 套裝軟體 (deltarpm 套件) 的一部分，可協助您建立和套用 delta RPM 套件。使用下列指令可建立名稱為 new.delta.rpm 的 delta RPM。下列指令假設 old.rpm 和 new.rpm 都已存在：

```
prepdeltarpm -s seq -i info old.rpm > old.cpio
prepdeltarpm -f new.rpm > new.cpio
xdelta delta -0 old.cpio new.cpio delta
writedeltarpm new.rpm delta info new.delta.rpm
```

最後，移除暫存工作檔案 old.cpio、new.cpio 以及 delta。

如果已經安裝舊套件，使用 applydeltarpm 即可從檔案系統重新建構新 RPM：

```
applydeltarpm new.delta.rpm new.rpm
```

若不要存取檔案系統，而要從舊 RPM 產生新 RPM，請使用 -r 選項：

```
applydeltarpm -r old.rpm new.delta.rpm new.rpm
```

請參閱 /usr/share/doc/packages/deltarpm/README" 以取得技術細節。

16.5 RPM 查詢

使用 -q 選項，rpm 會啟動查詢，可檢查 RPM 歸檔 (藉由新增選項 -p)，也可以查詢安裝套件的 RPM 資料庫。有多個切換參數可用於指定所需的資訊類型。請參閱表格 16.1「最重要的 RPM 查詢選項」[291頁]。

表格 16.1 *最重要的 RPM 查詢選項*

-i	套件資訊
-l	檔案清單
-f FILE	查詢包含 *FILE* 檔案的套件 (完整的路徑必須以 *FILE* 指定)
-s	含有狀態資訊的檔案清單 (隱含 -l)

`-d`	只列出文件檔案 (隱含 `-l`)
`-c`	只列出組態檔案 (隱含 `-l`)
`--dump`	含有完整詳細資訊的檔案清單 (與 `-l`、`-c` 或 `-d` 一起搭配使用)
`--provides`	列出另一個套件可以使用 `--requires` 要求的套件功能
`--requires`、`-R`	套件所需的功能
`--scripts`	安裝程序檔 (預先安裝、後續安裝、解除安裝)

例如，`rpm -q -i wget` 指令可顯示如 範例 16.1「rpm -q -i wget」[292頁] 中所示的資訊。

範例 16.1 *rpm -q -i wget*

```
Name        : wget                         Relocations: (not relocatable)
Version     : 1.9.1                             Vendor: SUSE LINUX AG,
Nuernberg, Germany
Release     : 50                         Build Date: Sat 02 Oct 2004
03:49:13 AM CEST
Install date: Mon 11 Oct 2004 10:24:56 AM CEST      Build Host: f53.suse.de
Group       : Productivity/Networking/Web/Utilities   Source RPM:
wget-1.9.1-50.src.rpm
Size        : 1637514                        License: GPL
Signature   : DSA/SHA1, Sat 02 Oct 2004 03:59:56 AM CEST, Key ID
a84edae89c800aca
Packager    : http://www.suse.de/feedback
URL         : http://wget.sunsite.dk/
Summary     : A tool for mirroring FTP and HTTP servers
Description :
Wget enables you to retrieve WWW documents or FTP files from a server.
This can be done in script files or via the command line.
[...]
```

只有在您指定完整檔案名稱及完整路徑時，選項 `-f` 才會有作用。提供盡可能多的檔案名稱。例如，以下指令

```
rpm -q -f /bin/rpm /usr/bin/wget
```

會產生：

```
rpm-4.1.1-191
wget-1.9.1-50
```

如果只知道檔案名稱的一部分，可使用範例16.2「搜尋套件的程序檔」[293頁]中所示的外圍程序程序檔。執行時，可將部份檔案名稱當作參數傳給程序檔。

範例 16.2 *搜尋套件的程序檔*

```
#! /bin/sh
for i in $(rpm -q -a -l | grep  $1); do
    echo "\"$i\" is in package:"
    rpm -q -f $i
    echo ""
done
```

`rpm -q --changelog rpm` 指令會依日期排序，顯示特定套件之相關變更資訊的詳細清單。此範例顯示套件 `rpm` 的相關資訊。

藉由安裝的 RPM 資料庫協助，可執行驗證檢查。以 `-V`、`-y` 或 `--verify` 啟動檢查。使用此選項，`rpm`可顯示從安裝開始，套件中所有變更過的檔案。`rpm` 使用八個字元的符號來提供下列變更的提示：

表格 16.2 *RPM 驗證選項*

`5`	MD5 檢查總數
`S`	檔案大小
`L`	符號連結
`T`	修改時間
`D`	主要和次要的設備編號
`U`	擁有者
`G`	群組
`M`	模式 (許可權和檔案類型)

如果是組態檔，會印出字母 `c`。例如，若 `/etc/wgetrc` (`wget`) 有變更：

```
rpm -V wget
S.5....T c /etc/wgetrc
```

RPM 資料庫的檔案放在 /var/lib/rpm。如果分割區 /usr 的大小為 1 GB，此資料庫將佔用 30 MB 左右的空間，尤其是在完整更新之後。如果資料庫遠大於預期，使用選項 --rebuilddb 來重建資料庫很有用。在執行之前，請備份舊的資料庫。cron 程序檔 cron.daily 會對資料庫做每日備份 (以 gzip 封裝)，並將備份儲存在 /var/adm/backup/rpmdb 中。副本數量由 /etc/sysconfig/backup 中的變數 MAX_RPMDB_BACKUPS (預設值：5) 控制。單一備份的大小大約是 1 GB 的 /usr 備份成 1 MB。

16.6 安裝與編譯來源套件

所有來源套件均帶有副檔名 .src.rpm (來源 RPM)。

提示

來源套件可從安裝媒體複製到硬碟，並用 YaST 解壓縮。但是，在套件管理員中，它們不會被標示為已安裝 ([i])。這是因為來源套件沒有輸入 RPM 資料庫中。只有*已安裝的*作業系統軟體會列在 RPM 資料庫中。您在「安裝」來源套件時，僅會將原始程式碼新增到系統中。

在 /usr/src/packages 中必須可以找到 rpm 和 rpmbuild 的下列目錄 (除非您在如 /etc/rpmrc 的檔案中指定自定設定)：

SOURCES
: 用於原始來源 (.tar.bz2 或 .tar.gz 檔案等) 和配送特定調整 (大部份是 .diff 或 .patch 檔案)

SPECS
: 用於 *.spec* 檔案，和中繼 Makefile 相似，可控制 *build* 程序

BUILD
: 所有來源在此目錄中解壓縮、修補和編譯

RPMS
: 儲存完整二進位套件的地方

SRPMS
此處為來源 RPM

當您使用 YaST 安裝來源套件時，所有需要的元件都會安裝在 /usr/src/packages 中：SOURCES 中的來源和調整以及 SPECS 中的相關 .spec。

警告

請勿試驗系統元件 (glibc、rpm、sysvinit 等)，因為這樣會危害系統的操作性。

以下範例使用 wget.src.rpm 套件。在使用 YaST 安裝套件之後，應該有類似下列清單的檔案：

```
/usr/src/packages/SOURCES/nops_doc.diff
/usr/src/packages/SOURCES/toplev_destdir.diff
/usr/src/packages/SOURCES/wget-1.9.1+ipvmisc.patch
/usr/src/packages/SOURCES/wget-1.9.1-brokentime.patch
/usr/src/packages/SOURCES/wget-1.9.1-passive_ftp.diff
/usr/src/packages/SOURCES/wget-LFS-20040909.tar.bz2
/usr/src/packages/SOURCES/wget-wrong_charset.patch
/usr/src/packages/SPECS/wget.spec
```

rpmbuild -b *X* /usr/src/packages/SPECS/wget.spec 可開始編譯。*X* 代表建立程序各種階段的萬用字元 (請參閱 --help 的輸出或 RPM 文件以取得詳細資訊)。以下僅為簡略的說明：

-bp
在 /usr/src/packages/BUILD 中準備來源：解壓縮和修補。

-bc
執行與 -bp 相同動作，但是會額外編譯。

-bi
執行與 -bp 相同的動作，但是會額外安裝建立的軟體。警告：如果套件不支援 BuildRoot 功能，您可能會覆寫組態檔。

-bb
執行與 -bi 相同的動作，但是會額外建立二進位套件。如果編譯成功，二進位應該在 /usr/src/packages/RPMS。

`-ba`
執行與 `-bb` 相同的動作，但是會額外建立來源 RPM。如果編譯成功，二進位應該在 `/usr/src/packages/SRPMS`。

`--short-circuit`
略過部分步驟。

現在可使用 `rpm -i` (最好使用 `rpm -U`) 來安裝所建立的二進位 RPM。使用 `rpm` 來安裝會讓它出現在 RPM 資料庫中。

16.7 以 build 編譯 RPM 套件

許多套件中都包含不想要的檔案，它們會在build程序中增到執行系統中，因為導致危險產生。為了避免發生此狀況，可使用 `build`，它會建立要在其中建立套件的已定義環境。若要建立此 chroot 環境，必須提供 `build` 程序檔與完整的套件樹狀結構。此樹狀結構可在硬碟上、透過 NFS 或從 DVD 取得。用 `build --rpms` *`directory`* 設定位置。和 `rpm` 不同，`build` 指令會在來源目錄中尋找 SPEC 檔。若要以裝載在系統中 `/media/dvd` 之下的 DVD 建立 wget (如上面的範例)，請以 `root` 的身分執行下列指令：

```
cd /usr/src/packages/SOURCES/
mv ../SPECS/wget.spec .
build --rpms /media/dvd/suse/ wget.spec
```

之後，系統便會在 `/var/tmp/build-root` 中建立一個最小的環境。套件將於此環境中建立。完成時，結果套件位於 `/var/tmp/build-root/usr/src/packages/RPMS` 中。

`build`程序檔可提供數個額外選項。例如，讓程序檔偏好您自己的RPM、省略建置環境的啟始化，或將`rpm`指令限制在上述階段之一。可使用`build --help`以及參閱 `build` man 頁面來存取其他資訊。

16.8 RPM 歸檔和 RPM 資料庫工具

Midnight Commander (`mc`) 可顯示 RPM 歸檔的內容，並複製部分內容。它將歸檔以虛擬檔案系統呈現，提供 Midnight Commander 的所有常見功能表選項。使

用 F3 可顯示 HEADER。使用游標和 Enter 可檢視歸檔結構。使用 F5 可複製歸檔元件。

KDE 提供 kpackage 工具做為 rpm 的前端。完整功能的套件管理員是以 YaST 模組的方式提供 (請參閱 第 8.3.1 節「安裝和移除軟體」 [119頁])。

17 系統監視公用程式

有許多程式和機制可用來檢查您的系統狀態，其中一些會在這裡介紹。另外還會介紹一些例行工作使用的公用程式，及其重要參數。

所有介紹的指令都會有相關的輸出範例。在這些範例中，第一行是指令本身(在 > 或 # 符號提示之後)。方括號 `[...]` 可用來表示省略，而且較長指令行會換行。較長指令行的斷行會用反斜線 (\) 表示。

```
# command -x -y
output line 1
output line 2
output line 3 is annoyingly long, so long that \
    we have to break it
output line 3
[...]
output line 98
output line 99
```

為了介紹更多公用程式，說明的部份保持簡短。如需所有指令的相關資訊，請參閱 man 頁面。大部份的指令都能解讀參數 `--help`，此參數可產生一份可能參數的簡短清單。

17.1 除錯

17.1.1 指定需要的程式庫：ldd

使用指令 `ldd` 來釐清哪些程式庫將根據指定引數載入動態可執行檔。

```
tux@mercury:~> ldd /bin/ls
        linux-gate.so.1 =>  (0xffffe000)
        librt.so.1 => /lib/librt.so.1 (0xb7f97000)
        libacl.so.1 => /lib/libacl.so.1 (0xb7f91000)
        libc.so.6 => /lib/libc.so.6 (0xb7e79000)
        libpthread.so.0 => /lib/libpthread.so.0 (0xb7e67000)
        /lib/ld-linux.so.2 (0xb7fb6000)
        libattr.so.1 => /lib/libattr.so.1 (0xb7e63000)
```

靜態二進位檔不需任何動態程式庫。

```
tux@mercury:~> ldd /bin/sash
        not a dynamic executable
tux@mercury:~> file /bin/sash
/bin/sash: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for
GNU/Linux 2.6.4, statically linked, for GNU/Linux 2.6.4, stripped
```

17.1.2 一個程式執行階段的程式庫呼叫：ltrace

指令 `ltrace` 讓您可以追蹤處理程序的程式庫呼叫。這個指令的用法與 `strace` 相似。參數 `-c` 會輸出程式庫呼叫的數目和持續時間。

```
tux@mercury:~> ltrace -c find ~
% time     seconds  usecs/call     calls      function
------ ----------- ----------- --------- --------------------
 34.37    6.758937         245     27554 __errno_location
 33.53    6.593562         788      8358 __fprintf_chk
 12.67    2.490392         144     17212 strlen
 11.97    2.353302         239      9845 readdir64
  2.37    0.466754          27     16716 __ctype_get_mb_cur_max
  1.17    0.230765          27      8358 memcpy
[...]
  0.00    0.000036          36         1 textdomain
------ ----------- ----------- --------- --------------------
100.00   19.662715                105717 total
```

17.1.3 執行程式的系統呼叫：strace

公用程式 `strace` 可用來追蹤執行中處理程序的所有系統呼叫。將 `strace` 新增至指令行的開頭，以正常方式來輸入指令：

```
tux@mercury:~> strace ls
execve("/bin/ls", ["ls"], [/* 61 vars */]) = 0
uname({sys="Linux", node="mercury", ...}) = 0
brk(0)                                  = 0x805c000
```

```
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or \
    directory)
open("/etc/ld.so.cache", O_RDONLY)      = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=89696, ...}) = 0
mmap2(NULL, 89696, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb7ef2000
close(3)                                = 0
open("/lib/librt.so.1", O_RDONLY)       = 3
read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\0\0\3\0\3\0\1\0\0\0000\36\0"..., 512) \
   = 512
fstat64(3, {st_mode=S_IFREG|0755, st_size=36659, ...}) = 0
[...]
stat64(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 0), ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) \
    = 0xb7ca7000
write(1, "bin  Desktop  Documents  music\tM"..., 55bin  Desktop  Documents \
   \  music       Music  public_html  tmp
) = 55
close(1)                                = 0
munmap(0xb7ca7000, 4096)                = 0
exit_group(0)                           = ?
```

例如，要追蹤所有試圖開啟某一特定檔案的動作，可使用下列指令：

```
tux@mercury:~> strace -e open ls .bashrc
open("/etc/ld.so.cache", O_RDONLY)      = 3
open("/lib/librt.so.1", O_RDONLY)       = 3
open("/lib/libacl.so.1", O_RDONLY)      = 3
open("/lib/libc.so.6", O_RDONLY)        = 3
open("/lib/libpthread.so.0", O_RDONLY)  = 3
open("/lib/libattr.so.1", O_RDONLY)     = 3
[...]
```

如果要追蹤所有子程序，可使用參數 `-f`。這樣便可充分掌控 strace 的行為和輸出格式。如需更多資訊，請參閱 `man strace`。

17.2 檔案和檔案系統

17.2.1 確定檔案類型：file

指令 `file` 可查看 `/etc/magic` 來確定檔案或檔案清單的類型。

```
tux@mercury:~> file /usr/bin/file
/usr/bin/file: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), \
    for GNU/Linux 2.2.5, dynamically linked (uses shared libs), stripped
```

參數 `-f list` 會使用檔案名稱清單來指定要檢驗的檔案。`-z` 可讓 `file` 查看壓縮檔案內部：

```
tux@mercury:~> file /usr/share/man/man1/file.1.gz
usr/share/man/man1/file.1.gz: gzip compressed data, from Unix, max compression
tux@mercury:~> file -z /usr/share/man/man1/file.1.gz
/usr/share/man/man1/file.1.gz: ASCII troff or preprocessor input text \
    (gzip compressed data, from Unix, max compression)
```

17.2.2 檔案系統與其使用率：mount、df 與 du

指令 `mount` 會顯示哪個檔案系統 (設備和類型) 已裝載於哪個裝載點：

```
tux@mercury:~> mount
/dev/sda3 on / type reiserfs (rw,acl,user_xattr)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
udev on /dev type tmpfs (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
/dev/sda1 on /boot type ext2 (rw,acl,user_xattr)
/dev/sda4 on /local type reiserfs (rw,acl,user_xattr)
/dev/fd0 on /media/floppy type subfs (rw,nosuid,nodev,noatime,fs=floppyfss,p
```

使用指令 `df` 來取得檔案系統總使用量的資訊。參數 `-h` (或 `--human-readable`) 可將輸出轉換為一般使用者容易瞭解的形式。

```
tux@mercury:~> df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda3              11G  3.2G  6.9G  32% /
udev                  252M  104K  252M   1% /dev
/dev/sda1              16M  6.6M  7.8M  46% /boot
/dev/sda4              27G   34M   27G   1% /local
```

使用指令 `du`，可以顯示指定目錄及其子目錄下所有檔案的大小。參數 `-s` 可以不顯示輸出的詳細資訊。`-h` 會再次將資料轉換成人類可判讀的格式：

```
tux@mercury:~> du -sh /local
1.7M    /local
```

17.2.3 其他有關 ELF 二進位檔的資訊

使用 `readelf` 公用程式來讀取二進位檔的內容。這個公用程式甚至可以搭配為其他硬體結構所建的 ELF 檔案使用。

```
tux@mercury:~> readelf --file-header /bin/ls
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           Intel 80386
  Version:                           0x1
  Entry point address:               0x8049b60
  Start of program headers:          52 (bytes into file)
  Start of section headers:          81112 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         9
  Size of section headers:           40 (bytes)
  Number of section headers:         30
  Section header string table index: 29
```

17.2.4 檔案內容：stat

指令 `stat` 會顯示檔案內容：

```
tux@mercury:~> stat /etc/profile
  File: `/etc/profile'
  Size: 8080            Blocks: 16         IO Block: 4096   regular file
Device: 806h/2054d      Inode: 64942       Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2007-07-16 23:28:18.000000000 +0200
Modify: 2006-09-19 14:45:01.000000000 +0200
Change: 2006-12-05 14:54:55.000000000 +0100
```

參數 `--filesystem` 會產生特定檔案的檔案系統詳細內容：

```
tux@mercury:~> stat /etc/profile --filesystem
  File: "/etc/profile"
    ID: 0        Namelen: 255     Type: reiserfs
Block size: 4096       Fundamental block size: 4096
Blocks: Total: 2622526    Free: 1809771    Available: 1809771
Inodes: Total: 0          Free: 0
```

17.3 硬體資訊

17.3.1 PCI 資源：lspci

指令 `lspci` 會列出 PCI 資源：

```
mercury:~ # lspci
00:00.0 Host bridge: Intel Corporation 82845G/GL[Brookdale-G]/GE/PE \
    DRAM Controller/Host-Hub Interface (rev 01)
00:01.0 PCI bridge: Intel Corporation 82845G/GL[Brookdale-G]/GE/PE \
    Host-to-AGP Bridge (rev 01)
00:1d.0 USB Controller: Intel Corporation 82801DB/DBL/DBM \
    (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #1 (rev 01)
00:1d.1 USB Controller: Intel Corporation 82801DB/DBL/DBM \
    (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #2 (rev 01)
00:1d.2 USB Controller: Intel Corporation 82801DB/DBL/DBM \
    (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #3 (rev 01)
00:1d.7 USB Controller: Intel Corporation 82801DB/DBM \
    (ICH4/ICH4-M) USB2 EHCI Controller (rev 01)
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev 81)
00:1f.0 ISA bridge: Intel Corporation 82801DB/DBL (ICH4/ICH4-L) \
    LPC Interface Bridge (rev 01)
00:1f.1 IDE interface: Intel Corporation 82801DB (ICH4) IDE \
    Controller (rev 01)
00:1f.3 SMBus: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) \
    SMBus Controller (rev 01)
00:1f.5 Multimedia audio controller: Intel Corporation 82801DB/DBL/DBM \
    (ICH4/ICH4-L/ICH4-M) AC'97 Audio Controller (rev 01)
01:00.0 VGA compatible controller: Matrox Graphics, Inc. G400/G450 (rev 85)
02:08.0 Ethernet controller: Intel Corporation 82801DB PRO/100 VE (LOM) \
    Ethernet Controller (rev 81)
```

使用 `-v` 可產生更詳細的清單：

```
mercury:~ # lspci
[...]
02:08.0 Ethernet controller: Intel Corporation 82801DB PRO/100 VE (LOM)\
    Ethernet Controller (rev 81)
        Subsystem: Fujitsu Siemens Computer GmbH: Unknown device 1001
        Flags: bus master, medium devsel, latency 66, IRQ 11
        Memory at d1000000 (32-bit, non-prefetchable) [size=4K]
        I/O ports at 3000 [size=64]
        Capabilities: [dc] Power Management version 2
```

關於設備名稱解析的資訊是從檔案 `/usr/share/pci.ids` 取得。本檔中未列出的 PCI ID 會標記為「未知的設備」。

參數 -vv 可產生所有可供程式查詢的資訊。若要檢視純數值，請使用參數 -n。

17.3.2 USB 設備：lsusb

指令 lsusb 會列出所有 USB 設備。加上選項 -v，會列出更多細節清單。詳細資訊讀取自目錄 /proc/bus/usb/。下列為 lsusb 在連接 USB 設備時的輸出：中樞器、記憶體晶片組、硬碟與滑鼠。

```
mercury:/ # lsusb
Bus 004 Device 007: ID 0ea0:2168 Ours Technology, Inc. Transcend JetFlash \
    2.0 / Astone USB Drive
Bus 004 Device 006: ID 04b4:6830 Cypress Semiconductor Corp. USB-2.0 IDE \
    Adapter
Bus 004 Device 005: ID 05e3:0605 Genesys Logic, Inc.
Bus 004 Device 001: ID 0000:0000
Bus 003 Device 001: ID 0000:0000
Bus 002 Device 001: ID 0000:0000
Bus 001 Device 005: ID 046d:c012 Logitech, Inc. Optical Mouse
Bus 001 Device 001: ID 0000:0000
```

17.3.3 SCSI 設備的相關資訊：scsiinfo

指令 scsiinfo 會列出 SCSI 設備的資訊。加上選項 -l，可列出所有系統已知的 SCSI 設備 (類似資訊經由指令 lsscsi 也可取得)。下列為 scsiinfo -i /dev/sda 的輸出，其中提供硬碟相關資訊。選項 -a 可提供更多資訊。

```
mercury:/ # scsiinfo -i /dev/sda
Inquiry command
---------------
Relative Address                  0
Wide bus 32                       0
Wide bus 16                       1
Synchronous neg.                  1
Linked Commands                   1
Command Queueing                  1
SftRe                             0
Device Type                       0
Peripheral Qualifier              0
Removable?                        0
Device Type Modifier              0
ISO Version                       0
ECMA Version                      0
ANSI Version                      3
AENC                              0
TrmIOP                            0
Response Data Format              2
```

```
Vendor:                 FUJITSU
Product:                MAS3367NP
Revision level:         0104A0K7P43002BE
```

選項 -d 會以兩個表格的形式列出硬碟不良區塊的缺失清單：首先列出廠商供應者 (製造商表格)，然後列出操作中出現的不良區塊 (生長表格)。如果成長表格的項目數量持續增加，此時最好更換硬碟。

17.4 網路

17.4.1 顯示網路狀態：netstat

netstat 顯示網路連線、路由表 (-r)、介面 (-i)、偽裝連接 (-M)、多重廣播成員 (-g) 以及統計資料 (-s)。

```
tux@mercury:~> netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags  MSS Window  irtt Iface
192.168.2.0     *               255.255.254.0   U        0 0          0 eth0
link-local      *               255.255.0.0     U        0 0          0 eth0
loopback        *               255.0.0.0       U        0 0          0 lo
default         192.168.2.254   0.0.0.0         UG       0 0          0 eth0
```

```
tux@mercury:~> netstat -i
Kernel Interface table
Iface   MTU Met   RX-OK RX-ERR RX-DRP RX-OVR  TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0   1500   0 1624507 129056      0      0   7055      0      0      0 BMNRU
lo    16436   0   23728      0      0      0  23728      0      0      0 LRU
```

顯示網路連線或數據時，您可指定要顯示的插槽類型：TCP (-t)、UDP (-u)netstat，或 raw (-r)。-p 選項顯示 PID 和每個插槽所屬的程式名稱。

下例列出所有 TCP 連接和使用這些連接的程式。

```
mercury:~ # netstat -t -p
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address  Foreign Address       State       PID/Pro

tcp      0      0 mercury:33513    www.novell.com:www-http ESTABLISHED 6862/fi
tcp      0    352 mercury:ssh      mercury2.:trc-netpoll     ESTABLISHED
19422/s
tcp      0      0 localhost:ssh  localhost:17828         ESTABLISHED -
```

以下會顯示 TCP 通訊協定的統計資料：

```
tux@mercury:~> netstat -s -t
Tcp:
    2427 active connections openings
    2374 passive connection openings
    0 failed connection attempts
    0 connection resets received
    1 connections established
    27476 segments received
    26786 segments send out
    54 segments retransmited
    0 bad segments received.
    6 resets sent
[...]
    TCPAbortOnLinger: 0
    TCPAbortFailed: 0
    TCPMemoryPressures: 0
```

17.5 /proc 檔案系統

/proc 檔案系統是一個虛擬檔案系統，核心會在此系統內以虛擬檔案的形式來保存重要資訊。例如，使用這個指令可顯示 CPU 類型。

```
tux@mercury:~> cat /proc/cpuinfo
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 6
model           : 8
model name      : AMD Athlon(tm) XP 2400+
stepping        : 1
cpu MHz         : 2009.343
cache size      : 256 KB
fdiv_bug        : no
[...]
```

使用下列指令來查詢位置和岔斷使用：

```
tux@mercury:~> cat /proc/interrupts
           CPU0
  0:    3577519          XT-PIC  timer
  1:        130          XT-PIC  i8042
  2:          0          XT-PIC  cascade
  5:     564535          XT-PIC  Intel 82801DB-ICH4
  7:          1          XT-PIC  parport0
  8:          2          XT-PIC  rtc
  9:          1          XT-PIC  acpi, uhci_hcd:usb1, ehci_hcd:usb4
 10:          0          XT-PIC  uhci_hcd:usb3
 11:      71772          XT-PIC  uhci_hcd:usb2, eth0
 12:     101150          XT-PIC  i8042
 14:      33146          XT-PIC  ide0
```

```
 15:     149202          XT-PIC  ide1
NMI:          0
LOC:          0
ERR:          0
MIS:          0
```

以下為一些重要檔案及其內容：

/proc/devices
: 可用設備

/proc/modules
: 載入的核心模組

/proc/cmdline
: 核心指令行

/proc/meminfo
: 關於記憶體使用的詳細資訊

/proc/config.gz
: gzip-已壓縮的核心目前執行中組態檔案

如需更多資訊，請參閱 /usr/src/linux/Documentation/filesystems/proc.txt。關於執行中程序的資訊，請參閱 /proc/*NNN* 目錄，其中 *NNN* 為相關程序的程序 ID (PID)。在 /proc/self/ 中可以找到每個程序的特性：

```
tux@mercury:~> ls -l /proc/self
lrwxrwxrwx 1 root root 64 2007-07-16 13:03 /proc/self -> 5356
tux@mercury:~> ls -l /proc/self/
total 0
dr-xr-xr-x 2 tux users 0 2007-07-16 17:04 attr
-r-------- 1 tux users 0 2007-07-16 17:04 auxv
-r--r--r-- 1 tux users 0 2007-07-16 17:04 cmdline
lrwxrwxrwx 1 tux users 0 2007-07-16 17:04 cwd -> /home/tux
-r-------- 1 tux users 0 2007-07-16 17:04 environ
lrwxrwxrwx 1 tux users 0 2007-07-16 17:04 exe -> /bin/ls
dr-x------ 2 tux users 0 2007-07-16 17:04 fd
-rw-r--r-- 1 tux users 0 2007-07-16 17:04 loginuid
-r--r--r-- 1 tux users 0 2007-07-16 17:04 maps
-rw------- 1 tux users 0 2007-07-16 17:04 mem
-r--r--r-- 1 tux users 0 2007-07-16 17:04 mounts
-rw-r--r-- 1 tux users 0 2007-07-16 17:04 oom_adj
-r--r--r-- 1 tux users 0 2007-07-16 17:04 oom_score
lrwxrwxrwx 1 tux users 0 2007-07-16 17:04 root -> /
-rw------- 1 tux users 0 2007-07-16 17:04 seccomp
-r--r--r-- 1 tux users 0 2007-07-16 17:04 smaps
```

```
-r--r--r-- 1 tux users 0 2007-07-16 17:04 stat
[...]
dr-xr-xr-x 3 tux users 0 2007-07-16 17:04 task
-r--r--r-- 1 tux users 0 2007-07-16 17:04 wchan
```

maps 檔中含有可執行檔和程式庫的位址指定：

```
tux@mercury:~> cat /proc/self/maps
08048000-0804c000 r-xp 00000000 03:03 17753      /bin/cat
0804c000-0804d000 rw-p 00004000 03:03 17753      /bin/cat
0804d000-0806e000 rw-p 0804d000 00:00 0          [heap]
b7d27000-b7d5a000 r--p 00000000 03:03 11867      /usr/lib/locale/en_GB.utf8/
b7d5a000-b7e32000 r--p 00000000 03:03 11868      /usr/lib/locale/en_GB.utf8/
b7e32000-b7e33000 rw-p b7e32000 00:00 0
b7e33000-b7f45000 r-xp 00000000 03:03 8837       /lib/libc-2.3.6.so
b7f45000-b7f46000 r--p 00112000 03:03 8837       /lib/libc-2.3.6.so
b7f46000-b7f48000 rw-p 00113000 03:03 8837       /lib/libc-2.3.6.so
b7f48000-b7f4c000 rw-p b7f48000 00:00 0
b7f52000-b7f53000 r--p 00000000 03:03 11842      /usr/lib/locale/en_GB.utf8/
[...]
b7f5b000-b7f61000 r--s 00000000 03:03 9109       /usr/lib/gconv/gconv-module
b7f61000-b7f62000 r--p 00000000 03:03 9720       /usr/lib/locale/en_GB.utf8/
b7f62000-b7f76000 r-xp 00000000 03:03 8828       /lib/ld-2.3.6.so
b7f76000-b7f78000 rw-p 00013000 03:03 8828       /lib/ld-2.3.6.so
bfd61000-bfd76000 rw-p bfd61000 00:00 0          [stack]
ffffe000-fffff000 ---p 00000000 00:00 0          [vdso]
```

17.5.1 procinfo

使用指令 procinfo 來摘述 /proc 檔案系統的重要資訊：

```
tux@mercury:~> procinfo
Linux 2.6.18.8-0.5-default (geeko@buildhost) (gcc 4.1.2 20061115) #1 2CPU

Memory:      Total        Used        Free      Shared     Buffers
Mem:       2060604     2011264       49340           0      200664
Swap:      2104472         112     2104360

Bootup: Tue Jul 10 10:29:15 2007    Load average: 0.86 1.10 1.11 3/118 21547

user  :       2:43:13.78   0.8%  page in :   71099181  disk 1:  2827023r 968
nice  :   1d 22:21:27.87  14.7%  page out:  690734737
system:      13:39:57.57   4.3%  page act:  138388345
IOwait:      18:02:18.59   5.7%  page dea:   29639529
hw irq:       0:03:39.44   0.0%  page flt: 9539791626
sw irq:       1:15:35.25   0.4%  swap in :         69
idle  :   9d 16:07:56.79  73.8%  swap out:        209
uptime:   6d 13:07:11.14         context :  542720687

irq  0: 141399308 timer          irq 14:   5074312 ide0
```

```
irq  1:     73784 i8042          irq 50:   1938076 uhci_hcd:usb1, ehci_
irq  4:         2                irq 58:         0 uhci_hcd:usb2
irq  6:         5 floppy [2]     irq 66:    872711 uhci_hcd:usb3, HDA I
irq  7:         2                irq 74:        15 uhci_hcd:usb4
irq  8:         0 rtc            irq 82: 178717720 0         PCI-MSI  e
irq  9:         0 acpi           irq169:  44352794 nvidia
irq 12:         3                irq233:   8209068 0         PCI-MSI  l
```

如果要檢視所有資訊，請使用參數 -a。參數 -nN 每隔 *N* 秒即更新資訊。在此範例中，按 Q 便可終止程式。

依照預設，這時會顯示累計的值。參數-d會產生差值。procinfo -dn5會顯示過去 5 秒內變更的值。

17.6 程序

17.6.1 程序間通訊：ipcs

指令 ipcs 可產生一份使用中的 IPC 資源清單：

```
------ Shared Memory Segments --------
key        shmid      owner     perms      bytes      nattch     status
0x00000000 58261504   tux    600        393216     2          dest
0x00000000 58294273   tux    600        196608     2          dest
0x00000000 83886083   tux    666        43264      2
0x00000000 83951622   tux    666        192000     2
0x00000000 83984391   tux    666        282464     2
0x00000000 84738056   root      644        151552     2          dest

------ Semaphore Arrays --------
key        semid      owner     perms      nsems
0x4d038abf 0          tux    600        8

------ Message Queues --------
key        msqid      owner      perms      used-bytes   messages
```

17.6.2 程序清單：ps

指令ps會產生一份程序清單。大部份的參數在寫入時都不可包含減號。請參閱 ps --help 取得簡短說明，或是參閱 man 頁面取得詳細的說明。

若要列出包含使用者和指令行資訊的所有程序，請使用 `ps axu`：

```
tux@mercury:~> ps axu
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0    696   272 ?        S    12:59   0:01 init [5]
root         2  0.0  0.0      0     0 ?        SN   12:59   0:00 [ksoftirqd
root         3  0.0  0.0      0     0 ?        S<   12:59   0:00 [events
[...]
tux    4047  0.0  6.0 158548 31400 ?        Ssl  13:02   0:06 mono-best
tux    4057  0.0  0.7   9036  3684 ?        Sl   13:02   0:00 /opt/gnome
tux    4067  0.0  0.1   2204   636 ?        S    13:02   0:00 /opt/gnome
tux    4072  0.0  1.0  15996  5160 ?        Ss   13:02   0:00 gnome-scre
tux    4114  0.0  3.7 130988 19172 ?        SLl  13:06   0:04 sound-juic
tux    4818  0.0  0.3   4192  1812 pts/0    Ss   15:59   0:00 -bash
tux    4959  0.0  0.1   2324   816 pts/0    R+   16:17   0:00 ps axu
```

若要檢查有多少個 `sshd` 程序正在執行，請一起使用 `-p` 選項和 `pidof` 指令來列出指定程序的程序 ID。

```
tux@mercury:~> ps -p `pidof sshd`
  PID TTY      STAT   TIME COMMAND
 3524 ?        Ss     0:00 /usr/sbin/sshd -o PidFile=/var/run/sshd.init.pid
 4813 ?        Ss     0:00 sshd: tux [priv]
 4817 ?        R      0:00 sshd: tux@pts/0
```

程序清單可依照您的需要採用格式。選項 `-L` 會傳回所有關鍵字的清單。輸入下列指令，發出一份依記憶體使用排序的程序清單：

```
tux@mercury:~> ps ax --format pid,rss,cmd --sort rss
  PID   RSS CMD
    2     0 [ksoftirqd/0]
    3     0 [events/0]
    4     0 [khelper]
    5     0 [kthread]
   11     0 [kblockd/0]
   12     0 [kacpid]
  472     0 [pdflush]
  473     0 [pdflush]
[...]
 4028 17556 nautilus --no-default-window --sm-client-id default2
 4118 17800 ksnapshot
 4114 19172 sound-juicer
 4023 25144 gnome-panel --sm-client-id default1
 4047 31400 mono-best --debug /usr/lib/beagle/Best.exe --autostarted
 3973 31520 mono-beagled --debug /usr/lib/beagle/BeagleDaemon.exe --bg --aut
```

17.6.3 程序網路樹：pstree

指令 `pstree` 會產生一份樹狀程序清單：

```
tux@mercury:~> pstree
init-+-NetworkManagerD
     |-acpid
     |-3*[automount]
     |-cron
     |-cupsd
     |-2*[dbus-daemon]
     |-dbus-launch
     |-dcopserver
     |-dhcpcd
     |-events/0
     |-gpg-agent
     |-hald-+-hald-addon-acpi
     |      `-hald-addon-stor
     |-kded
     |-kdeinit-+-kdesu---su---kdesu_stub---yast2---y2controlcenter
     |         |-kio_file
     |         |-klauncher
     |         |-konqueror
     |         |-konsole-+-bash---su---bash
     |         |         `-bash
     |         `-kwin
     |-kdesktop---kdesktop_lock---xmatrix
     |-kdesud
     |-kdm-+-X
     |     `-kdm---startkde---kwrapper
[...]
```

參數 `-p` 會將程序 ID 新增至指定的名稱。若要同時顯示指令行，請使用 `-a` 參數：

17.6.4 程序：top

指令 `top` (即「程序表 (table of processes)」的縮寫) 會顯示一個程序清單，每隔兩秒自動更新一次。若要終止程式，請按 Q。參數 `-n 1` 會在程序清單顯示一次之後，終止程式。以下是 `top -n 1` 指令的輸出範例：

```
tux@mercury:~> top -n 1
top - 17:06:28 up  2:10,  5 users,  load average: 0.00, 0.00, 0.00
Tasks:  85 total,   1 running,  83 sleeping,   1 stopped,   0 zombie
Cpu(s):  5.5% us,  0.8% sy,  0.8% ni, 91.9% id,  1.0% wa,  0.0% hi,  0.0% si
Mem:    515584k total,   506468k used,     9116k free,    66324k buffers
Swap:   658656k total,        0k used,   658656k free,   353328k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
    1 root      16   0   700  272  236 S  0.0  0.1   0:01.33 init
    2 root      34  19     0    0    0 S  0.0  0.0   0:00.00 ksoftirqd/0
    3 root      10  -5     0    0    0 S  0.0  0.0   0:00.27 events/0
    4 root      10  -5     0    0    0 S  0.0  0.0   0:00.01 khelper
    5 root      10  -5     0    0    0 S  0.0  0.0   0:00.00 kthread
   11 root      10  -5     0    0    0 S  0.0  0.0   0:00.05 kblockd/0
   12 root      20  -5     0    0    0 S  0.0  0.0   0:00.00 kacpid
  472 root      20   0     0    0    0 S  0.0  0.0   0:00.00 pdflush
  473 root      15   0     0    0    0 S  0.0  0.0   0:00.06 pdflush
  475 root      11  -5     0    0    0 S  0.0  0.0   0:00.00 aio/0
  474 root      15   0     0    0    0 S  0.0  0.0   0:00.07 kswapd0
  681 root      10  -5     0    0    0 S  0.0  0.0   0:00.01 kseriod
  839 root      10  -5     0    0    0 S  0.0  0.0   0:00.02 reiserfs/0
  923 root      13  -4  1712  552  344 S  0.0  0.1   0:00.67 udevd
 1343 root      10  -5     0    0    0 S  0.0  0.0   0:00.00 khubd
 1587 root      20   0     0    0    0 S  0.0  0.0   0:00.00 shpchpd_event
 1746 root      15   0     0    0    0 S  0.0  0.0   0:00.00 w1_control
 1752 root      15   0     0    0    0 S  0.0  0.0   0:00.00 w1_bus_master1
 2151 root      16   0  1464  496  416 S  0.0  0.1   0:00.00 acpid
 2165 messageb  16   0  3340 1048  792 S  0.0  0.2   0:00.64 dbus-daemon
 2166 root      15   0  1840  752  556 S  0.0  0.1   0:00.01 syslog-ng
 2171 root      16   0  1600  516  320 S  0.0  0.1   0:00.00 klogd
 2235 root      15   0  1736  800  652 S  0.0  0.2   0:00.10 resmgrd
 2289 root      16   0  4192 2852 1444 S  0.0  0.6   0:02.05 hald
 2403 root      23   0  1756  600  524 S  0.0  0.1   0:00.00 hald-addon-acpi
 2709 root      19   0  2668 1076  944 S  0.0  0.2   0:00.00 NetworkManagerD
 2714 root      16   0  1756  648  564 S  0.0  0.1   0:00.56 hald-addon-stor
```

如果您在 `top` 正在執行時按 **F**，這時會出現一個功能表，供您用來變更輸出格式。

參數 `-U` *UID* 只會監視與特定使用者有關的程序。以使用者的使用者 ID 來取代 *UID*。`top -U` `` `id -u` `` 指令會根據使用者名稱傳回使用者的 UID，並顯示其程序。

17.7 系統資訊

17.7.1 系統活動資訊：sar

若要使用sar，就必須執行sadc(系統活動資料收集器)。請檢查其狀態，或使用rcsysstat \{start|status\}來啟動它。

sar可以產生幾乎所有重要系統活動的充分報告，其中包括CPU、記憶體、IRQ用量、IO或網路。其選項繁多，因此很難在此細數。請參閱man頁面以取得舉例說明的延伸文件。

17.7.2 記憶體使用率：free

公用程式free會檢查RAM的使用狀況。顯示可使用和已使用的記憶體與交換區域的詳細資訊：

```
tux@mercury:~> free
             total       used       free     shared    buffers     cached
Mem:        515584     501704      13880          0      73040     334592
-/+ buffers/cache:      94072     421512
Swap:       658656          0     658656
```

-b、-k、-m、-g等選項會顯示以位元組(byte)、KB、MB或GB為單位的輸出。參數-d *延遲*會確定每隔*延遲*秒自動更新顯示內容。例如，free -d 1.5每1.5秒會更新一次。

17.7.3 使用者存取中檔案：fuser

這可用來判定哪些程序或使用者目前正存取著特定的檔案。例如，假設您要將裝載在/mnt上的檔案系統取消裝載，但umount指令傳回「設備忙碌」。接著，使用fuser指令來判定哪些程序正在存取此設備：

```
tux@mercury:~> fuser -v /mnt/*

                     USER        PID ACCESS COMMAND
/mnt/notes.txt       tux     26597 f....  less
```

當另一個終端機上所執行的less程序結束時，即可成功地解除裝載檔案系統。

17.7.4 核心環緩衝區：dmesg

Linux 核心會將某些訊息保存在環狀緩衝區內。如果要檢視這些訊息，請輸入指令 `dmesg`：

```
$ dmesg
[...]
end_request: I/O error, dev fd0, sector 0
subfs: unsuccessful attempt to mount media (256)
e100: eth0: e100_watchdog: link up, 100Mbps, half-duplex
NET: Registered protocol family 17
IA-32 Microcode Update Driver: v1.14 <tigran@veritas.com>
microcode: CPU0 updated from revision 0xe to 0x2e, date = 08112004
IA-32 Microcode Update Driver v1.14 unregistered
bootsplash: status on console 0 changed to on
NET: Registered protocol family 10
Disabled Privacy Extensions on device c0326ea0(lo)
IPv6 over IPv4 tunneling driver
powernow: This module only works with AMD K7 CPUs
bootsplash: status on console 0 changed to on
```

較早事件會記錄在檔案 `/var/log/messages` 和 `/var/log/warn` 中。

17.7.5 開啟檔案的清單：lsof

如果要檢視為了程序而開啟的所有檔案之清單及其程序 ID *PID*，可使用 `-p`。例如，要檢視目前外圍程序正在使用的所有檔案，可輸入：

```
tux@mercury:~> lsof -p $$
COMMAND  PID   USER   FD   TYPE DEVICE    SIZE   NODE NAME
bash    5552 tux  cwd    DIR    3,3    1512 117619 /home/tux
bash    5552 tux  rtd    DIR    3,3     584      2 /
bash    5552 tux  txt    REG    3,3  498816  13047 /bin/bash
bash    5552 tux  mem    REG    0,0              0 [heap] (stat: No such
bash    5552 tux  mem    REG    3,3  217016 115687 /var/run/nscd/passwd
bash    5552 tux  mem    REG    3,3  208464  11867 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3  882134  11868 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3 1386997   8837 /lib/libc-2.3.6.so
bash    5552 tux  mem    REG    3,3   13836   8843 /lib/libdl-2.3.6.so
bash    5552 tux  mem    REG    3,3  290856  12204 /lib/libncurses.so.5.5
bash    5552 tux  mem    REG    3,3   26936  13004 /lib/libhistory.so.5.1
bash    5552 tux  mem    REG    3,3  190200  13006 /lib/libreadline.so.5.
bash    5552 tux  mem    REG    3,3      54  11842 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3    2375  11663 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3     290  11736 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3      52  11831 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3      34  11862 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3      62  11839 /usr/lib/locale/en_GB.
```

```
bash    5552 tux  mem    REG    3,3     127  11664 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3      56  11735 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3      23  11866 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3   21544   9109 /usr/lib/gconv/gconv-m
bash    5552 tux  mem    REG    3,3     366   9720 /usr/lib/locale/en_GB.
bash    5552 tux  mem    REG    3,3   97165   8828 /lib/ld-2.3.6.so
bash    5552 tux    0u   CHR  136,5              7 /dev/pts/5
bash    5552 tux    1u   CHR  136,5              7 /dev/pts/5
bash    5552 tux    2u   CHR  136,5              7 /dev/pts/5
bash    5552 tux  255u   CHR  136,5              7 /dev/pts/5
```

特殊外圍程序變數 `$$`，其值就是外圍程序的程序 ID。

指令 `lsof` 列出所有目前開啟的檔案，使用時不需任何參數。因為通常其中有數千個檔案開啟，所以直接列出所有檔案並沒有太大用處。不過，可將這份包含所有檔案的清單結合搜尋功能，進而產生有用的清單。例如，將用到的字元設備全部列出：

```
tux@mercury:~> lsof | grep CHR
bash      3838     tux    0u      CHR  136,0                2 /dev/pts/0
bash      3838     tux    1u      CHR  136,0                2 /dev/pts/0
bash      3838     tux    2u      CHR  136,0                2 /dev/pts/0
bash      3838     tux  255u      CHR  136,0                2 /dev/pts/0
bash      5552     tux    0u      CHR  136,5                7 /dev/pts/5
bash      5552     tux    1u      CHR  136,5                7 /dev/pts/5
bash      5552     tux    2u      CHR  136,5                7 /dev/pts/5
bash      5552     tux  255u      CHR  136,5                7 /dev/pts/5
X         5646       root  mem       CHR    1,1            1006 /dev/mem
lsof      5673     tux    0u      CHR  136,5                7 /dev/pts/5
lsof      5673     tux    2u      CHR  136,5                7 /dev/pts/5
grep      5674     tux    1u      CHR  136,5                7 /dev/pts/5
grep      5674     tux    2u      CHR  136,5                7 /dev/pts/5
```

17.7.6 核心和 udev 事件順序檢視器：udevmonitor

`udevmonitor` 聆聽核心 uevents 和 udev 規則送出的事件，並列印事件到主控台的設備路徑 (DEVPATH)。這是連接 USB 隨身碟時的事件順序：

```
UEVENT[1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2
UEVENT[1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UEVENT[1138806687] add@/class/scsi_host/host4
UEVENT[1138806687] add@/class/usb_device/usbdev4.10
UDEV  [1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2
UDEV  [1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UDEV  [1138806687] add@/class/scsi_host/host4
UDEV  [1138806687] add@/class/usb_device/usbdev4.10
UEVENT[1138806692] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UEVENT[1138806692] add@/block/sdb
UEVENT[1138806692] add@/class/scsi_generic/sg1
UEVENT[1138806692] add@/class/scsi_device/4:0:0:0
UDEV  [1138806693] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UDEV  [1138806693] add@/class/scsi_generic/sg1
UDEV  [1138806693] add@/class/scsi_device/4:0:0:0
UDEV  [1138806693] add@/block/sdb
UEVENT[1138806694] add@/block/sdb/sdb1
UDEV  [1138806694] add@/block/sdb/sdb1
UEVENT[1138806694] mount@/block/sdb/sdb1
UEVENT[1138806697] umount@/block/sdb/sdb1
```

17.7.7 X11 用戶端使用的伺服器資源：xrestop

`xrestop` 針對每個連接之 X11 用戶端的伺服器端資源提供統計資料。其輸出與第 17.6.4 節「程序：top」[312頁]非常類似。

```
xrestop - Display: localhost:0
          Monitoring 40 clients. XErrors: 0
          Pixmaps:   42013K total, Other:     206K total, All:   42219K total

res-base Wins  GCs Fnts Pxms Misc   Pxm mem  Other   Total   PID Identifier
3e00000   385   36    1  751  107    18161K    13K  18175K   ?   NOVELL: SU
4600000   391  122    1 1182  889     4566K    33K   4600K   ?   amaroK - S
1600000    35   11    0   76  142     3811K     4K   3816K   ?   KDE Deskto
3400000    52   31    1   69   74     2816K     4K   2820K   ?   Linux Shel
2c00000    50   25    1   43   50     2374K     3K   2378K   ?   Linux Shel
2e00000    50   10    1   36   42     2341K     3K   2344K   ?   Linux Shel
2600000    37   24    1   34   50     1772K     3K   1775K   ?   Root - Kon
4800000    37   24    1   34   49     1772K     3K   1775K   ?   Root - Kon
2a00000   209   33    1  323  238     1111K    12K   1123K   ?   Trekstor25
1800000   182   32    1  302  285     1039K    12K   1052K   ?   kicker
1400000   157  121    1  231  477      777K    18K    796K   ?   kwin
3c00000   175   36    1  248  168      510K     9K    520K   ?   de.comp.la
3a00000   326   42    1  579  444      486K    20K    506K   ?   [opensuse-
0a00000    85   38    1  317  224      102K     9K    111K   ?   Kopete
4e00000    25   17    1   60   66       63K     3K     66K   ?   YaST Contr
2400000    11   10    0   56   51       53K     1K     55K 22061 suseplugge
0e00000    20   12    1   50   92       50K     3K     54K 22016 kded
3200000     6   41    5   72   84       40K     8K     48K   ?   EMACS
2200000    54    9    1   30   31       42K     3K     45K   ?   SUSEWatche
```

```
4400000     2   11    1   30   34       34K      2K     36K 16489 kdesu
1a00000   255    7    0   42   11       19K      6K     26K   ?   KMix
3800000     2   14    1   34   37       21K      2K     24K 22242 knotify
1e00000    10    7    0   42    9       15K    624B     15K   ?   KPowersave
3600000   106    6    1   30    9        7K      3K     11K 22236 konqueror
2000000    10    5    0   21   34        9K      1K     10K   ?   klipper
3000000    21    7    0   11    9        7K    888B      8K   ?   KDE Wallet
```

17.8 使用者資訊

17.8.1 功能介紹：w

使用指令 `w`，可清楚掌握登入系統的使用者和他們的動作。例如：

```
tux@mercury:~> w
 16:33:03 up  3:33,  2 users,  load average: 0.14, 0.06, 0.02
USER     TTY        LOGIN@   IDLE   JCPU   PCPU WHAT
tux   :0         16:33   ?xdm?   9.42s  0.15s /bin/sh /opt/kde3/bin/startk
tux   pts/0      15:59    0.00s  0.19s  0.00s w
```

如果其他系統的使用者從遠端登入，參數 `-f` 會顯示該使用者用來連接的電腦。

17.9 時間和日期

17.9.1 使用 time 進行時間管理

使用 `time` 公用程式來確定指令所花費的時間。這個公用程式有兩個版本，分別為外圍程序內建和程式 (`/usr/bin/time`)。

```
tux@mercury:~> time find . > /dev/null

real    0m4.051s
user    0m0.042s
sys     0m0.205s
```

18 使用外圍程序

Linux 系統開機後，您通常會進入圖形使用者介面，由它引導您執行登入程序以及後續與系統的互動。雖然圖形使用者介面已經變得非常重要和易於使用，但使用圖形使用者介面並不是與系統互動的唯一方式。您也可以使用文字導向的溝通方式，例如一般稱為外圍程序的指令行解譯器就可以讓您輸入指令。因為 Linux 提供從圖形使用者介面啟動外圍程序視窗的選項，所以您可以輕鬆使用兩種方式。

在管理工作上，若要控制慢速網路上的電腦，或是要以 `root` 身份在指令行執行任務，以外圍程序為基礎的應用程式尤其重要。對「Linux「新手」而言，在外圍程序中輸入指令或許有點不正常，但您很快就會瞭解到，外圍程序不只是系統管理員專用的—其實使用外圍程序，往往是執行一些日常任務最快、最簡單的方法。

UNIX 或 Linux 有許多種外圍程序。SUSE® Linux Enterprise 中的預設外圍程序是 Bash (GNU Bourne-Again Shell)。

本章會討論您在使用外圍程序時必須知道的幾個基本要點。這裡包含下列主題：如何輸入指令、Linux 的目錄結構、如何處理檔案和目錄、如何使用一些基本功能、Linux 的使用者和許可權概念、重要外圍程序指令的綜覽和 vi 編輯器 (這是 Unix 和 Linux 系統一定會提供的預設編輯器) 的簡介。

18.1 Bash 外圍程序入門

在 Linux 中，您可以使用功能和圖形使用者介面相同的指令行，並且輕鬆切換使用它們。若要在 KDE 中從圖形使用者介面來啟動終端機視窗，請按一下面板

中的 Konsole 圖示。在 GNOME 中，則請按一下面板的「GNOME 終端機」圖示。

這樣畫面就會出現 Konsole 或是 GNOME 終端機視窗，並在第一行顯示提示文字，如 圖形 18.1「Bash 終端機視窗範例」[320頁] 所示。這段提示文字通常會顯示您的登入名稱 (即本範例中的 `tux`)、電腦的主機名稱 (即此處的 `knox`)，以及目前的路徑 (在此範例中就是用波狀符號 ~ 標示的根目錄)。如果您是登入遠端電腦，這段資訊就會永遠為您顯示目前您正在運作的系統。當游標出現在這個提示名稱之後，您可以將指令直接傳送給電腦系統。

圖形 18.1 *Bash 終端機視窗範例*

18.1.1 輸入指令

指令是由幾個元件所組成。第一個元件一定是實際的指令，後面跟著參數或選項。您可以輸入指令，然後使用 ←、→、<—、Del 和 Space 進行編輯。您也可以加入選項，或是更正輸入錯誤。這個指令會在您按 Enter 之後開始執行。

重要：沒消息就是好消息

外圍程序不會提供詳細資訊 (Verbose)：相對於某些圖形使用者介面，通常外圍程序不會在指令完成執行時提供確認訊息。訊息只會在發生問題或錯誤時才會出現。

同時，請注意會刪除物件的指令。當您在輸入類似 `rm` 等刪除檔案的指令之前，請務必考慮自己是否真的要刪除該物件：因為指令執行之前不會先詢問您，並永遠刪除該物件。

使用不包含選項的指令

請透過下列簡單範例來觀察指令結構：用來列出目錄內容的 `ls` 指令。這個指令可包含或不包含選項情況下使用。只輸入 `ls` 指令可以顯示目前目錄的內容：

圖形 18.2 *ls 指令*

不同於其他作業系統，Linux 中的檔案可能包含副檔名 (例如 `.txt`)，但是並不是絕對需要。因此這會造成 `ls` 輸出中的檔案和資料夾難以進行區分。依預設，您可以透過顏色來區分：目錄通常會顯示為藍色，檔案會顯示為黑色。

使用包含選項的指令

另外一種取得目錄內容更多細節的較佳方法，就是搭配選項字串來使用 `ls` 指令。選項可以改變指令執行的方式，讓您能夠指引它執行特定的任務。選項會用空白來與指令區隔，而且開頭會加上連字號。指令 `ls -l` 可以完整詳細顯示相同目錄的內容 (長式清單)：

圖形 18.3 *ls -l 指令*

在每個物件名稱的左邊會有好幾個欄位，其中會顯示與該物件有關的資訊。最重要的是：第一欄會顯示物件的檔案類型 (在本範例中，d 是指目錄，而 - 是指一般檔案)。接下來的 9 個欄位則會顯示該物件的使用者許可權。欄位 11 和 12 則會顯示檔案擁有者和群組 (在本範例中是指 tux 和 users)。如需更多有關 Linux 使用者許可權和使用者概念的詳細資訊，請參閱第 18.2 節「使用者和存取許可權」[330頁]。接下來的欄位會顯示檔案大小，單位是位元組。接著顯示上次變更的日期和時間。最後一個欄位則顯示物件名稱。

如果您希望檢視更多資訊，請結合兩個選項來執行 ls 指令，即輸入 ls -la。這時外圍程序會同時在目錄中顯示隱藏檔案，即前面標示點符號的檔案 (例如，.hiddenfile)。

使用說明

並非所有人都必須記住所有指令的所有選項。如果您記得指令名稱但是不確定相關選項，您可以輸入指令，依序在後面加上空格以及 --help。有許多指令可以使用這個 --help 選項。輸入 ls --help，可顯示有關 ls 指令的所有選項。

18.1.2 Linux 目錄結構

因為外圍程序不會像檔案管理員以樹狀結構檢視方式來顯示圖形綜覽的目錄和檔案，因此對 Linux 系統的預設目錄結構有些基本瞭解是很有幫助的。您可以將目錄視為儲存檔案、程式及子目錄的電子資料夾。階層中最上層的目錄就是根目錄，以 / 表示。從此處可以存取其他所有目錄的位置。

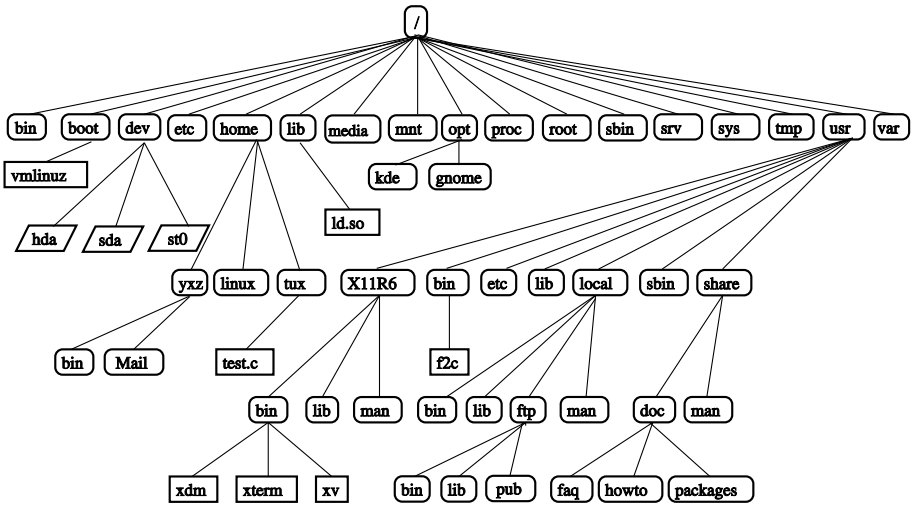
圖形 18.4 顯示 linux 的標準目錄網路樹，以及範例使用者的主目錄 yxz、linux 和 tux。/home 目錄包含個別使用者可以儲存個人檔案的目錄。

注：網路環境中的主目錄

當您正在網路環境中工作時，您的主目錄並不一定會稱為 /home。它可能會映射到檔案系統的任何目錄。

下列清單提供 Linux 中標準目錄的簡短描述。

圖形 18.4　摘錄自標準目錄網路樹



表格 18.1　標準目錄網路樹的綜覽

/	根目錄，目錄網路樹的起點
/home	使用者個人目錄
/dev	代表硬體元件的設備檔案
/etc	系統組態的重要檔案
/etc/init.d	開機程序檔
/bin、/sbin	開機程序早期需要的程式 (/bin)，和供管理員使用的程式 (/sbin)
/usr、/usr/local	所有應用程式和本地的分散獨立式延伸 (/usr/local)
/usr/bin、/usr/sbin	一般可存取的程式 (/usr/bin) 和保留供系統管理員使用的程式 (/usr/sbin)

/usr/share/doc	各種文件檔
/tmp、/var/tmp	暫存檔(勿將需要的檔案儲存在此目錄中)
/opt	選擇性軟體，大型的加入套裝程式 (例如 KDE、GNOME、Netscape)
/proc	程序檔案系統
/sys	系統檔案系統，為核心之所有設備資訊的集中存放位置
/var/log	系統記錄檔案

18.1.3 使用目錄和檔案

若要為特定檔案或目錄指定位置，您必須指定指向該目錄或檔案的路徑。指定路徑的方法有下列兩種：

- 從根目錄到個別檔案的完整 (絕對) 路徑
- 從目前目錄作為起點的路徑 (相對路徑)

絕對路徑永遠會以斜線作為開頭。相對路徑不會以斜線作為開頭。

注：Linux 會區分大小寫

Linux 會區分檔案系統中的大寫和小寫。例如，輸入 test.txt 或 Test.txt 對 Linux 來說是不同的意義。請在輸入檔名或路徑時注意這點。

若要變更目錄，請使用 cd 指令。

- 若要切換至您的主目錄，請輸入 cd。
- 使用一個點可表示目前的目錄 (.)。這主要用於其他指令 (cp、mv、...)。
- 網路樹中的上一層目錄是由兩個點表示(..)。例如，若要切換至目前目錄的父目錄，請輸入 cd ..。

指定檔案位址範例

第 18.1.3 節「使用目錄和檔案」[324頁]中的 cd 指令是使用相對路徑。您可以使用絕對路徑。舉例來說，假設您希望將主目錄的檔案複製到 /tmp 的子目錄：

1 首先，請從主目錄建立位在 /tmp 的子目錄：

1a 如果目前您不是在自己的主目錄下，請輸入 cd ~ 切換至您的主目錄。無論您是在檔案系統的任何位置，您只要輸入 cd ~ 就可切換到您的主目錄。

1b 在您的主目錄中，輸入 mkdir /tmp/test。mkdir 代表「make directory (建立目錄)」。這個指令會在 /tmp 目錄中建立名為 test 的新目錄。這個範例是使用絕對路徑來建立目錄。

1c 為了檢查產生的結果，現在讓我們輸入 ls -l /tmp。這時 /tmp 目錄的內容清單中應該會出現 test 這個新目錄。

2 現在，請在主目錄中建立新檔案，並使用相對路徑將其複製到 /tmp/test 目錄。

2a 輸入 touch myfile.txt。配合 myfile.txt 選項的 touch 指令，將會在目前目錄中建立一個全新、名為 myfile.txt 的空白檔案。

2b 請輸入 ls -l，檢查執行結果。這個新檔案應該會出現在內容清單中。

2c 輸入 cp myfile.txt ../tmp/test。這樣會將 myfile.txt 複製到目錄 /tmp/test 中，而且不改變檔案名稱。

2d 請輸入 ls -l /tmp/test，檢查執行結果。這時 /tmp/test 目錄的內容清單中應該會出現 myfile.txt 這個新檔案。

若要列示其他使用者的主目錄內容，請輸入 ls ~*username*。在圖形 18.4「摘錄自標準目錄網路樹」[323頁]的範例網路目錄樹中，其中一個範例使用者是 tux。在此範例例中，ls ~tux 會列示 tux 的主目錄內容。

注：處理檔名或目錄名稱中的空白

如果檔名中包含空格，請在空白前面使用反斜線 (\) 來避免發生空格，或是將檔名包括在單引號或是雙引號中。否則，Bash 會將類似 `My Documents` 的檔案名稱，解譯成為兩個檔案或是兩個目錄。單引號和雙引號的差別，在於使用雙引號時會產生不同的展開方式。使用單引號，可以確保外圍程序只會解譯單引號所括住的字串。

18.1.4 有用的外圍程序功能

使用 Bash 輸入指令可以包含大量輸入。下段內容將介紹一些 Bash 功能，讓您的工作能夠更輕鬆，並省去大量輸入的需要。

歷程和補齊

依預設，Bash 會「記憶」您輸入過的指令。這項功能稱為*歷程*。若要重複需要的指令，請按↑，直到該指令顯示在指令提示字元。按↓可往前翻閱以前輸入的指令清單。使用 Ctrl + R 可在歷程中搜尋。

按 Enter 以執行選取的指令之前，您可以編輯該指令，例如變更檔案的名稱。若要編輯指令行，只要使用方向鍵將游標移至想要的位置，然後開始輸入。

在輸入開頭字母之後補齊成該檔案或目錄的完整檔案名稱，是 Bash 提供的另外一項有用工具。若要這樣做，請輸入第一個字母，然後按 →|。如果該檔名或路徑可以唯一識別，該名稱就會立刻補齊，而且游標會移至該檔名的最後面。您可以接著輸入指令的下一個選項 (若有需要)。如果檔名或路徑無法提供唯一識別 (因為有好幾個檔名開頭都是相同字母)，檔名或路徑就只會重複補齊到可能有好幾種選項的位置。您可以接著第二次按 →|，取得這些選項的清單。在這個動作之後，您可以輸入檔案或路徑的下一個字母，然後按→| 再嘗試補齊。在配合 →| 補齊檔名和路徑時，您可以同時檢查您要輸入的檔案或路徑是否確實存在 (而且可以確定提供正確拼字)。

萬用字元

外圍程序提供的另一個便利之處，就是可以在路徑展開時使用萬用字元。萬用字元是指可以代表其他字元的字元。Bash 共用三種不同的萬用字元類型：

?
完全相符的任何字元

*
符合任何數目的字元

[*set*]
符合方括號中指定群組中的某個字元，在此是由字串 *set* 表示。在使用 *set* 時，您也可以使用語法 *[:class:]* 來指定字元類別，其中可指定 alnum、alpha 與 ascii 等類別。

利用！或 ^，於群組 ([!*set*]) 開端，會符合 *set* 以外的任一個字元。

假設您的 test 目錄包含 Testfile、Testfile1、Testfile2 和 datafile 幾個檔案。

- 指令 ls Testfile? 會列出 Testfile1 和 Testfile2 這兩個檔案。
- 指令 ls Testfile? 會列出 Testfile1 和 Testfile2 這兩個檔案。
- 使用 ls Test*，清單也會包括 Testfile。
- 指令 ls *fil* 會顯示所有範例檔案。
- 使用 set 萬用字元處理結尾字元為數字的所有範例檔案：ls Testfile[1-9]，或使用類別 ls Testfile[[:digit:]]。

在三種萬用字元中，應用最廣泛的是星號。它可以用來將某一目錄中的所有檔案，複製到另一個目錄，或者使用一個指令來刪除所有檔案。例如指令 rm *fil*，會刪除目前目錄中，名稱中包含字串 *fil* 的所有檔案。

使用 Less 和 More 來檢視檔案

Linux 包括兩個可以直接在外圍程序檢視文字檔案的小程式：less 和 more。不必啟動編輯器就可以讀取檔案，例如 Readme.txt，只要輸入 less Readme.txt 便可在主控台視窗中顯示文字。使用空格鍵往下捲動一頁。使用 Page Up 和 Page Down 鍵，往前或往回捲動文件。若要結束 less，請按 Q。

除了 less 外，您也可以使用較舊的 more 程式。不過，因為它不能讓您往回捲動，所以不是很方便。

程式 less 的名稱來源是得自 *less is more (少即是多)* 的俗諺，也且可以方便用來檢視指令的輸出。要知道如何使用，請參閱章節「重新導向和管道」[328頁]。

重新導向和管道

正常情況下，外圍程序的標準輸出是您的螢幕或主控台視窗，而標準輸入是鍵盤。然而，外圍程式還提供一項功能，可讓您將輸入或輸出重新導向到另一個物件，例如檔案或其他指令。舉例來說，在配合 > 和 < 符號情況下，您可以將指令輸出轉遞給檔案 (輸出重新導向)，或是將檔案當作指令的輸入來使用 (輸入重新導向)。例如，當您希望執行將類似 `ls` 的指令輸出寫入到檔案時，請輸入 `ls -l > file.txt`。這樣就可建立名為 `file.txt` 的檔案，其中會包含由 `ls` 指令所產生您目前所在目錄的內容。然而，如果已經有存在檔名 `file.txt` 的檔案，這個指令就會覆寫現有的檔案。若要預防這個情形，請使用 `>>`。這樣在輸入 `ls -l >> file.txt` 之後，便只會將 `ls` 指令的輸出結果附加到目前已存在的 `file.txt` 檔案。如果這個檔案不存在，接著就會建立。

有時這個功能也可以將檔案作為指令的輸入。舉例來說，您可以使用 `tr` 指令來置換從檔案重新導向的字元，再將結果寫入標準輸出，即您的螢幕。假設您要將上述範例所指 `file.txt` 中的所有 `t` 字元置換成 `x`，並將結果列印到螢幕上。輸入 `tr t x < file.txt` 便可完成這項工作。

和標準輸出一樣，標準錯誤輸出也會傳送至主控台。若要將標準錯誤輸出重新導向至名稱為 `errors` 的檔案，請在相對應的指令附加 `2> errors`。如果您附加 `>& alloutput`，標準輸出和標準錯誤會儲存至名稱為 `alloutput` 的檔案。

使用*管線*或*管道*也是一種重新導向，雖然使用管道時不會受到檔案限制。配合 (|) 時，您可以結合好幾種指令，將其中一項指令的輸出當作下一個指令的輸入。舉例來說，若要使用 `less` 來檢視內容或是目前所在目錄，請輸入 `ls | less`。這種做法只有在使用 `ls` 的標準輸出太長的時候，才有意義。例如，如果您檢視 `dev` 目錄內容時使用 `ls /dev`，視窗中只會看到一小部份。此時，請使用 `ls /dev | less` 檢視完整清單。

18.1.5 歸檔和資料壓縮

現在您已經建立一些檔案和目錄，請考慮歸檔和資料壓縮的用途。假設您將整個 test 目錄包裝成一個檔案，您可以儲存至 USB 隨身碟當成備份或透過電子郵件傳送。若要執行這個動作，請使用指令 tar (用於*磁帶歸檔設備*)。使用 tar --help，檢視 tar 指令的所有選項。在此說明最重要的選項：

-c
: (用於建立) 建立新歸檔。

-t
: (用於表格) 顯示歸檔的內容。

-x
: (用於擷取) 解開歸檔。

-v
: (用於詳細) 建立歸檔時，在螢幕顯示所有檔案。

-f
: (用於檔案) 選擇要歸檔的檔案名稱。建立歸檔時，此選項必須永遠在最後面。

若要將 test 目錄含所有檔案與子目錄包裝至一個名稱為 testarchive.tar 的歸檔，請執行下列動作：

1. 開啟外圍程序。
2. 使用 cd 切換到您的主目錄，test 目錄就位在該目錄中。
3. 輸入 tar -cvf testarchive.tar test。-c 選項會建立歸檔，使它依照 -f 的指示成為一個檔案。-v 選項會在處理過程中列出檔案。
4. 使用 tar -tf testarchive.tar 檢視歸檔檔案的內容。

test 目錄及其所有檔案與目錄會在磁碟中保留不變。若要解開歸檔，請輸入 tar -xvf testarchive.tar，但此時請先不要嘗試此選項。

如果要進行檔案壓縮，較常使用的選擇是使用 gzip，或者使用 bzip2 來取得更好的壓縮比例。 只要輸入 gzip testarchive.tar (或 bzip2

testarchive.tar，但這個範例是使用 gzip)。使用 ls，現在查看檔案 testarchive.tar 是否已經不在該處，而且已經建立 testarchive.tar .gz 檔案。此檔案比較小，因此更適合透過電子郵件傳送或儲存在 USB 隨身碟。

現在，在先前建立的 test2 目錄解開此檔案。要這樣做，請輸入 cp testarchive.tar.gz test2，將檔案複製至該目錄。使用 cd test2 變更至目錄。副檔名為 .tar.gz 的壓縮歸檔，可以使用 gunzip。輸入 gunzip testarchive.tar.gz，會產生檔案 testarchive.tar，然後需要使用 tar -xvf testarchive.tar 來解開或*展開*。您也可以透過 tar -xvf testarchive.tar.gz (不用再加入 -z 選項)，用一個步驟就解壓縮和擷取壓縮歸檔。使用 ls，您可以看到新的 test 目錄已經建立，而且與主目錄的 test 目錄，內容相同。

18.1.6 清除

在此課程之後，您應該熟悉 Linux 外圍程序或指令行的基礎概念。您可以使用 rm 和 rmdir 指令，刪除不同的測試檔案與目錄，清除您的主目錄。請在第 18.3 節「重要的 Linux 指令」[333頁]中，找出最重要指令的清單及其功能的簡要說明。

18.2 使用者和存取許可權

因為它早在 1990 年代便開始，Linux 已經開發成多重使用者系統。任何數目的使用者都可以同時在上面工作。使用者需要登入系統，才能在他們的工作站開始工作階段。每一個使用者都有一個使用者名稱以及相應的密碼。這種使用者區隔，可以確保未授權的使用者，無法看到他們沒有存取許可權的檔案。對系統的較大變更，例如安裝新程式，一般使用者通常也無法執行或被限制。只有根使用者或*超級使用者*，可以不受限制對系統進行變更以及自由存取所有檔案。在需要時才以完整 root 存取權登入，只要是遵循此概念的人就可以免除意外損失資料的風險。因為在正常情況下，只有 root 才可以刪除檔案或格式化硬碟，因此*特洛伊木馬效應*或意外輸入破壞性指令的情形，都可以大大降低。

18.2.1 檔案系統許可權

基本上，Linux檔案系統的每一個檔案都屬於一個使用者和一個群組。可以授權這些私有群組和所有其他人寫入、讀取或執行這些檔案。

在本案例中，群組可以定義成一組連接的使用者，具備特定的結合權限。例如，定義可以在特定專案 `project3` 工作的群組。Linux 系統的每一位使用者，至少屬於一個私有群組的成員，一般是`users`。需要時可以在系統新增很多群組，但只有 `root` 才可以新增群組。每一個使用者都可以使用指令 `groups`，知道成員的所屬群組。

檔案存取

檔案系統中的許可權組織，會因檔案和目錄而不同。檔案許可權資訊可以使用指令 `ls -l` 顯示。其輸出結果看起來可能會像範例 18.1「顯示檔案許可權的範例輸出」[331頁]。

範例 18.1 *顯示檔案許可權的範例輸出*

```
-rw-r----- 1 tux project3 14197 Jun 21  15:03 Roadmap
```

如第三欄顯示，此檔案屬於使用者 `tux`。它被指定至群組 `project3`。若要探查 `Roadmap` 檔案的使用者許可權，必須更詳細檢查第一欄。

-	rw-	r--	---
類型	使用者許可權	群組許可權	其他使用者許可權

此欄是由一個前導字元，後面加上 9 個字元 (3 個為一組) 所組成的。前 10 個字母標示檔案系統元件的類型。連字號 (`-`) 表示它是一個檔案。同時也可以表示目錄 (`d`)、連結 (`l`)、區塊設備 (`b`) 或字元設備。

後面的三個區塊都會依循標準樣式。前三個字元指出檔案是否可以讀取 (`r`) 或無法讀取 (`-`)。中間部分的 `w` 代表可以編輯相應的物件，而連字號 (`-`) 表示無法寫入檔案。第三個位置的 `x`，代表物件可以執行。因為此範例中的檔案是文字檔，而且不是可以執行的檔案，所以不需要此特殊檔案的執行權限。

在此範例中，`tux` 是檔案 `Roadmap` 的擁有者，擁有讀取 (`r`) 和寫入存取權 (`w`)，但無法執行它 (`x`)。群組 `project3` 的成員可以讀取檔案，不過無法

修改或執行。其他使用者沒有此檔案的任何存取權。其他許可權可以透過 ACL (存取控制清單) 來指定。

目錄許可權
: 目錄的存取許可權類型為 d。如果是目錄，個別許可權在意義上會稍稍不同。

範例 18.2 *顯示目錄許可權的範例輸出*

```
drwxrwxr-x 1 tux project3 35 Jun 21 15:15  ProjectData
```

在 範例 18.2「顯示目錄許可權的範例輸出」[332頁] 中，可以輕易辨別目錄 ProjectData 的擁有者 (tux) 和所屬群組 (project3)。不同於 檔案存取 [331頁] 的存取許可權，設定讀取許可權 (r) 表示可以顯示目錄的內容。寫入許可權 (w) 表示可以建立新檔案。執行許可權 (x) 表示使用者可以切換到這個目錄。在上面的範例中，這表示使用者 tux 以及群組 project3 的成員可以切換到 ProjectData 目錄 (x)、檢視內容 (r) 以及新增或刪除檔案 (w)。另一方面，其他的使用者被授予較少的存取權。他們可以進入目錄 (x) 以及瀏覽該目錄 (r)，但無法插入任何新檔案 (w)。

18.2.2 修改檔案許可權

變更存取許可權
: 擁有者和 root 可以使用指令 chmod 加上變更許可權的參數和一或多個檔案名稱，變更檔案或目錄的存取許可權。這些參數會形成不同類別：

1. 關於使用者

- u (*使用者*)—檔案的擁有者
- g (*群組*)—擁有檔案的群組
- o (*其他*)—其他使用者 (若未指定參數，變更會套用至所有類別)

2. 用於刪除 (-)、設定 (=) 或插入 (+) 的字元

3. 縮寫

- r—*讀取*

- w—*寫入*
- x—*執行*

4. 檔案名稱，或由空格分隔的多個檔案名稱

例如，如果 範例 18.2「顯示目錄許可權的範例輸出」[332頁] 中的使用者 tux 也想授予目錄 ProjectData 的寫入 (w) 存取權給其他人，他可以使用指令 chmod o+w ProjectData 來執行這個動作。

不過，如果他想拒絕自己以外的所有使用者的寫入許可權，他可以輸入指令 chmod go-w ProjectData 來執行這個動作。若要禁止所有使用者新增檔案至 ProjectData 資料夾，請輸入 chmod -w ProjectData。現在，即使是擁有者也必須先重建寫入許可權，然後才能在目錄中建立新檔案。

變更擁有權許可權

其他控制檔案系統元件擁有權與許可權的重要指令是 chown (變更擁有者) 和 chgrp (變更群組)。指令 chown 可將檔案擁有權移交給其他使用者。不過，只允許 root 執行此變更。

假設 範例 18.2「顯示目錄許可權的範例輸出」[332頁] 的檔案 Roadmap 不應該再屬於 tux，而是屬於使用者 geeko，則 root 應該輸入 chown geeko Roadmap。

chgrp 會變更檔案的群組擁有權。不過，檔案的擁有者必須是新群組的成員。使用此方法，只要 範例 18.1「顯示檔案許可權的範例輸出」[331頁] 的使用者 tux 是新群組的成員，便可以使用指令 chgrp project4 ProjectData，將所擁有檔案 ProjectData 的群組切換成 project4。

18.3 重要的 Linux 指令

此章節將帶您瞭解更重要的指令。本章節列示很多指令。除了列示個別指令、參數之外，會在適當之處，介紹典型的範例應用程式。若要深入瞭解各種指令，請使用 man 後加上指令名稱，取得並使用手冊頁，例如，man ls。

在 man 頁面，使用 PgUp 和 PgDn，上下移動。使用 Home 和 End，在文件的開頭和結尾之間移動。按 Q 結束此檢視模式。使用 man man，可詳細瞭解 man 指令本身的資訊。

在以下的介紹中，個別指令元素會以不同的字體表示。實際的指令以及它的必要選項，永遠會列印成指令選項。不要求一定要放在 [方括號] 中的規格或參數。

視您的需要調整設定。如果沒有名稱為 file 的檔案存在，就不需要寫入 ls file。您通常可以結合數個參數，例如使用 ls -la 而不是 ls -l -a。

18.3.1 檔案指令

下節列出最重要的檔案管理指令。其中涵蓋一般檔案管理以及檔案系統 ACL 的操作。

檔案管理

ls [選項] [檔案]
: 如果您執行 ls 而未加上任何其他參數，程式會以簡要格式，列示所在目錄的內容。

  -l
  : 詳細清單

  -a
  : 顯示隱藏的檔案

cp [選項] 來源 目標
: 複製來源到目標。

  -i
  : 在現有目標被覆寫之前，視需要等候確認

  -r
  : 循環複製 (包括子目錄)

mv [選項] 來源 目標
: 複製來源到目標，然後刪除原始的來源。

  -b
  : 移動前先建立來源的備份副本

-i
覆寫現有的 targetfile 之前，等候確認 (若有需要)

rm [選項] 檔案
從檔案系統移除指定的檔案。除非使用選項 -r，否則 rm 無法移除目錄。

-r
刪除任何現有的子目錄

-i
刪除每一個檔案前，等候確認

ln [選項] 來源 目標
建立從來源到目標的內部連結。通常，這種連結會直接指向同一檔案系統上的來源。不過，如果執行 ln 時使用 -s 選項，它會建立符號連結，只會指向來源所在的目錄位置，提供跨檔案系統的連結功能。

-s
建立符號連結

cd [選項] [目錄]
切換目前的目錄。cd 未加任何參數可以切換到使用者的主目錄。

mkdir [選項] 目錄
建立新目錄。

rmdir [選項] 目錄
如果指定的目錄已經是空的，則會刪除該目錄。

chown [選項] 使用者名稱[:[群組]] 檔案
將檔案的擁有權轉移給具有指定的使用者名稱的使用者。

-R
變更所有子目錄中的檔案與目錄

chgrp [選項] 群組名稱 檔案
將指定檔案的群組擁有權，移交至指定的群組。如果成員屬於目前群組和新群組，則檔案擁有者只可以變更群組擁有權。

chmod [選項] 模式 檔案

變更存取許可權。

模式 參數具有三個部份：群組、存取，與存取類型。group 可接受下列字元：

u
使用者

g
群組

o
其他

至於 access，使用 + 可授予存取權，使用 – 則可拒絕授予權限。

access type 將提供下列控制選項：

r
讀取

w
寫入

x
執行—執行檔案或變更至目錄

s
Setuid 位元—應用程式或程式可以依照檔案擁有者方式，進行啟動

做為替代方法，可以使用數字程式碼。此程式碼的四個位數組成值 4、2 和 1 的總合—二進位遮罩的十進位結果。第一個位數會設定使用者 ID (SUID) (4)，設定群組 ID (2) 以及黏貼 (1) 位元。第二個位數定義檔案擁有者的許可權。第三個位數定義群組成員的權限，最後的位數會設定所有其他使用者的許可權。讀取許可權設成4，寫入許可權設成2，而執行檔案許可權設成1。檔案的擁有者一般的執行檔案許可權是 6 或 7。

`gzip [參數] 檔案`

此程式會使用複雜算術演算法來壓縮檔案內容。用此方法壓縮的檔案，其副檔名是 `.gz`，而且使用前必須解壓縮。若要壓縮多個檔案或整個目錄，請使用 `tar` 指令。

-d
: 解壓縮包裝的 gzip 檔案，如此它們會恢復成原始大小而且可以正常處理 (和指令 `gunzip` 類似)。

`tar選項歸檔檔案`

`tar` 將一或多個檔案放到歸檔。壓縮是選擇性的，`tar` 是相當複雜的指令，有多個選項可用。最常用的選項是：

`-f`
: 通常用來將輸出寫入檔案而不是螢幕

`-c`
: 建立新的 tar 歸檔

`-r`
: 新增檔案至現有的歸檔

`-t`
: 輸出歸檔的內容

`-u`
: 但是只有在檔案比已包含在歸檔中的檔案更新時，才新增檔案

`-x`
: 解開歸檔的檔案 (*擷取*)

`-z`
: 使用 `gzip` 包裝產生的歸檔

`-j`
: 使用 `bzip2` 壓縮產生的歸檔

`-v`
: 列示處理的檔案

由 tar 建立，且副檔名是 .tar 的歸檔檔案。如果 tar 封存也使用 gzip 壓縮，則副檔名是 .tgz 或 .tar.gz。如果它使用 bzip2 壓縮，則副檔名是 .tar.bz2。

locate 樣式
: 這個指令只有在您已經安裝 findutils-locate 套件時才可使用。locate 指令可以尋找指定檔案的所在目錄。若有需要，請使用萬用字元來指定檔案名稱。程式執行速度非常快，因為它使用針對這個目的而建立的資料庫 (不必搜尋整個檔案系統)。然而此現象也會導致一項重大缺點：locate 無法找到資料庫最後更新之後所建立的所有檔案。資料庫可以由 root 使用 updatedb 來產生。

updatedb [選項]
: 此指令會更新 locate 所使用的資料庫。要在現有目錄包括檔案，請以 root 身份執行程式。加上 &，將它放到背景也是好的方法，如此您可以立即繼續在同一個指令行工作 (updatedb &)。這個指令通常當成每日 cron 工作 (請參閱 cron.daily) 來執行。

find [選項]
: 使用 find，搜尋指定目錄中的檔案。第一個引數會指定從什麼目錄開始搜尋。選項 -name 的後面必須有一個搜尋字串，搜尋字串也可以包含萬用字元。不同於使用資料庫進行搜尋的 locate，find 會掃描實際目錄。

用於存取檔案內容的指令

file[選項][檔案]
: 使用 file 可以偵測指定檔案的內容。

  -z
  : 可用來嘗試檢視壓縮檔案中的內容

cat [選項] 檔案
: cat 指令可在不中斷的前提下顯示檔案的內容、列印整個內容至螢幕。

  -n
  : 在輸出的左邊界加上編號

`less [選項] 檔案`

此指令可以用來瀏覽指定檔案的內容。使用 PgUp 和 PgDn 向上或向下捲動螢幕的一半頁面，或者使用空格鍵捲動整個螢幕頁面。使用 Home 和 End 移至檔案的開頭或結尾。按 Q 結束程式。

`grep [選項] 搜尋 檔案`

grep 指令可以在指定檔案中搜尋特定的搜尋字串。如果找到搜尋字串，指令會顯示所找到出現搜尋字串的一行文字以及檔案名稱。

`-i`

忽略大小寫

`-H`

只顯示相關檔案的名稱，不是文字行

`-n`

另外顯示發現符合資料的行數

`-l`

只列示不包含搜尋字串的檔案

`diff [選項] file1 file2`

`diff` 指令會比較任何兩個檔案的內容。程式產生的輸出，會列示不相符的所有行。只需要傳送程式變動的地方，而不是整個原始程式碼的程式設計人員經常使用此指令。

`-q`

只報告兩個檔案是否不同

`-u`

產生「制式」差異，使輸出更方便閱讀

檔案系統

`mount [選項] [設備] 裝載點`

此指令可以用來裝載任何資料媒體，例如硬碟、CD-ROM 光碟機以及其他磁碟機至 Linux 檔案系統的目錄。

`-r`
唯讀裝載

`-t 檔案系統`
指定檔案系統，通常 `ext2` 代表 Linux 硬碟、`msdos` 代表 MS-DOS 媒體、`vfat` 代表 Windows 檔案系統、`iso9660` 代表 CD

至於未定義在檔案 `/etc/fstab` 中的硬碟，也必須指定設備類型。在此狀況下，只有 `root` 可以裝載它。如果要讓其他使用者也能裝載該檔案系統，請在 `/etc/fstab` 的適當行，輸入選項 `user` (用逗點分開) 並儲存所做的變更。如需更多詳細資訊，請參閱 `mount(1)` man 頁面。

`umount [選項] 裝載點`
此指令會檔案系統中的已裝載磁碟機解除裝載。要防止資料遺失，從磁碟機取出抽取式資料媒體前，執行此指令。一般情況下，只有 `root` 可以執行 `mount` 與 `umount` 這兩個指令。若要讓其他使用者執行這些指令，請編輯 `/etc/fstab` 檔案，為相關的磁碟機指定選項 `user`。

18.3.2 系統指令

下節列出一些擷取系統資訊以及控制程序和網路時最重要的指令。

系統資訊

`df [選項] [目錄]`
`df` (磁碟可用空間) 指令，在未加上任何選項時，會顯示所有磁碟空間的資訊、目前使用的磁碟空間以及所有裝載磁碟機的可用空間。如果指定目錄，只會顯示該目錄所在磁碟機的資訊。

`-h`
顯示佔用的區塊數目 (GB、MB 或 KB)—以人類可判讀的格式。

`-T`
檔案系統的類型 (ext2、nfs 等等)

`du [選項] [路徑]`
在執行時未加上任何參數，此指令會顯示目前目錄的檔案和子目錄所佔用的總磁碟空間。

`-a`
顯示每一個別檔案的大小

`-h`
人類可判讀之格式的輸出

`-s`
只顯示計算的總共大小

`free [選項]`
指令 `free` 會顯示關於 RAM 以及交換空間使用的相關資訊，顯示二者類別的總計以及使用總數。如需相關資訊，請參閱第 22.1.6 節「free 指令」[394頁]。

`-b`
以位元組為單位的輸出

`-k`
以 KB 為單位的輸出

`-m`
以 MB 為單位的輸出

`date [選項]`
此簡單程式會顯示目前的系統時間。如果以 `root` 的身份執行，它也可以用來變更系統時間。如需有關此程式的詳細資料，請參閱 date(1) man 頁面。

程序

`top [選項]`
`top` 提供目前執行程序的快速綜覽。按 H 存取頁面，簡略說明主要選項來自定程式。

`ps [選項] [程序 ID]`
如果執行時不加任何選項，此指令會顯示由您啟動的程式或程序的相關表格。此指令的選項前面不會加上連字號。

aux
顯示所有程序的詳細清單，與擁有者無關

`kill [選項] 程序 ID`
不幸地，有時候程式無法以正常方式終止。大部份情況下，您仍然可以指定相關的程序 ID 來執行 `kill` 指令，以便停止執行中的程式 (請參閱`top` 和 `ps`)。`kill` 會傳送 *TERM* 訊號，指示程式關閉自己。如果此指令沒有作用，可以使用以下參數：

-9
傳送 *KILL* 訊號而非 *TERM* 訊號，這時多半可結束所指定的程序。

`killall [選項] 程序名稱`
此指令類似 `kill`，但是使用程序名稱 (而非程序 ID) 做為引數，刪除所有該名稱的處理程序。

網路

`ping [選項] 主機名稱或 IP 位址`
`ping` 指令是標準工具，用於測試 TCP/IP 網路的基本功能。它會傳送小的資料封包至目的地主機，要求立即回應。如果有作用，`ping` 會顯示訊息，指示網路連結基本上是正常的。

`-c` *數字*
決定要傳送的封包總數，並在分派後結束 (依預設，沒有限制設定)

`-f`
flood ping：儘可能傳送很多資料封包，一種常用方法，保留給 `root` 來測試網路

`-i` *值*
指定兩個資料封包之間的時間間隔 (秒) (預設值：1 秒鐘)

`nslookup`
網域名稱系統會將網域名稱解析成 IP 位址。使用此工具，傳送查詢至名稱伺服器 (DNS 伺服器)。

`telnet [選項] 主機名稱或 IP 位址 [埠]`
Telnet 實際上是一種網際網路通訊協定，供您透過網路在遠端主機上工作。Telnet 也是使用這種通訊協定的 Linux 程式名稱，用來在遠端電腦進行作業。

警告

不要在第三方可以「監聽的網路上使用 telnet。」特別是在網際網路上，使用加密傳送的方法，例如 `ssh`，防止密碼被惡意使用的危險 (請參閱 `ssh` 的手冊頁)。

其他

`passwd [選項] [使用者名稱]`
使用者可以使用此指令，在任何時候變更他們自己的密碼。管理員 `root` 可以使用此指令，變更系統上任何使用者的密碼。

`su [選項] [使用者名稱]`
`su` 指令可以從執行中的工作階段改用不同的使用者名稱登入。指定使用者名稱和相關密碼。將不會要求 `root` 的密碼，因為 `root` 已授權為可以使用任何使用者的身份。使用此指令時如果未指定使用者名稱，系統將提示您輸入 `root` 密碼並變更為超級使用者 (`root`)。

`-`
使用 `su -`，為不同使用者啟動登入外圍程序

`halt [選項]`
若要避免遺失資料，應該永遠使用這個程式關閉系統。

`reboot [選項]`
功能相同於 `halt`，只是系統會立即重新啟動。

`clear`
此指令會清除主控台的可見區域。它沒有任何選項。

18.3.3 如需更多資訊

本章節列示很多指令。如需關於其他指令或更詳細的資訊，建議參閱 O'Reilly 出版的《*Linux in a Nutshell*》。

18.4 vi 編輯器

許多的系統管理任務及程式設計仍然會用到文字編輯器。在 Unix 的世界中，vi 作為編輯器的表現十分突出，因為它提供了便利的編輯功能，而且支援滑鼠因而較其他編輯器更符合人體工學。

18.4.1 運作模式

注：按鍵的顯示

以下列出一些在 vi 中可以用按鍵輸入的指令。這些按鍵會依照鍵盤上的方式，以大寫字母顯示。如果您必須輸入大寫按鍵，便會以包含 Shift 鍵的按鍵組合方式明確顯示。

基本上，vi 利用三個作業模式：*插入*模式、*指*令模式，與*延伸*模式。按鍵會因為模式而有不同的功能。啟動時，vi 通常設定為「*指*令」模式。第一件要學習的事就是如何切換這些模式：

指令模式切換到插入模式
: 方法有很多種，包括輸入 A 為附加、I 為插入或 O 為在目前的行中插入新行。

插入模式切換到指令模式
: 按 Esc 鍵可離開「*插入*」模式。您無法在「*插入*」模式中終止 vi，因此習慣按 Esc 鍵是很重要的。

指令模式切換到延伸模式
: vi 的「*延伸*」模式可以藉由輸入冒號 (:) 來啟用。*延伸*或 *ex* 模式類似獨立的命令行編輯器，可用來處理各種簡單與更複雜的任務。

延伸模式切換到指令模式
: 在*延伸*模式中執行完指令後，編輯器會自動回到*指*令模式。如果您決定不要執行「*延伸*」模式中的任何指令，請用 <— 鍵刪除冒號。編輯器便會回到*指*令模式。

您無法直接從*插入*模式切換到*延伸*模式，而不先切換到*指*令模式。

vi 像其他的編輯器一樣都有自己的終止程式的程序。您無法在「*插入*」模式中終止 vi。首先，按 Esc 鍵離開*插入*模式。然後，您會有兩種選項：

1. *不儲存離開：*若不想除存變更而響終止編輯器，請按：– Q – ! 於*指*令模式中。驚歎號 (!) 會讓 vi 忽略任何變更。

2. *儲存與終止：*有許多方法可以儲存您的變更並終止編輯器。在*指*令模式中，使用Shift + Z Shift + Z。若要使用*延伸*模式離開程式並儲存所有的變更，請輸入：– W – Q。在「*延伸*」模式中，w 代表寫入，而 q 代表結束。

18.4.2 使用 vi

vi 可以用來做為一般編輯器。在*插入*模式中，輸入文字，然後用 <— 鍵與 Del 鍵刪除文字。使用方向鍵來移動游標。

不過，這些控制鍵常會造成問題，因為有很多種終端機使用特殊鍵碼。這就是*指*令模式派上用場之處。按 Esc 鍵從*插入*模式切換到*指*令模式。在*指*令模式中，請用 H、J、K 及 L 等鍵來移動游標。這些鍵有下列功能：

H
: 向左移動一個字元

J
: 向下移動一行

K
: 向上移動一行

L
: 向右移動一個字元

*指*令模式中的指令都允許不同的變化。若要執行一個指令數次，只要在輸入實際的指令之前，輸入要重複的次數即可。例如，輸入 5L 來將游標向左移動五個字元。

表格 18.2「vi 編輯器的簡單指令」[346頁]顯示部份重要指令。此清單並不完整。第 18.4.3 節「如需更多資訊」[347頁]中的文件有更完整的清單

表格 18.2 *vi 編輯器的簡單指令*

Esc	變更至指令模式
I	變更至插入模式 (字元顯示在目前的游標位置)
A	變更至插入模式 (字元會插入目前的游標位置之後)
Shift + A	變更至插入模式 (字元會新增至行尾)
Shift + R	變更至取代模式 (覆寫舊文字)
R	取代游標下的字元
O	變更至插入模式 (新行會插入目前行的後面)
Shift + O	變更至插入模式 (新行會插入目前行的前面)
X	刪除目前的字元
D – D	刪除目前的行
D – W	刪除至目前文字的最後
C – W	變更至插入模式(目前文字的其他部份會被您輸入的下一個資料覆寫)
U	復原上次指令
Ctrl + R	重做已復原的變更
Shift + J	將以下行與目前行結合
.	重複上次的指令

18.4.3 如需更多資訊

vi 支援許多指令。它讓您可以使用巨集、捷徑、具名緩衝區及許多其他有用的功能。不同選項的詳細說明已超出本手冊的範圍。SUSE Linux Enterprise 會隨附 vim (加強的 vi)，也就是 vi 的加強版。有數個此應用程式的資訊來源：

- vimtutor 是 vim 的互動式教學課程。
- 在 vim 中，請輸入 `:help` 指令以取得很多的說明主題。
- 關於 vim 的書籍可以在線上取得，網址為 http://www.truth.sk/vim/vimbook-OPL.pdf。
- vim計劃的網頁位於http://www.vim.org，網站中會有所有的新聞、郵件清單及其他的文件。
- 網際網路也有不少的 Vim 來源：http://www.selflinux.org/selflinux/html/vim.html、http://www.linuxgazette.com/node/view/9039與http://www.apmaths.uwo.ca/~xli/vim/vim_tutorial.html。請參閱http://linux-universe.com/HOWTO/Vim-HOWTO/vim-tutorial.html取得更多有關教學課程的連結。

重要：VIM 授權

vim 是「慈善軟體」，意即作者不索取任何的軟體費用，但鼓勵您以捐款贊助，來支持非營利的專案計畫。此計劃懇求您幫助烏干達的貧困孩童。更多的資訊可在線上取得，網址為http://iccf-holland.org/index.html、http://www.vim.org/iccf/ 及 http://www.iccf.nl/。

III. 系統

64 位元系統環境的 32 位元和 64 位元應用程式

19

SUSE Linux Enterprise® 可用於多種 64 位元平台。但這並不表示所有包含的應用程式都已移植到 64 位元平台。SUSE Linux Enterprise 支援在 64 位元系統環境中使用 32 位元應用程式。本章簡略說明這項支援在 64 位元 SUSE Linux Enterprise 平台的執行方式。它說明 32 位元應用程式的執行方式 (執行期間支援) 以及如何編譯 32 位元應用程式，讓它們在 32 位元和 64 位元兩種系統環境都可執行。另外，您可找到關於核心 API 和 32 位元應用程式如何在 64 位元核心下執行的相關資訊。

注：IBM System z 上的 31 位元應用程式：

IBM System z 上的 s390 使用 31 位元環境。對以下 32 位元應用程式的參考也同樣適用於 31 位元應用程式。

針對 64 位元平台 ia 64、ppc64、s390x 和 x86__64 所設計的 SUSE Linux Enterprise，讓現有的 32 位元應用程式「不需額外的設定」即可在 64 位元環境中執行。對應的 32 位元平台為 x86 (對應 ia64)、ppc (對應 ppc64)、s390 (對應 s390x) 以及 x86 (對應 x86_64)。這項支援意謂您可以繼續使用偏好的 32 位元應用程式，無需等到對應的 64 位元連接埠上市。目前的 ppc64 系統是以 32 位元模式執行大部分應用程式，不過您可以執行 64 位元應用程式。

19.1 執行期間支援

重要：不同應用程式版本之間的衝突

如果應用程式有 32 位元和 64 位元兩種版本，同時安裝二種版本，一定會發生問題。在這種狀況下，可在兩種版本中選定一種來安裝並使用。

要正確執行，每一個應用程式都需要一些程式庫。不幸的是，這些程式庫的 32 位元和 64 位元版本，名稱都一樣。它們必須透過其他方法來區分彼此。

64 位元平台 ppc64、s390x 和 86_64 所使用的方法相同：為保留與 32 位元版本的相容性，系統中儲存程式庫的位置與 32 位元環境下的系統相同。在 32 位元和 64 位元環境中，`libc.so.6` 的 32 位元版本都位於 `/lib/libc.so.6`。

所有 64 位元程式庫和物件檔案都位於名為 `lib64` 的目錄。您通常預期在 `/lib`、`/usr/lib` 和 `/usr/X11R6/lib` 之下找到的 64 位元物件檔案，現在放在 `/lib64`、`/usr/lib64` 以及 `/usr/X11R6/lib64` 底下。這表示在 `/lib`、`/usr/lib` 和 `/usr/X11R6/lib` 之下，有預留空間給 32 位元程式庫使用，因而兩種版本的檔案名稱能夠保持不變。

32 位元 `/lib` 目錄的子目錄，如果其資料內容不取決於字組大小，也不會移動。例如，X11 字型仍然可以在 `/usr/X11R6/lib/X11/fonts` 底下的一般位置找到。此配置與 LSB (Linux Standards Base) 以及 FHS (File System Hierarchy Standard) 相容。

► **ipf:** ia 64 的 64 位元程式庫位於標準 `lib` 目錄中。在這種狀況下，既沒有 `lib64` 目錄，也沒有 `lib32` 目錄。ia64 會模擬執行 32 位元 x86 程式碼。一組基本程式庫會安裝在 `/emul/ia32-linux/lib` 和 `/emul/ia32-linux/usr/X11R6/lib` 中。 ◄

19.2 軟體開發

所有 64 位元架構都支援 64 位元物件的開發。對 32 位元編譯的支援層級需視結構而定。以下為 GCC (GNU Compiler Collection) 和 Binutil 的工具鏈的各種執行實作選項，包括組合器 `as` 以及連結器 `ld`：

Biarch 編譯器
: Biarch 開發工具鏈可以產生 32 位元和 64 位元二種物件。幾乎所有平台都預設支援 64 位元物件的編譯。如果使用特殊旗標，就可以產生 32 位元物件。此特殊旗標是適用於 GCC 的 `-m32` (`-m31` 用於產生 s390 二進位檔案)。Binutil 的旗標視結構而定，但是 GCC 會將正確的旗標傳送至連結器和組合器。Biarch 開發工具鏈目前可用於 amd64 (支援 x86 和 amd64 說明的開發)、s390x 及 ppc64。32 位元物件一般是建立在 ppc64 平台上。要產生 64 位元物件，必須使用 `-m64` 旗標。

不支援
: 在所有平台上，SUSE Linux Enterprise 都不支援直接開發 32 位元軟體。若要在 ia64 底下開發 x86 應用程式，請使用 SUSE Linux Enterprise 對應的 32 位元版本。

所有標頭檔案必須使用與結構無關的形式來編寫。安裝的 32 位元和 64 位元程式庫，必須有一個與已安裝標頭檔案相符的 API (應用程式設計介面)。標準 SUSE Linux Enterprise 環境是根據此原則所設計。對於手動更新的程式庫，請自行解決這些問題。

19.3 Biarch 平台的軟體編譯

若要在 biarch 結構上，為其他結構開發二進位程式，必須為第二個結構額外安裝相關程式庫。如果第二個結構是 32 位元結構，這些套件稱為 `rpmname-32bit` 或 `rpmname-x86` (針對 ia64)；如果第二個結構是 64 位元結構，則套件稱為 `rpmname-64bit`。您還需要 `rpmname-devel` 套件的各個標頭和程式庫，以及 `rpmname-devel-32bit` 或 `rpmname-devel-64bit` 之第二個結構的開發程式庫。

例如，若要編譯一個使用 `libaio` 的程式，而所在系統的第二個結構是 32 位元結構 (x86_64 或 s390x)，您需要以下 RPM：

libaio-32bit
: 32 位元執行期間套件

libaio-devel-32bit
: 32 位元開發的標題和程式庫

libaio
: 64 位元執行期間套件

libaio-devel
    64 位元開發標題和程式庫

大部份開放原始碼程式使用的程式組態是以 `autoconf` 為基礎。若要使用 `autoconf` 為第二個結構設定程式，請覆寫 `autoconf` 的一般編譯器和連結器設定，方法是執行包含其他環境變數的 `configure` 程序檔。

以下範例參考第二個結構為 x86 的 x86_64 系統：以 s390 為第二個結構的 s390x，或以 ppc 為第二個結構的 ppc64，其範例都與此類似。此範例不適用於未建立 32 位元套件的 ia64。

提示

採用 s390 做為第二個結構時，必須使用 `-m31`，而不能使用 `-m32`，因為這是一個 31 位元系統。

---

1. 使用 32 位元編譯器：

   ```
   CC="gcc -m32"
   ```

2. 指示連結器處理 32 位元物件 (務必使用 gcc 做為連結器前端工具)：

   ```
   LD="gcc -m32"
   ```

3. 設定組合器來產生 32 位元物件：

   ```
   AS="gcc -c -m32"
   ```

4. 判斷 `libtool` 等等的程式庫是否來自 `/usr/lib`：

   ```
   LDFLAGS="-L/usr/lib"
   ```

5. 決定程式庫是否儲存在 `lib` 子目錄中：

   ```
   --libdir=/usr/lib
   ```

6. 決定是否使用 32 位元 X 程式庫：

   ```
   --x-libraries=/usr/X11R6/lib/
   ```

並非每一個程式都需要所有這些變數。將它們配合各程式使用。

在 x86_64、ppc64 或 s390x 上編譯原生 32 位元應用程式的 `configure` 呼叫範例如下：

```
CC="gcc -m32"            \
LDFLAGS="-L/usr/lib;"  \
        .configure       \
          --prefix=/usr  \
          --libdir=/usr/lib
make
make install
```

19.4 核心規格

x86_64、ppc64 和 s390x 適用的 64 位元核心，可提供 64 位元和 32 位元兩種核心 ABI (應用程式二進位介面)。後者與相對應 32 位元核心的 ABI 是相同的。這表示 32 位元應用程式可以用與 32 位元核心溝通相同的方式，來與 64 位元核心溝通。

32 位元系統模擬的 64 位元核心呼叫，不支援系統程式使用的所有 API。這要視平台而定。基於這個原因，少數應用程式 (例如 `lspci`) 必須編譯為 64 位元程式，才能在非 ppc64 平台上正確運作。在 IBM System z 上，並非所有 ioctl 都可用於 32 位元核心 ABI。

64 位元核心只可以載入為此核心特別編譯的 64 位元核心模組。它無法使用 32 位元核心模組。

提示

部份應用程式需要個別的核心可載入式模組。如果您想在 64 位元系統環境使用這種 32 位元應用程式，請洽詢此應用程式的提供者以及 Novell，確定是否可以取得此模組的核心可載入式模組的 64 位元版本以及核心 API 的 32 位元編譯版本。

20 啟動及設定 Linux 系統

啟動 Linux 系統需要各種不同的元件。硬體自身是由 BIOS 啟動的。BIOS 會藉由開機載入程式啟動核心。此後，關於 init 和 runlevel 的開機程序完全由作業系統控制。憑藉 runlevel 的概念，您可以保持日常使用的設定，以及對系統執行維護任務。

20.1 Linux 開機程序

Linux 開機程序由數個階段所組成，每個階段分別由元件所代表。以下清單簡短概述開機程序，以及所有相關主要元件的功能。

1. **BIOS**　啟動電腦之後，BIOS 會啟始螢幕和鍵盤並測試主記憶體。在此階段中，機器不會存取大量儲存媒體。接著，會從 CMOS 值載入目前日期、時間和最重要的周邊。識別第一個硬碟及其規格之後，系統控制將會從 BIOS 轉到開機載入程式。如果 BIOS 支援網路開機，則也可以設定提供開機載入程式的開機伺服器。在 x86 系統上，需要 PXE 開機。其他架構通常使用 BOOTP 通訊協定來取得開機載入程式。

2. **開機載入程式**　第一顆硬碟的第一個實體 512 位元組資料磁區，會載入主要記憶體。接著，在此磁區開頭的*開機載入程式*會接管開機程序。開機載入程式執行的指令，決定其他部份的開機程序。因此，第一個硬碟的前 512 位元組是*主開機紀錄* (MBR)。然後，開機載入程式會將控制傳送到實際作業系統，在這裡是指 Linux 核心。如需 GRUB (即 Linux 開機載入程式) 的詳細資訊，請參閱 第 21 章「*開機載入程式*」[371 頁]。進行網路開機時，BIOS 會充當開機載入程式。它會從開機伺服器取得要啟動的影像，然後啟動系統。這與本地硬碟完全無關。

3. **核心和初始 RAM 檔案系統**　為了送出系統控制，開機載入程式會將核心和初始 RAM 式檔案系統 (initramfs) 兩者都載入記憶體。核心可以直接使用 initramfs 的內容。initramfs 包含一個名為 init 的小執行檔，它可以處理實體根目錄檔案系統的裝載。若存取大量儲存之前需要什麼特殊硬體驅動程式的話，那一定就是 initramfs 了。如需 initramfs 的詳細資訊，請參閱 第 20.1.1 節「initramfs」[358頁]。如果系統沒有本機硬碟，則 initramfs 必須提供根檔案系統給核心。這可以藉由 iSCSI 或 SAN 這類網路區塊設備來完成，但也可以使用 NFS 做為根設備。

4. **initramfs 上的 init**　本程式將執行裝載適當根目錄檔案系統所需的全部動作，例如提供必要檔案的系統核心功能，並且提供包含 udev 之大量儲存控制器的設備驅動程式。找到根目錄檔案系統後，將會檢查是否有錯誤並進行裝載。若裝載成功，將會清除 initramfs 並執行根目錄檔案系統上的 init 程式。如需有關 init 的詳細資訊，請參閱第 20.1.2 節「initramfs 上的 init」[359頁]。如需更多有關 udev 的詳細資訊，請參閱第 24 章「*使用 udev 進行動態核心設備管理*」[423頁]。

5. **init**　init 會透過提供數個不同層級所提供的不同功能來負責系統實際開機作業。第 20.2 節「init 程序」[360頁] 中會描述 init。

20.1.1 initramfs

initramfs 是一個小型 cpio 歸檔，其可由核心載入 RAM 磁碟。在實際根目錄檔案系統裝載之前，它提供可讓程式執行的最小 Linux 環境。BIOS 常式會將這個最小的 Linux 環境載入記憶體，且除了記憶體要求外沒有其他特定硬體需求。此外，initramfs 還必須提供一個名為 init 的執行檔，這個執行檔會在處理開機程序的根目錄檔案系統上執行實際的 init 程式。

在根目錄檔案系統能夠裝載以及作業系統可以啟動之前，核心需要相應的驅動程式來存取根目錄檔案系統所在的設備。這些驅動程式可能包含特定類型硬碟的特殊驅動程式，或者甚至包含存取網路檔案系統的網路驅動程式。initramfs 上的 init 還會載入根目錄檔案系統所需的模組。當模組載入完成之後，udev 便會為 initramfs 提供所需的設備。在後來的開機程序中，變更根目錄檔案系統後，必須重新產生這些設備。結合使用 `boot.udev` 和指令 `udevtrigger` 可以實現此目的。

如果您要在已安裝的系統上變更硬體 (如硬碟)，且此硬體要求開機時核心中必須存在不同的驅動程式，則您必須更新 `initramfs` 檔案。更新 initramfs 與更新其前身 initrd 是採用一樣的更新方式，即是呼叫 `mkinitrd`。不使用任何引數

來呼叫 mkinitrd 將會建立一個 initramfs。呼叫 mkinitrd -R 則會建立一個 initrd。在 SUSE Linux Enterprise® 中，/etc/sysconfig/kernel 內的變數 INITRD_MODULES 會指定要載入的模組。安裝之後，此變數會自動設定為正確值。模組會確實依據它們在 INITRD_MODULES 中出現的順序來載入。如果您不依賴於 /dev/sd? 設備檔案的正確設定，這一點將無關緊要。不過，在目前系統中，您也可以使用 /dev/disk/ 下以幾個不同子目錄分類的設備檔案，這些子目錄的名稱為 by-id、by-path 和 by-uuid，這些目錄總是代表相同磁碟。在安裝時也可以透過指定相應的裝載選項來實現此目的。

重要：更新 initramfs 或 initrd

開機載入程式將採用與核心相同的方式載入 initramfs 或 initrd。更新 initramfs 或 initrd 之後不需要重新安裝 GRUB，因為開機時 GRUB 會在目錄搜尋正確的檔案。

20.1.2 initramfs 上的 init

Initramfs 上的 init 其主要目的是準備裝載實際根目錄檔案系統，以及存取實際根目錄檔案系統。根據您的系統組態，init 負責下列任務。

載入核心模組
: 根據硬體組態，存取您電腦的硬體元件可能需要特殊的驅動程式 (特別是您的硬碟)。若要存取根目錄檔案系統，核心需載入適當的檔案系統驅動程式。

提供區塊特殊檔案
: 對於每個載入的模組，核心均會產生設備事件。udev 會處理這些事件，並在 RAM 檔案系統的 /dev 中產生所需的區塊專用檔案。如果沒有這些專用檔案，便無法存取檔案系統和其他設備。

管理 RAID 和 LVM 設定
: 如果您將系統設定為 RAID 或 LVM 之下的根目錄檔案系統，init 會設定 LVM 或 RAID，以便之後能夠存取檔案根目錄系統。如需有關 RAID 的資訊，請參閱第 7.2 節「軟體 RAID 組態」 [111頁]。如需更多關於 LVM 的詳細資訊，請參閱第 7.1 節「LVM 組態」 [103頁]。如需更多 EVMS 與特殊儲存設定的相關資訊，請參閱《*儲存管理指南*》。

管理網路組態
: 如果您將系統設為使用網路裝載的根目錄檔案系統 (透過 NFS 裝載)，那麼 init 必須確認是否已載入適當的網路驅動程式，還有是否設定為允許存取根目錄檔案系統。

  如果檔案系統位在 iSCSI 或 SAN 一類網路區塊設備上，initramfs 也會設定與儲存伺服器的連線。

安裝程序中，當 init 在初始開機時被呼叫，它的任務會與之前討論的不同：

尋找安裝媒體
: 啟動安裝程序時，您的機器會從安裝媒體使用 YaST 安裝程式，來載入一個安裝核心，以及一個特殊 initrd。在 RAM 檔案系統中執行的 YaST 安裝程式，必需具備安裝媒體的實際位置相關資訊，以便存取該程式和安裝作業系統。

啟始硬體辨識並載入適當核心模組
: 如同在第 20.1.1 節「initramfs」[358頁]所述，開機程序會以最少驅動程式啟動，供大部份的硬體組態使用。init 將啟動一個硬體掃描程序，該程序會判斷驅動程式是否適用您的硬體組態。開機程序所需的模組名稱會寫入 `/etc/sysconfig/kernel` 的 `INITRD_MODULES` 中。這些名稱是用來產生系統開機所需的自定 initramfs。如果模組不是開機所需，而是供 coldplug 使用，則模組會寫入 `/etc/sysconfig/hardware/hwconfig-*`。用此目錄中的組態檔案描述的所有設備都會在開機程序中啟始化。

載入安裝系統或救援系統
: 在硬體妥善識別、適當驅動程式完成，以及 udev 已建立設備特殊檔案之後，init 就會啟動安裝系統，其中包含實際 YaST 安裝程式或救援系統。

啟動 YaST
: 最後，init 將會啟動 YaST，而 YaST 會啟動套件安裝和系統組態。

20.2 init 程序

init 程式是程序 ID 1 的程序，其負責以指定方式來啟始系統。init 由核心直接啟動，並拒絕訊號 9，這個訊號通常會刪除程序。所有其他程式是直接透過 init 或它其中一個子程序啟動。

init 主要是在 /etc/inittab 檔案中設定，*runlevels* 即是在該檔案中定義的 (請參閱第 20.2.1 節「Runlevel」[361頁])。這個檔案還會指定每一個層次可以使用的服務和精靈。視 /etc/inittab 中的項目而定，init 會執行數個程序檔。為避免混淆，這些稱做 *init 程序檔* 的程序檔都位於目錄 /etc/init.d 中 (請參閱第 20.2.2 節「Init 程序檔」[363頁])。

系統啟動和關閉的程序，是由 init 維護。依此觀點，核心可以視為背景程序，它的任務是維護所有其他程序，並根據其他程式的要求來調整 CPU 時間和硬體存取。

20.2.1 Runlevel

在 Linux 中是由 *runlevel* 定義啟動系統的方式，以及在所執行的系統可以使用哪些服務。開機之後，系統會按照 /etc/inittab 中的 initdefault 這一行文字的定義而啟動。一般是 3 或 5。請參閱表格 20.1「可用的 Runlevel」[361頁]。還有一種方法是，runlevel 可以在開機期間指定 (例如，在開機提示時新增 runlevel 編號)。所有不經過核心自身直接評估的參數，都會傳遞至 init。要開機進入 runlevel 3，只需在開機提示處新增數字 3。

表格 20.1 *可用的 Runlevel*

Runlevel	描述
0	系統暫停
S or 1	單一使用者模式
2	本地多重使用者模式，不包含遠端網路 (NFS 等)
3	完整的多重使用者模式，包含網路
4	未使用
5	完整多重使用者模式，包含網路、X顯示管理員—KDM、GDM 或 XDM
6	系統重新開機

重要：避免透過 NFS 裝載分割區的 Runlevel 2

如果系統會透過 NFS 裝載 `/usr` 一類分割區，您就不應該使用 runlevel 2。如果程式檔案或是程式庫因 runlevel 2 (本地多重使用者模式，無遠端網路) 無法提供 NFS 服務而發生遺失，系統可能無法正常運作。

若要在系統執行時變更 runlevel，請輸入 `telinit` 以及當成引數的對應數字。只有系統管理員可以執行此動作。以下清單列出 runlevel 區域中最重要的指令摘要。

`telinit 1` 或 `shutdown now`
: 系統變更為*單一使用者模式*。此模式是用於系統維護和管理任務。

`telinit 3`
: 可以啟動所有主要的程式和服務 (包括網路)，且可讓一般使用者登入並在非圖形環境下使用該系統。

`telinit 5`
: 啟用圖形式環境。通常這時會啟動顯示管理員，如 XDM、GDM 或 KDM。如果已啟用自動登入，本地使用者就可以登入事先選定的視窗管理員 (GNOME、KDE 或任何其他視窗管理員)。

`telinit 0` 或 `shutdown -h now`
: 暫停系統。

`telinit 6` 或 `shutdown -r now`
: 暫停系統後重新開機

所有 SUSE Linux Enterprise 標準安裝中，預設都使用 runlevel `5`。使用者會在提示之下使用圖形介面登入或預設的使用者會自動登入。如果預設 runlevel 是 `3`，這時 X Window System 必須依據第 26 章「*X Window System*」 [441 頁]所述適當設定，才能將 runlevel 切換至 `5`。完成這個動作之後，請輸入 `telinit 5` 來檢查系統是否已依指定方式運作。如果一切都如預期，您可以使用 YaST，將預設 runlevel 設定成 `5`。

警告：/etc/inittab 中有錯誤可能會造成系統開機發生錯誤

如果 /etc/inittab 損毀，系統可能無法正常開機。因此，在編輯 /etc/inittab 時要格外小心。將機器重新開機前，一定要用 telinit q 指令讓 init 重新讀取 /etc/inittab。

通常，當您變更 runlevel 時會發生兩件事。首先，啟動目前 runlevel 中的停止程序檔，關閉對目前 runlevel 很重要的一些程式。然後啟動新 runlevel 的啟動程序檔。在大部份的情況下，此時也會啟動一些程式。例如，從 runlevel 3 變更成 5 時，會發生以下事件：

1. 管理員 (root) 可以輸入 telinit 5，要求 init 變更為不同的 runlevel。
2. init 會檢查目前的 runlevel (runlevel) 並決定是否要以新 runlevel 作為啟動 /etc/init.d/rc 的參數。
3. 現在，如果新 runlevel 沒有啟動程序檔，rc 會呼叫目前 runlevel 的停止程序檔。在此範例中的所有程序檔，都位於 /etc/init.d/rc3.d(舊的 runlevel 是 3)，而且開頭是 K。K 後面的數字表示以 stop 參數執行程序檔的順序，因為還有其他的因素要考慮。
4. 新的 runlevel 啟動程序檔，會最後才啟動。在此範例中的所有程序檔，都位於 /etc/init.d/rc5.d，而且開頭是 S。同時，S 後面的數字決定要啟動程序檔的序列。

變更成與目前 runlevel 相同的 runlevel 時，init 只會檢查 /etc/inittab 是否變更，並啟動適當的步驟，例如，在另一個介面啟動 getty。使用指令 telinit q 也可以完成相同功能。

20.2.2 Init 程序檔

/etc/init.d 中的程序檔有兩種類型：

由 init 直接執行的程序檔
: 這種狀況僅出現於開機程序或立即關閉系統 (電源中斷或使用者按下 Ctrl + Alt + Del)時。對 IBM System z 系統而言，僅出現於開機程序期間或立即關閉系統時 (電源斷開或透過「訊號靜止」)。此程序檔定義於 /etc/inittab。

由 init 間接執行的程序檔
這些程序檔在變更 runlevel 時就會執行，而且永遠會呼叫主要程序檔 `/etc/init.d/rc`，以保證相關程序檔的順序正確。

所有程序檔都位於 `/etc/init.d`。在開機時執行的程序檔會透過 `/etc/init.d/boot.d` 的符號連結來呼叫。用於變更 runlevel 的程序也可以透過其中一個子目錄 (`/etc/init.d/rc0.d` 到 `/etc/init.d/rc6.d`) 的符號連結來呼叫。這樣的安排是為了明確執行，避免當程序檔用於多個 runlevel 時的重複執行。因為每一個程序檔都可當成啟動程序檔和停止程序檔來執行，所以這些程序檔必須了解參數 `start` 和 `stop`。程序檔也了解 `restart`、`reload`、`force-reload` 和 `status` 選項。這些不同選項在 表格 20.2「可能的 init 程序檔選項」[364頁] 都有說明。直接由 init 執行的程序檔沒有這些連結。他們可以根據需要從 runlevel 獨立執行。

表格 20.2 *可能的 init 程序檔選項*

選項	描述
`start`	啟動服務。
`stop`	停止服務。
`restart`	如果服務在執行中，先停止，再重新啟動。如果服務沒有執行，請啟動它。
`reload`	不需停止和重新啟動服務，就可以重新載入組態。
`force-reload`	如果服務支援，請重新載入組態。否則，執行與 `restart` 相同的動作。
`status`	顯示目前狀態。

每一個特定 runlevel 子目錄中的連結，可以將程序檔與不同 runlevel 產生關聯。安裝或解除安裝套件時，可以透過程式 insserv 的協助，新增和移除這些連結 (或透過 `/usr/lib/lsb/install_initd`，它是一個會呼叫該程式的程序檔)。如需詳細資訊，請參閱 insserv(8) 線上文件。

這些設定也都可以利用 YaST 模組來變更。如果您必須在指令行上檢查狀態，請使用 chkconfig(8) 線上文件中所說明的 chkconfig 工具。

簡介最先啟動的開機及最後啟動的停止程序檔，並說明維護程序檔。

boot
: 使用 init 直接啟動系統時會執行。它與所選的 runlevel 無關，而且只會執行一次。此時，會裝載 /proc 和 /dev/pts 檔案系統，啟用 blogd (開機記錄精靈)。如果系統是更新或安裝之後第一次開機，會啟動啟始系統組態。

  任何其他服務啟動之前，blogd 精靈是透過開機和 rc 啟動的服務。當這些程序檔（會執行多子程序檔，如使特殊檔案區塊能夠使用）啟動的動作完成之後就會停止。blogd 會將螢幕輸出的所有內容寫入記錄檔案 /var/log/boot.msg，但唯有當 /var 裝載為可讀寫，才會發生。否則，blogd 會緩衝處理所有螢幕資料，直到 /var 可以使用為止。您可以在 blogd(8) 線上文件取得進一步資訊。

  程序檔 boot 也負責啟動 /etc/init.d/boot.d 中，名稱開頭是 S 的所有程序檔。在該處，會檢查檔案系統，並在需要時設定迴圈設備。也會設定系統時間。如果自動檢查和修復檔案系統時發生錯誤，系統管理員輸入管理員密碼後即可介入。最後執行的是程序檔 boot.local。

boot.local
: 在此輸入開機時要執行的其他指令 (進入 runlevel 之前)。它就像是 DOS 系統的 AUTOEXEC.BAT。

boot.setup
: 從單一使用者模式變更成任何其他 runlevel 時會執行此程序檔，它負責一些基本設定，例如鍵盤配置和虛擬主控台的啟始化。

halt
: 只有進入 runlevel 0 或 6 時，才會執行此程序檔。在此，它是以 halt 或 reboot 的方式執行。系統是否關閉或重新啟動，取決於呼叫 halt 的方式。

rc
: 此程序檔會呼叫目前 runlevel 的適當停止程序檔，以及新選取 runlevel 的啟動程序檔。

您可以建立自己的程序檔，並輕鬆將它們整合至上述配置。如需關於格式化、命名以及組織自定程序檔的指示，請參閱 LSB 的規格和 init、init.d、

chkconfig 及 insserv 的線上文件。另請參閱startproc 和 killproc 的 man 頁面。

警告：錯誤的 init 程序檔可能會暫停系統

錯誤的 init 程序檔可能會讓您的機器暫停。編輯類似程序檔要格外小心，可能的話，讓它們在多重使用者環境下密集測試。如需關於 init 程序檔的一些有用資訊，請參閱第 20.2.1 節「Runlevel」 [361頁]。

若要為指定的程式或服務建立自定 init 程序檔，請將檔案 /etc/init.d/skeleton 當成樣板。使用新名稱儲存此檔案副本，編輯所需相關程式和檔案名稱、路徑，以及其他詳細資料。您也可能需要使用自己的組件來強化程式檔，好讓 init 程序觸發正確的動作。

上方的 INIT INFO 區塊是程序檔的必要組件，因此必須進行編輯。請參閱範例 20.1「迷你 INIT INFO 區塊」 [366頁]。

範例 20.1 *迷你 INIT INFO 區塊*

```
### BEGIN INIT INFO
# Provides:          FOO
# Required-Start:    $syslog $remote_fs
# Required-Stop:     $syslog $remote_fs
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Description:       Start FOO to allow XY and provide YZ
### END INIT INFO
```

請在 INFO 區塊的第一行中、Provides: 後面，為此 init 程序檔所控制的程式或服務指定名稱。Required-Start: 與 Required-Stop: 這兩行，會指定服務本身啟動或停止之前，必須先啟動或停止的所有服務。此資訊以後會用來產生程序檔名稱的編號，這些就是在 runlevel 目錄可以找到的編號。Default-Start: 和 Default-Stop: 之後，指定應該自動啟動或停止服務的 runlevel。最後，會在 Description: 簡要說明討論中的服務。

若要從 runlevel 目錄 (/etc/init.d/rc?.d/) 建立對應到 /etc/init.d/ 中的程序檔，請輸入 insserv *new-script-name*指令。insserv 程式會評估 INIT INFO 標題，為 runlevel 目錄 (/etc/init.d/rc?.d/) 中的啟動和停止程序檔建立必要的連結。該程式也會為這些連結的名稱加上必要的編號，即可依正確順序啟動和停止每一個 runlevel。如果您偏好以圖形工具來建立類似連結，請使用 YaST 提供的 runlevel 編輯器，請見第 20.2.3 節「使用 YaST 設定系統服務 (Runlevel)」 [367頁]。

如果 /etc/init.d/ 裡已經有程序檔，應該整合至現有的 runlevel 配置，使用 insserv 在 runlevel 目錄中立即建立連結，或者在 YaST 的 runlevel 編輯器中啟用對應的服務。下次重新啟動時，將會套用您所做的變更—新服務會自動啟動。

請勿手動設定這些連結。如果 INFO 區塊發生錯誤，稍後執行某些其他服務的 insserv 將會發生問題。為此程序檔手動新增的服務將會在下次執行 insserv 時予以移除。

20.2.3 使用 YaST 設定系統服務 (Runlevel)

使用「*YaST*」>「*系統系統服務 (Runlevel)*」>「*啟動 YaST*」模組後，會出現一個概觀清單，其中會列出所有可用的服務和每個服務目前的狀態 (停用或啟用)。決定要以「*簡單模式*」或「*進階模式*」使用模組。大部份情況下，預設的「*簡單模式*」應該都已夠用。左欄顯示服務的名稱，中間欄顯示它的目前狀態，而右欄提供簡短說明。視窗下方為選取的服務提供更詳細的說明。若要啟用服務，在表格中選取它，然後選取「*啟用*」。停用服務的步驟也一樣。

圖形 20.1 *系統服務 (Runlevel)*

要更仔細控制啟動或停止服務的 runlevel，或者變更預設的 runlevel，請先選取「*進階模式*」。目前預設的 Runlevel 或「initdefault」(系統開機時預設裝載的 Runlevel) 會出現在視窗上方。一般情況下，SUSE Linux Enterprise 系統的預設

runlevel 是 runlevel 5 (含網路和 X 的完整多重使用者模式)。合適的替代方法可能是 runlevel 3 (含網路的完整多重使用者模式)。

此 YaST 對話方塊允許其中一個 runlevel 選項 (如表格 20.1 「可用的 Runlevel」 [361頁]所列) 做為新預設。還可使用此視窗中的表格，啟用或停用個別服務和精靈。表格會列示可用的服務和精靈，顯示目前在您的系統上它們是否啟用，如果啟用，是哪一個 runlevel。使用滑鼠選取其中一列後，按一下代表 Runlevel 的核取方塊 (「*B*」、「*0*」、「*1*」、「*2*」、「*3*」、「*5*」、「*6*」和「*S*」) 來定義 Runlevel，以便在該處執行選取的服務或精靈。Runlevel 4 並未定義，以便建立自定 runlevel。表格概觀的正下方，提供目前所選服務或精靈的簡要說明。

使用「*啟動、停止或重新整理*」，決定是否啟動服務。「*重新整理狀態*」檢查目前狀態。您可以使用「*設定或重設*」來做選擇，將您的變更套用至系統，或復原啟動 runlevel 編輯器之前的設定。選取「*完成*」，就會將變更的設定儲存至磁碟。

警告：錯誤的 Runlevel 設定可能會造成系統損害

錯誤的 Runlevel 設定可能會造成系統無法使用。在您套用變更之前，請務必確定您知道它們的後果。

20.3 透過 /etc/sysconfig 設定系統

`/etc/sysconfig` 中的組態檔是控制 SUSE Linux Enterprise 主要組態的檔案。`/etc/sysconfig` 中個別的檔案只由相關的程式檔讀取。這可以確保例如網路設定，只由網路相關的程序檔來剖析。

有兩種方式可以編輯系統組態：使用 YaST Sysconfig 編輯器或手動編輯組態檔。

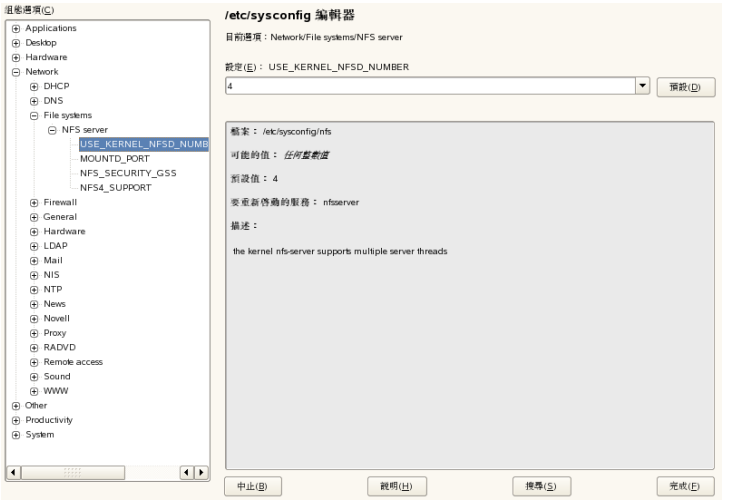
20.3.1 使用 YaST Sysconfig 編輯器變更系統組態

YaST Sysconfig 編輯器會提供一個易於使用的前端工具，方便您設定系統。假使您對要變更的組態變數實際位置不甚了解，您只要使用此模組內建的搜尋功能即可變更所需的組態變數值，接著 YaST 會套用這些變更、根據 `Sysconfig` 中設定的值更新組態，然後重新啟動服務。

警告：修改 /etc/sysconfig/* 檔案可能會損壞您的安裝檔案

如果您先前並無經驗和相關知識，請勿修改 /etc/sysconfig 檔案。它可能會嚴重破壞您的系統。/etc/sysconfig中的檔案包含每一個變數的簡短註解，說明它們的實際作用。

圖形 20.2　使用 sysconfig 編輯器的系統組態



YaST sysconfig 對話方塊分割成三個部份。對話方塊的左側顯示所有可設定變數的樹狀結構檢視。當您選取變數時，右側會顯示目前的選擇，以及此變數的目前設定。在下方，第三個視窗顯示變數用途的簡短說明、可能值、預設值以及產生此變數的實際組態檔。對話方塊也提供關於變更變數後要執行的組態程序檔，以及變更結果會啟動什麼新服務等等的相關資訊。YaST 會要求您確認變更，並告訴您在選取「*完成*」並離開對話方塊後會執行的程序檔。請選取目前要略過的服務和程序檔，它們會在稍後啟動。為了讓變更生效，YaST 會自動套用所有變更並重新啟動已變更組態的服務。

20.3.2 手動變更系統組態

若要手動變更系統組態，請執行下列步驟：

1 以 root 使用者身份登入。

2 使用 `init 1`，將系統置於單一使用者模式 (runlevel 1)。

3 使用選擇的編輯器對組態檔進行所需變更。

如果未使用 YaST 來變更 `/etc/sysconfig` 中的組態檔，請確定空的變數值是由兩個引號所表示 (`KEYTABLE=""`)，而且該值與其中的空格包夾在引號中。由一個文字組成的值不需要包夾在引號中。

4 執行 `SUSEconfig` 來確定變更生效。

5 使用如 `init default_runlevel` 指令，使系統組態還原成先前的 Runlevel。採用系統預設的 Runlevel 來取代 *`default_runlevel`*。若要返回有網路和X的完整多重使用者模式，請選擇 `5`；若要在有網路的完整多重使用者模式下工作，請選擇 `3`。

變更整個系統設定，例如網路組態時，才會用到此程序。小幅度變更不必進入單一使用者模式，不過您也可以進入單一使用者模式，即可百分之百確定所有相關程式都正確重新啟動。

提示：設定自動系統組態

若要停用由 SuSEconfig 自動設定的系統組態，請將 `/etc/sysconfig/suseconfig` 中的 `ENABLE_SUSECONFIG` 變數設為 `no`。如果您想使用 SUSE 安裝支援，請勿停用 SuSEconfig也可以停用部份自動組態。

21 開機載入程式

本章說明如何設定 GRUB，這是 SUSE Linux Enterprise® 使用的開機載入程式。一種特殊的 YaST 模組可以用於執行所有設定。如果您不熟悉 Linux 開機的要點，請閱讀以下章節，取得部份背景資訊。本章節也敘述使用 GRUB 開機時，經常遇到的部份問題以及它們的解決方案。

本章節著重於開機管理以及開機載入程式 GRUB 的組態。整個開機程序簡述於第 20 章「*啟動及設定 Linux 系統*」[357頁]。開機載入程式是機器 (BIOS) 與作業系統 (SUSE Linux Enterprise) 之間的介面。開機載入程式的組態會直接影響作業系統的開機。

下列詞彙將在本章節中經常出現，而且可能需要說明：

主開機記錄
: MBR 的結構是由作業系統 (獨立會議) 所定義。前 446 位元組是保留給程式碼。它們通常包含一部分開機載入程式程式或作業系統選取器。接下來的 64 位元組提供多達四個分割區表格的空間 (請參閱章節「分割區類型」[143頁])。分割區表包含磁碟分割以及檔案系統類型相關資訊。作業系統需要此表格來處理硬碟。在 MBR 中使用傳統標準程式碼的情況下，只有一個分割區必須標示為*作用中*。MBR 的最後兩個位元組必須包含靜態的「魔術編號」(AA55)。包含不同值的 MBR 會被一些 BIOS 視為無效，所以開機時不列入考慮。

開機磁區
: 開機磁區是除了擴充分割區以外，硬碟分割區的第一個磁區，它只會當成其他分割區的「容器」。這些開機磁區有 512 位元組的空間供程式碼使用，而這些程式碼是用來啟動各自分割區中安裝的作業系統。此適用於 DOS、Windows 和 OS/2 分割區格式的開機磁區，它們也包含檔案系統的部份重要

基本資料。相反的，Linux 分割區的開機磁區一開始是空白，直到設定不同於 XFS 的檔案系統之後，才會寫入記錄。因此，Linux 分割區無法自行啟動，即使它包含核心以及有效的根檔案系統。開機磁區如果包含可以啟動系統的有效程式碼，則它的魔術編號與 MBR 的最後兩個位元組相同 (`AA55`)。

21.1 選取開機載入程式

根據預設，SUSE Linux Enterprise 會使用 GRUB 開機載入程式。不過，有時候以及在特殊軟、硬體的配合，可能就必須使用 LILO。如果您更新使用 LILO 的舊版 SUSE Linux Enterprise，則會安裝 LILO。

如需關於安裝和設定 LILO 的資訊，請參閱「支援資料庫」中的關鍵字 LILO 的部份 `/usr/share/doc/packages/lilo`。

21.2 使用 GRUB 開機

GRUB (Grand Unified Bootloader) 包含兩個階段。第一個階段包含 512 個位元組，而且唯一任務是將載入開機載入程式的第二階段。接下來，會載入 stage2。這個階段會包含開機載入程式的主要部分。

某些組態會使用中繼階段 1.5，用來配置和載入適當檔案系統的階段 2。在適當情況下，這種方法在安裝或使用 YaST 初始設定 GRUB 時會是預設選項。

stage2 可以存取許多檔案系統。目前支援 Ext2、Ext3、ReiserFS、Minix 和 Winodws 使用的 DOS FAT 檔案系統。對於 XFS、UFS 以及 BSD 使用的 FFS，也支援到一定的程度。自 0.95 版開始，GRUB 也可以從包含符合「El Torito」規格的 ISO 9660 標準檔案系統的 CD 或 DVD 啟動。即使在系統啟動之前，GRUB 可以存取支援的 BIOS 磁碟機 (BIOS 偵測到的磁片或硬碟、CD 光碟機和 DVD 光碟機) 的系統。因此，對 GRUB 組態檔 (`menu.lst`) 所做的變更，將不再需要重新安裝開機管理員。啟動系統後，GRUB 會重新載入功能表檔案以及核心或起始 RAM 磁碟 (`initrd`) 的有效路徑和分割區資料，然後找這些檔案。

GRUB 的實際組態是以下列敘述的三個檔案為基礎：

`/boot/grub/menu.lst`
此檔案包含可以使用 GRUB 啟動的分割區或作業系統，所有的相關資訊。如果沒有這段資訊，GRUB 指令行就會提示使用者提供如何繼續執行 (如需詳細資訊，請參閱章節「在開機程序期間編輯功能表項目」[377頁])。

`/boot/grub/device.map`
此檔案會從 GRUB 和 BIOS 表示法，將設備名稱轉譯成 Linux 設備名稱。

`/etc/grub.conf`
此檔案包含 GRUB 外圍程序正確安裝開機載入程式時需要的指令、參數和選項。

有多種方法可以控制 GRUB。現有組態啟動項目，可以從圖形功能表選取 (開頭顯示畫面)。組態會從檔案 `menu.lst` 載入。

在 GRUB，啟動前可以變更所有啟動參數。例如，編輯功能表檔案發生的錯誤，可以用此方法更正。開機指令也可透過輸入提示輸入 (請參閱章節「在開機程序期間編輯功能表項目」[377頁])。GRUB 提供開機前，判斷核心和 `initrd` 二者位置的可能性。以此方法，您還可以為開機載入程式組態中不存在的項目，啟動安裝的作業系統。

GRUB 事實上有兩個版本：一個是開機載入程式，一個是位於 `/usr/sbin/grub` 的一般 Linux 程式。此程式稱為 *GRUB 外圍程序*。它可以在安裝系統中提供 GRUB 模擬功能，並可用來安裝 GRUB 或是在套用之前測試設定。將 GRUB 安裝成硬碟或磁片上的開機載入程式，這種功能是以指令 `install` 和 `setup` 的形式，整合於 GRUB。Linux 載入時，可以在 GRUB 外圍程序使用此指令。

21.2.1 GRUB 開機功能表

圖形開頭顯示畫面以及開機功能表是以 GRUB 組態檔 `/boot/grub/menu.lst` 為基礎的，它包含可以透過功能表啟動的所有分割區或作業系統，全部的相關資訊。

每次啟動系統時，GRUB 會從檔案系統載入功能表檔案。基於此因素，GRUB 不需要在每次變更檔案後，重新安裝。使用 YaST 開機載入程式來修改 GRUB 組態，如第 21.3 節「使用 YaST 設定開機載入程式」[380頁]所述一般。

功能表檔案包含指令。語法相當簡單。每一行包含一個指令，後面是由像外圍程序中的空格所分開的選用參數。基於歷史因素，部份指令可以允許 = 放在第一個參數前面。註解是以井字號 (#) 開頭。

若要識別功能表綜覽中的功能表項目，請為每一個項目設定 `title`。關鍵字 `title` 後面的文字 (包括任何空格) 將在功能表中顯示成可選取的選項。當這個功能表項目被選取時，就會執行下一個 `title` 指示的所有指令。

最簡單的案例是重新導向至其他作業系統的開機載入程式。指令是`chainloader` 及引數通常是 GRUB 區塊表示法中，其他分割區的啟動區塊。例如：

```
chainloader (hd0,3)+1
```

GRUB 中的設備名稱會在章節「硬碟和分割區的命名慣例」 [375頁]說明。這個範例會指定第一個硬碟中，第四個分割區的第一個區塊。

使用指令`kernel` 來指定核心影像。第一個引數是分割區中，核心影像的路徑。其他引數會傳送到指令行上的核心。

如果核心沒有內建驅動程式來存取根分割區、或是這時是使用包含進階 HotPlug 功能的最新 Linux 系統，`initrd` 就必須以個別的 GRUB 指令來指定，它唯一的引數是 `initrd` 檔案路徑。因為 `initrd` 的載入位址是記錄於載入的核心影像，所以指令 `initrd` 必須立即接在 `kernel` 指令的後面。

指令`root` 會簡化核心和 initrd 檔案的指定。`root` 的唯一引數是設備或分割區。這個設備會用於所有一直到下一個 `root` 指令指定時才會出現明確指定設備的所有核心、`initrd` 或其他檔案路徑。

`boot` 指令會在每一個功能表項目最後暗示，所以它不需要寫入功能表檔案。不過，如果您使用互動式 GRUB 來啟動，必須在最後輸入 `boot` 指令。指令本身沒有引數。它只是啟動載入的核心影像或指定的鏈結載入器。

撰寫所有功能表項目之後，將其中一個定義為`default` 項目。否則，會使用第一個 (項目 0) 做為預設項目。您也可以指定預設項目要在幾秒後啟動。`timeout` 和`default` 通常在功能表項目前面。如需參考範例，請參閱章節「功能表檔案範例」 [375頁]。

硬碟和分割區的命名慣例

GRUB 用於硬碟和分割區的命名慣例與用於一般 Linux 設備的命名慣例不同。它比較類似 BIOS 的簡易磁碟列舉方式，而且語法類似一些 BSD 衍生版本中所用的語法。在 GRUB，分割區的編號會從 0 開始。這表示 (hd0,0) 是第一個硬碟的第一個分割區。與一般桌上型機器的 primary master 硬碟對應的 Linux 設備名稱是 /dev/hda1。

四個可能的主要分割區會指定分割區編號 0 到 3。邏輯分割區是從 4 開始編號：

```
(hd0,0)    first primary partition of the first hard disk
(hd0,1)    second primary partition
(hd0,2)    third primary partition
(hd0,3)    fourth primary partition (usually an extended partition)
(hd0,4)    first logical partition
(hd0,5)    second logical partition
```

由於獨立於 BIOS 設備，GRUB 不會區分 IDE、SATA、SCSI 和硬體 RAID 設備。BIOS 或其他控制器識別的所有硬碟，會按照 BIOS 的開機順序編號。

可惜的是，它通常無法將 Linux 設備名稱正確對應至 BIOS 設備名稱。它會透過演算法的協助來產生此對應，然後儲存至檔案 device.map，需要時可以編輯它。如需關於檔案 device.map 的資訊，請參閱第 21.2.2 節「檔案 device.map」[378頁]。

完整 GRUB 路徑包含放在括號中的設備名稱，以及指定分割區中檔案系統的檔案路徑。路徑開頭是一個斜線。例如，如果系統有一個 IDE 硬碟，它的第一個分割區包含 Linux，則可以使用下列方式設定可開機核心：

```
(hd0,0)/boot/vmlinuz
```

功能表檔案範例

以下範例顯示 GRUB 功能表檔案的結構。這個範例安裝會在 /dev/hda5 下安裝 Linux 啟動分割區、在 /dev/hda7 下安裝根分割區，以及在 /dev/hda1 下進行 Windows 安裝。

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8
```

```
title linux
   kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd

title windows
   chainloader(hd0,0)+1

title floppy
   chainloader(fd0)+1

title failsafe
   kernel (hd0,4)/vmlinuz.shipped root=/dev/hda7 ide=nodma \
   apm=off acpi=off vga=normal nosmp maxcpus=0 3
   initrd (hd0,4)/initrd.shipped
```

第一個區塊定義開頭顯示畫面的組態：

gfxmenu (hd0,4)/message
: 背景影像 `message` 位於 `/dev/hda5` 分割區的最上層目錄。

color white/blue black/light-gray
: 顏色配置：白色 (前景)、藍色 (背景)，黑色 (選擇項目)，以及淺灰色 (選擇項目的背景)。色彩配置對於開頭顯示畫面沒有作用，只對您可以存取而且可以自定的 GRUB 功能表有作用 (您可以使用 Esc，結束開頭顯示畫面)。

default 0
: 第一個功能表項目 `title linux` 是預設要啟動的項目。

timeout 8
: 沒有任何用者輸入，經過 8 秒後，GRUB 會自動啟動預設項目。若要停用自動開機，請刪除 `timeout`。如果設定為 `timeout 0`，GRUB 會馬上啟動預設項目。

第二個 (最大的) 區塊會列示各種可開機的作業系統。個別作業系統的區段會從 `title` 開始。

- 第一個項目 (`title linux`) 負責啟動 SUSE Linux Enterprise。核心 (`vmlinuz`) 是位在第一個硬碟的第一個邏輯分割區 (啟動分割區)。核心參數，例如根分割區和 VGA 模式，會在此添加。根分割區是根據 Linux 命名慣例 (`/dev/hda7/`) 指定的，因為此資訊是由核心讀取，與 GRUB 無關。`initrd` 也是位在第一個硬碟的第一個邏輯分割區。

- 第二個項目負責載入 Windows。Windows 是從第一個硬碟的第一個分割區啟動 (hd0,0)。指令 `chainloader +1` 會造成 GRUB 載入並執行指定分割區的第一個磁區。
- 下一個項目允許從磁片開機，無需修改 BIOS 設定。
- 開機選項 `failsafe` 會以選擇的核心參數來啟動 Linux，可以啟動發生問題的 Linux 系統。

功能表檔案可以在需要時變更。GRUB 會在下次啟動時使用修改的設定。使用 YaST 或選擇的編輯器，永久地編輯檔案。另一種方法是，使用 GRUB 的編輯功能，以互動方式暫時變更 (請參閱章節「在開機程序期間編輯功能表項目」[377頁])。

在開機程序期間編輯功能表項目

在圖形化開機功能表中，可以使用方向鍵選取要啟動的作業系統。如果選取 Linux 系統，可以在開機提示時輸入其他啟動參數。若要直接編輯個別功能表項目，請按 Esc 結束開頭顯示畫面，然後跳到 GRUB 文字功能表，再按 E。用此方式所做的變更，只會套用到目前開機，因此不會永久變更。

重要：開機程序期間的鍵盤配置

US 鍵盤配置是啟動時唯一可以使用的鍵盤配置。請參閱圖形 51.1「美國鍵盤配置」[831頁]中的圖形。

編輯功能表項目有利於修復無法再啟動的缺陷系統，因為開機載入程式錯誤的組態檔，可以手動輸入參數，便得以解決。在開機程序期間手動輸入參數，對於測試新設定但可避免損壞原始系統，有很大的幫助。

啟用編輯模式之後，使用方向鍵選取功能表項目來編輯組態。若要讓組態變成可編輯狀態，請再按 E 一次。使用此方法，在對於開機程序產生負面影響之前編輯錯誤的分割區或路徑指定。按 Enter，結束編輯模式並返回功能表。然後按 B 來啟動此項目。底下的說明文字會顯示進一步可行的動作。

若要永久地輸入變更的開機選項，然後傳送至核心，以 `root` 身份開啟檔案 `menu.lst`，然後將各自的核心參數加入到現有的指令行，並以空格分隔：

```
title linux
   kernel (hd0,0)/vmlinuz root=/dev/hda3 additional parameter
```

```
initrd (hd0,0)/initrd
```

GRUB 會在下次啟動系統時，使用新參數。另一種方法是，使用 YaST 開機載入程式模組做此變更。將新參數加入至現有的行，並用空格分開。

21.2.2 檔案 device.map

檔案 `device.map` 會將 GRUB 和 BIOS 設備名稱對應至 Linux 設備名稱。在包含 IDE 和 SCSI 硬碟的混合系統中，GRUB 必須透過特定程序來判斷開機順序，因為 GRUB 可能不會存取 BIOS 的開機順序資訊。GRUB 儲存分析的結果至檔案 `/boot/grub/device.map`。在 BIOS 中的開機順序設定成 IDE 在 SCSI 前面的系統，檔案 `device.map` 會顯示如下：

```
(fd0)  /dev/fd0
(hd0)  /dev/hda
(hd1)  /dev/sda
```

因為 IDE、SCSI 和其他硬碟的順序取決於各種因素，而且 Linux 無法識別對應，所以檔案 `device.map` 可以手動設定。如果您在啟動時發生問題，檢查此檔案中的順序是否對應至 BIOS 的順序，然後在需要時使用 GRUB 提示來暫時修改它。啟動 Linux 系統之後，檔案 `device.map` 可以透過 YaST 開機載入程式或其他選擇的編輯器，永久性編輯。

重要：SATA 磁碟

根據控制器的差異，SATA 磁碟可以識別成 IDE (/dev/hd*x*) 或 SCSI (/dev/sd*x*) 設備。

在手動變更檔案 `device.map` 之後，執行以下指令來重新安裝 GRUB。此指令會造成檔案 `device.map` 重新載入，並以 `grub.conf` 列示的指令執行：

```
grub --batch < /etc/grub.conf
```

21.2.3 檔案 /etc/grub.conf

除了 menu.lst 和 device.map 之外，第三個重要的 GRUB 組態檔是 /etc/grub.conf。此檔案包含 GRUB 外圍程序正確安裝開機載入程式時需要的指令、參數和選項：

```
root (hd0,4)
   install /grub/stage1 (hd0,3) /grub/stage2 0x8000 (hd0,4)/grub/menu.lst
   quit
```

下面介紹個別項目的意義：

root (hd0,4)
: 此指令告訴 GRUB 將以下指令套用至第一個硬碟的第一個邏輯分割區 (開機檔案的位置)。

install 參數
: 指令 grub 應該與參數 install 一起執行。開機載入程式的 stage1 應該安裝在擴充分割區容器 (/grub/stage1 (hd0,3))。這是稍微有點深奧的組態，但已知在許多情況下都有效。stage2 應該載入到記憶體位址 0x8000 (/grub/stage2 0x8000)。最後項目 ((hd0,4)/grub/menu.lst) 告訴 GRUB 到什麼地方尋找功能表檔案。

21.2.4 設定啟動密碼

即使作業系統啟動之前，GRUB 也可以存取檔案系統。沒有 root 許可權的使用者，在此時可以存取 Linux 系統中的檔案 (這些檔案在系統啟動後，他們並無法存取)。若要封鎖此類型的存取、或者防止使用者啟動特定作業系統，請設定開機密碼。

重要：開機密碼和開頭顯示畫面

如果您在 GRUB 使用啟動密碼，將不會顯示一般的開頭顯示畫面。

按照以下方式，以使用者 root 的身份設定開機密碼：

1 在 root 提示下，使用 grub-md5-crypt 將密碼加密：

```
# grub-md5-crypt
Password: ****
Retype password: ****
Encrypted: $1$lS2dv/$JOYcdxIn7CJk9xShzzJVw/
```

2 將加密字串貼到檔案 menu.lst 的全域區段：

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8
password --md5 $1$lS2dv/$JOYcdxIn7CJk9xShzzJVw/
```

現在 GRUB 指令只可以在按 P 並輸入密碼後的啟動提示時執行。不過，使用者仍然可以從開機功能表，啟動所有作業系統。

3 要防止一或多個作業系統從開機功能表啟動，將項目 lock 新增至沒有密碼便不可以啟動的 menu.lst 每一個段落。例如：

```
title linux
   kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd
   lock
```

重新啟動系統並從開機功能表選取 Linux 之後，將會顯示以下錯誤訊息：

```
Error 32: Must be authenticated
```

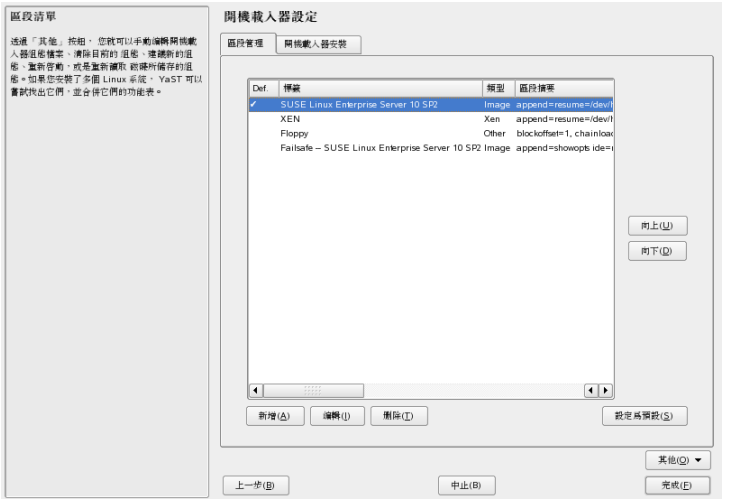
按 Enter 進入功能表。然後按 P，進入密碼提示要求。輸入密碼並按 Enter 之後，應該就可啟動選取的作業系統 (本範例為 Linux)。

21.3 使用 YaST 設定開機載入程式

在 SUSE Linux Enterprise 系統中設定開機載入程式最簡單的方法是使用 YaST 模組。在「YaST 控制中心」選取「*系統*」>「*開機載入程式組態*」。正如 圖形

21.1「開機載入程式設定」[381頁]所示，這時會顯示系統目前的開機載入程式組態，並允許您進行變更。

圖形 21.1　開機載入程式設定



使用「*磁區管理*」索引標籤來編輯、變更和刪除個別作業系統的開機載入程式磁區。要加入選項，請按一下「*新增*」。若要變更現有選項的值，請先用滑鼠選取，接著按一下「*編輯*」。如果要移除現有項目，請選取它並按一下「*刪除*」。如果不熟悉開機載入程式選項，請先閱讀第 21.2 節「使用 GRUB 開機」[372頁]。

使用「*開機載入程式安裝*」索引標籤，來檢視和變更關於類型、位置和進階載入器設定的設定。

存取下拉式功能表 (按一下「*其他*」即可開啓) 中的進階組態選項。內建的編輯器可讓您變更 GRUB 組態檔案 (請參閱第 21.2 節「使用 GRUB 開機」[372頁]以取得詳細資料)。還可以刪除現有的組態然後「*從頭開始*」，或者讓 YaST「*建議新的組態*」。也可以將組態寫入磁碟，或者從磁碟中重新讀取組態。要還原安裝期間儲存的原始主開機記錄，請選擇「*還原硬碟的 MBR*」。

21.3.1 開機載入程式類型

在「*開機載入程式安裝*」中設定開機載入程式類型。SUSE Linux Enterprise 會預設使用 GRUB 開機載入程式。若要使用 LILO，請繼續下列步驟：

過程 21.1 *變更開機載入程式類型*

1. 開啟「*開機載入程式安裝*」索引標籤。
2. 為「*開機載入程式*」選取「*LILO*」。
3. 在這時開啟的對話方塊中，選取下面其中一個動作：

   建議新組態
   指定 YaST 建議新的組態

   轉換目前的組態
   指定 YaST 轉換目前的組態。轉換組態時，可能會遺失某些設定。

   從頭開始設定新的組態
   寫入自定的組態。安裝 SUSE Linux Enterprise 時不能使用這個動作。

   讀取儲存在硬碟的組態
   載入您個人的 `/etc/lilo.conf`。安裝 SUSE Linux Enterprise 時不能使用這個動作。
4. 按一下「*確定*」來儲存變更。
5. 按一下主對話方塊視窗的「*完成*」，套用變更。

在轉換過程中，舊的 GRUB 組態會儲存在硬碟。若要使用它，只要將開機載入程式類型變更回GRUB，然後從快顯示功能表選擇「*還原轉換前儲存的組態*」。這個動作只能在已安裝系統上使用。

注：自定開機載入程式

如果要使用 GRUB 或 LILO 以外的開機載入程式，請選取「*不要安裝任何開機載入程式*」。請先詳細閱讀開機載入程式的說明文件，再選取這個選項。

21.3.2 開機載入程式位置

若要變更開機載入程式的位置，請執行下列步驟：

過程 21.2 *變更開機載入程式位置*

1. 請選取「*開機載入程式安裝*」索引標籤，然後為「*開機載入程式位置*」選取下面其中一個選項：

   從開機分割區開機
   : `/boot` 分割區的開機磁區

   從延伸分割區開機
   : 這會在延伸分割區容器中安裝開機載入程式。

   從主開機記錄開機
   : 這樣會在第一個磁碟的 MBR 中安裝開機載入程式 (根據 BIOS 中預設的開機順序)。

   從根分割區開機
   : 這會在 / 分割區的開機磁區中安裝開機載入程式。

   自定開機分割區
   : 這個選項可讓您手動指定開機載入程式的位置。

2. 按一下「*完成*」，套用變更。

21.3.3 預設系統

若要變更預設開機的系統，請依照下列步驟執行：

過程 21.3 *設定預設系統*

1. 開啟「*磁區管理*」索引標籤。

2. 從清單中選取所需項目。

3. 按一下「*設定為預設值*」。

4 按一下「*完成*」，啟用這些變更。

21.3.4 開機載入程式逾時

開機載入程式不會立即啟動預設系統。在逾時期間，您可以選取系統開機，或是寫入一些核心參數。若要設定開機載入程式逾時時間，請依照下列步驟執行：

過程 21.4 *變更開機載入程式逾時*

1 開啟「*開機載入程式安裝*」索引標籤。

2 按一下「*開機載入程式選項*」。

3 輸入新的值或使用滑鼠按住適當的箭號、或使用鍵盤上的箭號，來變更「*逾時秒數*」的設定值。

4 按一下「*確定*」。

5 按一下「*完成*」，儲存這些變更。

21.3.5 安全性設定

使用 YaST 模組，您也可以設定密碼來保護開機。這可以提供您另一層安全保護。

過程 21.5 *設定開機載入程式密碼*

1 開啟「*開機載入程式安裝*」索引標籤。

2 按一下「*開機載入程式選項*」。

3 在「*功能表介面密碼*」中設定您的密碼。

4 按一下「*確定*」。

5 按一下「*完成*」，儲存這些變更。

21.4 解除安裝 Linux 開機載入程式

YaST 可以用來解除安裝 Linux 開機載入程式，並將 MBR 還原回安裝 Linux 前的狀態。安裝時，YaST 會自動建立原始 MBR 的備份，並在需要時還原。

要解除安裝 GRUB，請開啟 YaST 開機載入程式模組 (「*系統*」>「*開機載入程式*」)。選取「*其他*」>「*還原硬碟的 MBR*」，並以「*是，重新寫入*」進行確認。

21.5 建立開機 CD

如果使用開機管理員來啟動系統發生問題，或者開機管理員無法安裝在硬碟或磁片的 MBR 上，也可以建立一張包含 Linux 所有必要啟動檔案的開機 CD。您的系統需要有一個 CD 燒錄器。

使用 GRUB 建立可開機的 CD-ROM 只需要一個特殊形式的 *stage2* (稱為 `stage2_eltorito`)，並可以選用自定的 `menu.lst`。不需要典型的檔案 `stage1` 和 `stage2`。

過程 21.6 *建立開機 CD*

1. 變更至要在其中建立 ISO 影像的目錄，例如：`cd /tmp`

2. 為 GRUB 建立子目錄：

   ```
   mkdir -p iso/boot/grub
   ```

3. 將核心以及`stage2_eltorito`、`initrd`、`menu.lst` 和 `message` 等檔案複製至 `iso/boot/`：

   ```
   cp /boot/vmlinuz iso/boot/
   cp /boot/initrd iso/boot/
   cp /boot/message iso/boot/
   cp /usr/lib/grub/stage2_eltorito iso/boot/grub
   cp /boot/grub/menu.lst iso/boot/grub
   ```

4. 調整 `iso/boot/grub/menu.lst` 中的路徑項目，使它們指向光碟機。以光碟機的設備名稱 (即 `(cd)`) 取代路徑名稱中以 `(sd*)` 格式列出的硬碟設備名稱，即可實現此目的。

```
timeout 8
default 0
gfxmenu (cd)/boot/message

title Linux
   root (cd)
   kernel /boot/vmlinuz root=/dev/sda5 vga=794 resume=/dev/sda1 \
   splash=verbose showopts
   initrd /boot/initrd
```

使用 `splash=silent` (而非使用 `splash=verbose`) 讓開機訊息不要出現在開機程序中。

5 使用以下指令建立 ISO 影像：

```
mkisofs -R -b boot/grub/stage2_eltorito -no-emul-boot \
-boot-load-size 4 -boot-info-table -o grub.iso /tmp/iso
```

6 使用您偏好的公用程式，將產生的檔案 `grub.iso` 燒錄至光碟。請勿將 ISO 影像燒錄為資料檔案；請在您的燒錄公用程式中使用燒錄光碟影像的選項。

21.6 圖形化 SUSE 畫面

從 SUSE Linux 7.2 開始，如果使用「`vga=value`」選項做為核心參數，則圖形 SUSE 螢幕會顯示在第一個主控台。如果您使用 YaST 來安裝，則會依照選取的解析度與圖形卡來自動啟用此選項。視需要，有三種方式可以停用 SUSE 畫面：

必要時停用 SUSE 畫面。

可以在指令行中輸入指令 `echo 0 >/proc/splash` 以停用圖形式畫面。若要再次啟用它，請輸入 `echo 1 >/proc/splash`。

預設停用 SUSE 畫面。

您可以新增核心參數 `splash=0` 到開機載入程式的組態。如需更多詳細資訊，請參閱第 21 章「*開機載入程式*」[371 頁]。但是，如果您想使用文字模式 (較早版本的預設值)，則請設定 `vga=normal`。

完全停用 SUSE 畫面

編譯新的核心且關閉「*框架緩衝區支援*」中的「*使用開機顯示畫面而非開機標幟*」選項。

提示

在核心中關閉 framebuffer 支援將會自動關閉開機顯示畫面。如果您使用自定核心來執行，SUSE 將不會為您的系統提供任何支援。

21.7 疑難排解

此章節列示使用GRUB啟動時，經常遇到的問題以及可能解決方案的簡要說明。部份問題已經在知識庫 (網址為 http://support.novell.com/) 的文章中做了說明。使用搜尋對話方塊尋找 *GRUB*、*開機*和*開機載入程式*這類的關鍵字。

GRUB 和 XFS

XFS 在分割區啟動區塊中，不會保留空間給 `stage1`。因此，不要將 XFS 分割區指定成開機載入程式的位置。您可以建立不是以 XFS 格式化的個別開機分割區來解決這個問題。

GRUB 報告 GRUB Geom 錯誤

GRUB 會在系統啟動時檢查連接硬碟的位置。有時候，BIOS 會傳回不一致的資訊，而且GRUB 會報告GRUB 位置錯誤。如果發生此狀況，使用LILO或更新BIOS。如需關於LILO的安裝、組態和維護的詳細資訊，請參閱「支援資料庫」中的關鍵字 LILO。

如果 Linux 安裝在其他硬碟上，而且未註冊在 BIOS，GRUB 也會傳回此錯誤訊息。開機載入程式的 *stage1* 可以正確找到和載入，不過 *stage2* 則找不到。將新磁碟登錄在 BIOS 就可以解決此問題。

包含 IDE 和 SCSI 硬碟的系統不會啟動

安裝時，YaST 可能已經判斷硬碟的開機順序錯誤。例如，GRUB 可以將 `/dev/hda` 當成 `hd0` 而將 `/dev/sda` 當成 `hd1`，即使 BIOS 中的啟動順序是其他方式 (SCSI 在 IDE *前面*)。

發生此狀況，在啟動程序時，透過 GRUB 指令行的協助來更正硬碟。在系統啟動之後，編輯 `device.map` 來永久套用新的對應。然後檢查檔案 `/boot/grub/menu.lst` 和 `/boot/grub/device.map` 中的GRUB設備名稱，然後使用以下指令，重新安裝開機載入程式：

```
grub --batch < /etc/grub.conf
```

從第二顆硬碟啟動 Windows

有些作業系統，例如 Windows，只可以從第一個硬碟啟動。在第一個硬碟以外的硬碟安裝類似作業系統時，會影響個別功能表項目的邏輯變更。

```
...
title windows
   map (hd0) (hd1)
   map (hd1) (hd0)
   chainloader(hd1,0)+1
...
```

在此範例中，Windows 是從第二個硬碟啟動。基於此目的，硬碟的邏輯順序是使用 `map` 變更的。此變更不會影響 GRUB 功能表檔案中的邏輯。因此，第二個硬碟必須指定成 `chainloader`。

21.8 如需更多資訊

如需關於 GRUB 更進一步的資訊，請參閱 http://www.gnu.org/software/grub/。另請參閱 `grub` info 頁面。您也可以在「技術資訊搜尋」搜尋關鍵字「GRUB」，取得關於特殊問題的相關資訊，網址是 http://www.novell.com/support。

22 特殊系統功能

本章節會提供關於軟體套件、虛擬主控台及鍵盤配置等資訊。介紹bash、cron和logrotate軟體元件，是因為這些元件自上一版後有所變更或加強。這些元件也許不很重要，但與系統的關係密切，使用者可能想變更它們的預設動作。本章最後一節則會介紹語言與國家的專用設定 (I18N 與 L10N)。

22.1 特殊軟體套件的資訊

程式bash、cron、logrotate、locate、ulimit和free以及檔案resolv.conf，對系統管理員和許多使用者而言十分重要。線上文件和 info 頁面是兩個很有用的指令資訊來源，但並非隨時都能使用。GNU Emacs 是非常普遍而且很好設定的文字編輯器。

22.1.1 bash 套件與 /etc/profile

Bash 是預設的系統外圍程序。如果以它做為登入外圍程序，可以讀取多種啟始化檔案。Bash 會以它們顯示在清單中的順序來處理。

1. /etc/profile
2. ~/.profile
3. /etc/bash.bashrc
4. ~/.bashrc

在 ~/.profile 或 ~/.bashrc 中進行自定設定。為了要確保這些檔案能正確的處理，您必須將基本設定從 /etc/skel/.profile 或 /etc/skel/.bashrc 中複製至使用者的主目錄。建議您在更新後從 /etc/skel 複製設定。請執行下列的外圍程式指令，以避免遺失您調整過的設定。

```
mv ~/.bashrc ~/.bashrc.old
cp /etc/skel/.bashrc ~/.bashrc
mv ~/.profile ~/.profile.old
cp /etc/skel/.profile ~/.profile
```

然後個人的調整設定需要從 *.old 檔案再複製回去。

22.1.2 cron 套件

如果您需要按預先定義的時間，在背景中定期自動執行指令，則可以使用 cron 工具。cron 由採用特殊格式的時間表驅動。其中某些表格是由系統提供，使用者可以視需要撰寫自己的表格。

cron 表格現在位於 /var/cron/tabs。/etc/crontab 做為整個系統的 cron 表格。在時間表格之後、指令之前，輸入要直接執行指令的使用者名稱。在 範例 22.1「/etc/crontab 中的項目」[390頁] 中，則是輸入 root。位於 /etc/cron.d 的套件專用表格有相同的格式。請參閱 cron 線上文件 (man cron)。

範例 22.1 */etc/crontab 中的項目*

```
1-59/5 * * * *   root   test -x /usr/sbin/atrun && /usr/sbin/atrun
```

您不能呼叫 crontab -e 指令來編輯 /etc/crontab。這個檔案必須直接載入編輯器中進行修改，然後儲存。

有些套件會將外圍程式程序檔安裝至 /etc/cron.hourly、/etc/cron.daily、/etc/cron.weekly 及 /etc/cron.monthly 等目錄中，其執行由 /usr/lib/cron/run-crons 控制。/usr/lib/cron/run-crons 每隔 15 分鐘會從主表格 (/etc/crontab) 執行一次。這會保證被忽略的程序可以在適當的時間執行。

如果要按照自定時間，執行 hourly、daily 或其他定期維護程序檔，請定期使用 /etc/crontab 項目移除時間戳記檔案 (請參閱 範例 22.2「/etc/crontab：移除時戳檔案」[391頁]，它可以在整點前移除 hourly、在每天的 2:14 a.m. 移除 daily 等)。

範例 22.2　/etc/crontab：移除時戳檔案

```
59 *  * * *     root  rm -f /var/spool/cron/lastrun/cron.hourly
14 2  * * *     root  rm -f /var/spool/cron/lastrun/cron.daily
29 2  * * 6     root  rm -f /var/spool/cron/lastrun/cron.weekly
44 2  1 * *     root  rm -f /var/spool/cron/lastrun/cron.monthly
```

或者，可以將 /etc/sysconfig/cron 中的 DAILY_TIME 設定為 cron.daily 啟動的時間。MAX_NOT_RUN 設定確保能夠觸發日常工作的執行，即使使用者在長時間內未按指定的 DAILY_TIME 開啟電腦。MAX_NOT_RUN 的最大值為 14 天。

為明確起見，日常系統維護工作會配送至不同的程序檔。它們包含在 aaa_base 套件中。例如，/etc/cron.daily 中有 suse.de-backup-rpmdb、suse.de-clean-tmp 或 suse.de-cron-local 等元件。

22.1.3 記錄檔：套件 logrotate

某些系統服務 (*daemon*) 以及核心本身，會定期將系統狀態與特定事件記錄到記錄檔中。這樣管理員可以定期在某個時間點檢查系統的狀態、找出錯誤或有問題的功能，並且用精確的方式來排除它們。這些記錄檔通常以 FHS 所指定的方式儲存於 /var/log，而且會日益增大。logrotate 套件有助於控制這些檔案增大的方式。

使用檔案 /etc/logrotate.conf 來設定 logrotate。尤其 include 規格主要是設定其他要讀取的檔案。產生記錄檔的程式會在 /etc/logrotate.d 中單獨安裝組態檔。例如，套件隨附的檔案，如 apache2 (/etc/logrotate.d/apache2) 與 syslogd (/etc/logrotate.d/syslog)。

範例 22.3　/etc/logrotate.conf 的範例

```
# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
#compress

# RPM packages drop log rotation information into this directory
include /etc/logrotate.d

# no packages own lastlog or wtmp - we'll rotate them here
#/var/log/wtmp {
#    monthly
#    create 0664 root utmp
#    rotate 1
#}

# system-specific logs may be also be configured here.
```

logrotate 是透過 cron 來控制，並且每日是經由 `/etc/cron.daily/logrotate` 來呼叫。

重要

create 選項會讀取 `/etc/permissions*` 中由管理員所做的所有設定。請確定個人的修改不會造成衝突。

22.1.4 locate 指令

可以快速尋找檔案的 locate 指令，並不包含在安裝軟體的標準範圍中。若有需要，請安裝 `findutils-locate` 套件。updatedb 程序會在每晚自行啟動，或啟動系統後的 15 分鐘左右啟動。

22.1.5 ulimit 指令

利用 ulimit (*使用者限制*) 指令，您可以限制系統資源的使用，並顯示這些限制。ulimit 對於限制應用程式可用的記憶體特別有用。利用它，可以避免應用程式使用過多的記憶體空間，使用過多的記憶體空間可能會導致系統暫停。

ulimit 可以搭配多種選項來使用。若要限制記憶體的使用，請利用 表格 22.1 「ulimit：設定使用者的資源」 [393頁] 中所列的選項。

表格 22.1 *ulimit：設定使用者的資源*

-m	實體記憶體的最大大小
-v	虛擬記憶體的最大大小
-s	堆疊的最大大小
-c	核心檔案的最大大小
-a	限制集的顯示

您可以在 /etc/profile 中設定全系統的項目。在此可建立核心檔，以供程式設計人員*除錯*之用。一般使用者無法增加系統管理員在 /etc/profile 中所指定的值，但可以在 ~/.bashrc 中建立特殊的設定項目。

範例 22.4 *ulimit：~/.bashrc 中的設定*

```
# Limits of physical memory:
ulimit -m 98304

# Limits of virtual memory:
ulimit -v 98304
```

記憶體的單位必須為 KB。如需詳細資訊，請參閱 man bash。

重要

並非所有的外圍程序都支援 ulimit 指示詞。在您根據這些限制的內含設定時，PAM (例如 pam_limits) 會提供全面的調整設定。

22.1.6 free 指令

如果您的目標是要找出目前使用了多少 RAM 的話，則是稍微誤解 `free` 指令的用途。您可以在 `/proc/meminfo` 找到此資訊。近來，使用 Linux 之類新式作業系統的使用者，真的不太需要擔心記憶體的問題。*可用的RAM*的概念要回溯到聯合記憶體管理的年代之前。*記憶體要物盡其用的*口號非常適用於 Linux。所以，Linux 一直致力於平衡快取，而不允許有剩餘或未使用的記憶體。

基本上，核心不會有任何應用程式或使用者資料的的直接知識。相反地，它會在*頁面快取*中管理應用程式與使用者資料。如果記憶體不足，部份的記憶體資料會寫入交換分割區或檔案中，這部份一開始就能用 `mmap` 指令的協助來讀取 (請參閱 `man mmap`)。

核心也可以有其他的快取，例如*slab 快取*，網路存取的快取資料會儲存於此處。這可以解釋 `/proc/meminfo` 中計數器之間的不同。它們大部分但非全部都可以透過 `/proc/slabinfo` 來存取。

22.1.7 /etc/resolv.conf 檔案

領域名稱解析會透過 `/etc/resolv.conf` 檔案來處理。 請參閱 第 33 章「*網域名稱系統*」[559頁].

只有 `/sbin/modify_resolvconf` 程序檔可以更新此檔案，其他的程式沒有權限可以直接修改 `/etc/resolv.conf`。能保證系統的網路組態與相關檔案都保持在一致狀態的唯一方法，就是強制執行此規則。

22.1.8 線上文件和資訊頁面

某些 GNU 應用程式 (例如 tar) 不再支援 man 頁面。針對這些指令，請使用 `--help` 選項來取得 info 頁面的快速綜覽，這些都會提供更深入詳盡的說明。info 是 GNU 的超文字系統。您可以輸入 `infoinfo` 來讀取此系統的介紹。您可以輸入 `emacs -f info` 或直接在主控台中使用 `info`，以便使用 Emacs 檢視 info 頁面。您也可以使用 tkinfo、xinfo 或 說明系統檢視 info 頁面。

22.1.9 GNU Emacs 的設定

GNU Emacs 是個複雜的工作環境。以下幾個小節包含在 GNU Emacs 啟動時組態檔案的處理情形。更多相關資訊可在 http://www.gnu.org/software/emacs/ 取得。

啟動時，Emacs 會讀取多個檔案，包含使用者、系統管理員與供應商的設定，以及取得自定或預設組態的設定。啟始化檔案 ~/.emacs 會從 /etc/skel 安裝至個別使用者的主目錄。.emacs 接著會讀取 /etc/skel/.gnu-emacs 檔案。如果要自定程式，請將 .gnu-emacs 複製到主目錄 (利用 cp /etc/skel/.gnu-emacs ~/.gnu-emacs 指令)，並依照您的需求來設定。

.gnu-emacs 定義 ~/.gnu-emacs-custom 檔案為 *自定檔案*。如果使用者是使用 Emacs 中的 *自定* 選項來進行設定，這些設定會儲存至 ~/.gnu-emacs-custom 中。

透過 SUSE® Linux Enterprise，emacs 套件可將檔案 site-start.el 安裝至目錄 /usr/share/emacs/site-lisp 中。site-start.el 檔案會在啟始化檔案 ~/.emacs 前載入。此外，site-start.el 會確保那些以 Emacs 附加套件來散佈的特定組態檔案皆能自動載入，例如 psgml。此類型的組態檔案也位於 /usr/share/emacs/site-lisp 中，並且會以 suse-start- 為開頭。本地系統管理員可在 default.el 中指定整個系統的設定。

有關這些檔案的詳細資訊可在 *Init File* 下的 Emacs 資訊檔案中取得：info:/emacs/InitFile。關於如何關閉這些檔案的載入 (若有需要) 的資訊，也可在此取得。

Emacs 的元件分成數個套件：

- emacs 基本套件。
- emacs-x11 (通常已安裝)：*具有* X11 支援的程式。
- emacs-nox：*沒有* X11 支援的程式。
- emacs-info：info 格式的線上文件。
- emacs-el：以 emacs lisp 編寫的未編譯文件庫檔案。執行期間用不到這類檔案。

- 需要的話可安裝多種外掛套件：`emacs-auctex` (LaTeX 用)、`psgml` (SGML 與 XML 用)、`gnuserv` (用戶端與伺服器作業用)，以及其他。

22.2 虛擬主控台

Linux 是多重使用者及多工的作業系統。這些功能的優點即使在獨立的個人電腦系統中一樣令人讚賞。在文字模式中，有六個虛擬主控台可用。請使用 Alt + F1 到 Alt + F6 這些鍵來切換虛擬主控台。第七個主控台保留給 X 使用，第十個主控台可以顯示核心訊息。您可以修改 `/etc/inittab` 檔案來指定較多或較少的主控台。

若要在不關閉主控台的情況下，從 X 切換到主控台，請使用 Ctrl + Alt + F1 到 Ctrl + Alt + F6 這些鍵。若要回到 x，請按 Alt + F7。

22.3 鍵盤配置

若要標準化程式的鍵盤配置，請變更下列的檔案：

```
/etc/inputrc
/etc/X11/Xmodmap
/etc/skel/.Xmodmap
/etc/skel/.exrc
/etc/skel/.less
/etc/skel/.lesskey
/etc/csh.cshrc
/etc/termcap
/usr/lib/terminfo/x/xterm
/usr/share/X11/app-defaults/XTerm
/usr/share/emacs/VERSION/site-lisp/term/*.el
```

這些變更僅會影響使用 `terminfo` 項目的應用程式，或直接變更其組態檔的應用程式 (`vi`、`less` 等等。)。未隨附於此系統的應用程式必須相容於這些預設值。

在 X 中，可以使用 ctrl + shift (右邊) 來存取組合鍵 (組合鍵)。請參閱 `/etc/X11/Xmodmap` 中的對應項目。

使用 X 鍵盤延伸程式 (XKB)，可以進行進一步的設定。GNOME (gswitchit) 以及 KDE (kxkb) 桌面環境也會使用此延伸程式。

提示：如需更多資訊

有關 XKB 資訊可在 `/etc/X11/xkb/README` 檔案中取得，也可以從列於該檔案中的文件取得。

關於中文、日文以及韓文 (CJK) 輸入的詳細資訊，可在 Mike Fabian 網頁中取得：`http://www.suse.de/~mfabian/suse-cjk/input.html`。

22.4 語言與國家專用的設定

本系統在很大程度上是國際化的軟體，而且能以彈性的方式針對當地的需求進行修改。換句話說，國際化 (*I18N*) 允許特定的當地語系化 (*L10N*)。I18N 與 L10N 這兩個縮寫是取首尾兩個字母，兩字母中間再加上省略的字母數目。

設定位於 `/etc/sysconfig/language` 中所定義的 `LC_` 變數。設定範圍除了*本地語言支援*外，還包括*訊息* (語言)、*字元集*、*排序順序*、*時間和日期*、*數字*及*貨幣*等類別。每種類別都可以用自己的變數來直接定義，或用在 `language` 檔案中的主要變數來間接定義 (請參閱 `locale` man 頁面)。

`RC_LC_MESSAGES`、`RC_LC_CTYPE`、`RC_LC_COLLATE`、`RC_LC_TIME`、`RC_LC_NUMERIC`、`RC_LC_MONETARY`
: 這些變數會傳送到外圍程序，但不會包含 `RC_` 字首，並代表列出的類別。相關外圍程序設定檔會列於下面。目前的設定可以用 `locale` 指令來顯示。

`RC_LC_ALL`
: 此變數 (如果設定) 會覆寫先前所提到的變數值。

`RC_LANG`
: 如果沒有設定前面的變數，則此為備用變數。依照預設，只會設定 `RC_LANG`。這讓使用者更容易輸入自己的值。

`ROOT_USES_LANG`
: 有 `yes` 或 `no` 兩個變數。如果設為 `no` 的話，則 `root` 永遠可在 POSIX 環境中作業。

變數可以用 YaST sysconfig 編輯器來設定 (請參閱第 20.3.1 節「使用 YaST Sysconfig 編輯器變更系統組態」[368頁])。這樣的變數值中包含語言碼、國碼、編碼及修飾元。個別的元件會以特定的字元來連接：

```
LANG=<language>[[_<COUNTRY>].<Encoding>[@<Modifier>]]
```

22.4.1 一些範例

您必須將語言與國碼一起設定。語言設定必須符合http://www.evertype.com/standards/iso639/iso639-en.html和http://www.loc.gov/standards/iso639-2/中的標準 ISO 639。國碼列在http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/en_listp1.html中的 ISO 3166。

只有設定那些可以在 `/usr/lib/locale` 中找到的可用描述檔案的值，才會有意義。您可以用 `localedef` 指令從`/usr/share/i18n` 中的檔案建立其他描述檔；描述檔屬於 `glibc-i18ndata` 套件的一部份。`en_US.UTF-8` (針對美式英文) 的描述檔可以用以下指令建立：

```
localedef -i en_US -f UTF-8 en_US.UTF-8
```

`LANG=en_US.UTF-8`
: 如果安裝期間選擇美式英文的話，則此為預設設定。如果您選擇了其他語言，則仍然可以使用該語言，但會以 UTF-8 做為字元編碼。

`LANG=en_US.ISO-8859-1`
: 這會將語言設成英文、國家設成美國、字元集設成 `ISO-8859-1`。此字元集並不支援歐元符號，但有時對於尚未支援 `UTF-8` 的程式卻非常實用。然後，有些程式將會評估定義字元集的 (此例為 `ISO-8859-1`) 的字串，像是 Emacs。

`LANG=en_IE@euro`
: 上方範例在語言設定中明確包括歐元符號。嚴格來說，這個設定現在已不再使用，因為 UTF-8 也涵蓋歐元符號。它只有在應用程式不支援 UTF-8 而只支援 ISO-8859-15 時才有用。

SuSEconfig 會讀取 `/etc/sysconfig/language` 中的變數，並將必要的變更寫入 `/etc/SuSEconfig/profile` 與 `/etc/SuSEconfig/csh.cshrc` 中。`/etc/profile` 將讀取 */etc/SuSEconfig/profile*，或將之做為來源。`/etc/SuSEconfig/csh.cshrc` 則是 `/etc/csh.cshrc` 的來源。這讓這些設定能在整個系統中使用。

使用者可以適當地編輯自己的 ~/.bashrc 來覆寫系統預設值。例如，若不要讓整個系統的程式訊息皆使用 en_US 時，請加入 LC_MESSAGES=es_ES，便會改用西班牙文來顯示訊息。

22.4.2 ~/.i18n 中的地區設定

如果您對區域設定的系統預設值不滿意，可以根據Bash指令碼語法在 ~/.i18n 中變更設定值。~/.i18n 中的項目會覆寫 /etc/sysconfig/language 中的系統預設值。使用相同變數名稱，但不用 RC_ 名稱空間字首，例如，以 LANG 取代 RC_LANG：

```
LANG=cs_CZ.UTF-8
LC_COLLATE=C
```

22.4.3 語言支援的設定

按照規定，在*訊息*類別中的檔案僅會儲存於對應的語言目錄中 (像是 en)，以便有備用可用。如果您將 LANG 設為 en_US，而且 /usr/share/locale/en_US/LC_MESSAGES 中的訊息檔案不存在的話，則它會回到 /usr/share/locale/en/LC_MESSAGES 中。

您也可以定義備用鍊，例如，不列塔尼文之於法文，或是加里斯亞文之於西班牙文之於葡萄牙文：

```
LANGUAGE="br_FR:fr_FR"

LANGUAGE="gl_ES:es_ES:pt_PT"
```

如有需要，請改用挪威文變體 Nynorsk 與 Bokmal (讓其他備用為否)：

```
LANG="nn_NO"

LANGUAGE="nn_NO:nb_NO:no"
```

或

```
LANG="nb_NO"

LANGUAGE="nb_NO:nn_NO:no"
```

請注意，在挪威文中，會以不同方式處理 LC_TIME。

如果無法適當的辨識界定位數群組的分隔符號，可能會發生問題。如果 LANG 設定為類似 de 的兩個字母的語言碼，但卻使用 /usr/share/lib/de_DE/LC_NUMERIC 中的定義檔 glibc，就會發生這種情形。因此，LC_NUMERIC 必須設定為 de_DE，讓系統能辨識分隔符號定義。

22.4.4 如需更多資訊

- *The GNU C Library Reference Manual* 的「Locales and Internationalization」一章。包含在 glibc-info 中。
- Markus Kuhn 所寫的 *UTF-8 and Unicode FAQ for Unix/Linux*，目前網址如下：http://www.cl.cam.ac.uk/~mgk25/unicode.html。
- *Unicode-Howto*，Bruno Haible 著：/usr/share/doc/howto/en/html/Unicode-HOWTO.html。

23 印表機操作

SUSE Linux Enterprise® 可支援以多種類型的印表機進行列印，包括遠端網路印表機。印表機可經由 YaST 或手動設定。圖形和指令行公用程式都可用來啟動和管理列印工作。如果您的印表機無法如預期般運作，請參閱第 23.9 節「疑難排解」 [415頁]。

CUPS 是 SUSE Linux Enterprise 中的標準列印系統。CUPS 為高度使用者導向。在許多情況下，可與 LPRng 相容，或稍做努力便可與其搭配。SUSE Linux Enterprise 包含 LPRng 完全是基於相容性考量。

印表機可藉由介面 (例如 USB 或網路) 和印表機語言加以區分。購買印表機時，請確定印表機的介面 (例如 USB 或並列埠) 是否也可在您的硬體上找到，並請確定印表機的語言符合所需。印表機可根據下列三種印表機語言來分類：

PostScript 印表機
: Linux 和 Unix 內部列印系統以 PostScript 印表機語言產生和處理大部分列印工作。此語言已經很舊且非常有效率。如果印表機可直接處理 PostScript 文件，且不需轉換到列印系統中其他階段，潛在錯誤來源的次數便會減少。因為 PostScript 受限於高額的授權成本，這些印表機通常較無 PostScript 解譯器的印表機昂貴。

標準印表機 (PCI 和 ESC/P 語言)
: 雖然這些印表機語言很舊，它們仍在擴充以處理印表機的新功能。如果已知印表機語言，列印系統可以藉由 Ghostscript 的協助，將 PostScript 工作轉換為對應的印表機語言。此處理階段稱為解譯。最知名的語言為 PCL，大多為 HP 印表機與其複製品所使用；另一個是 ESC/P，大多為 Epson 印表機所使用。Linux 通常支援這些印表機語言，並可產生不錯的列印效果。Linux 可能無法處理非常新且新潮的印表機的部分功能，因為開放原始碼開發人員

仍在研究這些功能。除了 HP 所開發的 `hpijs` 驅動程式之外，目前沒有印表機製造商在開發 Linux 驅動程式並將開放原始碼授權提供給 Linux 散發者。這些印表機大部分屬於中等價位。

專屬印表機 (也稱為 GDI 印表機)
: 這些印表機並不支援任何一般的印表機語言。它們使用自己的印表機語言，而當有新型號發行，那些語言也可能有所變更。這些印表機通常指有 Windows 驅動程式。如需相關資訊，請參閱第 23.9.1 節「沒有標準印表機語言模式支援的印表機」[415頁]。

在購買新印表機之前，請參考下列來源以檢查您想要購買的印表機之支援性：

`http://www.linuxprinting.org/`
: LinuxPrinting.org 印表機資料庫。

`http://www.cs.wisc.edu/~ghost/`
: Ghostscript 網頁。

`/usr/share/doc/packages/ghostscript/catalog.devices`
: 已包含驅動程式的清單。

線上資料庫會永遠顯示最新的 Linux 支援狀態。但是，Linux 版本僅可與生產期間可用的驅動程式整合。因此，目前被評比為「完全支援」的印表機，在最新的 SUSE Linux Enterprise 版本發行之後，可能情況就會改變。因此，資料庫不一定能指出正確狀態，而僅提供估計值。

23.1 列印系統的工作流程

使用者會建立列印工作。列印工作的組成元素為要列印的資料加上暫存序列器的資訊 (例如印表機的名稱或是印表機佇列的名稱)，以及非必要的過濾器資訊 (例如，印表機的特定選項)。

每一台印表機都至少有一個專屬的印表機佇列。暫存序列器會在佇列中列印工作，直到所需的印表機已準備好接收資料。當印表機備妥時，暫存序列器會透過過濾器與後端，傳送資料至印表機。

過濾器會將列印應用程式所產生的資料 (通常為 PostScript 或 PDF，但也會有 ASCII、JPEG 等) 轉換為印表機特定資料 (PostScript、PCL、ESC/P 等)。印表機

的特性描述在 PPD 檔案中。PPD 檔案含有印表機特定選項以及在印表機上啟用它們所需的參數。過濾器系統可確保啟用使用者所選取的選項。

如果您是使用 PostScript 印表機，過濾器系統會將資料轉換為印表機特定的 PostScript。這並不需要印表機驅動程式。如果您是使用非 PostScript 印表機，過濾器系統會將資料轉換為使用 Ghostscript 的印表機特定資料。這將需要印表機適用的 Ghostscript 印表機驅動程式。後端會從過濾器接收印表機特定的資料，然後將它傳送至印表機。

23.2 連接印表機的方法和通訊協定

有各種方法可將印表機連接到系統。CUPS 列印系統的組態無法辨識本地印表機和透過網路連接到系統的印表機。在 Linux 中，本地印表機必須依照印表機製造商的手冊所述方式連接。CUPS 支援序列埠、USB、並列埠和 SCSI 連接。如需關於印表機連接的詳細資訊，請參閱「支援資料庫」文章〈*CUPS in a Nutshell*〉，網址為 http://en.opensuse.org/SDB:CUPS_in_a_Nutshell。

► **zseries:** CUPS 或 LPRng 不支援 z/VM 提供的可在本地連接到 IBM System z 主機的印表機或相似設備。在這些平台上，僅可透過網路列印。網路印表機的電纜必須根據印表機製造商的說明來安裝。 ◄

警告：在執行中的系統變更纜線連接

在將印表機連接到機器時，請不要忘記只有 USB 設備可在操作中插上和拔除。若要避免損壞您的系統或印表機，請先關機再變更任何非 USB 的連接。

23.3 安裝軟體

PPD (PostScript 印表機描述) 為描述內容 (如解析度) 和選項 (如雙面列印模組的可用性) 的電腦語言。這些描述是使用 CUPS 中各種印表機選項所需。沒有 PPD 檔案，列印資料會被轉送給處於「raw」狀態的印表機，這通常不是想要的狀態。在 SUSE Linux Enterprise 安裝期間，會預先安裝許多 PPD 檔案，以便即使在沒有 PostScript 支援的情況下仍能使用印表機。

若要設定 PostScript 印表機，最好的方法是取得適當的 PPD 檔。在標準安裝範圍中自動安裝的套件 `manufacturer-PPDs` 提供許多 PPD 檔案。請參閱第

23.8.3 節「各種套件中的 PPD 檔案」[413頁]和第 23.9.2 節「PostScript 印表機沒有可用的 PPD 檔案」[415頁]。

新的 PPD 檔案可儲存在目錄 `/usr/share/cups/model/` 中，或以 YaST (請參閱 章節「以 YaST 新增 PPD 檔案」[407頁]) 新增到列印系統。之後，便可在安裝期間選取 PPD 檔案。

請小心，印表機製造商是否除了修改組態檔之外，要您安裝整個軟體套裝。首先，這種安裝會導致遺失 SUSE Linux Enterprise 所提供的支援，其次印表機指令可能會以不同方式運作，使系統無法處理其他製造商的設備。基於此原因，不建議安裝製造商軟體。

23.4 設定印表機

YaST 可用來設定直接連接到您機器 (一般使用 USB 或並列埠) 的本地印表機，或設定網路上列印。亦可以 YaST 將 PPD (PostScript 印表機描述) 檔案新增到您的印表機。

23.4.1 設定本地印表機

若偵測到未設定的本地印表機，YaST 會自動開始設定。如果並列埠或 USB 埠可以自動設定，並偵測到連接的印表機，YaST 就能夠自動設定印表機。印表機型號必須列於自動硬體偵測過程中所使用的資料庫中。

若印表機型號不詳或無法自動偵測，請手動設定。未自動偵測到印表機的可能原因有二：

- 印表基本身無法正確表明身份。有可能是非常舊的設備。請嘗試使用章節「手動設定」[405頁]所述方式設定您的印表機。
- 若手動設定不成功，印表機與電腦間就無法通訊。請檢查纜線與接頭，並確認已穩固連接印表機。若是這個狀況的話，問題可能與印表機無關，而是 USB 或並列埠的相關問題。

手動設定

若要手動設定印表機，請選擇 YaST 控制中心裡的「*硬體*」>「*印表機*」。這樣會開啟主要「*印表機組態*」視窗，所偵測到的設備會列在視窗上方。下方部份列出目前為止已設定的所有佇列 (請參閱第 23.1 節「列印系統的工作流程」[402頁]以瞭解印表機佇列的更多資訊)。若未偵測到印表機，則組態視窗的兩個部份都會空白。使用「*編輯*」變更列出印表機的組態，或使用「*新增*」設定未自動偵測到的印表機。編輯現有組態，是使用與手動新增本地印表機 [405頁]中相同的對話。

您亦可在「*印表機組態*」中，「*刪除*」現有項目。按一下「*其他*」，開啟具有進階選項的清單。選取「*重新啟動偵測*」，以手動啟動印表機自動偵測。若電腦連接了多部印表機，或印表機上設有多個佇列，則您可將現用項目設為預設。「*CUPS 進階設定*」與「*變更 IPP 監聽*」為進階組態選項，請參閱第 23 章「*印表機操作*」[401頁]以得知詳細資訊。

過程 23.1 *手動新增本地印表機*

提示：YaST 列印測試

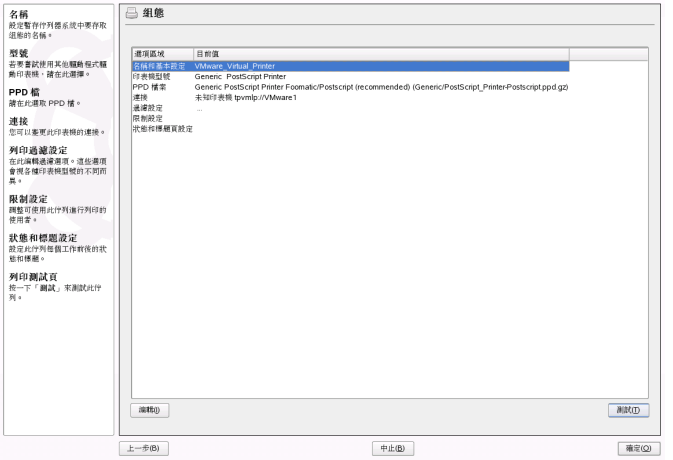
為確保一切功能正常，每個重要設定步驟都要用 YaST 的列印測試功能檢測。YaST 的測試頁也提供受測組態的重要資訊。例如，如果輸出時出現問題，而且有幾頁幾乎空白，您可移除所有紙張，然後停止 YaST 的測試，停止列印。

1. 啟動 YaST 並選擇「*硬體*」>「*印表機*」以開啟「*印表機組態*」對話。
2. 按一下「*新增*」開啟「*印表機類型*」視窗。
3. 選擇「*直接連接的印表機*」。
4. 選取印表機所連接的埠 (通常是 USB 或並列埠)，並在下一個設定畫面選擇設備。建議您在此時「*測試印表機連線*」。若發生問題，請選取正確設備，或選擇「*上一步*」回到上一個對話。
5. 在「*佇列名稱*」中，設定列印佇列。一定要指定「*列印名稱*」。建議選擇可辨識的名稱，以方便您日後於應用程式的列印對話中識別印表機。使用「*印表機描述*」與「*印表機位置*」進一步描述印表機。這是選擇性的，但若您電腦上連接多部印表機，或設定列印伺服器的話，就非常實用。若是本地印表機，須勾選「*執行本地過濾*」。

6 在「*印表機型號*」中，由「*製造商*」與「*型號*」指定印表機。若您的印表機未列出，可嘗試使用製造商清單中的「*未知的製造商*」，並從型號清單中選取適當的標準語言 (控制印表機的一組指令)，請參閱您印表機的說明文件以得知您印表機使用的語言。若不成功的話，請參閱章節「以 YaST 新增 PPD 檔案」[407頁]瞭解其他可能解決方案。

7 「*組態*」畫面列出印表機設定的摘要。從此 YaST 模組中編輯現有印表機組態時，也會顯示此對話。

圖形 23.1 印表機組態摘要



摘要包含下列項目，您亦可如下修改，使用「*編輯*」：

- 遵循此步驟時，「*名稱與基本設定*」、「*印表機型號*」，與「*連線*」可讓您變更項目。
- 請參閱章節「以 YaST 選擇其他 PPD 檔案」[407頁]以瞭解「*PPD 檔案*」的詳細資料。
- 透過「*過濾設定*」微調印表機設定。這裡有「*頁面大小*」、「*色彩模式*」，與「*解析度*」等設定選項。
- 依照預設，每個使用者都可使用印表機。使用「*限制設定*」，列出限制使用此印表機的使用者，或列出允許使用此印表機的使用者。

- 例如，您可使用「*狀態與標題頁設定*」，將印表機狀態變更為停用，或指定列印工作前後是否要列印「*開始標題頁*」或「*結束標題頁*」(預設是不列印的)。

以 YaST 新增 PPD 檔案

若您的印表機未顯示於「*印表機型號*」對話中，表示您印表機型號的 PPD (PostScript 印表機描述) 檔案遺失 (請參閱第 23.3 節「安裝軟體」[403頁]以進一步瞭解 PPD 檔案的相關資訊)。使用「*新增 PPD 檔案至資料庫*」，從本地檔案系統、FTP 或 HTTP 伺服器新增 PPD 檔案。

從您的印表機廠商或驅動程式光碟，直接取得 PPD 檔案 (請參閱第 23.9.2 節「PostScript 印表機沒有可用的 PPD 檔案」[415頁]以得知詳細資訊)。PPD 檔案的另一個來源為`http://www.linuxprinting.org/`，也就是「Linux 列印資料庫」。從 linuxprinting.org 下載 PPD 檔案時，請記住這裡顯示的一律是最新的 Linux 支援狀態，不一定符合 SUSE Linux Enterprise。

以 YaST 選擇其他 PPD 檔案

許多印表機型號都有很多 PPD 檔案可用。設定印表機時，一般規則是 YaST 預設會使用標示為*建議*的檔案。若要取得印表機可用的 PPD 檔案清單，請在「*組態*」中選取「*PPD 檔案*」，並按一下「*編輯*」。請參閱圖形 23.1「印表機組態摘要」[406頁]。

一般無須變更 PPD 檔案，YaST 所選的 PPD 應可產生最佳結果。但舉例而言，若您希望彩色印表機進行黑白列印，最方便的方法就是使用不支援彩色列印的 PPD 檔案。若您列印圖形時，PostScript 印表機出現效能問題，從 PostScript PPD 檔案切換到 PCL PPD 檔案 (提供您印表機可使用的 PCL) 或許會有幫助。

23.4.2 以 YaST 設定網路印表機

網路印表機為自動獲得偵測，必須使用 YaST 印表機模組手動設定。視網路設定而定，您可以列印至印表機伺服器 (CUPS、LPD、SMB 或 IPX) 或直接至網路印表機 (最好透過 TCP)。如需在您的環境中設定網路印表機，請洽詢網路管理員。

過程 23.2　使用 YaST 設定網路印表機

1. 啟動 YaST 並選擇「*硬體*」>「*印表機*」來開啟「*印表機組態*」對話方塊。
2. 請按一下「*新增*」開啟「*印表機類型*」視窗。
3. 請選擇「*網路印表機*」開啟對話方塊，在其中指定網路管理員提供給您的詳細資訊。

23.5 網路印表機

網路印表機可支援各種通訊協定，有些甚至可同時支援。雖然大部分支援的通訊協定為標準的，部分製造商會因為測試系統無法正確地執行標準，或因為想要提供標準無法提供的特定功能，而擴充 (修改) 標準。然後製造商僅對少數作業系統提供驅動程式，以減少那些系統的困難。不幸地，他們很少提供 Linux 驅動程式。目前的情況是，您無法以每一個通訊協定均能在 Linux 中順暢執行的假設來行事。因此，您必須試驗各種選項以達到功能性組態。

重要：遠端存取設定

依預設，cupsd 僅監聽內部網路介面 (`localhost`)。設定 CUPS 網路伺服器時，需要調整 `/etc/cups/cupsd.conf` 中的 `Listen` 指示詞以監聽外部網路。

CUPS 支援 `socket`、`LPD`、`IPP` 和 `smb` 通訊協定。

插槽
: *Socket* 指不須先執行資料信號交換，而將資料傳送到網際網路插槽的連接。經常使用的插槽連接埠號碼為 `9100` 或 `35`。設備 URI (資源識別字串) 的語法為 socket://*`IP.of.the.printer`*:*`port`*，例如 `socket://192.168.2.202:9100/`。

LPD (行列式印表機精靈，Line Printer Daemon)
: 經過實驗的 LPD 通訊協定描述於 RFC 1179 中。在此通訊協定之下，部分工作相關資料 (如印表機佇列的 ID) 會在傳送實際列印資料之前傳送。因此，在設定資料傳輸的 LPD 通訊協定時，必須指定印表機佇列。不同印表機製造商的執行具有足夠彈性接受任何名稱做為印表機佇列。如有需要，印表機手冊應該會指出要使用的名稱。通常使用 LPT、LPT1、LP1或相似名稱。

LPD 佇列也可以在 CUPS 系統中不同 Linux 或 Unix 主機上設定。LPD 服務的連接埠號碼為 515。某個設備 URI 的範例為 lpd://192.168.2.202/LPT1。

IPP (網際網路列印通訊協定，Internet Printing Protocol)
IPP 是相對較新的 (1999) 通訊協定，以 HTTP 通訊協定為基礎。有了 IPP，可比使用其他通訊協定傳輸更多工作相關資料。CUPS 使用 IPP 進行內部資料傳輸。這是在兩個 CUPS 伺服器之間轉送佇列偏好的通訊協定。正確設定 IPP 必須要有列印佇列的名稱。IPP 的連接埠號碼為 631。設備 URI 的範例為 ipp://192.168.2.202/ps 和 ipp://192.168.2.202/printers/ps。

SMB (Windows 共享)
CUPS 也支援在連接到 Windows 共享的印表機上列印。此用途使用的通訊協定為 SMB。SMB 使用連接埠號碼 137、138 和 139。設備 URI 的範例為 smb://user:password@workgroup/smb.example.com/printer、smb://user:password@smb.example.com/printer 和 smb://smb.example.com/printer。

必須在設定組態之前決定印表機支援的通訊協定。如果製造商未提供所需資訊，可使用 nmap 指令 (nmap 套件) 來猜測通訊協定。nmap 會檢查主機上開啟的通訊埠。例如：

```
nmap -p 35,137-139,515,631,9100-10000 printerIP
```

23.5.1 以指令行工具設定 CUPS

除了使用 YaST 設定 CUPS 選項之外，設定網路印表機時，CUPS 可以由 lpadmin 和 lpoptions 之類的指令行工具進行設定。您需要包含後端 (如 USB 和參數 /dev/usb/lp0) 的設備 URI。例如，完整的 URI 可為 parallel:/dev/lp0 (連接到第一並列埠的印表機) 或 usb:/dev/usb/lp0 (第一個偵測到連接到 USB 埠的印表機)。

使用 lpadmin，CUPS 伺服器管理員可新增、移除或管理類別何列印佇列。若要新增印表機佇列，請使用下列語法：

```
lpadmin -p queue -v device-URI -P PPD-file -E
```

然後設備 (-v) 會變為可用 *queue* (-P)，使用指定的 PPD 檔案 (-P)。這表示如果要手動設定印表機，您必須知道 PPD 檔案以及設備名稱。

請勿使用 -E 做為第一選項。對於所有 CUPS 指令，第一個引數 -E 設定使用加密連接。若要啟用印表機，必須依照下列範例所示使用 -E：

```
lpadmin -p ps -v parallel:/dev/lp0 -P \
/usr/share/cups/model/Postscript.ppd.gz -E
```

下列範例是設定網路印表機：

```
lpadmin -p ps -v socket://192.168.2.202:9100/ -P \
/usr/share/cups/model/Postscript-level1.ppd.gz -E
```

如需 lpadmin 的更多選項，請參閱 lpadmin(1) 的線上文件。

在印表機設定期間，某些選項會設成預設。可針對每一個列印工作修改這些選項(視所使用的列印工具而定)。也可以使用 YaST 變更這些預設選項。使用指令行工具，可依下列方式設定預設選項：

1. 首先，列出所有選項：

   ```
   lpoptions -p queue -l
   ```

   範例：

   ```
   Resolution/Output Resolution: 150dpi *300dpi 600dpi
   ```

   啟用的預設選項前面會加上星號 (*)，用以識別。

2. 以 lpadmin 變更選項：

   ```
   lpadmin -p queue -o Resolution=600dpi
   ```

3. 檢查新設定：

   ```
   lpoptions -p queue -l

   Resolution/Output Resolution: 150dpi 300dpi *600dpi
   ```

當一般使用者執行 lpoptions 時，設定會寫至 ~/.lpoptions。然而，根部設定會會寫至 /etc/cups/lpoptions。

23.6 圖形列印介面

Xpp 和 KDE 程式 KPrinter 等工具備有圖形介面，以便選擇佇列，並可設定 PPD 檔案中可用的 CUPS 標準選項和印表機專用的選項。您甚至可以使用 KPrinter

做為非KDE應用程式的標準列印介面。在這些應用程式的列印對話方塊中，請指定 `kprinter` 或 `kprinter --stdin` 做為列印指令。要使用的指令根據應用程式的資料傳輸方式而有不同—只需看看哪一個指令可以啟動KPrinter即可。如果設定正確，應用程式會在發出列印工作時開啟 KPrinter 對話方塊，您就可以使用對話方塊來選取佇列，並設定其他列印選項。但前提是該應用程式自有的列印設定與 KPrinter 的列印設定不相衝突，而且列印選項啟用後只會經由 KPrinter 做變更。

23.7 由指令行開始列印

若要由指令行進行列印，請輸入 `lp -d` *佇列名稱 檔案名稱*；請以相對應的名稱來取代*佇列名稱*及*檔案名稱*。

有些應用程式有賴 `lp` 指令來進行列印。在此情況下，請在應用程式列印對話方塊中 (通常並未指定*檔案名稱*) 輸入正確的指令，例如 `lp -d` *佇列名稱*。

23.8 SUSE Linux Enterprise 中的特殊功能

CUPS 的許多功能已經可適用於 SUSE Linux Enterprise。此處涵蓋部份最重要的變更。

23.8.1 CUPS 與防火牆

執行 SUSE Linux Enterprise 的預設安裝後，SuSEfirewall2 隨即會啟用，且外部網路設備會設定為處於「外部區域」中，這會阻擋內送流量。使用 CUPS 時，必須調整這些預設的設定。如需SUSEfirewall2組態設定的詳細資訊，請參閱第 43.4 節「SUSEfirewall2」[749頁]。

CUPS 用戶端

CUPS用戶端通常在防火牆後網路中的一般工作站上執行。在此情況下，建議將外部網路設備設定為處於「內部區域」中，以便可從該網路中存取工作站。

CUPS 伺服器

如果 CUPS 伺服器位於受防火牆保護的網路中，則應將外部網路設備設定為處於防火牆的「內部區域」中。做為外部區域的組成部份時，TCP 和 UDP 連接埠 631 需要處於開啟狀態，以使 CUPS 伺服器在網路中可用。

23.8.2 CUPS 列印服務中的變更

BrowseAllow 和 BrowseDeny 的通用功能

BrowseAllow 和 BrowseDeny 設定的存取權限可套用於傳送給 cupsd 的所有類型套件。/etc/cups/cupsd.conf 中的預設設定值如下：

```
BrowseAllow @LOCAL
BrowseDeny All
```

和

```
<Location />
  Order Deny,Allow
  Deny From All
  Allow From 127.0.0.1
  Allow From 127.0.0.2
  Allow From @LOCAL
</Location>
```

利用此方法，僅 LOCAL 主機可存取 CUPS 伺服器上的 cupsd。LOCAL 主機是指屬於非 PPP 介面的 IP 位址之主機 (是沒有設定 IFF_POINTOPOINT 旗標的介面)，而其 IP 位址和 CUPS 伺服器屬於相同的網路。來自其他伺服器的封包會立刻被拒絕。

預設會啟用 cupsd

在標準安裝中，會自動啟用 cupsd，以便不需其他手動作業即可存取 CUPS 網路伺服器的佇列。章節「BrowseAllow 和 BrowseDeny 的通用功能」[412頁]中的各個項目是本功能的重要先決條件。因為如果未達到這些條件，則自動啟用 cupsd 將不夠安全。

23.8.3 各種套件中的 PPD 檔案

YaST 印表機組態僅使用安裝於系統上 /usr/share/cups/model/ 中的 PPD 檔案來設定 CUPS 的佇列。為了尋找適合印表機型號的 PPD 檔案，YaST 會將硬體偵測期間決定的廠商和型號與系統上 /usr/share/cups/model/ 中提供的所有 PPD 檔案內的廠商和型號相比較。基於此原因，YaST 印表機組態將從 PPD 檔案中取出的廠商和型號資訊產生資料庫。當您從廠商和型號清單中選取印表機時，會接收到相符廠商和型號的 PPD 檔案。

僅使用 PPD 檔案且不使用其他資訊來源的組態，好處在於 /usr/share/cups/model/ 中的 PPD 檔案可自由修改。YaST 印表機組態可辨識變更並重新產生廠商和型號資料庫。例如，如果您只有 PostScript 印表機，通常不需要 cups-drivers 套件中的 Foomatic PPD 檔案，或 cups-drivers-stp 套件中的 Gimp-Print PPD 檔案。您可以直接將 PostScript 印表機的 PPD 檔案複製到 /usr/share/cups/model/ (如果在 manufacturer-PPDs 套件中尚未存在)，以達到印表機的最佳組態。

cups 套件中的 CUPS PPD 檔案

cups 套件中的一般 PPD 檔案已經以 PostScript Level 1 和 Level 2 印表機適當的 Foomatic PPD 檔案補充。

- /usr/share/cups/model/Postscript-level1.ppd.gz
- /usr/share/cups/model/Postscript-level2.ppd.gz

cups-drivers 套件中的 PPD 檔案

一般情況下，Foomatic 印表機過濾器 foomatic-rip 會與非 PostScript 印表機的 Ghostscript 搭配使用。適當的 Foomatic PPD 檔案有 *NickName: ... Foomatic/Ghostscript driver 和 *cupsFilter: ... foomatic-rip 的項目。這些 PPD 檔案位於 cups-drivers 套件中。

如果擁有 *NickName: ... Foomatic ... (建議) 項目的 Foomatic PPD 檔案與印表機型號相符，而且 manufacturer-PPDs 套件中沒有其他更適當的 PPD 檔案，那麼 YaST 就偏向於使用 Foomatic PPD 檔案。

cups-drivers-stp 套件中 Gimp-Print PPD 的檔案

除了 foomatic-rip 之外，來自 Gimp-Print 的 CUPS 過濾器 rastertoprinter 也可以用在許多非 PostScript 印表機上。此過濾器和適合的 Gimp-Print PPD 檔案可在 cups-drivers-stp 套件中找到。Gimp-Print PPD 檔案位於 /usr/share/cups/model/stp/ 中，並含有項目 *NickName: ... CUPS+Gimp-Print 和 *cupsFilter: ... rastertoprinter。

位於 manufacturer-PPDs 套件中印表機製造廠商的 PPD 檔案

manufacturer-PPDs 套件包含具有充分自由授權的印表機製造廠商所發行的 PPD 檔案。PostScript 印表機應該以印表機製造廠商的適合 PPD 檔案來設定，因為有此檔案才可使用 PostScript 印表機的所有功能。如果符合下列條件，YaST 偏好來自 manufacturer-PPDs 套件的 PPD 檔案：

- 在硬體偵測期間決定的廠商和型號符合 manufacturer-PPDs 套件中 PPD 檔案內的廠商和型號。
- manufacturer-PPDs 套件的 PPD 檔案是此印表機型號唯一適合的 PPD 檔案，或者某個擁有 *NickName: ... Foomatic/Postscript (建議) 項目的 Foomatic PPD 也符合此印表機型號。

同時，在下列情況中，YaST 不使用任何來自 manufacturer-PPDs 套件的 PPD 檔案：

- 來自 manufacturer-PPDs 套件的 PPD 檔案不符合廠商和型號。如果 manufacturer-PPDs 套件對相似型號僅包含一個 PPD 檔案會發生此狀況，例如，型號系列中個別型號沒有各自的 PPD 檔案，但是在 PPD 檔案中以類似 Funprinter 1000 series 的格式指定型號名稱。
- 不建議使用該 Foomatic PostScript PPD 檔案。這可能是因為印表機型號無法在 PostScript 模式下有效率的操作，例如，此模式可能不信任該印表機，因為記憶體太少或印表機處理器太弱而使得速度太慢。此外，印表機可能預設不支援 PostScript，例如因為只有選用模組提供 PostScript。

如果來自 manufacturer-PPDs 套件的 PPD 檔案不適合 PostScript 印表機，但是 YaST 基於上述因素無法加以設定，請在 YaST 中手動選取對應印表機型號。

23.9 疑難排解

下列章節涵蓋印表機硬體和軟體最常遭遇的問題，以及解決或避免這些問題的方式。涵蓋的主題包含 GDI 印表機、PPD 檔案和連接埠組態，並討論了一般網路印表機問題、列印瑕疵、佇列處理。

23.9.1 沒有標準印表機語言模式支援的印表機

這些印表機不支援任何的一般印表機語言，且只有特殊的專屬控制序列才能處理。因此它們僅可在製造廠商針對其開發驅動程式的作業系統版本上使用。GDI 是 Microsoft* 為繪圖設備所開發的程式設計介面。製造廠商通常只提供 Windows 適用的驅動程式，而因為 Windows 驅動程式使用 GDI 介面，所以這些印表機也稱為 *GDI 印表機*。實際問題不在於程式設計介面，而是這些印表機僅可使用對應印表機型號的專用印表機語言處理。

部分 GDI 印表機可切換到 GDI 模式或標準印表機語言來操作。若有可能，請參閱印表機的手冊。某些型號需要特殊的 Windows 軟體來進行切換 (請注意，從 Windows 列印時，Windows 印表機驅動程式可能會一直將印表機切回 GDI 模式)。對於其他 GDI 印表機，則可以使用標準印表機語言的延伸模組。

部分製造廠商提供其 印表機的專用驅動程式。專用印表機驅動程式的壞處在於不保證可與安裝的列印系統搭配使用，且不保證適用於各種硬體平台。相反的，支援標準印表機語言的印表機不需依賴特殊的列印系統版本或特殊硬體平台。

除了不需花費時間嘗試使專用 Linux 驅動程式運作，也不需花費更多成本購買支援的印表機。如此可一次解決所有驅動程式問題、減少安裝與設定特殊驅動程式軟體以及取得列印系統中新開發所需的驅動程式更新的需要。

23.9.2 PostScript 印表機沒有可用的 PPD 檔案

如果 `manufacturer-PPDs` 套件不包含任何適用於 PostScript 印表機的 PPD 檔案，應該可以使用印表機製造廠商驅動程式光碟中的 PPD 檔案，或從印表機製造廠商的網頁下載適合的 PPD 檔案。

如果 PPD 檔案以壓縮保存檔 (.zip) 或自解壓縮保存檔 (.exe) 形式提供，請以 unzip 解壓縮。首先，檢閱 PPD 檔案的授權條款。然後，請使用 cupstestppd 公用程式來檢查 PPD fi檔案是否符合「Adobe PostScript Printer Description File Format Specification, version 4.3」(Adobe PostScript 印表機說明檔案格式規格，版本 4.3)。如果公用程式傳回「FAIL」，就表示 PPD 檔案非常嚴重，可能造成重大問題。應該要減少 cupstestppd 所報告的問題點。若有需要，請詢問印表機製造廠商以取得適合的 PPD 檔案。

23.9.3 並列埠

最安全的方法是將印表機直接連接到第一並列埠，並在 BIOS 中選取下列並列埠設定值：

- I/O 位址：378 (十六進位)
- 中斷：無關
- 模式：Normal、Spp 或 Output Only
- DMA：停用

如果沒有這些設定值，印表機無法在並列埠上定址，請依照 BIOS 中的設定值，以 0x378 的格式在 /etc/modprobe.conf 中明確輸入 I/O 位址。如果有兩個並列埠，I/O 位址分別設為 378 和 278 (十六進位)，請以 0x378,0x278 格式輸入。

如果沒有使用中斷 7，可以使用 範例 23.1「/etc/modprobe.conf：第一並列埠的中斷模式」[416頁] 中所示的項目啟用。在啟用中斷模式之前，請檢查檔案 /proc/interrupts 以瞭解哪些中斷已經在使用中。只會顯示目前正在使用中的中斷。這可能因為作用中的硬體元件而有變化。其他任何設備都不能使用並列埠的中斷。如果您不確定，請以 irq=none 使用輪詢模式。

範例 23.1 */etc/modprobe.conf：第一並列埠的中斷模式*

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378 irq=7
```

23.9.4 網路印表機連接方式

查明網路問題
: 將印表機直接連接到電腦。基於測試因素，請將印表機設為本地印表機。如果可以，問題便與網路相關。

檢查 TCP/IP 網路
: TCP/IP 網路和名稱解析必須可作用。

檢查遠端可存取性
: 依預設，cupsd 僅監聽內部網路介面 (`localhost`)。檢查 `/etc/cups/cupsd.conf` 中的 `Listen` 指示詞是否允許從外部網路存取。

```
Listen 192.168.2,*:631
```

檢查防火牆設定
: CUPS 伺服器必須處於內部防火牆區域中，或者，如果處於外部區域中，則必須能夠在 UDP 和 TCP 連接埠 631 上傳送與接收資料。

檢查遠端 `lpd`
: 使用以下指令來測試是否可在*主機*上建立到 `lpd` (連接埠 `515`) 的 TCP 連接：

```
netcat -z host 515 && echo ok || echo failed
```

如果無法建立到 `lpd` 的連接，可能是 `lpd` 不在作用中，或是有基本網路問題。

以使用者 `root` 的身份，使用以下指令來查詢 (可能很長) 遠端*主機*上*佇列*的狀態報告，假使對應的 `lpd` 在作用中且主機接受查詢：

```
echo -e "\004queue" \
| netcat -w 2 -p 722 host 515
```

如果 `lpd` 沒有回應，它可能不在作用中，或是有基本網路問題。如果 `lpd` 有回應，回應應該會顯示主機上的佇列為何無法列印。如果您收到像 範例 23.2「來自 lpd 的錯誤訊息」[418頁] 中的回應，問題可能是因為遠端 `lpd` 所造成。

範例 23.2　來自 lpd 的錯誤訊息

```
lpd: your host does not have line printer access
lpd: queue does not exist
printer: spooling disabled
printer: printing disabled
```

檢查遠端 cupsd
: 根據預設，CUPS 網路伺服器應該每隔三十秒在 UDP 連接埠 631 上廣播其佇列。同時，可使用以下指令來測試網路中是否有 CUPS 網路伺服器。

```
netcat -u -l -p 631 & PID=$! ; sleep 40 ; kill $PID
```

如果廣播 CUPS 網路伺服器存在，輸出將如 範例 23.3「來自 CUPS 網路伺服器的廣播」[418頁] 中所示。

範例 23.3　來自 CUPS 網路伺服器的廣播

```
ipp://192.168.2.202:631/printers/queue
```

► **zseries:** 請注意，根據預設 IBM System z 乙太網路設備不會接收廣播。◄

可使用以下指令來測試是否可建立到*主機*上 cupsd (連接埠 631) 的 TCP 連線：

```
netcat -z host 631 && echo ok || echo failed
```

如果無法建立與 cupsd 的連接，則表明 cupsd 可能不在作用中或可能有基本的網路問題。假設對應的 cupsd 在作用中，而且主機可以接受查詢，則 `lpstat -h host -l -t` 會傳回 *host* 上所有佇列的狀態報告 (可能很大)。

此指令可用來測試*主機*上的*佇列*是否可接受由單一換行字元組成的列印工作。應該不會印出任何資料。可能會退出一張空白頁。

```
echo -en "\r" \
| lp -d queue -h host
```

網路列印或列印伺服器盒疑難排解
: 在列印伺服器盒中執行的暫存序列器在執行大量列印工作時，有時會造成問題。因為這是列印伺服器盒中的暫存序列器所造成，您無法解決此問題。處理方式是，透過 TCP 插槽將印表機直接連接到列印伺服器盒，以規避列印伺服器盒中的暫存序列器。請參閱第 23.5 節「網路印表機」[408頁]。

利用此方法，可減少列印伺服器盒在不同資料格式之間的轉換問題 (TCP/IP 網路和本地印表機連接)。若要使用此方法，您必須知道列印伺服器盒上的 TCP 連接埠。如果印表機連接到列印伺服器盒且電源開啟，此 TCP 連接埠通常可在列印伺服器盒電源開啟一段時間之後，以 nmap 套件的 nmap 公用程式決定。例如，nmap *IP-address* 會傳送列印伺服器盒的以下輸出：

```
Port       State       Service
23/tcp     open        telnet
80/tcp     open        http
515/tcp    open        printer
631/tcp    open        cups
9100/tcp   open        jetdirect
```

此輸出表示連接到列印伺服器盒的印表機可透過連接埠 9100 上的 TCP 插槽定址。根據預設，nmap 僅會檢查幾個 /usr/share/nmap/nmap-services 中所列出一般熟知的連接埠。若要檢查所有可能的連接埠，請使用指令 nmap -p *from_port-to_port IP-address*。這可能會花費一些時間。如需詳細資訊，請參閱 nmap 的線上文件。

輸入以下指令

```
echo -en "\rHello\r\f" | netcat -w 1 IP-address port
cat file | netcat -w 1 IP-address port
```

將字元字串或檔案直接傳送到對應連接埠以測試印表機是否可在此連接埠上定址。

23.9.5 列印成品損毀而無錯誤訊息

對列印系統而言，在 CUPS 後端完成資料至接收者 (印表機) 的資料傳輸時，列印工作便完成。如果接收者的進一步處理失敗(例如，如果印表機無法列印印表機特定資料)，列印系統並不知道。如果印表機無法列印印表機特定資料，請選取更適用於印表機的不同 PPD 檔案。

23.9.6 停用佇列

如果到接收者的資料傳輸在數次嘗試之後完全失敗，CUPS 後端 (如 USB 或 socket) 會向列印系統報告錯誤 (向 cupsd)。後端會決定在報告資料傳輸失敗之前，是否要繼續嘗試以及要嘗試幾次是合理的。因為進一步的嘗試可能徒勞

無功，cupsd 會停用對應佇列的列印。除去問題的起因之後，系統管理者必須以指令 /usr/bin/enable 重新啟動列印。

23.9.7 CUPS 瀏覽：刪除列印工作

如果 CUPS 網路伺服器透過瀏覽向用戶端主機廣播它的佇列，而且在用戶端主機上有適合的本地 cupsd 在作用中，用戶端 cupsd 會從應用程式接收列印工作，並將它們轉送給伺服器上的 cupsd。當 cupsd 接收列印工作時，會被指定新的工作號碼。因此，用戶端主機上的工作號碼和伺服器上的工作號碼不同。因為列印工作通常會立刻轉送，所以無法以用戶端主機上的工作號碼來刪除，因為用戶端 cupsd 一旦將列印工作轉送給伺服器 cupsd，便認為列印工作已完成。

若要刪除伺服器上的列印工作，假使伺服器尚未完成列印工作 (即尚未將工作完全傳送至印表機)，請使用 lpstat -h cups.example.com -o 這類指令來決定伺服器上的工作編號。使用此工作號碼，伺服器上的列印工作便可刪除：

```
cancel -h cups.example.com queue-jobnnumber
```

23.9.8 損毀的列印工作與資料傳輸錯誤

如果您在列印程序中將印表機電源關閉再打開，或是關機再重新啟動電腦，列印工作仍然在佇列中，而且會繼續列印。必須以 cancel 將損毀的列印工作從佇列中移除。

如果列印工作損毀，或是主機和印表機之間的通訊發生問題，印表機會印出數頁含有不明字元的紙張，因為它無法正確地處理資料。若要處理這個問題，請遵循下列步驟：

1. 若要停止列印，請從噴墨印表機取出所有紙張，或是打開雷射印表機的紙匣。高品質的印表機會有按鈕可取消目前的列印成品。

2. 列印工作可能仍在佇列中，因為只有將工作完全傳送到印表機之後，才會移除。使用 lpstat -o 或 lpstat -h cups.example.com -o 檢查目前列印中的佇列。使用 cancel *queue-jobnumber* 或 cancel -h cups.example.com *queue-jobnumber* 刪除列印工作。

3 即使列印工作已從佇列刪除，部份資料可能仍會傳送到印表機。請檢查對應佇列的CUPS後端程序是否仍在執行中，並將它終止。例如，對於連接到並列埠的印表機，可使用指令 `fuser -k /dev/lp0` 來終止所有仍在存取印表機的程序 (更精確的說，就是並列埠)。

4 將印表機關閉一段時間以完全重設印表機。然後裝入紙張並開啟印表機電源。

23.9.9 除錯 CUPS 列印系統

使用以下標準程序以找出 CUPS 列印系統中的問題：

1 設定 `/etc/cups/cupsd.conf` 中的 `LogLevel debug`。

2 停止 `cupsd`。

3 移除 `/var/log/cups/error_log*` 以避免必須搜尋很大的記錄檔。

4 啟動 `cupsd`。

5 重覆造成問題的動作。

6 檢查 `/var/log/cups/error_log*` 中的訊息以辨識問題的起因。

使用 udev 進行動態核心設備管理

24

自 2.6 版之後，核心可以在執行系統中新增或移除大多數的任何設備。設備狀態 (已插入或移除設備) 中的變更必須傳播至使用者空間。在插入和探查設備時，必須立刻設定設備。特定設備的使用者必須收到任何狀態變更的通知。udev 會提供必要的基礎結構以便動態維護設備 /dev 目錄中的節點檔案和符號連結。udev 規則會提供將外部工具插入核心設備事件處理的方法。這項工具可用來自定 udev 設備處理，例如，新增要執行的特定程序檔來作為核心設備處理的一部份，或是在設備處理過程中要求並輸入其他資料來進行分析。

24.1 /dev 目錄

/dev 中的設備節點可用來存取對應的核心設備。透過 udev，/dev 目錄會反映核心的目前狀態。每個核心設備都有一個對應的設備檔案。如果設備與系統的連接中斷，該設備節點就會遭到移除。

/dev 目錄內容會保存在暫存檔系統中，而且所有檔案都會在每次系統開機時重新建立。特別經過手動建立或變更的檔案在重新開機後都不會存在。無論可存放於 /lib/udev/devices 目錄的對應核心設備狀態為何，靜態檔案和目錄都必須存在於 /dev 目錄中。在系統啟動時，該目錄內容將複製到 /dev 目錄，並具備與 /lib/udev/devices 中檔案相同的擁有權和許可權。

24.2 核心 uevent 和 udev

sysfs 檔案系統會輸出必要的設備資訊。每個核心已偵測和啟始化的設備，都會建立包含其設備名稱的目錄。其中會包含設備特定的屬性內容。每次新增或移除設備時，核心都會傳送 uevent 來通知 udev 此變更狀況。

udev 精靈會在啟動時讀取和分析一次 `/etc/udev/rules.d/*.rules` 檔案的所有指定規則，並將其保存在記憶體中。精靈會在規則檔案遭到變更、新增或移除時收到一個事件，並更新出現在記憶體內部的規則。

每個收到的事件都將與提供的規則集合進行比對。這些規則可新增或變更事件環境識別碼、要求要建立設備節點的特定名稱、新增指向該節點的符號連結，或是新增要在設備節點建立後執行的程式。驅動程式核心 uevents 會從核心網路連結插槽接收。

24.3 驅動程式、核心模組和設備

核心匯流排驅動程式會查探設備。核心會為每個偵測到的設備建立內部設備結構，而驅動程式核心會向 udev 精靈傳送 uevent。匯流排設備會以特殊格式的 ID 識別本身，表明其為何種設備。通常這些 ID 會包含廠商和產品 ID，以及其他子系統特定值。每個匯流排都會指定自己的 ID 配置，即所謂的 `MODALIAS`。核心會接收這些設備資訊、組織 `MODALIAS` ID 字串，並隨事件傳送該字串。例如，USB 滑鼠的 ID 字串將如下所示：

```
MODALIAS=usb:v046DpC03Ed2000dc00dsc00dp00ic03isc01ip02
```

每個設備驅動程式都包含有設備可處理的已知別名清單。清單會包含在核心模組檔案本身。depmod 程式會讀取 ID 清單，並且為目前所有可用模組在核心的 `/lib/modules` 目錄中建立 `modules.alias` 檔案。透過此基礎結構，模組載入方式就會像在每次出現帶有 `MODALIAS` 識別碼的事件時呼叫 `modprobe` 一樣容易。如果是呼叫 `modprobe $MODALIAS`，此次呼叫就會比對設備的已組織設備別名和模組指定別名。如果有找到符合項目，該模組就可載入。這些動作都是由 udev 觸發，而且是自動發生。

24.4 開機和初始設備設定

在執行 udev 精靈之前於開機程序期間發生的設備事件都會遺失，這是因為處理這些事件的基礎結構是執行於根檔案系統中，而在該階段進行時無法使用。核心將會為 sysfs 檔案系統中的每個設備提供 `uevent` 檔案，以彌補該損失。使用 `add` 寫入該檔案，核心便可重新傳送與開機期間所遺失的相同事件。負責 `/sys` 中所有 `uevent` 檔案的簡易迴圈，可以再次觸發所有事件，建立設備節點並執行設備設定。

例如，用於開機期間的 USB 滑鼠可能無法由最初的開機邏輯啟始，這是因為當時並無法使用驅動程式。設備探查事件遺失，而且無法找到設備的核心模組。udev 只需在可以使用根目錄檔案系統之後要求核心提供所有設備事件，這樣 USB 滑鼠設備的事件就可再次執行，因此並不需要手動搜尋可能連結的設備。現在，它會在已裝載根目錄檔案系統中找到核心模組，並讓 USB 滑鼠完成啟始化。

從使用者空間的角度，執行期間的設備冷插拔 (ColdPlud) 順序和設備探查並沒有明顯的不同。這兩種情況都會使用相同規則來進行比對，而且會執行相同的設定程式。

24.5 除錯 udev 事件

`udevmonitor` 程式可用來視覺化驅動程式核心事件，以及 udev 事件程序的時間。

```
UEVENT[1132632714.285362] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2
UEVENT[1132632714.288166] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2/2-2:1.0
UEVENT[1132632714.309485] add@/class/input/input6
UEVENT[1132632714.309511] add@/class/input/input6/mouse2
UEVENT[1132632714.309524] add@/class/usb_device/usbdev2.12
UDEV  [1132632714.348966] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2
UDEV  [1132632714.420947] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2/2-2:1.0
UDEV  [1132632714.427298] add@/class/input/input6
UDEV  [1132632714.434223] add@/class/usb_device/usbdev2.12
UDEV  [1132632714.439934] add@/class/input/input6/mouse2
```

`UEVENT` 行會顯示核心已透過網路連結傳送的事件。`UDEV` 行會顯示已完成的 udev 事件處理常式。列印時間是百萬分之一秒。介於 `UEVENT` 和 `UDEV` 之間的時間是指 udev 處理此事件所耗費的時間，或者是 udev 精靈延遲執行以便此事

件能與執行中相關事件同步的時間。例如，硬碟分割區的事件永遠會等待主要磁碟設備事件完成，因為分割區事件可能與主要磁碟事件向硬碟查詢的資料有關。

`udevmonitor --env` 會顯示完整的事件環境：

```
UDEV  [1132633002.937243] add@/class/input/input7
UDEV_LOG=3
ACTION=add
DEVPATH=/class/input/input7
SUBSYSTEM=input
SEQNUM=1043
PHYSDEVPATH=/devices/pci0000:00/0000:00:1d.1/usb2/2-2/2-2:1.0
PHYSDEVBUS=usb
PHYSDEVDRIVER=usbhid
PRODUCT=3/46d/c03e/2000
NAME="Logitech USB-PS/2 Optical Mouse"
PHYS="usb-0000:00:1d.1-2/input0"
UNIQ=""
EV=7
KEY=70000 0 0 0 0 0 0 0
REL=103
```

udev 也會將訊息傳送到 syslog。控制哪個訊息要傳送到 syslog 的預設 syslog 優先順序，會在 udev 組態檔 `/etc/udev/udev.conf` 中指定。執行精靈的記錄順序可使用 `udevcontrol log_priority=`*`level/number`* 來變更。

24.6 透過 udev 規則影響核心設備事件處理

udev 規則可以比對核心新增至事件本身的任何內容，或者任何由核心輸出到 `sysfs` 的資訊。規則也可向外部程式要求其他資訊。每個事件都會與所有指定規則進行比對。所有規則都位於 `/etc/udev/rules.d` 目錄中。

規則檔案中的每一行都包含至少一個金鑰值組合。金鑰類型共有兩種，包括比對和指定金鑰。當所有比對金鑰都與指定值相符時就會套用規則，而該指定值就會指定給指定金鑰。相符的規則可以指定設備節點名稱、新增指向該節點的符號連結，或是執行指定程式成為事件處理的一部份。如果找不到任何符合規則，就會使用預設的設備節點名稱來建立設備節點。規則語法和針對比對或輸出資料所提供的金鑰將描述於 udev man 頁面。

24.7 永久設備命名

動態設備目錄和 udev 規則基礎結構可以為所有磁碟設備提供固定名稱，無論其辨識順序或設定使用的連接為何。核心所建立的每個適當區塊設備，都會採用針對特定匯流排、磁碟類型或檔案系統所設計的工具進行檢查。udev 會根據動態核心指定設備節點名稱，維護指向該設備的永久符號連結類別：

```
/dev/disk
|-- by-id
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B -> ../../sda
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part1 -> ../../sda1
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part6 -> ../../sda6
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part7 -> ../../sda7
|   |-- usb-Generic_STORAGE_DEVICE_02773 -> ../../sdd
|   `-- usb-Generic_STORAGE_DEVICE_02773-part1 -> ../../sdd1
|-- by-label
|   |-- Photos -> ../../sdd1
|   |-- SUSE10 -> ../../sda7
|   `-- devel -> ../../sda6
|-- by-path
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0 -> ../../sda
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part1 -> ../../sda1
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part6 -> ../../sda6
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part7 -> ../../sda7
|   |-- pci-0000:00:1f.2-scsi-1:0:0:0 -> ../../sr0
|   |-- usb-02773:0:0:2 -> ../../sdd
|   |-- usb-02773:0:0:2-part1 -> ../../sdd1
`-- by-uuid
    |-- 159a47a4-e6e6-40be-a757-a629991479ae -> ../../sda7
    |-- 3e999973-00c9-4917-9442-b7633bd95b9e -> ../../sda6
    `-- 4210-8F8C -> ../../sdd1
```

24.8 取代的熱插拔 (Hotplug) 套件

之前使用的熱插拔 (Hotplug) 套件，已經全部由 udev 和 udev 相關的核心基礎結構取代。以下原先屬於舊有熱插拔 (Hotplug) 基礎結構的部分，已經廢除不用或是由 udev 取代其功能：

`/etc/hotplug/*.agent`
: 不再需要，或已移到 `/lib/udev`

`/etc/hotplug/*.rc`
: 替代成 `/sys/*/uevent` 觸發

`/etc/hotplug/blacklist`
: 替代成 `modprobe.conf` 的 `blacklist`選項

`/etc/dev.d/*`
: 替代成 udev 規則 `RUN` 金鑰

`/etc/hotplug.d/*`
: 替代成 udev 規則 `RUN` 金鑰

`/sbin/hotplug`
: 替代成傾聽網路連結的 udevd；僅於初始 RAM 檔案系統中使用，當根目錄檔案系統可進行裝載時便會停用。

`/dev/*`
: 替代成 `/lib/udev/devices/*` 中動態 udev 和靜態內容

下列檔案和目錄包含了 udev 基礎結構的重要元件：

`/etc/udev/udev.conf`
: 主要 udev 組態檔

`/etc/udev/rules.d/*`
: udev 事件符合規則

`/lib/udev/devices/*`
: 靜態 `/dev` 內容

`/lib/udev/*`
: 從 udev 規則中呼叫的協助程式

24.9 如需更多資訊

如需關於 udev 基礎結構的詳細資訊，請參閱下列 man 頁面：

udev
: 關於 udev、金鑰、規則和其他重要組態問題的一般資訊。

udevinfo
: udevinfo 可用來查詢 udev 資料庫中的設備資訊。

udevd
: 關於 udev 事件管理精靈的資訊。

udevmonitor
: udevmonitor 可將核心和 udev 事件順序列印至主控台。此工具主要用於儲存。

25 Linux 的檔案系統

SUSE Linux Enterprise® 隨附多種檔案系統，包括 ReiserFS、Ext2、Ext3 與 XFS，您可在安裝時選取。各種檔案系統均有其優缺點，可適用於不同情況。為了滿足高效能叢集案例的要求，SUSE Linux Enterprise Server 包含了 OCFS2 (Oracle Cluster File System 2)。

25.1 術語

中繼資料
: 一種檔案系統–可確保磁碟上的所有資料都能適當組織而且可以存取的內部資料結構。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，這也是檔案系統顯示不同效能特性的部分原因。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
: Inode 包含檔案的各種資訊，包括大小、連結數目、實際存放檔案內容的磁碟區塊指標、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，包含檔案系統會在其中儲存對檔案系統的中繼資料進行何種變更的一種記錄。日誌處理大大降低 Linux 的復原時間，因為它放棄系統啟動時檢查整個檔案系統的冗長搜尋程序。而是只重複檢查日誌。

25.2 Linux 的主要檔案系統

不像兩、三年前，選擇 Linux 的檔案系統不再是幾秒鐘的事了 (Ext2 或 ReiserFS?)。從 2.4 開始的核心，提供各種檔案系統可供選擇。下面是這些檔案系統基本工作方式及其優點的綜覽。

世上沒有符合所有應用程式類型的檔案系統，這一點很重要，要牢記於心。每一種檔案系統都有自己特殊的優、缺點，必須考慮在內。但是即使是最頂級的檔案系統，也無法取代合理的備份策略。

本章內容所提到的*資料整合性*和*資料一致性*兩個詞彙，並不是指使用者空間資料的一致性 (應用程式寫入檔案中的資料)。這項資料是否一致必須由應用程式本身控制。

重要：設定檔案系統

除非在本章節中提及，否則設定或變更分割以及檔案系統的一切步驟，都可以使用 YaST 來執行。

25.2.1 ReiserFS

正式說來，2.4 核心版本的重要功能之一，ReiserFS，其 6.4 版早已被 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。它的關鍵價值是較佳的磁碟空間利用、較好的磁碟存取效能以及快速損毀復原。

詳細說明 ReiserFS 的長處，包括：

更佳的磁碟空間利用
: 在 ReiserFS，所有資料按照稱為 B^*-平衡樹的結構整理的。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 也支援資料日誌，以及 ordered 資料模式，此模式類似 Ext3 章節中 (第 25.2.3 節「Ext3」[434頁]) 描述的概念。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。

25.2.2 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。有了日誌檔案系統的建立以及其驚人的快速回復時間，Ext2 就變得不那麼重要了。

簡短的 Ext2 功能摘要可以協助瞭解為什麼它過去是 (在某些領域依然是) 很多 Linux 使用者最喜愛的 Linux 檔案系統。

穩固性
: 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。在檔案系統無法完全取消裝載而導致系統中斷後，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
: 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。它的可靠性和穩固性，巧妙地結合了日誌檔案系統的優點。

25.2.3 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：

可輕易從 Ext2 升級，並具有很高的可靠性
: 因為 Ext3 是以 Ext2 程式碼為基礎，而且會共用它的磁碟上 (On-Disk) 格式和中繼資料格式，所以從 Ext2 升級至 Ext3 十分簡單。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。它也非常安全，因為重新建立整個檔案系統，不保證萬無一失。思考一下現有 Ext2 系統等候升級至日誌檔案系統的數量，您就可以輕易地發現為什麼 Ext3 對很多系統管理員具有一定的重要性。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永遠會保存在一致狀態，但是同樣地無法自動保證檔案系統資料本身。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。除非您另有其他指定，否則 Ext3 是預設與 `data=ordered` 一起執行的。

25.2.4 將 Ext2 檔案系統轉換成 Ext3

若要將 Ext2 檔案系統轉換成 Ext3，請依照下列程序執行：

1. 以 `root` 身份執行 `tune2fs -j` 來建立 Ext3 日誌。這樣會以預設參數建立 Ext3 日誌。

若要由自己決定日誌大小以及它所在的位置，請執行 `tune2fs -J`，而不要同時使用需要的日誌選項 `size=` 和 `device=`。如需更多 tune2fs 程式的詳細資訊，請參閱 tune2fs man 頁面。

2. 若要確定 Ext3 檔案系統會被視為 Ext3 檔案系統，請以 `root` 身份編輯檔案 `/etc/fstab`，將指定給相應分割區的檔案系統類型從 `ext2` 變更成 `ext3`。完成的變更會在下次啟動時生效。

3. 若要為設定為 Ext3 分割區的根目錄檔案系統進行開機，請在 `initrd` 中包含 (Include) 模組 `ext3` 和 `jbd`。若要執行這個動作，請以 `root` 身份編輯 `/etc/sysconfig/kernel`，將 `ext3` 和 `jbd` 新增到 `INITRD_MODULES` 變數中。儲存這些變更之後，再執行 `mkinitrd` 指令。這樣就可以建置新的 initrd，並準備使用。

25.2.5 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 背後的想法是建立高效能 64 位元日誌檔案系統，以符合今日嚴格的計算挑戰。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 在專注於中繼資料整合性，不重視資料整合性。

下面針對 XFS 重要特性的快速回顧可為您說明，為什麼它會在高階計算方面成為其他日誌檔案系統的重要競爭對手。

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。這項特性就是 XFS 具有優良擴充性的關鍵。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^+ 樹大幅改善了 XFS 的效能和擴充性。XFS 會使用*延遲配置*。它會將程序分成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 仍然沒有決定資料到底要儲存在什麼地方 (提到檔案系統區塊的時候)。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，因為當決定 XFS 決

定實際儲存位置時，它可能已經過時了。因此 XFS 會增加寫入效能並降低檔案系統零散化。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。

25.2.6 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配檔案系統，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 (NM)。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。為避免多個節點直接存取檔案系統而產生混亂，OCFS2 還包含了鎖定管理員 DLM (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

如需更多關於 OCFS2 的詳細資訊，請參閱第 14 章「*Oracle Cluster File System 2*」[263頁]。

25.3 其他支援的檔案系統

表格 25.1「Linux 的檔案系統類型」[437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 25.1 *Linux 的檔案系統類型*

`cramfs`	*壓縮的ROM 檔案系統*：ROM 的一種壓縮唯讀檔案系統。
`hpfs`	*高效能檔案系統*：IBM OS/2 標準檔案系統，僅在唯讀模式中支援。
`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`	透過網路掛載 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的任何機器，而且可以經由授權從網路存取。
`smbfs`	*有些產品 (例如 Windows) 會使用*伺服器訊息區塊，透過網路來存取檔案。
`sysv`	用於 SCO UNIX、Xenix 和 Coherent (個人電腦的商用 UNIX 系統)。
`ufs`	用於 BSD、SunOS 和 NeXTSTEP。僅支援唯讀模式。
`umsdos`	*UNIX on MSDOS*：套用在一般 `fat` 檔案系統之上，建立特殊檔案來達到 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。

25.4 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會提供 LFS 支援，讓您用來執行高階運算。表格 25.2「檔案系統的大小上限 (磁碟上格式)」[438頁] 提供 Linux 檔案和檔案系統目前限制的綜覽。

表格 25.2 *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：Linux 核心限制

表格 25.2「檔案系統的大小上限 (磁碟上格式)」[438頁]會說明磁碟上 (On-Disk) 格式的限制。2.6 核心會強制檔案大小和其處理的檔案系統依循特定大小限制。如下：

檔案大小
: 在 32 位元系統，檔案不得超過 2 TB (2^{41} 位元組)。

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

25.5 如需更多資訊

請造訪上述每種檔案系統計畫維護的專屬網頁，找出計畫相關的郵件清單資訊、進一步文件以及常見問題。

- http://e2fsprogs.sourceforge.net/
- http://www.zipworld.com.au/~akpm/linux/ext3/
- http://chichkin_i.zelnet.ru/namesys/
- http://oss.sgi.com/projects/xfs/
- http://oss.oracle.com/projects/ocfs2/

關於 Linux 系統深入的多層級教學課程，請造訪 *IBM developerWorks*，網址是：http://www-106.ibm.com/developerworks/library/l-fs.html。您可在 Wikipedia 專案 http://en.wikipedia.org/wiki/Comparison_of_file_systems#Comparison 中找到一個非常有深度的檔案系統比較 (不只有 Linux 檔案系統)。

26 X Window System

X Window System (X11) 是 UNIX 中既成現實標準的圖形化使用者介面。X 採網路結構，可讓應用程式在一個主機上啟動而在透過任何種類的網路 (LAN 或網際網路) 連接的其他主機上顯示。本章說明 X Window System 環境的安裝與最佳化，並提供有關在 SUSE Linux Enterprise® 中使用字型的背景資訊。

提示：IBM System z：設定圖形使用者介面

X.Org 並不支援 IBM System z 的輸入與輸出設備。因此，本節說明的組態程序皆不適用。如需 IBM System z 的更多相關資訊，請參閱第 8.6 節「網路設備」[150頁]。

26.1 手動設定 X Window System

依照預設，X Window System 是以 SaX2 介面設定的，如第 8.14 節「SaX2」[175頁]中所述。但也可手動編輯組態檔進行設定。

警告：錯誤的 X 組態可能會損壞您的硬體

進行 X Window System 組態時請務必小心。在完成組態之前，絕對不可以啟動 X Window System。錯誤設定的系統會對硬體造成無法挽回的損害 (特別是對固定頻率的監視器)。本書與 SUSE Linux Enterprise 製作者對所致損害不負任何責任。本資訊是仔細研究的結果，但不保證提及的所有方法都正確以及不會損害您的硬體。

指令 sax2 可建立 /etc/X11/xorg.conf 檔案。這是 X Window System 的主要組態檔。請在此處找出和您的圖形卡、滑鼠以及監視器相關的所有設定。

重要：使用 X -configure

若之前嘗試 SUSE Linux Enterprise 的 SaX2 失敗的話，請使用 X -configure 設定您的 X 設定。若您的設定包含專屬的二進位檔驅動程式，則 X -configure 無法工作。

下列段落會描述組態檔 /etc/X11/xorg.conf 的結構。它由多個段落組成，每一個都會處理組態的某個層面。每一個段落以關鍵字 Section <designation> 開始，並以 EndSection 結束。下列轉換會套用到所有部份：

```
Section "designation"
  entry 1
  entry 2
  entry n
EndSection
```

可用的段落類型列出在 表格 26.1「/etc/X11/xorg.conf 中的段落」[442頁] 中。

表格 26.1 */etc/X11/xorg.conf 中的段落*

類型	代表意義
Files	用於字型與 RGB 顏色表的路徑。
ServerFlags	伺服器行為的一般切換器。
Module	伺服器應載入的模組清單。
InputDevice	輸入設備，例如，鍵盤和特殊輸入設備 (觸控板、搖桿等) 都是在這個段落設定。這個段落含有 Driver 的重要參數以及定義 Protocol 與 Device 的選項。您每個連接到電腦的設備通常有一個 InputDevice 段落。
Monitor	所用的伺服器。此段落的重要元素為：Identifier (稍後的 Screen 定義中將會參考它)、重新整理速率 VertRefresh 以及同步頻率限制 (HorizSync 和

類型	代表意義
	VertRefresh)。設定值以 MHz、kHz 和 Hz 提供。通常，伺服器會拒絕和監視器規格不對應的模式行。這樣可防止不小心將太高的頻率送往顯示器。
Modes	特定螢幕解析度的模式行參數。這些參數可以由 SaX2 依據使用者給定的值計算，通常不需要變更。如果您要連接固定頻率監視器，現在可以手動變更。在 /usr/share/doc/howto/en/html/XFree86-Video-Timings-HOWTO 目錄 (howtoenh 套件中提供) 中的 HOWTO 檔案中，可以找到各個數值之意義的詳細資料。
Device	特定圖形卡。它是以本身描述名稱敘述。
Screen	這個段落和 Monitor 以及 Device 一起構成 X.Org 所有必要的設定。在 Display 子段落，請指定虛擬螢幕的大小 (Virtual)、ViewPort，以及該螢幕使用的 Modes。
ServerLayout	單螢幕顯示或多螢幕顯示組態的配置。這個段落連結輸入設備 InputDevice 和顯示設備 Screen。
DRI	提供資訊給「直接算圖基礎結構 (DRI)」。

Monitor、Device，與 Screen 將在以下詳細說明。如需其他段落的進一步資訊，請參閱 X.Org 和 xorg.conf 的手冊頁。

xorg.conf 中可能有多個不同的 Monitor 和 Device 段落。也可能有多個 Screen 段落。ServerLayout 段落決定使用這些段落中的哪些段落。

26.1.1 Screen 段落

Screen 段落是由 monitor 和 device 段落組成，已決定要使用的解析度和色彩深度。Screen 段落可能如同 範例 26.1「/etc/X11/xorg.conf 檔的螢幕段落」[444頁] 一般。

範例 26.1 *`/etc/X11/xorg.conf` 檔的螢幕段落*

```
Section "Screen"❶
  DefaultDepth  16❷
  SubSection "Display"❸
    Depth       16❹
    Modes       "1152x864" "1024x768" "800x600"❺
    Virtual     1152x864❻
  EndSubSection
  SubSection "Display"
    Depth       24
    Modes       "1280x1024"
  EndSubSection
  SubSection "Display"
    Depth       32
    Modes "640x480"
  EndSubSection
  SubSection "Display"
    Depth        8
    Modes       "1280x1024"
  EndSubSection
  Device        "Device[0]"
  Identifier    "Screen[0]"❼
  Monitor       "Monitor[0]"
EndSection
```

❶ `Section` 決定段落類型，在此情況下為 `Screen`。

❷ `DefaultDepth` 決定若無指定色彩深度的話，預設使用的色彩深度。

❸ 會為每種色彩深度指定不同的 `Display` 子段落。

❹ `Depth` 決定與該組 `Display` 設定搭配使用的色彩深度。可能的值為 `8`、`15`、`16`、`24` 與 `32`，雖然並不是每個值都受所有 X 伺服器模組或解析度支援。

❺ `Modes` 段落包括可能螢幕解析度的清單。X 伺服器將由左而右檢查這個清單。對於每一個解析度，X 伺服器會在 `Modes` 段落搜尋適合的 `Modeline`。`Modeline` 同時由顯示器與圖形卡的功能決定。`Monitor` 設定值將決定 `Modeline` 的結果。

第一個找到的解析度是 `Default mode`。按 Ctrl + Alt + + (數字鍵盤)，可以切換到清單右邊的下一個解析度。按 Ctrl + Alt + – (數字鍵盤) 可切換到上一個解析度。您可以在 X 執行中變更解析度。

❻ `Display` 子段落最後一行的 `Depth 16` 指虛擬螢幕大小。虛擬螢幕可能的最大大小由圖形卡上安裝的記憶體數量以及所要的色彩深度決定，而不是顯示器的最大解析度。如果省略此行，虛擬解析度便是實體解析度。因

為目前的圖形卡都附有大量的視訊記憶體，您可以建立非常大的虛擬桌面。不過，如果將大部份的視訊記憶體用於虛擬桌面，將可能無法使用 3D 功能。例如，如果卡上有 16 MB 的視訊 RAM，虛擬螢幕最大可使用 4096x4096 像素，8 位元色彩探度。不過，不建議將所有記憶體用於虛擬螢幕，特別是加速卡，因為卡的記憶體還要用於各種字型與圖形的快取。

❼ `Identifier` 行 (此處為 `Screen[0]`) 對這個段落提供一個定義的名稱，以便對下列 `ServerLayout` 段落提供唯一性參照。`Device` 和 `Monitor` 行指定屬於這個定義的圖形卡和顯示器。它們透過本身對應的名稱或*識別碼*和 `Device` 以及 `Monitor` 段落連結。以下詳細討論這些段落。

26.1.2 Device 段落

Device 段落描述特定的圖形卡。`xorg.conf` 中的設備項目數沒有限制，但每一個設的名稱要使用關鍵字 `Identifier` （識別碼）來區別。若您安裝多張圖形卡，段落將只依序編號。第一個稱為 `Device[0]`，第二個稱為 `Device[1]`，以下類推。以下檔案顯示一個使用 Matrox Millennium PCI 圖形卡的電腦 (如 SaX2 所設定)，其 `Device` 段落的例外情況：

```
Section "Device"
  BoardName      "MGA2064W"
  BusID          "0:19:0"❶
  Driver         "mga"❷
  Identifier     "Device[0]"
  VendorName     "Matrox"
  Option         "sw_cursor"
EndSection
```

❶ `BusID` 表示要在其中安裝圖形卡的 PCI 或 AGP 插槽。這個值和指令 `lspci` 顯示的 ID 相符。X 伺服器需要十進位格式的詳細資料，但 `lspci` 以十六進位格式顯示這些值。SaX2 會自動偵測 `BusID` 的值。

❷ SaX2 會自動偵測 `Driver` 的值，並指定您的圖形卡要使用哪個驅動程式。如果是 Matrox Millennium 圖形卡，驅動程式模組稱為 `mga`。X 伺服器會在定義於 `drivers` 子目錄的 `Files` 段落中的 `ModulePath` 中搜尋。在標準安裝中，為 `/usr/X11R6/lib/modules/drivers` 或 `/usr/X11R6/lib64/modules/drivers`目錄。名稱會附加 `_drv.o`，因此，如果是 `mga` 驅動程式，將載入驅動動程式檔案 `mga_drv.o`。

X 伺服器或驅動程式的行為可以透過附加的選項來操作。Device 段落中的選項 `sw_cursor` 便是一個範例。它可以停用硬體滑鼠游標並描述使用軟體的滑鼠指

標。視驅動程式而定，各有不同的選項，可以在 /usr/share/doc/*package_name*目錄內的驅動程式模組描述檔案中找到。通常情況下，也可以在手冊頁面 (man xorg.conf、man X.Org 和 man 4 chips) 找到有效的選項。

如果圖形卡有多個視訊連接器，可將此卡的各個不同設備設定為單個檢視窗。請使用 SaX2 以此方式設定圖形介面。

26.1.3 Monitor 和 Modes 段落

和 Device 段落類似，Monitor 和 Modes 段落分別描述一個顯示器。組態檔 /etc/X11/xorg.conf 可以含有沒有個數限制的 Monitor 段落。每個 Monitor 段落都會參考帶有 UseModes 行 (若可用) 的 Modes 段落。如果沒有可用於 Monitor 段落的 Modes 段落，X 伺服器將從一般同步值計算適當的值。ServerLayout 段落指定相關的 Monitor 段落。

顯示器定義應該由有經驗的使用者進行設定。模式行是 Monitor 段落的重要組成部份。模式行可設定對應解析度的水平與垂直計時。顯示器的內容，特別是容許頻率，儲存在 Monitor 段落中。

警告

除非您深入瞭解監視器與圖形卡功能，否則請勿變更模式行，因為這樣做可能會嚴重損壞您的監視器。

嘗試開發自己的監視器描述者，應十分熟悉 /usr/X11R6/lib/X11/doc/ 中的文件 (必須安裝 xorg-x11-doc 套件)。

在今天，需要手動指定模式行的的情況非常少見。如果您使用先進的多頻顯示器，依規則，X 伺服器可以透過 DDC 直接從顯示器讀取容許頻率以及最佳解析度值，如 SaX2 組態段落中的說明那樣。如果因某種原因而不適用，請使用 X 伺服器內附的一種 VESA 模式。該模式幾乎對所有的圖形卡與監視器組合都可起作用。

26.2 安裝與設定字型

SUSE Linux Enterprise 上安裝其他字型非常簡單。只要將字型複製到位於 X11 字型路徑中的任何目錄即可 (請參閱第 26.2.1 節「X11 核心字型」[448頁])。安裝目錄應為 `/etc/fonts/fonts.conf` (請參閱第 26.2.2 節「Xft」[449頁]) 中所設目錄的子目錄，或包含於檔案 `/etc/fonts/suse-font-dirs.conf` 中。

以下內容摘自 `/etc/fonts/suse-font-dirs.conf`。該檔案已包含在組態中，因為它連結至 `/etc/fonts/fonts.conf` 包含的 `/etc/fonts/conf.d` 目錄。在此目錄中，所有以兩位數值開頭的檔案或符號連結將由 fontconfig 載入。關於此功能的詳細說明，請參閱 `/etc/fonts/conf.d/README`。

```
<!-- Font directory list -->
<dir>/usr/share/fonts</dir>
<dir>/usr/X11R6/lib/X11/fonts</dir>
<dir>/opt/kde3/share/fonts</dir>
<dir>/usr/local/share/fonts</dir>
<dir>~/.fonts</dir>
<dir>~/.fonts/kde-override</dir>
<include ignore_missing="yes">suse-font-dirs.conf</include>
```

`/etc/fonts/suse-font-dirs.conf` 是自動產生的，以將包含於 (多半是協力廠商) 應用程式，如 OpenOffice.org、Java 或 Adobe Acrobat Reader 中的字型拉進來。某些 `/etc/fonts/suse-font-dirs.conf` 中的典型項目看起來像這樣：

```
<dir>/usr/lib/ooo-2.0/share/fonts</dir>
<dir>/usr/lib/ooo-2.0/share/fonts/truetype</dir>
<dir>/usr/lib/jvm/java-1.5.0-sun-1.5.0_update10/jre/lib/fonts</dir>
<dir>/usr/X11R6/lib/Acrobat7/Resource/Font</dir>
<dir>/usr/X11R6/lib/Acrobat7/Resource/Font/PFM</dir>
```

若要在整個系統上安裝其他字型，請以 `root` 身份手動將字型檔案複製到合適的目錄，如 `/usr/share/fonts/truetype`。此外，這個作業可以透過「KDE 控制中心」中的 KDE 字型安裝程式來執行。結果完全相同。

您也可以建立符號連結，來替代複製實際字型。例如，如果您擁有裝載 Windows 分割區上的授權字型並且想要使用它們，便可能需要這樣做。接下來，請執行 `SuSEconfig --module fonts` 。

`SuSEconfig --module fonts` 會執行指令碼 `/usr/sbin/fonts-config` 來處理字型組態。關於此指令碼的詳細資訊，請參閱其手冊頁 (`man fonts-config`)。

點陣字型、TrueType 與 OpenType 字型以及 Type1 (PostScript) 字型的程序完全相同。所有這些字型類型可安裝到任何目錄。

X.Org 包含兩種完全不同的字型系統：舊的「*X11 核心字型系統*」與新設計的「*Xft 與 fontconfig*」系統。以下數節簡短描述這兩種系統。

26.2.1 X11 核心字型

目前，X11 核心字型系統不僅支援點陣字型，也支援縮放字型如 Type1 字型、TrueType 與 OpenType 字型。縮放字型僅只支援不需要消除鋸齒及子像素 (subpixel) 處理的字型，含有更多語言的文字符號的縮放字型所需的載入時間也越長。也支援 Unicode 字型，但是處理速度較慢且需要更多的記憶體。

X11 核心字型系統具有某些先天性的弱點。該字型系統已過時，無法再以有效的方法擴充。保留這種字型是為了回溯相容，但最好儘可能使用更先進的 Xft 和 fontconfig 系統。

為了能夠進行作業，X 伺服器必須知道它有哪些字型可用，以及系統中的哪些地方可以找到這些字型。這是由 `FontPath` 變數負責處理，該變數包含所有有效系統字型目錄的路徑。每個目錄中的 `fonts.dir` 檔案，可以列出目錄中可用的字型有哪些。`FontPath` 是 X 伺服器在啟動時產生的。它會在 `/etc/X11/xorg.conf` 組態檔案的 `FontPath` 項目中，搜尋一個有效的 `fonts.dir` 檔案。這些項目會在 `Files` 區段中找到。使用 `xset q` 顯示實際的 `FontPath`。也可以使用 `xset` 在執行時變更此路徑。若要新增其他路徑，請使用 `xset+fp <path>`。若要移除不要的路徑，請使用 `xset-fp <path>`。

如果 X 伺服器已經啟動，裝載目錄中新安裝的字型可以透過指令 `xsetfp rehash` 來設成可用。這個指令以 `SuSEconfig--module fonts` 執行。由於指令 `xset` 必須存取執行中的 X 伺服器，因此只有在 `SuSEconfig--module fonts` 是從可存取之執行中 X 伺服器的外圍程序啟動時才有效。實現此目的的最簡單方法是輸入 `su` 和 `root` 密碼，採用 `root` 權限。`su` 可以將啟動 X 伺服器之使用者的存取權傳送到 `root` 外圍程序。要檢查字型是否安裝正確以及是否可以透過 X11 核心字型系統來使用，請使用 `xlsfonts` 指令來列出所有可用字型。

根據預設，SUSE Linux Enterprise 使用 UTF-8 語言環境。因此，最好使用 Unicode 字型 (在 `xlsfonts` 的輸出中，字型名稱的結尾為 `iso10646-1`)。`xlsfonts| grep iso10646-1` 可以列出所有可用的 Unicode 字型。SUSE Linux Enterprise 隨附的 Unicode 字型絕大部分都具有歐洲語言所需的文字符號 (舊編碼方式為 `iso-8859-*`)。

26.2.2 Xft

從一開始，Xft 的程式設計人員便保證，會支援平滑美觀的可縮放字型。如果使用 Xft，字型將由使用字型的應用程式處理，而不是如 X11 核心字型系統那樣由 X 伺服器處理。在這種方式，個別的應用程式需要存取實際字型檔並完全控制文字符號的處理方式。多種語言文字的正確顯示基礎便是由此構成。直接存取字型檔對於在列印中內嵌字型，來保證列印出的外觀和螢幕輸出一致非常有用。

在 SUSE Linux Enterprise 中，KDE 和 GNOME 這兩個桌面環境、Mozilla 以及其他許多應用程式已經預設使用 Xft。越來越多的應用程式已經從舊 X11 核心字型系統改用 Xft。

Xft 使用 fontconfig 程式庫來尋找字型以及操作字型的處理方式。fontconfig 的內容是由全域組態檔 `/etc/fonts/fonts.conf` 和使用者專屬組態檔 `~/.fonts.conf` 控制。這些 fontconfig 組態檔每一個都必須以下列開頭

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
```

並以下列結束

```
</fontconfig>
```

要新增字型搜尋目錄，請附加下列行：

```
<dir>/usr/local/share/fonts/</dir>
```

不過，通常並不需要這樣做。依預設，使用者專屬目錄 `~/.fonts` 已經輸入到 `/etc/fonts/fonts.conf` 中。因此，安裝額外的字型時，只要將它們複製到 `~/.fonts` 即可。

您還可以插入操作字型外觀的規則。例如，輸入

```
<match target="font">
```

```
 <edit name="antialias" mode="assign">
  <bool>false</bool>
 </edit>
</match>
```

可以關閉所有字型的消除鋸齒效果，

```
<match target="font">
 <test name="family">
  <string>Luxi Mono</string>
  <string>Luxi Sans</string>
 </test>
 <edit name="antialias" mode="assign">
 <bool>false</bool>
 </edit>
</match>
```

可以關閉特定字型的消除鋸齒效果。

依預設，大部份的應用程式都是使用字型名稱 `sans-serif`(或相等的 `sans`)、`serif` 或 `monospace`。這些都不是真實的字型，而是可以解析為適當字型的別名，視語言設定而定。

使用者很容易在 `~/.fonts.conf` 新增規則來將這些別名解析為屬意的字型：

```
<alias>
 <family>sans-serif</family>
 <prefer>
  <family>FreeSans</family>
 </prefer>
</alias>
<alias>
 <family>serif</family>
 <prefer>
  <family>FreeSerif</family>
 </prefer>
</alias>
<alias>
 <family>monospace</family>
 <prefer>
  <family>FreeMono</family>
 </prefer>
</alias>
```

因為幾乎所有的應用程式都預設使用這些別名，所以幾乎整個系統會受到影響。這樣，您幾乎可以簡單地隨處使用喜歡的字型而不必針對個別應用程式修改字型設定。

請使用指令 fc-list 來找出已安裝且可用的字型。例如，指令 fc-list 指令可傳回所有字型的清單。要找出含有希伯來文 (:lang=he) 所有文字符號的可用縮放字型 (:scalable=true)、這些字型的名稱 (family)、樣式 (style)、粗細 (weight) 以及包含這些字型的檔案名稱，請輸入下列指令：

```
fc-list ":lang=he:scalable=true" family style weight
```

這個指令的輸出看起來如下：

```
FreeSansBold.ttf: FreeSans:style=Bold:weight=200
FreeMonoBoldOblique.ttf: FreeMono:style=BoldOblique:weight=200
FreeSerif.ttf: FreeSerif:style=Medium:weight=80
FreeSerifBoldItalic.ttf: FreeSerif:style=BoldItalic:weight=200
FreeSansOblique.ttf: FreeSans:style=Oblique:weight=80
FreeSerifItalic.ttf: FreeSerif:style=Italic:weight=80
FreeMonoOblique.ttf: FreeMono:style=Oblique:weight=80
FreeMono.ttf: FreeMono:style=Medium:weight=80
FreeSans.ttf: FreeSans:style=Medium:weight=80
FreeSerifBold.ttf: FreeSerif:style=Bold:weight=200
FreeSansBoldOblique.ttf: FreeSans:style=BoldOblique:weight=200
FreeMonoBold.ttf: FreeMono:style=Bold:weight=200
```

可以使用 fc-list 來查詢的重要參數：

表格 26.2 *fc-list 的參數*

參數	意義及可用值
family	字型系列的名稱，例如，FreeSans。
foundry (製造商)	字型的製造商，例如，urw。
style (樣式)	字型樣式，例如，Medium (中)、Regular (一般)、Bold (粗體)、Italic (斜體) 或是 Heavy (濃)。
lang	字型支援的語言，例如，de 表示德文、ja 表示日文、zh-TW 表示繁體中文、zh-CN 表示簡體中文。
weight	字型粗細，例如，80 表示一般，或是 200 表示粗體。
slant (斜度)	通常，斜度 0 表示沒有斜度，100 表示斜體。

參數	意義及可用值
file	字型檔案的名稱。
outline（外框）	true（真）表示外框字型，或是 false（偽）表示其他字型。
scalable（縮放）	true（真）表示縮放字型，或是 false（偽）表示其他字型。
bitmap（點陣）	true（真）表示點陣字型，或是 false（偽）表示其他字型。
pixelsize（點大小）	以點表示的字型大小。這個選項和 fc-list 一起使用時只對點陣字型有意義。

26.3 如需更多資訊

請安裝 xorg-x11-doc 和 howtoenh 套件來取得有關 X11 的詳盡資訊。可在專案的首頁 http://www.x.org 上找到關於 X11 開發的更多資訊。

27

使用 PAM 驗證

Linux 在驗證程序中將 PAM (可插式驗證模組，Pluggable Authentication Modules) 當做使用者與應用程式之間的溝通層。PAM 模組適用於整個系統，任何應用程式皆可要求。本章節說明模組驗證機制如何運作以及如何設定。

系統管理員與程式設計人員通常會想要限制系統某些部份的存取，或是限制應用程式某些功能的使用。如果沒有 PAM，則每次引用新的驗證機制 (例如 LDAP、Samba 或 Kerberos) 時都必須調整應用程式。然而這個程序相當耗費時間，而且容易產生錯誤。避免這些缺點的方法就是將應用程式與驗證機制區隔開來，並將驗證委託給集中管理的模組。如此一來每當需要新的驗證配置時，就能夠調整或撰寫適當的 PAM 模組以供有問題的程式使用。

每個依賴 PAM 機制的程式都有自己的組態檔，位置在 `/etc/pam.d/`*`programname`*。這些檔案是定義用來驗證的 PAM 模組。除此之外，`/etc/security` 之下還有幾個 PAM 模組的全域組態檔可以定義這些模組的精確行為 (這些範例包括 `pam_env.conf`、`pam_pwcheck.conf`、`pam_unix2.conf` 以及 `time.conf`)。每個應用程式使用 PAM 模組時，其實就是呼叫一組 PAM 函數，然後處理各種組態檔中的資訊，並將結果傳回呼叫的應用程式。

27.1 PAM 組態檔的結構

在 PAM 組態檔中的每一行最多包含四個資料欄：

```
<Type of module> <Control flag> <Module path> <Options>
```

PAM 模組是以堆疊的方式處理。不同模組的類型就有不同的目的，例如，一個模組檢查密碼，另一個驗證存取系統的位置，還有一個則讀取使用者特定的設定值。PAM 共有四種不同類型的模組：

驗證
: 此類型模組的目的就是要檢查使用者的驗證性。傳統上是以查詢密碼的方式來檢查，但是這也可以使用晶片卡或透過生物測定 (指紋或虹彩掃描) 來完成。

帳戶
: 此類型的模組會檢查使用者是否具有使用所要求服務的一般權限。舉例而言，執行這類檢查是要確保沒有人可以用使用者已過期的帳戶登入。

密碼
: 此類型模組的目的就是要能夠變更驗證權杖。在大部份情況下，這個權杖是密碼。

工作階段
: 此類型的模組是用來管理和設定使用者的工作階段。模組在驗證之前和之後啟動，以便在系統日誌中註冊登入嘗試，並設定使用者的特定環境 (郵件帳戶、主目錄、系統限制等等)。

第二個資料欄包含多種控制旗標，它們會影響已啟動模組的行為：

需要
: 具有此旗標的模組必須成功地處理完成後，才能開始進行驗證。具有需要旗標的模組失敗後，需待所有具有相同旗標的其他模組處理完畢，使用者才會收到關於驗證嘗試失敗的訊息。

必要
: 具有此旗標的模組也必須成功地處理，方式與具有需要旗標的模組相似。然而在失敗時，具有此旗標的模組會對使用者發出立即回應，而且不會再處理其他的模組。當成功時，就會接著處理其他的模組，像是任何具有需要旗標的模組。必要旗標可以當做基本的過濾器，檢查是否具備正確驗證所必需的條件。

充分
: 具有此旗標的模組經成功處理後，呼叫應用程式會收到關於成功的立即訊息，而且不會再處理其他的模組(假設具有需要旗標的模組之前沒有失敗)。

具有充分旗標的模組若失敗並不會有直接的結果，這是因為任何後續的模組都是以個別的順序處理。

選擇性
: 具有此旗標的模組不論是成功或失敗都不會有任何直接的結果。這對於只想顯示訊息(例如，通知使用者郵件已寄達)，但不想採取任何進一步動作的模組而言非常有用。

包含
: 如果給予這個旗標，則指定為引數的檔案會在這個位置插入。

模組只要是位於 `/lib/security` 的預設目錄中即可，其路徑並不需要明確指定，(至於 SUSE Linux Enterprise® 所支援的所有 64 位元平台，目錄則為 `/lib64/security`)。第四個資料欄可能包含指定模組的選項，例如 `debug` (啟用除錯)或 `nullok` (允許使用空密碼)。

27.2 sshd 的 PAM 組態

為了顯示 PAM 實際運作的理論，請將 sshd 的 PAM 組態當做實際範例：

範例 27.1 *sshd 的 PAM 組態*

```
#%PAM-1.0
auth     include        common-auth
auth     required       pam_nologin.so
account  include        common-account
password include        common-password
session  include        common-session
# Enable the following line to get resmgr support for
# ssh sessions (see /usr/share/doc/packages/resmgr/README.SuSE)
#session  optional      pam_resmgr.so fake_ttyname
```

典型的應用程式 (本例中為 sshd) PAM 組態會包含四個 include 陳述式，它們參照四個模組類型的組態檔：`common-auth`、`common-account`、`common-password` 和 `common-session`。這四個檔案具有每個模組類型的預設組態。藉由包括它們而不是為每個 PAM 應用程式分別呼叫每個模組，這樣如果管理員變更預設值，就會自動取得更新的 PAM 組態。在以前，當 PAM 有變更或是安裝新應用程式時，必須為所有的應用程式手動調整所有的組態檔案。現在 PAM 組態是由中央的組態檔案組成，每個服務的 PAM 組態都會自動繼承所有的變更內容。

第一個 include 檔案 (common-auth) 會呼叫 auth 類型的兩個檔案：pam_env 與 pam_unix2。請參閱範例 27.2「auth 區段的預設組態」[456頁]。

範例 27.2 *auth 區段的預設組態*

```
auth    required        pam_env.so
auth    required        pam_unix2.so
```

第一個 pam_env 會載入 /etc/security/pam_env.conf 檔案，以根據此檔案的指定來設定環境變數。這可用來將 DISPLAY 變數設為正確的值，因為 pam_env 模組知道登入正在發生的位置。第一個模組 pam_unix2 會根據 /etc/passwd 與 /etc/shadow 檢查使用者的登入名稱與密碼。

在成功地呼叫 common-auth 中所指定的模組後，第三個模組會呼叫 pam_nologin，以檢查 /etc/nologin 是否存在。如果它存在，則除了 root 以外的使用者都不能登入。在處理完 auth 模組的整個堆疊後，sshd 才會收到關於登入是否成功的回應。假定堆疊的所有模組都有 required 控制旗標，則必須成功地處理完所有的模組後，sshd 才會收到成功結果的訊息。如果其中一個模組沒有成功，仍然會處理整個模組堆疊，並且會在此時通知 sshd 失敗的結果。

只要已成功處理 auth 類型的所有模組，就會處理另一個 include 陳述式，在此例為 範例 27.3「account 區段的預設組態」[456頁] 中的陳述式。common-account 只包含一個模組：pam_unix2。如果 pam_unix2 傳回的結果是使用者存在，則 sshd 會收到此已成功的訊息，而且會接著處理下一個堆疊的模組 (password)，如 範例 27.4「password 區段的預設組態」[456頁] 中所示。

範例 27.3 *account 區段的預設組態*

```
account required        pam_unix2.so
```

範例 27.4 *password 區段的預設組態*

```
password required       pam_pwcheck.so  nullok
password required       pam_unix2.so    nullok use_first_pass use_authtok
#password required      pam_make.so     /var/yp
```

此外，sshd 的 PAM 組態只有一個 include 陳述式參照 password 模組的預設組態，它是在 common-password 中。不論應用程式何時要求變更驗證權杖，這些模組皆必須順利完成 (控制旗標 required)。變更密碼或另一個驗證權杖時，必需做安全性檢查。這是以 pam_pwcheck 模組來達成。之後使用的 pam_unix2 模組會沿用任何 pam_pwcheck 的舊密碼或新密碼，因此使用者不必再進行驗

證。這也使人無法規避 pam_pwcheck 所執行的檢查。當 account 或 auth 類型前面的模組是設定成會針對過期密碼發出警告時，就應該使用 password 類型的模組。

範例 27.5 *session 區段的預設組態*

```
session required        pam_limits.so
session required        pam_unix2.so
session optional        pam_umask.so
```

最後一個步驟會呼叫與 common-session 檔案繫結在一起的 session 類型的模組，以便根據有問題的使用者之設定值來設定工作階段。雖然會已再次處理 pam_unix2，但是由於在此模組 pam_unix2.conf 的個別組態中指定了 none 選項，所以不會有實際的結果。pam_limits 模組會載入 /etc/security/limits.conf 檔案，它可以定義某些系統資源的使用限制。當使用者登出時，會再次呼叫工作階段模組。

27.3 PAM 模組的組態

有些 PAM 模組是可設定的。對應的組態檔位於 /etc/security。本小節簡單說明與 sshd 範例相關的組態檔—pam_unix2.conf、pam_env.conf、pam_pwcheck.conf 以及 limits.conf。

27.3.1 pam_unix2.conf

傳統的密碼驗證方法是由 PAM 模組 pam_unix2 所控制。它可以從 /etc/passwd、/etc/shadow、NIS 對應、NIS+ 表格或 LDAP 資料庫讀取所需的資料。設定個別應用程式本身的 PAM 選項或編輯 /etc/security/pam_unix2.conf 以進行全域設定，即可影響此模組的行為。範例 27.6「pam_unix2.conf」[457頁]中顯示了模組最基本組態檔。

範例 27.6 *pam_unix2.conf*

```
auth:   nullok
account:
password:       nullok
session:        none
```

auth 與 password 模組類型的 nullok 選項是指定對應的帳戶類型可以使用空密碼。也允許使用者變更帳戶密碼。session 模組類型的 none 選項是指定其本身不記錄訊息 (這是預設值)。如需其他組態選項的資訊，請參閱檔案本身的註解以及 pam_unix2(8) 的手冊頁。

27.3.2 pam_env.conf

這個檔案可用來定義使用者的標準環境，每當呼叫pam_env模組時就會設定該環境。使用它即可使用下列語法預設環境變數：

```
VARIABLE  [DEFAULT=[value]]  [OVERRIDE=[value]]
```

VARIABLE
　要設定的環境變數名稱。

[DEFAULT=[value]]
　管理員想要設定的預設值。

[OVERRIDE=[value]]
　可以由 pam_env 查詢及設定的值，用以覆寫預設值。

每次執行遠端登入時調整的 DISPLAY 變數，就是使用 pam_env 的典型範例。這會顯示於 範例 27.7「pam_env.conf」[458頁]。

範例 27.7 *pam_env.conf*

```
REMOTEHOST      DEFAULT=localhost OVERRIDE=@{PAM_RHOST}
DISPLAY         DEFAULT=${REMOTEHOST}:0.0 OVERRIDE=${DISPLAY}
```

第一行將 REMOTEHOST 變數的值設為 localhost，每當 pam_env 無法決定任何其他值時，就會使用這個值。DISPLAY 變數接著就會包含 REMOTEHOST 的值。如需詳細資訊，可以從 /etc/security/pam_env.conf 檔案中的註解取得。

27.3.3 pam_pwcheck.conf

此組態檔是供pam_pwcheck模組使用，該模組會從該檔案讀取所有password類型模組的選項。此檔案所儲存設定值的優先順序，高於個別應用程式的 PAM

設定值。如果尚未定義應用程式特定的設定值，應用程式就會使用全域設定值。範例 27.8「pam_pwcheck.conf」[459頁] 說明 pam_pwcheck 允許空的密碼和密碼的修改。模組的更多選項，請見 /etc/security/pam_pwcheck.conf 檔案中的說明。

範例 27.8 *pam_pwcheck.conf*

```
password:    nullok
```

27.3.4 limits.conf

系統限制可以以使用者或群組為單位，在 limits.conf 檔案中做設定，該檔案是由 pam_limits 模組讀取。該檔案允許您設定硬性限制，也就是完全不能超過，也可設定軟性限制，也就是可以暫時超過。若要瞭解關於語法以及可用選項的資訊，請參閱檔案中所包括的註解。

27.4 如需更多資訊

在已安裝系統的 /usr/share/doc/packages/pam 目錄中，可以找到下列的其他文件：

README
: 在此目錄的最上層中，有一些一般性的 README 檔案。modules 子目錄中存有關於可用 PAM 模組的 README 檔案。

Linux-PAM 系統管理員指南
: 此文件包括系統管理員就 PAM 所應該知道的每件事。它討論很多主題，涵蓋組態檔的語法至 PAM 的安全性層面。該文件有三種格式：PDF 檔案、HTML 格式以及純文字。

Linux-PAM 模組撰寫者手冊
: 此文件是以開發人員的觀點將主題做成摘要，內含如何撰寫標準相容 PAM 模組的資訊。它具有三種格式：PDF 檔案、HTML 格式以及純文字。

Linux-PAM 應用程式開發人員指南
: 想要使用 PAM 程式庫的應用程式開發人員所需的全部內容都包括在此文件中。它具有三種格式：PDF 檔案、HTML 格式以及純文字。

Thorsten Kukuk 已開發一些 PAM 模組，並提供一些可用的相關資訊，網址為 http://www.suse.de/~kukuk/pam/。

28 電源管理

電源管理對筆記型電腦十分重要，對其他系統也很有用。有兩種技術可供使用：APM (進階電源管理) 和 ACPI (進階組態和電源介面)。除此之外，也可以控制 CPU 頻率比例，這有助於省電及降低噪音。您可以手動或使用特殊 YaST 模組來設定這些選項。

► **zseries:** 本章所描述的功能與硬體，IBM System z 不提供，因此本章的內容與這些平台無關。 ◄

電源管理對筆記型電腦十分重要，對其他系統也很有用。ACPI (進階組態和電源介面) 可以在所有新式電腦 (筆記型電腦、桌上型電腦和伺服器) 上使用。電源管理技術需要配備合適的硬體與 BIOS 常式。大多數筆記型電腦和許多新式的桌上型電腦及伺服器都符合這些需求。此技術還可以控制 CPU 頻率比例，這有助於省電及降低噪音。

過去許多電腦都是使用 APM。由於 APM 大部份是由實作在 BIOS 中的一組功能所組成，使得 APM 支援的等級會因硬體而異。ACPI 更加有此特性，其組成也更加複雜。因此，實際上是無法說明哪種技術比較好。只有對硬體進行各種測試，然後再選出支援性最好的技術。

28.1 省電功能

省電功能不僅對於筆記型電腦的行動用途很重要，對於桌上型系統也很重要。這些主要功能在電源管理 APM 和 ACPI 中有下列用途：

待命

此操作模式關閉畫面顯示。在某些電腦上，會調節 (Throttling) 處理器的效能。此功能等同於 ACPI 狀態 S1 或 S2。

暫停 (記憶體)

此模式會將整個系統狀態寫入 RAM 中。接著，除了 RAM 以外，整個系統都會進入睡眠狀態。在此狀態中，電腦所使用的電源極少。此狀態的好處是可以在幾秒內將工作復原到暫停之前的狀態，而不用開機或重新啟動應用程式。此功能等同於 ACPI 狀態 S3。對此狀態的支援仍在開發中，因此支援程度多半因硬體而異。

睡眠 (暫停磁碟)

在此操作模式，會將整個系統狀態寫入硬碟，然後關閉系統。至少要有與 RAM 一樣大的交換分割區，才能寫入所有作用中資料。要從此狀況重新啟用需耗時 30 到 90 秒。還原時會回到暫停前的狀態。有些製造商會為此模式提供有用的混合功能，例如 IBM Thinkpad 中的 RediSafe。等同的 ACPI 狀態為 S4。在 Linux 中，暫停寫入到磁碟是由獨立於 APM 與 ACPI 之外的核心常式來執行。

電池監視器

ACPI 與 APM 會檢查電池充電的狀態，並提供相關資訊。此外，兩個系統都會在電力到達某個關鍵狀態時，協調要執行的動作。

自動關閉電源

關機後，電腦會關閉電源。此功能很重要，尤其是在電池用盡前所執行的自動關機。

關閉系統元件

就整個系統而言，關閉硬碟是最省電的方式。依整個系統的穩定度而異，有時可讓硬碟進入睡眠狀態。不過，睡眠時期的持續時間愈長，遺失資料的風險愈大。其他如 PCI 設備這一類可設為特殊省電模式的元件，可用 ACPI 予以停用 (至少理論上可行)，或在 BIOS 設定中永久停用。

處理器速度控制

與 CPU 連結時有三種方式可節省電源：頻率和電壓比例 (也稱為 PowerNow! 或 Speedstep)、節流，以及使處理器暫休 (C 狀態)。依據電腦的操作模式，也可以合併這些操作方法。

28.2 APM

APM BIOS 本身能執行一些省電功能。許多筆記型電腦則可透過組合鍵或關閉蓋子等方式來啟用待命狀態或暫停狀態，而不需任何特別地操作系統功能。不過，如果要以指令啟用這些模式，必須先觸發一些動作才能使系統暫停。如果要檢視電池充電程度，需要有特別的程式套件與適用的核心。

SUSE Linux Enterprise® 核心內建即支援 APM。不過，APM 只在 ACPI 未在 BIOS 中實行以及偵測到 APM BIOS 時才會啟用。如果要啟用 APM 支援，必須在開機提示中輸入 `acpi=off` 以停用 ACPI。請輸入 `cat /proc/apm` 以檢查 APM 是否啟動。輸出中含有各種數字的話，表示一切正常。現在若使用 `shutdown -h` 指令，應可關閉電腦。

BIOS 實行若未能完全符合標準則會導致 APM 發生問題。有些問題可以使用特定的開機參數來加以避免。所有參數以 `apm=`*`parameter`* 形式輸入於開機提示中，*`parameter`* 為下列之一：

on or off
: 開啟或關閉 APM 支援。

(no-)allow-ints
: 允許在 BIOS 功能執行期間出現中斷。

(no-)broken-psr
: BIOS 的「GetPowerStatus」功能無法正常運作。

(no-)realmode-power-off
: 在關機前，重新設定處理器至真實模式。

(no-)debug
: 在系統記錄中記錄 APM 事件。

(no-)power-off
: 在關機後關閉系統電源。

bounce-interval=*n*
: 暫停事件發生後，要經過多少時間才能執行下個暫停事件，該時間以 0.01 秒為單位。

idle-threshold=*n*
執行 BIOS 的 `idle` 功能後，系統閒置的百分比 (0=永遠，100=永不)。

idle-period=*n*
系統啟動後所經過的時間，該時間以 0.01 秒為單位。

APM 精靈 (apmd) 已停用。其功能目前是由新的 powersaved 處理，powersaved 也支援 ACPI，而且提供許多其他功能。

28.3 ACPI

ACPI (進階組態與電源介面) 可讓作業系統設定和控制個別的硬體元件。ACPI 可取代 PnP 與 APM。它能提供一些資訊，包括電池、變電器、溫度、風扇以及「關閉蓋子」或「電池電力不足」等系統事件。

BIOS 會提供一些表格，內含關於個別元件與硬體的存取方法等資訊。作業系統會使用這此資訊來執行任務，像是指定中斷或啟用和停用元件。因為作業系統會執行儲存於 BIOS 中的指令，所以 BIOS 實行會決定其功能。ACPI 能偵測和載入的表格在 `/var/log/boot.msg` 中可以找到。請參閱第 28.3.4 節「疑難排解」[469頁]，以取得更多有關 ACPI 問題疑難排解的資訊。

28.3.1 ACPI 的動作

如果核心在系統啟動時偵測到 ACPI BIOS，會自動啟用 ACPI。有些較舊的機器可能需要用到開機參數 `acpi=force`。電腦需支援 ACPI 2.0 或以後的版本。請檢查 `/var/log/boot.msg` 中的核心開機訊息，以查看 ACPI 是否啟用。

接著，需載入一些模組。這會由 acpid 的啟動程序檔來完成。如果其中任何一個模組導致問題發生，該項模組就不會在 `/etc/Sysconfig/powersave/common` 中載入或取消載入。系統記錄 (`/var/log/messages`) 內有模組的訊息，從中可以知道已偵測到哪些元件。

`/proc/acpi` 目前含有一些檔案，用來提供有關系統狀態的資訊，也可以用來對某些狀態進行變更。有些功能還不能使用，因為仍在開發中，而且有些功能的支援主要是依靠製造商是否在產品中實行。

所有檔案 (`dsdt` 與 `fadt` 除外) 都可使用 `cat` 來讀取。有些檔案可使用 `echo` 修改設定，例如 `echo x > file` 可為 X 指定適合的值。`powersave` 指令是

可存取這些值的一種簡單方式，可做為省電精靈的前端工具。最重要的檔案說明如下：

`/proc/acpi/info`
有關 ACPI 的一般資訊。

`/proc/acpi/alarm`
在此指定何時從睡眠狀態中喚醒系統。目前，尚未完整支援此功能。

`/proc/acpi/sleep`
提供可能的睡眠狀態的相關資訊。

`/proc/acpi/event`
所有事件都會會在此報告，並經由 Powersave 精靈來加以處理 (`powersaved`)。如果沒有精靈存取此檔案，則諸如快速按一下電源按鈕或是關閉蓋子等事件，可使用 `cat /proc/acpi/event` 來讀取 (按 Ctrl + C 來終止)。

`/proc/acpi/dsdt` 與 `/proc/acpi/fadt`
這些檔案包含 ACPI 的 (不同系統說明表格) DSDT 表格和 (固定 ACPI 說明表格) FADT 表格。可以使用 `acpidmp`、`acpidisasm` 與 `dmdecode` 來讀取它們。這些程式及其文件位於 `pmtools` 套件中。例如，`acpidmp DSDT | acpidisasm`。

`/proc/acpi/ac_adapter/AC/state`
顯示是否已連接 AC 轉換器。

`/proc/acpi/battery/BAT*/{提醒、資訊、狀態}`
有關電池狀態的詳細資訊。透過比較資訊中的上次完整電量以及狀態中的剩餘電量來讀取充電等級。另一種更方便的方法，便是使用在 第 28.3.3 節「ACPI 工具」[469頁] 中所介紹的特別程式。會觸發電池事件的充電等級 (例如警告、低和極低) 可在提醒中指定。

`/proc/acpi/button`
此目錄包含各種切換 (如筆記型電腦蓋和按鈕) 的資訊。

`/proc/acpi/fan/FAN/state`
顯示風扇是否正在運作。以手動方式透過在此檔案中寫入 `0` (開啟) 或 `3` (關閉)，以啟用或停用風扇。不過，在系統過熱時，核心與硬體(或 BIOS) 中的 ACPI 程式碼都會覆寫此設定。

`/proc/acpi/processor/*`
系統中的每個 CPU 都有個別的子目錄。

`/proc/acpi/processor/*/info`
關於處理器的省電選項的資訊。

`/proc/acpi/processor/*/power`
關於目前處理器狀態的資訊。在 `C2` 旁邊有星號表示處理器閒置中。查看*使用*值時，這是最常出現的狀態。

`/proc/acpi/processor/*/throttling`
能用來設定調節處理器的時脈。通常，調節可以有八個層級。這和 CPU 的頻率控制是兩回事。

`/proc/acpi/processor/*/limit`
如果是由精靈來自動控制效能(過時) 及調節功能，則可在此指定最大上限。部份限制是由系統所決定。部份則可由使用者來調整。

`/proc/acpi/thermal_zone/`
每個溫度區都有一個子目錄。溫度區是指一個具有類似的溫度屬性的區域，由硬體製造商指定區域的數值及名稱。不過大部份由 ACPI 所提供的可能選項，很少被實作。通常反而是使用 BIOS 來控制溫度。作業系統很少有干預的機會，因為這可能會減少硬體的使用壽命。因此，有一部份的檔案僅具理論價值。

`/proc/acpi/thermal_zone/*/temperature`
溫度區目前的溫度。

`/proc/acpi/thermal_zone/*/state`
指示是否一切都*正常*，或 ACPI 是否套用*主動*或*被動*冷卻規則。如果風扇控制獨立於 ACPI 之外，則狀態會一直是*正常*。

`/proc/acpi/thermal_zone/*/cooling_mode`
選取由 ACPI 控制的冷卻方法。從被動 (效能較低，經濟型) 選擇或是作用中的冷卻模型 (完整的效能、風扇噪音)。

/proc/acpi/thermal_zone/*/trip_points
: 啟用判定溫度上限以觸發指定動作的功能，像是被動或主動冷卻、暫停 (過熱) 或關機 (嚴重)。在 DSDT (視設備而有所不同) 中定義可能動作。在 ACPI 指定中的啟動點分別是嚴重、過熱、被動、主動 1 及主動 2。即使不會每個都會實作，仍必須依此順序將它們輸入在此檔案中。例如，echo 90:0:70:0:0 > trip_points 項目即是將嚴重的溫度設為 90，將被動設為 70 (以攝氏測量的所有溫度)。

/proc/acpi/thermal_zone/*/polling_frequency
: 如果在 temperature 檔案中的值，未在溫度變更時自動更新，請在此切換輪詢模式。echo X > /proc/acpi/thermal_zone/*/polling_frequency 指令能限定每 X 秒查詢一次溫度。設定 X=0 以關閉輪詢。

這些設定、資訊及事件都不需以手動方式編輯。可以使用省電精靈 (powersaved) 及其各種前端工具來完成，像是 powersave、kpowersave 及 wmpowersave。請參閱第 28.3.3 節「ACPI 工具」[469頁]。

28.3.2 控制 CPU 效能

CPU 有三種省電方法。依據電腦的操作模式的不同，這些方法可合併使用。省電也表示能降低系統溫度，減低風扇的使用頻率。

頻率和電壓比例
: PowerNow! 與 Speedstep 是 AMD 與 Intel 分別針對此技術所使用的實作。不過，此技術也套用於其他製造商的處理器中。CPU 的時脈頻率及其核心電壓會同時降低，產生高於線性的省電效能。也就是當頻率減半 (效能減半)，使用的電量卻能遠低於原本的一半。此技術與 APM 或 ACPI 無關。CPU 頻率比例的執行有兩種主要的方式—利用核心本身或利用使用者空間應用程式。因此，有不同的核心調節器，這可在 /sys/devices/system/cpu/cpu*/cpufreq/ 下設定。

  使用者空間調節器
  : 如果設定使用者空間調節器，核心會將 CPU 頻率比例的控制權交給使用者空間應用程式 (通常是一個精靈)。在 SUSE Linux Enterprise 版本中，這個精靈是 powersaved 套件。使用此執行方式時，CPU 頻率會根據目前的系統負載來調整。預設狀態下會使用核心執行方式之一。但是，

對某些硬體，或特定的處理器或驅動程式而言，使用者空間執行仍是唯一可行的解決方案。

要求式調節器
: 這是動態 CPU 頻率規則的核心執行方式，應該適用於大部分系統。只要系統負載一過高，就立即提高 CPU 頻率。系統負載降低，則會隨之降低。

保守調節器
: 此調節器類似要求式調節器，但所用的規則比較保守。系統負載過高的情況必須持續一段時間後，才會提高 CPU 頻率。

省電調節器
: CPU 頻率固定設為可能的最小值。

效能調節器
: CPU 頻率固定設為可能的最大值。

調節時脈頻率
: 此技術會忽略部分 CPU 的時脈訊號脈衝。到達 25% 調節時，會省略四分之一脈衝，到達 87.5% 時則每八次脈衝僅有一次會到達處理器。不過，節省用電量稍低於線性。通常調節功能僅在無此頻率比例時使用，或是為了最大化省電效果時使用。此外，此技術必須使用特定程序來進行控制。系統介面是 `/proc/acpi/processor/*/throttling`。

使處理器進入睡眠
: 作業系統會在沒事可做時使處理器進入睡眠。在此情況中，作業系統會傳送 `halt` 指令給 CPU。一共有三種狀態：C1、C2 和 C3。在最節省的 C3 狀態中，連處理器快取與主記憶體間的同步化也會暫停。因此，僅能在沒有任何設備透過 Bus master 活動來修改主記憶體內容時可以套用此狀態。有些驅動程式會禁止使用 C3。目前的狀態會顯示在 `/proc/acpi/processor/*/power` 中。

頻率比例及調節只在處理器忙碌時使用，因為在處理器閒置時，一定會套用最節省的 C 狀態。如果 CPU 正忙碌，頻率比例是建議的省電方法。通常處理器僅有部份的工作負載。在此情況中，可以使用較低的頻率。通常，最佳方法是使用核心要求式調節器或精靈 (例如 powersaved) 來控制動態頻率比例。對電池的操作而言，靜態設定為較低頻率比較好，也可以用在您想降低電腦溫度或減低噪音時。

調節應做最後手段使用，例如，在高度系統負載下仍要延伸電池操作時間時。不過在調節過多時，有些系統無法運作順暢。此外，當 CPU 要做的事不多時，調節 CPU 是無意義的動作。

在 SUSE Linux Enterprise 中，這些技術是由 powersave 精靈所控制。組態的說明位於 第 28.5 節「powersave 套件」[472頁]。

28.3.3 ACPI 工具

ACPI 公用程式包含僅顯示電池充電等級與溫度等資訊的工具 (acpi、klaptopdaemon 及 wmacpimon 等等。)、協助在 `/proc/acpi` 中存取結構或協助監控變更 (akpi、acpiw、gtkacpiw) 的工具，以及在 BIOS 中編輯 ACPI 表格的工具 (`pmtools` 套件)。

28.3.4 疑難排解

共有兩種不同類型的問題。一方面是核心的 ACPI 程式碼包含無法及時偵測到的錯誤。在這種情況中，將會有可供下載的解決方案。不過通常問題是因 BIOS 而起。有時，會刻意在 BIOS 中整合與 ACPI 規格不符的技術，以避免在其他常見作業系統中的 ACPI 實作錯誤。會在黑名單中將那些在 ACPI 實行中有重大錯誤的硬體元件記錄下來，以避免 Linux 核心對這些元件使用 ACPI。

發生問題時要做的第一件事是更新 BIOS。如果電腦未能開機，下列中的某一個開機參數也許有幫助：

pci=noacpi
: 不使用 ACPI 來設定 PCI 設備。

acpi=ht
: 僅執行一個簡單的資源組態。不將 ACPI 用於其他目的。

acpi=off
: 關閉 ACPI。

警告：未使用 ACPI 的開機問題

有些較新的機器 (尤其是 SMP 系統及 AMD64 系統) 需透過 ACPI 以正確設定硬體。關閉這些機器的 ACPI 會發生隨之而來的問題。

開機後，可使用 `dmesg | grep -2i acpi` 指令來監控系統的開機訊息 (或所有訊息，因為也可能是 ACPI 以外的因素構成問題)。如果是在分析 ACPI 表格時發生問題，則最重要的 DSDT 表格可用改良版本來替換。在此情況中，會忽略 BIOS 的錯誤 DSDT。程序在 第 28.5.4 節「疑難排解」[477頁] 中描述。

在核心組態中，有個啟用ACPI除錯訊息的切換。如果已編譯並安裝好一個具有 ACPI 除錯能力的核心，則專家將能取得詳細資訊支援，以便搜尋錯誤。

如果您曾遇到BIOS問題或硬體問題，建議您聯絡製造商。尤其是哪些一直未提供 Linux 支援的製造商，更應該出面解決這些問題。唯有讓製造商得知他們有不少使用 Linux 的客戶，他們才會嚴肅地處理這些問題。

如需更多資訊

ACPI 的其他文件和說明：

- `http://www.cpqlinux.com/acpi-howto.html` (詳細的 ACPI HOWTO，內含 DSDT 修補程式)
- `http://www.intel.com/technology/iapc/acpi/faq.htm` (ACPI FAQ @Intel)
- `http://acpi.sourceforge.net/` (Sourceforge 的 ACPI4Linux 計劃)
- `http://www.poupinou.org/acpi/` (Bruno Ducrot 的 DSDT 修補程式)

28.4 硬碟的休眠

在 Linux 中，可在不需使用硬碟時，讓硬碟完全進入睡眠狀態，或是讓硬碟以更省電、更安靜的方式來運作。在目前的筆記型電腦中，您不用手動關閉硬碟，因為它們會在不用的時候自動進入省電操作模式。不過，如果您想最大化省電效果，可嘗試下列幾種方法。Powersaved 和 YaST 電源管理模組能控制大部份這方面的功能，詳細資訊請參閱 第 28.6 節「YaST電源管理模組」[479頁]。

hdparm 應用程式能修改各種硬碟設定。`-y` 選項能立即將硬碟切換到待命模式。`-Y` 能讓它進入睡眠。`hdparm -s x` 則會讓硬碟閒置一段時期後關閉。如下所示取代 *x*：`0` 會停用此機制，使得硬碟持續執行。`1` 到 `240` 的值將乘以 5 秒。

241 到 251 的值則是以 30 分鐘為一個單位，依序從 30 分鐘的閒置到 11 倍的 330 分鐘的閒置。

可以使用 -B 選項來控制硬碟內部的省電選項。可從 0 到 255 中選取一個值，以最大化省電效果或最大化電力輸出。其結果視硬碟用途而定，難以評估。如果要減少硬碟噪音，請使用 -M 選項。從 128 到 254 中選取一個值，以決定要安靜或快速。

通常，要讓硬碟進入睡眠不是件容易的事。在 Linux 中，會有多個程序寫入硬碟中，因而重複喚醒硬碟。因此，有必要去瞭解 Linux 如何處理那些要寫入硬碟的資料 首先，會將所有資料在 RAM 中做緩衝處理。核心更新精靈 (kupdated) 可以監控緩衝區。當資料到達特定的時間限制，或當緩衝區已填滿至某一程度時，會將緩衝區的內容注入硬碟。緩衝區的大小則動態地由記憶體地的大小及系統負載來決定。根據預設，kupdated 會設成較短的間隔，以最大化資料的完整性。它會每 5 秒檢查一次緩衝區，並會在資料時間大於 30 秒或緩衝區的容量已達 30% 時通知 bdflush。接著，bdflush 精靈會將資料寫入硬碟。它也可以在不經由 kupdated 之下寫入硬碟，例如，在緩衝區已滿時。

警告：損害資料完整性

變更核心更新精靈的設定有害資料的完整性。

除了這些程序之外，像是 ReiserFS 與 Ext3 等日誌檔案系統，不經由 bdflush 會將中繼資料寫入硬碟，也會使得硬碟無法停止運作。為了避免這類情形，正在開發適用於行動設備的核心延伸程式。請參閱 /usr/src/linux/Documentation/laptop-mode.txt 以取得詳細資訊。

另一個重要因素在於啟動程式的行為方式。例如，好的編輯器會定期為修改中的檔案，將隱藏備份檔寫入硬碟，因而喚醒硬碟。停用這類功能可能會傷害資料的完整性。

與此相關，postfix 郵件精靈會使用 POSTFIX_LAPTOP 變數。如果將此變數設為 yes，postfix 會減少存取硬碟的頻率。不過，若增加 kupdated 的時間間隔，則此項設定將無關緊要。

28.5 powersave 套件

powersave套件負責處理先前說明的所有省電功能。由於減低能源消耗的需求普遍提高，因此它的部分功能對工作站和伺服器也很重要，例如暫停、待命或CPU 頻率比例等。

此套件包含電腦的所有電源管理功能。它支援使用 ACPI、APM、IDE 硬碟和 PowerNow! 或 SpeedStep 技術的硬體。來自套件 apmd、acpid、ospmd 和 cpufreqd (現在為 cpuspeed) 的功能已合併於 powersave 套件中。這些套件中的精靈 (做為 acpi 事件多工器的 acpid 除外) 不得與省電精靈同時執行。

即使您的系統不包含上述所有硬體元件，仍可以使用powersave精靈來控制省電功能。因為 ACPI 和 APM 互斥，僅可在電腦上使用其中一種系統。精靈會自動偵測硬體組態的變化。

28.5.1 設定 powersave 套件

powersave的組態散佈於數個檔案。各檔案中所列的每個組態選項都包含其他有關其功能的文件。

/etc/sysconfig/powersave/common
: 此檔案包含 powersave 精靈的一般設定。例如，可以增加變數 DEBUG 的值以增加 /var/log/messages 中的偵錯訊息數量。

/etc/sysconfig/powersave/events
: powersave 精靈需要此檔案以處理系統事件。事件可以是指定的外部動作或精靈本身執行的動作。對於外部動作，精靈會嘗試執行 /usr/lib/powersave/scripts/ 中的執行檔 (通常是 Bash 程序檔)。預先定義的內部動作為：

  - ignore
  - throttle
  - dethrottle
  - suspend_to_disk

- suspend_to_ram
- standby
- do_suspend_to_disk
- do_suspend_to_ram
- do_standby
- notify
- screen_saver
- reread_cpu_capabilities

`throttle` 以 `MAX_THROTTLING` 中定義的值減緩處理器。此值視目前的配置而定。`dethrottle` 將處理器設為完整效能。`suspend_to_disk`、`suspend_to_ram` 和 `standby` 會觸發休眠模式的系統事件。這三個動作一般負責觸發休眠模式，但是它們應該永遠與特定系統事件關聯。

目錄 `/usr/lib/powersave/scripts` 包含處理事件的程序檔：

switch_vt
: 在暫停或待命之後，如果螢幕發生錯置時會很有用。

wm_logout
: 儲存設定並從 GNOME、KDE 或其他視窗管理員登出。

wm_shutdown
: 儲存 GNOME 或 KDE 設定並關閉系統。

set_disk_settings
: 執行 `/etc/sysconfig/powersave/disk` 中的磁碟設定。

例如，如果設定 `EVENT_GLOBAL_SUSPEND2DISK="prepare_suspend_to_disk do_suspend_to_disk"` 變數，兩個程序檔或動作會在使用者給 powersaved 休眠模式指令 `suspend to disk` 時，立即以特定順序處理。精靈會執行外部程序檔 `/usr/lib/powersave/scripts/prepare_suspend_to_disk`。在此程序檔成功處理之後，精靈會執行內部動作

do_suspend_to_disk，並在程序檔卸載關鍵模組並停止服務之後，將電腦設為休眠模式。

休眠按鈕事件的動作可在 EVENT_BUTTON_SLEEP="notify suspend_to_disk" 中修改。在此例下，使用者從 X 的快顯視窗或主控台的訊息中收到暫停通知。之後，會產生 EVENT_GLOBAL_SUSPEND2DISK 事件，導致執行上述動作及安全系統暫停模組。您可以用 /etc/sysconfig/powersave/common 中的 NOTIFY_METHOD 變數自定 notify 內部動作。

/etc/sysconfig/powersave/cpufreq
包含最佳化動態 CPU 頻率設定以及應使用使用者空間或核心執行方式的變數。

/etc/sysconfig/powersave/battery
包含電池限制和其他電池特定設定。

/etc/sysconfig/powersave/sleep
在此檔案中，啟用休眠模式並決定應卸載哪個關鍵模組，以及在暫停或待命事件之前應停止的服務。當系統繼續時，這些模組會重新載入並啟動服務。例如，您甚至可以延遲觸發的休眠模式以儲存檔案。預設值主要考量 USB 和 PCMCIA 模組。特定模組通常會造成暫停或待命的錯誤。請參閱第 28.5.4 節「疑難排解」 [477頁]以取得關於辨識錯誤的詳細資訊。

/etc/sysconfig/powersave/thermal
啟用冷卻和熱控制。關於此主題的詳細資訊也可在檔案 /usr/share/doc/packages/powersave/README.thermal 中找到。

/etc/sysconfig/powersave/disk
此組態檔控制針對硬碟進行的動作和設定。

/etc/sysconfig/powersave/scheme_*
這些為搭配特定部署狀況電源消耗的不同配置。數種配置已預先設定好並可供使用。可在此儲存自定配置。

28.5.2 設定 APM 和 ACPI

暫停和待命

有三種基本 ACPI 休眠模式和兩種 APM 休眠模式：

Suspend to Disk (ACPI S4、APM 暫停)
: 將全部記憶體內容儲存到硬碟。電腦完全關閉，不消耗任何電力。此休眠模式預設為啟用，而且應該適用於所有系統。

Suspend to RAM (ACPI S3、APM 暫停)
: 將所有設備的狀態儲存到主要記憶體。僅主要記憶體繼續消耗電力。雖然許多機器可以使用睡眠模式，但 SUSE Linux Enterprise 一般不支援。

  睡眠模式是預設啟用的，但僅在目前機器列於支援此模式的資料庫時才會*執行*。資料庫包含於 `suspend` 套件所提供的 `/usr/sbin/s2ram` 二進位檔。

  若要修改預設參數 (例如，全面停用 `suspend to ram` 睡眠模式，或即使機器未列於資料庫中也強制執行此模式)，請在 `/etc/sysconfig/powersave/sleep` 組態檔中尋找可用選項的更多資訊。

  若要進一步瞭解 `s2ram` 二進位檔，請參閱 `/usr/share/doc/packages/suspend` 中的 README 檔案。

Standby (ACPI S1、APM 待命)
: 關閉部份設備 (視製造商而定)。

請確定檔案 `/etc/sysconfig/powersave/events` 中的下列預設選項設為暫停、待命和繼續的正確處理 (預設值為在安裝 SUSE Linux Enterprise 之後)：

```
EVENT_GLOBAL_SUSPEND2DISK=
    "prepare_suspend_to_disk screen_saver do_suspend_to_disk"
EVENT_GLOBAL_SUSPEND2RAM=
    "prepare_suspend_to_ram screen_saver do_suspend_to_ram"
EVENT_GLOBAL_STANDBY=
    "prepare_standby screen_saver do_standby"
EVENT_GLOBAL_RESUME_SUSPEND2DISK=
    "restore_after_suspend_to_disk"
EVENT_GLOBAL_RESUME_SUSPEND2RAM=
    "restore_after_suspend_to_ram"
EVENT_GLOBAL_RESUME_STANDBY=
    "restore_after_standby"
```

自定電池狀態

在檔案 /etc/sysconfig/powersave/battery 中，定義三種電池充電強度(以百分比)，當達到時會觸發警示或特定動作。

```
BATTERY_WARNING=12
BATTERY_LOW=7
BATTERY_CRITICAL=2
```

當充電強度掉到特定限制之下，要執行的動作或程序檔定義於組態檔 /etc/sysconfig/powersave/events 中。按鈕的標準動作可以修改為如第 28.5.1 節「設定 powersave 套件」[472頁] 中所述。

```
EVENT_BATTERY_NORMAL="ignore"
EVENT_BATTERY_WARNING="notify"
EVENT_BATTERY_LOW="notify"
EVENT_BATTERY_CRITICAL="wm_shutdown"
```

將電源消耗與不同條件搭配

系統行為可與電源供應類型搭配。當系統從 AC 電源供應中斷連接並以電池操作時，系統的電源消耗應減少。相似地，系統一連接到 AC 電源供應時，效能應自動增加。CPU 頻率、IDE 的省電功能和許多其他參數均可修改。

當電腦連接到或從 AC 電源供應中斷連接時，要執行的動作定義於 /etc/sysconfig/powersave/events 中。請在 /etc/sysconfig/powersave/common 中選取要使用的配置：

```
AC_SCHEME="performance"
BATTERY_SCHEME="powersave"
```

配置儲存在 /etc/sysconfig/powersave 中的檔案內。檔案名稱的格式為 scheme_name-of-the-scheme。範例參考兩種配置：scheme_performance 與 scheme_powersave。performance、powersave、presentation，和 acoustic 都是預先配置的。藉由第 28.6 節「YaST電源管理模組」[479頁] 中描述的 YaST 電源管理模組的協助，可編輯、建立、刪除現有配置或和不同電源供應狀態相關聯。

28.5.3 其他 ACPI 功能

如果您使用 ACPI，可以控制系統對 *ACPI 按鈕* (電源、休眠、開蓋、關蓋) 的回應。請在 `/etc/sysconfig/powersave/events` 中設定動作的執行。請參閱此組態檔以取得個別選項的說明。

EVENT_BUTTON_POWER="wm_shutdown"
: 當按下電源按鈕時，系統回應為關閉對應視窗管理員 (KDE、GNOME、fvwm 等)。

EVENT_BUTTON_SLEEP="suspend_to_disk"
: 當按下休眠按鈕時，系統設為休眠 (suspend-to-disk) 模式。

EVENT_BUTTON_LID_OPEN="ignore"
: 當開啟蓋子時，不發生任何事。

EVENT_BUTTON_LID_CLOSED="screen_saver"
: 當關上蓋子時，會啟用螢幕保護程式。

EVENT_OTHER="ignore"
: 如果精靈遭遇未知的事件，就會發生此事件。未知的事件包含某些機器上的 ACPI 快速鍵。

如果 CPU 負載在特定時間未超過特定限制，則可以進一步節流 CPU 效能。在 `PROCESSOR_IDLE_LIMIT` 中指定負載限制，在 `CPU_IDLE_TIMEOUT` 中指定逾時。如果 CPU 負載未超過限制的時間大於逾時，則會啟用 `EVENT_PROCESSOR_IDLE` 中設定的事件。如果 CPU 再度忙碌，便執行 `EVENT_PROCESSOR_BUSY`。

28.5.4 疑難排解

所有錯誤訊息和警示會記錄在檔案 `/var/log/messages` 中。如果您找不到所需資訊，請使用 `/etc/sysconfig/powersave/common` 檔案中的 `DEBUG` 提高 powersave 訊息的資料詳細程度。將變數值增加到 `7` 或甚至 `15`，並重新啟動精靈。`/var/log/messages` 中更詳細的錯誤訊息可協助您找出錯誤。下列小節涵蓋 powersave 最常見的問題。

以硬體支援啟用 ACPI，但沒有作用

如果您遭遇到 ACPI 的問題，請使用指令 `dmesg|grep -i acpi` 來搜尋 ACPI 特定訊息的 `dmesg` 輸出。必須更新 BIOS 以解決問題。請到您筆記型電腦製造商的首頁，尋找更新的 BIOS 版本並安裝。請詢問製造商以符合最新 ACPI 規格。在 BIOS 更新後，如果錯誤持續發生，請繼續以更新的 DSDT 取代您 BIOS 中的錯誤 DSDT 表格：

1. 從 `http://acpi.sourceforge.net/dsdt/index.php` 下載您系統的 DSDT。檢查檔案是否已解壓縮，並以所示副檔名 `.aml` (ACPI 機器語言) 編譯。如果是此狀況，請繼續步驟 3。

2. 如果下載的表格副檔名為 `.asl` (ACPI 原始語言)，請以 iasl (`pmtools` 套件) 編譯。請輸入 `iasl -sa file.asl` 指令。最新版本的 iasl (Intel ACPI 編譯器) 可在 `http://developer.intel.com/technology/iapc/acpi/downloads.htm` 中找到。

3. 將檔案 `DSDT.aml` 複製到任何位置 (建議 `/etc/DSDT.aml`)。編輯 `/etc/sysconfig/kernel` 並將路徑與 DSDT 檔案搭配。啟動 `mkinitrd` (套件 `mkinitrd`)。只要您安裝核心並使用 `mkinitrd` 來建立 `initrd`，當系統啟動時，修改的 DSDT 便會整合並載入。

CPU 頻率沒有作用

請參考核心來源 (`kernel-source`) 以查看是否支援您的處理器。您需要特殊核心模組或模組選項以啟用 CPU 頻率控制。此資訊可在 `/usr/src/linux/Documentation/cpu-freq/*` 中找到。如果需要特殊核心模組或模組選項，可在檔案 `/etc/sysconfig/powersave/cpufreq` 中藉由變數 `CPUFREQD_MODULE` 和 `CPUFREQD_MODULE_OPTS` 加以設定。

暫停和待命沒有作用

由於 DSDT 實做 (BIOS) 的錯誤，ACPI 系統在暫停或待命時可能會有問題。若這樣的話，請更新 BIOS。

在 ACPI 與 APM 系統上：當系統嘗試卸載錯誤模組時，系統會停止或不觸發暫停事件。如果您不卸載模組或停止服務 (導致無法成功暫停)，也可能發生此狀況。在兩種情況下，都請嘗試辨識無法啟動休眠模式的錯誤模組。在此狀況下，

在 /var/log/suspend2ram.log 和 /var/log/suspend2disk.log 中的省電精靈所產生的記錄檔非常有用。如果電腦無法進入休眠模式，原因在於最後卸載的模組。請操作 /etc/sysconfig/powersave/sleep 中的下列設定，在暫停或待命之前卸載有問題的模組。

```
UNLOAD_MODULES_BEFORE_SUSPEND2DISK=""
UNLOAD_MODULES_BEFORE_SUSPEND2RAM=""
UNLOAD_MODULES_BEFORE_STANDBY=""
SUSPEND2DISK_RESTART_SERVICES=""
SUSPEND2RAM_RESTART_SERVICES=""
STANDBY_RESTART_SERVICES=""
```

如果您在變動網路環境中或以遠端裝載檔案系統 (如 Samba 和 NIS) 的連接使用暫停或待命，請在上述變數中使用自動裝載器以裝載或新增對應服務，例如 smbfs 或 nfs。如果應用程式在暫停或待命之前存取遠端裝載檔案系統，則服務無法正確停止，而檔案系統無法正確取消裝載。在系統繼續之後，檔案系統可能損毀並必須重新裝載。

28.5.5 如需更多資訊

- /usr/share/doc/packages/powersave—本地省電精靈文件
- http://powersave.sourceforge.net—最新的省電精靈文件
- http://www.opensuse.org/Projects_Powersave—openSUSE wiki 中的專案頁面

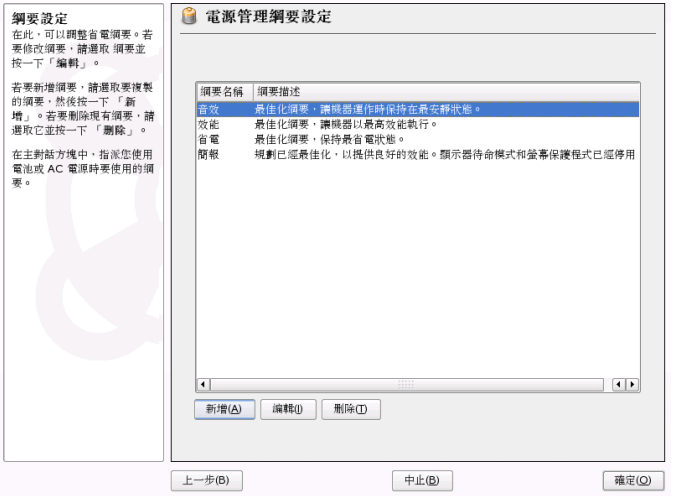
28.6 YaST電源管理模組

YaST 電源管理模組可以設定先前描述的所有電源管理設定。在您以「*系統*」>「*電源管理*」從「YaST 控制中心」啟動模組時，會開啟模組的第一個對話方塊 (請參閱 圖形 28.1「配置選取區域」 [480頁])。

圖形 28.1　配置選取區域

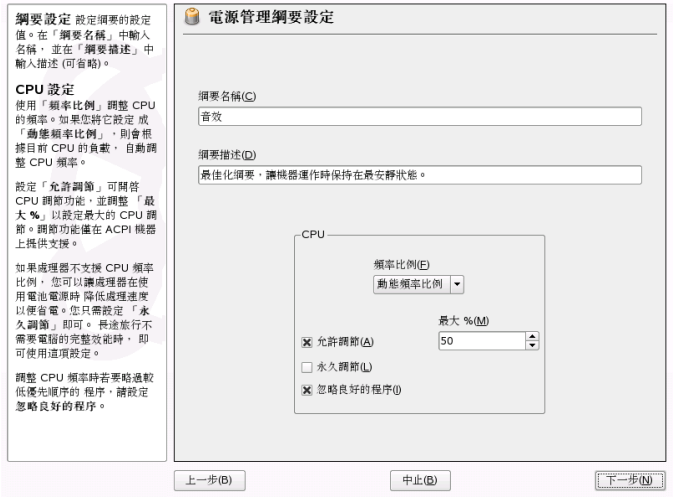
在此對話方塊中，選取電池操作和 AC 操作要使用的配置。若要新增或修改配置，請按一下「*編輯配置*」，便可開啟如圖形 28.2「現有配置的綜覽」[480頁]所示的現有配置綜覽。

圖形 28.2　現有配置的綜覽



在配置綜覽中，選取要修改的配置，再按一下「*編輯*」。若要建立新的配置，請按一下「*新增*」。兩種狀況開啟的對話方塊均相同，如圖形 28.3「設定配置」[481頁]中所示。

圖形 28.3 *設定配置*



首先，為新的或編輯的配置輸入適合的名稱和描述。判斷此配置是否應控制CPU效能及如何控制。決定是否應使用頻率比例和調節設定以及使用到什麼程度，以及是否應在調整 CPU 頻率時忽略低優先程度的處理器 (*niced* processes)。在硬碟的下列對話方塊中，定義最大效能或省電的「*待命規則*」。「*柔和式聲響規則*」控制硬碟的噪音強度 (少數硬碟支援)。「*冷卻規則*」決定要使用的冷卻方法。不幸地，BIOS 很少支援此類型的熱控制。請閱讀 `/usr/share/doc/packages/powersave/powersave_manual.html#Thermal` 瞭解如何使用風扇和被動冷卻方法。

也可以使用「*電池警告*」、「*ACPI 設定*」或「*暫停許可*」等起始對話方塊進行全域電源管理設定。按一下「*其他設定*」並從功能表選擇適當選項，以存取這些控制項。按一下「*電池警告*」以存取電池充電強度對話方塊，如圖形 28.4「電池充電強度」[482頁]中所示。

圖形 28.4　電池充電強度

當充電強度掉到特定可設定的限制之下時，您系統的BIOS會通知作業系統。在此對話中，定義三個限制：「*警告電量*」、「*低電量*」，與「*危險電量*」。指定當充電強度掉到此限制之下時，要觸發的動作。通常，前兩種狀態僅觸發對使用者的通知。第三種重要強度會觸發關機，因為剩餘電力不足以繼續系統操作。選取適合的充電強度和想要的動作，然後按一下「*確定*」以回到開始對話方塊。

圖形 28.5　ACPI 設定

使用「*ACPI 設定*」存取設定 ACPI 按鈕的對話方塊。如圖形 28.5「ACPI 設定」[483頁]所示。ACPI 按鈕的設定決定系統對特定切換應如何回應。設定系統對按下電源按鈕、按下休眠按鈕和關閉筆記型電腦蓋子加以回應。按一下「*確定*」以完成組態並回到開始對話方塊。

按一下「*啟用暫停*」以進入對話方塊，決定此系統使用者是否可使用暫停或待命功能，以及使用方式。按一下「*確定*」以回到主要對話方塊。再按一下「*確定*」以結束模組並確認您的電源管理設定。

29 無線通訊

無線 LAN 可用來建立 SUSE Linux Enterprise® 機器之間的通訊。本章將介紹無線網路的原則以及無線網路的基本組態。

29.1 無線區域網路

無線區域網路已成為行動運算世界中不可或缺的一環!如今，大多數的筆記型電腦都有內建的 WLAN 卡。WLAN 卡所使用的無線通訊 802.11 標準是由 IEEE 組織所制定。此標準最初用於最大傳輸率 2 MBit/s。其間已增加許多新的標準來提高資料傳輸率。這些補充項目定義調變、傳輸輸出及傳輸率等詳細資訊：

表格 29.1　WLAN 標準綜覽

名稱	頻段 (GHz)	最大傳輸率 (MBit/s)	記事
802.11	2.4	2	過時的；實際上無法取得終端設備
802.11b	2.4	11	普遍的
802.11a	5	54	較不普遍
28.29oz	2.4	54	與 11b 反向相容

此外還有一些專賣標準，像是德州儀器的 802.11b 變異標準，其最大傳輸率為 22 MBit/s (有時稱為 802.11b+)。不過使用此標準的網路卡數量有限。

29.1.1 硬體

SUSE Linux Enterprise®不支援 802.11 卡。但支援大部分使用 802.11a、802.11b 及 802.11g 的卡。新卡通常符合 802.11g 標準，但也有一些是 802.11b 的卡。一般來說，附有下列晶片的網路卡可受到支援：

- Aironet 4500、4800
- Atheros 5210、5211 及 5212
- Atmel at76c502、at76c503、at76c504 及 at76c506
- Intel PRO/Wireless 2100、2200BG、2915ABG、3945ABG
- Intersil Prism2/2.5/3
- Intersil PrismGT
- Lucent/Agere Hermes
- Texas Instruments ACX100 及 ACX111
- ZyDAS zd1201

有些極少人使用、且不再生產的舊網路卡產品也可受到支援。「*AbsoluteValue 系統*」的網站上可找到詳細的 WLAN 卡清單及其使用的晶片種類：http://www.linux-wlan.org/docs/wlan_adapters.html.gz在此尋找各種 WLAN 晶片的綜覽：http://wiki.uni-konstanz.de/wiki/bin/view/Wireless/ListeChipsatz

某些網路卡需要在啟動驅動程式時載入韌體影像。例如 Intersil PrismGT、Atmel 以及 TI ACX100 和 ACX111。使用 YaST 線上更新，即可輕易地安裝韌體。Intel PRO/Wireless 卡的韌體與 SUSE Linux Enterprise 一起提供，並且會在偵測到此類型的卡時，由 YaST 自動安裝。關於這個主角在安裝系統中的詳細資訊，請參閱 `/usr/share/doc/packages/wireless-tools/README.firmware`。

29.1.2 函數

使用無線網路時，您可以用各種不同的技術和組態來確保快速、高品質的安全連接。不同的作業類型適合不同的設定方式。選擇正確的驗證方法相當困難。可用的加密方法會有不同的優缺點。

操作模式

基本上，無線網路可分為管理網路和臨機操作網路。管理網路中有一個管理元件，即存取點。在此模式 (又稱為基礎結構模式) 之下，網路上所有 WLAN 工作站的連接都會通過該存取點，此存取點亦可連接至乙太網路。臨機操作網路中沒有存取點。工作站之間直接進行通訊。臨機操作網路中的傳輸範圍及連接工作站的數目相當有限。因此，存取點的效率通常較高。WLAN 卡甚至可做為存取點。大部份的網路卡均支援此項功能。

無線網路比有線網路更容易受到攔截和危害，因此各項標準均包含驗證和加密方式。在較早版本的 IEEE 802.11 標準中，可在 WEP 條款下找到這些項目的說明。然而，WEP 已證實不夠安全 (請參閱章節「安全性」[493頁])，WLAN 業者 (組成「*Wi-Fi 聯盟*」) 已定義一項新的安全標準，稱為 WPA，用來提高 WEP 的安全性。更新的 IEEE 802.11i 標準 (又稱為 WPA2，WPA 建立於 802.11i 草擬版本) 包含 WPA 及其他驗證和加密方式。

驗證

目前管理網路時會使用各種不同的驗證機制，以便確定只有獲得授權的工作站才可進行連接：

開啟

開放的系統，即不需要驗證的系統。任何工作站均可加入網路。但可使用 WEP 加密 (請參閱章節「加密」 [489頁])。

共用金鑰 (根據 IEEE 802.11)

此程序使用 WEP 金鑰進行驗證。不過，並不建議採用此程序，因為它使 WEP 金鑰更容易受到攻擊。攻擊者只需要截聽工作站與存取點間的通訊達足夠的時間就行了。在驗證過程中，雙方交換相同的資訊，先是以加密的形式，然後是以未加密形式。這樣就可以利用適合的工具重新建構金鑰。由於此方式是以 WEP 金鑰來驗證和加密，因此無法提高網路的安全性。擁有正確 WEP 金鑰的工作站可以驗證、加密及解密。缺乏金鑰的工作站則無法解密已收到的封包。因此，不論是否需要驗證其身份，該工作站均無法進行通訊。

WPA-PSK (根據 IEEE 802.1x)

WPA-PSK (PSK 即 Pre-Shared Key (預先共用金鑰) 的縮寫) 的作用方式與共用金鑰程序相似。所有連接工作站及存取點都要有相同的金鑰。此金鑰長度為 256 位元，且通常以密碼片語的方式輸入。本系統不需要如 WPA-EAP 一樣複雜的金鑰管理，並且更適合個人使用。因此，WPA-PSK 有時稱為 WPA「家用」。

WPA-EAP (根據 IEEE 802.1x)

WPA-EAP 實際上並非驗證系統，而是用來傳送驗證資訊的協定。WPA-EAP 用來保護企業中的無線網路。在私人網路中幾乎很少用到。因此，WPA-EAP 有時稱為 WPA「企業」。

WPA-EAP 需要 Radius 伺服器才能驗證使用者。EAP 提供三種不同方式連接與驗證伺服器：TLS (輸送層安全性)、TTLS (通道傳輸層安全性)，與 PEAP (保護擴展驗證協議)。在 Nutshell 中，這些選項的運作方式如下：

EAP-TLS

TLS 驗證仰賴伺服器和用戶端相互交換憑證。首先，伺服器會向評估它的用戶端提供其憑證。如果用戶端認為該憑證有效，就會接著向伺服器提供其憑證。雖然 TLS 是安全的，它仍需要網路中的工作憑證管理基礎結構。這種基礎結構在私有網路中很難找到。

EAP-TTLS 和 PEAP

TTLS 和 PEAP 都是兩階段的協定。第一個階段建立安全性，而第二個階段則交換用戶端驗證資料。它們需要的憑證管理負荷 (如果有的話) 遠低於 TLS。

加密

可使用各種不同的加密方式，防止未經授權者讀取無線網路中交換的資料封包，或進入網路：

WEP (定義於 IEEE 802.11 中)
: 此標準使用 RC4 加密演算法，最初的金鑰長度為 40 位元，後來增加為 104 位元。視 24 位元的啟始向量是否包含其中而定，其長度通常為 64 位元或 128 位元。然而此標準具有某些弱點。此系統所產生的金鑰可能受到攻擊。儘管如此，使用 WEP 仍然比完全不加密的網路來得好。

TKIP (定義於 WPA/IEEE 802.11i 中)
: 此金鑰管理協定定義於 WPA 標準中，使用與 WEP 相同的加密演算法，其弱點則均已消除。因為每個資料封包都有一個新的金鑰，所以攻擊這些金鑰等於白費力氣。TKIP 與 WPA-PSK 必須搭配使用。

CCMP (定義於 IEEE 802.11i 中)
: CCMP 說明金鑰管理。通常與 WPA-EAP 搭配使用，但也可配合 WPA-PSK 使用。根據 AES 的規定所進行的加密，比 WEP 標準下的 RC4 加密更安全。

29.1.3 使用 YaST 進行設定

若要設定您的無線網路卡，先啟動 YaST「*網路卡*」模組。在此您也可以選擇要使用 YaST 或 NetworkManager 來管理您的網路卡。若選擇 YaST，請在「*網路位址設定*」中選取設備類型「*無線*」，然後按一下「*下一步*」。在「*無線網路卡組態設定*」中 (顯示於 圖形 29.1「YaST：設定無線網路卡」 [490頁] 中)，請指定 WLAN 的基本操作設定：

*圖形 **29.1** YaST：設定無線網路卡*

操作模式

工作站可用三種模式來連接 WLAN。適合的模式根據所通訊的網路而有所不同：「*Ad-hoc*」(無存取點的對等網路)、「*Managed*」(由存取點管理的網路)，或「*Master*」(您的網路卡當作存取點使用)。若要使用任何 WPA-PSK 或 WPA-EAP 模式，就應該將作業模式設為「*管理模式*」。

網路名稱 (ESSID)

在無線網路中，所有工作站都要有相同的 ESSID 才能互相通訊。在未指定的情況下，網路卡會自動選擇一個存取點，該存取點可能不是您想要使用的。

驗證模式

請選擇適合您網路的驗證方式：「*開放*」、「*共用金鑰*」、「*WPA-PSK*」，或「*WPA-EAP*」。如果您選取 WPA 驗證，則必須設定網路名稱。

進階設定

此按鈕會開啟一個對話方塊，說明 WLAN 連接的組態設定細節。稍後會提供此對話方塊的詳細說明。

完成基本設定之後，您的工作站即可部署在 WLAN 中。

重要：無線網路的安全性

記得使用支援的驗證和加密方式，以保護您的網路資料傳輸。第三者可在未加密的 WLAN 連接上截取所有的網路資料。即使是不嚴密的加密 (WEP) 也比

不加密來得好。如需詳細資訊，請參閱章節「加密」[489頁]和章節「安全性」[493頁]。

視所選取的驗證方式而定，YaST 提示會要求您在其他對話方塊中微調您的設定。「*開放*」組態並不需要設定，因為它不需要驗證即可執行未加密的作業。

共用金鑰
: 設定金鑰輸入類型。選擇「*通關密語*」、「*ASCII*」或「*16 進位*」其中之一。您最多可使用四個金鑰來加密傳送的資料。按一下「*WEP 金鑰*」來進入金鑰組態對話方塊。設定金鑰長度：「*128 bit*」或「*64 bit*」。預設值為「*128 位元*」。在對話方塊底下的清單中，最多可指定四個金鑰，讓您的工作站用來加密。按「*設定為預設值*」將其中之一設為預設金鑰。除非您對此做變更，否則 YaST 都會使用第一個輸入金鑰為預設金鑰。如果標準金鑰被刪除了，則必須手動標記其他金鑰為預設金鑰。按一下「*編輯*」來修改現有的清單項目或建立新金鑰。在此例中，快顯視窗會提示您選擇一個輸入類型 (「*密碼片語*」、「*ASCII*」或「*16 進位*」)。如果您選取了「*密碼片語*」，請輸入一個單字或字元，會據此及先前指定的長度建立金鑰。「*ASCII*」必須輸入 5 個字元以建立 64 位元金鑰；輸入 13 個字元以建立 128 位元金鑰。「*16 進位*」則必須輸入 10 個字元以建立 64 位元金鑰，或 26 個字元以在 16 進位表示法中建立 128 位元金鑰。

WPA-PSK
: 若要輸入一個 WPA-PSK 金鑰，請選取「*密碼片語*」輸入法或「*16 進位*」。在「*密碼片語*」模式下，必須輸入 8 至 63 個字元。在「*16 進位*」模式下，必須輸入 64 個字元。

WPA-EAP
: 輸入網路管理員給您的身份證明。若為 TLS，請提供「*識別*」、「*用戶端憑證*」、「*用戶端金鑰*」以及「*伺服器憑證*」。TTLS 和 PEAP 需要「*識別*」和「*密碼*」。「*伺服器憑證*」和「*匿名識別*」是選擇性項目。YaST 會在 `/etc/cert` 下搜尋任何憑證，所以請將您的憑證儲存在這個位置，並將這些檔案的存取權限制為 `0600` (擁有者讀取和寫入)。

  按一下「*詳細資料*」來進入進階驗證對話方塊，以便設定 WPA-EAP。選擇 EAP-TTLS 或 EAP-PEAP 通訊之第二階段的驗證方法。如果您在上一個對話方塊中選取 TTLS，請選擇 `any`、`MD5`、`GTC`、`CHAP`、`PAP`、`MSCHAPv1` 或 `MSCHAPv2`。若您選取 PEAP，請選擇 `any`、`MD5`、`GTC` 或 `MSCHAPv2`。如果您無法使用自動決定的設定，請使用「*PEAP 版本*」來強制使用特定 PEAP 執行方式。

按一下「*進階設定*」以離開 WLAN 連接的基本設定對話方塊，並進入進階組態。此對話方塊包含下列選項：

通道
: 應只有「*臨機操作*」和「*主要*」模式會用到WLAN工作站的工作通道規格。在「*管理*」模式下，網路卡會自動搜尋可用的通道以連接存取點。在「*臨機操作*」模式中，必須從提供的 12 個通道中選取其中之一，讓您的工作站可與其他工作站通訊。在「*主要*」模式下，必須決定一個通道，讓您的網路卡可以用它來提供存取點功能。此選項的預設值為「*自動*」。

位元率
: 視您網路的效能而定，您可以設定點對點間特定的傳輸位元率。在預設值為「*自動*」的情況下，系統會試著選擇使用最高的資料傳輸率。有些 WLAN 卡不支援位元率的設定。

存取點
: 在擁有多個存取點的環境中，只要指定 MAC 位址即可預選其中一個存取點。

29.1.4 公用策

hostap (`hostap` 套件) 可用來使 WLAN 卡發揮存取點的功能。如需更多有關此套件的資訊，請參閱專案首頁 (`http://hostap.epitest.fi/`)。

kismet (`kismet` 套件) 是一種網路診斷工具，可用來傾聽 WLAN 的封包資料傳輸。您也可以藉此偵測您網路上任何侵入的企圖。如需更多相關資訊，請參閱 `http://www.kismetwireless.net/`及手冊。

29.1.5 設定 WLAN 的秘訣與技巧

這些秘訣可協助您調整 WLAN 的速度、穩定性及安全性。

穩定性及速度

無線網路的效能及可靠性，要看連接工作站是否能從其他工作站收到清楚的訊號。牆壁之類的障礙物會大大減弱訊號強度。訊號強度愈弱，傳輸速度愈慢。指令行 (連結品質 欄位) 上的 iwconfig 公用程式、NetworkManager 或

KNetworkManager，可檢查運作時的訊號強度。若您的訊強度出現問題，試著將您的設備安裝在其他地方，或調整您存取點的天線方向。許多 PCMCIA WLAN 卡都有輔助天線，可大幅提高接收度。由廠商指定的速率 (例如 54 MBit/s) 為一額定值，代表推定的最大值。實際上，最大資料產生量還不到該值的一半。

安全性

如果您要建立一個無線網路，請記得在缺乏安全措施的情況下，任何在傳輸範圍內的人都可輕易地進入您的網路。因此，應確定啟用加密方式。所有 WLAN 卡和存取點都支援 WEP 加密。雖然不是安全無虞，但仍足以阻礙可能的攻擊。WEP 通常足夠個人使用。WPA-PSK 則是更佳的選擇，不過較舊型的存取點或路由器並未在其 WLAN 功能中使用 PA-PSK。有些設備只要透過防火牆的更新，即可執行 WPA。此外，Linux 並不支援所有硬體元件上的 WPA。在開發此軟體時，WPA 只適用於使用 Atheros、Intel PRO/Wireless 或 Prism2/2.5/3 晶片的網路卡。就 Prism2/2.5/3 而言，只有在使用 hostap 驅動程式的情況下，才能使用 WPA (請參閱章節「Prism2 網路卡的問題」 [493頁])。如果沒有 WPA，則 WEP 仍然比完全不加密來得好。對於需要進階安全性的企業來說，只有在執行 WPA 的情況下才可操作無線網路。

29.1.6 疑難排解

如果您的 WLAN 卡沒有回應，檢查看看您是否已下載所需的韌體。請參閱第 29.1.1 節「硬體」 [486頁]。以下段落為您說明一些可能出現的問題。

多重網路設備

現代的筆記型電腦通常具備一張網路卡和一張 WLAN 卡。如果您以 DHCP 來設定這兩者 (自動指定位址)，則可能會出現名稱解析和預設閘道的問題。如果您可以偵測到路由器，卻無法瀏覽網際網路，表示已出現此問題。支援資料庫提供本主題的文章，網址是：http://en.opensuse.org/SDB:Name_Resolution_Does_Not_Work_with_Several_Concurrent_DHCP_Clients。

Prism2 網路卡的問題

有許多驅動程式可用於裝有 Prism2 晶片的設備。各種不同的網路卡多多少少都可以與不同的驅動程式配合運作。使用這些卡時，WPA 只能配合 hostap 驅動程

式來運作。如果這些網路卡無法順利運作，或完全無法運作，或者您想使用 WPA，請參閱 /usr/share/doc/packages/wireless-tools/README.prism2。

WPA

WPA 支援是 SUSE Linux Enterprise 中的新功能，仍在開發階段。因此，YaST 並未支援所有 WPA 驗證方法的組態。並非所有無線 LAN 卡和驅動程式都支援 WPA。某些卡需要韌體更新才能啟用 WPA。如果您要使用 WPA，請參閱 /usr/share/doc/packages/wireless-tools/README.wpa。

29.1.7 如需更多資訊

Linux「*無線工具*」的開發者 Jean Tourrilhes 在其網頁中提供大量有關無線網路的有用資訊。請參閱http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html。

IV. 服務

30

基本網路

Linux 提供所有必要的網路工具及功能，以整合到所有類型的網路結構。自定的 Linux 通訊協定、TCP/IP，具有各種服務與特殊功能，將會在這裏討論。使用網路卡、數據機或其他設備進行網路存取可以使用 YaST 來設定。也可使用手動方式來設定組態。本章節僅討論基本機制及相關的網路組態檔。

Linux 及其他 Unix 作業系統使用 TCP/IP 通訊協定。它不是單一網路通訊協定，而是能夠提供各種服務的網路通訊協定家族的一員。表格 30.1「TCP/IP 通訊協定家族中的數種通訊協定」[497頁]中列示的通訊協定，提供透過 TCP/IP 在兩個機器之間交換資料的用途。由 TCP/IP、全球網路所結合而成的網路，整體上就是指「網際網路」。

RFC 代表 *要求建議 (Request for Comments)*。RFC 是描述作業系統及其應用程式的各種網際網路通訊協定和執行程序的文件。RFC 文件描述網際網路通訊協定的設定。若要擴展您對於任何通訊協定的知識，請參閱適當的 RFC 文件。這些文件可從 `http://www.ietf.org/rfc.html` 線上取得。

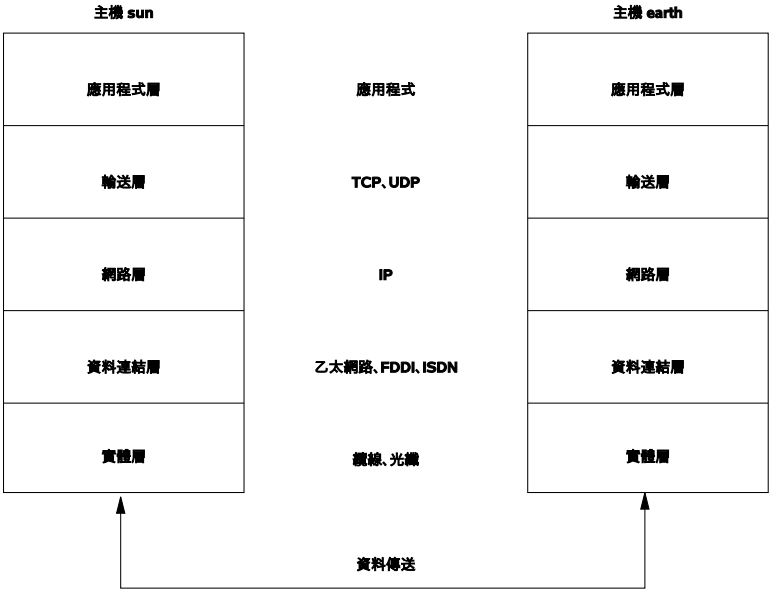
表格 30.1 *TCP/IP 通訊協定家族中的數種通訊協定*

協定	描述
TCP	傳輸控制通訊協定：連接導向的安全性通訊協定。傳輸的資料首先由應用程式當做資料流傳送出去，然後再由作業系統轉換為適當格式。資料以最初傳送的原始資料流格式，抵達到目的地主機的相關應用程式。TCP 會決定傳輸期間是否遺失任何資料，且沒有混亂。只要是資料順序很重要的地方，就會執行 TCP。

協定	描述
UDP	使用者資料包通訊協定：無連接、不安全的通訊協定。要傳送的資料以應用程式產生的封包形式加以傳送。不會保證資料在收件者端抵達的順序，而且可能遺失資料。UDP 適用以記錄為導向的應用程式。它的特點是延遲時間比 TCP 短。
ICMP	網際網路控制訊息通訊協定：基本上，這不是適用一般使用者的通訊協定，而是發佈錯誤報表的特殊控制通訊協定，能夠控制參與 TCP/IP 資料傳送之機器的行為。此外，它還提供特殊的回音模式，可以使用 ping 程式檢視。
IGMP	網際網路群組管理通訊協定：此通訊協定在實做 IP 多點廣播時控制機器行為。

如 圖形 30.1「TCP/IP 的簡化層模型」[499頁] 中所顯示，資料交換發生在不同層。實際的網路層是透過 IP (網際網路通訊協定，Internet Protocol) 進行不安全的資料傳輸。在 IP 的上方，TCP (傳輸控制通訊協定，Transmission Control Protocol) 可以保證資料傳輸某種程度的安全性。IP 層是由基本的硬體相依通訊協定所支援，例如乙太網路。

圖形 30.1　TCP/IP 的簡化層模型

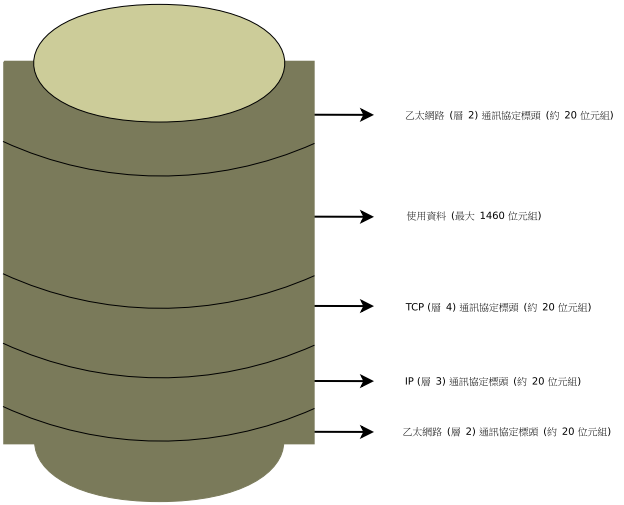


圖表提供每層的一或兩個範例。層的順序是依據*抽象階層 (abstraction level)*。最低層非常靠近硬體。不過，最上層對硬體而言幾乎是完全抽象的。每層都有自己的特殊功能。這些特殊功能通常隱含於其描述中。資料連結及實體層代表使用的實體網路 (如乙太網路)。

幾乎所有的硬體通訊協定都是採用封包導向模式。要傳輸的資料是裝在*封包*中，因為無法一次傳送它。TCP/IP 封包的大小上限約為 64 KB。封包一般而言較小，因為可能受限於網路硬體的關係。乙太網路上資料封包的最大上限約為 1500 個位元組。在乙太網路上傳送資料時，TCP/IP 封包的大小受限於此數量。如果傳送更多資料，則需要由作業系統傳送更多資料封包。

因為每層有自己指定的功能，關於每層的其他資訊必須儲存於資料封包中。這些資訊放在封包的「*標頭*」中。每層皆在產生的封包前端附加小的資料區塊，稱為通訊協定標頭。在乙太網路纜線上傳送的 TCP/IP 資料封包範例，可參閱在圖形 30.2「TCP/IP 乙太網路封包」[500 頁]中的說明。proof sum 位於封包結尾，不在開頭處。這樣可幫助網路硬體簡化程序。

圖形 30.2　TCP/IP 乙太網路封包



當應用程式在網路上傳送資料時，資料會經過每一層，除實體層外，全部在 Linux 核心執行。每層都負責準備資料使其能夠傳送到下一層。最底層最後要負責傳送資料。接收到資料時則反轉執行整個程序。就如同洋蔥的層級一般，在每層中，會從已傳輸的資料上移除通訊協定標頭。最後，傳輸層負責讓目的地端的應用程式可以使用資料。以這種方式，每層僅直接與上下兩層通訊。對於應用程式而言，無論資料是透過 100 MBit/s FDDI 網路或 56-Kbit/s 數據線進行傳輸，都沒有關係。同樣地，對於資料線而言，只要封包的格式正確，無論傳送的是哪種類型的資料也是無關的。

30.1 IP 位址與路由

在此節中的討論僅限於 IPv4 網路。如需有關 IPv6 通訊協定 (IPv4 的後繼者) 的資訊，請參閱第 30.2 節「IPv6—下一代的網際網路」 [503頁]。

30.1.1 IP 位址

網際網路上的每台電腦都有唯一的 32 位元位址。這些 32 位元 (或 4 位元組) 一般所寫入的格式，如 範例 30.1「寫入 IP 位址」[501頁] 中的第二列所述。

範例 30.1 *寫入 IP 位址*

```
IP Address (binary):  11000000 10101000 00000000 00010100
IP Address (decimal):      192.     168.       0.      20
```

採用十進位格式，四位元組以十進位數字系統撰寫，以句號分隔。IP 位址是指定給主機或網路介面。其他地方無法使用。此規則有例外狀況，但是與下文中並無關聯。

IP 位址中的點表示階層系統。直到 1990 年代，IP 位址仍嚴格地以類別加以分類。然而，此系統已證明太過死板而已停止採用。現在，則是使用*無類別路由 (classless routing)*，即 CIDR (無類別網域間路由，classless interdomain routing)。

30.1.2 網路遮罩與路由

網路遮罩是用來定義子網路的位址範圍。如果兩台主機位於相同的子網路遮罩，他們可以直接相互連結，如果他們不在同一個子網路遮罩，則需要閘道位址，以處理子網路和其他網路的通訊。若要檢查兩個 IP 位址是否位於同一子網路，只要使用網路遮罩「AND」兩個位址。如果結果相同，兩個 IP 位址位於同一個網路。如果不同，遠端的 IP 位址，即為遠端介面，只能透過閘道來通訊。

若要瞭解網路遮罩如何作用，請參閱範例 30.2「連結 IP 位址到網路遮罩」[502頁]。網路遮罩由 32 位元組成，可辨認 IP 位址屬於哪個網路。這些位元為 `1` 標示 IP 位址中的對應位元，即表示為同屬一個網路。所有位元為 `0` 標示位元在子網路內。這表示愈多位元為 `1`，子網路就愈小。因為網路遮罩永遠由多個連續的 `1` 組成，也可以計算網路遮罩內的位元數。範例 30.2「連結 IP 位址到網路遮罩」[502頁]中，第一個 24 位元的網路也可寫成 `192.168.0.0/24`。

範例 30.2 *連結 IP 位址到網路遮罩*

```
IP address (192.168.0.20):  11000000 10101000 00000000 00010100
Netmask    (255.255.255.0):  11111111 11111111 11111111 00000000
----------------------------------------------------------------
Result of the link:         11000000 10101000 00000000 00000000
In the decimal system:           192.     168.       0.       0

IP address (213.95.15.200): 11010101 10111111 00001111 11001000
Netmask    (255.255.255.0): 11111111 11111111 11111111 00000000
----------------------------------------------------------------
Result of the link:         11010101 10111111 00001111 00000000
In the decimal system:           213.      95.      15.       0
```

舉另外一個例子：使用相同乙太網路纜線連接的所有機器，通常位於同一個子網路中，而且可以直接存取。即使以交換器或橋接器實際分配子網路時，仍然可以直接連接這些主機。

位於本地子網路外的 IP 位址只能在設定目標網路的閘道時，才能與本地通訊。在大部分的狀況下，只能有一個閘道來處理所有對外的通訊。但是，您也可以為不同的子網路，設定多個閘道。

如果已經設定閘道，所有的外部 IP 封包會傳送到適當的閘道。然後此閘道會試圖以同樣方式傳送封包--主機對主機—直到連結到目標主機或封包 TTL (持續時間) 過期。

表格 30.2 *特定位址*

位址類型	描述
基本的網路位址	這是網路遮罩「及」網路中的任何位址，如 `Result` 下的範例 30.2「連結 IP 位址到網路遮罩」[502頁]所顯示。此位址不能指定給任何主機。
廣播位址	基本來說，即為「存取此子網路的所有主機」。若要產生此位址，網路遮罩會以二進位格式反轉，連結到具有邏輯 OR 的基本網路位址。因此以上範例會得到 192.168.0.255。此位址無法指派給任何主機。
本地主機	位址 `127.0.0.1` 是指定到每個主機上的「迴路設備」(loopback device)。使用此位址可以設定到您自己機器的連接。

因為 IP 位址在全世界必須是唯一的，您不能只選取隨機位址。如果要設立私人 IP 結構的網路，有三種位址網域可以使用。這些將無法從其他網際網路取得連結，因為他們無法透過網路傳送。這些位址網域在 RFC 1597 指定並列於 表格 30.3「私人 IP 位址網域」[503頁] 中。

表格 30.3 *私人 IP 位址網域*

網路/網路遮罩	網域
`10.0.0.0/255.0.0.0`	`10.x.x.x`
`172.16.0.0/255.240.0.0`	`172.16.x.x – 172.31.x.x`
`192.168.0.0/255.255.0.0`	`192.168.x.x`

30.2 IPv6—下一代的網際網路

重要：IBM System z：IPv6 支援

IBM System z 硬體的 CTC 和 IUCV 網路連線不支援 IPv6。

由於全球資訊網 (World Wide Web，WWW) 的出現，過去十五年內，在網際網路上透過 TCP/IP 進行通訊的電腦數目暴增。自從 CERN 的 Tim Berners-Lee (`http://public.web.cern.ch`) 於 1990 年發明 WWW 以來，網際網路主機的數量從幾千台成長為幾百萬台。

如前面所述，IPv4 位址僅由 32 個位元組成。而且，損失了一些 IP 位址—由於組織網路的方式，使得這些 IP 位址無法使用。您的子網路中可用的位址數目是位元數的平方減 2。例如，子網路有 2 個、6 個或 14 個位址可用。例如，如果要連接 128 個主機到網際網路，則子網路需要 256 個 IP 位址，但是其中只有 254 個可用，因為子網路結構本身需要用掉兩個 IP 位址：廣播與基本網路位址。

在目前的 IPv4 通訊協定之下，DHCP 或 NAT (網路位址轉譯，Network Address Translation) 是典型的機制，可用來避免位址可能不足的問題。搭配保持私人和公用位址空間分開的方式，能夠減輕短少的情形。其中產生的問題是在於其組態，設定麻煩且難於維護。若要在 IPv4 網路中設定主機，需要一些位址項目，如主機自己的 IP 位址、子網路遮罩、閘道位址，可能還需要名稱伺服器位址。您必須知道所有這些項目，且無法從其他地方取得。

透過 IPv6，位址短少及繁複組態的情形應該都成為過去式了。以下小節說明更多 IPv6 改善的部分及它帶來的好處，還有關於從舊通訊協定轉移到新通訊協定的資訊。

30.2.1 優點

新通訊協定帶來的最重要、最顯而易見的改善，是能夠大量擴充可用的位址空間。IPv6 位址是由 128 個位元值組成，而不是傳統的 32 位元。這樣提供了數以千兆的 IP 位址。

然而，IPv6 位址不僅是在長度方面與之前的位址不同；這些位置的內部結構也不同，可能包含有關系統及其所屬網路的更明確資訊。有關 IPv6 的詳細資訊，可以在 第 30.2.2 節「定址類型與結構」[505頁] 中找到。

以下是新通訊協定一些其他優勢的清單：

自動設定
: IPv6 讓網路能夠「隨插即用」(plug and play)，表示新設定的系統不需經過任何手動設定，即可整合到(區域)網路。新主機使用其自動設定組態機制，從鄰近的路由器上可用的資訊取得自己的位址，依賴的是稱為「*網路芳鄰探查*」(Neighbor Discovery，ND) 的通訊協定。這個方法不需要管理員的介入，而且不需要維護分配位址的中央伺服器，這是 IPv4 的另一個優勢，因為自動位址分配需要 DHCP 伺服器。

機動性
: IPv6 能夠同時將數個位址指定給一個網路介面。這樣可讓使用者輕鬆存取多個網路，有時可與行動電話服務公司提供的國際漫遊服務相比：當您帶您的行動電話出國時，一到對應區域，電話就會自動登入外國服務，因此仍可用同一個號碼聯絡到您，且您也可和在本國一樣撥打電話。

安全通訊
: 使用 IPv4，網路安全性是附加的功能。IPv6 包括 IPSec 為其中一個核心功能，允許系統在安全的通道上進行通訊，避免網際網路上的外人竊聽。

反向相容性
: 實際上，不可能一次將整個網際網路從 IPv4 切換到 IPv6。因此，很重要的是，兩個通訊協定不僅要能夠共存於網際網路上，也得要能夠共存於一個系統中。這是藉由相容的位址 (IPv4 位址可以輕易轉譯為 IPv6 位址) 以及使用一些通道來確保共存。請參閱第 30.2.3 節「IPv4 與 IPv6 的共存」[509頁]。另外，系統可以仰賴「*雙重堆疊 IP*」(Dual Stack IP) 技術，同時支援這兩種

通訊協定，這表示系統有兩個完全分開的網路堆疊，如此一來，兩種通訊協定版本不會相互干擾。

透過多重廣播自定量身訂做的服務
: 利用 IPv4，有些服務 (如 SMB) 需要廣播它們的封包到區域網路上的所有主機。IPv6 以更精細的方法，透過*「多重廣播」(multicasting)*—將一些主機定位為群組的一部分，讓伺服器定址主機 (這與透過*「廣播」(broadcasting)* 定位所有主機，或透過*「單點廣播」(unicasting)* 個別定址每個主機的方式不同)。定址為群組的主機，取決於具體的應用程式。例如，有些預先定義的群組可以定址所有名稱伺服器 (*「所有名稱伺服器多重廣播群組」*) 或所有路由器 (*「所有路由器多重廣播群組」*)。

30.2.2 定址類型與結構

如上述，目前 IP 通訊協定有兩個重要缺失：IP 位址的短缺越來越嚴重，以及設定網路與管理輪遞表成為複雜而繁重的任務。IPv6 透過擴充位址空間到 128 個位元解決了第一個問題。第二個問題的解決方式則是引入階層位址結構，結合配置網路位址的複雜技術，以及 *multihoming* (指定數個位址到一個設備，可以存取數個網路)。

使用 IPv6 時，瞭解三種不同類型的位址是很有用的：

單點廣播 (Unicast)
: 這類位址恰好與一個網路介面有關聯。這類位址的封包僅傳送到一個目的地。因此，單點廣播位址用來傳送封包到區域網路或網際網路上的個別主機。

多重廣播 (Multicast)
: 這類位址與一組網路介面有關聯。這類位址的封包會傳送到屬於該組的所有目的地。多重廣播位址主要由特定網路服務使用，可直接與特定主機群組通訊。

任點廣播 (Anycast)
: 這類位址與一組介面有關聯。這類位址的封包會根據基礎路由通訊協定的原則，傳送到最靠近傳送者的群組成員。使用任點廣播位址，讓主機更易於找出在指定網路區域中提供特定服務的伺服器。相同類型的所有伺服器擁有一樣的任點廣播位址。只要主機要求服務，它會從最靠近位置的伺服器接收回覆，由路由通訊協定決定。如果此伺服器因為某種原因失敗，通訊協定會自動選取第二個最靠近的伺服器，或是選取第三個伺服器，依此類推。

IPv6位址由八個四位數欄位組成，每個都代表16個位元，以十六進位標記法寫入。這些位址以冒號(:)分隔。指定欄位內的任何前導零位元組可以刪除，但是欄位內或尾端的零不能刪除。另一個慣例是多於四個連續的零位元組可能會摺疊成兩個冒號。然而，每個位址只允許一個 :: 。這類的簡略的標記法，顯示於範例 30.3「範例 IPv6 位址」[506頁]中，其中三行都是代表相同的位址。

範例 30.3 *範例 IPv6 位址*

```
fe80 : 0000 : 0000 : 0000 : 0000 : 10 : 1000 : 1a4
fe80 :    0 :    0 :    0 :    0 : 10 : 1000 : 1a4
fe80 :                           : 10 : 1000 : 1a4
```

IPv6位址的每個部分都有定義的功能。第一個位元組形成字首，指定位址類型。中間的部分是位址的網路部分，但是可能不會使用。位址的尾端形成主機部分。透過 IPv6，在位址尾端的斜線後表示字首的長度，可以定義網路遮罩。位址如範例 30.4「指定字首長度的 IPv6 位址」[506頁]中所示，包含的資訊是形成位址網路部分的前 64 個位元以及形成其主機部分的最後 64 個位元。換句話說，64 表示網路遮罩從左邊開始填入 64 個 1 位元值。就像 IPv4 一樣，IP 位址使用 AND 結合網路遮罩的值，判斷主機是否位於相同的子網路或另一個子網路。

範例 30.4 *指定字首長度的 IPv6 位址*

```
fe80::10:1000:1a4/64
```

IPv6知道關於數個字首的預定類型。有一些顯示在表格 30.4「各種IPv6字首」[506頁]中。

表格 30.4 *各種 IPv6 字首*

字首 (十六進位)	定義
00	IPv4 位址與透過 IPv6 的 IPv4 相容位址。這些位址用來維護與 IPv4 的相容性。其使用仍然要求路由器能夠轉譯 IPv6 封包為 IPv4 封包。數個特殊的位址，如迴路設備的位址，也有此字首。
2 或 3 做為第一個數字	可彙總的全域單點廣播位址。在 IPv4 的情形中，可以指定介面形成部分的特定子網路。目前有下列位址空間：2001::/16 (產品品質位址空間) 與 2002::/16 (6to4 位址空間)。

字首 (十六進位)	定義
`fe80::/10`	連結本地位址。具有這種字首的位址不應該傳送，因此僅能從相同的子網路內連接。
`fec0::/10`	本地網站位址。這些位址可以傳送，但是僅能在所屬組織的網路內傳送。事實上，它們等同於目前私人網路位址空間的 IPv6 (例如，`10.x.x.x`)。
`ff`	這些都是多重廣播位址。

單點廣播位址由三個基本元件組成：

公用拓撲 (Public Topology)
: 第一個部分 (也包含上述的其中一種字首) 用來透過公用網際網路傳送封包。它包含了有關提供網際網路存取的公司或機構資訊。

網站拓撲 (Site Topology)
: 第二個部分包含有關傳送封包的目的地子網路的傳送資訊。

介面識別碼 (Interface ID)
: 第三個部分識別傳送封包的介面。它也允許 MAC 形成部分的位址。這個前提是 MAC 在全球是唯一的，由硬體製造商在設備中編碼固定識別碼，可相當程度地簡化組態程序。事實上，結合前 64 個位址位元形成 `EUI-64` 記號，加上從 MAC 取得的最後 48 個位元，而其餘的 24 個位元則包含有關記號類型的特殊資訊。如此一來，就可以指定 `EUI-64` 記號給沒有 MAC 的介面，例如以 PPP 或 ISDN 為基礎的介面。

在此基本結構的最上層，IPv6 會分辨五種不同類型的單點廣播位址：

`::` (未指定的)
: 第一次啟始介面時，如果還未用其他方法判定位址時，主機會使用此位址做為其來源位址。

`::1` (迴路)
: 迴路設備的位址。

IPv4 相容位址
: IPv6 位址是由 IPv4 位址以及由 96 個零位元組成的字首形成的。這類相容性位址用於通道 (請參閱第 30.2.3 節「IPv4 與 IPv6 的共存」[509頁])，允許 IPv4 與 IPv6 主機在純 IPv4 環境中彼此通訊。

對應到 IPv6 的 IPv4 位址
: 這類位址以 IPv6 標記法指定純 IPv4 位址。

本地位址
: 有兩種位址類型用於本地：

  連結本地
  : 這類位址僅能用於本地子網路。具有此類型來源或目標位址的封包不應該傳送到網際網路或其他子網路。這些位址包含特殊字首 (`fe80::/10`) 以及網路卡的介面識別碼，加上由空位元組所組成的中間部分。自動設定組態以便與屬於相同子網路中的其他主機通訊時，會使用這類位址。

  網站本地
  : 具有這種位址的封包可以傳送到其他子網路，但是不能到更寬廣的網際網路，而必須保留在組織自己的網路內。這類位址用於內部網路，而且等同於 IPv4 所定義的私人位址空間。它們包含特殊字首 (`fec0::/10`)、介面識別碼，以及指定子網路識別碼的 16 位元欄位。同樣地，其他則是填入空位元組。

因為引入了 IPv6 這種全新的功能，所以每個網路介面通常會取得數個 IP 位址，其優點是可透過相同介面存取數個網路。其中一個網路可以設定完全自動化 (使用 MAC 和已知的字首)，只要一啟用 IPv6 (使用連結本地位址) 即可連接區域網路上的所有主機。利用形成位址部分的 MAC，全球使用的任何 IP 位址都成為唯一的。位址的唯一變數部分，是指定*網站拓撲*和*公用拓撲*，該部分視主機目前正在操作的實際網路而定。

如果主機要在不同的網路之間往返，至少需要兩個位址。其中一個，即*主位址*，不僅包含了介面識別碼，也包含了其通常所屬之主網路 (及其對應字首) 的識別碼。主位址是靜態位址，因此它通常不會變更。儘管如此，預定要送到行動主機的所有封包，還是可以傳送到主位址，無論是在主網路或其他外部網路中操作。這可藉由 IPv6 全新功能來達成，如「*無狀態自動設定*」與「*網路芳鄰探查*」。除了其主位址外，行動主機也取得一或多個其他的位址，這些位址屬於漫遊的外部網路。這些外部網路稱為 *care-of* 位址。主網路具有封包在外部漫遊時轉寄預定要送到主機的設備。在 IPv6 環境中，這個任務是由*主代辦*執行的，

它會取得所有預定要送到主位址的封包，透過通道轉送它們。另一方面，預定送到 care-of 位址的封包會直接傳送到行動主機，不會特別繞行。

30.2.3 IPv4 與 IPv6 的共存

連接網際網路的所有主機從 IPv4 轉移到 IPv6 是一種漸進程序。這兩種通訊協定某些時候會共存。在一個系統上共存，可保證執行兩種通訊協定的「*雙重堆疊*」。但是仍出現一些問題，就是使用 IPv6 的主機如何與 IPv4 主機通訊，以及由 IPv4 結構主導的目前網路如何傳輸 IPv6 封包。最佳的解決方案是提供通道及相容性位址 (請參閱第 30.2.2 節「定址類型與結構」 [505頁])。

IPv6 主機或多或少孤立於 (全球) IPv4 網路間，可透過通道通訊：IPv6 封包會被包成 IPv4 封包，在 IPv4 網路中移動。兩個 IPv4 主機之間的連接，稱為「*通道*」。若要完成這個目的，封包必須包含 IPv6 目的地位址 (或對應字首) 以及通道接收端上遠端主機的 IPv4 位址。基本通道可以根據主機管理員之間的協議「手動」設定；這也稱為「*靜態通道*」。

不過，靜態通道的組態及維護通常需要密集勞力，才能使用它們應付每天的通訊需求。因此，IPv6 提供三種不同的「*動態通道*」方法：

6over4
: IPv6 封包會自動封裝成 IPv4 封包，透過能夠多重廣播的 IPv4 網路進行傳送。IPv6 的訣竅是將整個網路 (網際網路) 視為一個大型的區域網路 (LAN)。如此即能自動判定 IPv4 通道的接收端。然而，這個方法不能適當的延伸，而且也因 IP 多重廣播目前在網際網路上並不普遍的事實而受到阻礙。所以，它僅能為啟用多重廣播的小型公司或機構的網路提供解決方案。這個方法的規格詳述於 RFC 2529。

6to4
: 利用此方法，IPv4 位址會自動從 IPv6 位址產生，使得隔離的 IPv6 主機能夠在 IPv4 網路上通訊。不過，有關這些隔離的 IPv6 主機及網際網路之間的通訊，目前已出現一些問題。該方法詳述於 RFC 3056。

IPv6 通道代理
: 這個方法仰賴提供 IPv6 主機專屬通道的特殊伺服器。詳述於 RFC 3053。

30.2.4 設定 IPv6

若要設定 IPv6，您通常不需要在個別工作站中做任何變更。IPv6 預設會開啟這個選項。您可在第 3.14.3 節「網路組態」 [35頁]所述的網路設定步驟中的安裝期間停用。若要在安裝的系統上停用或啟用 IPv6，請使用 YaST「*網路卡*」。勿變更方法，並按一下「*下一步*」。再選取網路卡、按一下「*位址*」索引標籤中的「*進階*」>「*IPv6*」。若要手動啟用 IPv6，請以 `root` 身份輸入 `modprobe ipv6`。

由於 IPv6 的自動組態概念，網路卡會在*連結本地*網路中指定一個位址。工作站通常不會進行路由表格管理。工作站可使用「*路由器通告通訊協定*」，向網路路由器查詢應使用的前置號碼和閘道。可使用 radvd 程式來設定 IPv6 路由器。此程式會通知工作站該 IPv6 位址應使用的前置號碼和路由器。或者，也可使用 zebra 自動設定位址和路由的組態。

請參閱 ifup 的手冊頁，瞭解如何使用 `/etc/sysconfig/network` 檔案來設定不同類型的通道。

30.2.5 如需更多資訊

上述綜覽沒有完整地涵蓋 IPv6 主題。如需更深入的探討這種新的通訊協定，請參閱以下線上文件和書籍：

`http://www.ipv6.org/`
所有有關 IPv6 的入門資訊。

`http://www.ipv6day.org`
啟動您 IPv6 網路所需的所有資訊。

`http://www.ipv6-to-standard.org/`
啟用 IPv6 產品的清單。

`http://www.bieringer.de/linux/IPv6/`
在此處可找到 Linux IPv6-HOWTO 和許多與此主題相關的連結。

RFC 2640
有關 IPv6 的基本 RFC。

IPv6 Essentials
: 描述此主題所有重要面向的書籍，《*IPv6 Essentials*》由 Silivia Hagen 所著 (ISBN 0-596-00125-8)。

30.3 名稱解析

DNS 協助指定 IP 位址給一或多個名稱以及指定名稱給 IP 位址。在 Linux 中，這種轉換通常是由已知為 bind 的特殊類型軟體執行的。處理這個轉換的機器稱為「*名稱伺服器*」(name server)。名稱組成階層系統，其中每個名稱元件以點分隔。但是，名稱階層與上述的 IP 位址階層無關。

考慮使用完整名稱，如 `earth.example.com`，以 `hostname.domain` 格式來表示。完整名稱，也就是*完整領域名稱 (Fully Qualified Domain Name，FQDN)*，是由主機名稱和領域名稱 (`example.com`) 組成的。後者也包含了「*最上層網域*」(top level domain)或 TLD (`com`)。

TLD 指定因為過去的緣故變得相當混淆。習慣上，美國使用三個字母的網域名稱。全世界的其他國家，則是使用兩個字母的ISO國際代碼為標準。除此之外，2000 年引入了較長的 TLD，代表特定活動範圍 (例如，`.info`、`.name`、`.museum`)。

在早期的網際網路 (1990 年前)，是使用檔案 `/etc/hosts` 儲存網際網路上所有機器的代表名稱。這種方式，對於連接到網際網路、快速增長的電腦數量層面而言，很快就證實是不切實際的。基於此因素，又開發出分散式的資料庫，以廣泛分散的方式來儲存主機名稱。這種資料庫與名稱伺服器類似，沒有有關網際網路上所有主機的立即可用資料，但是可以分散要求到其他名稱伺服器。

階層的最上層是由「*root 名稱伺服器*」(root name server)所使用。這些 root 名稱伺服器管理最上層網域，且由「網路資訊中心」(Network Information Center，NIC) 負責管理。每個 root 名稱伺服器知道負責指定最上層網域的名稱伺服器。有關最上層網域 NIC 的資訊可從 `http://www.internic.net` 取得。

DNS 的功能不只是解析主機名稱。名稱伺服器也知道哪個主機，即*「郵件交換器」(Mail Exchanger，MX)*，負責接收該網域的電子郵件。

若要讓您的機器能夠解析 IP 位址，它必須知道至少一個名稱伺服器及其 IP 位址。透過 YaST 的幫助可以輕鬆指定這類名稱伺服器。如果您使用數據機撥接連接，完全不需要手動設定名稱伺服器。撥接通訊協定在連接建立時會提供名

稱伺服器位址。如需關於設定 SUSE Linux Enterprise® 名稱伺服器存取權限的詳細資訊，請參閱第 33 章「*網域名稱系統*」[559頁]。

`whois` 通訊協定與 DNS 密切相關。利用此程式，可快速找出伺服器負責哪個指定網域。

注：MDNS 和 .local 網域名稱

`.local` 最上層網域將被解析程式視為連結本地網域。DNS 要求將做為多路廣播 DNS 要求予以傳送，而非通常的 DNS 要求。如果已在名稱伺服器組態中使用了 `.local` 網域，則必須在 `/etc/host.conf` 中關閉此選項。同時，請檢視 `host.conf` 手冊頁。

如果要在安裝期間關閉 MDNS，請使用 `nomdns=1` 做為開機參數。

如需有關多路廣播 DNS 的詳細資訊，請參閱 `http://www.multicastdns.org`。

30.4 使用 YaST 手動設定網路連線

Linux 可支援多種網路類型。大多數使用不同的設備名稱和組態檔，會分佈在檔案系統的不同位置。要更瞭解手動網路組態的綜覽，請參閱第 30.7 節「手動設定網路連線」[531頁]。

安裝期間，YaST 可以自動設定所有偵測到的介面。安裝後，可隨時在安裝系統設定其他的硬體。下列章節將說明 SUSE Linux Enterprise 支援之所有網路連結類型的網路組態。

提示：IBM System z：熱插式網路卡

IBM System z 平台支援熱插式網路卡，但不支援透過 DHCP 進行的自動網路整合 (和 PC 的情況相同)。完成偵測後，接著以手動設定介面。

30.4.1 使用 YaST 設定網路卡

若要在 YaST 中設定您的有線或無線網路卡，請選擇「*網路設備*」>「*網路卡*」。啟動模組後，YaST 會顯示一般網路組態對話方塊。選擇要使用 YaST 或 NetworkManager 來管理您的網路設備。若您要用 YaST 以傳統方式設定網路，

請勾選「*用 ifup 的傳統方式*」並按一下「*下一步*」。若要使用 NetworkManager，請勾選「*使用者以 NetworkManager 控制*」，並按一下「*下一步*」。如需有關 NetworkManager 的詳細資訊，請參閱 第 30.6 節「使用 NetworkManager 管理網路連線」[529頁]。

注：網路方法和 Xen

NetworkManager 無法搭配 Xen 運作。Xen 之中只有「*用 ifup 的傳統方式*」。

下一個對話上半部顯示一個清單，列出所有可以設定的網路卡。任何正確偵測的介面會在此列出名稱。若要變更所選設備的組態，請按一下「*編輯*」。無法偵測的設備，可依 章節「設定未偵測到的網路卡」[518頁] 中說明的方式使用「*新增*」來設定。

圖形 30.3 *設定網路卡*

變更網路卡組態

若要變更網路卡組態，請在 YaST 網路卡組態模組偵測到的網路卡清單中選擇，再按一下「*編輯*」。會出現「*網路卡組態設定*」對話，讓您從「*位址*」與「*一般*」索引標籤調整組態。如需關於無線網路卡組態的更多資訊，請參閱第 29.1.3 節「使用 YaST 進行設定」[489頁]。

設定 IP 位址

若可用的話，會在安裝時自動設定連接網路卡，以使用自動位址設定，DHCP。

注：IBM System z 和 DHCP

在 IBM System z 平台上，只有擁有 MAC 位址的網路卡才會支援 DHCP 式的位址組態。此情況只適用於 OSA 和 OSA 高速網路卡。

使用的 DSL 連接若未由 ISP 指定靜態 IP，則應該使用 DHCP。如果您決定使用 DHCP，請在「*DHCP 用戶端選項*」中設定其詳細資料。請選取「*進階*」>「*DHCP 選項*」，從「*位址*」索引標籤找出此對話方塊。指定 DHCP 伺服器是否應該執行廣播要求以及使用任何識別碼。如果有虛擬主機設定，透過相同介面與不同主機通訊，會需要使用識別子來分辨他們。

DHCP 對於用戶端組態是不錯的選擇，但不適用於伺服器組態。若要設定靜態 IP 位址，請如下執行：

1. 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。
2. 在「*位址*」索引標籤中選擇「*靜態位址設定*」。
3. 輸入「*IP 位址*」與「*子網路遮罩*」。
4. 按「*下一步*」。
5. 若要啟用組態，請按一下「*完成*」。

如果您使用靜態位址，就不會自動設定名稱伺服器和預設閘道。若要設定閘道，請按一下「*路由*」，然後新增預設閘道。若要設定名稱伺服器，請按一下「*主機名稱與名稱伺服器*」，然後新增名稱伺服器和領域的位址。

設定別名

一張網路卡可擁有多 IP 位址，稱為別名。若要設定您網路卡的別名，請如下執行：

1. 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。
2. 在「*位址*」索引標籤中，選擇「*進階*」>「*其他位址*」。

3 按一下「*新增*」。

4 輸入「*別名名稱*」、「*IP 位址*」和「*網路遮罩*」。

5 按一下「*確定*」。

6 再按一次「*確定*」。

7 按「*下一步*」。

8 若要啟用組態，請按一下「*完成*」。

設定主機名稱和 DNS

若您在安裝期間未變更網路組態，且有連接網路卡可用，則會自動為您的電腦產生主機名稱並啟用 DHCP。同時也會自動產生您主機要整合至網路環境所需的的名稱服務資訊。若網路位址設定使用 DHCP，則網域名稱伺服器清單會自動填入適當的資料。若您希望使用靜態設定，請手動設定數值。

若要變更您電腦的名稱並調整名稱伺服器搜尋清單，請如下執行：

1 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。

2 在「*位址*」清單中，按一下「*主機名稱與名稱伺服器*」。

3 若要停用 DHCP 驅動的主機名稱組態，請取消選擇「*透過 DHCP 變更主機名稱*」。

4 輸入「*主機名稱*」，且若需要的話，亦輸入「*網域名稱*」。

5 若要停用 DHCP 驅動的名稱伺服器清單更新，請取消選擇「*透過 DHCP 更新名稱伺服器與搜尋清單*」。

6 輸入名稱伺服器與網域搜尋清單。

7 按一下「*確定*」。

8 按「*下一步*」。

9 若要啟用組態，請按一下「*完成*」。

設定路由

若要讓您的電腦與其他電腦和其他網路通訊，必須提供路由資訊，以讓網路流量採取正確的路徑。若使用 DHCP，會自動提供此資訊。若使用靜態設定，必須手動新增此資料。

1. 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。
2. 在「*位址*」索引標籤中，按一下「*路由*」。
3. 輸入「*預設閘道*」的 IP。
4. 按一下「*確定*」。
5. 按「*下一步*」。
6. 若要啟用組態，請按一下「*完成*」。

新增特殊硬體選項

有時網路卡模組需要特殊參數才能正確運作。若要以 YaST 進行設定，請如下執行：

1. 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。
2. 在「*位址*」索引標籤中，按一下「*進階*」>「*硬體詳細資料*」。
3. 在「*選項*」中，輸入您網路卡的參數。若使用相同模組設定兩張網路卡，則兩張卡都會使用這些參數。
4. 按一下「*確定*」。
5. 按「*下一步*」。
6. 若要啟用組態，請按一下「*完成*」。

啟動設備

如果您使用 ifup 的傳統方法，就可以設定您的設備在下列時機啟動：開機時、連結纜線時、偵測到網路卡時、以手動方式啟動，或者永不啟動。若要變更設備啟動，請執行下列步驟：

1. 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。
2. 在「*一般*」索引標籤中，從「*設備啟用*」中選擇希望的項目。
3. 按「*下一步*」。
4. 若要啟用組態，請按一下「*完成*」。

設定防火牆

您不需輸入 第 43.4.1 節「以 YaST 設定防火牆」[750頁] 所說明的詳細防火牆資訊，僅需在設備設定時判斷基本防火牆設定即可。請執行下列步驟：

1. 從YaST網路卡組態模組偵測到的清單中選擇網路卡，並按一下「*編輯*」。
2. 進入網路組態對話中的「*一般*」索引標籤。
3. 決定您要為介面指派的防火牆區域。可用的選項如下：

   「*沒有區域，所有的流量已封鎖*」
   將會阻擋此介面的所有流量。

   「*內部區域 (未保護)*」
   會執行防火牆，但不強制任何規則以保護此介面。唯有您的機器位於受外部防火牆保護的更大網路中時，才能使用此選項。

   「*廢除區域*」
   廢除區域是內部網路與 (有潛在風險的) 網際網路之前的另一道防線。從內部網路與網際網路都可連接到指派至此區域的主機，但主機無法連存取內部網路。

   「*外部區域*」
   防火牆執行於此介面，且會徹底保護其免於網路流量的其他可能危險。此為預設選項。

4 按「*下一步*」。

5 按一下「*完成*」來啟用組態。

設定未偵測到的網路卡

您的網路卡有可能未正確偵測到。若是這樣的話，網路卡就不會出現在偵測到的網路卡清單中。若您確定您的系統具備網路卡的驅動程式，可手動設定。若要設定未偵測到的網路卡，請如下執行：

1 按一下「*新增*」。

2 以可用的選項(「*組態名稱*」和「*模組名稱*」)設定「*設備類型*」。如果網路卡是 PCMCIA 或 USB 設備，請啟用個別的核取方塊並使用「*下一步*」來結束對話方塊。否則，從「*從清單選擇*」選擇網路卡型號。接著 YaST 會為網路卡選擇適用的核心模組。

「*硬體組態名稱*」會指出 `/etc/sysconfig/hardware/hwcfg-*` 檔案的名稱，其中包含您網路卡的硬體設定。這包含核心模組名稱，以及所需的選項以啟始化硬體。

3 按「*下一步*」。

4 在「*位址*」索引標籤中，設定介面的設備類型、組態名稱與 IP 位址。若要使用靜態位址，請選擇「*靜態位址設定*」，再填妥「*IP 位址*」與「*子網路遮罩*」。您亦可從此選擇設定主機名稱、名稱伺服器和路由詳細資訊(請參閱章節「設定主機和DNS」[515頁]和章節「設定路由」[516頁])。

若您介面設備類型選擇「*無線*」，請在下一個對話設定無線連接。關於無線設備設定的詳細資料位於第 29.1 節「無線區域網路」[485頁]。

5 在「*一般*」索引標籤中，設定「*防火牆區域*」和「*設備啟用*」。透過「*使用者控制*」，授予一般使用者連接控制權。

6 按「*下一步*」。

7 若要啟用新網路組態，請按一下「*完成*」。

有關組態名稱慣例的資訊，請參閱 `getcfg(8)` 線上文件。

30.4.2 數據機

提示：IBM System z：數據機

IBM System z 平台不支援此類型的硬體組態。

在「YaST 控制中心」中，您可透過「*網路卡設備*」>「*數據機*」存取數據機組態。如果無法自動偵測到您的數據機，請按一下「*新增*」來開啟手動組態對話方塊。在開啟的對話方塊中，輸入數據機在「*數據機設備*」下用來連接的介面。

提示：CDMA 和 GPRS 數據機

請以 YaST 數據機模組來設定支援的 CDMA 和 GPRS 數據機，與您設定一般數據機時的方式一樣。

圖形 30.4　數據機組態

若使用專用交換機 (PBX)，您可能需要輸入撥號前置號碼。通常為 0。請參閱 PBX 隨附的說明。另外，請選擇是否使用按鍵式或轉盤式撥號、是否應該打開喇叭以及數據機偵測撥號音前是否應等待。如果數據機連接到分機，就不能啟用最後一個選項。

在「*詳細資料*」下，設定傳輸速率和數據機的啟始字串。只有在無法自動偵測您的數據機或是數據機需要特殊設定來傳輸資料時，才能變更這些設定。這些設定主要適用於 ISDN 終端機介面卡。按一下「*確定*」來結束此對話方塊。如果要在沒有 root許可權的情況下將數據機控制權委託給一般使用者，請啟用「*使用者控制*」。使用此方式，使用者即可啟用和停用介面，而不需要管理員的許可。在「*撥號前置號碼正規表示式*」下，指定一個正規表示式。KInternet 中的「*撥號前置號碼*」(一般使用者均可修改)，必須符合此正規表示式。如果此欄位為空白，則使用者必須有管理員許可才能設定不同的「*撥號前置號碼*」。

在下一個對話方塊中，選擇 ISP (網際網路服務提供者)。如果要從國內的 ISP 預先定義清單中選擇，請選取「*國家*」。或者，可按一下「*新增*」來開啟一個對話方塊，您可以在其中輸入您的ISP資料。這包括撥接連接名稱、ISP 名稱以及您的 ISP 提供的登入名稱和密碼。啟用「*永遠詢問密碼*」，以提示您在每次連接時輸入密碼。

在最後一個對話方塊中，可以指定其他的連接選項：

「*視需要撥號*」
如果您啟用了視需要撥號，請至少設定一個名稱伺服器。

「*連接時修改 DNS*」
此選項是依照預設啟用的，每次您連接到網際網路時，即會更新名稱伺服器位址。

「*自動取回 DNS*」
如果提供者沒有在連接後傳輸其領域名稱伺服器，則應停用此選項並手動輸入 DNS 資料。

「*簡易模式*」
此選項預設為啟用。使用它，就會忽略ISP伺服器送出的輸入提示，以避免干擾連接過程。

「*外部防火牆介面*」
選取此選項可啟用 SUSEfirewall2 並將介面設為外部。這樣就可讓系統在連接網際網路時免於外部攻擊。

「*閒置逾時 (秒)*」
使用此選項來指定網路靜止一段時間後即自動中斷數據機的連接。

「*IP 詳細資料*」
這會開啟位址組態對話方塊。如果您的 ISP 沒有為您的主機指定動態 IP 位址，請停用「*動態IP位址*」然後輸入您主機的本地 IP 位址和遠端 IP 位址。請向您的ISP詢問此資訊。讓「*預設路由*」保持為啟用狀態並選取「*確定*」來結束此對話方塊。

選取「*下一步*」回到原先的顯示數據機組態摘要的對話方塊。按一下「*完成*」以結束此對話方塊。

30.4.3 ISDN

提示：IBM System z：ISDN

IBM System z 平台不支援此類型的硬體組態。

使用此模組來為您的系統設定一個或多個 ISDN 卡。如果 YaST 無法自動偵測您的ISDN卡，請按一下「*新增*」來手動選取。可以有多個介面，但多個ISP 只能設定一個介面。在接下來的對話方塊中，設定網路卡正常運作所需的 ISDN 選項。

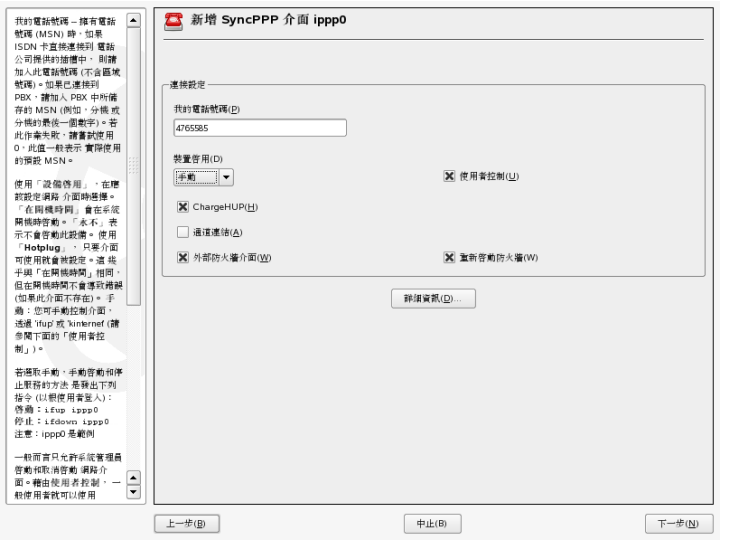
圖形 30.5 *ISDN 組態*

在下一個對話方塊中 (如 圖形 30.5「ISDN 組態」[521頁] 中所示)，選取要使用的通訊協定。預設為「*Euro-ISDN (EDSS1)*」，但如果是較舊或較大型的交換機，則選取「*1TR6*」。如果您是在美國，請選取「*NI1*」。在相關欄位中選取您的國家。其對應的國家代碼會顯示在旁邊的欄位。最後，提供您的「*區域號碼*」和「*撥號前置號碼*」(如有需要)。

「*設備啟用*」可定義 ISDN 介面的啟動方式：「*開機*」讓 ISDN 驅動程式在每次系統開機時起始。「*手動*」要求您以 `root` 身份使用 `rcisdn start` 指令來載入 ISDN 驅動程式。「*熱插*」用於 PCMCIA 或 USB 設備，會在插入設備後載入驅動程式。完成上述設定之後，請選取「*確定*」。

在下一個對話方塊中，指定您 ISDN 卡的介面類型，並將 ISP 新增至現有的介面。介面可能為 `SyncPPP` 或 `RawIP` 類型，但大多數 ISP 是在 `SyncPPP` 模式中操作，其說明如下。

圖形 30.6 *ISDN 介面組態*



視您的特殊設定而定，在「*我的電話號碼*」中必須輸入的號碼會有所不同：

ISDN 卡直接連至電話插孔
: 標準 ISDN 連接會提供 3 組電話號碼 (稱為多重用戶號碼或 MSN)。如果訂閱者要求更多，最多可以到 10，所有的 MSN 都必須在此輸入，但不需輸入

區域碼。如果您輸入錯誤的號碼，您的電話操作員會自動回復指定給您的 ISDN 連接的第一組 MSN。

連接至專用交換機的 ISDN 卡
此外，視安裝的設備而定，組態可能會有所不同：

1. 小型的專用交換機 (PBX) 大多使用 Euro-ISDN (EDSS1) 通訊協定來撥接內線電話。這些電話交換機有一個內部 S0 匯流排並在連接設備上使用內部號碼。

   使用其中一組內部號碼做為您的 MSN。您至少可使用一組電話交換機的 MSN，這些 MSN 已啟用可直接對外撥號。如果無法使用，請嘗試撥 0。如需更多資訊，請參閱您的電話交換機隨附的文件。

2. 較大型的商用電話交換機通常使用 1TR6 通訊協定來撥接內線電話。其 MSN 稱為 EAZ，通常可對應直撥號碼。如果要在 Linux 中設定組態，請輸入 EAZ 的最後一碼即可。最後一步是試著撥從 1 到 9的每個數字。

如果想在下一個收費單位開始之前終止連接，可啟用「*ChargeHUP*」。不過，要記得並不是每個 ISP 都適用。您也可以勾選對應的選項來啟用通道合併 (多重連結 PPP)。最後，您可以選取「*外部防火牆介面*」和「*重新啟動防火牆*」來啟用連接的 SUSEfirewall2。如果要讓一般使用者不需要管理員許可即可啟用和停用介面，請選取「*使用者控制*」。

「*詳細資料*」將開啟一個對話方塊，您可在其中設定回呼模式、與介面的遠端連接和其他 `ippd` 選項。選取「*確定*」以結束「*詳細資料*」對話方塊。

下一個對話方塊中可進行 IP 位址設定。如果您的提供者沒有給您一個靜態 IP，請選取「*動態 IP 位址*」。或者，可根據您 ISP 的規格，在提供的欄位中輸入您主機的本地 IP 位址和遠端 IP 位址。如果該介面應做為網際網路的預設路由，請選取「*預設路由*」。每個主機只能設定一個介面做為預設路由。選取「*下一步*」來結束此對話方塊。

下列對話方塊中可讓您設定您的國家和選取 ISP。清單中所列的 ISP 只有撥號計費提供者。如果您的 ISP 不在清單上，請選取「*新增*」。會開啟「*提供者參數*」對話方塊，您可在其中輸入您 ISP 的所有詳細資料。輸入電話號碼時，在數字之間不能有空白或逗號。最後，請輸入您的 ISP 所提供的登入名稱和密碼。完成後，請選取「*下一步*」。

如果要在獨立的工作站上使用「*視要求撥號*」，請指定名稱伺服器 (DNS 伺服器)。大多數 ISP 均支援動態 DNS，即在您每次連接時 ISP 都會送出一個名稱伺

服器的 IP 位址。不過，如果您使用的是單一工作站，您仍需提供一個替代位址，例如 `192.168.22.99`。如果您的 ISP 不支援動態 DNS，請指定 ISP 的名稱伺服器 IP 位址。如有需要，可為連接指定一個時限，即為;未使用網路的時間(以秒為單位)，之後即會自動終止連接。使用「*下一步*」來確認您的設定。YaST 會顯示已設定介面的摘要。如果要啟用這些設定，請選取「*完成*」。

30.4.4 纜線數據機

提示：IBM System z：纜線數據機

IBM System z 平台不支援此類型的硬體組態。

在某些國家 (奧地利、美國)，透過電視纜線來存取網際網路是相當普遍的。有線電視用戶通常會有一部數據機，一邊連接到電視纜線的輸出端子，另一邊 (使用 10Base-TG 雙絞纜線) 連接到電腦網路卡。接著纜線數據機會以固定的 IP 位址提供專用的網際網路連線。

視您的 ISP 所提供的說明而定，設定網路卡時可選取「*自動位址設定 (透過 DHCP)*」或「*靜態位址設定*」。現在大多數提供者都使用 DHCP。靜態 IP 位址通常是特殊商用帳戶的一部分。

有關纜線數據機組態的進一步資訊，請參閱「支援資料庫」文章中的相關主題，網址為 http://en.opensuse.org/SDB:Setting_Up_an_Internet_Connection_via_Cable_Modem_with_SuSE_Linux_8.0_or_Higher。

30.4.5 DSL

提示：IBM System z：DSL

IBM System z 平台不支援此類型的硬體組態。

如果要設定您的 DSL 設備，請從 YaST「*網路卡設備*」區段選取「*DSL*」模組。此 YaST 模組含有多個對話方塊，可根據下列其中一種通訊協定在其中設定 DSL 連接的參數：

- PPP over Ethernet (PPPoE)
- PPP over ATM (PPPoATM)

- CAPI for ADSL (Fritz 網路卡)
- 點對點通道通訊協定 (PPTP)—奧地利

PPPoE 或 PPTP 類型的 DSL 連接組態，會要求對應的網路卡必須正確設定。如果您尚未完成此步驟，請先選取「*設定網路卡*」來進行設定 (請參閱 第 30.4.1 節「使用YaST設定網路卡」[512頁])。如果是DSL連接，會自動指定位址但不是透過 DHCP，，因此您不可啟用「*自動位址設定 (透過 DHCP)*」選項，而是應該輸入介面的靜態虛擬位址，例如`192.168.22.1`。在「*子網路遮罩*」中，輸入 `255.255.255.0`。如果您要設定獨立的工作站，則應保留 「*預設閘道*」為空白。

提示

「*IP 位址*」和「*子網路遮罩*」中的值只是預留位置。只用來啟始化網路卡，而不代表 DSL 連接等等。

圖形 30.7 *DSL 組態*

如果要開始設定 DSL 組態 (請參閱 圖形 30.7「DSL 組態」[525頁])，請先選取 PPP 模式和用來連接 DSL 數據機的乙太網路卡 (大多為 `eth0`)。然後使用「*設備啟用*」來指定是否要在開機時建立 DSL 連接。按一下「*使用者控制*」來授權一般使用者，讓他們不需要 root 許可權即可透過 KInternet 來啟用或停用介面。您也可以使用該對話方塊來選擇您的國家和當地一些主控它的ISP。後續的DSL

組態對話方塊詳細資料會視目前已設定的選項而定，因此下列段落只提供簡單的介紹。如需可用選項的詳細資訊，請參閱對話方塊中的詳細說明。

如果要在獨立的工作站上使用「*視要求撥號*」，請指定名稱伺服器 (DNS 伺服器)。大部份 ISP 均支援動態 DNS，即為每次連接時，名稱伺服器送出的 IP 位址。不過，如果是單一工作站，請提供一個替代位址，例如 `192.168.22.99`。如果您的 ISP 不支援動態 DNS，請輸入您 ISP 提供的名稱伺服器 IP 位址。

「*閒置逾時 (秒)*」定義網路靜止一段時間後即自動中斷連接。合理的逾時秒數為 60 和 300 秒。如果停用「*視需要撥號*」，將逾時值設為零將有助於避免自動掛斷。

T-DSL 的組態非常類似 DSL 的設定。只要選取「*T-Online*」做為您的提供者，YaST 就會開啟 T-DSL 組態對話方塊。在此對話方塊中，提供一些 T-DSL 要求的其他資訊—線路 ID、T-Online 號碼、使用者代碼和您的密碼。在您加入 T-DSL 後，應該會收到這些資訊。

30.4.6 IBM System z：設定網路設備

IBM System z 用的 SUSE Linux Enterprise 支援多種不同類型的網路介面。您可使用 YaST 設定所有類型。

qeth-hsi 設備

若要在已安裝的系統中新增 `qeth-hsi` (Hipersocket) 介面，請啟動 YaST 網路卡模組 (「*網路設備*」>「*網路卡*」)。選取標記為「*IBM Hipersocket*」的設備做為「讀取」設備位址，然後按一下「*設定*」。在「*網路位址設定*」對話方塊中，指定新介面的 IP 位址和網路遮罩，然後按「*下一步*」和「*完成*」來結束網路組態。

qeth-ethernet 設備

如果要在已安裝的系統上新增 `qeth-ethernet` (IBM OSA 高速乙太網路卡) 介面，請啟動 YaST 網路卡模組 (「*網路設備*」>「*網路卡*」)。選取標記為「*IBM OSA 高速乙太網路卡*」的設備來做為「讀取」設備位址，然後按一下「*設定*」。輸入所需的連接埠名稱、一些其他選項 (請參閱 http://www.ibm.com/developerworks/linux/linux390/documentation_novell_suse.html 中的《*適用於 IBM System z 的 Linux：設備驅動程式、功能和指令*》手

冊)、IP 位址和適當的網路遮罩。使用「*下一步*」和「*完成*」來結束網路組態。

ctc 設備

如果要在已安裝的系統中新增 `ctc` (IBM 平行埠 CTC 介面卡) 介面，請啟動 YaST 網路卡模組 (「*網路設備*」>「*網路卡*」)。選取標記為「*IBM 平行埠 CTC 介面卡*」的設備做為您的讀取通道，然後按一下「*設定*」。選擇適合設備的「*設備設定*」(通常為「*相容模式*」)。指定您和遠端合作夥伴的 IP 位址。如有需要，可透過「*進階*」>「*細節設定*」來調整 MTU 的大小。使用「*下一步*」和「*完成*」來結束網路組態。

警告

不建議使用此介面。未來版本的 SUSE Linux Enterprise 將不支援此介面。

lcs 設備

如果要在已安裝的系統中新增 `lcs` (IBM OSA-2 介面卡) 介面，請啟動 YaST 網路卡模組 (「*網路設備*」>「*網路卡*」)。選取標示為「*IBM OSA-2 介面卡*」的設備，然後按一下「*設定*」。輸入所需的連接埠名稱、一些其他選項 (請參閱 `http://www.ibm.com/developerworks/linux/linux390/documentation_novell_suse.html` 中的《*適用於 IBM System z 的 Linux：設備驅動程式、功能和指令*》手冊)、IP 位址和適當的網路遮罩。使用「*下一步*」和「*完成*」來結束網路組態。

IUCV 設備

如果要在已安裝的系統中新增 `iucv` (IUCV) 介面，請啟動 YaST 網路卡模組 (「*網路設備*」>「*網路卡*」)。選取標示為「*IUCV*」的設備，然後按一下「*設定*」。YaST 會提示您輸入 IUCV 合作夥伴的名稱。輸入名稱 (本項目區分大小寫) 然後選取「*下一步*」。指定您和合作夥伴的 IP 位址。如有需要，可透過「*進階*」>「*細節設定*」來調整 MTU 的大小。使用「*下一步*」和「*完成*」來結束網路組態。

警告

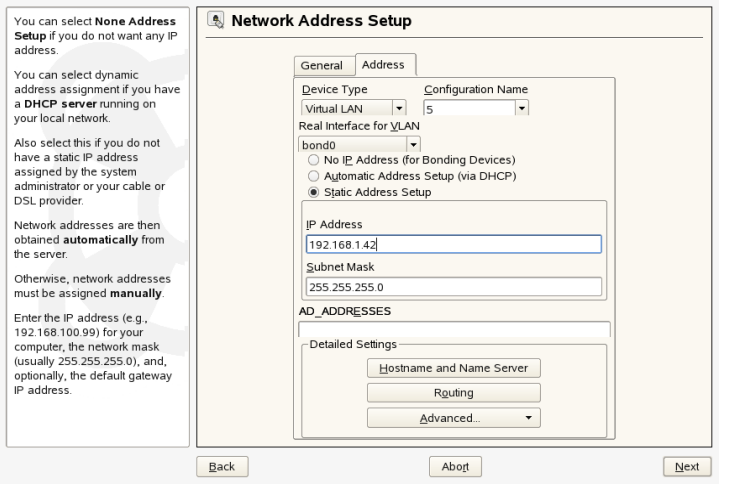
不建議使用此介面。未來版本的 SUSE Linux Enterprise 將不支援此介面。

30.5 在 SUSE Linux 上設定 VLAN 介面

VLAN 是*虛擬區域網路* (Virtual Local Area Network) 的縮寫。它允許多個*邏輯* (虛擬) 乙太網路在一個單一實體乙太網路上執行，會在邏輯上將網路分割成不同的廣播網域，以便封包只可在為同一 VLAN 指定的連接埠之間進行交換。若要在網路設定中使用 VLAN，請確定已安裝套件 `vlan`。

若 Linux 的網路連線不是專用於特定的邏輯 LAN，您可以設定對其中一或多個邏輯 LAN 的存取權限。您也可以透過所有其他網路介面所用的一般 ifup 與 ifdown 程序檔來支援 VLAN 介面組態。YaST 支援對 VLAN 設備進行設定。

圖形 30.8 *YaST VLAN 組態*



執行 YaST 模組「*網路設備*」>「*網路卡*」，選取「*使用 ifup 的傳統方法*」並按「*下一步*」。遵循以下程序設定 VLAN 設備：

過程 30.1 *使用 YaST 設定 VLAN 介面*

1 按「*新增*」建立新的網路介面。

2 在「*網路組態*」中，選取「*設備類型*」>「*虛擬 LAN*」。

3 將「*組態名稱*」的值變更為 VLAN 的 ID。請注意，VLAN ID 1 通常用於管理用途。

4 按「*下一步*」。

5 在「*VLAN 的實際介面*」下方選取 VLAN 設備應連接的介面。

6 選取為 VLAN 設備指定 IP 位址時所需使用的方法。

7 按「*下一步*」完成組態。

如需 VLAN 的詳細資訊，請參閱 http://www.candelatech.com/~greear/vlan.html 以及位於 /usr/share/doc/packages/vlan/ 中的套件文件。

30.6 使用 NetworkManager 管理網路連線

NetworkManager 是行動工作站理想的解決方案。如果使用 NetworkManager，則當您的位置變更時，就不需要擔心網路介面的設定以及網路之間的切換事宜。NetworkManager 能夠自動連接到已知的 WLAN 網路。如果您有兩種或更多種連接方式，它還能夠連到速度較快的連接。

NetworkManager 解決方案不適用於下列情況：

- 您想要一個介面使用多個撥號提供者。
- 您的電腦是網路的路由器。
- 您的電腦會為網路的其他電腦提供網路服務，例如，它是 DHCP 或 DNS 伺服器。
- 您的電腦為 Xen 伺服器，或您的系統是 Xen 之中的虛擬系統。
- 您要使用 SCPM 來健行網路組態管理。若同時使用 SCPM 和 NetworkManager，SCPM 就無法控制網路資源。

- 您想要同時使用多個使用中的網路連線。

若要在安裝期間啟用或停用 NetworkManager，請在「*網路組態*」的「*網路模式*」中按一下「*啟用 NetworkManager*」或「*停用 NetworkManager*」。若要在已安裝的系統上啟用或停用 NetworkManager，請執行下列步驟：

1. 開啟 YaST。
2. 選擇「*網路設備*」>「*網路卡*」。
3. 在第一個畫面上，將「*網路設定方法*」選項設定為「*使用者以 NetworkManager 控制*」，以使用 NetworkManager。若要停用 NetworkManager，則將「*網路設定方法*」選項設定為「*用 ifup 的傳統方式*」。

選擇方法後，請透過 DHCP 或靜態 IP 位址而使用自動組態來設定您的網路卡，或請設定您的數據機。如需使用 YaST 設定網路組態的詳細資訊，請參閱第 30.4 節「使用 YaST 手動設定網路連線」[512頁]和第 29.1 節「無線區域網路」[485頁]。直接在 NetworkManager 中設定支援的無線網路卡。

若要設定 NetworkManager，請使用 NetworkManager Applet。KDE 和 GNOME 都擁有自己的 NetworkManager Applet。適當的 Applet 應該會跟著桌面環境一起自動啟動。然後 Applet 會在系統匣成為一個圖示。這些 Applet 的功能類似，但介面稍有不同。它們也可以用在有標準系統匣支援的其他圖形環境。

30.6.1 ifup 與 NetworkManager 的差異

如果您使用 NetworkManager 設定網路，就可以隨時使用 Applet 在您的桌面環境上輕鬆切換、停止或啟動網路連線。NetworkManager 也讓您不需要 `root` 權限就可以變更和設定無線網路卡連接。因此，NetworkManager 是行動工作站理想的解決方案。

傳統使用 ifup 設定組態時，雖然也有一些方法可以在有或沒有使用者互動 (例如由使用者管理的設備) 的情況下切換、停止或啟動連接，但是一定要有 `root` 權限才能變更或設定網路設備。這對行動計算而言，往往會造成問題，因為不可能預先定義好所有可能的連接方式。

不論是傳統組態或 NetworkManager 都能夠使用 DHCP 和靜態組態，來處理無線網路 (WEP、WPA-PSK 和 WPA-Enterprise 存取)、撥號和有線網路的網路連線。它們也支援透過 VPN 的連接。

NetworkManager 會嘗試使用最好的連接，讓您的電腦隨時保持連接狀態。如果可能，它會使用最快的有線連接。如果網路纜線突然中斷，它會嘗試重新連接。它可以從您的無線連接清單中找到訊號最強的網路，並自動用它來連接。如果要用 ifup 達到相同的功能，必須執行很多組態工作。

30.6.2 如需更多資訊

如需 NetworkManager 的詳細資訊，請參閱下列網站和目錄：

- http://www.gnome.org/projects/NetworkManager/—NetworkManager 專案頁面
- http://en.opensuse.org/Projects/KNetworkManager—NetworkManager KNetworkManager 專案頁面

30.7 手動設定網路連線

網路軟體的手動組態應該永遠是最後的替代方案。建議使用 YaST。不過，這個有關網路組態的背景資訊也可協助您使用 YaST。

所有內建的網路卡以及熱插式網路卡 (PCMCIA、USB、一些 PCI 卡) 會透過熱插方式進行偵測到並加以設定。系統將網路卡視為兩種不同的項目，首先是視為實體設備，再來是視為介面。插入設備或偵測到設備時則會觸發熱插事件。此熱插事件使用程序檔 `hwup` 觸發設備的啟始化。啟始化網路卡作為新的網路介面時，核心會產生另一個熱插事件，使用 `/ifup` 觸發介面的設定。

核心根據介面名稱註冊的暫時順序加以編號。啟始化順序是指定名稱的決定因素。如果數個網路卡中的一個失敗，所有依序啟始化網路卡的編號就會改變。對於真正的可熱插式網路卡，設備連接的順序才是決定因素。

為了讓組態具有彈性，已經分開設備 (硬體) 及介面的組態，而且不再以介面名稱來管理組態與設備和介面之間的對應。設備組態位於 `/etc/sysconfig/hardware/hwcfg-*`。介面組態位於 `/etc/sysconfig/network/ifcfg-*`。組態名稱是以描述設備及其關聯介面的方式來指定的。因為之前驅動程式與介

面名稱的對應需要靜態介面名稱，所以此對應不再發生於 /etc/modprobe.conf。在這種新的概念中，此檔案中的別名項目造成不想要的副作用。

組態名稱 (hwcfg- 或 ifcfg- 之後的任何項目) 都可以透過插槽、設備專用 ID 或介面名稱來描述設備。例如，PCI 卡的組態名稱可以是 bus-pci-0000:02:01.0 (PCI 插槽) 或 vpid-0x8086-0x1014-0x0549 (廠商和產品 ID)。相關聯介面的名稱可以是 bus-pci-0000:02:01.0 或 wlan-id-00:05:4e:42:31:7a (MAC 位址)。

若要指定特定網路組態到任何特定類型的卡 (一次僅能插入一種) 而不是特定卡，請選取較不特定的組態名稱。例如，bus-pcmcia 可以用於所有 PCMCIA 卡。在另一方面，之前的介面類型會限制名稱。例如，wlan-bus-usb 可以指定給連接到 USB 埠的 WLAN 卡。

系統永遠使用最佳描述介面或提供介面之設備的組態。搜尋最適用組態是由 getcfg 處理的。getcfg 的輸出會傳達可以用來描述設備的所有資訊。有關組態名稱規格的詳細資料，請參閱 getcfg 的手冊頁。

透過描述的方法，即使網路設備不一定永遠以相同的順序啟始化，網路介面還是可以具有正確的組態設定。不過，介面的名稱仍然取決於啟始化順序。有兩種方式可以確保確實存取特定網路卡介面：

- getcfg-interface *configuration name* 傳回相關聯網路介面的名稱。因此，可以輸入一部份的組態名稱 (例如，防火牆、dhcpd、路由、各種虛擬網路介面 (通道))，取代介面名稱，因為這種名稱並不是永久不變的。
- 永久介面名稱會自動指定給每個介面。您可以自行調整以符合您的需要。建立介面名稱時，請依照 /etc/udev/rules.d/30-net_persistent_names.rules 中概述的方式執行。然而，永久名稱 *pname* 不能與核心自動指定的名稱相同。所以，不允許 eth*、tr*、wlan*、qeth*、iucv* 等名稱。相反地，請使用 net* 或是 external、internal、dmz 之類的描述名稱。必須確定不要重複使用相同的介面名稱。介面名稱中可用的字元僅限於 [a-zA-Z0-9]。永久名稱僅能在註冊後立即指定給介面，這表示必須重新載入網路卡的驅動程式或執行 hwup *device description*。針對此用途，僅使用指令 rcnetwork restart 是不夠的。

重要：使用永久介面名稱

永久介面名稱的使用尚未測試於所有方面。因此，一些應用程式可能無法自由地處理選取的介面名稱。

ifup需要現有介面，因為它不會啟始化硬體。硬體的啟始化是由指令hwup(由hotplug或coldplug執行)處理的。啟始化設備時，會透過hotplug自動為新介面執行ifup，如果開始模式是onboot、hotplug或auto，就會設定介面，而且會啟動network服務。之前，是使用指令ifup *interfacename*來觸發硬體啟始化。現在已經反轉此程序。首先，會啟始化硬體元件，接著是所有其他的動作。採取這種方式，則永遠可以使用現有的組態集，儘可能以最佳的方式設定不同數目的設備。

表格 30.5「手動網路組態程序檔」[533頁]總結了與網路組態相關的最重要程序檔。只要可能，會依硬體和介面區分程序檔。

表格 30.5 *手動網路組態程序檔*

組態階段	指令	函數
硬體	hw{up,down,status}	hw* 程序檔由熱插式子系統執行，用來啟始化設備、復原啟始化、或查詢設備狀態。詳細資訊請參閱hwup的手冊頁。
介面	getcfg	getcfg可以用來查詢與組態名稱或硬體描述相關聯的介面名稱。詳細資訊請參閱getcfg的手冊頁。
介面	if{up,down,status}	if* 程序檔會開啟現有網路介面或傳回指定介面的狀態。詳細資訊請參閱ifup的手冊頁。

有關熱插拔和永久設備名稱的詳細資訊，請參閱第 24 章「*使用 udev 進行動態核心設備管理*」[423頁]。

30.7.1 組態檔案

本節提供網路組態檔的綜覽，並說明其用途和使用的格式。

/etc/syconfig/hardware/hwcfg-*

這些檔案包含網路卡及其他設備的硬體組態；其中包含所需的參數，如核心模組、啟動模式和程序檔關聯。詳細資訊請參閱 `hwup` 的手冊頁。無論現有硬體為何，啟動 coldplug 時會套用 `hwcfg-static-*` 組態。

/etc/sysconfig/network/ifcfg-*

這些檔案包含網路介面的組態；其中包含啟動模式和 IP 位址等資訊。可以使用的參數請參閱 `ifup` 的手冊頁。此外，如果一般設定僅用於一個介面，則檔案 `dhcp`、`wireless` 和 `config` 中的所有變數都可以用於 `ifcfg-*` 檔案。

► **zseries:** IBM System z 不支援 USB。介面檔名稱和網路別名包含 `qeth` 等 System z 專屬元件。 ◄

/etc/sysconfig/network/{config,dhcp,wireless}

檔案 `config` 包含 `ifup`、`ifdown`、和 `ifstatus` 行為的一般設定；`dhcp` 則包含無線區域網路介面卡之 DHCP 和 `wireless` 的設定。所有三個組態檔中的變數都已註解，而且可以在 `ifcfg-*` 檔案中使用，以更高優先順序處理。

/etc/sysconfig/network/{routes,ifroute-*}

TCP/IP 封包的靜態路由在此決定。您可以將各種系統任務所需的靜態路由都輸入 `/etc/sysconfig/network/routes` 檔案：主機的路油、透過閘道前往主機的路由，以及網路的路由。對於需要個別路由的介面，請定義額外的組態檔案：`/etc/sysconfig/network/ifroute-*`。以介面的名稱取代 `*`。在路由組態檔中的項目看起來就像這樣：

```
# Destination     Dummy/Gateway     Netmask           Device
#
127.0.0.0         0.0.0.0           255.255.255.0     lo
204.127.235.0     0.0.0.0           255.255.255.0     eth0
default           204.127.235.41    0.0.0.0           eth0
```

```
207.68.156.51     207.68.145.45     255.255.255.255    eth1
192.168.0.0       207.68.156.51     255.255.0.0        eth1
```

路由的目的地是在第一個資料欄。這個資料欄可能包含網路或主機的 IP 位址，這是指「*可到達的*」名稱伺服器、完整合格的網路或主機名稱。

第二個資料欄包含預設的閘道或是可以存取主機或網路的閘道。第三個資料欄包含在閘道後面的網路或主機的網路遮罩。例如，遮罩為 `255.255.255.255`，供在閘道後面的主機使用。

第四欄只與連接至本地主機的網路相關，例如迴路、乙太網路、ISDN、PPP 以及虛擬設備。必須在這裏輸入設備名稱。

第五欄 (可選) 可指定路由的類型。不需要的欄位必須包含 - 減號，以確保解析程式可正確解譯指令。如需詳細資訊，請參閱 `routes(5)` 線上文件。

/etc/resolv.conf

主機所屬的網域指定於此檔案 (關鍵字 `search`)；另外也會列出要存取的名稱伺服器位址的狀態 (關鍵字 `nameserver`)。可以指定多個領域名稱。解析不完整的名稱時，會嘗試附加個別 `search` 項目產生一個名稱。透過輸入數行且每行都以 `nameserver` 開頭的方法，可使用多個名稱伺服器。在註解前加上 `#` 符號。YaST 在此檔案中輸入指定名稱伺服器。範例 30.5「`/etc/resolv.conf`」[535頁] 顯示出 `/etc/resolv.conf` 可能的樣子。

範例 30.5 *`/etc/resolv.conf`*

```
# Our domain
search example.com
#
# We use sun (192.168.0.20) as nameserver
nameserver 192.168.0.20
```

一些如 `pppd` (`wvdial`)、`ipppd` (`isdn`)、`dhcp` (`dhcpcd` 和 `dhclient`)、`pcmcia` 及 `hotplug` 之類的服務會修改檔案 `/etc/resolv.conf`，方法是使用程序檔 `modify_resolvconf`。如果檔案 `/etc/resolv.conf` 已經由此程序檔暫時修改，它會包含預先定義的註解，提供的資訊包含修改它的服務、原始檔案的備份位置以及如何關閉自動修改機制。如果數次修改 `/etc/resolv.conf`，檔案會以巢狀形式包含所作的修改。即使反轉時使用與修改順序不同

的順序，還是可以完全反轉此程序。需要這種彈性的服務包括 isdn、pcmcia、hotplug。

如果沒有以正常的方式終止服務，可以使用 modify_resolvconf 還原原始檔。另外，在系統開機時，會執行檢查，看看是否有沒有清理、已修改的 resolv.conf (例如，系統當機後)，在這種情況下，會還原原始 (未修改) 的 resolv.conf。

YaST 使用指令 modify_resolvconf check 找出是否已修改 resolv.conf，接著將警告使用者還原檔案會遺失所有變更。除此之外，YaST 不會依靠 modify_resolvconf，意即會透過 YaST 變更 resolv.conf 的影響與手動變更的影響是一樣的。無論是哪種情形，變更永遠是有效的。上述提及的服務所需的修改只是暫時的。

/etc/hosts

在此檔中 (請參閱 範例 30.6「/etc/hosts」[536頁])，IP 位址是指定給主機名稱。如果沒有執行任何名稱伺服器，將使用此 IP 連接設定的所有主機將列示於此。對於每個主機，分別在檔案中輸入一行包含 IP 位址、完全合法的主機名稱及主機名稱的項目。IP 位址必須在行的開頭，然後以空格和定位點分隔這些項目。註解的前面永遠是 # 符號。

範例 30.6 */etc/hosts*

```
127.0.0.1 localhost
192.168.0.20 sun.example.com sun
192.168.0.1 earth.example.com earth
```

/etc/networks

在此檔中，網路名稱會轉換為網路位址。格式與 hosts 檔案格式相似，但是網路名稱在位址前。請參閱 範例 30.7「/etc/networks」[537頁]。

範例 30.7　/etc/networks

```
loopback      127.0.0.0
localnet      192.168.0.0
```

/etc/host.conf

名稱解析，即透過*解析程式庫* (Resolver Library) 翻譯主機及網路名稱，是由此檔案控制的。該檔案僅用於與 libc4 或 libc5 連結的程式。對於目前的 glibc 程式，請參閱 `/etc/nsswitch.conf` 中的設定。每個參數必須永遠是獨立一行。註解的前面是 # 符號。表格 30.6「/etc/host.conf 的參數」[537頁] 顯示出可用的參數。
`/etc/host.conf` 範例是顯示在 範例 30.8「 /etc/host.conf 」[538頁]。

表格 30.6　/etc/host.conf 的參數

order *hosts*, *bind*	指定名稱解析時服務的存取順序。可用的引數有 (以空格或逗號分隔)： *hosts*：搜尋 `/etc/hosts` 檔案 *bind*：存取名稱伺服器 *nis*：使用 NIS
multi *on*/*off*	定義在 `/etc/hosts` 中所輸入的主機是否可以有多個 IP 位址。
nospoof *on* spoofalert *on*/*off*	這些參數會影響名稱伺服器 *spoofing*，但除此之外，並不會對網路組態有任何影響。
trim *domainname*	指定的網域名稱在主機名稱解析後會與主機名稱分隔(只要主機名稱包括網域名稱)。只有在本地網域分離出來的名稱位於 `/etc/hosts` 檔案，但是仍然使用附加的網域名稱進行辨識時，這個選項才有用。

範例 30.8 */etc/host.conf*

```
# We have named running
order hosts bind
# Allow multiple addrs
multi on
```

/etc/nsswitch.conf

GNU C Library 2.0 的介紹伴隨*名稱服務切換* (NSS，Name Service Switch) 的介紹。詳細資訊請參閱 `nsswitch.conf(5)` 一文和*GNU C Library 參考手冊*。

查詢的順序定義於檔案 `/etc/nsswitch.conf`。`nsswitch.conf` 範例是顯示在範例 30.9「`/etc/nsswitch.conf`」[538頁]。註解從 # 符號開始。在此範例中，`hosts` 資料庫下的項目表示要求是透過 DNS 傳送到 `/etc/hosts` (`files`) (請參閱第 33 章「*網域名稱系統*」[559頁])。

範例 30.9 */etc/nsswitch.conf*

```
passwd:     compat
group:      compat

hosts:      files dns
networks:   files dns

services:   db files
protocols:  db files

netgroup:   files
automount:  files nis
```

NSS 上可用的「資料庫」列示於表格 30.7「透過 /etc/nsswitch.conf 的可用資料庫」[539頁]。此外，將來應該還有 `automount`、`bootparams`、`netmasks`、和 `publickey`。 NSS 資料庫的組態選項將列於表格 30.8「NSS「資料庫」的組態選項」[539頁]。

表格 30.7　透過 /etc/nsswitch.conf 的可用資料庫

aliases	sendmail 所執行的郵件別名；請參閱 man 5 aliases。
ethers	乙太網路位址。
group	getgrent 所使用的使用者群組。請參閱 group 的 man 頁面。
hosts	gethostbyname 及類似功能所使用的主機名稱與 IP 位址。
netgroup	在網路中有效的主機與使用者清單，以利控制存取權限，請參閱 netgroup(5) 一文。
networks	getnetent 所使用的網路名稱與位址。
passwd	getpwent 所使用的使用者密碼；請參閱 passwd(5) 一文。
protocols	getprotoen 所使用的網路通訊協定；請參閱 protocols(5) 一文。
rpc	getrpcbyname 及類似功能所使用的遠端程序呼叫名稱與位址。
services	getservent 使用的網路服務。
shadow	getspnam 所使用的使用者遮蔽密碼；請參閱 shadow(5) 一文。

表格 30.8　NSS「資料庫」的組態選項

files	直接存取檔案，例如 /etc/aliases
db	透過資料庫存取

nis、nisplus	NIS，請參閱第 35 章「*使用 NIS*」[597頁]
dns	只能做為 hosts 與 networks 的延伸
compat	只能做為 passwd、shadow 以及 group 的延伸

/etc/nscd.conf

此檔案用來設定 nscd (名稱服務快取精靈)。請參閱 nscd(8) 與 nscd.conf(5)。依預設，passwd 與 groups 的系統項目是由 nscd 快取。這對於目錄服務 (如 NIS 和 LDAP) 的效能而言是很重要的，否則每次存取名稱或群組時都需要使用網路連線。預設是不會快取 hosts，因為 nscd 快取主機的機制會造成本地系統無法信任轉寄以及反向查詢檢查。不要要求 nscd 快取名稱，而是設定快取 DNS 伺服器。

如果啟用 passwd 的快取，通常需要 15 秒，才能辨識新增的本地使用者。使用指令 rcnscd restart 重新啟動 nscd，縮短這段等待時間。

/etc/HOSTNAME

這是沒有附加網域名稱的主機名稱。機器開機時數個程序檔會讀取該檔案。它可以只包含一行，其中設定了主機名稱。

30.7.2 測試與組態

將組態寫入您的組態檔案之前，可先進行測試。若要設定測試組態，請使用 ip 指令。若要測試連接，請使用 ping 指令。同時也可使用較舊的組態工具，如 ifconfig 和 route。

ip、ifconfig 和 route 等指令會以不儲存組態檔案的方式直接變更網路組態。除非您將組態輸入正確的組態檔案，否則重新開機之後網路組態的變更就會遺失。

以 ip 設定網路介面

ip這個工具可顯示設定路由、網路設備、原則路由和通道。它是用來取代ifconfig和 route 等舊工具用的。

ip 是非常複雜的工具。其一般語法是 ip *options object command*。您可使用下列物件：

link
　此物件代表網路設備。

address
　此物件代表設備的 IP 位址。

neighbour
　此物件代表 ARP 或 NDISC 快取項目。

route
　此物件代表路由表格項目。

rule
　此物件代表路由原則資料庫中的規則。

maddress
　此物件代表多重廣播位址。

mroute
　此物件代表多重廣播路由快取項目。

tunnel
　此物件表示 IP 上的通道。

若未提供指令，會使用預設指令，通常是 list。

您可使用 ip link set *device_name command* 指令變更設備狀態。例如，若要停用設備 eth0，請輸入 ip link set eth0 down。若要重新啟用，請使用 ip link set eth0 up。

啟用設備之後，就可加以設定。若要設定 IP 位址，請使用 ip addr add *ip_address* + dev *device_name*。例如，若要將介面 eth0 的 IP 位址以標

準廣播 (選項 brd) 設定為 192.168.12.154/30，請輸入 `ip addr add 192.168.12.154/30 brd + dev eth0`。

若要具備作用中連接，必須設定預設閘道。若要為您的系統設定閘道，請輸入 `ip route get gateway_ip_address`。若要轉換某個 IP 位址，請使用 `nat`: `ip route add nat ip_address via other_ip_address`。

若要顯示所有設備，請使用 `ip link ls`。若只希望顯示運作中介面，請使用 `ip link ls up`。若要列印設備的介面統計值，請輸入 `ip -s link ls device_name`。若要檢是您設備的位址，請輸入 `ip addr`。在 `ip addr` 的輸出中同時也可找到您設備的 MAC 位址相關資訊。若要顯示所有路由，請使用 `ip route show`。

如需使用 ip 的詳細資訊，請輸入 `ip help` 或參閱 `ip(8)` 線上文件。`help` 選項也適用於所有 ip 物件。例如，假設您要閱讀 `ip addr` 的說明，請輸入 `ip addr help`。請在 `/usr/share/doc/packages/iproute2/ip-cref.pdf` 中尋找 ip 的手冊。

以 ping 測試連接

`ping` 指令是測試 TCP/IP 連接運作的標準工具。其使用 ICMP 通訊協定，將小型資料封包 ECHO_REQUEST 傳送至目的地主機，要求立即回應。如果有作用，`ping` 會顯示訊息，指示網路連結基本上是正常的。

`ping` 所做的不僅止於測試兩台電腦之間的連接狀態，它還可以提供某些有關於連接品質的基本資訊。您可在 範例 30.10「指令 ping 的輸出」[543頁] 中看到 `ping` 輸出的一些範例。倒數第二行包含已傳送封包數、遺失封包數、執行 `ping` 總時間等資訊。

對於目的地，可使用主機名稱或 IP 位址，例如 `ping example.com` 或 `ping 130.57.5.75`。程式會持續傳送封包，直到您按下 Ctrl + C 為止。

若您只需要檢查連接功能性，您可以 `-c` 選項限定封包數目。例如，若要將 ping 限制在三個封包，請輸入 `ping -c 3 192.168.0`。

範例 30.10　指令 ping 的輸出

```
ping -c 3 example.com
PING example.com (130.57.5.75) 56(84) bytes of data.
64 bytes from example.com (130.57.5.75): icmp_seq=1 ttl=49 time=188 ms
64 bytes from example.com (130.57.5.75): icmp_seq=2 ttl=49 time=184 ms
64 bytes from example.com (130.57.5.75): icmp_seq=3 ttl=49 time=183 ms
--- example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 183.417/185.447/188.259/2.052 ms
```

兩個封包之間的預設間隔為一秒。若要變更間隔，ping 提供了選項 `-i`。例如，若要將 ping 間隔增加到十秒，請輸入 `ping -i 10 192.168.0`。

在具備多網路設備的系統中，透過特定介面位址傳送 ping 非常實用。若要執行此功能，請使用所選設備名稱加上 `-i` 選項，例如 `ping -I wlan1 192.168.0`。

如需使用 ping 的選項與詳細資訊，請輸入 `ping -h` 或參閱 `ping (8)` 線上文件。

以 ifconfig 設定網路

`ifconfig` 是傳統網路設定工具。與 ip 相反，此指令只能用於介面組態。若您希望設定路由，請使用 `route`。

注：ifconfig 和 ip

ifconfig 這個程式已過時。請改用 ip。

無疑的，ifconfig 會顯示目前作用中介面的狀態。如同您在 範例 30.11「ifconfig 指令的輸出」[544頁] 中所見，ifconfig 具有排列整齊而詳盡的輸出。輸出第一行亦包含您設備的 MAC 位址、`HWaddr` 數值等資訊。

範例 30.11　ifconfig 指令的輸出

```
eth0      Link encap:Ethernet  HWaddr 00:08:74:98:ED:51
          inet6 addr: fe80::208:74ff:fe98:ed51/64 Scope:Link
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:634735 errors:0 dropped:0 overruns:4 frame:0
          TX packets:154779 errors:0 dropped:0 overruns:0 carrier:1
          collisions:0 txqueuelen:1000
          RX bytes:162531992 (155.0 Mb)  TX bytes:49575995 (47.2 Mb)
          Interrupt:11 Base address:0xec80

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8559 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8559 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:533234 (520.7 Kb)  TX bytes:533234 (520.7 Kb)

wlan1     Link encap:Ethernet  HWaddr 00:0E:2E:52:3B:1D
          inet addr:192.168.2.4  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: fe80::20e:2eff:fe52:3b1d/64 Scope:Link
          UP BROADCAST NOTRAILERS RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:50828 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43770 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:45978185 (43.8 Mb)  TX bytes:7526693 (7.1 Mb)
```

如需使用 ifconfig 的選項與詳細資訊，請輸入 `ifconfig -h` 或參閱 `ifconfig (8)` 線上文件。

以 route 設定路由

`route` 是操作 IP 路由表格的程式。您可使用此指令檢是您的路由組態，並新增或移除路由。

注：route 與 ip

route 這個程式已過時。請改用 ip。

若您需要快速又易於理解的路由組態資訊以判別路由問題，route 是特別實用的工具。若要檢視您目前的路由組態，請以 `root` 身份輸入 `route -n`。

範例 30.12　route -n 指令的輸出

```
route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.20.0.0       *               255.255.248.0   U         0 0          0 eth0
link-local      *               255.255.0.0     U         0 0          0 eth0
loopback        *               255.0.0.0       U         0 0          0 lo
default         styx.exam.com   0.0.0.0         UG        0 0          0 eth0
```

如需使用 route 的選項與詳細資訊，請輸入 `route -h` 或參閱 `route (8)` 線上文件。

30.7.3 啟動程序檔

除了上述的組態檔之外，還有在機器開機時載入網路程式的各種程序檔。只要系統切換到其中一個 *multiuser runlevels*，就會啟動這些程序檔。在 表格 30.9 「網路程式的一些啟動程序檔」 [545頁] 中提供了一些程序檔的說明。

表格 30.9　網路程式的一些啟動程序檔

`/etc/init.d/network`	這個程序檔可處理網路介面的組態。硬體必須先由 `/etc/init.d/coldplug` (透過 `hotplug`) 啟始化。如果沒有啟動 `network` 服務，透過熱插拔插入網路介面時，將無法執行它們。
`/etc/init.d/inetd`	啟動 xinetd。xinetd 可以用來讓伺服器服務能夠在系統上使用。例如，只要開啟FTP 連接，它即可啟動 vsftpd。
`/etc/init.d/portmap`	啟動 RPC 伺服器 (如 NFS 伺服器) 所需的埠對應程式 (Portmapper)。
`/etc/init.d/nfsserver`	啟動 NFS 伺服器。
`/etc/init.d/postfix`	控制後置程序。
`/etc/init.d/ypserv`	啟動 NIS 伺服器。

`/etc/init.d/ypbind`	啟動 NIS 用戶端。

30.8 smpppd 做為撥號助理

部份家庭使用者沒有連接到網際網路的專線。而是使用撥號連接。視撥號方法 (ISDN 或 DSL) 而定，連接是由 ipppd 或 pppd 來控制。基本上，連接所需做的事就是正確地啟動這些程式。

如果您有單一速率的連接，不會產生撥號連接的其他成本，請直接啟動個別的精靈。使用 KDE Applet 或指令行介面以控制撥號連接。如果網際網路閘道不是您所使用的主機，您可能需要透過網路主機來控制撥號連接。

這裏將會需要 smpppd。它會為輔助程式提供一致的介面並以兩個方向運作。首先，它會撰寫所需的 pppd 或 ipppd 程式，然後控制其撥號內容。其次，它會提供各個提供者給使用者程式，並傳輸關於連接目前狀態的資訊。因為 smpppd 也可以透過網路控制，所以它適合從私人子網路的工作站中，控制連至網際網路的撥號連接。

30.8.1 設定 smpppd

smpppd 所提供的連接會自動由 YaST 設定。實際的撥號程式 KInternet 與 cinternet 也是預先設定的。手動設定只需設定 smpppd 的其他功能，例如遠端控制。

smpppd 的組態檔為 `/etc/smpppd.conf`。根據預設，它不會啟用遠端控制。此組態檔最重要的選項為：

open-inet-socket = *`yes`*|*`no`*
: 若要透過網路控制 smpppd，此選項必須設定為 `yes`。smpppd 所傾聽的埠為 `3185`。如果此參數設為 `yes`，則也應該設定 `bind-address`、`host-range` 以及 `password` 參數。

bind-address = *`ip address`*
: 如果主機具有數個 IP 位址，請使用此參數以決定那個 IP 位址 smpppd 應該接受連接。預設為監聽所有位址。

host-range = *min ip max ip*

`host-range` 參數會定義網路範圍。在此範圍中的 IP 位址之主機擁有 smpppd 的存取權。所有不在此範圍中的主機都將拒絕存取。

password = *password*

透過指定密碼，將用戶端限制為授權的主機。因為這是純文字的密碼，您不應該高估它所提供的安全性。如果沒有指定密碼，則所有的用戶端都將允許存取 smpppd。

slp-register = *yes|no*

使用此參數，smpppd 服務就可以透過 SLP 在網路中宣告。

有關 smpppd 的詳細資訊，請參閱 `smpppd(8)` 和 `smpppd.conf(5)`。

30.8.2 設定 KInternet、cinternet 與 qinternet 以供遠端使用

KInternet、cinternet 和 qinternet 可以用來控制本地或遠端的 smpppd。cinternet 指令行相當於圖形模式的 KInternet。qinternet 基本上和 KInternet 一樣，但並不使用 KDE 文件庫，因此它不需 KDE 即可供您使用，而您必須個別進行安裝。若要準備這些公用程式以供遠端 smpppd 使用，請手動編輯 `/etc/smpppd-c.conf` 組態檔或使用 KInternet。這個檔案只使用三個選項：

sites = *list of sites*

在此，可看出要搜尋 smpppd 的前端。這些前端將會依這裏所指定的順序來測試選項。`local` 選項可命令建立與本地 smpppd 的連線。`gateway` 指向閘道上的 smpppd。這個連接將會依 `config-file` 中 `server` 下的指定來建立。`slp` 會命令前端，連接至透過 SLP 所找到的 smpppd。

server = *server*

此處指定 smpppd 所執行的主機。

password = *password*

插入為 smpppd 所選取的密碼。

如果 smpppd 為作用中，您現在可以嘗試存取它，例如，使用 `cinternet --verbose --interface-list`。如果現在碰到任何困難，請參閱 `smpppd-c.conf(5)` 與 `cinternet(8)`。

31 網路中的 SLP 服務

服務位置通訊協定 (SLP) 是開發用來簡化區域網路內的網路用戶端組態。若要設定網路用戶端 (包含所有必要的服務)，一般而言管理員需要對於網路中可用的伺服器有詳細的瞭解。SLP 可讓區域網路中的所有用戶端，都知道選定服務的可用性。支援 SLP 的應用程式可以使用散佈的資訊並可自動設定。

SUSE Linux Enterprise® 支援使用 SLP 所提供的安裝來源進行安裝，並包含許多有 SLP 整合支援的系統服務。YaST 和 Konqueror 都包含 SLP 適當的前端介面。您可以使用 SLP 以提供主要的功能給網路上的用戶端，例如系統上的安裝伺服器、檔案伺服器或是列印伺服器。

重要：SUSE Linux Enterprise 中的 SLP 支援

提供 SLP 支援的服務包含 cupsd、rsyncd、ypserv、openldap2、openwbem (CIM)、ksysguardd、saned、kdm vnc login、smpppd、rpasswd、postfix 以及 sshd (透過 fish)。

31.1 啟用 SLP

您系統必須執行 slpd，才能提供 SLP 服務。如果只是要做服務查詢，並不需要啟動此精靈。就像大部份在 SUSE Linux Enterprise 中的系統服務一樣，slpd 精靈是利用獨立的安裝程序檔來控制。預設精靈為非作用中。若要啟用精靈，讓在工作階段期間持續執行，以 `root` 的身份執行 `rcslpd start` 可啟動精靈，而執行 `rcslpd stop` 則可停止精靈。請以 `restart` 或 `status` 執行重新啟動或狀態檢查。若 slpd 預設應為啟動，請在 YaST「*系統*」>「*系統服務*

(Runlevel)」中啟用 slpd，或以 `root` 身份執行 `insserv slpd` 指令。這將會在系統開機時，自動在要啟動的服務集合中包括 slpd。

31.2 在 SUSE Linux Enterprise 中的 SLP 前端

若要尋找 SLP 在您網路上提供的服務，請使用 SLP 前端。SUSE Linux Enterprise 包含多種前端：

slptool
: slptool 是簡易的指令行程式，可以在網路中宣告 SLP 查詢或宣告專用的服務。`slptool --help` 可以列出所有可用的選項與功能。您也可以從處理 SLP 資訊的程序檔呼叫 slptool。

YaST SLP 瀏覽器
: YaST 包含獨立的 SLP 瀏覽器，以樹狀圖表在 網路服務 SLP 瀏覽器下列出區域網路中所有透過 SLP 所宣告的服務。以「*網路服務*」>「*SLP 瀏覽器*」尋找。

Konqueror
: 當 Konqueror 做為網路瀏覽器時，可以在 `slp:/` 顯示區域網路中所有可用的 SLP 服務。在主要視窗中按一下圖示，以取得更多關於相關服務的詳細資訊。如果您使用 Konqueror 加上 `service:/`，則請在瀏覽器視窗中按一次相關圖示，以設定與選取服務的連接。

31.3 透過 SLP 安裝

如果您在網路中提供具有 SUSE Linux Enterprise 安裝媒體的安裝伺服器，就可以使用 SLP 註冊。如需詳細資料，請參閱 第 4.2.1 節「使用 YaST 設定安裝伺服器」[51頁]. 如果已選取 SLP 安裝，linuxrc 會在系統從選取的開機媒體開機後啟動 SLP 查詢，並顯示所找到的來源。

31.4 以 SLP 提供服務

許多在 SUSE Linux Enterprise 中的應用程式，透過 `libslp` 程式庫的使用，已經具有整合的 SLP 支援。如果尚未使用 SLP 支援來編譯服務，請使用下列其中一種方式讓 SLP 編譯服務：

使用 `/etc/slp.reg.d` 的靜態註冊
: 針對每個新的服務建立個別的註冊檔。下列是註冊掃描器服務的檔案範例：

```
## Register a saned service on this system
## en means english language
## 65535 disables the timeout, so the service registration does
## not need refreshes
service:scanner.sane://$HOSTNAME:6566,en,65535
watch-port-tcp=6566
description=SANE scanner daemon
```

在此檔案中最重要的一行為 *service URL*，它是以 `service:` 開始。這包含服務類型 (`scanner.sane`) 以及位址 (可在其中找到伺服器可用的服務)。*`$HOSTNAME`* 會以完整的主機名稱自動取代。接著可以找到相關服務的 TCP 埠名稱，它們之間是以冒號分隔。然後輸入服務應該出現的語言以及註冊期間的秒數。這些都應該使用逗號與服務 URL 分隔。在 `0` 與 `65535` 之間設定註冊期間的值。`0` 會防止註冊。`65535` 會移除所有的限制。

註冊檔另外也包含 `watch-tcp-port` 和 `description` 兩個變數。`watch-tcp-port` 利用 slpd 檢查服務的狀態，來將 SLP 服務宣告連結至是否為作用中的相關服務。第二個變數是對顯示在適當瀏覽器中的服務，提供更為精確的描述。

提示：YaST 以及 SLP

當您在模組對話方塊中啟用 SLP 時，一些由 YaST 所仲介的服務 (例如安裝伺服器或 YOU 伺服器) 會自動執行此註冊。YaST 接著就會為這些服務建立註冊檔。

使用 `/etc/slp.reg` 的靜態註冊
: 程序與 `/etc/slp.reg.d` 唯一的差異是在中央檔案內所有服務的群組方式。

以 slptool 動態註冊
: 如果應該從專用的程序檔為 SLP 註冊服務，請使用 slptool 指令行前端。

31.5 如需更多資訊

下列來源提供關於 SLP 的進一步資訊：

RFC 2608、2609、2610
: RFC 2608 一般會處理 SLP 的定義。RFC 2609 會處理更加詳細的服務 URL 語法，而 RFC 2610 則會透過 SLP 處理 DHCP。

`http://www.openslp.org/`
: OpenSLP 計劃的首頁。

`/usr/share/doc/packages/openslp`
: 這個目錄包含所有關於 SLP 的可用文件，包括 `README.SuSE` (含有 SUSE Linux Enterprise 詳細資訊)、RFC 以及兩個介紹性的 HTML 文件。想要使用 SLP 功能的程式設計人員，應該安裝 `openslp-devel` 套件，以參閱它所提供的*《程式設計人員指南》*。

32 使用 NTP 進行時間同步化

NTP (網路時間協定) 機制是一種協定，用於同步化網路上的系統時間。首先，機器可以從提供可靠時間來源的伺服器取得時間。其次，機器本身在網路中可以做為其他電腦的時間來源。這個目標是雙重的，即維護絕對正確的時間，並同步化網路內所有機器的系統時間。

維護精準的系統時間對於許多情況都非重要。內建的硬體 (BIOS) 時鐘通常無法符合像是資料庫等應用程式的需求。手動校正系統時間有可能會造成嚴重的問題，因為，例如時間倒退將可能造成重要應用程式無法正常運作。在網路中，通常需要同步所有機器中的系統時間，而手動調整時間的做法並不可取。xntp 提供了一種用於解決這種問題的機制。它會透過網路中可靠的時間伺服器來持續調整系統時間。它可以進一步管理本地參考的時鐘，例如收音機控制的時鐘。

32.1 使用 YaST 設定 NTP 用戶端

xntp 會預先設定為使用本地電腦時鐘做為時間參考。不過，使用 (BIOS) 時鐘，僅供萬一沒有更精確的時間來源時備用。YaST 使用了 NTP 用戶端的組態。對於為執行的系統，請使用快速組態或進階組態。對於受防火牆保護的系統，進階組態可以開啟 SuSEfirewall2 中的必要連接埠。

32.1.1 快速 NTP 用戶端組態

快速的 NTP 用戶端組態 (「*網路服務*」>「*NTP 用戶端*」) 是由兩個對話方塊所組成。請在第一個對話方塊中設定 xntpd 的開始模式以及要查詢的伺服器。當系統開機時，若要自動啟動 xntpd，請按一下「*開機時*」。然後指定「*NTP 伺服器*

組態」。按一下「*使用 pool.ntp.org 的隨機伺服器*」(如果無法使用本地時間伺服器)，或按一下「*選取*」存取第二個對話方塊，在其中選取您的網路適用的時間伺服器。

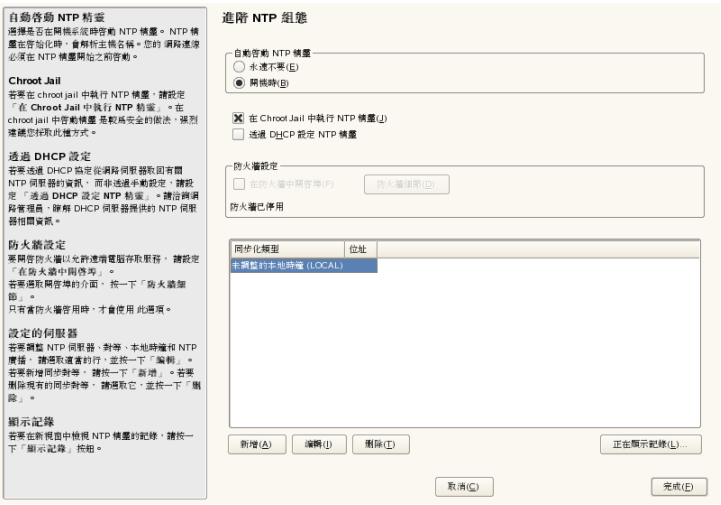
圖形 32.1 *YaST：設定 NTP 用戶端*

在詳細的伺服器選取對話方塊中，決定執行時間同步化時，要使用區域網路中的時間伺服器 (「*本地 NTP 伺服器*」)，還是使用可以處理時區的網際網路時間伺服器 (「*公用 NTP 伺服器*」)。若需本地時間伺服器，請按一下「*查詢*」，開始 SLP 查詢，在網路中尋找可用的時間伺服器。從搜尋結果清單中選取最合適的時間伺服器，並按一下「*確定*」，結束對話方塊。若需公用時間伺服器，請選取您的國家 (時區) 並從「*公用 NTP 伺服器*」下的清單選取適當的伺服器，然後按一下「*確定*」，結束對話方塊。在主對話方塊中，使用「*測試*」來測試所選取伺服器的可用性，並按一下「*完成*」，結束對話方塊。

32.1.2 進階的 NTP 用戶端組態

依照快速組態所述選取了啟動模式後，即可從「*NTP 用戶端*」模組主對話方塊中的「*進階組態*」下，存取 NTP 用戶端的進階組態，如圖形 32.1「YaST：設定 NTP 用戶端」[554頁]所顯示。

圖形 32.2　YaST：複雜的 NTP 組態



在「*進階 NTP 組態*」中，決定是否要在 Chroot Jail 中啟動 xntpd。依照預設，會啟用「*在 Chroot Jail 中執行 NTP 精靈*」。因為它可以防止攻擊者損毀整個系統，因此在 xntpd 遭受攻擊時，會有較高的安全性。「*透過 DHCP 設定 NTP 精靈*」可以設定 NTP 用戶端，而能透過 DHCP 取得網路中可用的 NTP 伺服器清單。

如果 SuSEfirewall 在作用中 (這是預設值)，請啟用「*開啟防火牆的連接埠*」。如果您讓連接埠保持為關閉，就不可能對時間伺服器建立連接。

用戶端要查詢的伺服器以及其他時間來源會列在清單下半部。修改清單時，可依需要使用「*新增*」、「*編輯*」以及「*刪除*」。「*顯示記錄*」可用來檢視用戶端的記錄檔。

按一下「*新增*」以新增時間資訊的新來源。在下列對話方塊中，選取進行時間同步化的來源類型。可用的選項如下：

伺服器
: 另一個對話方塊可讓您選取 NTP 伺服器 (如 第 32.1.1 節「快速 NTP 用戶端組態」[553頁]中所述)。在系統開機時，請啟用「*用以啟始同步化*」來觸發伺服器與用戶端之間的時間資訊同步化。「*選項*」可讓您指定 xntpd 的其他選項。如需詳細資訊，請參閱 `/usr/share/doc/packages/xntp-doc` (`xntp-doc` 套件的一部份)。

點
: 點 (peer)，是指一台建立了對稱關係的機器：它可同時做為時間伺服器與用戶端。若要在相同的網路中使用點而非伺服器，請輸入系統的位址。其餘的對話方塊與「*伺服器*」對話方塊相同。

收音機時鐘
: 若要在系統中使用收音機時鐘來進行時間同步化，請在此對話方塊中輸入時鐘類型、單位編號、設備名稱以及其他選項。按一下「*驅動程式校正*」，即可微調驅動程式。`/usr/share/doc/packages/xntp-doc/refclock.html` 中提供了關於本地無線電時鐘作業的詳細資訊。

外寄廣播
: 時間資訊與查詢也可以透過網路中的廣播傳輸。請在此對話方塊中輸入廣播所應傳送至的位址。除非您已經有類似收音機控制時鐘的可靠時間來源，否則請勿啟用廣播。

內送廣播
: 如果您想要讓用戶端透過廣播接收其資訊，請在這些欄位中輸入應該接受的個別封包位址。

32.2 設定網路中的 xntp

在網路上使用時間伺服器的最簡單方式就是設定伺服器參數。例如，如果可以從網路存取名為 `ntp.example.com` 的時間伺服器，那麼，請新增以下行，將此伺服器的名稱新增到 `/etc/ntp.conf` 檔案：

```
server ntp.example.com
```

若要新增更多時間伺服器，請以關鍵字 `伺服器` 插入其他行。在以 `rcntpd start` 指令啟始 xntpd 後，將需要約一個小時才能使時間穩定下來，而且會建立累積記錄檔案以校正本地電腦時鐘。使用累積記錄檔案，就可以在電腦一開機後立即計算硬體時鐘的系統錯誤。它會立即使用校正，使系統時間具有更高的穩定性。

有兩種方法可以將 NTP 機制做為用戶端：首先，用戶端可固定在每段間隔時間後向已知伺服器查詢時間。隨著用戶端的增加，此方法可能造成伺服器的高負載。其次，用戶端可以等待網路中的廣播時間伺服器所送出的 NTP 廣播。此方法具有伺服器品質未知的缺點，而且伺服器送出錯誤的資訊可能造成嚴重的問題。

如果時間是經由廣播取得，就不需要伺服器名稱。在這樣的情形下，請在 /etc/ntp.conf 組態檔中輸入 broadcastclient。若要完全使用一或多個已知的時間伺服器，請輸入以 servers 開頭的名稱。

32.3 設定本地參考時鐘

軟體套件 xntp 包含與本地參考時鐘連接的驅動程式。檔案 /usr/share/doc/packages/xntp-doc/refclock.html 的 xntp-doc 套件中提供了支援的時鐘清單。每個驅動程式都與數字關聯。在 xntp 中，實際組態工作是利用虛擬 IP 位址來執行。把時鐘當成在網路中一樣，將它輸入 /etc/ntp.conf 檔案中。因此，會指定特殊的 IP 位址，127.127.*t*.*u* 給它們。在此，*t* 代表時鐘的類型並可決定使用哪一個驅動程式，而 *u* 是代表單位，可決定使用哪一個介面。

一般而言，個別設備都具有描述組態細節的特殊參數。檔案 /usr/share/doc/packages/xntp-doc/drivers/driver*NN*.html (其中的 *NN* 為驅動程式的編號) 提供了關於特定時鐘類型的資訊。例如，「type 8」時鐘 (透過序列介面的收音機時鐘) 需要其他可以更精確地指定時鐘的模式。例如，Conrad DCF77 接收器模組具有模式 5。若要使用此時鐘做為偏好的參考，請指定 prefer 關鍵字。Conrad DCF77 接收器模組的完整 server 行如下所示：

```
server 127.127.8.0 mode 5 prefer
```

其他的時鐘也使用相同的模式。安裝 xntp-doc 套件後，便可在 /usr/share/doc/packages/xntp-doc/ 目錄中找到 xntp 的文件。檔案 /usr/share/doc/packages/xntp-doc/refclock.html 提供了描述驅動程式參數的驅動程式頁面連結。

33 網域名稱系統

必須使用 DNS (網域名稱系統) 將網域名稱和主機名稱解析為 IP 位址。例如，藉由這種方式，IP 位址 192.168.0.1 會指派給主機名稱 `earth`。在設定您自己的名稱伺服器前，請參閱第 30.3 節「名稱解析」[511頁]中有關 DNS 的一般資訊。以下設定範例是指 BIND。

33.1 DNS 詞彙

區域
: 網域名稱空間細分成一個個區域。例如，如果您擁有 `example.org`，您就擁有 `org` 網域的 `example` 區段或區域。

DNS 伺服器
: DNS 伺服器是為網域維護名稱和 IP 資訊的伺服器。您可以擁有用於主要區域的主要 DNS 伺服器、用於從屬區域的次要伺服器、或沒有任何區域處理快取功能的從屬伺服器。

  主要區域 DNS 伺服器
  : 主要區域包含您網路上的所有主機，而且 DNS 伺服器主要區域會儲存您的網域中所有主機最新的記錄。

  從屬區域 DNS 伺服器
  : 從屬區域是主要區域的副本。從屬區域 DNS 伺服器會用區域傳輸作業從其主伺服器取得區域資料。只要從屬區域 DNS 伺服器擁有有效 (未過期) 的區域資料，它就有權代表區域回應。如果從屬無法取得區域資料的新副本，它就會停止代表區域回應。

轉遞者
: Forwarder 是當您的 DNS 伺服器無法答覆查詢時，它應該轉送查詢的目標 DNS 伺服器。

記錄
: 記錄是有關名稱和 IP 位址的資訊。有關支援的記錄及其語法，請參閱 BIND 文件中的說明。一些特殊的記錄如下：

  NS 記錄
  : NS 記錄會告訴名稱伺服器哪些機器負責管理特定網域區域。

  MX 記錄
  : MX (郵件交換) 記錄會說明在網際網路上傳送郵件時要聯絡的機器。

  SOA 記錄
  : SOA (授權啟動) 記錄是區域檔案中的第一筆記錄。SOA 記錄是在使用 DNS 同步化多部電腦之間的資料時使用。

33.2 使用 YaST 進行設定

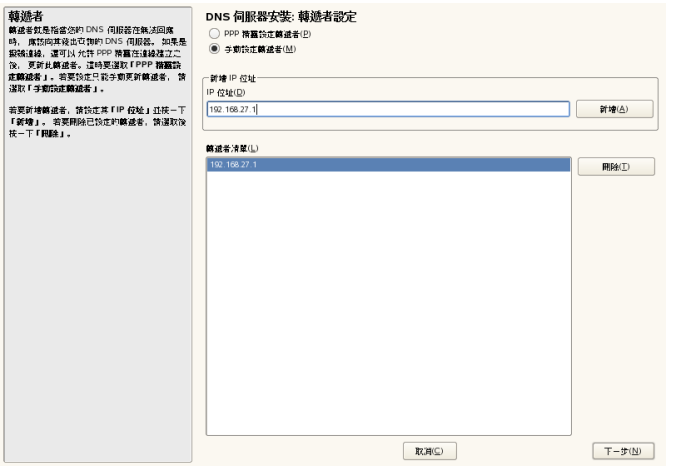
您可以使用 YaST 的 DNS 模組設定區域網路的 DNS 伺服器。若要設定 Samba 伺服器，請啟動 YaST 並選取「*網路服務*」>「*DNS 伺服器*」。第一次啟動模組時，會啟動精靈提示您決定有關伺服器管理的一些基本設定。完成此初始設定程序會產生一個非常基本的伺服器組態，適用於基本的狀況。進階模式可以用來處理更進階的組態工作，如設定 ACL、記錄、TSIG 金鑰和其他選項。

33.2.1 精靈組態

精靈包含三個步驟或對話方塊。在對話方塊中適當的地方，能夠讓您進入進階組態模式。

1 第一次啟動模組時，會開啟圖形 33.1 「DNS 伺服器安裝：轉遞者設定」[561頁]中所示的「*轉遞者設定*」對話方塊。它可以決定 PPP 精靈是否應該提供透過 DSL 或 ISDN 在撥接時 forwarder 的清單 (「*PPP 精靈設定 Forwarder*」)，或者是否要提供您自己的清單 (「*手動設定 Forwarder*」)。

圖形 33.1　DNS 伺服器安裝：轉遞者設定



2 「*DNS 區域*」對話方塊包含數個部分，負責管理區域檔案，如第 33.5 節「區域檔案」[574頁]所述。對於新區域，請在「*區域名稱*」中提供名稱。若要新增反向區域，名稱的結尾必須是 `.in-addr.arpa`。最後，選取「*區域類型*」(主要或從屬)。請參閱圖形 33.2「DNS 伺服器安裝：DNS 區域」[561頁]。 按一下「*編輯區域*」，設定現有區域的其他設定值。若要移除區域，按一下「*刪除區域*」。

圖形 33.2　DNS 伺服器安裝：DNS 區域

3 在最後的對話方塊中，可按一下「*在防火牆中開啟埠*」，在防火牆中開啟 DNS 連接埠。然後決定是否要啟動 DNS 伺服器 (「*開啟*」或「*關閉*」)。您亦可啟用LDAP 支援。請參閱圖形 33.3「DNS 伺服器安裝：完成精靈」[562頁]。

圖形 33.3　DNS 伺服器安裝：完成精靈

33.2.2 進階組態

啟動模組後，YaST 會開啟顯示數個組態選項的視窗。完成該視窗可讓 DNS 伺服器組態的基本功能就位運作：

啟動 DNS 伺服器。

在「*服務啟動*」下，定義是在系統開機(啟動系統時)時啟動DNS伺服器，還是手動啟動。若要立即啟動 DNS 伺服器，請選取「*立即啟動 DNS 伺服器*」。若要停止DNS伺服器，請選取「*立即停止 DNS 伺服器*」。若要儲存目前的設定，請選取「*立即儲存設定並重新啟動DNS 伺服器*」。您可以使用「*開啟防火牆中的連接埠*」開啟防火牆中的 DNS 埠，並使用「*防火牆詳細資訊*」修改防火牆設定。

若選取「*LDAP 主動支援*」，區域檔案將由 LDAP 資料庫來管理。只要 DNS 伺服器重新啟動或提示重新載入其組態時，就會抓取寫入到 LDAP 資料庫中區域資料的變更。

DNS 伺服器基本選項

在此區段中，可設定基本伺服器選項。從「*選項*」功能表中，選取所需項目，然後在對應的項目欄位中指定值。選取「*新增*」以包含新項目。

記錄

若要設定 DNS 伺服器應記錄的內容和記錄方式，請選取「*記錄*」。在「*記錄類型*」下，指定 DNS 伺服器應該寫入記錄資料的位置。選取「系統記錄」來使用全系統的記錄檔「*/var/log/messages*」或選取「*檔案*」指定不同的檔案。在選取後者的情況下，另外還要指定名稱、最大檔案大小 (以 MB 為單位) 以及要儲存的記錄檔的版本數量。

進一步選項可從「*其他記錄*」下存取。啟用「*記錄所有 DNS 查詢*」會記錄*每個*查詢，此選項會讓記錄檔變得非常大。所以，除了偵錯用途外，啟用此選項並不是理想的作法。若要記錄 DHCP 與 DNS 伺服器之間在區域更新期間的資料流量，請啟用「*記錄區域更新*」。若要記錄從主伺服器到從屬伺服器在區域傳輸期間的資料流量，請啟用「*記錄區域轉送*」。請參閱圖形 33.4「DNS 伺服器：記錄」 [564頁]。

圖形 33.4　DNS 伺服器：記錄

使用 ACL

使用此視窗可定義 ACL (存取控制清單) 以執行存取限制。在「*名稱*」下提供獨特名稱後，在「*值*」下指定 IP 位址 (有或沒有網路遮罩)，格式如下：

```
{ 10.10/16; }
```

組態檔的語法要求位址以分號結尾，而且放置在大括號之間。

TSIG 金鑰

TSIG (交易簽章) 的主要目的是保護 DHCP 與 DNS 伺服器之間的通訊。在第 33.7 節「安全交易」 [578頁]中有所描述。

若要產生 TSIG 金鑰，請在標籤為「*金鑰 ID*」的欄位中輸入特別的名稱，並指定用來儲存金鑰的檔案 (「*檔案名稱*」)。以「*新增*」 確認您的選項。

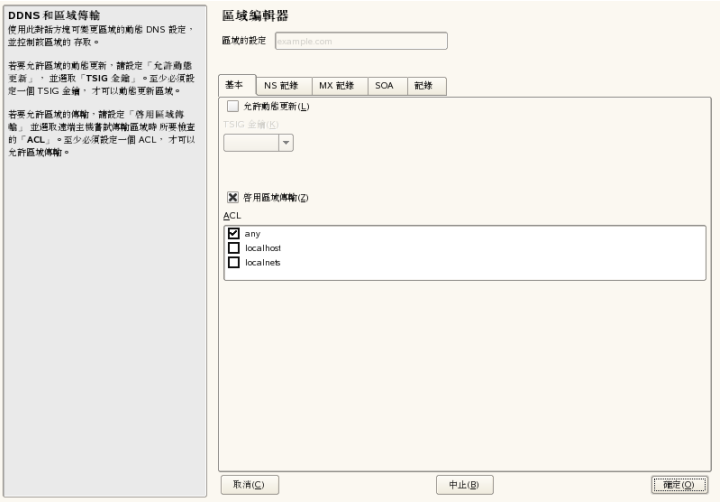
若要使用之前建立的金鑰，請將「*金鑰 ID*」欄位保留空白，並在「*檔案名稱*」下選取用來儲存的檔案。接著，以「*新增*」 確認您的選項。

新增從屬區域

若要新增從屬區域，請選取「*DNS 區域*」，選擇「*從屬*」區域類型，並按一下「*新增*」。

在「*主 DNS 伺服器 IP*」的「*區域編輯器*」下，指定從屬伺服器要從中取得其資料的主伺服器。若要限制對伺服器的存取，可以從清單選取其中一個 ACL。請參閱圖形 33.5「DNS 伺服器：從屬區域編輯器」[565頁]。

圖形 33.5 *DNS 伺服器：從屬區域編輯器*



新增主要區域

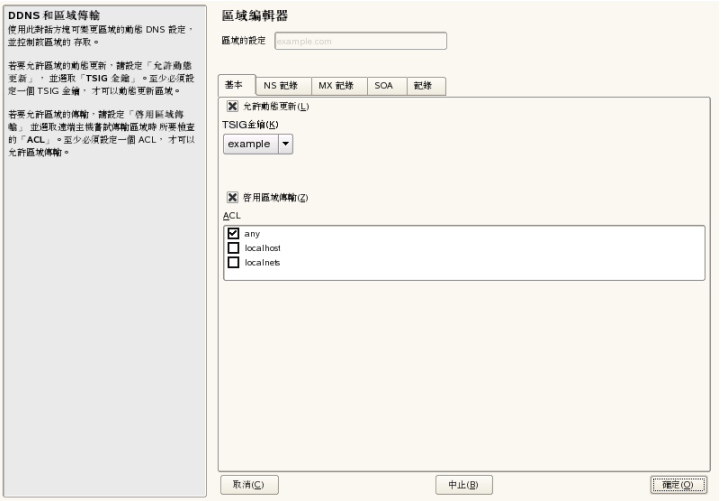
若要新增主要區域，請選取「*DNS 區域*」，選擇「*主要*」區域類型，指定新區域的名稱，並按一下「*新增*」。

編輯主要區域

若要編輯主要區域，請選取「*DNS 區域*」，選擇「*主要*」區域類型，然後從表格中選取主要區域，並按一下「*編輯*」。對話方塊由數個頁面組成：「*基本*」(第一個開啟的頁面)、「*NS 記錄*」、「*MX 記錄*」、「*SOA*」以及「*記錄*」。

基本對話方塊 (如圖形 33.6「DNS 伺服器：區域編輯器 (基本)」[566頁]所示)，可讓您定義動態 DNS 的設定以及到用戶端及從屬名稱伺服器之區域傳輸的存取選項。若要允許動態更新區域，請選取「*允許動態更新*」以及對應的 TSIG 金鑰。更新動作開始前，必須先定義金鑰。若要啟用區域傳輸，請選取對應的 ACL。必須先行定義 ACL。

圖形 33.6 *DNS 伺服器：區域編輯器 (基本)*



區域編輯器 (NS 記錄)

此對話方塊允許您為指定的區域定義替代名稱伺服器。請確定您自己的名稱伺服器包含於清單中。若要新增記錄，請在「*要新增的名稱伺服器*」下輸入其名稱，然後使用「*新增*」確認動作。請參閱圖形 33.7「DNS 伺服器：區域編輯器 (NS 記錄)」[567頁]。

圖形 33.7　DNS 伺服器：區域編輯器 (NS 記錄)

區域編輯器 (MX 記錄)

若要新增目前區域的郵件伺服器到現有清單，請輸入對應的位址及優先順序值。完成後，選取「*新增*」確認該動作。請參閱圖形 33.8「DNS 伺服器：區域編輯器 (MX 記錄)」[567頁]。

圖形 33.8　DNS 伺服器：區域編輯器 (MX 記錄)

區域編輯器 (SOA)

此頁允許您建立 SOA (授權啟動) 記錄。如需個別選項的說明，請參閱範例 33.6「檔案 /var/lib/named/world.zone」[574頁]。透過 LDAP 管理的動態區域，並不支援變更 SOA 記錄。

圖形 33.9　DNS 伺服器：區域編輯器 (SOA)

區域編輯器 (記錄)

此對話方塊可管理名稱解析。在「*記錄金鑰*」中，輸入主機名稱，然後選取其類型。「*A-Record*」代表主項目。此項目的值應為 IP 位址。「*CNAME*」是別名。使用「*NS*」與「*MX*」類型，可取得「*NS 記錄*」與「*MX 記錄*」標籤提供之資訊的詳細或部分擴充記錄。這三個類型都可以解析成現有的 A 記錄。「*PTR*」是供反向區域所使用。它是 A 記錄的相反。

33.3 啟動名稱伺服器 BIND

在 SUSE Linux Enterprise® 系統上，名稱伺服器 BIND (*柏克萊網際網路名稱網域*)已經預先設定，所以安裝後即可啟動，不會有任何問題。如果您已具有可以運作的網際網路連線，而且在 `/etc/resolv.conf` 中輸入了 `127.0.0.1` 做為 `localhost` 的名稱伺服器位址，則通常表示您已經具有可以運作的名稱解析，因而無需瞭解提供者的 DNS。BIND 透過根名稱伺服器執行名稱解析，顯見處理程序較慢。一般而言，應該在 `forwarders` 下的組態檔 `/etc/named.conf` 中輸入提供者的 DNS 及其 IP 位址，以確保有效及安全的名稱解析。如

果目前此辦法可行，名稱伺服器會當成純粹的「*僅快取*」名稱伺服器執行。只有在您為名稱伺服器設定它自己的區域時，它才會變成適合的 DNS。有關此種情況的簡單範例，請參閱 `/usr/share/doc/packages/bind/config` 中的文件。

提示：自動使用名稱伺服器資訊

名稱伺服器資訊可自動根據目前的情況調整，視網際網路連線或網路連線的類型而定。若要這樣做，將檔案 `/etc/sysconfig/network/config` 中的變數 `MODIFY_NAMED_CONF_DYNAMICALLY` 設定為 `yes`。

不過，在相關機構指派正式網域前，請勿進行任何設定。即使您有自己的網域而且是由提供者管理，最好也不要使用，否則 BIND 不會轉遞此網域的要求。例如，此網域將無法存取提供者的網頁伺服器。

若要啟動名稱伺服器，請以 `root` 的身份輸入指令 `rcnamed start`。如果右邊出現綠色的「完成」，即表示已成功啟動稱為 named 的名稱伺服器程序。使用 `host` 或 `dig` 程式立即測試本地系統上的名稱伺服器，應該會傳回 `localhost` 做為預設伺服器，位址為 `127.0.0.1`。如果出現的不是這種情況，則可能是 `/etc/resolv.conf` 包含不正確的名稱伺服器項目，或是檔案根本不存在。如果是第一次測試，請輸入 `host 127.0.0.1`，這通常都能成功。如果看到錯誤訊息，請使用 `rcnamed status`，檢查伺服器是否真的執行。如果名稱伺服器沒有啟動或是未以預期的方式運作，通常在 `/var/log/messages` 記錄檔中可以找到原因。

若要使用提供者的名稱伺服器或將網路上已經執行的名稱伺服器作為轉遞者，請在 `forwarders` 下的 `options` 區段中輸入對應的 IP 位址。範例 33.1 「named.conf 中的轉寄選項」 [569頁]中包含的位址只是範例。請根據您自己的設定調整這些項目。

範例 33.1 *named.conf 中的轉寄選項*

```
options {
        directory "/var/lib/named";
        forwarders { 10.11.12.13; 10.11.12.14; };
        listen-on { 127.0.0.1; 192.168.0.99; };
        allow-query { 127/8; 192.168.0/24; };
        notify no;
        };
```

options 項目後面跟著區域的項目：localhost 以及 0.0.127.in-addr.arpa。在「.」之下的 type hint 項目應該永遠是存在的。對應的檔案無需修改，而且應該依其原狀運作。另外也請確定每個項目的末尾都有「;」，且大括號在正確的位置。變更組態檔 /etc/named.conf 或區域檔後，使用 rcnamed reload 可要求 BIND 重新讀取這些檔案。使用 rcnamedrestart 停止並重新啟動名稱伺服器會達成相同的結果。任何時候，輸入 rcnamed stop 都可停止伺服器。

33.4 組態檔 /etc/named.conf

BIND 名稱伺服器本身的所有設定都儲存於檔案 /etc/named.conf。不過，要處理之網域的區域資料，包括主機名稱、IP 位址等，儲存於 /var/lib/named 目錄中的不同檔案中。詳細資訊會在稍後說明 。

/etc/named.conf 粗略分為兩個部分。其中一個是一般設定的 options 區段，另一個則是由個別網域的 zone 項目組成。logging 區段和 acl (存取控制清單) 項目是選擇性的。註解行的開頭是 # 符號或 //。在 範例 33.2「基本的 /etc/named.conf」 [570頁] 中顯示了最基本的 /etc/named.conf。

範例 33.2 *基本的 /etc/named.conf*

```
options {
        directory "/var/lib/named";
        forwarders { 10.0.0.1; };
        notify no;
};

zone "localhost" in {
       type master;
       file "localhost.zone";
};

zone "0.0.127.in-addr.arpa" in {
        type master;
        file "127.0.0.zone";
};

zone "." in {
        type hint;
        file "root.hint";
};
```

33.4.1 重要組態選項

directory "*filename*";
指定BIND可以在其中找尋包含區域資料之檔案的目錄。通常是`/var/lib/named`。

forwarders { *ip-address*; };
指定無法直接解析 DNS 要求時應該將其轉遞至哪個名稱伺服器 (一般屬於提供者)。以 `10.0.0.1` 之類的 IP 位址取代 *ip-address*。

forward first;
將會在嘗試透過根名稱伺服器解析DNS要求之前先加以轉遞。除了`forward first`，可以寫入`forward only`以轉遞所有要求，且不會有任何要求傳送到根名稱伺服器。這對於防火牆組態是可以理解的。

listen-on port 53 { 127.0.0.1; *ip-address*; };
告訴 BIND 哪個網路介面和哪個連接埠要接受用戶端查詢。`port 53` 不需要明確指定，因為 `53` 是預設連接埠。輸入 `127.0.0.1` 將允許來自本地主機的要求。如果完全省略此項目，預設會使用所有介面。

listen-on-v6 port 53 {any; };
告訴 BIND 哪個連接埠應該傾聽 IPv6 用戶端要求。除了 `any` 外只能使用 `none`。就 IPv6 而言，伺服器僅接受萬用字元位址。

query-source address * port 53;
如果防火牆封鎖 DNS 要求外送，則需要這個項目。這樣會告訴 BIND 從外部的連接埠 53 張貼要求，而不是從任何高於 1024 的連接埠張貼。

query-source-v6 address * port 53;
告訴 BIND 哪個連接埠用於 IPv6 查詢。

allow-query { 127.0.0.1; *net*; };
定義用戶端可以張貼 DNS 要求的網路。使用如 `192.168.1/24` 的位址資訊取代 *net*。尾端的 `/24` 是網路遮罩的縮寫表示式，在此例中為 `255.255.255.0`。

allow-transfer ! *;;
控制哪些主機可以要求區域傳輸。在範例中，這類要求是使用 ！ *. 如果沒有這個項目，就可以從任一處要求區域傳輸，沒有限制。

statistics-interval 0;
: 如果沒有這個項目，BIND 每小時都會在 `/var/log/messages` 中產生數行統計資訊。指定 0 則完全不會顯示這些統計數字，或設定以分鐘為單位的間隔時間。

cleaning-interval 720;
: 此選項定義 BIND 清除其快取記憶體的時間間隔。這樣每次清除時會觸發 `/var/log/messages` 中的項目。時間規格單位為分鐘。預設值是 60 分鐘。

interface-interval 0;
: BIND 會定期搜尋網路介面，尋找新的或不存在的介面。如果此值設定為 `0`，就不會執行這個動作，且 BIND 僅會傾聽啟動時偵測到的介面。如果不想出現這種情況，請以分鐘為單位定義間隔時間。預設值是 60 分鐘。

notify no;
: 當變更區域資料或重新啟動名稱伺服器時，`no` 會防止通知其他名稱伺服器。

33.4.2 記錄

記錄的內容、方式及位置皆可在 BIND 中詳細設定。一般而言，預設設定應該足夠。範例 33.3「關閉記錄的項目」[572頁]顯示出這類項目的最簡單格式，而且完全停用任何記錄。

範例 33.3 *關閉記錄的項目*

```
logging {
        category default { null; };
};
```

33.4.3 區域項目

範例 33.4 *my-domain.de 的區域項目*

```
zone "my-domain.de" in {
      type master;
      file "my-domain.zone";
      notify no;
};
```

在 zone 之後，指定要管理的網域名稱 (my-domain.de)，後面跟上 in 以及大括號包住的相關選項區塊，如範例 33.4「my-domain.de 的區域項目」[572頁]中所示。若要定義 *slave zone*，切換 type 為 slave 並指定管理此區域的名稱伺服器為 master (也可能成為另一個主要的從屬)，如 範例 33.5「other-domain.de 的區域項目」[573頁] 中所示。

範例 33.5 *other-domain.de 的區域項目*

```
zone "other-domain.de" in {
      type slave;
      file "slave/other-domain.zone";
      masters { 10.0.0.1; };
};
```

區域選項：

type master;
: 藉由指定 master，可以告訴 BIND 區域由本地名稱伺服器處理。這假設區域檔案已經以正確格式建立。

type slave;
: 此區域傳輸自另一部名稱伺服器。必須與 masters 一起使用。

type hint;
: 區域 . 屬於 hint 類型，可用來設定根名稱伺服器。此區域定義可以維持原狀。

file my-domain.zone or file「slave/other-domain.zone」;
: 此項目可指定網域之區域資料所在的檔案。從屬區域並不需要此檔案，它們從另一部名稱伺服器取得此資料。若要分別主要和從屬檔案，請為從屬檔案使用目錄 slave。

masters { *server-ip-address*; };
: 僅從屬區域需要此項目。它指定應該傳輸區域檔案的名稱伺服器。

allow-update {! *; };
: 此選項可控制外部寫入存取，將允許用戶端產生 DNS 項目 — 出於安全性考量，通常不應使用此項目。如果沒有此項目，將禁止區域更新。以下項目會產生相同的結果，因為 ! * 可有效地禁止任何這類活動。

33.5 區域檔案

需要兩種類型的區域檔案。一個會指派 IP 位址給主機名稱，另一個的作用恰恰相反：為 IP 位址提供主機名稱。

提示：在區域檔案中使用點符號

每則訊息前的 . 在區域檔案中具有重要意義。如果主機名稱最後不用 . 結尾，則會附加區域。與完整網域名稱一同指定的完整主機名稱必須以 . 結尾，才能避免再次附加網域。缺少點符號或錯置其位置最經常造成名稱伺服器組態錯誤。

第一個要考慮的情況是負責領域 `world.cosmos` 的區域檔案 `world.zone`，如 範例 33.6「檔案 /var/lib/named/world.zone」 [574頁] 中所示。

範例 33.6 *檔案 /var/lib/named/world.zone*

```
$TTL 2D
world.cosmos. IN SOA      gateway  root.world.cosmos. (
            2003072441  ; serial
            1D          ; refresh
            2H          ; retry
            1W          ; expiry
            2D )        ; minimum

            IN NS       gateway
            IN MX       10 sun

gateway     IN A        192.168.0.1
            IN A        192.168.1.1
sun         IN A        192.168.0.2
moon        IN A        192.168.0.3
earth       IN A        192.168.1.2
mars        IN A        192.168.1.3
www         IN CNAME    moon
```

行 1：

`$TTL` 定義應該套用到此檔案中所有項目的預設有效時間。在此範例中，項目的有效時間是兩天 (`2 D`)。

行 2：

這是 SOA (授權開始) 控制記錄開始的地方：

- 首位上的 world.cosmos 為要管理的網域名稱。名稱以 . 結尾，以免第二次附加區域。或者，可以在此輸入 @，這樣會從 /etc/named.conf 中的對應項目擷取區域。

- 在 IN SOA 之後是名稱伺服器的名稱，做為此區域的主伺服器。名稱會從 gateway 擴展為 gateway.world.cosmos，因為它沒有用 . 結尾。

- 後面跟著此名稱伺服器之負責人的電子郵件地址。因為 @ 符號已經具有特殊意義，所以在此輸入 . 來代替。對於 root@world.cosmos，項目必須寫成 root.world.cosmos.。每則訊息前的 .，以防止新增區域。

- （將所有到）的行都包含在 SOA 記錄中。

行 3：

serial number 是任意號碼，每次此檔案變更時就會增加。通知次要名稱伺服器 (從屬伺服器) 發生變更，這是必要的。對於這種情形，十個數字的日期及執行號碼，寫法是 YYYYMMDDNN，已成為習慣格式。

行 4：

refresh rate 指定次要名稱伺服器確認區域 serial number 的時間間隔。在此例中，是一天。

行 5：

retry rate 指定在發生錯誤時次要名稱伺服器嘗試再次聯絡主要伺服器的時間間隔。在此例中，是兩小時。

行 6：

expiration time 指定次要名稱伺服器無法重新取得與主要伺服器的聯絡時，在此時間範圍後丟棄快取資料。在此例中，是一週。

行 7：

SOA 記錄中的最後一個項目，指定 negative caching TTL — 亦即可在此時間內快取其他伺服器無法解析之 DNS 查詢的結果。

行 9：

IN NS 指定負責此網域的名稱伺服器。gateway 會擴充為 gateway.world.cosmos，因為它沒有以 . 結尾。可能會有數行與此類似 — 主要名稱伺服器佔用一行，每部次要名稱伺服器也各自佔用一行。如果 /etc/named.conf 中的 notify 不是設定為 no，此處列出的所有名稱伺服器會收到區域資料變更的通知。

行 10：

MX 記錄指定為網域 `world.cosmos` 接收、處理和轉遞電子郵件的郵件伺服器。在此範例中，郵件伺服器為主機 `sun.world.cosmos`。主機名稱前的號碼是偏好設定值。如果有多個 MX 項目，會先優先使用具有最小值的郵件伺服器，而如果郵件無法送到此伺服器，就會嘗試使用下一個較高的值。

行 12–17：

這些是指派給主機名稱的一或多個 IP 位址的實際位址記錄。此處列出的名稱不含 `.`，因為它們不包含其網域，所以會將 `world.cosmos` 新增到所有名稱。兩個 IP 位址指派給主機 `gateway`，因為它有兩張網路卡。如果主機位址是傳統位址 (IPv4)，記錄會使用 `AAAA` 標示。如果位址是 IPv6 位址，項目會使用 `AAAA 0` 標示。IPv6 位址之前的記號只包括 `AAAA`，現在已廢除不用。

注：IPv6 語法

IPv6 記錄與 IPv4 的語法稍有不同。因為可以分段，所以必須在位址前提供有關遺漏位元的資訊。即使您要使用完全未分段的位址，也必須提供此資訊。對於使用如下語法的 IPv4 記錄

```
pluto IN            AAAA 2345:00C1:CA11:0001:1234:5678:9ABC:DEF0
pluto IN            AAAA 2345:00D2:DA11:0001:1234:5678:9ABC:DEF0
```

您在 IPv6 格式中必須新增有關遺漏位元的資訊。因為上述範例是完整的 (未遺漏任何位元)，所以此記錄的 IPv6 格式為：

```
pluto   IN            AAAA 0 2345:00C1:CA11:0001:1234:5678:9ABC:DEF0
pluto   IN            AAAA 0 2345:00D2:DA11:0001:1234:5678:9ABC:DEF0
```

請勿搭配使用 IPv4 位址與 IPv6 對應。

行 18：

別名 `www` 可以用來定址 `mond` (`CNAME` 表示 *canonical name* (標準名稱))。

虛擬網域 `in-addr.arpa` 用來反向查詢 IP 位址到主機名稱。它會以反向標記法附加到位址的網路部分。因此 `192.168.1` 會解析為 `1.168.192.in-addr.arpa`。請參閱範例 33.7「反向查詢」[577頁]。

範例 33.7 *反向查詢*

```
$TTL 2D
1.168.192.in-addr.arpa. IN SOA gateway.world.cosmos. root.world.cosmos. (
                        2003072441      ; serial
                        1D              ; refresh
                        2H              ; retry
                        1W              ; expiry
                        2D )            ; minimum

                        IN NS           gateway.world.cosmos.

1                       IN PTR          gateway.world.cosmos.
2                       IN PTR          earth.world.cosmos.
3                       IN PTR          mars.world.cosmos.
```

行 1：
$TTL 定義套用到此處所有項目的標準 TTL。

行 2：
組態檔應該為網路 `192.168.1.0` 啟用反向查詢。假設區域稱為 `1.168.192.in-addr.arpa`，則不應該新增到主機名稱。因此輸入的主機名稱都使用完整格式 — 附帶網域並以 `.` 做為結尾。其餘的項目與之前 `world.cosmos` 範例中所述的項目相同。

行 3 – 7：
請參閱之前的 `world.cosmos` 範例。

行 9：
同樣地，此行指定負責此區域的名稱伺服器。不過，這一次，以完整格式輸入名稱，亦即包含網域以及結尾的 。

行 11 – 13：
這些是相關主機上 IP 位址的指標記錄提示。行的開頭僅輸入了 IP 位址的最後一部分，結尾無 `.`。對此附加區域(不加上 `.in-addr.arpa`) 會造成完整 IP 位址變成反向順序。

通常，不同 BIND 版本之間的區域傳輸應該是沒有問題的。

33.6 區域資料的動態更新

「*動態更新*」這個詞是指新增、變更或刪除主伺服器的區域檔案項目的作業。此機制於 RFC 2136 中有詳細描述。利用新增選擇性的 `allow-update` 或 `update-policy` 規則，可為每個區域項目個別設定動態更新。動態更新的區域不應該手動修改。

使用指令 `nsupdate` 將要更新的項目傳送到伺服器。如需此指令的完整語法，請查閱 nsupdate 的手冊頁 (`man 8 nsupdate`)。為了安全性的緣故，這類更新應該使用 TSIG 金鑰加以執行，如 第 33.7 節「安全交易」[578頁] 所述。

33.7 安全交易

透過採用共享秘密金鑰 (也稱為 TSIG 金鑰) 的交易簽章 (TSIG)，可以實現安全交易。本節說明如何產生及使用這類金鑰。

不同伺服器之間的通訊，以及區域資料的動態更新，都需要安全交易。讓存取控制依靠金鑰比單純依靠 IP 位址要來得安全許多。

使用以下指令可產生 TSIG 金鑰 (有關詳細資訊，請參閱 `man dnssec-keygen`)：

```
dnssec-keygen -a hmac-md5 -b 128 -n HOST host1-host2
```

這樣會建立兩個檔案，名稱類似如下：

```
Khost1-host2.+157+34265.private Khost1-host2.+157+34265.key
```

金鑰本身 (如 `ejIkuCyyGJwwuN3xAteKgg==` 的字串) 在兩個檔案中都可找到。如果要用於交易，第二個檔案 (`Khost1-host2.+157+34265.key`) 必須傳輸到遠端主機，最好是以安全的方式傳輸 (例如，使用 scp)。在遠端伺服器上，金鑰必須包含於檔案 `/etc/named.conf` 內，才能開啟 `host1` 與 `host2` 之間的安全通訊：

```
key host1-host2. {
 algorithm hmac-md5;
 secret ";ejIkuCyyGJwwuN3xAteKgg==;
};
```

警告：/etc/named.conf*的檔案權限*

請確定 /etc/named.conf 的許可權受到適當的限制。此檔案的預設值是 0640，擁有者為 root 及群組 named。另一種方法是，將金鑰移到具有特別限定許可權的其他檔案，然後將檔案從 /etc/named.conf 包含進來。若要包含外部檔案，請使用：

```
include  "filename"
```

將*檔案名稱*取代為金鑰所在檔案的絕對路徑。

若要讓伺服器 host1 能夠使用 host2 (在此範例中位址為 192.168.2.3) 的金鑰，伺服器的 /etc/named.conf 必須包含以下規則：

```
server 192.168.2.3 {
  keys { host1-host2. ;};
};
```

類比項目必須包含於 host2 的組態檔中。

針對為 IP 位址和位址範圍定義的任何 ACL (存取控制清單，切勿與檔案系統 ACL 混淆) 新增 TSIG 金鑰，以確保交易安全性。對應項目應該看起來如下：

```
allow-update { key host1-host2. ;};
```

此主題在 update-policy 下的 *BIND Administrator Reference Manual* 中有詳細討論。

33.8 DNS 安全性

DNSSEC 或 DNS 安全性細述於 RFC 2535。DNSSEC 的可用工具在 BIND 手冊中有詳加討論。

與一或多個區域金鑰關聯的區域才是安全區域。與主機金鑰一樣，這些金鑰也是使用 dnssec-keygen 產生。目前是使用 DSA 加密演算法產生這些金鑰。產生的公用金鑰應該包含於套用 $INCLUDE 規則的對應區域檔案中。

藉由指令 dnssec-makekeyset，產生的所有金鑰封裝為一組，必須透過安全的方法將其傳輸到父區域。在父區域上，會使用 dnssec-signkey 簽署金鑰

組。接著會使用透過此指令產生的檔案，以 dnssec-signzone 指令簽署區域，然後會為 /etc/named.conf 中每個區域產生要包含在內的檔案。

33.9 如需更多資訊

如需其他資訊，請參閱 bind-doc 套件的 *BIND Administrator Reference Manual* (安裝於 /usr/share/doc/packages/bind/ 下)。另外也請參閱手冊參考的 RFC 以及 BIND 隨附的手冊頁。有關 SUSE Linux Enterprise 之 BIND 的最新資訊，請參閱 /usr/share/doc/packages/bind/README.SuSE。

34

DHCP

動態主機組態通訊協定 (DHCP) 用於從伺服器集中指派網路設定，而不是在每個工作站上分別設定。設定要使用 DHCP 的主機對於自己的靜態位址並沒有控制權。它可以根據伺服器的指示完全且自動地設定自己本身。如果您在用戶端使用 NetworkManager，就完全不必設定用戶端。這在環境多變、而且一次只使用一個介面的情況下非常有用。絕對不要在執行 DHCP 伺服器的機器上使用 NetworkManager。

提示：IBM System z：DHCP 支援

在 IBM System z 平台上，DHCP 僅能用於使用 OSA 和 OSA Express 網路卡的介面。這些網路卡是唯一使用 MAC 的網路卡，是 DHCP 自動組態功能的必要元件。

DHCP 伺服器的一種設定方式是識別每個使用網路卡硬體位址 (大部分的情況下是固定的) 的用戶端，以後每次用戶端連接伺服器時，將提供相同的設定值給用戶端。或者也可以將 DHCP 設定成從專門設定的位址池動態指派位址給每個相關用戶端。在後者的情形中，DHCP 伺服器在每次接到要求時會指派相同的位址給用戶端，即使經過較長的時間也是一樣。這只適用於網路用戶端數少於網路位址數的情況。

DHCP 可簡化系統管理員的工作。任何與位址及一般網路組態相關的變更 (包括較大的變更) 都可集中執行，只要編輯伺服器的組態檔。與重新設定眾多的工作站相比，這種方法要便利許多。另外也更易於整合機器 (特別是新機器) 到網路中，因為可以從集區對其指定 IP 位址。從 DHCP 伺服器擷取適當網路設定，對於經常使用不同網路的筆記型電腦特別有用。

DHCP 伺服器不僅提供 IP 位址和網路遮罩，也提供主機名稱、網域名稱、閘道和名稱伺服器位址供用戶端使用。除此之外，DHCP 也允許集中設定一些其他的參數，例如，用戶端可以輪詢目前時間的時間伺服器或甚至是列印伺服器。

34.1 使用 YaST 設定 DHCP 伺服器

重要：LDAP 支援

在此 SUSE Linux Enterprise 版本中，YaST DHCP 模組可以設定成在本地儲存伺服器組態 (在執行 DHCP 伺服器的主機上)，或是由 LDAP 伺服器來管理其組態資料。如果要使用 LDAP，應在設定 DHCP 伺服器之前先設定您的 LDAP 環境。

YaST DHCP 模組允許您為區域網路設定您自己的 DHCP 伺服器。模組能夠以簡單模式或是進階模式運作。

34.1.1 初始組態 (精靈)

第一次啟動模組時，會啟動精靈以提示您決定有關伺服器管理的一些基本設定。完成此初始設定程序會產生一個非常基本的伺服器組態，適用於基本的狀況。進階模式可以處理更多進階設定任務。

選擇介面卡

在第一步驟中，YaST 會尋找系統上可用的網路介面，然後顯示清單。從清單中選取 DHCP 伺服器要傾聽的介面，並按一下「*新增*」，然後選取「*開啟選取介面的防火牆*」以開啟此介面的防火牆。請參閱圖形 34.1 「DHCP 伺服器：介面卡選項」 [583頁]。

圖形 34.1　DHCP 伺服器：介面卡選項

全域設定

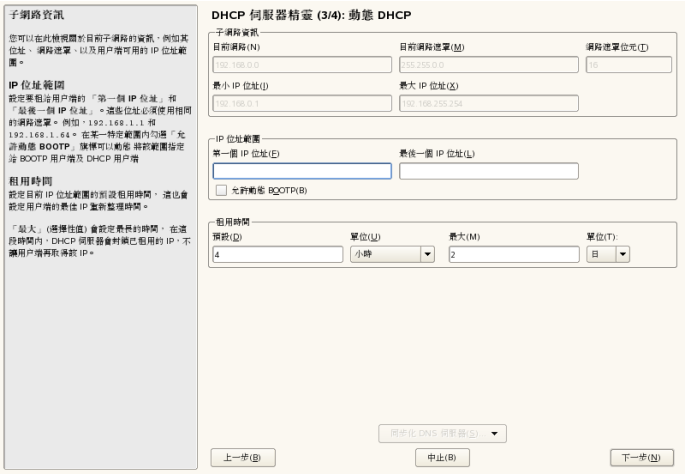
使用核取方塊決定您的 DHCP 設定是否要由 LDAP 伺服器自動儲存。在輸入欄位，提供 DHCP 伺服器應該管理的所有用戶端的網路細節。這些細節包括網域名稱、時間伺服器的位址、主要及次要名稱伺服器的位址、列印和 WINS 伺服器的位址 (供同時具有 Windows 與 Linux 用戶端的混和網路使用)、閘道位址以及租用時間。請參閱圖形 34.2「DHCP 伺服器：全域設定」[584頁]。

圖形 34.2　DHCP 伺服器：全域設定

動態 DHCP

在此步驟中，設定應該如何將動態 IP 位址指派給用戶端。若要這樣做，指定伺服器可以指派位址給 DHCP 用戶端的 IP 範圍。所有這些位址應該涵蓋在相同的網路遮罩下。另外也指定租用時間，在此時間段內用戶端可以保留其 IP 位址，無需要求延續租用。或者，指定最長租用時間，也就是伺服器保留特定用戶端之 IP 位址的時間。請參閱圖形 34.3「DHCP 伺服器：動態 DHCP」[585頁]。

圖形 34.3　DHCP 伺服器：動態 DHCP



完成組態及設定啟動模式

在組態精靈的第三步驟後，會顯示最後一個對話方塊，用於定義 DHCP 伺服器的啟動方式。在此您可以指定是在系統開機時自動啟動DHCP伺服器，還是在需要時手動啟動(例如，為了測試)。按一下「*完成*」完成伺服器的組態。請參閱圖形 34.4「DHCP 伺服器：啟動」[586頁]。另一種方式是，從左邊樹狀結構中選取「*主機管理*」，設定基本組態之外的特殊主機管理功能(請參閱 圖形 34.5「DHCP 伺服器：主機管理」[587頁])。

圖形 34.4　DHCP 伺服器：啟動

主機管理

除了以上一節描述的方式使用動態 DHCP 外，您也可以設定伺服器以準靜態方式指派位址。若要這樣做，使用下半部所提供的輸入欄位指定用戶端清單，以這種方式進行管理。具體而言，就是提供指定給這類用戶端的「*名稱*」和「*IP 位址*」，也要指定「*硬體位址*」和「*網路類型*」(記號環或乙太網路)。使用「*新增*」、「*編輯*」和「*從清單中刪除*」修改顯示於上半部的用戶端清單。請參閱圖形 34.5「DHCP 伺服器：主機管理」[587頁]。

圖形 34.5　DHCP 伺服器：主機管理

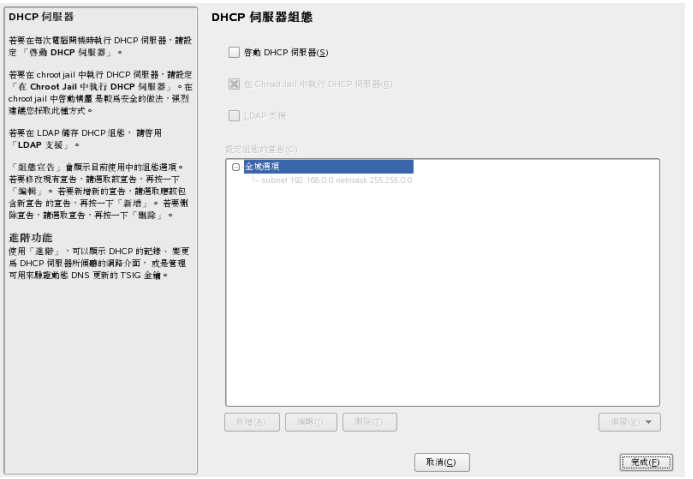
34.1.2 進階組態

除了稍早討論的組態方法外，還有進階組態模組，可讓您變更 DHCP 伺服器設定的每個細節。選取對話方塊左邊樹狀檢視中的「*進階設定*」，開始進階組態。

Chroot 環境與宣告

在此第一個對話方塊中，選取「*啟動DHCP 伺服器*」讓現有組態可以編輯。DHCP 伺服器行為的一個重要功能是它能夠在 chroot 環境或 chroot jail 中執行，以確保伺服器主機的安全。如果 DHCP 伺服器會遭受外部攻擊，駭客將會被阻絕於 chroot jail 的保護之外，防止其染指系統的其他部分。對話方塊的下半部顯示已經定義之宣告的樹狀檢視。使用「*新增*」、「*刪除*」和「*編輯*」修改這些宣告。選取「*進階*」將帶您到其他的進階對話方塊。請參閱圖形 34.6「DHCP 伺服器：Chroot Jail 和宣告」[588頁]。 選取「*新增*」後，定義要新增的宣告類型。使用「*進階*」，檢視伺服器的記錄檔、設定 TSIG 金鑰管理，並根據 DHCP 伺服器的設定調整防火牆的組態。

圖形 34.6　DHCP 伺服器：Chroot Jail 和宣告



選取宣告類型

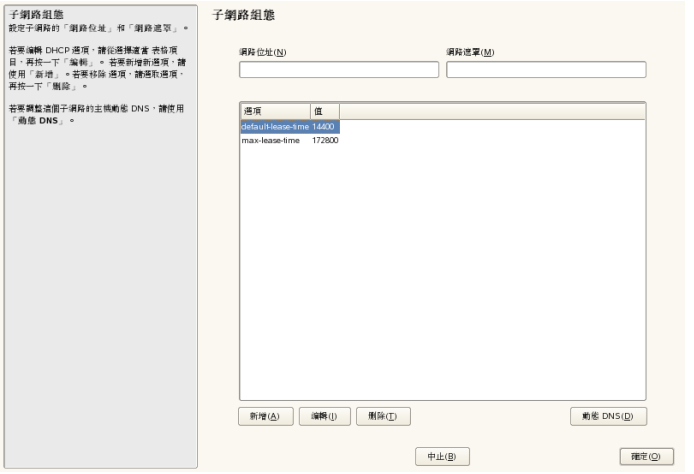
DHCP伺服器的「*全域選項*」由數個宣告組成。此對話方塊可讓您設定宣告類型：「*子網路*」、「*主機*」、「*共享網路*」、「*群組*」、「*位址池*」和「*類別*」。此範例顯示新子網路的選項(請參閱 圖形 34.7「DHCP 伺服器：選取宣告類型」[588頁])。

圖形 34.7　DHCP 伺服器：選取宣告類型

子網路組態

此對話方塊可讓您使用 IP 位址及網路遮罩來指定新的子網路。在對話方塊的中間部分，使用「*新增*」、「*編輯*」和「*刪除*」修改所選子網路的DHCP伺服器啟動選項。若要設定子網路的動態 DNS，選取「*動態 DNS*」。

圖形 34.8 *DHCP 伺服器：設定子網路*



TSIG 金鑰管理

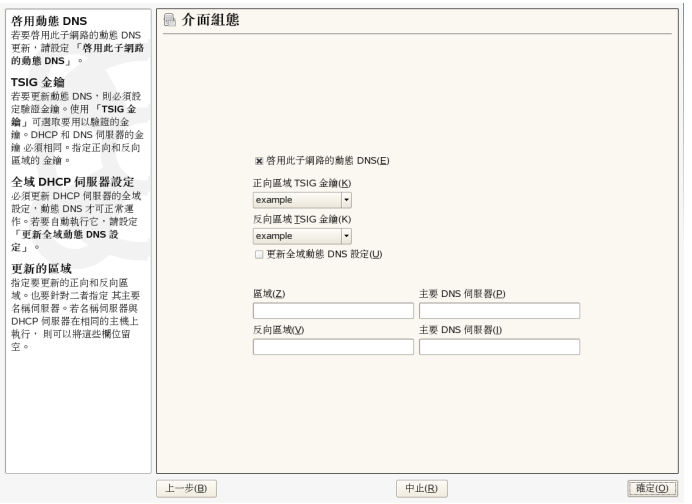
如果在上一個對話方塊中選擇設定動態 DNS，現在可以針對安全區域傳輸設定金鑰管理。選取「*確定*」會帶您到另一個對話方塊，讓您設定動態DNS的介面 (請參閱 圖形 34.10「DHCP 伺服器：動態 DNS 的介面組態」[591頁])。

圖形 34.9　DHCP 伺服器：TSIG 組態

動態 DNS：介面組態

選取「*啟用此子網路的動態 DNS*」，即可啟用子網路的動態 DNS。執行這個動作後，使用下拉式清單選擇轉遞和反向區域的 TSIG 金鑰，確定這些金鑰對於 DNS 與 DHCP 伺服器都是相同的。使用「*更新全域動態 DNS 設定*」，來依據動態 DNS 環境自動更新及調整全域 DHCP 伺服器設定。最後，定義每個動態 DNS 的哪些轉遞和反向區域需要更新，為兩個區域都指定主要名稱伺服器的名稱。如果名稱伺服器與 DHCP 伺服器在同一部主機上執行，可以將這些欄位保留空白。選取「*確定*」返回子網路組態對話方塊 (請參閱 圖形 34.8「DHCP 伺服器：設定子網路」[589頁])。再次選取「*確定*」返回最初的進階組態對話方塊。

圖形 34.10　DHCP 伺服器：動態 DNS 的介面組態



網路介面組態

若要定義 DHCP 伺服器傾聽的介面以及調整防火牆組態，請從進階組態對話方塊選取「*進階*」>「*介面組態*」。從顯示的介面清單中，選取一或多個要由 DHCP 伺服器處理的介面。如果所有子網路中的用戶端能夠與伺服器通訊，且伺服器主機也要執行防火牆，請依照需要調整防火牆。若要這樣做，請選取「*調整防火牆設定*」。接著 YaST 會將 SuSEfirewall2 的規則調整為新的條件 (請參閱圖形 34.11「DHCP 伺服器：網路介面和防火牆」[592頁])，之後您即可選取「*確定*」返回最初的對話方塊。

圖形 34.11　DHCP 伺服器：網路介面和防火牆

完成所有組態步驟後，按一下「*確定*」關閉對話方塊。伺服器現在會以新的組態啟動。

34.2 DHCP 軟體套件

DHCP 伺服器與 DHCP 用戶端皆可用於 SUSE Linux Enterprise。可用的 DHCP 伺服器是 dhcpd (由 Internet Software Consortium 發佈)。對於用戶端，可於兩個不同的 DHCP 用戶端程式間做出選擇：`dhcp-client` (也來自 ISC) 以及 `dhcpcd` 套件中的 DHCP 用戶端精靈。

SUSE Linux Enterprise 預設會安裝 dhcpcd。該程式易於使用，而且會在每次系統開機時自動啟動，尋找 DHCP 伺服器。它不需要組態檔來執行其工作，而且大部分的標準設定可以直接使用。如果情況較為複雜，請使用 ISC dhcp-client，此程式可透過組態檔 `/etc/dhclient.conf` 來控制。

34.3 DHCP 伺服器 dhcpd

任何 DHCP 系統的核心都是動態主機組態通訊協定精靈。這個伺服器會依照組態檔 `/etc/dhcpd.conf` 中定義的設定，*租用*位址並監看其使用情形。藉由變更此檔案中的參數及值，系統管理員可以透過數種方式影響程式的行為。請參

閱 範例 34.1「組態檔 /etc/dhcpd.conf」[593頁] 中的 /etc/dhcpd.conf 基本範例檔案。

範例 34.1 *組態檔 /etc/dhcpd.conf*

```
default-lease-time 600;          # 10 minutes
max-lease-time 7200;             # 2  hours

option domain-name "cosmos.all";
option domain-name-servers 192.168.1.1, 192.168.1.2;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option subnet-mask 255.255.255.0;

subnet 192.168.1.0 netmask 255.255.255.0
 {
  range 192.168.1.10 192.168.1.20;
  range 192.168.1.100 192.168.1.200;
 }
```

這個簡單的組態檔應足以讓 DHCP 伺服器在網路中指派 IP 位址。請確定每行結尾都插入分號，否則不會啟動 dhcpd。

範例檔可以分為三個部分。第一個部分定義要求用戶端預設可租用 IP 位址的秒數(default-lease-time)，此時間過後將需要申請續約。其中也包含機器保留 DHCP 伺服器指定之 IP 位址、不用申請續約的最長時間的陳述式(max-lease-time)。

在第二部份中，一些基本網路參數定義於全域層級：

- 行 option domain-name 定義了您網路的預設網域。
- 利用 option domain-name-servers 項目，最多可指定三個 DNS 伺服器值，用來將 IP 位址解析為主機名稱，或將主機名稱解析為 IP 位址。最好在設定 DHCP 前先設定機器上或網路上其他位置的名稱伺服器。該名稱伺服器也應為每個動態位址定義主機名稱，或為每個主機名稱定義動態位址。若要瞭解如何設定您自己的名稱伺服器，請參閱 第 33 章「*網域名稱系統*」[559頁]。
- option broadcast-address 這一行定義要求用戶端應使用的廣播位址。

- 利用 `option routers`，可設定伺服器要將無法傳送至區域網路上之主機的資料封包傳送到什麼地方 (依據提供的來源和目的主機位址及子網路遮罩)。在大多數的情況中，特別是較小的網路，此路由與網際網路閘道是完全一樣的。

- 利用 `option subnet-mask`，指定指定給用戶端的網路遮罩。

檔案的最後一個部分定義網路，包括子網路遮罩。若要完成設定，請指定DHCP精靈用來指派 IP 位址給相關用戶端的位址範圍。在 範例 34.1 「組態檔 /etc/dhcpd.conf」 [593頁] 中，用戶端的指定位址可介於 `192.168.1.10` 與 `192.168.1.20` 之間以及 `192.168.1.100` 與 `192.168.1.200` 之間。

編輯這幾行後，就能夠使用指令 `rcdhcpd start` 啟用 DHCP 精靈。該精靈可以立即使用。使用指令 `rcdhcpd check-syntax` 執行簡短的語法檢查。如果在進行設定時遇到未預期的問題，例如伺服器出現錯誤而中止或在啟動時沒有傳回 `done`，則查詢主系統記錄 `/var/log/messages` 或主控台 10 (Ctrl + Alt + F10) 上的資訊，應可找出問題所在。

在預設的 SUSE Linux Enterprise 系統上，出於安全性考量，DHCP 精靈會在 chroot 環境中啟動。組態檔必須複製到 chroot 環境，如此精靈才可以找到這些檔案。通常情況下不需要擔心發生這種情形，因為指令 `rcdhcpd start` 會自動複製檔案。

34.3.1 使用固定 IP 位址的用戶端

DHCP 也可以用來指派預先定義的靜態位址給特定用戶端。明確指派的位址永遠比集區的動態位址優先。當可用位址不足，伺服器必須在用戶端之間重新分配位址時 (舉例而言)，動態位址便會過期，靜態位址則不同，它永遠不會過期。

為了識別使用靜態位址設定的用戶端，dhcpd 會使用硬體位址，硬體位址是全球唯一的固定數字代碼，由六對八位元組組成，用來識別所有網路設備 (例如，`00:00:45:12:EE:F4`)。如果相對的行 (如 範例 34.2 「組態檔的增加部分」 [595頁] 中所示) 新增到 範例 34.1 「組態檔 /etc/dhcpd.conf」 [593頁] 的組態檔，DHCP 精靈始終會指定相同的資料集給對應的用戶端。

範例 34.2 *組態檔的增加部分*

```
host earth {
hardware ethernet 00:00:45:12:EE:F4;
fixed-address 192.168.1.21;
}
```

在第一行輸入對應用戶端的名稱(`host` *`hostname`*，範例中為`earth`)，在第二行輸入 MAC 位址。在 Linux 主機上，使用指令 `ip link show` 並在後面接上網路設備 (例如，`eth0`) 可尋找 MAC 位址。輸出應該包含如下內容

```
link/ether 00:00:45:12:EE:F4
```

在上述範例中，系統會自動指派 IP 位址 `192.168.1.21` 及主機名稱 earth 給具有網路卡且 MAC 位址為 `00:00:45:12:EE:F4` 的用戶端。幾乎所有情況中輸入的硬體類型會是`ethernet`，儘管通常在IBM系統上找到的是`token-ring`，也是可以支援的。

34.3.2 SUSE Linux Enterprise 版本

為了改善安全性，SUSE Linux Enterprise 版 ISC DHCP 伺服器在提供時即已應用 Ari Edelkind 的 non-root/chroot 修補程式。這樣可讓 dhcpd 利用使用者 ID `nobody` 執行，而且也能在 chroot 環境下 (`/var/lib/dhcp`) 執行 dhcpd。若要這樣做，組態檔 `dhcpd.conf` 必須位於 `/var/lib/dhcp/etc`。init 程序檔啟動時會自動複製檔案到此目錄。

透過檔案 `/etc/sysconfig/dhcpd` 中的項目，可控制伺服器與此功能相關的行為。若是不要在 chroot 環境下執行 dhcpd，請將 `/etc/sysconfig/dhcpd` 中的 `DHCPD_RUN_CHROOTED` 變數設定為「no」。

若要讓 dhcpd 從 chroot 環境內解析主機名稱，必須也要複製以下其他組態檔案：

- `/etc/localtime`
- `/etc/host.conf`
- `/etc/hosts`
- `/etc/resolv.conf`

啟動 init 程序檔時，這些檔案會複製到 /var/lib/dhcp/etc/。如果 /etc/ppp/ip-up 之類的程序檔動態修改了這些檔案，則進行所需變更時，也要考量到這些副本。不過，如果組態檔僅指定 IP 位址 (而不是指定主機名稱) 時，則不需要擔心。

如果組態包含要複製到 chroot 環境的其他檔案，請在檔案 /etc/sysconfig/dhcpd 中的變數 DHCPD_CONF_INCLUDE_FILES 下設定這些檔案。為了確保重新啟動 syslog-ng 精靈後，DHCP 記錄設備仍然能夠持續運作，/etc/sysconfig/syslog 檔案中還有另一個項目 SYSLOGD_ADDITIONAL_SOCKET_DHCP。

34.4 如需更多資訊

如需有關 DHCP 的詳細資訊，請參閱 *Internet Software Consortium* 的網站 (http://www.isc.org/products/DHCP/)。在 dhcpd、dhcpd.conf、dhcpd.leases 和 dhcp-options 線上文件中也可以找到資訊。

35 使用 NIS

一旦網路中的多個 UNIX 系統想要存取常用資源，所有使用者和群組身份對於該網路中所有機器均相同就變得很重要。網路對使用者而言應該透明化：無論他們使用什麼機器，提供給他們的環境應該一律相同。可透過 NIS 和 NFS 服務完成此動作。NFS 透過網路分送檔案系統。

NIS (網路資訊服務) 可描述成像資料庫的服務，提供透過網路對 `/etc/passwd`、`/etc/shadow` 和 `/etc/group` 內容的存取權。NIS 也可用於其他用途 (例如，提供如 `/etc/hosts` 或 `/etc/services` 檔案的內容以供使用)，但這超過本介紹的範圍。人們通常將 NIS 視為 *YP*，因它與網路「黃頁」的概念相似。

35.1 設定 NIS 伺服器

若要在網路上散佈 NIS 資訊，您可以用單一伺服器 (*主要*) 來服務所有用戶端，或是讓 NIS 從屬伺服器向主伺服器要求此資訊，然後轉送給各自的用戶端。

- 若只要為網路設定一部 NIS 伺服器，請用 第 35.1.1 節「設定主要 NIS 伺服器」[598頁] 繼續進行。
- 如果您的主要 NIS 伺服器應該將其資料輸出給次要伺服器，請依 第 35.1.1 節「設定主要 NIS 伺服器」[598頁] 所述設定主要伺服器，並依 第 35.1.2 節「設定次要 NIS 伺服器」[602頁] 所述設定次要伺服器。

35.1.1 設定主要 NIS 伺服器

若要為您的網路設定主要 NIS 伺服器，請執行下列步驟：

1. 啟動「*YaST*」>「*網路服務*」>「*NIS 伺服器*」。

2. 如果網路中只需要一部 NIS 伺服器，或是此伺服器要做為其他 NIS 從屬伺服器的主伺服器，請選「取「*安裝和設定 NIS 主伺服器*」。YaST 就會安裝需要的套件。

   提示

   如果您的機器已安裝 NIS 伺服器軟體，請按一下「*建立主要 NIS 伺服器*」開始建立主要 NIS 伺服器。

圖形 35.1　NIS 伺服器設定

3. 決定基本的 NIS 設定選項：

   3a 輸入 NIS 領域名稱。

   3b 選取「*此主機也是 NIS 用戶端*」，以定義主機是否也是 NIS 用戶端，讓使用者能夠登入，並從 NIS 伺服器存取資料。

選取「*變更密碼*」，讓您網路中的使用者 (包括本機使用者和透過 NIS 伺服器管理的使用者) 能夠變更在 NIS 伺服器的密碼 (使用 `yppasswd` 指令)。

如此便可使用「*允許變更 GECOS 欄位*」和「*允許變更登入外圍程序*」選項。「GECOS」表示使用者還可使用指令 `ypchfn` 變更其名稱與位址設定。「SHELL」讓使用者能夠使用 `ypchsh` 指令變更預設外圍程序，例如從 bash 切換到 sh。新的外圍程序必須是 `/etc/shells` 中預先定義的項目之一。

3c 如果您的 NIS 伺服器應該做為其他子網路中次要 NIS 伺服器的主要伺服器，請選取「*啟用現有的次要 NIS 伺服器*」。

3d 選取「*在防火牆中開啟埠*」讓 YaST 針對 NIS 伺服器調整防火牆設定。

圖形 35.2　主要伺服器設定

3e 按「*下一步*」結束此對話方塊，或按一下「*其他全域設定*」進行其他設定。「*其他全域設定*」包括變更 NIS 伺服器的來源目錄 (預設為 `/etc`)。此外，還可在此處合併密碼。設定值應為「*是*」，以便使用檔案 (`/etc/passwd`、`/etc/shadow` 和 `/etc/group`) 建立使用者資料庫。另外，請決定 NIS 提供的最小使用者和群組 ID。按一下「*確定*」確認您的設定，並返回上一個畫面。

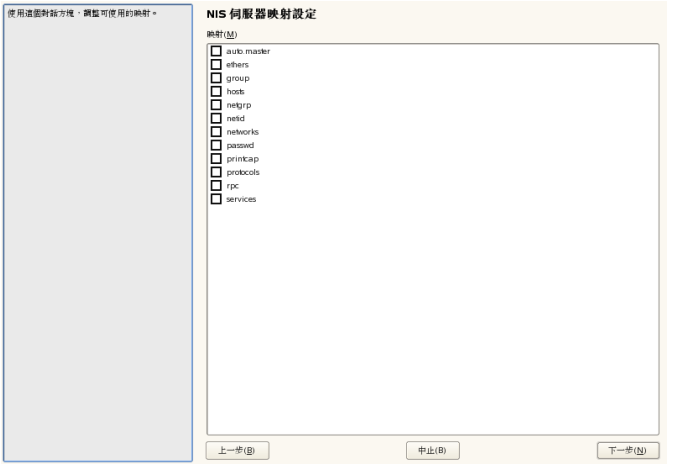
圖形 35.3　變更 NIS 伺服器的目錄與同步化檔案

4 如果您先前啟用了「*可用的從屬 NIS 伺服器存在*」，請輸入要做為次要伺服器的主機名稱然後按「*下一步*」。

5 如果您不使用從屬伺服器，系統會略過從屬伺服器組態，直接跳至資料庫組態的對話方塊。此處，請指定「*映射*」，從NIS伺服器傳輸到用戶端的部分資料庫。預設值通常足夠使用。使用「*下一步*」結束此對話方塊。

6 勾選可用的對應，並按一下「*下一步*」以繼續進行。

圖形 35.4　NIS 伺服器主要設定

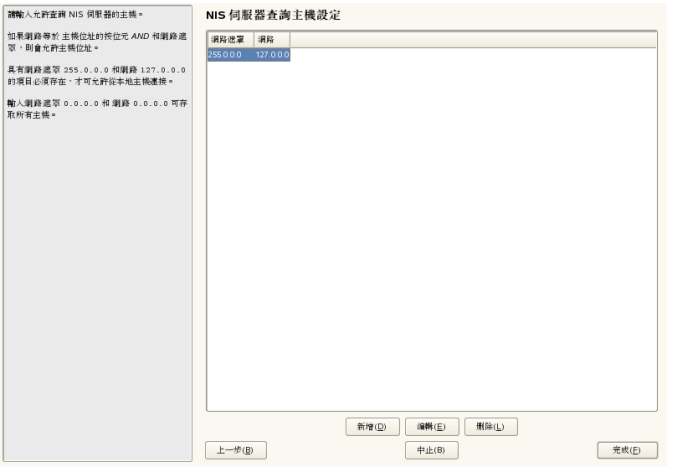


7 輸入可查詢NIS伺服器的主機。您可以按一下適當按鈕新增、編輯或刪除主機。指定可從哪些網路傳送要求至NIS伺服器。通常是您的內部網路。在此例中，應該是下列兩個項目：

```
255.0.0.0       127.0.0.0
0.0.0.0         0.0.0.0
```

第一個項目讓您可從自己的主機連線，即NIS伺服器。第二個項目讓所有主機都可以傳送要求至伺服器。

圖形 35.5　設定 NIS 伺服器的要求權限



8 按一下「*完成*」儲存變更，並結束設定。

35.1.2 設定次要 NIS 伺服器

若要在您的網路中設定其他 NIS *次要伺服器*，請執行下列步驟：

1 啟動「*YaST*」>「*網路服務*」>「*NIS 伺服器*」。

2 選取「*安裝與設定次要 NIS 伺服器*」並按一下「*下一步*」。

提示

如果您的機器已安裝 NIS 伺服器軟體，請按一下「*建立 NIS 次要伺服器*」開始建立 NIS 次要伺服器。

3 完成 NIS 次要伺服器的基本設定：

3a 輸入 NIS 領域。

3b 輸入主要伺服器的主機名稱或 IP 位址。

3c 如果要允許使用者登入此伺服器，請設定「*此主機也是NIS用戶端*」。

3d 使用「*開啟防火牆中的連接埠*」調整防火牆設定。

3e 按一下「下一步」「」。

4 輸入可查詢NIS伺服器的主機。您可以按一下適當按鈕新增、編輯或刪除主機。指定可從哪些網路傳送要求至NIS伺服器。通常，這是所有主機。在此例中，應該是下列兩個項目：

```
255.0.0.0       127.0.0.0
0.0.0.0         0.0.0.0
```

第一個項目讓您可從自己的主機連線，即NIS伺服器。第二個讓所有具有相同網路存取權的主機，可以傳送要求至伺服器。

5 按一下「*完成*」儲存變更，並結束設定。

35.2 設定 NIS 用戶端

使用 YaST 模組「*NIS 用戶端*」可以設定要使用 NIS 的工作站。選擇主機是否有靜態 IP 位址，或是要接受 DHCP 所發出的 IP 位址。DHCP 也可以提供 NIS 領域和 NIS 伺服器。若需有關 DHCP 的詳細資訊，請參閱第 34 章「*DHCP*」[581頁]。如果使用靜態 IP 位址，請手動指定 NIS 領域和 NIS 伺服器。請參閱圖 圖形 35.6「設定 NIS 伺服器的領域和位址」[604頁]。「*尋找*」可讓 YaST 搜尋網路中作用中的 NIS 伺服器。依區域網路的大小而定，這可能會是非常耗時的程序。「*廣播*」會在指定伺服器無法回應後，在區域網路中要求 NIS 伺服器。

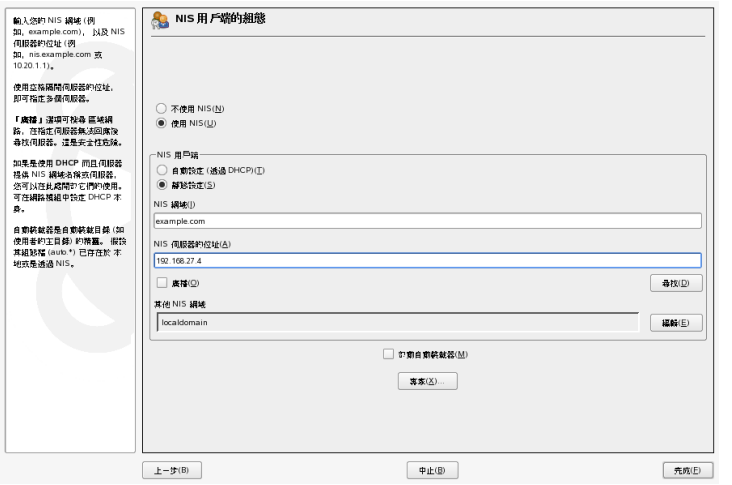
您也可以在「*NIS 伺服器位址*」輸入位址，並以空格分隔，指定多個伺服器。

依本機的安裝內容而定，您可能也想啟用自動掛載器。此選項也會依需要安裝其他軟體。

在進階設定中，如果您不想要其他主機查詢您的用戶端所使用的伺服器，請關閉「*回答遠端主機*」。勾選「*中斷伺服器*」，則用戶端可接收透過未授權的連接埠通訊的伺服器之回覆。若需進一步資訊，請參閱 `manypbind`。

完成設定後，按一下「*完成*」儲存設定，並回到 YaST 控制中心。

圖形 35.6 *設定 NIS 伺服器的領域和位址*



36 LDAP—一種目錄服務

「輕量型目錄存取協定」(Lightweight Directory Access Protocol，LDAP) 是通訊協定集合用以存取和維護資訊目錄。LDAP 可做為許多用途，像是使用者與群組管理、系統組態管理或是位址管理。本章針對 OpenLDAP 如何運作以及如何使用 YaST 來管理 LDAP 資料提供基本的概念。雖然 LDAP 協定有數種執行方式，但是本章只著重於 OpenLDAP 執行方式。

在網路環境中保持重要資訊的結構性和容易存取是相當重要的。要達到此一目的，可使用黃頁之類的目錄服務，它會以結構良好並且快速搜尋的形式來保存資訊以供存取。

在理想的狀況下，中央伺服器會將資料保存在目錄中，並使用特定的協定將資料分送給所有用戶端。資料的結構允許各種不同的應用程式進行存取。如此，個別的行事曆工具和電子郵件用戶端就不需要維護自己的資料庫，而是存取一個中央儲存區。此一特性明顯降低管理資訊的需要。使用 LDAP 等開放且標準化的協定，以盡量確保所有不同的用戶端應用程式都能存取這類的資訊。

此處所指的目錄是一種最佳化的資料庫，可供快速而有效的讀取和搜尋：

- 為了能夠同時大量讀取，寫入權限僅限管理員所執行的少量更新作業。舊式的資料庫會進行最佳化，以容許在最短時間內接受最多的資料量。
- 由於寫入的存取限制極大，因此採用目錄服務來管理大多數未變更的靜態資料。舊式資料庫中的資料經常變更 (*動態*資料)。舉例來說，企業目錄中的電話號碼變更頻率，會低於會計部門所管理數字的變更頻率。
- 管理靜態資料時，很少會更新現存的資料集。管理動態資料時，特別是銀行帳戶或會計相關的資料集時，最重要的是保持這些資料的一致性。如果要從

一筆資料扣除一個數目，然後將它加到另一筆資料，則必須在同一筆交易內同時進行這兩個動作，以確保整個資料集的餘額正確無誤。資料庫可支援此種交易。目錄則否。目錄可接受短暫性的資料不一致。

LDAP 這類的目錄服務並不是設計來支援複雜的更新或查詢。所有的應用程式在存取這種服務時，都應該以快速簡便的方式進行。

36.1 LDAP 與 NIS 的比較

Unix 系統管理員習慣使用 NIS 伺服器進行名稱解析和網路資料散佈。包含於 `/etc` 中的檔案和 `group`、`hosts`、`mail`、`netgroup`、`networks`、`passwd`、`printcap`、`protocols`、`rpc` 和 `services` 等目錄下的組態資料是由網路中的用戶端進行散發。因為這些檔案是簡單的文字檔，因此相當容易維護。不過，因為缺乏結構，較大量的資料處理變得更加困難。NIS 的設計只適用於 Unix 平台，這意味著它不適合做為異質性網路的中央資料管理工具。

和 NIS 不同，LDAP 伺服器不限於純 Unix 網路。Windows 伺服器 (2000 或更新) 支援 LDAP 的目錄服務功能。其他非 Unix 系統也可為前述應用程式任務提供更多支援。

LDAP 規則可套用於任何必須進行中央管理的資料結構。以下為幾個應用程式範例：

- 做為 NIS 服務的替代品使用
- 郵件路由 (postfix、sendmail)
- 郵件用戶端的通訊錄，例如 Mozilla、Evolution 和 Outlook。
- BIND9 名稱伺服器的區域描述管理
- 在異質網路中，利用 Samba 進行使用者驗證

這個清單還可加長，因為 LDAP 是可以延伸的，和 NIS 有所不同。由於清楚定義的資料階層結構可方便搜尋，使得管理大量資料的工作更為容易。

36.2 LDAP 目錄樹的結構

為了取得關於 LDAP 伺服器工作方式以及資料儲存方式的深層背景知識，請務必瞭解資料在伺服器上的組織方式，以及您如何透過此結構利用 LDAP 快速存取所需的資料。若要成功操作 LDAP 設定，還需要熟悉一些基本的 LDAP 術語。本章節介紹 LDAP 目錄樹的基本配置，並提供 LDAP 網路位置中使用的基本術語。如果您已具備一定的 LDAP 背景知識，現在只想知道如何在 SUSE Linux Enterprise 中設定 LDAP 環境，那麼請跳過此介紹章節。請分別參閱第 36.5 節「使用 YaST 設定 LDAP 伺服器」[619頁]或第 36.3 節「使用 slapd.conf 來設定伺服器」[610頁]。

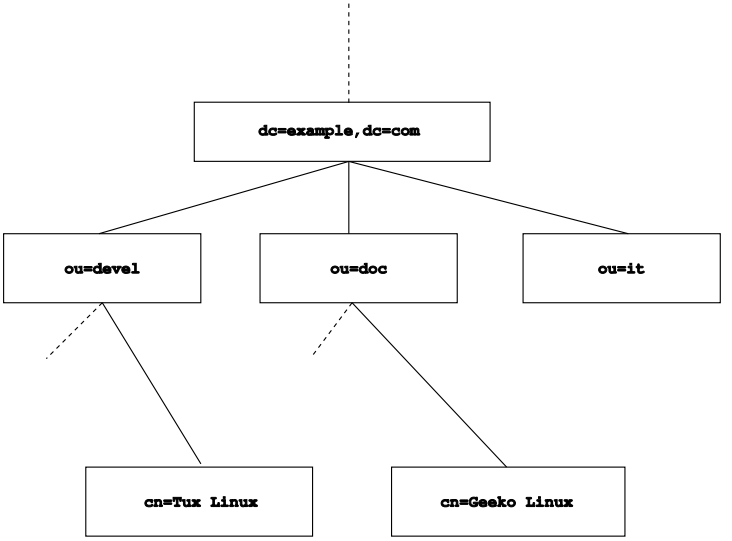
LDAP 的目錄具有樹狀結構。階層中的所有目錄項目 (稱為物件) 都有一個規定的位置。此階層稱為*目錄資訊樹*(DIT)。通往所需項目的完整路徑稱為*可辨識名稱*或 DN，可用來清楚辨識路徑。此項目路徑中的單一節點稱為*相關可辨識名稱*或 RDN。可將物件指定為下列兩種類型：

容器
: 這些物件可包含其他物件。這些物件的類別為 `root` (目錄樹的根元素，此元素其實並不存在)、`c` (國家)、`ou` (組織單位) 和 `dc` (領域元件)。此模型可媲美檔案系統中的目錄 (資料夾)。

分葉
: 這些物件位於分支末端，而且沒有附屬物件。例如 `person`、`InetOrgPerson` 或 `groupofNames`。

目錄階層的頂端有一個根元素 `root`。其中可包含 `c` (國家)、`dc` (領域元件) 或 `o` (組織) 作為其附屬元件。從下列範例 (顯示於圖形 36.1「LDAP 目錄結構」[608頁]) 可清楚看出 LDAP 目錄樹中的關係。

圖形 36.1　LDAP 目錄結構



這個完整的結構圖是由一棵虛擬的目錄資訊樹所組成。其中包含三個階層的項目。每個項目分別對應到圖上的一個方塊。在此範例中，虛擬的 員工 *Geeko Linux* 有一個完整有效的可辨識名稱，即 `cn=Geeko Linux,ou=doc,dc=example,dc=com`。此名稱的形成是將 RDN `cn=Geeko Linux` 加到前一個項目 `ou=doc,dc=example,dc=com` 的 DN。

物件類型應儲存於 DIT 且遵循*綱要*全域判斷。物件類型則是由*物件類別*來決定。物件類別決定相關物件必須或可以被指定哪一種屬性。因此，綱要的內容必須包括所有物件類別的定義，以及所需應用程式案例中使用的屬性。有幾個通用綱要 (請參閱 RFC 2252 和 2256)。不過，仍可建立自定的綱要。如果用來操作 LDAP 的伺服器環境需要，也可使用多個綱要來互相輔助。

表格 36.1「常用物件類別和屬性」[609頁]提供一個小型綜覽，介紹範例中所使用的 `core.schema` 和 `inetorgperson.schema` 物件類別，包括所需的屬性和有效的屬性值。

表格 36.1 *常用物件類別和屬性*

物件類別	代表意義	範例項目	必要屬性
dcObject	*domainComponent* (領域的名稱元件)	範例	dc
organizationalUnit	*organizationalUnit* (組織單位)	doc	ou
inetOrgPerson	*inetOrgPerson* (內部網路或網際網路的相關人員資料)	Geeko Linux	sn 與 cn

範例 36.1「摘自 schema.core 」[609頁]中摘述一段綱要指示詞及其說明 (將各行編號以便說明)。

範例 36.1 *摘自 schema.core*

```
#1 attributetype (2.5.4.11 NAME ( 'ou' 'organizationalUnitName')
#2        DESC 'RFC2256: organizational unit this object belongs to'
#3        SUP name )

...
#4 objectclass ( 2.5.6.5 NAME 'organizationalUnit'
#5        DESC 'RFC2256: an organizational unit'
#6        SUP top STRUCTURAL
#7        MUST ou
#8 MAY (userPassword $ searchGuide $ seeAlso $ businessCategory
    $ x121Address $ registeredAddress $ destinationIndicator
    $ preferredDeliveryMethod $ telexNumber
    $ teletexTerminalIdentifier $ telephoneNumber
    $ internationaliSDNNumber $ facsimileTelephoneNumber
    $ street $ postOfficeBox $ postalCode $ postalAddress
    $ physicalDeliveryOfficeName
    $ st $ l $ description) )
...
```

`organizationalUnitName` 屬性類型和對應的 `organizationalUnit` 物件類別在此作為範例之用。第 1 行的重點為屬性名稱、其唯一的 OID (*物件識別碼*) (數值) 和屬性的縮寫。

第 2 行提供屬性說明和 `DESC`。另外還包括對應的 RFC，可據此列出相關定義。第 3 行的 `SUP` 表示此屬性所隸屬的上級屬性類型。

第 4 行開始為 organizationalUnit 物件類型的定義，內容包括 OID 和物件類別名稱，如同屬性定義一樣。第 5 行為物件類別的簡短說明。第 6 行的 SUP top 項目表示此物件類別不隸屬於其他物件類別。第 7 行始於 MUST，列出所有必須和 organizationalUnit 類型物件配合使用的屬性類型。第 8 行始於 MAY，列出所有允許和此物件類別配合使用的屬性類型。

有關綱要的使用，可在 OpenLDAP 文件中找到一篇很好的介紹。安裝好之後，請至 /usr/share/doc/packages/openldap2/admin-guide/index.html 尋找這篇文章。

36.3 使用 slapd.conf 來設定伺服器

在您安裝好的系統上，/etc/openldap/slapd.conf 中有一個完整的組態檔，可供您的 LDAP 伺服器使用。此處將簡要介紹單一項目，並對必要的調整進行說明。開頭為井字符號(#)的項目為非作用中的項目。必須移除這個備註字元，才能啟用這些項目。

36.3.1 slapd.conf 中的全域指示詞

範例 36.2 *slapd.conf：包括綱要指示詞*

```
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/rfc2307bis.schema
include         /etc/openldap/schema/yast.schema
```

如範例 36.2「slapd.conf：包括綱要指示詞」[610頁]所顯示的 slapd.conf 中，第一個指示詞會指定供 LDAP 目錄做為組織依據的綱要。core.schema 項目為必需。其他必需的綱要都附加於此指示詞之後。請在包含的 OpenLDAP 文件中尋找更多資訊。

範例 36.3 *slapd.conf：pidfile 與 argsfile*

```
pidfile /var/run/slapd/slapd.pid
argsfile /var/run/slapd/slapd.args
```

這兩個檔案包含 PID (處理程序 ID) 和一些隨 slapd 程序啟動的引數。此處不需要修改。

範例 36.4 *slapd.conf：存取控制*

```
# Sample Access Control
#       Allow read access of root DSE
# Allow self write access
#       Allow authenticated users read access
#       Allow anonymous users to authenticate
# access to dn="" by * read
  access to * by self write
               by users read
               by anonymous auth
#
# if no access controls are present, the default is:
#       Allow read by all
#
# rootdn can always write!
```

範例 36.4「slapd.conf：存取控制」[611頁]摘自 slapd.conf，此檔案在伺服器上負責管制 LDAP 目錄的存取權。只要資料庫特定的區段中沒有宣告任何自定的規則，在 slapd.conf 全域區段中的設定均為有效。自定規則可覆寫全域宣告。正如此處所示，所有使用者都有讀取目錄的權利，但只有管理員 (rootdn) 才可在目錄中寫入資料。LDAP 中的存取控制管制是一個非常複雜的程序。下列秘訣可提供協助：

- 每一項存取規則都具有下列結構：

  ```
  access to <what> by <who> <access>
  ```

- *what* 是一個佔位符，代表有存取權的物件或屬性。個別的目錄分支可由個別規則來明確地保護。也可使用一般表示式來處理目錄樹的區域。slapd 會按照組態檔中所列的順序來評估所有規則。較一般性的規則列在較特定的規則之後 — slapd 視為有效的第一條規則會受到評估，其後的所有項目則被忽略。

- *who* 決定誰應獲得存取權，以進入由 *what* 所決定的區域。可使用一般表示式。第一次比對完成後 slapd 會再次中止 who 的評估，因此較特定的規必須列在較一般性的規則之前。在表格 36.2「使用者群組及其存取權」[612頁]中所顯示的項目是有可能的。

表格 36.2 *使用者群組及其存取權*

標籤	範圍
*	沒有例外的所有使用者
anonymous	未驗證的 (「匿名」) 使用者
users	驗證的使用者
self	以目標物件連線的使用者
dn.regex=<regex>	符合一般表示式的所有使用者

- *access* 會指定存取類型。使用表格 36.3「存取類型」[612頁]中所列的選項。

表格 36.3 *存取類型*

標籤	存取範圍
none	沒有存取權
auth	連絡伺服器
compare	比較物件的存取

標籤	存取範圍
search	搜尋過濾器的使用
read	讀取權
write	寫入權

slapd 會比較用戶端所要求的存取權與在 slapd.conf 中授予的存取權。如果用戶端所要求的權利比規則中的等級更低或同等，則可獲得存取權。如果用戶端要求的權利比規則中的宣告更高，則會被拒絕。

範例 36.5「slapd.conf：存取控制範例」[613頁]顯示使用一般表示式即可任意開發簡單存取控制權的範例。

範例 36.5 *slapd.conf：存取控制範例*

```
access to  dn.regex="ou=([^,]+),dc=example,dc=com"
by dn.regex="cn=Administrator,ou=$1,dc=example,dc=com" write
by user read
by * none
```

此規則宣告只有各個管理員有寫入個別 ou 項目的權利。其他所有通過驗證的使用者都有讀取權，其餘的人則沒有存取權。

提示：建立存取規則

如果沒有 access to 規則或者沒有 by 指示詞的符合項，則無法取得存取權。只會明確宣告的存取權可獲授予。如果沒有任何宣告的規則，則管理員依照預設原則擁有寫入權，其他人則有讀取權。

請至已安裝的openldap2套件的線上文件尋找詳細資訊及LDAP存取權的組態範例。

除了使用 (slapd.conf) 中央組態檔來管理存取權之外，也可使用存取控制資訊(ACI)。ACI 允許在 LDAP 樹中儲存個別物件的存取資訊。此種類型的存取控制尚未普及，開發者目前仍將它視為實驗性質。請參閱http://www.openldap.org/faq/data/cache/758.html以取得更多資訊。

36.3.2 slapd.conf 中的資料庫特定指示詞

範例 36.6　slapd.conf：資料庫特定指示詞

```
database bdb❶
suffix "dc=example,dc=com"❷
checkpoint      1024    5❸
cachesize       10000❹
rootdn "cn=Administrator,dc=example,dc=com"❺
# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw secret❻
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd/tools. Mode 700 recommended.
directory /var/lib/ldap❼
# Indices to maintain
index   objectClass     eq❽
overlay ppolicy❾
ppolicy_default "cn=Default Password Policy,dc=example,dc=com"
ppolicy_hash_cleartext
ppolicy_use_lockout
```

❶ 此區段的第一行設定的是資料庫類型 (在此案例中為 Berkeley 資料庫)，請參閱範例 36.6「slapd.conf：資料庫特定指示詞」[614頁]。

❷ `suffix` 決定此伺服器應負責的 LDAP 目錄樹部分。

❸ `checkpoint` 決定在寫入實際資料庫之前可保留在交易記錄中的資料數量 (以 KB 為單位) 以及兩次寫入動作之間的時間 (以分鐘為單位)。

❹ `cachesize` 設定可保留在資料庫快取記憶體中的物件數量。

❺ `rootdn` 決定可擁有此伺服器管理權的使用者。此處宣告的使用者不需要有 LDAP 項目或以一般使用者的身份出現。

❻ `rootpw` 設定管理員密碼。□在這裡可以不使用 `secret` 而是以雜湊值來輸入 `slappasswd` 所建立的管理員密碼。

❼ `directory` 指示詞表示資料庫目錄在檔案系統中所在的目錄，儲存在伺服器中。

❽ 最後一個指示詞 `index objectClass eq` 會啟動所有物件類別之索引的維護作業。在此處可根據經驗，加入一個使用者最常搜尋的屬性。

❾ `overlay ppolicy` 新增密碼控制機制的層級。在指定使用者的項目中未設定任何特定規則時，`ppolicy_default` 可指定要使用之 pwdPolicy 物

件的 DN。如果項目沒有特定規則，且未給定預設值，則不強制執行任何規則。ppolicy_hash_cleartext 指定在請求儲存到資料庫之前雜湊新增請求與修改請求的純文字密碼。使用此選項時，建議拒絕所有目錄使用者對 userPassword 的比較、搜尋及讀取存取，因為 ppolicy_hash_cleartext 違反 X.500/LDAP 資訊模型。ppolicy_use_lockout 會在用戶端嘗試連接鎖定的帳戶時，傳送特定的錯誤碼。如果您的網站對安全性問題比較敏感，請停用此選項，因為錯誤碼會給攻擊者提供有用資訊。

此處使用的是資料庫自定的 Access 規則，而非全域的 Access 規則。

36.3.3 啟動及停止伺服器

待 LDAP 伺服器已完全設定且所有需要的項目均已依照第 36.4 節「LDAP 目錄中的資料處理」[615頁]中描述的樣式建立之後，輸入 rcldap start 以 root 的身份來啟動 LDAP 伺服器。輸入指令 rcldap stop 可手動停止伺服器。使用 rcldap status 可請求執行中之 LDAP 的狀態。

第 20.2.3 節「使用 YaST 設定系統服務 (Runlevel)」[367頁]中所描述的 YaST runlevel 編輯器，可讓伺服器在系統啟動和暫停時自動啟動及停止。也可以透過第 20.2.2 節「Init 程序檔」[363頁]中所述的指令提示，利用 insserv 指令來建立對應的連結至啟動和停止程序檔。

36.4 LDAP 目錄中的資料處理

OpenLDAP 提供一系列的工具，用來管理 LDAP 目錄中的資料。以下說明四種最重要的工具，分別用來新增、刪除、搜尋和修改資料集。

36.4.1 將資料加入 LDAP 目錄

LDAP 伺服器 /etc/openldap/slapd.conf 中的組態正確並且就緒之後 (即擁有正確的 suffix、directory、rootdn、rootpw 和 index 項目)，即可開始輸入記錄。OpenLDAP 為此項任務提供了 ldapadd 指令。如果可能，請以套裝方式將多個物件一次加入資料庫，這是較實用的作法。LDAP 處理 LDIF 格式 (LDAP 資料互換格式) 的能力可在此時發揮作用。LDIF 是一個簡單的文字

檔，可包含任意數量的屬性與值的組合。如需可用物件類別和屬性的相關資訊，請參閱 `slapd.conf` 中宣告的綱要檔。在 圖形 36.1「LDAP 目錄結構」[608頁] 中用來建立概略架構的 LDIF 檔會和 範例 36.7「LDIF 檔案範例」[616頁] 中的檔案相似。

範例 36.7 LDIF 檔案範例

```
# The Organization
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
o: Example dc: example

# The organizational unit development (devel)
dn: ou=devel,dc=example,dc=com
objectClass: organizationalUnit
ou: devel

# The organizational unit documentation (doc)
dn: ou=doc,dc=example,dc=com
objectClass: organizationalUnit
ou: doc

# The organizational unit internal IT (it)
dn: ou=it,dc=example,dc=com
objectClass: organizationalUnit
ou: it
```

重要：LDIF 檔的編碼

LDAP 可使用 UTF-8 (Unicode)。母音字元的編碼必須正確。請使用支援 UTF-8 的編輯器，例如 Kate 或最新版的 Emacs。否則，應避免母音字元和其他特殊字元，或使用 `recode` 將輸入重新編碼為 UTF-8。

儲存字尾為 `.ldif` 的檔案，然後用下列指令傳送給伺服器。

```
ldapadd -x -D <dn of the administrator> -W -f <file>.ldif
```

`-x` 在此例中會關閉 SASL 驗證。`-D` 會宣告呼叫此作業的使用者。在此處輸入管理員的有效 DN，如同 `slapd.conf` 中的設定一樣。在此範例中，有效 DN 為 `cn=Administrator,dc=example,dc=com`。`-W` 可避免將密碼輸入指令行 (以純文字) 並啟用單獨的密碼提示。此密碼是先前在 `slapd.conf` 中使用 `rootpw` 決定的。`-f` 會傳送檔案名稱。請參閱 範例 36.8「ldapadd 和 example.ldif」[617頁] 中執行 `ldapadd` 的詳細內容。

範例 36.8 *ldapadd 和 example.ldif*

```
ldapadd -x -D cn=Administrator,dc=example,dc=com -W -f example.ldif

Enter LDAP password:
adding new entry "dc=example,dc=com"
adding new entry "ou=devel,dc=example,dc=com"
adding new entry "ou=doc,dc=example,dc=com"
adding new entry "ou=it,dc=example,dc=com"
```

個別的使用者資料可建立在不同的 LDIF 檔。範例 36.9「Tux 的LDIF 資料」[617頁] 會新增 `Tux` 至新的 LDAP 目錄。

範例 36.9 *Tux 的LDIF 資料*

```
# coworker Tux
dn: cn=Tux Linux,ou=devel,dc=example,dc=com
objectClass: inetOrgPerson
cn: Tux Linux
givenName: Tux
sn: Linux
mail: tux@example.com
uid: tux
telephoneNumber: +49 1234 567-8
```

LDIF檔可包含任意數量的物件。既可以將整個目錄分支一次性傳給伺服器，也可只傳送一部分，如個別物件範例所示。如果需要經常修改某些資料，建議使用單一物件的細分。

36.4.2 修改 LDAP 目錄中的資料

工具`ldapmodify`可用來修改資料集。最簡單的修改方法是先修改對應的LDIF檔，然後將修改過的檔案傳送給 LDAP 伺服器。如果要將同事 Tux 的電話號碼從 `+49 1234 567-8` 改為 `+49 1234 567-10`，則必須依照 範例 36.10「修改過的 LDIF 檔 tux.ldif」[617頁] 中的方式來編輯 LDIF 檔案。

範例 36.10 *修改過的 LDIF 檔 tux.ldif*

```
# coworker Tux
dn: cn=Tux Linux,ou=devel,dc=example,dc=com
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

使用下列指令將修改過的檔案輸入 LDAP 目錄：

```
ldapmodify -x -D cn=Administrator,dc=example,dc=com -W -f tux.ldif
```

或者，也可以直接將要修改的屬性傳送給 ldapmodify。此項作業程序描述如下：

1 啟動 ldapmodify 並輸入您的密碼：

```
ldapmodify -x -D cn=Administrator,dc=example,dc=com -W
Enter LDAP password:
```

2 輸入變更並注意是否符合下列語法順序：

```
dn: cn=Tux Linux,ou=devel,dc=example,dc=com
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

請在 ldapmodify 線上文件中尋找有關 ldapmodify 及其語法的詳細資訊。

36.4.3 搜尋或讀取 LDAP 目錄中的資料

OpenLDAP 提供的指令行工具 ldapsearch，可用來搜尋和讀取 LDAP 目錄中的資料。下列為簡易查詢的語法：

```
ldapsearch -x -b dc=example,dc=com "(objectClass=*)"
```

-b 選項決定搜尋基礎 — 用來執行搜尋的目錄樹區段。在目前這個範例中，搜尋基礎為 dc=example,dc=com。如果要在 LDAP 目錄的次區段中執行更詳細的搜尋 (例如，只在 devel 部門搜尋)，可使用 -b 將此區段傳送至 ldapsearch。-x 會要求啟用簡單的驗證。(objectClass=*) 宣告應讀取目錄中的所有物件。在建立一個新的目錄樹後，可使用本指令來確認所有項目都已正確記錄，而且伺服器的回覆也符合需要。請在對應的線上文件 (ldapsearch(1)) 中尋找更多關於使用 ldapsearch 的詳細資訊。

36.4.4 刪除 LDAP 目錄中的資料

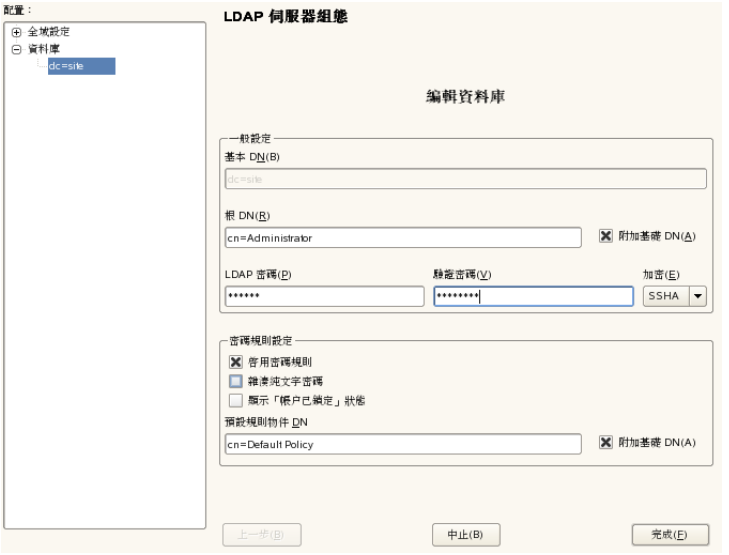
使用 `ldapdelete` 可刪除不需要的資料。其語法和其他指令語法相似。例如，要刪除 `Tux Linux` 的整個項目：可發出下列指令：

```
ldapdelete -x -D cn=Administrator,dc=example,dc=com -W cn=Tux \
Linux,ou=devel,dc=example,dc=com
```

36.5 使用 YaST 設定 LDAP 伺服器

使用 YaST 可設定 LDAP 伺服器。LDAP 伺服器的一般使用案例包括使用者帳號資料與郵件、DNS 和 DHCP 伺服器組態的管理。

圖形 36.2　YaST LDAP 伺服器組態



若要為使用者帳戶資料設定 LDAP 伺服器，請如下執行：

1. 以 `root` 身份登入。
2. 啟動 YaST 並選取「*網路服務*」>「*LDAP 伺服器*」。

3 將 LDAP 設為系統開機時啟動。

4 若 LDAP 伺服器必須透過 SLP 宣告服務，請勾選「*在 SLP 精靈註冊*」。

5 選擇「*設定*」以設定「*一般設定*」和「*資料庫*」。

若要設定您 LDAP 伺服器的「*全域設定*」，請如下執行：

1 選取對話方塊左邊的「*綱要檔案*」，接受或修改伺服器組態中所包含的綱要檔案。綱要檔案的預設選擇會套用到提供 YaST 使用者帳戶資料來源的伺服器。

2 透過「*記錄層級設定*」，設定 LDAP 伺服器記錄活動的程度 (詳細的程度)。依照您的需要，從預先定義清單中選取或取消選取記錄選項。啟用的選項愈多，您的記錄檔就愈大。

3 決定 LDAP 伺服器允許的連線類型。從下列選擇：

bind_v2
: 此選項可利用前一個版本的協定 (LDAPv2) 來啟動用戶端的連線要求 (繫結要求)。

bind_anon_cred
: LDAP 伺服器通常會拒絕缺乏憑證 (DN 或密碼) 的驗證嘗試。不過，啟用這個選項可以透過密碼而非 DN 來建立匿名連線。

bind_anon_dn
: 啟用這個選項，可使用 DN 而非密碼，以非驗證 (匿名) 的方式進行連線。

update_anon
: 啟用這個選項可允許進行未驗證 (匿名) 的更新操作。在 ACL 和其它規則下的存取權限制相當嚴格 (請參閱 第 36.3.1 節「slapd.conf 中的全域指示詞」[610頁])。

4 若要設定用戶端與伺服器間的安全通訊，請以「*TLS 設定*」執行：

4a 將「*TLS 可用*」設定為「*是*」，以啟用用戶端/伺服器通訊的 TLS 和 SSL 加密。

4b 按一下「*選取憑證*」並決定如何獲得有效憑證。選擇「*輸入憑證*」(從外部來源輸入憑證) 或「*使用公用伺服器憑證*」(使用安裝時建立的憑證)。

- 若您選擇輸入憑證，YaST 會提示您指定其位置的正確路徑。
- 若您選擇使用一般伺服器證書，但在安裝時未建立的話，則會接著建立。

若要設定您 LDAP 伺服器管理的資料庫，請如下執行：

1 在對話左邊選擇「*資料庫*」項目。

2 按一下「*新增資料庫*」以新增資料庫。

3 輸入所需資料：

「*基本 DN*」
輸入您 LDAP 伺服器的基本 DN。

「*根 DN*」
輸入負責管理伺服器的管理員 DN。如果您要檢查「*附加基礎DN*」，則只要提供管理員的 `cn` 即可，系統會自動填入資料。

LDAP 密碼
輸入資料庫管理員的密碼。

加密
決定要用來保護根 DN 密碼的加密運算法。選擇「*crypt*」、「*smd5*」、「*ssha*」或「*sha*」。對話方塊中還有「*純文字*」選項，可啟用純文字密碼，但是出於安全性的考量不建議使用。若要確認您的設定，並返回先前的對話方塊，請選取「*確定*」。

4 強制執行密碼規則，加強 LDAP 伺服器的安全性：

4a 選取「*密碼規則設定*」，以便指定密碼規則。

4b 啟用「*雜湊純文字密碼*」，可在新增或修改純文字密碼時，先對其進行雜湊，再將其寫入資料庫。

4c 「*顯示帳戶已鎖定狀態*」可提供有意義的錯誤訊息，幫助將要求繫結至鎖定的帳戶。

警告：安全性敏感環境中的已鎖定帳戶

如果環境對安全性問題比較敏感，請不要使用「*顯示帳戶已鎖定狀態*」選項，因為「帳戶已鎖定」錯誤訊息提供的安全性敏感資訊可能會被潛在攻擊者利用。

4d 輸入預設規則物件的 DN。若要使用非 YaST 建議的 DN，請輸入您的選擇。否則，請接受預設設定。

5 按一下「*完成*」，完成資料庫組態。

如果未選擇使用密碼規則，此時伺服器即可開始執行。如果您選擇啟用密碼規則，則繼續詳細設定密碼規則。如果您選擇的密碼規則物件不存在，YaST 會建立：

1 輸入 LDAP 伺服器密碼。

2 設定密碼變更規則：

2a 決定儲存在密碼歷程中的密碼數量。使用者也許無法重新使用儲存的密碼。

2b 決定使用者是否可以變更他們的密碼，管理員重設密碼之後使用者是否需要變更密碼。選擇性地要求變更舊密碼。

2c 決定密碼是否需要接受質量檢查，以及檢查的程度。設定最小密碼長度，符合此長度的密碼才是有效密碼。如果您選取「*接受不可減檢查的密碼*」，將允許使用者使用加密的密碼，即使無法執行質量檢查。如果您選擇「*僅接受檢查過的密碼*」，則只有通過質量測試的密碼才會生效。

3 設定密碼期限規則：

3a 決定密碼最短壽命 (有效密碼變更之間的時間間隔) 與密碼最長壽命。

3b 決定密碼過期警告與實際密碼過期的時間間隔。

3c 設定在密碼完全過期之前，允許使用過期密碼的次數。

4 設定鎖定規則：

4a 啟用密碼鎖定。

4b 決定繫結失敗幾次後觸發密碼鎖定。

4c 決定密碼鎖定的持續時間。

4d 決定密碼失敗保留在快取記憶體中的時間 (過期即清除)。

5 使用「*接受*」套用密碼規則設定。

如果要編輯先前建立的資料庫，請由目錄樹左方選取其基礎 DN。YaST 會在視窗右邊顯示與資料庫新建對話方塊相似的對話方塊 — 主要差異在於 DN 項目呈灰色而無法變更。

選擇「*完成*」離開 LDAP 伺服器設定之後，即表示已為 LDAP 伺服器建立一個基本工作組態。如果要微調此設定，可接著編輯 `/etc/openldap/slapd.conf` 檔案，然後重新啟動伺服器。

36.6 使用 YaST 設定 LDAP 用戶端

YaST 中包含一個模組，可用來設定 LDAP 式的使用者管理。如果您在安裝時未啟用此功能，則可選取「*網路服務*」>「*LDAP 用戶端*」來啟用模組。YaST 會自動啟用 LDAP 所需的任何 PAM 和 NSS 相關變更，並安裝必要檔案。

36.6.1 標準程序

具備作用於用戶端機器背景的處理程序基本知識，可協助您瞭解 YaST LDAP 用戶端模組的運作方式。如果已啟用 LDAP 以進行網路驗證或者已呼叫 YaST 模組，則表示已安裝 `pam_ldap` 和 `nss_ldap` 套件，而且兩個對應的組態檔也已經調整。`pam_ldap` 為 PAM 模組，負責登入程序和 LDAP 目錄之間的協調，

也是驗證資料的來源。已安裝 pam_ldap.so 專用模組並調整 PAM 組態 (請參閱 範例 36.11「適用於 LDAP 的 pam_unix2.conf」[624頁])。

範例 36.11 *適用於 LDAP 的 pam_unix2.conf*

```
auth:       use_ldap
account:    use_ldap
password:   use_ldap
session:    none
```

在手動設定額外服務以使用 LDAP 時，應納入 PAM組態檔中的 PAM LDAP 模組，該組態檔與 /etc/pam.d 中的服務相對應。可在 /usr/share/doc/packages/pam_ldap/pam.d/ 中找到適用於個別服務的組態檔。將適當的檔案複製到 /etc/pam.d。

透過 nsswitch 機制進行的 glibc 名稱解析適用於含有 nss_ldap 的 LDAP。安裝此套件時，已在 /etc/ 中建立一個新的調整過的 nsswitch.conf 檔案。如需 nsswitch.conf 運作的更多資訊，請參閱第 30.7.1 節「組態檔案」[534頁]。nsswitch.conf 必須包含下列幾行，以供 LDAP 進行使用者管理及驗證。請參閱範例 36.12「nsswitch.conf 中的調整」[624頁]。

範例 36.12 *nsswitch.conf 中的調整*

```
passwd: compat
group: compat

passwd_compat: ldap
group_compat: ldap
```

這幾行會先命令 glibc 的查詢程式庫評估 /etc 中的對應檔案，然後存取 LDAP 伺服器，以它作為驗證及使用者資料的來源。可對此機制進行測試，例如使用 getent passwd 指令來讀取使用者資料庫的內容。所傳回的資料集應包含一份有關您系統上的本地使用者和 LDAP 伺服器儲存的所有使用者的調查。

如果要避免一般使用者透過 LDAP 以 ssh 或 login 登入伺服器，則 /etc/passwd 和 /etc/group 檔案必須各自增加一行。也就是 /etc/passwd 加上 +::::::/sbin/nologin 行，/etc/group 加上 +::: 行。

36.6.2 設定 LDAP 用戶端

在 YaST 完成 `nss_ldap`、`pam_ldap`、`/etc/passwd` 和 `/etc/group` 的初始調整之後，您只需將用戶端連接到伺服器，讓 YaST 管理 LDAP 上的使用者即可。章節「基本組態」[625頁]中有說明這個基本設定。

使用 YaST LDAP 用戶端進一步設定 YaST 群組和使用者組態模組。這包括操作新使用者和群組的預設設定，以及指定給使用者或群組的屬性數目和性質。與傳統的使用者或群組管理解決方案相比，LDAP 使用者管理可讓您將更多不同屬性指定給使用者和群組。如需詳細資訊，請參閱章節「設定 YaST 群組和使用者管理模組」[628頁]。

基本組態

如果您選擇 LDAP 使用者管理，或是在已安裝系統的「YaST 控制中心」選取「*網路服務*」>「*LDAP 用戶端*」，就會在安裝期間開啟基本 LDAP 用戶端組態對話方塊 (圖形 36.3「YaST: LDAP 用戶端的組態」[625頁])。

圖形 36.3 *YaST: LDAP 用戶端的組態*

若要對 OpenLDAP 伺服器驗證機器使用者，並透過 OpenLDAP 啟用使用者管理，請按照下列步驟進行：

1 按一下「*使用 LDAP*」以使用 LDAP。如果您要使用 LDAP 進行驗證，但是不想讓其他使用者登入這個用戶端，請改為選取「*使用 LDAP 但停用登入*」。

2 輸入要使用的 LDAP 伺服器的 IP 位址。

3 輸入「*LDAP 基礎 DN*」以選取 LDAP 伺服器上的搜尋基礎。如果要自動取得基礎 DN，請按一下「*取得 DN*」。然後，YaST 就會在以上指定的伺服器位址檢查任何 LDAP 資料庫。請從 YaST 提供的搜尋結果中選擇適當的基礎 DN。

4 如果需要伺服器的 TLS 或 SSL 受保護通訊，請選取「*LDAP TLS/SSL*」。

5 如果 LDAP 伺服器仍然使用 LDAPv2，請選取「*LDAP 版本 2*」以明確地使用此協定。

6 選取「*啟動自動裝載器*」以將遠端目錄 (例如遠端管理的 `/home`) 裝載至用戶端。

7 選取「*登入時建立主目錄*」，在使用者第一次登入時自動建立主目錄。

8 按一下「*完成*」以套用設定。

圖形 36.4 *YaST:進階組態*

如果要以管理者身份修改伺服器上的資料，按一下「*進階組態*」。下列對話方塊分為兩個索引標籤。請參閱圖形 36.4「YaST:進階組態」[626頁]。

1 在「*用戶端設定*」索引標籤中，依照您的需要來調整下列設定：

1a 如果使用者、密碼和群組的搜尋基礎與「*LDAP 基礎 DN*」指定的全域搜尋基礎不同，請在「*使用者對應*」、「*密碼對應*」和「*群組對應*」中，輸入這些不同的命名內容。

1b 指定密碼變更協定。變更密碼時使用的標準方法是 `crypt`，表示會使用由 `crypt` 產生的密碼雜湊。如需這個選項和其他選項的詳細資料，請參閱 `pam_ldap` man 頁面。

1c 指定要與「*群組成員屬性*」搭配使用的 LDAP 群組。它的預設值是 `member`。

2 在「*管理設定*」中，調整下列設定：

2a 透過「*組態基礎 DN*」來設定儲存使用者管理資料的基礎。

2b 在「*管理員 DN*」輸入適當的值。這個 DN 必須與 `/etc/openldap/slapd.conf` 所指定的 `rootdn` 值相同，才能讓這位特定使用者操作 LDAP 伺服器中儲存的資料。輸入完整 DN (例如 `cn=Administrator,dc=example,dc=com`) 或啟用「*附加基礎 DN*」以便在您輸入 `cn=Administrator` 後自動加上基礎 DN。

2c 勾選「*建立預設組態物件*」，在伺服器上建立基本組態物件，以透過 LDAP 啟用使用者管理。

2d 如果用戶端機器會成為網路中的主目錄的檔案伺服器，請勾選「*此機器上的主目錄*」。

2e 使用「*密碼規則*」部份，選取、新增、刪除或修改使用的密碼規則設定。YaST 的密碼規則組態是 LDAP 伺服器設定的一部分。

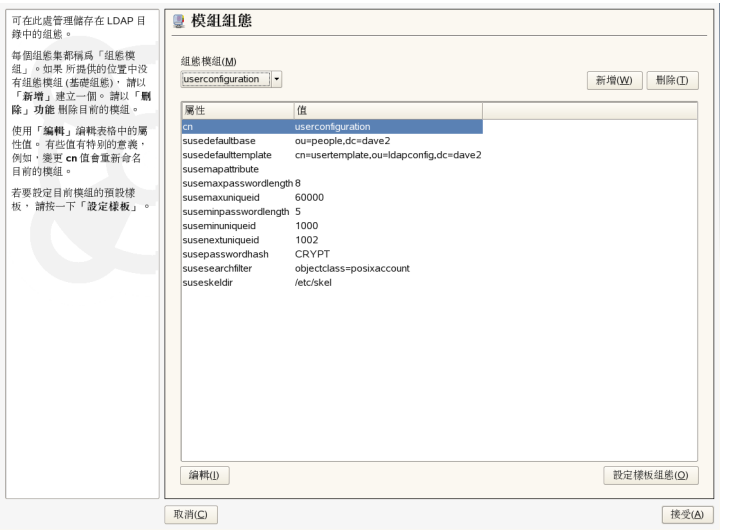
2f 按一下「*接受*」以離開「*進階組態*」，再按「*完成*」以套用設定。

按一下「*設定使用者管理設定*」以編輯 LDAP 伺服器上的項目。接著會根據伺服器中儲存的 ACL 和 ACI 來授予伺服器組態模組的權限。遵循 章節「設定 YaST 群組和使用者管理模組」[628頁] 中概述的程序。

設定 YaST 群組和使用者管理模組

使用 YaST LDAP 用戶端來調整 YaST 模組，以進行使用者和群組管理，並在需要時加以延伸。利用個別屬性的預設值來定義範本，以簡化資料註冊。此處所建立的預設會以 LDAP 物件的形式儲存在 LDAP 目錄中。使用者資料註冊仍以使用者和群組管理的一般 YaST 模組來完成。所註冊的資料會在伺服器中儲存為 LDAP 物件。

圖形 36.5 *YaST: 模組組態*



模組組態對話方塊 (圖形 36.5「YaST: 模組組態」[628頁]) 允許建立新模組、選取和修改現有的模組組態，以及設計和修改這些模組的範本。

若要建立新的組態模組，請按照下列步驟進行：

1 按一下「*新增*」並選取要建立的模組類型。若為使用者組態模組，請選取 `suseuserconfiguration`；若為群組組態，請選擇 `susegroupconfiguration`。

2 選擇新範本的名稱。接著該內容會顯示一個表格，列出所有可用於此模組的屬性和其指定值。除了所有設定的屬性之外，表格中也會列出現用綱要允許但未使用的其他屬性。

3 選取個別屬性，按「*編輯*」，然後輸入新值，即可接受或調整要用於群組和使用者組態的預設值。變更模組的 `cn` 屬性，即可重新命名模組。按一下「*刪除*」可刪除目前選取的模組。

4 按一下「*接受*」之後，新模組就會加入選項功能表。

群組和使用者管理的 YaST 模組會將合理的標準值內嵌至範本中。若要編輯與組態模組相關聯的範本，請按照下列步驟進行：

1 在「*模組組態*」對話方塊中，按一下「*設定範本*」。

2 根據您的需求來決定要指定給此範本的一般屬性值，或是保留空白。LDAP 伺服器上的空白屬性會被刪除。

3 修改、刪除或新增新物件的預設值 (LDAP 樹中的使用者或群組組態)。

圖形 36.6 *YaST：物件範本組態*

將模組的 `susedefaulttemplate` 屬性值設定為調整過的範本 DN，以連接範本與其模組。

提示

使用變數來取代絕對值，即可透過其他屬性建立一個屬性的預設值。例如，建立新使用者時，會自動以 `sn` 和 `givenName` 的屬性值建立 `cn=%sn %givenName`。

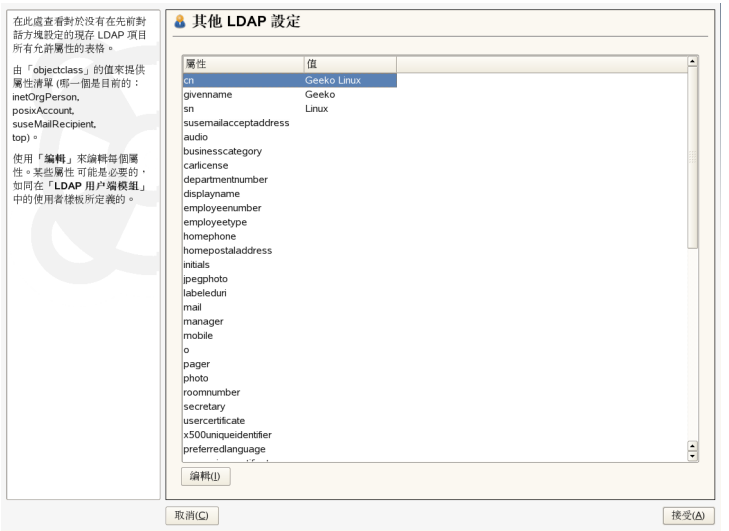
一旦所有模組和範本都已正確設定並可執行時，可使用 YaST 以一般方式來註冊新群組和使用者。

36.7 在 YaST 中設定 LDAP 使用者和群組

實際的使用者和群組資料註冊與不使用 LDAP 時的程序只有些微不同。以下是使用者管理簡介。群組管理程序與此類似。

1. 經由「*安全性與使用者*」>「*使用者管理*」進入 YaST 使用者管理。
2. 使用「*設定過濾器*」，將使用者的檢視畫面限制為 LDAP 使用者，並輸入根 DN 的密碼。
3. 按一下「*新增*」並輸入新使用者的組態。就會開啟有四個索引標籤的對話方塊：
   - **3a** 使用者資料「*索引標籤中的指定使用者名稱、登入及密碼。*」
   - **3b** 勾選新使用者的群組成員、登入外圍程序及主目錄的「*詳細資料*」索引標籤。若有需要，可將預設值變更為更符合您需求的值。利用章節「設定 YaST 群組和使用者管理模組」[628頁]中說明的程序，即可定義預設值以及密碼設定的預設值。
   - **3c** 修改或接受預設的「*密碼設定*」。
   - **3d** 進入「*外掛程式*」索引標籤，選取 LDAP 外掛程式，然後按一下「*啟動*」，即可設定指定給新使用者的其他 LDAP 屬性 (請參閱圖形 36.7「YaST：其他 LDAP 設定」[631頁])。
4. 按一下「*接受*」以套用設定，並離開使用者組態。

圖形 36.7　YaST：其他 LDAP 設定



最初的使用者管理輸入表中包含「*LDAP 選項*」。此功能可用來將 LDAP 搜尋過濾器套用到可用的使用者，或者也可選取「*LDAP 使用者及群組組態*」來連接設定 LDAP 使用者和群組的模組。

36.8 瀏覽 LDAP 目錄網路樹

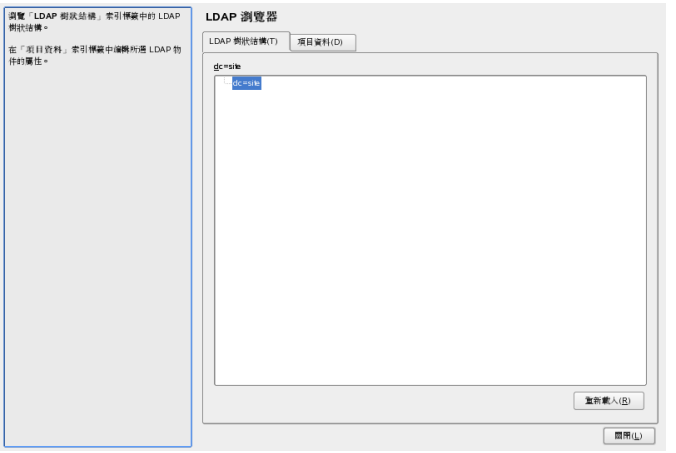
若要輕鬆瀏覽 LDAP 目錄網路樹，以及其所有項目，請使用 YaST LDAP 瀏覽器：

1. 以 `root` 身份登入。
2. 啟動「*YaST*」>「*網路服務*」>「*LDAP 瀏覽器*」。
3. 若您需要讀寫此伺服器上儲存的資料，請輸入 LDAP 伺服器的位址、AdministratorDN，以及此伺服器 RootDN 的密碼。

   否則請選擇「*匿名存取*」，不提供密碼取得目錄存取權。

   「*LDAP 網路樹*」索引標籤會顯示您所連接 LDAP 目錄的內容。按一下項目展開子項目。

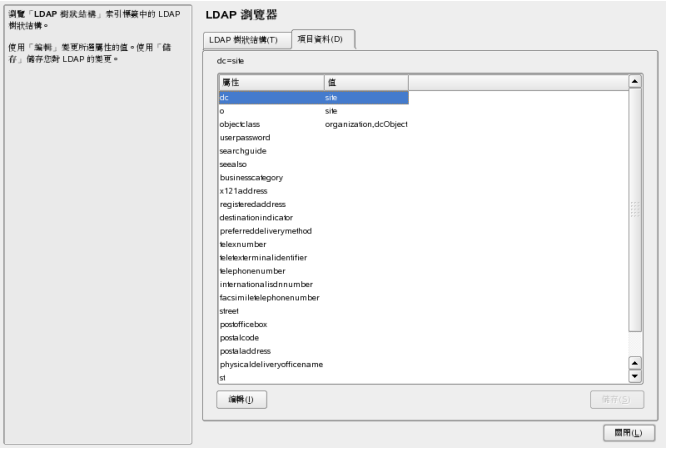
圖形 36.8　瀏覽 LDAP 目錄網路樹



4 若要檢視所有項目詳細資料，請選取「*LDAP 網路樹*」檢視，並開啟「*項目資料*」索引標籤。

會顯示此項目的所有屬性與相關數值。

圖形 36.9　瀏覽項目資料



5 若要變更這些屬性值，請按一下「*編輯*」，輸入新值，再按一下「*儲存*」，並在提示時提供 RootDN 密碼。

6 按一下「*關閉*」離開 LDAP 瀏覽器。

36.9 如需更多資訊

本章節刻意將一些較為複雜的主題排除在外，例如 SASL 組態或分散工作量至多個從屬伺服器的 LDAP 伺服器的複製。有關這兩個主題的詳細資訊，可在 *OpenLDAP 2.2 管理員指南*中找到。

OpenLDAP 專案的網站提供 LDAP 初學者及進階使用者詳盡的說明。

OpenLDAP Faq-O-Matic
: 是一個豐富的問答集，內容包括安裝、組態設定及 OpenLDAP 的使用方法。請參閱http://www.openldap.org/faq/data/cache/1.html。

快速入門指南
: 簡潔的逐步示範，教您如何安裝您的第一部 LDAP 伺服器。請參閱http://www.openldap.org/doc/admin22/quickstart.html，或是參閱已安裝系統的 /usr/share/doc/packages/openldap2/admin-guide/quickstart.html。

OpenLDAP 2.2 管理員指南
: 詳細介紹所有重要的 LDAP 組態，包括存取控制及加密。請參閱http://www.openldap.org/doc/admin22/，或是參閱已安裝系統的 /usr/share/doc/packages/openldap2/admin-guide/quickstart.html。

瞭解 LDAP
: 有關 LDAP 基本原則的一般性介紹：http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf。

有關 LDAP 的出版品：

- *LDAP System Administration (LDAP 系統管理)*，作者 Gerald Carter (ISBN 1-56592-491-6)

- *Understanding and Deploying LDAP Directory Services (瞭解和部署 LDAP 目錄服務)*，作者 Howes, Smith, and Good (ISBN 0-672-32316-8)

有關LDAP 的最終參考資料為對應的RFC (要求建議)，2251 至 2256。

37 Samba

使用 Samba，就可以將 Unix 機器設為 DOS、Windows 以及 OS/2 機器的檔案與列印伺服器。Samba 已經是一個開發至完全成熟且相當複雜的產品。請使用 YaST、SWAT (一種網頁介面) 或組態檔設定 Samba。

37.1 術語

下列為 Samba 文件和 YaST 模組中常用的詞彙。

SMB 通訊協定
: Samba 使用基於 NetBIOS 服務的 SMB (伺服器訊息區塊) 通訊協定。由於來自 IBM、Microsoft 發行了此通訊協定的壓力，因此其他的軟體製造商可以建立對 Microsoft 領域網路的連接。使用 Samba，SMB 通訊協定就可以在 TCP/IP 通訊協定上運作，因此 TCP/IP 通訊協定必須安裝在所有的用戶端上。

提示：IBM System z：NetBIOS 支援

IBM System z 只支援在 TCP/IP 上的 SMB。在這些系統上不提供 NetBIOS 支援。

CIFS 通訊協定
: CIFS (一般網際網路檔案系統) 通訊協定是 Samba 所支援的另一種通訊協定。CIFS 定義用於網路上的標準遠端檔案系統存取通訊協定，讓使用者群組可以透過網路分工合作和共享文件。

NetBIOS
: NetBIOS 是用來進行機器之間通訊的軟體介面 (API)。在此提供了名稱服務。它允許連接至網路的機器保留自己的名稱。在保留後，就可以使用名稱來定址這些機器。在此沒有檢查名稱的中央程序。在網路上的任何機器都可以保留它所需的任何數量名稱，只要這些名稱尚未使用。現在可以針對不同的網路結構實行 NetBIOS 介面。有一個與網路硬體一起緊密運作的執行程序，稱為 NetBEUI，不過這通常稱為 NetBIOS。與 NetBIOS 一起執行的網路通訊協定是 Novell 的 IPX (經由 TCP/IP 的 NetBIOS) 與 TCP/IP。

  經由 TCP/IP 所傳送的 NetBIOS 名稱，與 `/etc/hosts` 中所使用的名稱，或由 DNS 所定義的名稱完全不相同。NetBIOS 使用自己完全獨立的命名慣例。不過一般建議使用與 DNS 主機名稱相對應的名稱，如此可使管理較為輕鬆。Samba 預設是使用此對應名稱。

Samba 伺服器
: Samba 伺服器是為用戶端提供 SMB/CIFS 服務和 NetBIOS over IP 命名服務的伺服器。在 Linux 上，Samba 伺服器可使用兩種精靈：smnd 用於 SMB/CIFS 服務，而 nmbd 用於命名服務。

Samba 用戶端
: Samba 用戶端是透過 SMB 通訊協定使用 Samba 伺服器所提供之 Samba 服務的系統。所有一般的作業系統 (例如 Mac OS X、Windows 以及 OS/2) 都支援 SMB 通訊協定。TCP/IP 通訊協定必須安裝在所有的電腦上。Samba 提供適用於不同 UNIX 類別的用戶端。就 Linux 而言，有一個 SMB 的核心模組，允許在 Linux 系統層級上整合 SMB 資源。您不必為 Samba 用戶端執行任何精靈。

共用
: SMB 伺服器透過共用提供用戶端硬碟空間。共用是指印表機和位在伺服器上的目錄及其子目錄。它是利用名稱來輸出，並且可藉由其名稱來存取。共享名稱可以設成任何名稱—它並不需要是輸出目錄的名稱。也會指定一個名稱給印表機。用戶端可以透過其名稱存取印表機。

37.2 啟動和停止 Samba

您可以手動或在開機時自動啟動或停止 Samba 伺服器。啟動和停止原則是 YaST Samba 伺服器組態的一部分 (如第 37.3.1 節「使用 YaST 設定 Samba 伺服器」[637頁]所述)。

若要用 YaST 啟動或停止 Samba 服務，請使用「*系統*」>「*系統服務 (Runlevel)*」。在指令行中，使用 `rcsmb stop && rcnmb stop` 可停止 Samba 所需的服務，使用 `rcnmb start && rcsmb start` 則可啟動服務。

37.3 設定 Samba 伺服器

SUSE Linux Enterprise® 中的 samba 伺服器可以兩種方式設定：以 YaST 設定或手動設定。手動設定組態可以提供較詳細的設定，但是缺乏 YaST GUI 提供的方便性。

37.3.1 使用 YaST 設定 Samba 伺服器

若要設定 Samba 伺服器，請啟動 YaST 並選取「*網路服務*」>「*Samba 伺服器*」。第一次啟動模組時，會啟動「*Samba 伺服器安裝*」對話方塊，提示您決定有關伺服器管理的一些基本設定，然後在組態設定結束時，會出現提示要您輸入 Samba root 的密碼。之後啟動時，會顯示「*Samba 伺服器組態*」對話方塊。

Samba 伺服器安裝「 *對話方塊由兩個步驟組成：* 」

工作群組或網域名稱
: 在「*工作群組或網域名稱*」中選取現有的名稱，或輸入新的名稱，並按一下「*下一步*」。

Samba 伺服器類型
: 在下一個步驟中，指定您的伺服器是否應做為 PDC，並按一下「*下一步*」。

之後，您可以用「*Samba 伺服器組態*」對話方塊的「*識別*」索引標籤變更「*Samba 伺服器安裝*」中的所有設定。

透過 YaST 進行進階 Samba 設定

第一次啟動 Samba 伺服器模組時，「*Samba 伺服器組態*」對話方塊會緊跟在「*Samba 伺服器安裝*」對話方塊之後顯示。用此調整您的 Samba 伺服器組態。

編輯組態之後，按一下「*結束*」關閉組態。

啟動伺服器

在「*啟動*」索引標籤中，設定 Samba 伺服器的啟動。若每次系統開機時都要啟動服務，請選取「*開機時*」。若要啟用手動啟動，請選擇「*手動*」。如需有關啟動 Samba 伺服器的詳細資訊，請參閱第 37.2 節「啟動和停止 Samba」[636 頁]。

在此索引標籤中，您也可以開啟您的防火牆中的連接埠。若要執行此動作，請選取「*開啟防火牆中的連接埠*」。如果您有多個網路介面，請按一下「*防火牆詳細資訊*」，選取介面，並按一下「*確定*」來選取 Samba 服務的網路介面。

共享

在「*共享*」索引標籤中，決定要啟用的 Samba 共享。標籤中有一些預先定義的共同，如 home 和 printer。使用「*切換狀態*」以切換「*作用中*」與「*非作用中*」。按一下「*新增*」可新增新的共用，按一下「*刪除*」可刪除選取的共用。

識別

在「*識別*」中，您可以決定主機的關聯領域 (「*基礎設定*」)，以及是否要在網路中使用替代的主機名稱 (「*NetBIOS 主機名稱*」)。若要查看進階全域設定或設定使用者驗證，如 LDAP，請按一下「*進階設定*」。

其他網域的使用者

若要讓其他網域的使用者存取您的網域，請在「*信任的網域*」索引標籤中進行適當的設定。若要新增網域，請按一下「*新增*」。若要移除所選網域，請按一下「*移除*」。

使用 LDAP

在索引標籤「*LDAP 設定*」中，您可決定 LDAP 伺服器是否使用驗證。若要測試 LDAP 伺服器的連線，請按一下「*測試連線*」。若要查看進階 LDAP 設定或使用者預設值，請按一下 「*進階設定*」。

如需更多有關 LDAP 組態的詳細資訊，請參閱第 36 章「*LDAP—一種目錄服務*」[605 頁]。

37.3.2 使用 SWAT 管理網頁

另一種管理 Samba 伺服器的工具是 SWAT (Samba Web Administration Tool，Samba 網頁管理工具)。它提供簡單的網頁介面，以用於設定 Samba 伺服器。若要使用 SWAT，請在網頁瀏覽器中開啟 `http://localhost:901`，並以 `root` 使用者的身份登入。如果您沒有特殊的 Samba root 帳戶，請使用系統的 `root` 帳戶。

注：啟用 SWAT

安裝 Samba 伺服器後，並不會啟用 SWAT。若要啟用 SWAT，請在 YaST 中開啟「*網路服務*」>「*網路服務 (xinetd)*」，啟用網路服務組態，選取表格中的「*swat*」，並按一下「*切換狀態 (開啟或關閉)*」。

37.3.3 手動設定伺服器

如果您想要使用 Samba 做為伺服器，請安裝 `samba`。Samba 的主要組態檔為 `/etc/samba/smb.conf`。這個檔案可以分成兩個邏輯部份。`[global]` 區段包含中央與全域設定值。`[share]` 區段包含個別檔案與印表機共享。利用此方法，就可以在 `[global]` 區段中以不同的方式或以全域方式設定關於共享的細節，它可以加強組態檔的結構透明化。

全域區段

下列 `[global]` 區段的參數需要做一些調整，以符合網路設定的需求，讓其他機器可以透過 Windows 環境中的 SMB 存取 Samba 伺服器。

workgroup = TUX-NET
: 這一行是將 Samba 伺服器指定給工作群組。以網路環境中適當的工作群組取代 `TUX-NET`。除非您已將這個名稱指定給網路中的其他機器，否則 Samba 伺服器將會以其 DNS 名稱來顯示。如果沒有 DNS 名稱，請使用 `netbiosname=MYNAME` 設定伺服器名稱。如需關於此參數的詳細資訊，請參閱 `mansmb.conf`。

os level = 2
: 此參數會觸發 Samba 伺服器是否嘗試變成其工作群組的 LMB (本地主要的瀏覽器)。請選擇一個非常低的值，以使現有的 Winodws 網路能避免設定不

當的 Samba 伺服器所造成的任何問題。如需關於此重要主題的詳細資訊，請參閱套件文件中 `textdocs` 子目錄下的 `BROWSING.txt` 與 `BROWSING-Config.txt` 檔案。

如果沒有其他的 SMB 伺服器在網路中 (例如，Windows NT 或 2000 伺服器)，而且您想要 Samba 伺服器保留在本地環境中所存在的所有系統清單，請將 `os level` 設成更高的值 (例如，`65`)。接著就會將 Samba 伺服器選擇成本地網路的 LMB。

當變更此設定值時，請小心地考慮這個值將會如何影響現有的 Windows 網路環境。首先請在獨立的網路中或在一天中非重要的時間測試變更。

wins support 與 wins server
: 若要將 Samba 伺服器整合至含有主動 WINS 伺服器的現有 Windows 網路中，請啟用 `wins server` 選項，並將其值設為該 WINS 伺服器的 IP 位址。

  如果您的 Windows 機器已連接至獨立的子網路中，而且應該要能夠看到彼此，則需要設定 WINS 伺服器。若要將 Samba 伺服器變成像這樣的 WINS 伺服器，請設定 `wins support = Yes` 選項。請確定網路中只有一個 Samba 伺服器啟用了這個設定值。`wins server` 與 `wins support` 選項絕不能在 `smb.conf` 檔案中同時啟用。

共享

下列範例說明如何將 CD-ROM 光碟機與使用者目錄 (`homes`) 開放給 SMB 用戶端使用。

[cdrom]
: 若要避免不小心將 CD-ROM 光碟機開放成共享，請以備註符號停用這些行 (在此例中為分號)。請在第一個資料欄中移除分號，以便和 Samba 共享 CD-ROM 光碟機。

範例 37.1 *CD-ROM 共用*

```
;[cdrom]
;       comment = Linux CD-ROM
;       path = /media/cdrom
;       locking = No
```

`[cdrom]` and `comment`

`[cdrom]` 這個項目是在網路上所有的 SMB 用戶端都可以看到的共享名稱。可以另外再加入一個 `comment`，以進一步描述共享。

`path = /media/cdrom`

`path` 會輸出 `/media/cdrom` 目錄。

利用限制非常嚴格的預設組態，就可以將這種共享只開放給出現在此系統上的使用者共享。如果這個共享應該開放每個人使用，請將 `guest ok = yes` 加入組態。這個設定值可以將讀取權限開放給網路上的每個人使用。建議您處理此參數時必須極為小心。這將會在 `[global]` 區段中套用更多此參數的使用。

`[homes]`

`[home]` 共用在這裏特別重要。如果使用者擁有 Linux 檔案伺服器以及其自己主目錄的有效帳戶與密碼，就可以連接到主目錄。

範例 37.2 *主目錄共享*

```
[homes]
 comment = Home Directories
 valid users = %S
 browseable = No
 read only = No
 create mask = 0640
 directory mask = 0750
```

[homes]

只要沒有其他的共享，使用共享的使用者名稱連接至 SMB 伺服器，就會使用 `[homes]` 共享指示來動態產生共享。共享的產生名稱是使用者名稱。

`valid users = %S`

只要成功地建立連接，就會以共享的具體名稱取代 `%S`。`[homes]` 共享，永遠為使用者名稱。因此，使用者共享的存取權限僅限於使用者。

`browseable = No`

這個設定值讓共享在網路環境變成無形的。

`read only = No`
根據預設，Samba 會利用 `read only = Yes` 參數，以禁止寫入任何輸出共享的權限。若要開放共享為可寫入的，請設定 `read only = No` 的值，這與 `writable = Yes` 同義。

`create mask = 0640`
以 MS Windows NT 為基礎的系統將無法理解 UNIX 權限的概念，因此在建立檔案時，它們將無法指定權限。`create mask` 參數可以定義指定給新建立檔案的存取權限。這只會套用至可寫入的共享。實際上，這個設定值表示擁有者具有讀取與寫入權限，而擁有者的主要群組成員則具有讀取權限。`valid users = %S` 可以在即使群組具有讀取權限時禁止讀取權限。若想要使群組具有讀取或寫入權限，請停用 `valid users = %S` 一行。

安全性層級

為了提高安全性，每個共用存取權都以密碼保護。SMB 具有三個檢查權限的可能方法：

共用層級安全性 (安全性 = 共用)
每個共享都必須指定密碼。每個知道此密碼的人員都具有該共享的存取權。

使用者層級安全性 (安全性 = 使用者)
這個變化引用了使用者對 SMB 的概念。每個使用者都必須以自己的密碼註冊伺服器。在註冊後，伺服器可以視使用者名稱將存取權授予個別輸出的共用。

伺服器層級安全性 (安全性 = 伺服器)：
對其用戶端而言，Samba 會模擬在使用者層級模式中工作。不過，它會將所有的密碼查詢傳遞給另一個將會處理驗證的使用者層級模式伺服器。這個設定值將需要另一個參數 (`password server`)。

共用、使用者或伺服器層級安全性的選擇會套用至整部伺服器。因為無法針對伺服器組態的個別共用提供共用層級的安全性，並針對其他的共用提供使用者層級的安全性。然而，您可以針對系統上每個設定的 IP 位址執行個別的 Samba 伺服器。

在「Samba HOWTO 文件集」中可以找到關於此主題的詳細資訊。至於在一個系統上的多個伺服器，請注意 `interfaces` 與 `bind interfaces only` 選項。

37.4 設定用戶端

用戶端只能透過 TCP/IP 存取 Samba 伺服器。NetBEUI 與透過 IPX 的 NetBIOS 無法與 Samba 一起使用。

37.4.1 使用 YaST 設定 Samba 用戶端

設定 Samba 用戶端以存取 Samba 伺服器上的資源 (檔案或印表機)。在「*網路服務*」>「*Windows 網域成員*」對話方塊中輸入網域或工作群組。按一下「*瀏覽*」以顯示可以使用滑鼠選取的所有可用的群組與網域。如果您啟用「*Linux 驗證也使用 SMB 資訊*」，使用者驗證將會在 Samba 伺服器上執行。在完成所有的設定後，按一下「*完成*」以完成組態。

37.4.2 Windows 9x 與 ME

Windows 9x 與 ME 已經有 TCP/IP 的內建支援。不過，預設並不會安裝它。如果要新增 TCP/IP，請進入「*控制台*」>「*系統*」中，然後選擇「*新增*」>「*通訊協定*」>「*Microsoft 的 TCP/IP*」。在重新啟動 Windows 機器後，按兩下網路環境中的桌面圖示以尋找 Samba 伺服器。

提示

如果要使用 Samba 伺服器上的印表機，請從對應的 Windows 版本安裝標準或 Apple-PostScript 印表機驅動程式。最好是將此與 Linux 印表機佇列連結，它可以接受 Postscript 做為輸入格式。

37.5 做為登入伺服器的 Samba

在以 Windows 用戶端為主的網路中，通常會建議使用者只註冊一個有效的帳戶與密碼。在以 Windows 為基礎的網路中，這個任務是由主要網域控制器 (PDC) 來處理。您可使用設定為 PDC 的 Windows NT 伺服器，但此任務亦可透過 Samba 伺服器的協助完成。在 `smb.conf` 的 `[global]` 區段中必須編輯的項目如 範例 37.3「在 smb.conf 中的全域區段」[644頁] 所示。

範例 37.3 *在 smb.conf 中的全域區段*

```
[global]
workgroup = TUX-NET
domain logons = Yes
domain master = Yes
```

如果使用加密的密碼來驗證—這是維護良好的 MS Windows 9x 安裝，MS Windows NT 4.0 加上 Service Pack 3 以及所有之後產品的預設值—Samba 伺服器必須能夠處理它們。在 `[global]` 區段中的 `encrypt passwords = yes` 項目可以啟用此功能 (加上 Samba 版本 3，現在這個是預設值)。除此之外，必須準備符合 Winodws 加密格式的使用者帳戶與密碼。請使用 `smbpasswd-a name` 指令來執行此動作。使用下列指令為電腦建立領域帳戶 (Windows NT 領域概念所需)：

範例 37.4 *設定機器帳戶*

```
useradd hostname\$
smbpasswd -a -m hostname
```

使用 `useradd` 指令，就會加上貨幣符號。當使用 `-m` 參數時，`smbpasswd` 指令就會自動插入這個符號。加備註的組態範例 (`/usr/share/doc/packages/Samba/examples/smb.conf.SuSE`) 包含自動化此任務的設定值。

範例 37.5 *機器帳戶的自動化設定*

```
add machine script = /usr/sbin/useradd -g nogroup -c "NT Machine Account" \
-s /bin/false %m\$
```

若要確定 Samba 可以正確地執行此程序檔，請選擇具有所需管理員權限的 Samba 使用者。若要這麼做，請選取一個使用者，並將它加入 `ntadmin` 群組。在此之

後，就可以使用下列指令指定 `Domain Admin` 狀態給所有屬於此 Linux 群組的使用者：

```
net groupmap add ntgroup="Domain Admins" unixgroup=ntadmin
```

如需關於此主題的詳細資訊，請參閱 `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf` 中第 12 章的「Samba HOWTO 文件集」。

37.6 含 Active Directory 的網路中之 Samba 伺服器

若您同時執行 Linux 伺服器與 Windows 伺服器，您可建立兩個獨立的驗證系統與網路，或將兩部伺服器透過一個中央驗證系統連接到一個網路。由於 Samba 可與 active directory 網域共同運作，因此您可將 SUSE Linux Enterprise Server 加入 Active Directory (AD) 中。

加入現有 AD 網域的動作可以在安裝期間進行，也可以後來藉由在安裝好的系統中使用 YaST 啟動 SMB 使用者驗證來進行。安裝時加入網域的說明可在第 3.14.7 節「使用者」 [40頁]中找到。

若要在運作中的系統加入 AD 網域，請執行下列步驟：

1. 以 `root` 身份登入並啟動 YaST。
2. 啟動「*網路服務*」>「*Windows 網域成員*」。
3. 在「*Windows 網域成員*」畫面的「*網域或工作群組*」中輸入要加入的網域。或者，使用「*瀏覽*」取得所有可用網域清單，從中選擇一個網域。

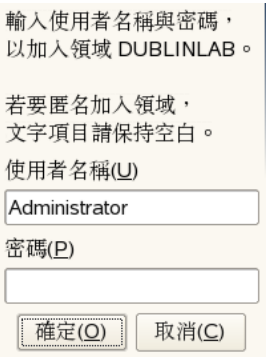
圖形 37.1　決定 Windows 網域成員

4 勾選「*Linux 驗證也使用 SMB 資訊*」，以在 SUSE Linux Enterprise Server 上使用 SMB 來源進行 Linux 驗證。

5 按一下「*完成*」，並在出現提示時確認要加入網域。

6 為 Windows 管理員提供 AD 伺服器上的密碼，並按一下「*確定*」。

圖形 37.2　提供管理員證件



您的伺服器現在已可使用 Active Directory 網域控制器上的所有驗證資料。

37.7 將 Windows NT 伺服器移轉至 Samba

與 Samba 和 LDAP 組態不同，將 Windows NT 伺服器移轉至 SUSE Linux Enterprise Server Samba 伺服器需要兩個基本步驟。首先，先移轉設定檔，再移轉帳戶。

37.7.1 準備 LDAP 伺服器

移轉的第一步驟是設定 LDAP 伺服器。您必須為軟體用戶端帳戶與密碼新增基本 DN 資訊與項目。關於 LDAP 組態的詳細資訊，請參閱第 36 章「*LDAP—一種目錄服務*」[605頁]。

設定無需手動進行。您可使用 smbldap 工具的程序檔。這些程序檔是 `samba-doc` 套件的一部份，安裝套件之後，可在 `/usr/share/doc/packages/samba/examples/LDAP` 找到。

注：LDAP 與安全性

LDAP 管理 DN 應不同於根 DN。為了讓網路更加安全，您亦可使用 TSL 安全連線。

37.7.2 準備 Samba 伺服器

開始移轉之前，請設定 Samba 伺服器。請在 YaST「*Samba 伺服器*」模組的「*共用*」索引標籤中找到 `profile`、`netlogon` 和 `home` 共用的組態。若要執行預設值，請選擇共用並按一下「*編輯*」。

若要新增 Samba 伺服器的 LDAP 組態與 LDAP 管理員的身份證明，請使用 YaST「*Samba 伺服器*」模組的「*LDAP 設定*」索引標籤。您必須具備 LDAP 管理 DN (標籤「*管理 DN*」) 和密碼才能新增或修改儲存於 LDAP 目錄中的帳戶。

37.7.3 移轉 Windows 設定檔

對於要移轉的每個設定檔，請完成下列步驟：

過程 37.1　移轉設定檔

1. 在您的NT4網域控制器上，在「*我的電腦*」上按一下右鍵，然後選取「*內容*」。選擇「*使用者設定檔*」標籤。
2. 選擇您要移轉的使用者設定檔並按一下。
3. 按一下「*複製到*」。
4. 在「*複製設定檔*」中新增新路徑，例如 `c:\temp\profiles`。
5. 按一下「*允許*」中的「*變更*」。
6. 按一下「*任何人*」。若要關閉方塊，請按一下「*確定*」。
7. 儲存完設定檔之後，按一下「*確定*」。
8. 將儲存的設定檔複製至您 Samba 伺服器上的適當目錄。

37.7.4 移轉 Windows 帳戶

過程 37.2　帳戶移轉程序

1. 使用 NT Server Manager，在 Samba 伺服器的舊 NT4 網域中建立一個 BDC 帳戶。此時不得執行 Samba。

```
     net rpc join -S NT4PDC -w DOMNAME -U Administrator%passwd net rpc
vampire
     -S NT4PDC -U administrator%passwd pdbedit -L
```

2. 將各個 UNIX 群組指派至 NT 群組：

範例 37.6　範例程序檔 initGroups.sh

```
 #!/bin/bash #### Keep this as a shell script for future re-use #
       Known domain global groups net groupmap modify ntgroup="Domain
Admins"
       unixgroup=root net groupmap modify ntgroup="Domain Users"
       unixgroup=users net groupmap modify ntgroup="Domain Guests"
       unixgroup=nobody # Our domain global groups net groupmap add
       ntgroup="Operation" unixgroup=operation type=d net groupmap add
       ntgroup="Shipping" unixgroup=shipping type=d
```

3 檢查是否已辨識所有群組：

```
net groupmap list
```

37.8 如需更多資訊

可在數位文件中取得更多有關 Samba 的資訊。如果已安裝 Samba 文件，若要取得更多的線上文件與範例，請在指令行輸入 `apropossamba` 以顯示一些手冊頁或直接瀏覽 `/usr/share/doc/packages/samba` 目錄。在 `examples` 子目錄中有加備註的範例組態 (`smb.conf.SuSE`)。

Samba 團隊所提供的「Samba HOWTO 文件集」，包含疑難排解一節。除此之外，文件的第五部份提供檢查組態的逐步指南。安裝 `samba-doc` 套件後，您可以在 `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf` 中找到 Samba HOWTO 文件集。

如需 LDAP 以及從 Windows NT 或 2000 移轉的詳細資訊，請參閱 `/usr/share/doc/packages/samba/examples/LDAP/smbldap-tools-*/doc`，其中 * 是您的 smbldap 工具版本。

38 使用 NFS 共享檔案系統

透過網路配送和共享檔案系統在公司環境中極為常見。NFS是一個成熟的系統，也可以與黃頁通訊協定 NIS 搭配使用。如需可與 LDAP 搭配使用並且也可 Kerberos 化的更安全通訊協定，請勾選 NFSv4。

與 NIS 一起使用 NFS 會讓網路對使用者而言透明化。可以使用 NFS 透過網路配送任意檔案系統。如果安裝妥當，使用者將會發現，不論他們目前使用哪個終端機，他們始終處於同一個環境中。

NFS 與 NIS 一樣，都是主從式系統。但一台機器可同時扮演這兩種角色 — 它可透過網路提供檔案系統 (輸出)，也可以從其他主機裝載檔案系統 (輸入)。

重要：DNS 所需

原則上，可以只使用 IP 位址來進行所有的輸出。為了避免逾時，您應該具有運作中的 DNS 系統。這是基於記錄目的所必備的，因為裝載的精靈會進行反向查詢。

38.1 安裝必要軟體

您不需安裝其他軟體就能將您的主機設定為 NFS 用戶端。設定 NFS 用戶端時所需的套件都已預設安裝。

NFS 伺服器軟體並不屬於預設安裝的部分。若要安裝 NFS 伺服器軟體，請啟動 YaST 並選取「*軟體*」>「*軟體管理*」。接著請選擇「*過濾器*」>「*模式*」，再

選擇「*其他伺服器*」或使用「*搜尋*」選項來搜尋NFS 伺服器。請確認套件的安裝，完成此安裝程序。

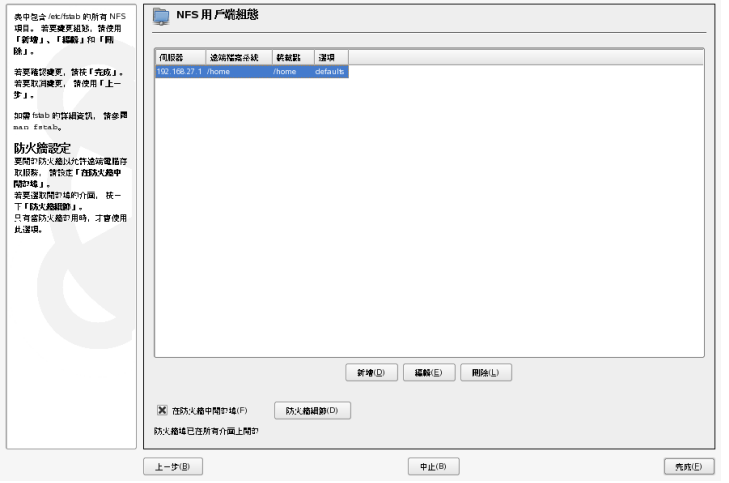
38.2 以 YaST 輸入檔案系統

被授權的使用者可從NFS伺服器裝載NFS目錄到自己的檔案目錄樹。可以使用YaST 模組「*NFS 用戶端*」實現此目的。僅需鍵入 NFS 伺服器的主機名稱、要輸入的目錄和在本地裝載此目錄的裝載點。按一下第一個對話方塊中的「*新增*」後，變更就會生效。按一下「*開啟防火牆中的連接埠*」開啟防火牆，可從遠端電腦存取服務。防火牆的狀態顯示於核取方塊旁。您可以按一下「*完成*」來儲存變更。請參閱圖形 38.1「使用 YaST 的 NFS 用戶端組態」[652頁]。

組態將會寫入/etc/fstab中，並會裝載指定的檔案系統。當您稍後啟動YaST組態用戶端時，它也會從這個檔案讀取現有組態資訊。

NFSv4 檔案系統現在只能以手動的方式輸入。第 38.3 節「手動輸入檔案系統」[653頁]中有相關的說明。

圖形 38.1 *使用 YaST 的 NFS 用戶端組態*



38.3 手動輸入檔案系統

您也可以從NFS伺服器手動輸入檔案系統。先決條件是執行中的RPC埠對應程式，以根部使用者身份輸入 `rcportmap start` 便可啟動。一旦符合先決條件，就可以下列方式使用`mount`指令，讓遠端輸入的檔案系統就能像本地硬碟一樣地在檔案系統內進行裝載：

```
mount host:remote-path local-path
```

例如，如果要輸入來自機器 sun 的使用者目錄，請使用以下指令：

```
mount sun:/home /home
```

38.3.1 輸入 NFSv4 檔案系統

idmapd服務必須在用戶端上啟用並執行，才能進行NFSv4的輸入作業。請在指令提示下以 `rcidmapd start` 啟動 idmapd 服務。您可以使用 `rcidmapd status` 來檢查 idmapd 的狀態。

idmapd 服務會將它的參數儲存在 `/etc/idmapd.conf` 檔案中。請讓 `Domain` 參數保持為 `localdomain`。請確認您分別針對 NFS 用戶端和 NFS 伺服器所指定的值是一樣的。

請在外圍程序的提示下輸入指令，以進行 NFSv4 的輸入作業。若要輸入 NFSv4 遠端檔案系統，請使用下列指令：

```
mount -t nfs4 host:/ local-path
```

請以代管一或多個 NFSv4 輸出作業的 NFS 伺服器取代 `host`，而以用戶端中將用於裝載的目錄位置取代 `local-path`。例如，若要將透過 sun 之 NFSv4 輸出的 `/home` 輸入到 `/local/home`，請使用下列指令：

```
mount -t nfs4 sun:/ /local/home
```

伺服器名稱和冒號所接的遠端檔案系統路徑為斜線「/」。這與輸入 v3 時的指定方式不同，該方式會提供遠端檔案系統的確切路徑。這個概念稱為*虛擬檔案系統*，第 38.4.1 節「NFSv4 用戶端的輸出」[656頁]中有相關說明。

38.3.2 使用自動裝載服務

除了一般本地設備的裝載之外，autofs 精靈也可以用來裝載遠端檔案系統。若要這麼做，請將下列項目加入您的 `/etc/auto.master` 檔案：

```
/nfsmounts /etc/auto.nfs
```

如果 `auto.nfs` 檔案能適當完成，`/nfsmounts` 目錄此後就可做為用戶端上所有 NFS 裝載作業的根部。`auto.nfs` 這個名稱的選擇是以方便為考量，您可以自行選擇任何名稱。在選取的檔案中 (若不存在，則請您建立) 加入所有 NFS 裝載作業的項目，如以下範例所示：

```
localdata -fstype=nfs server1:/data
nfs4mount -fstype=nfs4 server2:/
```

請以 `rcautofs start` 啟用設定。在此範例中，`/data` 伺服器 1 的 `/nfsmounts/localdata` 目錄會裝載 NFS，而伺服器 2 的 `/nfsmounts/nfs4mount` 會裝載 NFSv4。

如果 `/etc/auto.master` 檔案在 autofs 服務的執行過程中接受編輯，則自動裝載器必須重新啟動才能使變更生效。請以 `rcautofs restart` 執行此操作。

38.3.3 手動編輯 /etc/fstab

`/etc/fstab` 中典型的 NFS 裝載項目如下：

```
host:/data /local/path nfs rw,noauto 0 0
```

您也可以手動將 NFSv4 的裝載加入 `/etc/fstab` 檔案。對於這些裝載作業，請在第三欄中使用 `nfs4` (而非使用 `nfs`)，並確定遠端檔案系統在第一欄中的 *`host:`* 之後做為 `/` 提供。將此項資訊儲存在 `/etc/fstab` 的好處是可以讓裝載指令縮短，只需指出本地裝載點即可，例如：

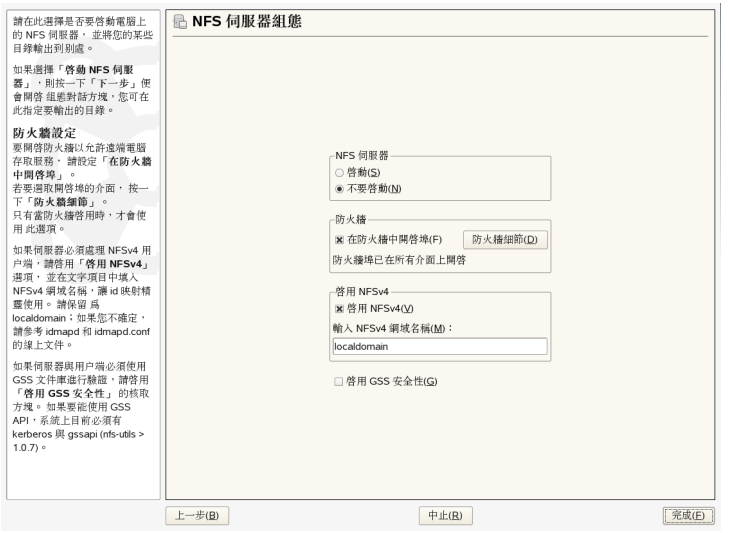
```
mount /local/path
```

38.4 以 YaST 輸出檔案系統

使用 YaST，將您網路中的主機轉變為 NFS 伺服器，此類伺服器可將目錄和檔案輸出到所有擁有存取權的主機。可不在每一個主機上本地安裝時，提供應用

程式給團隊中的所有工作夥伴。若要安裝此種伺服器，請啟動 YaST 並選取「*網路服務*」>「*NFS 伺服器*」。會開啟像 圖形 38.2「NFS 伺服器組態工具」[655頁] 中的對話方塊。

圖形 38.2 *NFS 伺服器組態工具*

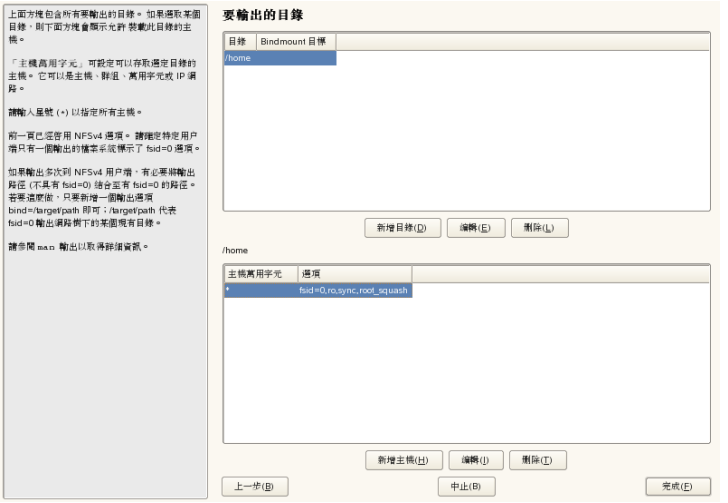


接著，請啟用「*啟動 NFS 伺服器*」並輸入「*NFSv4 領域名稱*」。

若您需要安全存取伺服器，請按一下「*啟用 GSS 安全性*」。先決條件是您的領域中有安裝 Kerberos，且伺服器和用戶端都已獲監督 (kerberized)。按「*下一步*」。

在上方的文字欄位中，鍵入要輸入的目錄。在下方，輸入可以存取它們的主機。在 圖形 38.3「以 YaST 設定 NFS 伺服器」[656頁] 中顯示了此對話方塊。此圖顯示當 NFSv4 已於前一個對話方塊中啟用的狀況。結合裝載目標將顯示於右側窗格中。如需詳細資訊，請參閱左側窗格中的說明。對話方塊的下半部有四個選項可針對各主機進行設定：「單一主機」、「網路群組」、「萬用字元」和「IP 網路」。如需這些選項的詳細說明，請參閱「輸出」線上文件。按一下「*完成*」以完成組態。

圖形 38.3　以 YaST 設定 NFS 伺服器



重要：自動防火牆組態

如果您的系統上有防火牆在作用中 (SuSEfirewall2)，YaST 會在選取「*在防火牆中開啟埠*」時，啟用 `nfs` 服務以調整其 NFS 伺服器組態。

38.4.1 NFSv4 用戶端的輸出

請啟用「*啟用 NFSv4*」來支援 NFSv4 用戶端。擁有 NFSv3 的用戶端仍然可以存取伺服器所輸出的目錄，只要輸出正確即可。如需詳細說明，請參閱第 38.4.3 節「共存的 v3 和 v4 輸出」[659頁]。

啟用 NFSv4 以後，請輸入適當的領域名稱。請確定所輸入的名稱與所有存取此特定伺服器之 NFSv4 用戶端中 `/etc/idmapd.conf` 檔案顯示的名稱相同。此參數適用於 idmapd 服務，而此服務在支援 (伺服器和用戶端二者的) NFSv4 時是必要的。如果您沒有特殊需求，則請讓它維持為 `localdomain` (預設)。如需詳細資訊，請參閱第 38.7 節「如需更多資訊」[662頁]。

按一下「*下一步*」。接著顯示的對話方塊擁有兩個區段：上半部有兩欄，分別稱為「*目錄*」和「*結合裝載目標*」。「*目錄*」欄可讓您直接編輯，其中列出要輸出的目錄。

對於固定的用戶端集合來說，能輸出的目錄有兩種類型—做為虛擬根部檔案系統的目錄，以及結合至虛擬檔案系統之某些子目錄的目錄。虛擬檔案系統可做為基礎點，底下所有為這些用戶端集合而輸出的檔案系統都有其位置。對於用戶端或用戶端集合來說，伺服器上只有一個目錄可以設定為虛擬根部來進行輸出。針對此用戶端，可在輸出多個目錄時將它們繫結至虛擬根部中某些現有的子目錄。

圖形 38.4 *以 NFSv4 輸出目錄*

請在對話方塊的下半部輸入用戶端 (萬用字元) 並輸出特定目錄的目錄。將某目錄新增至上半部以後，就會自動出現一個對話方塊讓您輸入用戶端和選項資訊。然後，若要新增用戶端 (用戶端集合)，請按一下「*新增主機*」。

請在開啟的小對話方塊中輸入主機萬用字元。有四種主機萬用字元類型可讓您針對各主機進行設定：單一主機 (名稱或 IP 位址)、網路群組、萬用字元 (例如 * 標是所有機器都可存取伺服器) 以及 IP 網路。然後，請在「*選項*」將加入 `fsid=0` 加入逗號分隔的選項清單，以將目錄設為虛擬根部。如果此目錄應該繫結至某個已設定虛擬根部之下的目錄，請確定已使用`bind=/target/path` 在選項清單中提供目標繫結路徑。

舉例來說，假設目錄 `/exports` 被選為所有可存取伺服器之用戶端的虛擬根部目錄。然後請將其加入上半部，並確定針對此目錄所輸入的選項包含了`fsid=0`。如果另一個目錄 (`/data`) 也需要使用 NFSv4 輸出，請將該目錄輸出到上半部。

針對此狀況輸入選項時，請確定 `bind=/exports/data` 位於清單中，而且 `/exports/data` 是 `/exports` 的現有子目錄。`bind=/target/path` 選項中的任何變更，不論是值的新增、刪除或變更，都會反映在「*結合裝載目標*」中。此欄無法供您直接編輯，而只是列出目錄及其屬性。填完此資訊時，請按一下「*完成*」完成組態，或按一下「*啟動*」來重新啟動服務。

38.4.2 NFSv3 和 NFSv2 的輸出

請在按一下「*下一步*」之前先確定起始對話方塊中的「*啟用NFSv4*」並未勾選。

下一個對話方塊包括兩個部分。在上方的文字欄位中，鍵入要輸入的目錄。在下方，輸入可以存取它們的主機。有四種主機萬用字元類型可讓您針對各主機進行設定：單一主機 (名稱或 ip 位址)、網路群組、萬用字元 (例如 * 摽是所有機器都可存取伺服器) 以及 IP 網路。

在 圖形 38.4「以 NFSv4 輸出目錄」[657頁] 中顯示了此對話方塊。如需這些選項的詳細說明，請參閱 `man exports`。按一下「*完成*」以完成組態。

圖形 38.5 *以 NFSv2 和 v3 輸出目錄*

38.4.3 共存的 v3 和 v4 輸出

NFSv3 和 NFSv4 的輸出可共存於同一個伺服器上。在起始組態對話方塊啟用了對 NFSv4 的支援之後，對於選項清單中沒有納入 `fsid=0` 和 `bind=/target/path` 的輸出，v3 的輸出作業會予以考慮。請參閱圖形 38.4「以 NFSv4 輸出目錄」[657頁]中的範例。如果您使用「*新增目錄*」來加入其他目錄 (例如 `/data2`)，而對應的選項清單並未列出 `fsid=0` 或 `bind=/target/path`，則此輸出即屬於 v3 的輸出作業。

重要

自動防火牆組態

如果您的系統上有 SuSEfirewall2 在作用中，YaST 會在選取「*開啟防火牆中的連接埠*」時，啟用服務來調整其 NFS 伺服器組態。

38.5 手動輸出檔案系統

NFS 輸出服務的組態檔案為 `/etc/exports` 和 `/etc/sysconfig/nfs`。除了這些檔案之外，NFSv4 伺服器組態還需要 `/etc/idmapd.conf`。若要啟動或重新啟動服務，請執行 `rcnfsserver restart` 和 `rcidmapd restart` 指令。NFS 伺服器需依賴執行的 RPC 埠對應程式。因此，也請您以 `rcportmap restart` 啟動或重新啟動埠對應程式服務。

38.5.1 以 NFSv4 輸出檔案系統

NFSv4 是 SUSE Linux Enterprise 10 上最新版的 NFS 協定。設定要以 NFSv4 輸出的目錄，程序和之前的版本有些許不同。

/etc/exports 檔案

此檔案含有一個項目清單。每一個項目都指出一個共享的目錄，並記錄它的共享方式。`/etc/exports` 中的典型項目會包含：

```
/shared/directory host(option_list)
```

例如：

```
/export 192.168.1.2(rw,fsid=0,sync)
/data 192.168.1.2(rw,bind=/export/data,sync)
```

對於 `fsid=0` 已在選項清單中指定的目錄，稱為虛擬根部檔案系統。這裡使用了 IP 位址 192.168.1.2。您可以使用主機的名稱，那是一個萬用字元，指向一組主機 (`*.abc.com`、`*` 等) 或網路群組。

對於固定的用戶端集合，只有兩種目錄可以使用 NFSv4 輸出：

- 用以做為虛擬根部檔案系統的單一目錄。在此範例中，`/exports` 為虛擬根部目錄，因為 `fsid=0` 已於此項目的選項清單中指定。
- 用以結合至虛擬檔案系統之某些現有子目錄的目錄。在上面的項目範例中，`/data` 這個目錄就是繫結至 `/export` 虛擬檔案系統的現有子目錄 (`/export/data`)。

虛擬檔案系統是最上層的目錄，底下的所有需使用 NFSv4 輸出的檔案系統都有其位置。對於用戶端或用戶端集合來說，伺服器上只有一個目錄可以設定為虛擬根部來進行輸出。對此用戶端或用戶端集合，您可以將其他多個目錄繫結之虛擬根部中的某些現有子目錄，進而將它們輸出。

/etc/sysconfig/nfs

此檔案含有某些參數來決定 NFSv4 伺服器精靈的行為。NFSv4_SUPPORT 參數必須設定為 yes，這點相當重要。此參數可決定 NFS 伺服器是否可支援 NFSv4 輸出和用戶端。

/etc/idmapd.conf

Linux 機器的所有使用者都有名稱和 ID。idmapd 會針對伺服器接收的 NFSv4 要求進行名稱和 ID 的對應，然後對用戶端發出回覆。這必須同時在 NFSv4 的伺服器和用戶端上執行，因為 NFSv4 只會在通訊中使用名稱。

對於可能使用 NFS 來共用檔案系統的機器，請確定有一個統一的方式來對各機器之間的使用者指定使用者名稱和 ID (uid)。您可以透過 NIS、LDAP 或您領域中的任何統一領域驗證機制來達成這個目的。

為求運作正常，必須在此檔案中為用戶端和伺服器設定相同的 Domain 參數。如果您不確定，請讓伺服器和用戶端檔案中的領域都維持為 localdomain。我們在此提出一個組態檔案的例子，如下所示：

```
[General]

Verbosity = 0
Pipefs-Directory = /var/lib/nfs/rpc_pipefs
Domain = localdomain

[Mapping]

Nobody-User = nobody
Nobody-Group = nobody
```

除非您確定自己在做什麼，否則請勿變更這些參數。日後如需參考，請參閱 idmapd 和 idmapd.conf; man idmapd, man idmapd.conf 的線上文件。

啟動和停止服務

對 /etc/exports 或 /etc/sysconfig/nfs 進行變更後，請以 rcnfsserver restart 啟動或重新啟動 NFS 伺服器服務。變更了 /etc/idmapd.conf 以後，請以 rcidmapd restart 啟動或重新啟動 idmapd 服務。請確定兩個服務都在執行。

38.5.2 以 NFSv2 和 NFSv3 輸出檔案系統

這只適用於 NFSv3 和 NFSv2 的輸出作業。如需以 NFSv4 輸出的詳細資訊，請參閱第 38.5.1 節「以 NFSv4 輸出檔案系統」[659頁]。

以 NFS 輸出檔案系統時需要兩個組態檔案：/etc/exports 和 /etc/sysconfig/nfs。典型 /etc/exports 檔案項目的格式如下：

```
/shared/directory host(list_of_options)
```

例如：

```
/export 192.168.1.2(rw,sync)
```

這裡的 /export 目錄與主機 192.168.1.2 共用，選項清單為 rw,sync。這個 IP 位址可使用萬用字元來以用戶端名稱或用戶端集合取代(例如 *.abc.com)，甚至也可以網路群組取代。

如需所有選項及其意義的詳細說明，請參閱 `exports` (`man exports`) 的線上文件。

變更了 `/etc/exports` 或 `/etc/sysconfig/nfs` 以後，請以 `rcnfsserver restart` 指令啟動或重新啟動 NFS 伺服器。

38.6 NFS 搭配使用 Kerberos

若要為 NFS 使用 Kerberos 驗證，則 GSS 安全性必須啟用。若要這麼做，請在 YaST 起始對話方塊中選取「*啟用 GSS 安全性*」。請完成以下的額外步驟：

- 請確定伺服器和用戶端位於相同的 Kerberos 領域中。這表示它們會存取相同的 KDC (金鑰發佈中心) 伺服器，並會共用它們的 `krb5.keytab` 檔案 (所有機器上的預設位置都是 `/etc/krb5.keytab`)。
- 請以 `rcgssd start` 來啟動用戶端上的 gssd 服務。
- 請以 `rcsvcgssd start` 來啟動伺服器上的 svcgssd 服務。

如需有關設定已獲監督 (kerberized) 之 NFS 的詳細資訊，請參閱第 38.7 節「如需更多資訊」[662頁]中的連結。

38.7 如需更多資訊

除了 `exports`、`nfs` 和 `mount` 的線上文件以外，`/usr/share/doc/packages/nfs-tils/README` 以及下列的網頁文件中也有 NFS 伺服器和用戶端的設定資訊：

如需詳細的線上技術文件，請造訪 SourceForge [`http://nfs.sourceforge.net/`]

如需設定已監督之 NFS 的指示，請參閱 NFS 第 4 版開放原始碼實作參考 [`http://www.citi.umich.edu/projects/nfsv4/linux/krb5-setup.html`]

如果您有任何關於 NFSv4 的問題，請參閱 Linux NFSv4 常見問題 [`http://www.citi.umich.edu/projects/nfsv4/linux/faq/`]的 FAQ。

39 檔案同步化

今日有許多人使用數台電腦—在家使用一台、在辦公室使用一台或數台電腦，而且可能在外面還使用筆記型電腦或 PDA。許多檔案都需要各存一份在所有這些電腦上。您可能希望能在每一部電腦上工作、修改檔案，之後還能讓所有的電腦都擁有最新的資料。

39.1 可用的資料同步化軟體

對於永久以快速網路連結的電腦而言，資料同步化不是問題。在此狀況下，使用 NFS 之類的網路檔案系統，並將檔案儲存在伺服器上，所有主機就可以透過網路來存取相同的資料。但如果網路連線品質很差或不是永久的，就無法使用此方法。當您出門在外使用筆記型電腦時，所有需要的檔案複本都必須在本地硬碟上。不過之後需要將修改過的檔案同步化。當您在某一台電腦上修改檔案時，請確定也更新了所有其他電腦上的同一檔案。至於一些零散的複本，則可以使用 scp 或 rsync 手動更新它。然而，如果有許多檔案需要處理，則該程序有可能變得很複雜，而且需要更小心才能避免類似以舊檔案覆寫新檔案的錯誤。

警告：資料遺失的風險

在您開始使用同步化系統管理資料之前，應該先好好的認識要使用的程式並測試其功能。對於重要檔案而言，備份是不可或缺的動作。

手動同步化資料非常耗時，而且是一種容易發生錯誤的任務，要避免這些缺點，可以使用一種以多種方法將此工作自動化的程式。下列摘要讓您概略瞭解這些程式的運作方式和使用方法。如果您打算使用它們，請詳閱程式文件。

39.1.1 CVS

CVS 主要是用來管理程式來源版本，它使得檔案複本可以保留在多台電腦上。因此，它也適用於資料同步化。CVS 負責維護伺服器上的中央儲存庫，檔案及檔案的變更都儲存在此。在本地執行的變更會交付至儲存庫，並且可以利用更新從其他電腦擷取。兩個程序都必須由使用者起始。

CVS 對於錯誤有非常大的彈性與包容性，所以可以應付多台電腦同時進行變更的情況。變更會合併，而且如果在相同行發生變更，就會報告衝突。衝突發生時，資料庫會維持一致的狀態。只有在用戶端主機上才能看到衝突，並加以解決。

39.1.2 rsync

當不需要版本控制，但是大型目錄結構需要透過緩慢的網路連線進行同步化時，rsync 工具針對僅傳送檔案中的變更可提供已開發成熟的機制。這不只包含文字檔，也包含二進位檔。為了偵測檔案之間的不同，rsync 會將檔案分為區塊並計算它們的檢查總數。

偵測變更將需要付出相當大的成本。要同步化的系統應該要具有相當的硬體配備，才能使用 rsync。RAM 尤其重要。

39.2 選取程式時所要考慮的決定性因素

在決定要使用哪個程式時，必須考慮一些重要的因素。

39.2.1 用戶端對伺服器與點對點

配送資料時常使用的模式有兩種。第一個模式是，所有的用戶端都以中央伺服器為準，將其檔案同步化。伺服器至少必須偶爾可以讓所有的用戶端存取。CVS 使用此模式。

另一種可能性就是，讓網路上所有主機都以點對點的方式將彼此間的資料同步化。rsync 實際作用於用戶端模式，但任何用戶端都可以當作伺服器使用。

39.2.2 可攜式

在許多其他的作業系統上也可以使用CVS以及rsync，包含各種Unix與Windows系統。

39.2.3 互動式與自動化

在CVS中，資料同步化是由使用者以手動方式啟動。這讓使用者對於要同步化的資料進行良好的控制，並可輕鬆地處理衝突。然而，如果同步化間隔太長，就比較可能發生衝突。

39.2.4 衝突：發生與解決

即使有數個人員同時在某個大型的程式專案上一起工作，在CVS中發生衝突的機率還是相當地少。這是因為文件是在個別的行列上進行合併。當發生衝突時，只有一個用戶端會受到影響。CVS中的衝突通常都可以輕易解決。

rsync中則無衝突處理功能。使用者必須小心不要覆寫檔案，並手動解決所有可能的衝突。基於安全著想，可以另外使用RCS這一類的版本設定系統。

39.2.5 選取和新增檔案

在CVS中，必須分別使用`cvs`或`add`指令，明確地新增目錄與檔案。這讓使用者對於要同步化的檔案擁有更大的控制權。另一方面，新檔案時常會被忽略，特別是在處理大量的檔案而忽略了`cvs`以及`update`輸出中的問號時。

39.2.6 歷程

CVS還有另一項功能，那就是可以重新建構舊的檔案版本。每個變更都可以插入簡短的編輯符號，而且之後可以根據其內容與符號輕易地追蹤檔案的發展。這對論文與程式文字而言，是一種很珍貴的助力。

39.2.7 資料量與硬碟需求

所有相關主機的硬碟都需要有足夠的可用空間來儲存所有分散式的資料。CVS 在伺服器上還需要額外的空間，供儲存庫資料庫使用。檔案歷程記錄也會儲存在伺服器上，因此需要更多的空間。當文字格式的檔案變更時，只需儲存修改過的那幾行。每當變更檔案時，二進位檔案就會需要與該檔案大小相同的額外空間。

39.2.8 GUI

有經驗的使用者通常會從指令行執行 CVS。然而，在 cervisia 之類的 Linux 系統中，以及 wincvs 之類的其他作業系統中都有圖形使用者介面。許多開發工具 (像是 kdevelop) 以及文字編輯器 (像是 Emacs) 都支援 CVS。使用這些前端程式的話，衝突的解決方案通常會更容易執行。

39.2.9 使用者親切性

rsync 較易於使用且適合新進人員。CVS 某種程度上較難操作。使用者必須瞭解儲存庫與本地資料之間的互動。對資料的變更應該先在本地與儲存庫合併。這是使用 `cvs` 或 `update` 指令來執行。接著必須使用 `cvs` 或 `commit` 指令將資料傳送回儲存庫。只要瞭解此程序，新進人員就可輕鬆使用 CVS。

39.2.10 防止攻擊的安全性

在傳輸期間，應該保護資料以防攔截和竄改。CVS 與 rsync 都可以透過 ssh (安全的外圍程式) 來使用，提供安全性以防護此類攻擊。應該避免透過 rsh (遠端外圍程式) 執行 CVS。同樣地不建議在不安全的網路中使用 *pserver* 機制存取 CVS。

39.2.11 針對資料遺失的防護

開發人員使用 CVS 來管理程式專案已經有很長的一段時間，而且極為穩定。因為開發的歷程記錄皆已儲存，所以 CVS 甚至提供保護，防止某些使用者錯誤發生，例如不小心刪除檔案。

表格 39.1　檔案同步化工具的功能：-- = 很差，- = 差或無法使用，o = 中等，+ = 良好，++ = 優異，x = 可用

	CVS	rsync
主/從	C-S	C-S
可攜式	Lin、Un*x、Win	Lin、Un*x、Win
互動	x	x
速度	o	+
衝突	++	o
檔案 Sel。	Sel./file, dir.	目錄
歷程	x	-
硬碟空間	--	o
GUI	o	-
困難度	o	+
攻擊	+ (ssh)	+(ssh)
資料損失	++	+

39.3 CVS 簡介

如果經常編輯某些個別檔案，並且以檔案格式儲存，例如 ASCII 文字或程式來源文字，就非常適合使用 CVS 進行同步化。您可以使用 CVS 同步化其他格式的資料，例如 JPEG 檔案，但是會造成大量的資料，因為檔案的所有變體都會永久儲存在 CVS 伺服器上。在這種情形下，CVS 的大部份功能都將無法使用。只有在所有工作站都可以存取同一伺服器時，才能使用 CVS 同步化檔案。

39.3.1 設定 CVS 伺服器

*server*是所有有效檔案所在的主機，這包含所有檔案的最新版本。任何靜態工作站都可做為伺服器。可能的話，CVS 儲存庫的資料應該包含在定期備份中。

當設定CVS伺服器時，透過SSH授予使用者該伺服器的存取權可能是不錯的方式。如果使用者是以tux的身份登入伺服器，而且CVS軟體既安裝在伺服器上也安裝在用戶端上，則必須在用戶端上設定下列環境變數：

```
CVS_RSH=ssh CVSROOT=tux@server:/serverdir
```

cvsinit 指令可用來從用戶端啟始 CVS 伺服器。這只需要做一次。

最後，必須為同步化指定名稱。在用戶端上選取或建立一個目錄，專門存放CVS所管理的檔案 (目錄也可以是空的)。該目錄的名稱也就是同步化的名稱。在此範例中，該目錄稱為 synchome。變更至此目錄，然後輸入下列指令將同步化名稱設為 synchome：

```
cvs import synchome tux wilber
```

許多 CVS 指令需要註解。因此，CVS 會啟動編輯器 (如果沒有定義編輯器，則會啟動環境變數 $EDITOR 或 vi 中所定義的編輯器。)在指令行先輸入註解，就可以避免編輯器的呼叫，如下列範例所示：

```
cvs import -m 'this is a test' synchome tux wilber
```

39.3.2 使用 CVS

現在可以使用 cvsco synchome 從所有主機取出同步化儲存庫。這會在用戶端上建立新的子目錄synchome。若要將變更交付至伺服器，請換至synchome目錄 (或其中一個子目錄)，然後輸入 cvscommit。

根據預設，所有的檔案 (包括子目錄) 都會交付給伺服器。若只要交付個別的檔案或目錄，請在 cvscommit file1 directory1 中指定它們。在將它們交

付至伺服器前，必須使用像是 cvsadd file1 directory1 之類的指令，將新檔案與目錄新增至儲存庫。接著，使用 cvscommit file1 directory1 交付新增的檔案與目錄。

您更換至另一個工作站時，如果先前的工作階段尚未取出該工作站的同步化儲存庫，請現在取出。

使用 cvsupdate 啟動與伺服器的同步化。依照 cvsupdate file1 directory1 中的個別檔案或目錄來更新。若要查看目前檔案與伺服器上所存版本的差異，請使用 cvsdiff 或 cvsdiff file1 directory1 指令。使用 cvs-nq update 來查看哪些檔案會受到更新的影響。

以下是更新期間所顯示的一些狀態符號：

U
: 已更新本地版本。這將會影響伺服器所提供及在本地系統上所遺失的全部檔案。

M
: 已修改本地版本。如果在伺服器上有一些變更，可以在本地複本中將差異合併。

P
: 本地版本已使用伺服器版本修補。

C
: 本地檔案與儲存庫中目前的版本衝突。

?
: 這個檔案並不存在於 CVS。

M 狀態是指在本地修改過的檔案。或是將本地複本交付至伺服器，或是移除本地檔案後再執行一次更新。在此例中，會從伺服器擷取遺失的檔案。如果您交付本地已修改的檔案，而該檔案是在同一行中變更並交付，則有可能造成衝突，這是以 C 表示。

如果發生這種狀況，請查看檔案中的衝突記號 (»> 與 «<)，並在兩個版本之間做一選擇。這有可能是相當麻煩的工作，您可能會決定捨棄變更、刪除本地檔案，然後輸入 cvsup，以便從伺服器擷取目前的版本。

39.3.3 如需更多資訊

本小節僅針對 CVS 的許多可能性提供簡短的介紹。下列 URL 提供許多相關文件：

- CVS：http://www.cvshome.org
- Rsync：http://www.gnu.org/manual

39.4 rsync 簡介

當有大量的資料需要定期傳輸，但是並無太多變更時，rsync 就非常有用。例如，當建立備份時就非常適用。另一個應用程式則將焦點放在建置伺服器。這些伺服器儲存了網頁伺服器的完整目錄樹狀結構，會定期鏡像處理至 DMZ 中的網頁伺服器。

39.4.1 組態與作業

rsync 可以使用兩個模式來操作。它是用來歸檔或複製資料。若要完成此動作，在目標系統上只需要像是 ssh 的遠端外圍程序。然而，rsync 也可以像 daemon 一樣，用來為網路提供目錄。

rsync 的基本作業模式並不需要任何特殊的組態。rsync 直接允許將完整的目錄鏡像處理至另一個系統。例如，下列指令可在名為 sun 的備份伺服器上建立 tux 主目錄的備份：

```
rsync -baz -e ssh /home/tux/ tux@sun:backup
```

下列指令可用來還原目錄：

```
rsync -az -e ssh tux@sun:backup /home/tux/
```

到此為止，其處理方式與一般的複製工具 (例如 scp) 的處理沒有太大的差別。

rsync 應該在「rsync」模式中操作，才能充分使用其所有的功能。執行方法是在其中一個系統上啟動 rsyncd 精靈。在 /etc/rsyncd.conf 檔案中設定它。例如，如果要讓 /srv/ftp 目錄可供 rsync 使用，請使用下列組態：

```
gid = nobody
uid = nobody
read only = true
use chroot = no
transfer logging = true
log format = %h %o %f %l %b
log file = /var/log/rsyncd.log

[FTP]
        path = /srv/ftp
        comment = An Example
```

接著使用 rcrsyncdstart 來啟動 rsyncd。在開機程序期間也可以自動啟動 rsyncd。設定的方式有兩種，一是在 YaST 所提供的 runlevel 編輯器中啟用此服務，另一是手動輸入 insservrsyncd 指令。rsyncd 也可以由 xinetd 啟動。然而，只有對很少使用 rsyncd 的伺服器才建議這麼做。

本範例也建立了傾聽所有連接的記錄檔。這個檔案是儲存在 /var/log/rsyncd.log。

這樣才可以從用戶端系統測試傳輸。使用下列指令來執行此動作：

```
rsync -avz sun::FTP
```

此指令會列出伺服器上 /srv/ftp 目錄中所有存在的檔案。這個要求也會記錄在 /var/log/rsyncd.log 記錄檔中。若要啟動實際的傳輸，請提供目標目錄。請使用 。使用 來代表目前的目錄。例如：

```
rsync -avz sun::FTP .
```

依照預設，在使用rsync同步化時不會刪除檔案。如果要強制刪除檔案，就必須加上額外的選項 --delete。若要確保不會刪除較新的檔案，則可以改用 --update 選項。任何產生的衝突都必須手動解決。

39.4.2 如需更多資訊

關於 rsync 的重要資訊，請參閱手冊頁面。指令為 `manrsync` 和 `manrsyncd.conf`。如需關於 rsync 作業原則的技術參考資料，可在 `/usr/share/doc/packages/rsync/tech_report.ps` 中找到。您可以在專案網站 http://rsync.samba.org/ 上找到關於 rsync 的最新消息。

若您想要 Subversion 或其他工具，請下載 SDK。請參閱http://developer.novell.com/wiki/index.php/SUSE_LINUX_SDK。

40 Apache HTTP 伺服器

根據 http://www.netcraft.com/ 的調查結果顯示，Apache HTTP 伺服器 (Apache) 在市面上佔有率已超過 70%，是目前全世界最多人使用的網頁伺服器。由 Apache 軟體基金會 (http://www.apache.org/) 研發的 Apache 可在大部分作業系統上使用。SUSE® Linux Enterprise Server 隨附 Apache 2.2 版。本章將介紹如何安裝、組態設定和設定網頁伺服器，如何使用 SSL、CGI 和其他模組，以及如何排解 Apache 疑難。

40.1 快速入門

本節的說明可協助您快速設定和啟動 Apache。時間。您的身份必須為 root，才能安裝和設定 Apache。

40.1.1 要求

請先確定您已符合下列需求，再設定 Apache 網頁伺服器：

1. 此機器的網路已正確設定。若需有關這個主題的詳細資訊，請參閱 第 30 章 「*基本網路*」 [497 頁]。

2. 此機器的實際系統時間已透過時間伺服器進行同步維護。這是必要動作，因為HTTP通訊協定的部分內容會依據正確時間來運作。如需更多有關這個主題的詳細資訊，請參閱第 32 章「*使用 NTP 進行時間同步化*」[553頁]。

3. 已安裝最新的安全性更新。如果不清楚是否已安裝，請執行「YaST 線上更新」。

4. 防火牆上已開啟預設的網頁伺服器連接埠 (連接埠 80)。針對這點，請將 SUSEFirewall2 設定成允許在外部區域執行「*HTTP 伺服器*」服務。您可以使用 YaST 執行這個動作。如需詳細資訊，請參閱第 43.4.1 節「以 YaST 設定防火牆」[750頁]。

40.1.2 安裝

SUSE Linux Enterprise Server 上預設未安裝 Apache。若要進行安裝，請啟動 YaST，再依序選取「*軟體*」>「*軟體管理*」。接著依序選擇「*過濾器*」>「*模式*」，然後選取「*主要功能*」下方的「*Web 與 LAMP 伺服器*」。請確蓋安裝個別套件，完成此安裝程序。

Apache 會依據預先定義的標準組態來完成安裝，該組態在「預設情況下」(out of the box) 即可執行。此安裝包括多重處理模組 `apache2-prefork` 和 PHP5 模組。如需更多關於各種模組的詳細資訊，請參閱第 40.4 節「安裝、啟用和設定模組」[690頁]。

40.1.3 開始

若要啟動 Apache 並確保其將在開機時自動啟動，請啟動 YaST，再依序選取「*系統*」>「*系統服務 (Runlevel)*」。搜尋「*apache2*」並「*啟用*」該服務。網頁伺服器將立即啟動。使用「*完成*」儲存變更，即可設定系統在開機階段的 runlevel 3 和 runlevel 5 自動啟動 Apache。如需 SUSE Linux Enterprise Server 中 runlevel 的詳細資訊與 YaST runlevel 編輯器的說明，請參閱第 20.2.3 節「使用 YaST 設定系統服務 (Runlevel)」[367頁]。

若要使用外圍程序來啟動 Apache，請執行 `rcapache2 start`。若要確定 Apache 會在開機階段以 runlevel 3 和 5 自動啟動，請使用 `chkconfig -a apache2`。

如果在啟動 Apache 時未收到任何錯誤訊息，即表示網頁伺服器現在已在執行中。啟動瀏覽器，並開啟 `http://localhost/`。這時您應該會看到開頭為「如果您可以看到這段文字，表示您已成功在這部系統中安裝Apache網頁伺服器軟體 的 Apache 測試頁面。」如果此頁面沒有出現，請參閱第 40.8 節「疑難排解」[707頁]。

現在網頁伺服器已經開始執行，您可以加入自己的文件、根據個人需求調整組態，或是安裝模組來新增功能。

40.2 設定 Apache

可以使用下面兩種方法設定 SUSE Linux Enterprise Server 中的 Apache：使用 YaST 或者手動設定。手動設定組態可以提供較詳細的設定，但是缺乏 YaST GUI 提供的方便性。

重要：組態變更

大部分 Apache 組態值在變更之後，必須等到 Apache 重新啟動或是重新載入才能生效。如果是使用 YaST 完成組態設定，而且已針對「*HTTP 服務*」核取「*已啟用*」，Apache 會自動重新啟動或重新載入。如需有關手動重新啟動的詳細資訊，請參閱第 40.3 節「啟動和停止 Apache」[688頁]。大多數的組態變更只需要透過 `rcapache2 reload` 進行重新載入。

40.2.1 手動設定 Apache

手動設定 Apache 是指透過 `root` 使用者身份來編輯純文字組態檔案。

組態檔案

您可以在下列兩個不同位置找到 Apache 組態檔案：

- `/etc/sysconfig/apache2`
- `/etc/apache2/`

/etc/sysconfig/apache2

/etc/sysconfig/apache2 可控制部分的 Apache 全域設定，例如要載入的模組、要包含的其他組態檔案、伺服器應該啟動的旗標，以及應該新增至指令行的旗標。此檔案對每個組態選項都進行了詳細說明，因此本文不予以介紹。針對一般用途的網頁伺服器，在 /etc/sysconfig/apache2 中的設定應該可以符合任何組態需求。

/etc/apache2/

/etc/apache2/ 代管了 Apache 的所有組態檔案。以下各節將說明每個檔案的用途。每個檔案都包含了數個組態選項 (又稱為*指示詞*)。在這些檔案中的每個組態選項都會詳加說明，因此本文將不予以介紹。

Apache 組態檔案的組織方式如下：

```
/etc/apache2/
 |
 |- charset.conv
 |- conf.d/
 |   |
 |   |- *.conf
 |
 |- default-server.conf
 |- errors.conf
 |- httpd.conf
 |- listen.conf
 |- magic
 |- mime.types
 |- mod_*.conf
 |- server-tuning.conf
 |- ssl.*
 |- ssl-global.conf
 |- sysconfig.d
 |   |
 |   |- global.conf
 |   |- include.conf
 |   |- loadmodule.conf . .
 |
 |- uid.conf
 |- vhosts.d
 |   |- *.conf
```

在 /etc/apache2/ 中的 Apache 組態檔案

charset.conv
: 指定不同語言所要使用的字元集。請勿進行編輯。

`conf.d/*.conf`
組態檔案由其他模組新增。這些組態檔案可以依實際需要包含至虛擬主機組態。如需範例，請參閱 `vhosts.d/vhost.template`。若要執行這個動作，您可以為不同的虛擬主機提供不同的模組組合。

`default-server.conf`
使用合理預設值設定所有虛擬主機的全域組態。這時不是變更組態值，而是採用虛擬主機組態覆寫組態值。

`errors.conf`
定義 Apache 處理錯誤的方式。若要自定這些傳送給所有虛擬主機的訊息，請編輯此檔案。另外一種方法是覆寫虛擬主機組態中的這些指示詞。

`httpd.conf`
主要的 Apache 伺服器組態檔案。請勿變更此檔案。此檔案主要包含 Include 陳述式和全域設定。分別為此處列出的每個組態檔案覆寫全域設定。變更虛擬主機組態的主機特定設定 (例如文件根目錄)。

`listen.conf`
繫結 Apache 至特定的 IP 位址與連接埠這裡也可以設定以名稱為基礎之虛擬主機的組態 (請參閱章節「以名稱為基礎的虛擬主機」 [679頁])。

`magic`
mime_magic 模組的資料，此模組可協助 Apache 自動判斷不明檔案的 MIME 類型。請勿進行變更。

`mime.types`
系統已知的 MIME 類型 (實際上是 `/etc/mime.types` 的連結)。請勿進行編輯。如果您需要新增這裡未列出的 MIME 類型，請將它們新增到 `mod_mime-defaults.conf`。

`mod_*.conf`
預設已安裝之模組的組態檔案。如需詳細資訊，請參閱第 40.4 節「安裝、啟用和設定模組」 [690頁]。請注意，選用模組的組態檔案會存放在 `conf.d` 目錄。

`server-tuning.conf`
包含不同 MPM 的組態指示詞 (請參閱第 40.4.4 節「多重處理模組」 [694頁]) 和可控制 Apache 效能的一般組態選項。請在變更此檔案之後為網頁伺服器進行適當測試。

`ssl-global.conf` 和 `ssl.*`
: 全域 SSL 組態和 SSL 憑證資料。如需詳細資訊，請參閱第 40.6 節「設定提供 SSL 的安全網頁伺服器」 [699頁]。

`sysconfig.d/*.conf`
: 自動從 `/etc/sysconfig/apache2` 產生的組態檔案。請勿變更其中任何一個檔案，但可以編輯 `/etc/sysconfig/apache2`。請勿在此目錄中放置其他組態檔案。

`uid.conf`
: 指定要在哪個使用者和群組 ID 之下執行 Apache。請勿進行變更。

`vhosts.d/*.conf`
: 此為虛擬主機組態。此目錄包含採用或不採用SSL之虛擬主機的樣板檔案。在此目錄中，以 `.conf` 做為結尾的每個檔案，都會自動包含至 Apache 組態。如需詳細資訊，請參閱章節「虛擬主機組態」 [678頁]。

虛擬主機組態

「*虛擬主機*」一詞，是形容 Apache 從同一部實體機器提供多個 URI (統一資源識別元，Universal Resource Identifier) 的能力。意思是一部實體機器上的單一網頁伺服器能同時執行數個網域，例如，www.example.com 和 www.example.net。

使用虛擬主機的目的，經常是為了節省管理工作 (只需要維護一部網頁伺服器) 和硬體開銷(不需要將各個網域安裝在專屬伺服器上)。虛擬主機可以使用名稱、IP 或是連接埠作為基礎。

虛擬主機可經由 YaST (請參閱章節「虛擬主機」 [685頁]) 或是手動編輯組態檔案進行設定。依預設，系統會根據 `/etc/apache2/vhosts.d/` 中每部虛擬主機一個組態檔案的設定，為在 SUSE Linux Enterprise Server 中執行的 Apache 做好準備。此目錄中副檔名為 `.conf` 的所有檔案，都會自動包含至組態中。這個目錄會提供虛擬主機的基本樣板 (`vhost.template`，或是適用於提供 SSL 支援之虛擬主機的 `vhost-ssl.template`)。

提示：永遠要建立虛擬主機組態

建議您務必要建立虛擬主機組態檔案，即使網頁伺服器只代管一個網域。建立此檔案時，您不但可以將網域特定組態存放在一個檔案中，還可以隨時恢

復工作環境的基本組態 (只需簡單地移動、刪除或重新命名虛擬主機的組態檔案即可)。同樣地，您應該也要分別為每個虛擬主機建立組態。

<VirtualHost></VirtualHost> 區塊包含要套用到特定網域的資訊。當 Apache 接收到來自定義的虛擬主機的用戶端要求時，就會使用本節所包含的指示詞。幾乎所有指示詞都可以用於虛擬主機網路位置。如需更多有關 Apache 組態指示詞的詳細資訊，請參閱 http://httpd.apache.org/docs/2.2/mod/quickreference.html。

以名稱為基礎的虛擬主機

使用以名稱為基礎的虛擬主機時，每個 IP 位址可以為數個網站提供服務。Apache 會使用用戶端所傳送之 HTTP 標頭中的主機欄位，將要求連接到符合其中一個虛擬主機宣告的 ServerName 項目。如果沒有找到相符的 ServerName，就會預設使用第一個指定的虛擬主機。

NameVirtualHost 指示詞會通知 Apache，要在哪個 IP 位址及哪個連接埠 (選擇性) 上傾聽來自 HTTP 標頭中包含網域名稱的用戶端的要求。這個選項會設定在組態檔案 /etc/apache2/listen.conf。

第一個引數可以是完全合格的網域名稱 (Fully Qualified Domain Name)，但是建議最好使用 IP 位址。第二個引數是連接埠，此引數是可選的。根據預設會使用連接埠 80，而且可以透過 Listen 指示詞設定。

IP 位址和連接埠號碼都可以使用萬用字元 *，來接收所有介面上的要求。IPv6 位址必須包在方括號中。

範例 40.1 *以名稱為基礎的 VirtualHost 項目變化*

```
# NameVirtualHost IP-address[:Port]
NameVirtualHost 192.168.3.100:80
NameVirtualHost 192.168.3.100
NameVirtualHost *:80
NameVirtualHost *
NameVirtualHost [2002:c0a8:364::]:80
```

開啟的 VirtualHost 標籤會將前面經由 NameVirtualHost 宣告的 IP 位址 (或完全合格的網域名稱) 當作以名稱為基礎之虛擬主機的組態引數。先前使用 NameVirtualHost 指示詞宣告的連接埠號碼屬於選擇性。

允許使用萬用字元 * 做為 IP 位址的替代符號。此語法只適用於在 NameVirtualHost * 中結合使用萬用字元的情形。如果是使用 IPv6 位址，該位址就必須用方括號包住。

範例 40.2 *以名稱為基礎的 VirtualHost 指示詞*

```
<VirtualHost 192.168.3.100:80>
  ...
</VirtualHost>

<VirtualHost 192.168.3.100>
  ...
</VirtualHost>

<VirtualHost *:80>
  ...
</VirtualHost>

<VirtualHost *>
  ...
</VirtualHost>

<VirtualHost [2002:c0a8:364::]>
  ...
</VirtualHost>
```

以 IP 為基礎的虛擬主機

此虛擬主機組態替代方法需要為一台機器設定多個IP。一個Apache例項可裝載多個網域，每個網域都會指定不同的 IP。

實體伺服器必須為每部以IP為基礎的虛擬主機設定一個IP位址。當該電腦沒有安裝多張網路卡時，也可以使用虛擬網路介面 (IP 別名)。

下列範例中，Apache 正執行於 IP 為 192.168.3.100 的電腦上，並另外在 IP 192.168.3.101 與 192.168.3.102 上裝載了兩個網域。每部虛擬伺服器都必須具備個別的 VirtualHost 區塊。

範例 40.3 *以 IP 為基礎的 VirtualHost 指示詞*

```
<VirtualHost 192.168.3.101>
  ...
</VirtualHost>

<VirtualHost 192.168.3.102>
  ...
</VirtualHost>
```

這裡出現的 VirtualHost 指示詞僅指定給了除 192.168.3.100 以外的其他介面。如果 192.168.3.100 也有設定 Listen 指示詞，這時就必須建立另一個以 IP 為基礎的虛擬主機，向該介面回應 HTTP 要求，否則將套用預設伺服器組態 (/etc/apache2/default-server.conf) 中的指示詞。

基本虛擬主機組態

每個虛擬主機組態中至少要出現下列指示詞，才能設定虛擬主機。如需了解更多選項的詳細資訊，請參閱 /etc/apache2/vhosts.d/vhost.template。

ServerName
: 完整網域名稱，其下是應該要建立位址的主機。

DocumentRoot
: 目錄路徑，Apache 會從此路徑為此主機提供檔案。基於安全性考量，存取整個檔案系統是預設禁止的動作，所以您必須明確解除鎖定這個位在 Directory 容器中的目錄。

ServerAdmin
: 伺服器管理員的電子郵件地址. 這個地址可顯示在 Apache 建立的錯誤頁面 (舉例說明)。

ErrorLog
: 此虛擬主機的錯誤記錄檔案。雖然沒必要為每個虛擬主機分別建立錯誤記錄檔案，但是多數人會這樣做，以便更加容易地進行除錯。/var/log/apache2/ 是 Apache 保存記錄檔案的預設目錄。

CustomLog
: 此虛擬主機的存取記錄檔案。雖然沒必要為每個虛擬主機分別建立存取記錄檔案，但是多數人會這樣做，以便分別為每個主機分析存取統計資料。/var/log/apache2/ 是 Apache 保存記錄檔案的預設目錄。

正如前面所述，存取整個檔案系統已因安全性考量而預設為禁止動作。因此，請將 Apache 要處理之檔案所在的目錄明確解除鎖定 — 例如 DocumentRoot。

```
<Directory "/srv/www/www.example.com/htdocs">
  Order allow,deny
  Allow from all
</Directory>
```

此完整組態看起來如下：

範例 40.4　基本 VirtualHost 組態

```
<VirtualHost 192.168.3.100>
  ServerName www.example.com;
  DocumentRoot /srv/www/www.example.com/htdocs
  ServerAdmin webmaster@example.com
  ErrorLog /var/log/apache2/www.example.com_log
  CustomLog /var/log/apache2/www.example.com-access_log common
  <Directory "/srv/www/www.example.com/htdocs">
    Order allow,deny
    Allow from all
  </Directory>
</VirtualHost>
```

40.2.2 使用 YaST 設定 Apache

若要使用 YaST 來設定您的網頁伺服器，請啟動 YaST 並依序選取「*網路服務*」>「*HTTP 伺服器*」。第一次啟動模組時，HTTP 伺服器精靈會啟動，提示您如何決定有關伺服器管理的一些基本設定。完成精靈之後，每當您呼叫「*HTTP 伺服器*」模組時，就會啟動章節「HTTP 伺服器組態」[686頁]中介紹的對話方塊。

HTTP 伺服器精靈

HTTP 伺服器精靈包含有五個步驟。在最後一個步驟的對話方塊中，您可以進入進階組態模式以進行更多特定的設定。

網路設備選擇

在此，您可以指定 Apache 用來傾聽內送要求的網路介面和連接埠。您可以選取任何現有網路介面及其 IP 位址的組合。若連接埠 (連接埠隸屬以下三種：已知埠、註冊埠和動態或私人埠) 不供其他服務使用，則皆可供您使用。預設設定為傾聽所有在連接埠 80 上的網路介面 (IP 位址)。

核取「*開啟選取埠的防火牆*」選項，可開啟防火牆中網頁伺服器傾聽的連接埠。若要使網頁伺服器在網路 (包括 LAN、WAN 或公用網際網路) 上為可用狀態，請核取此選項。但是，若是處於測試階段且不須由外部網路存取網頁伺服器，就可以關閉此連接埠。

按一下「*下一步*」繼續設定。

模組

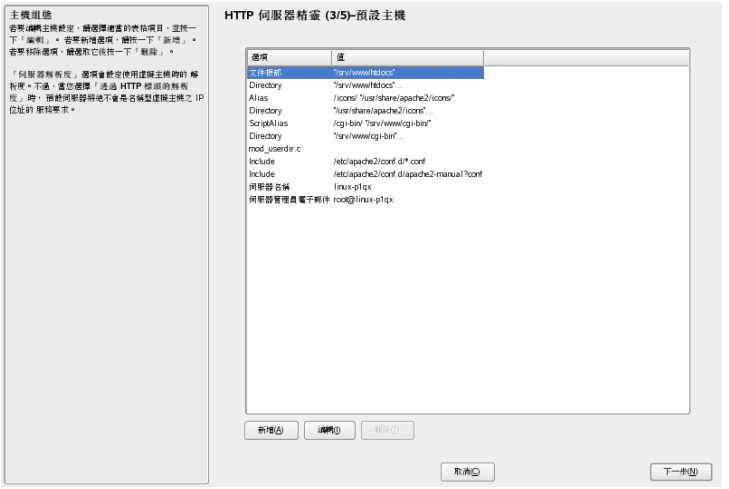
這個「*模組*」組態選項可以用來啟用、或停用網頁伺服器需支援的程序檔語言。如需有關啟用或停用其他模組的詳細資訊，請參閱章節「伺服器模組」[687頁]。按一下「*下一步*」，繼續進行下一個對話方塊。

預設主機

此選項與預設網頁伺服器相關。正如章節「虛擬主機組態」[678頁]內容所述，Apache 可以在一部實體機器上提供多個虛擬主機。組態檔案中第一個宣告的虛擬主機通常會被視為*預設主機*。每部虛擬主機都會繼承預設主機的組態。

若要編輯主機設定(又稱為*指示詞*)，請在表格中選擇適當項目，然後按一下「*編輯*」。若要新增指示詞，請按一下「*新增*」。若要刪除指示詞，請選取該指示詞，然後按一下「*刪除*」。

圖形 40.1 *HTTP 伺服器精靈：預設主機*



這是伺服器預設值的清單：

`Document Root`
: 目錄路徑，Apache 會從此路徑為此主機提供檔案。`/srv/www/htdocs` 是預設位置。

Alias
在配合 Alias 指示詞時，URL 可以映射到實體檔案系統位置。這表示某個路徑即使不在檔案系統的 Document Root 中，仍可藉由該路徑別名的 URL 進行存取。

預設的 SUSE Linux Enterprise Alias /icons 會指向 /usr/share/apache2/icons，做為目錄索引檢視中所顯示的 Apache 圖示。

ScriptAlias
功能相似於 Alias 指示詞，ScriptAlias 指示詞可以將 URL 映射到檔案系統位置。兩者差別在於 ScriptAlias 可以將目標目錄指定作為 CGI 位置，表示該 CGI 程序檔必須在該位置執行。

Directory
使用 Directory 設定時，您可以限制只會套用到特定目錄的組態選項群組。

在此可以設定 /usr/share/apache2/icons 和 /srv/www/cgi-bin 等目錄的存取和顯示選項。其中預設值應該不需要進行改變。

Include
使用 Include 可以指定其他的組態檔案。預先設定的 Include 指示詞有兩個：/etc/apache2/conf.d/ 為外部模組隨附之組態檔所在的目錄。使用此指示詞時，此目錄中所有以 .conf 結尾的檔案都會包含在內。使用第二個指示詞時，/etc/apache2/conf.d/apache2-manual.conf，即 apache2-manual 組態檔將包含在內。

Server Name
這個項目可以指定用戶端用來聯絡網頁伺服器的預設 URL。請使用完全合格的網域名稱 (FQDN) 來連接 http://*FQDN*/ 的網頁伺服器或其 IP 位址。您不能在此選擇任意名稱 — 採用此名稱的伺服器必須是「已知」的。

Server Administrator E-Mail
伺服器管理員的電子郵件地址. 這個地址可顯示在 Apache 建立的錯誤頁面 (舉例說明)。

完成設定「*預設主機*」步驟後，請按一下「*下一步*」，繼續下一個組態步驟。

虛擬主機

在此步驟中，精靈會顯示已完成設定之虛擬主機的清單(請參閱章節「虛擬主機組態」[678頁])。如果在啟動 YaST HTTP 精靈之前尚未進行手動變更，則不會顯示虛擬主機。

若要新增主機，請按一下「*新增*」開啟對應的對話方塊，並在其中輸入主機的基本資訊。「*伺服器識別*」中包括伺服器名稱、伺服器內容根目錄 (`DocumentRoot`) 和管理員電子郵件。「*伺服器解析*」可用來決定主機的識別方式 (以名稱為基礎或是以 IP 為基礎)。透過「*變更虛擬主機 ID*」指定名稱或 IP 位址

按一下「*下一步*」，繼續進入虛擬主機組態對話方塊的第二部分。

在虛擬主機組態對話方塊的第二部分中，您可以指定是否要啟用 CGI 程序檔、以及這些程序檔要使用哪個目錄。您也可以在此啟用 SSL。如果執行了這個動作，您就必須同時指定憑證的路徑。如需有關 SSL 和憑證的詳細資訊，請參閱第 40.6.2 節「設定提供 SSL 的 Apache」[703頁]。使用「*目錄索引*」選項時，您可以指定當用戶端要求目錄 (依預設是指 index.html) 時要顯示哪個檔案。如果您要變更這個設定，請新增一個或多個檔名 (以空格分隔)。使用「*啟用公用HTML*」時，使用者公用目錄 (`~user/public_html/`) 的內容就可由伺服器公開於 `http://www.example.com/~user`。

重要：建立虛擬主機

您不能在此隨意新增虛擬主機。如果使用以名稱為基礎的虛擬主機，就必須在網路上解析每部主機名稱。如果是使用以 IP 為基礎的虛擬主機，每個可用 IP 位址就只能指派一部主機。

摘要

這是精靈的最後一個步驟。您可以在此處決定 Apache 伺服器啟動的方式和時間：開機時啟動或手動啟動。同時可檢視目前已完成之組態的簡短摘要。如果您接受目前設定，請按一下「*完成*」以完成組態設定。如果要變更某些設定，請按「*上一步*」，直到出現您需要的對話方塊。按一下「*HTTP 伺服器進階組態*」便可開啟章節「HTTP 伺服器組態」[686頁]所介紹的對話方塊。

圖形 40.2　HTTP 伺服器精靈：摘要

HTTP 伺服器組態

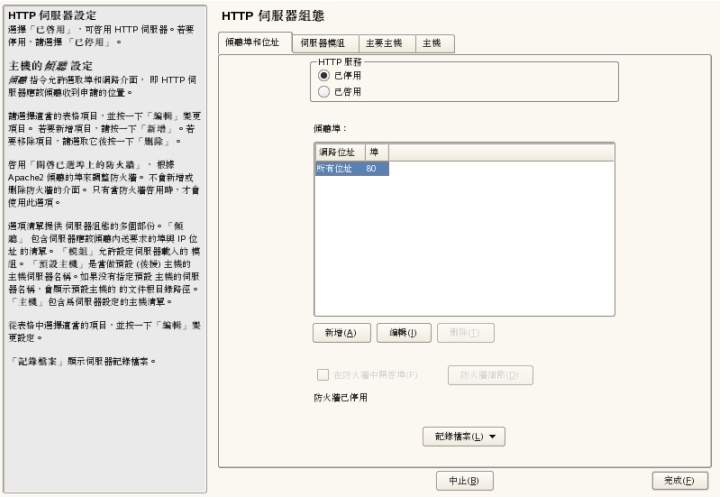
「*HTTP 伺服器組態*」對話方塊可提供比精靈更多的組態調整 (精靈 只會在第一次設定網頁伺服器時執行)。其中包含下列要介紹的四個索引標籤。在此變更的任何組態選項都無法立即生效 — 您必須先按一下「*完成*」進行確認之後，它們才會生效。按一下「*取消*」，便可保留組態模組不變並捨棄您的變更。

傾聽連接埠和位址

在「*HTTP 服務*」中，選取執行 (「*啟用*」) 或停止 (「*停用*」) Apache。在「*傾聽埠*」中，「*新增*」、「*編輯*」或「*刪除*」可透過其使用伺服器的位址和連接埠。預設設定是傾聽在連接埠 80 上的所有介面。您應核取「*開啟已選埠上的防火牆*」，否則就不能從外部連接網頁伺服器。但是，若是處於測試階段且不須由外部網路存取網頁伺服器，就可以關閉此連接埠。

使用「*記錄檔案*」時，可檢視存取記錄或錯誤記錄。測試組態時此選項非常有用。記錄檔案會在單獨的視窗中開啟，在此您還可以重新啟動或重新載入網頁伺服器 (如需詳細資訊，請參閱第 40.3 節「啟動和停止 Apache」[688頁])。這些指令會立即生效。

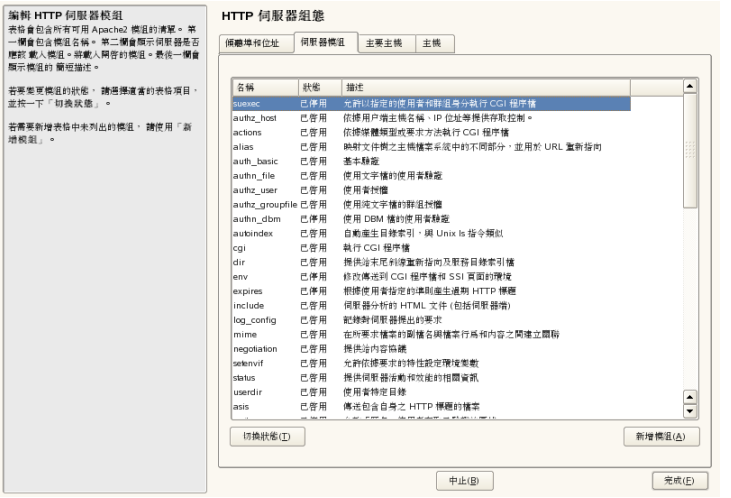
圖形 40.3　HTTP 伺服器組態：傾聽連接埠和位址



伺服器模組

您可以按一下「*切換狀態*」，變更 Apache2 模組的狀態 (啟用或停用)。按一下「*新增模組*」，可新增已經安裝但是未列出的新模組。若要更進一步認識模組，請參閱第 40.4 節「安裝、啟用和設定模組」[690頁]。

圖形 40.4　HTTP 伺服器組態：伺服器模組



主要主機

這些對話方塊相同於前面已介紹過的對話方塊。請參閱章節「預設主機」[683頁]和章節「虛擬主機」[685頁]。

40.3 啟動和停止 Apache

如果是透過 YaST 來設定 (請參閱第 40.2.2 節「使用 YaST 設定 Apache」[682頁])，Apache 會於開機階段的 runlevel 3 和 5 啟動，在 runlevel 0、1、2 和 6 停止。您可以使用 YaST 的 runlevel 編輯器或指令行工具 `chkconfig` 來變更此行為。

若要啟動、停止或操作執行中之系統上的 Apache，請使用 init 程序檔 `/usr/sbin/rcapache2` (如需 init 程序檔的一般資訊，請參閱第 20.2.2 節「Init 程序檔」[363頁])。`rcapache2` 指令會接受下列參數：

`start`
　啟動尚未執行的 Apache。

`startssl`
若採用 SSL 支援的 Apache 不在執行中，則將其啟動。如需更多有關 SSL 支援的詳細資訊，請參閱第 40.6 節「設定提供 SSL 的安全網頁伺服器」[699頁]。

`stop`
透過終止父處理程序來停止 Apache。

`restart`
停止 Apache，然後重新啟動。啟動之前並未在執行中的網頁伺服器。

`try-restart`
僅停止之前已在執行中的 Apache，然後重新啟動。

`reload 或 graceful`
通知所有 Apache 衍生處理程序在關機之前先完成各自的要求，以停止網頁伺服器。當每個處理程序都結束之後，就會取代成新開始的處理程序，最後完成「重新啟動」 Apache。

提示

`rcapache2 reload` 是在生產環境中較受歡迎的 Apache 重新啟動方法 (例如用於啟用組態的變更)，因為這種方法可以讓所有用戶端在不會導致連線中斷的情況下取得服務。

`configtest`
在不影響執行中之網頁伺服器的情況下，檢查組態檔案的語法。因為這項檢查會在每次伺服器啟動、重新載入或重新啟動時強制進行，所以通常並不需要明確執行該測試 (如果這時有找到組態錯誤，網頁伺服器就不會完成啟動、重新載入或是重新啟動)。

`probe`
查探重新載入的重要性 (檢查組態是否有變更) 並建議 `rcapache2` 指令的必要引數。

`server-status 和 full-server-status`
分別傾印簡要或完整的狀態畫面。要求必須安裝 lynx 或 w3m 並啟用 mod_status 模組。除此之外，`status` 必須加入 `/etc/sysconfig/apache2` 檔案的 `APACHE_SERVER_FLAGS`。

提示：其他旗標

如果您為 `rcapache2` 指定其他旗標，這些旗標就會傳遞通過網頁伺服器。

40.4 安裝、啟用和設定模組

Apache 軟體屬於模組化設計：除了部分核心任務，其餘所有功能皆由模組處理。這種進步幅度很大，甚至連 HTTP 都是由模組 (http_core) 處理。

Apache 模組可以在建立階段編譯成 Apache 二進位檔案，或是在執行階段動態載入。如需了解如何動態載入模組的詳細資訊，請參閱第 40.4.2 節「啟用和停用」[691頁]。

Apache 模組可以分成四種不同類別：

基本模組
: 基本模組會依預設編譯到 Apache。SUSE Linux 中的 Apache 只有編譯 mod_so (必須載入其他模組) 和 http_core。所有其他模組都可透過共享物件方式提供：這些模組會在 Runtime 時進行包含 (Include)，而不用包含至伺服器二進位檔案本身。

延伸模組
: 一般說來，Apache 軟體套件會包含標示為延伸的模組，但是通常不會使用靜態方式將這些模組編譯到伺服器中。在 SUSE Linux Enterprise Server 中，這類模組以共享物件方式提供，並可在執行階段載入到 Apache。

外部模組
: 標示為外部的模組不會包含於 Apache 正式發行版本中。SUSE Linux Enterprise Server 提供了幾種可以立即使用的外部模組。

多重處理模組
: MPM 會負責接收和處理網頁伺服器所收到的要求，因此屬於網頁伺服器軟體的核心部分。

40.4.1 模組安裝

如果您已經依照安裝 Apache 的預設方式 (請參閱第 40.1.2 節「安裝」 [674頁]) 安裝 Apache，則所有的基礎模組和延伸模組、多重處理模組 Prefork MPM，以及外部模組 mod_php5 和 mod_python 都會隨之安裝。

您可以啟動 YaST 並依序選擇「*軟體*」>「*軟體管理*」，以安裝其他外部模組。現在，請依序選擇「*過濾器*」>「*搜尋*」，然後搜尋 *apache*。在其他套件之中，此結果清單會包含所有可用的外部 Apache 模組。

40.4.2 啟用和停用

使用 YaST 時，您可以使用章節「HTTP 伺服器精靈」 [682頁]說明的模組組態來啟用或停用程序檔語言模組 (PHP5、Perl、Python)。所有其他模組都可以依據章節「伺服器模組」 [687頁]說明步驟來啟用或停用。

如果您偏好手動啟用或停用模組，請分別使用 `a2enmod mod_foo` 或 `a2dismod mod_foo` 指令。`a2enmod -l` 會輸出目前所有的使用中模組清單。

重要：包含外部模組的組態檔案

如果您已經手動啟用外部模組，請確定將其組態檔案載入至所有的虛擬主機組態。外部模組的組態檔案會存放在 `/etc/apache2/conf.d/` 之下，而且不會預設載入。如果您需要在每個虛擬主機上載入相同模組，您可以含入此目錄中的 `*.conf`。另一種做法是含入個別檔案。如需取得範例說明，請參閱 `/etc/apache2/vhost.d/vhost.template`。

40.4.3 基本和延伸模組

Apache 說明文件中詳細介紹了所有的基本模組和延伸模組。本文件只提供最重要模組的概要說明。如需關於每個模組的詳細資訊，請參閱 `http://httpd.apache.org/docs/2.2/mod/`。

mod_actions
: 提供在需要特定 MIME 類型 (例如 `application/pdf`)、具有特定副檔名的檔案 (例如 `.rpm`) 或特定要求方法 (例如 `GET`) 時執行程序檔的方法。這是預設啟用的模組。

mod_alias
: 提供 `Alias` 和 `Redirect` 指示詞，供您用來將 URL 映射至特定目錄 (`Alias`) 或將所要求的 URL 重新導向至其他位置。這是預設啟用的模組。

mod_auth*
: 驗證模組可提供不同的驗證方式：使用 mod_auth_basic 的基本驗證，或是使用 mod_auth_digest 的摘要驗證。Apache 2.2 的摘要驗證方式仍屬實驗性質。

  mod_auth_basic 和 mod_auth_digest 必須與驗證提供者模組 mod_authn_* (例如適用於以文字檔案為基礎之驗證的 mod_authn_file) 以及驗證模組 mod_authz_* (例如適用於使用者驗證的 mod_authz_user) 結合。

  如需更多有關此主題的詳細資訊，請參閱 `http://httpd.apache.org/docs/2.2/howto/auth.html` 的「驗證 HOWTO」。

mod_autoindex
: Autoindex 會在沒有任何索引檔案 (例如，`index.html`) 出現時產生目錄清單。這些索引的外觀可加以設定。這是預設啟用的模組。然而，目錄清單預設由 `Options` 指示詞停用 — 這會覆寫虛擬主機組態的此項設定。這個模組的預設組態檔案會存放在 `/etc/apache2/mod_autoindex-defaults.conf`。

mod_cgi
: 執行 CGI 程序檔時必須使用 mod_cgi。這是預設啟用的模組。

mod_deflate
: 使用這個模組時，Apache 可以設定成即時壓縮成指定檔案類型之後，再進行傳送。

mod_dir
: mod_dir 可提供 `DirectoryIndex` 指示詞，供您用來設定當要求目錄 (預設是 `index.html`)時要自動傳遞哪類檔案。它也會提供當目錄要求沒有包含末尾斜線時自動重新導向到正確 URL 的功能。這是預設啟用的模組。

mod_env
: 控制傳遞給 CGI 程序檔或 SSI 頁面的環境。可以在呼叫 httpd 程序的外圍程序中設定、取消設定或傳遞環境變數。這是預設啟用的模組。

mod_expires
使用 mod_expires 時，您可以傳送 `Expires` 標頭，控制代理和瀏覽器快取記憶體重新整理文件的頻率。這是預設啟用的模組。

mod_include
mod_include 可讓您使用 Server Side Include (SSI)，這項工具會提供動態產生 HTML 頁面的基本功能。這是預設啟用的模組。

mod_info
可透過 http://localhost/server-info/ 提供伺服器組態的綜合綜覽。基於安全性考量，您應該永遠限制這個 URL 的存取權限。依預設，只有 `localhost` 允許存取這個 URL。mod_info 會在 `/etc/apache2/mod_info.conf` 完成設定。

mod_log_config
使用這個模組時，您可以設定 Apache 記錄檔案的外觀。這是預設啟用的模組。

mod_mime
此 MIME 模組會依據傳送檔案的副檔名來檢查其是否包含正確的 MIME 標頭 (例如 HTML 文件應為 `text/html`)。這是預設啟用的模組。

mod_negotiation
內容協商 (Content Negotiation) 所需的模組。如需更多詳細資訊，請參閱 `http://httpd.apache.org/docs/2.2/content-negotiation.html`。這是預設啟用的模組。

mod_rewrite
可提供 mod_alias 的功能，但是具備更多功能和使用彈性。使用 mod_rewrite 時，您可以依據多重的規則、要求標頭和其他條件來重新導向 URL。

mod_setenvif
根據用戶端的要求設定環境變數，如用戶端傳送的瀏覽器字串或用戶端的 IP 位址。這是預設啟用的模組。

mod_speling
mod_speling 會嘗試自動修正 URL 中出現的打字錯誤，例如大小寫錯誤。

mod_ssl
: 啟用網頁伺服器和用戶端之間的加密連線。如需詳細資料，請參閱 第 40.6 節「設定提供 SSL 的安全網頁伺服器」[699頁]。這是預設啟用的模組。

mod_status
: 可透過 http://localhost/server-status/ 提供有關伺服器活動及效能的資訊。基於安全性考量，您應該永遠限制這個 URL 的存取權限。依預設，只有 `localhost` 允許存取這個 URI。mod_status 會在 `/etc/apache2/mod_status.conf` 完成設定。

mod_suexec
: mod_suexec 可讓您以不同使用者和群組身份來執行 CGI 程序檔。這是預設啟用的模組。

mod_userdir
: 啟用在 *~user/* 之下提供使用者特定目錄。在組態中必須指定 `UserDir` 指示詞。這是預設啟用的模組。

40.4.4 多重處理模組

SUSE Linux Enterprise Server 提供了兩種不同的多重處理模組 (MPM) 搭配 Apache 使用。

Prefork MPM

Prefork MPM 會實作未產生執行緒、正在進行 Prefork 的網頁伺服器。這個模組會使得網頁伺服器產生類似 Apache 版本 1.x 的行為，也就是透過衍生出獨立的子處理程序，將每個要求獨立分開並加以處理。這樣發生問題的要求就不會影響其他要求，進而避免網頁伺服器出現鎖定現象。

雖然透過這種以處理程序為主的方法可以提供穩定性，但是比起 Worker MPM，Prefork MPM 會耗用較多的系統資源。Unix 作業系統會將 Prefork MPM 當作預設 MPM。

重要：本文件的 MPM

本文件會假設 Apache 是使用 Prefork MPM。

Worker MPM

Worker MPM 會提供多執行緒網頁伺服器。執行緒是「輕量級」的處理程序。執行緒和處理程序相比的優點是，它消耗的資源較少。Worker MPM 不只會衍生子處理程序，它還可使用執行緒和伺服器處理程序，來為要求提供服務。完成 Prefork 的子處理程序屬於多執行緒。這種方法因為耗用比 Prefork MPM 更少的系統資源，因此可以提高 Apache 的執行效能。

一個主要缺點就是 Worker MPM 的穩定性：當某執行緒毀損時，處理程序的所有執行緒都會受到影響。最嚴重的情況下，甚至還會造成伺服器當機。尤其是在負載量高的 Apache 上使用通用閘道介面 (Common Gateway Interface, CGI) 時，可能就會因執行緒無法與系統資源進行通訊而產生內部伺服器錯誤。在 Apache 上使用 worker MPM 的另外一點爭議，就是並非所有可用的 Apache 模組都能安全地使用執行緒，這樣就無法配合 worker MPM 使用。

警告：搭配 MPM 使用 PHP 模組

並非所有可用的 PHP 模組都是安全執行緒。因此最好不要搭配 worker MPM 來使用 mod_php。

40.4.5 外部模組

在此處尋找 SUSE Linux Enterprise Server 隨附的所有外部模組清單。在列出目錄中找出模組的說明文件。

mod-apparmor
: 為 Apache 新增支援，給由模組 (如 mod_php5 和 mod_perl) 處理的個別 CGI 程序檔提供 Novell AppArmor 限制。

  套件名稱：`apache2-mod_apparmor`
  詳細資訊：*Novell AppArmor Administration Guide* (↑*Novell AppArmor Administration Guide*)

mod_perl
: mod_perl 可讓您使用內嵌解譯器來執行 Perl 程序檔。內嵌在伺服器的常駐解譯器可以避免因啟動外部解譯器造成的負荷，以及在 Perl 啟動階段時降低速度。

  套件名稱：`apache2-mod_perl`

組態檔案：/etc/apache2/conf.d/mod_perl.conf
詳細資訊：/usr/share/doc/packages/apache2-mod_perl

mod_php5
PHP 是一種伺服器端、跨平台式的 HTML 內嵌程序檔語言。

套件名稱：apache2-mod_php5
組態檔案：/etc/apache2/conf.d/php5.conf
詳細資訊：/usr/share/doc/packages/apache2-mod_php5

mod_python
mod_python 允許在 Apache HTTP 伺服器中內嵌 Python，以便大幅提高效能和增加網頁應用程式的設計彈性。

套件名稱：apache2-mod_python
詳細資訊：/usr/share/doc/packages/apache2-mod_python

40.4.6 編譯

Apache 允許進階使用者編寫自定模組進行延伸。若要開發 Apache 模組或編譯協力廠商模組，除了相對應開發工具外，還需要套件 apache2-devel。apache2-devel 也包含了 apxs2 工具，這是在編譯 Apache 其他模組時，需要用到的工具。

apxs2 可以從原始程式碼進行模組編譯和安裝(其中包括必要的組態檔案變更)，並建立可於 Runtime 載入 Apache 的*動態共享物件* (DSO)。

apxs2 二進位檔案位在 /usr/sbin 下方：

- /usr/sbin/apxs2 — 適合用來建立可配合任何 MPM 使用的延伸模組。安裝位置是 /usr/lib/apache2。
- /usr/sbin/apxs2-prefork — 適合用於 Prefork MPM 模組。安裝位置是 /usr/lib/apache2-prefork。
- /usr/sbin/apxs2-worker — 適合用於 Worker MPM 模組。

apxs2 會安裝可供所有 MPM 使用的模組。其他兩個程式會安裝模組，使它們只可以用於各自的 MPM。apxs2 會將模組安裝到 /usr/lib/apache2，而

`apxs2-prefork` 和 `apxs2-worker` 會將模組安裝到 `/usr/lib/apache2-prefork` 或 `/usr/lib/apache2-worker` 中。

使用 `cd /path/to/module/source; apxs2 -cia mod_foo.c` 指令從原始程式碼來安裝和啟用模組 (`-c` 負責編譯模組、`-i` 負責安裝，而 `-a` 負責啟用)。如需有關 `apxs2` 的其他選項資訊，請參閱 `apxs2(1)` man 頁面。

40.5 啟用 CGI 程序檔

Apache 的通用閘道介面 (CGI) 可讓您使用程式或程序檔 (通常指 CGI 程序檔) 建立動態內容。CGI 程序檔可以用任何程式設計語言來編寫。通常會使用類似 Perl 或 PHP 等程式檔設計語言。

若要啟用 Apache 來傳送 CGI 程序檔建立的內容，就必須啟用 mod_cgi 模組。這時也需要用到 mod_alias 模組。這兩種都是預設啟用的模組。如需啟用模組的詳細資訊，請參閱第 40.4.2 節「啟用和停用」 [691頁]。

警告：CGI 安全性

允許伺服器執行 CGI 程序檔會產生潛在的安全性弱點。請參考第 40.7 節「避免安全性問題」 [705頁]，以取得其他資訊。

40.5.1 Apache 組態

在 SUSE Linux Enterprise Server 中，CGI 程序檔只能在 `/srv/www/cgi-bin/` 目錄中執行。這個位置已設定用來執行 CGI 程序檔。如果您已經建立虛擬主機組態 (請參閱章節「虛擬主機組態」 [678頁]) 並想要將程序檔放置到主機特定的目錄，則必須解除鎖定和設定此目錄。

範例 40.5　VirtualHost CGI 組態

```
ScriptAlias /cgi-bin/ "/srv/www/www.example.com/cgi-bin/"❶

<Directory "/srv/www/www.example.com/cgi-bin/">
 Options +ExecCGI❷
 AddHandler cgi-script .cgi .pl❸
 Order allow,deny❹
 Allow from all
</Directory>
```

❶ 告知 Apache 依照 CGI 程序檔方式來處理位在這個目錄中的所有檔案。

❷ 啟用 CGI 程序檔執行

❸ 告知伺服器依照 CGI 程序檔方式來處理包含 .pl 和 .cgi 副檔名的檔案。依據個人需要來加以調整。

❹ `Order` 和 `Allow` 指示詞可控制 Allow 和 Deny 指示詞在評估時的預設存取狀態和順序。在這個範例中，Apache 會先評估「deny」陳述式，接著才評估「allow」陳述式，並啟用可從任何位置存取。

40.5.2 執行程序檔範例

CGI 程式設計不同於「一般」程式設計；因為 CGI 程式和程序檔的最前面必須是 MIME-Type 標頭，例如 `Content-type: text/html`。這個標頭會傳送到用戶端，使其了解所接收內容的類型。其次，程序檔的輸出必須是用戶端(通常是指 Web 瀏覽器) 可了解的內容 — 舉例來說，在多數情況下是指 HTML、純文字或影像。

Apache 套件會在 `/usr/share/doc/packages/apache2/test-cgi` 提供簡單的測試程序檔。這個程序檔會以純文字方式輸出部分環境變數的內容。請將這段程序檔複製到 `/srv/www/cgi-bin/` 或是虛擬主機的程序檔目錄 (`/srv/www/example.com_cgi-bin/`)，並將其命名為 `test.cgi`。

可由網頁伺服器存取的檔案，應該屬於 `root` 使用者所有 (如需更多詳細資訊，請參閱第 40.7 節「避免安全性問題」[705頁])。因為網頁伺服器可由不同使用者身份執行，所以 CGI 程序檔必須具備可供全球執行和可供全球讀取等特性。變更 CGI 目錄和使用 `chmod 755 test.cgi` 指令，便可套用適當的許可權。

現在，請呼叫 `http://localhost/cgi-bin/test.cgi` 或 `http://www.example.com/cgi-bin/test.cgi`。這時應該會顯示「CGI/1.0 測試程序檔報告」。

40.5.3 疑難排解

如果這時沒有顯示測試程式的輸出結果，而是出現錯誤訊息，請檢查下列項目：

CGI 疑難排解

- 您是否有在變更組態之後重新載入伺服器？請使用 `rcapache2 probe` 進行檢查。
- 您是否已正確設定自定的 CGI 目錄 (若有的話)？如果您不確定，請在預設的 CGI 目錄 `/srv/www/cgi-bin/` 中測試此程序檔，並使用 `http://localhost/cgi-bin/test.cgi` 進行呼叫。
- 檔案許可權是否正確？請切換至 CGI 目錄，並執行 `ls -l test.cgi`。此測試輸出開頭應該是

  ```
  -rwxr-xr-x  1 root root
  ```

- 請確定程序檔沒有包含任何程式設計錯誤。如果您沒有變更 test.cgi 應該就不會出淚錯誤，但是如果您是使用自己的程式，請務必確蓋這些程式沒有包含任何程式設計錯誤。

40.6 設定提供 SSL 的安全網頁伺服器

如果網頁伺服器和用戶端之間會傳輸敏感性資料 (例如信用卡資料)，這時最好要提供帶驗證的安全加密連線。mod_ssl 會使用安全通訊端層 (Secure Sockets Layer, SSL) 及傳輸層安全性 (Transport Layer Security, TLS) 通訊協定，為用戶端和網頁伺服器之間的 HTTP 通訊提供強式加密。使用 SSL/TSL 時，網頁伺服器和用戶端之間就會建立私人連線。如此便可確保資料完整性，使用戶端和伺服器端可以彼此進行驗證。

為了完成這個目的，伺服器會在回覆任何 URL 要求之前，先傳送可證明伺服器有效身份之相關資訊的 SSL 憑證。如此即可確保該伺服器是此通訊的唯一正確端點。此外，該憑證會在用戶端和伺服器端建立加密連線，以便在沒有洩漏敏感、純文字內容的風險情況下傳輸資訊。

mod_ssl 本身不會實作 SSL/TSL 通訊協定，而是扮演 Apache 和 SSL 程式庫之間的介面。在 SUSE Linux Enterprise Server 中是使用 OpenSSL 程式庫。OpenSSL 會自動隨 Apache 完成安裝。

使用mod_ssl搭配Apache的最明顯特徵，就是URL的字首都會加上`https://`，而不是`http://`。

40.6.1 建立 SSL 憑證

為了搭配使用SSL/TSL與網頁伺服器，您必須建立SSL憑證。網頁伺服器和用戶端在彼此驗證時要用到這項憑證，以便讓任一方可以清楚識別對方。為了確保憑證的完整性，其必須由每位使用者信任的一方加以簽章。

您可以建立下列三種類型的憑證：僅供測試使用的「虛擬」憑證、供已定義信任圈使用者使用的自我簽發憑證，以及由獨立、公開的憑證授權機構(CA)簽發的憑證。

憑證建立基本上可分為兩個步驟。首先產生憑證授權機構的私密金鑰，接著再使用該金鑰簽發伺服器憑證。

提示：如需更多資訊

若要更進一步蓋識SSL/TSL的概念和定義，請參閱http://httpd.apache.org/docs/2.2/ssl/ssl_intro.html。

建立「虛擬」憑證

產生虛構憑證的步驟很簡單。您只要呼叫程序檔`/usr/bin/gensslcert`，便可建立此類憑證，或是覆寫下列檔案：

- `/etc/apache2/ssl.crt/ca.crt`
- `/etc/apache2/ssl.crt/server.crt`
- `/etc/apache2/ssl.key/server.key`
- `/etc/apache2/ssl.csr/server.csr`

`ca.crt`的複製本也會放在`/srv/www/htdocs/CA.crt`提供下載。

重要

虛構憑證絕對不可用於生產環境系統。這類憑證只能用於測試目的。

建立自我簽發憑證

如果您要設定供內部網路 (Intranet)、或已定義信任圈使用者使用的安全網頁伺服器，透過您本身憑證授權機構(CA) 簽發憑證就可有效符合此時的憑證需求。

自我簽署憑證的建立程序分為互動的九個步驟。請切換至目錄 `/usr/share/doc/packages/apache2`，並執行下列指令 `./mkcert.sh make --no-print-directory /usr/bin/openssl /usr/sbin/ custom`。請勿嘗試從這個目錄外面執行這個指令。這個程式會提供一系列提示訊息，其中有部分提示需要使用者輸入。

過程 40.1 *使用 mkcert.sh 建立自我簽發憑證*

1 決定憑證所使用的簽章演算法

選擇 RSA (R，此為預設選項)，因為有些早期瀏覽器無法使用 DSA。

2 產生 CA 的 RSA 私密金鑰 (1024 位元)

這時不須進行任何互動。

3 產生 CA 的 X.509 憑證簽發要求

在此建立CA的可辨識名稱。這時系統會要求您回答一些問題，例如國家/地區名或組織名稱。請輸入有效資料，因為您在此輸入的每項資料將來都會顯示在憑證中。您不需要回答每個問題。如果有不適用的問題或是您希望保留空白，請使用「.」。通用名稱是指 CA 本身的名稱 — 請選擇明顯的名稱，例如 *My company* CA。

4 產生 CA 本身簽發的 X.509 憑證

選擇憑證版本 3 (預設選項)。

5 產生 SERVER 的 RSA 私密金鑰 (1024 位元)

這時不須進行任何互動。

6 產生 SERVER 的 X.509 憑證簽發要求

在此建立伺服器金鑰的可識別名稱。這時要回答的問題與建立 CA 可識別名稱時回答的問題幾乎一樣。在此輸入的資料會套用到網頁伺服器，因此不需要和 CA 資料完全相同 (例如，伺服器可能會出淚任意位置)。

重要：選取通用名稱

您在此輸入的通用名稱必須是安全伺服器的完全合格主機名稱 (例如，www.example.com)。否則瀏覽器會在存取網頁伺服器時發出警告，表示憑證與伺服器不相符。

7 產生本身 CA 簽發的 X.509 憑證

選擇憑證版本 3 (預設選項)。

8 使用密碼片語為 CA 的 RSA 私密金鑰加密，以提高安全性

我們強烈建議您使用密碼來加密 CA 的私密金鑰，請選擇 Y 來輸入密碼。

9 使用密碼片語為 SERVER 的 RSA 私密金鑰加密，以提高安全性

若使用密碼為伺服器金鑰加密，則每當要啟動網頁伺服器時就必須輸入這個密碼。如此一來就很難自動在開機時啟動伺服器或是重新啟動網頁伺服器。因此，使用者通常會在回答這個問題時選擇 N。請注意金鑰在沒有使用密碼加密時是不受保護狀態，同時請確定只有經授權人員可以存取該金鑰。

重要：加密伺服器金鑰

如果您選擇使用密碼為伺服器金鑰進行加密，請為存放在 /etc/sysconfig/apache2 的 APACHE_TIMEOUT 提高設定值。否則，您在嘗試啟動伺服器之前可能來不及輸入密碼片語，但是伺服器早就啟動失敗而停止。

程序檔的結果頁面會出現憑證清單，以及其所產生的金鑰。不同於程序檔輸出的結果，這些檔案並不是產生到本地目錄 conf，而是產生到 /etc/apache2/ 下面的正確目錄。

最後一個步驟就是從 /etc/apache2/ssl.crt/ca.crt 將 CA 憑證檔案複製到使用者可存取的位置，以便使用者將該檔案納入其網頁瀏覽器已知和信任的 CA 清單中。否則，瀏覽器會報告該憑證是由不明授權機構所簽發。這類憑證的有效期限是一年。

重要：自行簽署的證書

僅在供認識您、且信任您為認證機構之使用者存取的網頁伺服器上，方可使用自我簽署憑證。我們不建議您在例如公開商店等場所中使用這類憑證。

取得官方簽發憑證

目前有一些可簽署憑證的官方認證機構。這類憑證是由值得信任的協力廠商所簽署，因此可以完全信任。對外運作的安全網頁伺服器通常已取得官方簽發憑證。

最有名的官方 CA 是 Thawte (http://www.thawte.com/) 或 Verisign (http://www.verisign.com)。這些 CA 和其他 CA 都已經編譯到所有瀏覽器中，所以瀏覽器會自動接受這些憑證授權機構簽發的憑證。

在要求官方簽署的憑證時，您並不需要向 CA 傳送憑證，而只需傳送憑證簽署要求 (Certificate Signing Request, CSR)。若要建立 CSR，請呼叫程序檔 /usr/share/ssl/misc/CA.sh -newreq。

首先，程序檔會要求提供該 CSR 進行加密時所使用的密碼。接著，要求您輸入可識別名稱。這時系統會要求您回答一些問題，例如國家/地區名或組織名稱。請輸入有效資料 — 您在此輸入的每項資料將來都會顯示在憑證中並用於檢查。您不需要回答每個問題。如果有不適用的問題或是您希望保留空白，請使用 .。通用名稱是指 CA 本身的名稱 — 請選擇明顯的名稱，例如 *My company* CA。最後，必須輸入挑戰密碼和替用的公司名稱。

從您呼叫程序檔的目錄中找出此 CSR。這個檔案名稱是 newreq.pem。

40.6.2 設定提供 SSL 的 Apache

在網頁伺服器端上，SSL 和 TLS 要求的預設連接埠是 443。同時有傾聽連接埠 80 的「一般」Apache 和傾聽連接埠 443 之已啟用 SSL/TLS 的 Apache，並不會產生衝突。事實上，HTTP 和 HTTPS 可以執行相同的 Apache 例項。通常這時

會使用不同的虛擬主機，將連接埠 80 和連接埠 443 的要求分派到不同的虛擬伺服器。

重要：防火牆組態

請不要忘記為連接埠 443 上已啟用 SSL 的 Apache 開啟防火牆。您可以依據第 43.4.1 節「以 YaST 設定防火牆」[750頁]所述方式，透過 YaST 完成這個動作。

若要使用 SSL，必須在全域伺服器組態中將其啟用。請在編輯器中開啟 `/etc/sysconfig/apache2`，並搜尋 `APACHE_MODULES`。若模組清單中沒有出現此模組，請將「SSL」新增到清單中 (mod_ssl 是預設啟用的模組)。然後，搜尋 `APACHE_SERVER_FLAGS` 並新增「SSL」。如果已經選擇使用密碼來加密伺服器憑證，您就必須同時提高 `APACHE_TIMEOUT` 的設定值，以便您在啟動 Apache 時有足夠時間輸入該密碼片語。請重新啟動伺服器來確保這些變更生效。只是重新載入並無法保證變更生效。

虛擬主機組態目錄包含了樣板 `/etc/apache2/vhosts.d/vhost-ssl.template` 和 SSL 特定指示詞 (將提供詳細文件說明)。如需一般虛擬主機組態的詳細資訊，請參閱章節「虛擬主機組態」[678頁]。

若要開始，請將樣板複製到 `/etc/apache2/vhosts.d/`*`mySSL-host`*`.conf` 並進行編輯。充分調整下列指示詞的值：

- `DocumentRoot`
- `ServerName`
- `ServerAdmin`
- `ErrorLog`
- `TransferLog`

重要：以名稱為基礎的虛擬主機和 SSL

在只有一個 IP 位址的伺服器上，不可能同時執行多部已啟用 SSL 的虛擬主機。連接這類設定的使用者，將會在每次造訪該 URL 時收到警告訊息，告知

憑證與伺服器名稱不符。每個啟用 SSL 的網域都需要使用不同的 IP 位址或連接埠，才能根據有效的 SSL 憑證來完成通訊。

40.7 避免安全性問題

向公用網際網路公開的網頁伺服器，必須持續進行系統管理。軟體和意外的錯誤設定不可避免地會產生安全性問題。下面是可用來處理這些問題的幾項秘訣。

40.7.1 更新軟體

SUSE 會在發現 Apache 軟體弱點時，發出安全性建議事項。其中會包含應該要儘速套用的弱點修正指示。請由下列位置取得 SUSE 安全性公告：

- **網頁** http://www.novell.com/linux/security/securitysupport.html
- **郵件清單** http://en.opensuse.org/Communicate#Mailinglists
- **RSS Feed** http://www.novell.com/linux/security/suse_security.xml

40.7.2 DocumentRoot 許可權

根據 SUSE Linux Enterprise Server 預設，`DocumentRoot` 目錄 `/srv/www/htdocs` 與 CGI 目錄 `/srv/www/cgi-bin` 的所有權屬於 `root` 使用者和群組。這些許可權不可變更。如果目錄對所有人開放寫入權限，則任何使用者都可以將檔案放到目錄中。然後，這些檔案可能會由具有 `wwwrun` 許可權的 Apache 執行，而這種情況可能會造成使用者取得非預期的檔案系統資源存取權限。使用 `/srv/www` 子目錄來存放虛擬主機的 `DocumentRoot` 和 CGI 目錄，並確定這些目錄所有權屬於 `root` 使用者和群組。

40.7.3 檔案系統存取

依預設，`/etc/apache2/httpd.conf` 已設定成拒絕存取整個檔案系統。您絕對不可以覆寫這些指示詞，但是可以特別啟用可由 Apache 讀取之所有目錄的存取權限 (如需詳細資訊，請參閱章節「基本虛擬主機組態」 [681頁])。如果要執行這個動作，請確保沒有任何重要檔案 (例如密碼或系統組態檔案) 可由外界進行讀取。

40.7.4 CGI 程序檔

使用 Perl、PHP、SSI 或是任何其他程式設計語言的互動式程序檔，基本上都可以執行任意指令，因此會產生常見的安全性問題。將從伺服器執行的程序檔，只能由伺服器管理員信任的來源進行安裝 — 通常最好不要讓使用者執行他們自己的程序檔。同時建議您為所有程序檔進行安全性稽核。

為了盡可能簡化程序檔的管理工作，通常建議您限制 CGI 程序檔在特定目錄中執行，而不是全域性開放執行。您可以使用 `ScriptAlias` 和 `Option ExecCGI` 指示詞來進行組態設定。SUSE Linux Enterprise Server 的預設組態不允許隨處執行 CGI 程序檔。

所有 CGI 程序檔都是以相同使用者身份執行，所以不同的程序檔彼此之間可能會產生衝突。module suEXEC 可讓您以不同使用者和群組身份來執行 CGI 程序檔。

40.7.5 使用者目錄

在啟用使用者目錄 (使用 mod_userdir 或 mod_rewrite) 時，您應該審慎考量不要允許 `.htaccess` 檔案，因為此類檔案會允許使用者覆寫安全性設定。至少您應該使用 `AllowOverRide` 指示詞來限制使用者的應用範圍。在 SUSE Linux Enterprise Server 中，`.htaccess` 檔案預設為啟用，但是使用者在使用 mod_userdir 時不能覆寫任何 `Option` 指示詞 (請參閱 `/etc/apache2/mod_userdir.conf` 組態檔案)。

40.8 疑難排解

如果Apache未啟動，網頁就無法存取，或者使用者無法連接網頁伺服器，因此找出問題的根源是很重要的工作。下面是可在其中尋找錯誤原因、以及檢查重點項目的幾個常見位置。

首先，`rcapache2` 可提供詳細的錯誤相關資訊 (請參閱第 40.3 節「啟動和停止 Apache」[688頁])，因此在實際操作 Apache 時如果有使用這個指令，將對您非常有幫助。有時某些因素可能會誘使您使用二進位檔案 `/usr/sbin/httpd2`，來啟動或停止網頁伺服器。請避免這樣做，並改而使用 `rcapache2` 程序檔。`rcapache2` 甚至還會提供解決組態錯誤的秘訣和提示。

其次，不可忽視記錄檔案的重要性。無論是嚴重或不嚴重的錯誤，都可以從Apache 記錄檔案中找出錯誤發生原因。此外，如果需要檢視記錄檔案中更多的詳細資訊，還可以透過 `LogLevel` 指示詞來控制記錄訊息的詳細程度。依預設，錯誤記錄檔案是位在 `/var/log/apache2/error_log`。

提示：簡單測試

請使用 `tail -F /var/log/apache2/`*`my_error_log`* 指令來檢視 Apache 記錄訊息。然後執行 `rcapache2 restart`。現在，請嘗試連接到瀏覽器，並檢查輸出結果。

一個常見的錯誤為，沒有在伺服器防火牆組態中開啟Apache的連接埠。如果是使用 YaST 來設定 Apache，可以透過一個單獨的選項來檢查這個特定問題 (請參閱第 40.2.2 節「使用 YaST 設定 Apache」[682頁])。如果您要手動設定 Apache，請透過 YaST 防火牆模組來開啟 HTTP 和 HTTPS 的連接埠。

如果無法透過這些功能來查出錯誤原因，則請查閱 `http://httpd.apache.org/bug_report.html` 的線上 Apache 錯誤資料庫。此外，也可以從 `http://httpd.apache.org/userslist.html` 取得可用的郵件清單，聯絡Apache使用者社群。推薦的新聞群組是 `comp.infosystems.www.servers.unix`。

40.9 如需更多資訊

apache2-doc 套件在許多位置包含了完整的 Apache 手冊，用於本機安裝及作為參考文件。這個套件不是預設安裝選項 — 安裝此套件最快的方式就是使用 yast -i apache2-doc 指令。完成安裝之後，您就可以從 http://localhost/manual/ 使用 Apach 手冊。您也可以從 http://httpd.apache.org/docs-2.2/ 網站位置來存取這份手冊。/usr/share/doc/packages/apache2/README.* 目錄會提供 SUSE 特定組態秘訣資訊。

40.9.1 Apache 2.2

如需 Apache 2.2 最新功能的清單，請參閱 http://httpd.apache.org/docs/2.2/new_features_2_2.html。如需從 2.0 升級至 2.2 版的資訊，請參閱下列網址資訊：http://httpd.apache.org/docs-2.2/upgrading.html。

40.9.2 Apache 模組

如需有關第 40.4.5 節「外部模組」 [695頁]所介紹外部模組的詳細資訊，請參閱下列主題說明：

mod-apparmor
: http://en.opensuse.org/AppArmor

mod_perl
: http://perl.apache.org/

mod_php5
: http://www.php.net/manual/en/install.unix.apache2.php

mod_python
: http://www.modpython.org/

40.9.3 開發

如需更多有關開發 Apache 模組或是參與 Apache 網頁伺服器計畫的詳細資訊，請參閱下列主題內容：

Apache 開發人員資訊
http://httpd.apache.org/dev/

Apache 開發人員說明文件
http://httpd.apache.org/docs/2.2/developer/

使用 Perl 和 C 來編寫 Apache 模組
http://www.modperl.com/

40.9.4 其他資源

如果您在 SUSE Linux Enterprise Server 中遇到與 Apache 有關的問題，請查閱「技術資訊搜尋」，網址為：http://www.novell.com/support。關於 Apache 的歷程，請參閱 http://httpd.apache.org/ABOUT_APACHE.html。此頁面也說明稱伺服器為 Apache 的原因。

41 代理伺服器 Squid

Squid 是 Linux 與 UNIX 平台普遍使用的代理快取。這表示它會將要求的網際網路物件 (例如網頁伺服器或 FTP 伺服器上的資料)，儲存在比伺服器更接近要求工作站的機器上。您可設定多階層，以確保即使在終端使用者無法察覺的模式中，也能達到最佳的反應時間及較低的頻寬使用率。您可使用其他軟體如 squidGuard，以過濾網路內容。

Squid 可做為代理快取記憶體。它會將物件要求從用戶端 (在此例中是從網頁瀏覽器) 重新導向至伺服器。當從伺服器而來的要求物件到達時，它會將物件傳送到用戶端，並在硬碟快取記憶體中保留物件的副本。快取的其中一個優點為，當有數個用戶端要求相同的物件時，可從硬碟快取記憶體來提供。這可讓用戶端比從網際網路更快地擷取資料。這個程序也可以減少網路流量。

除了實際快取之外，Squid 還提供眾多功能，例如將負載分散到互相通訊的代理伺服器階層、為所有存取代理的用戶端定義嚴格的存取控制清單、允許或拒絕使用其他應用程式來存取特定網頁，以及產生經常瀏覽之網頁的統計資料，以評估使用者的瀏覽習慣。Squid 不是一般的代理。一般而言，它只會代理 HTTP 連線。它也支援 FTP、Gopher、SSL 以及 WAIS 等通訊協定，但不支援其他的網際網路通訊協定，例如 Real Audio、新聞或視訊會議。因為 Squid 只支援 UDP 通訊協定提供不同快取之間的通訊，許多其他的多媒體程式並不支援。

41.1 關於代理快取的說明

當 Squid 做為代理快取記憶體時，使用方法有多種。若與防火牆合併，它有助於提高安全性。多個代理可一起使用。它也可以判斷應該快取物件類型和持續的時間長短。

41.1.1 Squid 以及安全性

Squid可與防火牆配合使用，以便使用代理快取記憶體來保護內部網路不受外部的存取。防火牆將會拒絕所有的用戶端存取 Squid 以外的外部服務。所有的網路連線都必須由代理來建立。藉由這種組態方式，Squid可完全控制網頁存取。

如果防火牆組態包含 DMZ，則代理應在此區域中操作。第 41.5 節「設定操作順暢的代理」[721頁]將描述如何執行「*透明*」代理。這簡化了用戶端的組態，因為在此情況下，它們不需要有關代理的任何資訊。

41.1.2 多個快取

經過設定之後，可在多個 Squid 例項之間交換物件。這樣可以減少系統的總負載，並可增加在本地網路中找到現有物件的機會。您也可以設定快取記憶體階層，使快取記憶體可以將物件要求轉遞至同層級或上層的快取記憶體—使其可從本地網路的另一個快取記憶體或直接從來源取得物件。

為快取記憶體階層選擇適當的拓樸是非常重要的，因為這樣它就不會增加網路的整體流量。就大型的網路而言，就非常合適為每個子網路設定代理伺服器，並將它們連線至上層的代理，這樣就可以連線至 ISP 的代理快取。

這些通訊都是由在 UDP 通訊協定最上層執行的 ICP (網際網路快取通訊協定) 所處理。在快取之間的資料傳輸是使用以 TCP 為基礎的 HTTP (超文字傳輸通訊協定，Hypertext Transmission Protocol) 來處理。

為了能找到最適合的伺服器來取得物件，某個快取記憶體會將 ICP 要求傳送到所有同層級的代理。如果有偵測到物件，這些代理就會透過具有 HIT 代碼的 ICP 回應來回覆這些要求；如果沒有偵測到物件，則會透過具有 MISS 代碼的 ICP 回應來回覆這些要求。如果找到多個 HIT 回應，代理伺服器會根據某些因素 (例如哪個快取記憶體傳送回覆的速度最快，哪個伺服器距離最近) 來決定要從哪部伺服器下載。如果沒有收到符合的回應，則會將要求傳送到上層快取。

提示

為了避免網路上不同的快取記憶體中出現物件重複，系統會使用其他 ICP 通訊協定。例如 CARP (快取陣列路由通訊協定) 或 HTCP (超文字快取通訊協定)。在網路中維護愈多的物件，則找到所需物件的機率也就愈大。

41.1.3 快取網際網路物件

網路中所有可用的物件並不全是靜態。其中有許多動態產生的 CGI 頁面、訪客計數器以及加密的 SSL 內容文件。諸如此類的物件是不會被快取的，因為每次存取它們時，它們都會改變。

還有一個問題就是，所有儲存在快取記憶體中的其他物件，應該在那里保留多長時間。為了決定停留時間，系統會指派各種可能的狀態給快取記憶體中的所有物件。網頁以及代理伺服器會藉由新增標頭至這些物件來找出物件的狀態，例如「上一次修改」或「到期」以及對應的日期。也會使用不應該快取指定該物件的其他標題。

由於缺少可用的硬碟空間，系統一般會使用 LRU (最近使用) 等演算法來取代快取記憶體中的物件。基本上這表示代理會清空那些最久沒有被要求的物件。

41.2 系統需求

最重要的事情是要決定系統必須承受的最大網路負載。因此需要特別注意負載尖峰，尖峰值有可能是每天平均值的四倍。當存在疑問時，最好高估系統的需求，因為如果讓 Squid 在接近其容量的限制下工作，有可能會造成服務品質的嚴重損失。接下來的小節將依重要順序指出系統因素。

41.2.1 硬碟

速度在快取處理過程中扮演很重要的角色，因此應該特別注意這個因素。對硬碟而言，此參數可稱為「*隨機搜尋時間*」，以毫秒為單位。因為 Squid 所讀取或寫入硬碟的資料區塊通常都相當的小，所以硬碟的搜尋時間比其資料輸送量還要重要。如果要使用代理，最好選擇具有高旋轉速的硬碟，因為這種硬碟可以較快的速度將讀寫頭放置在所需的位置。有一個方法可能可以增加系統的速度，就是同時使用多個磁碟或是運用分段 RAID 陣列。

41.2.2 磁碟快取的大小

在小的快取記憶體中，HIT (發現要求的物件已在該處) 的機率比較小，因為快取記憶體很容易就會填滿，且新的物件會取代不太被要求的物件。例如，如果

快取有 1 GB 可用，而且使用者一天只瀏覽 10 MB，則需要一百天以上才能將快取填滿。

決定所需快取記憶體大小的最簡易方法為，考量連線的最大傳輸速率。連線速率為 1 Mbit/s 時，其最大的傳輸速率為 125 125 KB/s。如果所有這些流量最後都保留在快取記憶體中，則 1 個小時內就會增至 450 MB，假設所有的流量都只在 8 個工作小時中產生，則一天就可達到 3.6 GB。由於連線一般都不會用到其容量上限，因此可以假設快取記憶體所處理的總資料容量大約為 2 GB。這就是為什麼範例中 Squid 需要 2 GB 的磁碟空間，以快取一天中已瀏覽的資料量。

41.2.3 RAM

Squid 所需的記憶體容量 (RAM) 與快取記憶體中的物件數目直接相關。Squid 也會將快取記憶體物件的參照以及常要求的物件儲存在主記憶體中以加速此資料的擷取速度。隨機存取記憶體比硬碟的速度快很多。

除此之外，Squid 需要在記憶體中保留其他的資料，例如所有已處理 IP 位址的表格、精確的網域名稱快取、最常要求的物件、存取控制清單、緩衝區等等。

為 Squid 處理程序保留足夠的記憶體是非常重要的，因為若使用磁碟進行交換，系統效能就會大幅地降低。cachemgr.cgi 工具可用於快取記憶體管理。此工具會在第 41.6 節「cachemgr.cgi」 [724頁] 中加以介紹。網路流量龐大的網站應該考量使用配備超過 4 GB 記憶體的 AMD64 或 Intel 64 系統。

41.2.4 CPU

Squid 不是一個需要密集使用 CPU 的程式。只有在載入或檢查快取記憶體的內容時，才會增加處理器的負載。使用多處理器的機器並不能提升系統的效能。若要提升效率，最好買更快的磁碟或新增更多的記憶體。

41.3 啟動 Squid

Squid 在 SUSE® Linux Enterprise Server 中已經過預先設定，所以可在安裝後立即啟動。為了確保啟動更平順，應該將網路設定為至少使用一部名稱伺服器，而且可連接網際網路。如果撥號連線是使用動態 DNS 組態，就有可能產生問

題。在此範例中，至少應該輸入名稱伺服器，因為如果 Squid 在 /etc/resolv.conf 中偵測不到 DNS 伺服器，它就不會啟動 。

41.3.1 開始和停止 Squid 的指令

若要啟動 Squid，請以 root 的身份在指令行中輸入 rcsquid start。第一次啟動時，必須先在 /var/cache/squid 中定義快取記憶體的目錄結構。目錄的定義可由 /etc/init.d/squid 啟動程序檔自動完成，這可能需要花費數秒鐘，甚至幾分鐘。如果 done 以綠色出現在右邊，則表示已成功載入 Squid。若要在本地系統上測試 Squid 的功能，請在瀏覽器中輸入 localhost 做為代理，並輸入 3128 做為埠。

若要允許本地系統和其他系統的使用者存取 Squid 和網際網路，請將 /etc/squid/squid.conf 組態檔中的 http_access deny all 項目變更為 http_access allow all。然而，當您這麼做時，請考量到此動作將使 Squid 可供任何人完全存取。因此，請定義控制代理存取權限的 ACL。如需關於此的詳細資訊，請參閱 第 41.4.2 節「存取控制的選項」 [719頁]。

在修改 /etc/squid/squid.conf 組態檔後，Squid 必須重新載入組態檔。請以 rcsquid reload 執行此動作。或者也可以使用 rcsquid restart 完全重新啟動 Squid。

rcsquid status 指令可用來檢查代理是否正在執行。rcsquid stop 指令可關閉 Squid。這可能需要花一段時間，因為 Squid 會在中斷與用戶端的連線並將其資料寫入磁碟前，先等待半分鐘 (在 /etc/squid/squid.conf 中的 shutdown_lifetime 選項)。

警告：終止 Squid

使用 kill 或 killall 終止 Squid 可能會損毀快取記憶體。若要能夠重新啟動 Squid，就必須先刪除損毀的快取。

如果 Squid 在成功啟動後，仍然於一小段時間後即停止運作，請檢查是否有錯誤的名稱伺服器項目，或者是否缺少 /etc/resolv.conf 檔案。Squid 會在 /var/log/squid/cache.log 檔案中記錄啟動失敗的原因。如果要在系統開機時自動載入 Squid，請使用 YaST Runlevel 編輯器針對所需的 Runlevel 啟用 Squid。請參閱第 8.5.12 節「系統服務 (Runlevel)」 [148頁]。

解除安裝 Squid 時不會移除快取記憶體階層或記錄檔。若要移除這些階層，請手動刪除 /var/cache/squid 目錄。

41.3.2 本地 DNS 伺服器

即使本地 DNS 伺服器不管理自己的網域，也可以設定本地 DNS 伺服器。它可以做為僅供快取的名稱伺服器，也可以透過根 名稱伺服器來解析DNS要求，而不需任何特殊的組態 (請參閱第 33.3 節「啟動名稱伺服器 BIND」[568頁])。如何達成此目的，端視您在設定網際網路連線的組態時，是否選擇動態的DNS而定。

動態 DNS

一般而言，使用動態 DNS 時，提供者會在網際網路連線建立期間設定 DNS 伺服器，而 /etc/resolv.conf 本地檔案會自動進行調整。這種運作方式的控制是在 /etc/sysconfig/network/config 檔案中將 MODIFY_RESOLV_CONF_DYNAMICALLY sysconfig 變數設為 "yes"。請用 YaST sysconfig 編輯器將此變數設為 "no" (請參閱第 20.3.1 節「使用 YaST Sysconfig 編輯器變更系統組態」[368頁])。然後在 /etc/resolv.conf 檔案中輸入本地 DNS 伺服器，以 IP 位址 127.0.0.1 做為 localhost。這樣一來在啟動時，Squid 就可以永遠找到本地名稱伺服器。

為了能夠存取提供者的名稱伺服器，必須在 forwarders 下的 /etc/named.conf 組態檔中輸入提供者的名稱及其 IP 位址。若使用動態 DNS，只要將 MODIFY_NAMED_CONF_DYNAMICALLY sysconfig 變數設為 YES，就可以在連線建立時自動完成此動作。

靜態 DNS

若使用靜態 DNS，建立連線時將不會執行任何自動的 DNS 調整，所以不必變更任何 sysconfig 變數。但是您必須依照上述方式在 /etc/resolv.conf 檔案中輸入本地的 DNS 伺服器。除此之外，在 forwarders 下的 /etc/named.conf 檔案中必須手動輸入提供者的靜態名稱伺服器及其 IP 位址。

提示：DNS 與防火牆

如果您有執行防火牆，請確定 DNS 要求可以通過防火牆。

41.4 /etc/squid/squid.conf 組態檔案

所有的 Squid 代理伺服器的設定值都是在 /etc/squid/squid.conf 檔案中設定。在第一次啟動 Squid 時，此檔案不需做任何變更，但是外部用戶端一開始為拒絕存取。代理可供 localhost 使用。預設的連接埠是 3128。預先安裝的 /etc/squid/squid.conf 組態檔可提供關於選項及許多範例的詳細資訊。幾乎所有的項目都是以 # (加備註的行) 開始，而且可以在行的結尾找到相關的規格。指定的值永遠都與預設值相關，因此在大部分情況下，如果移除備註符號而不變更任何參數，實際上沒有什麼效果。如果有可能，請保留原始的範例，並在行的下方插入選項以及修改過的參數。如此一來，就可以輕易復原預設值，並與變更做比較。

提示：在更新後調整組態檔案

如果您是從較早的 Squid 版本更新，建議您編輯新的 /etc/squid/squid.conf，並且只套用在舊檔案中所做的變更。如果您嘗試使用舊的 squid.conf，請注意該組態可能不再有效，因為選項有時會被修改並增加新的變更。

41.4.1 一般組態選項 (選擇)

http_port 3128

這是 Squid 傾聽用戶端要求所用的連接埠。預設的連接埠是 3128，但是 8080 也是常用的連接埠。如果有需要，請指定數個以空白分隔的埠號碼。

cache_peer *hostname type proxy-port icp-port*

在此可以輸入上層代理，例如如果您想要使用 ISP 的代理。針對 *hostname* 的部分，輸入要使用的代理 IP 位址，並針對 *type* 部分，輸入 parent。至於 *proxy-port* 的部分，則輸入上層運算子也會指定的埠號碼，以便在瀏覽器中使用，通常是 8080。如果上層的 ICP 埠是未知的，而且其用途與提供者無關，請將 *icp-port* 設為 7 或 0。除此之外，還可在埠號碼後面指定 default 與 no-query，禁止使用 ICP 通訊協定。就提供者代理而言，Squid 接著就會像一般的瀏覽器一樣地運作。

cache_mem 8 MB

這個項目定義 Squid 可用於常見回覆的記憶體容量。預設值為 `8 MB`。這不指定 Squid 的記憶體使用量，而且可以超過。

cache_dir ufs /var/cache/squid/ 100 16 256

cache_dir 項目定義磁碟上儲存所有物件的目錄。在結尾處的數目代表可以使用的最大磁碟空間 (MB)，以及在第一層與第二層的目錄數目。`ufs` 參數不可變更。在預設情況下，`/var/cache/squid` 目錄佔用 100 MB 磁碟空間，而且可以在其中建立 16 個子目錄，每個目錄還各包含 256 個子目錄。當指定要使用的磁碟空間時，請保留足夠的預留磁碟空間。在此指定可用磁碟空間的 50% (最小) 至 80% (最大) 的值最為合理。最後兩個目錄值應該謹慎地增加，因為太多的目錄也有可能導致效能降低。如果您有數個共享快取的磁碟，請輸入數行 *cache_dir*。

cache_access_log /var/log/squid/access.log , cache_log /var/log/squid/cache.log , cache_store_log /var/log/squid/store.log

這三個項目指定 Squid 記錄其所有動作的路徑。一般而言，這裏不會有所變更。如果 Squid 負荷過重，則將快取與記錄檔分散到數個磁碟會是一個好辦法。

emulate_httpd_log off

如果該項目設為 *on*，就會取得可以讀取的記錄檔。然而有些試用程式無法解譯此檔案。

client_netmask 255.255.255.255

使用這個項目時，可以遮罩記錄檔中用戶端的 IP 位址。如果您在這裏輸入 `255.255.255.0`，IP 位址的最後一位數就會設為 0。您可以用這種方式保護用戶端的隱私。

ftp_user Squid@

使用這個項目可設定匿名 FTP 登入應該使用的 Squid 密碼。在此也可以指定有效的電子郵件地址，因為某些 FTP 伺服器會檢查這些地址的有效性。

cache_mgr webmaster

如果 Squid 意外損毀，Squid 應傳送訊息給此電子郵件地址。預設值為*網站管理員*。

logfile_rotate 0

如果您執行 `squid -k rotate`，則 `Squid` 會輪換安全的記錄檔。在此程序中會計算檔案的數目，而且在到達指定的值後，就會覆寫最舊的檔案。預

設值為 0，因為歸檔和刪除在 SUSE Linux Enterprise Server 中的記錄檔是由設定在 /etc/logrotate/squid 組態檔中的 cron 工作所執行。

append_domain <domain>
使用 *append_domain*，可在沒有給定網域時，指定要自動附加的網域。通常，可在此處輸入您自己的網域，因此在瀏覽器中輸入 *www* 就可以存取您自己的網頁伺服器。

forwarded_for on
如果您將項目設為 *off*，則 Squid 會從 HTTP 要求移除 IP 位址以及用戶端系統名稱。否則，它會在標頭中新增類似下一行

```
X-Forwarded-For: 192.168.0.1
```

negative_ttl 5 minutes; negative_dns_ttl 5 minutes
一般而言，您不需要變更這些值。不過，就算您有撥號連線，網際網路有時也可能無法使用。雖然已重新建立網際網路連線，Squid 仍然會記錄失敗的要求，並拒絕發出新的要求。像這個例子中，將「*分鐘*」變更為「*秒*」，然後在瀏覽器中按一下「*重新載入*」後，幾秒鐘之後就會進行撥號程序。

never_direct allow *acl_name*
為了防止 Squid 直接從網際網路接受要求，請使用上述指令以強制連接另一個代理。此指令應該在之前已於 *cache_peer* 中輸入。如果 all 是指定為 *acl_name*，請強制將所有的要求直接轉遞至 *parent*。例如，如果您所使用的提供者，嚴格地規定其代理的使用方式或拒絕其防火牆直接存取網際網路，就可能需要執行此動作。

41.4.2 存取控制的選項

Squid 會提供一個詳細系統來控制代理存取。透過執行 ACL，可以輕鬆並完整地設定該系統。這牽涉到列出依序處理之規則的清單。在使用 ACL 前必須先進行定義。某些預設的 ACL，例如 *all* 與 *localhost* 已經存在。然而，僅定義 ACL 並不代表實際上會套用。這只會發生在連接 *http_access* 規則使用時。

acl <acl_name> <type> <data>
至少需要三種規格才能定義 ACL。可以任意選擇 <*acl_name*> 的名稱。至於 <*type*> 的部分，可從各種不同的選項中選取 (選項位於 /etc/squid/squid.conf 檔案中的 *ACCESS CONTROLS* 小節)。<*data*> 的規格取決於個別的 ACL 類型，而且也可以從檔案讀取，例如，透過主機名稱、IP 位址或 URL。下列是一些簡單的範例：

```
acl mysurfers srcdomain .my-domain.com
acl teachers src 192.168.1.0/255.255.255.0
acl students src 192.168.7.0-192.168.9.0/255.255.255.0
acl lunch time MTWHF 12:00-15:00
```

http_access allow <acl_name>

http_access 定義哪些人可使用代理，以及哪些人可存取網際網路的哪些內容。因此，必須指定 ACL。已在上文定義的 *localhost* 與 *all*，可透過 *deny* 或 *allow* 拒絕或允許存取。您可以建立包含任何數目之 *http_access* 項目的清單，系統會從上至下處理這些項目，並按照前面優先的原則允許或拒絕存取個別 URL。最後一個項目應該永遠為 *http_access deny all*。在下列範例中，*localhost* 對於每個項目都擁有完整的存取權，而所有其他的主機則完全無法存取。

```
http_access allow localhost
http_access deny all
```

在另一個使用這些規則的範例中，`teachers` 群組永遠具有網際網路的存取權。students 群組只能取得在星期一到星期五午休時間的存取權。

```
http_access deny localhost
http_access allow teachers
http_access allow students lunch time
http_access deny all
```

基於可讀性的理由，*http_access* 項目清單，只能在 `/etc/squid/squid.conf` 檔案中的指定位置中輸入。也就是，在文字之間

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR
# CLIENTS
```

以及最後

```
http_access deny all
```

redirect_program /usr/bin/squidGuard

使用此選項，可以指定如 squidGuard 之類的重新導向器，來封鎖不需要的 URL。使用代理驗證以及適當的 ACL，就可以為不同的使用者群組個別地控制網際網路的存取。squidGuard 是可以安裝和設定的獨立套件。

auth_param basic program /usr/sbin/pam_auth

如果在代理上必須對使用者進行驗證，請設定對應的程式，例如pam_auth。第一次存取 pam_auth 時，使用者會看到一個登入視窗，請在此輸入使用者名稱與密碼。除此之外，仍然需要 ACL，因此只有擁有有效登入的用戶端可以使用網際網路：

```
acl password proxy_auth REQUIRED

http_access allow password
http_access deny all
```

在 *proxy_auth* 之後的 *REQUIRED* 可以使用允許的使用者名稱清單或是到達這類清單的路徑來取代。

ident_lookup_access allow <acl_name>

使用這個項目，就會為所有 ACL 定義的用戶端執行 ident 要求以尋找每個使用者的身份。如果您將 *all* 套用至 *<acl_name>*，會對所有用戶端生效。另外，ident 精靈必須在所有的用戶端上執行。在 Linux 上，請為此用途安裝 pidentd 套件。在 Microsoft Windows 上，可以從網際網路下載可用的免費軟體。若要確保只允許 ident 查詢為成功的用戶端，請在此定義對應的 ACL：

```
acl identhosts ident REQUIRED

http_access allow identhosts
http_access deny all
```

同樣地，請使用允許的使用者名稱清單來取代 *REQUIRED*。使用 *ident* 會大幅增加存取時間，因為每個要求都會重複 ident 查詢。

41.5 設定操作順暢的代理

使用代理伺服器的一般方法如下：網頁瀏覽器將要求傳送至代理伺服器的某個連接埠，然後代理伺服器會提供這些所需物件(無論它們是否在快取記憶體中)。使用網路工作時，可能會發生數種狀況：

- 基於安全理由，建議所有的用戶端都使用代理瀏覽網際網路。
- 所有的用戶端都必須使用代理，不論它們是否注意到它。
- 雖然代理在網路上是變動的，但現有的用戶端應該保留其舊有的組態。

在所有上述情況下，都可使用透明代理。原則很簡單：代理會攔截和回應網頁瀏覽器的要求，因此網頁瀏覽器不必知道頁面的來源，即可收到要求的頁面。如名稱所示，整個程序會流暢地執行。

41.5.1 在 /etc/squid/squid.conf 中的組態選項

在 `/etc/squid/squid.conf` 檔案中啟動選項，以啟動和執行操作順暢的 Proxy：

- `httpd_accel_host` virtual
- `httpd_accel_port` 80

  實際 HTTP 伺服器所在的埠號碼
- `httpd_accel_with_proxy` on
- `httpd_accel_uses_host_header` on

41.5.2 使用 SuSEfirewall2 的防火牆組態

現在，請使用連接埠轉遞規則，透過防火牆將所有的內送要求重新導向至 Squid 埠。若要這麼做，請使用 第 43.4.1 節「以 YaST 設定防火牆」[750頁] 中說明的隨附工具 SuSEfirewall2。其組態檔位於 `/etc/sysconfig/SuSEfirewall2`。組態檔由詳細記錄的項目所組成。若要設定操作順暢的代理，您必須設定數個防火牆選項：

- 指向網際網路的設備：`FW_DEV_EXT="eth1"`
- 指向網路的設備：`FW_DEV_INT="eth0"`

在防火牆上，定義供不受信任的 (外部) 網路 (如網際網路) 存取的連接埠與服務 (請參閱 `/etc/services`)。在此範例中，僅提供對外的 Web 服務：

```
FW_SERVICES_EXT_TCP="www"
```

在從安全 (內部) 網路存取的防火牆上，定義埠或服務 (請參閱 /etc/services)，兩者都是透過 TCP 與 UDP 服務：

```
FW_SERVICES_INT_TCP="domain www 3128"
FW_SERVICES_INT_UDP="domain"
```

如此可允許存取 Web 服務與 Squid (預設埠為 3128)。「domain」服務代表 DNS (網域名稱服務)。這個服務使用非常普遍。否則，請直接將它從上述項目移除，並將下列選項設為 no：

```
FW_SERVICE_DNS="yes"
```

最重要的選項是選項數字 15：

範例 41.1 *防火牆組態：選項 15*

```
# 15.)
# Which accesses to services should be redirected to a local port
# on the firewall machine?
#
# This can be used to force all internal users to surf via your
# Squid proxy, or transparently redirect incoming Web traffic to
# a secure Web server.
#
# Choice: leave empty or use the following explained syntax of
# redirecting rules, separated with spaces.
# A redirecting rule consists of 1) source IP/net,
# 2) destination IP/net, 3) original destination port and
# 4) local port to redirect the traffic to, separated by a colon,
# e.g. "10.0.0.0/8,0/0,80,3128 0/0,172.20.1.1,80,8080"
```

上方的註解顯示要遵照的語法。首先，輸入存取代理防火牆之內部網路的 IP 位址與網路遮罩。其次，輸入這些用戶端的要求傳送到的 IP 位址與網路遮罩。如果是網頁瀏覽器，請將網路指定為 0/0，萬用字元表示「可到任何位置」。完成後，輸入這些要求傳送到的原始目的埠，最後，輸入重新導向所有這些要求的目的埠。由於 Squid 支援 HTTP 以外的通訊協定，請將要求從其他連接埠重新導向到代理，例如 FTP (埠 21)、HTTPS、或 SSL (埠 443)。在此範例中，Web 服務 (埠 80) 會重新導向至代理埠 (埠 3128)。如果需要新增更多的網路或服務，必須在各個項目中以空格隔開。

```
FW_REDIRECT="192.168.0.0/16,0/0,tcp,80,3128 192.168.0.0/16,0/0,tcp,21,3128"
FW_REDIRECT="192.168.0.0/16,0/0,udp,80,3128 192.168.0.0/16,0/0,udp,21,3128"
```

若要啟動防火牆並使用它所包含的新組態，請在 /etc/sysconfig/SuSEfirewall2 檔案中變更項目。START_FW 項目必須設為 "yes"。

啟動 Squid，如第 41.3 節「啟動 Squid」[714頁]所示。若要檢查每個項目是否能正常運作，請檢查在 /var/log/squid/access.log 中的 Squid 記錄。 若要驗證是否已正確設定所有的連接埠，請從網路外部的任何電腦對機器執行連接埠掃描。只應開啟 Web 服務 (埠 80)。若要以 nmap 掃描連接埠，指令語法為 nmap -O IP_address。

41.6 cachemgr.cgi

快取管理員 (cachemgr.cgi) 是一種 CGI 公用程式，用來顯示正在執行之 Squid 處理程序的記憶體使用率統計資料。它也是管理快取和檢視統計資料較方便的方式，因為不需要登入伺服器。

41.6.1 設定

首先，需要在系統上執行網頁伺服器。依第 40 章「*Apache HTTP 伺服器*」[673頁]中所述方式設定 Apache。若要檢查 Apache 是否已在執行，請以 root 的身份輸入 rcapache status 指令。如果出現像這類的訊息：

```
Checking for service httpd: OK
Server uptime: 1 day 18 hours 29 minutes 39 seconds
```

表示 Apache 正在機器上執行。否則，請輸入 rcapache start，以使用 SUSE Linux Enterprise Server 預設值啟動 Apache。設定 Apache 的最後一個步驟是將 cachemgr.cgi 檔案複製到 Apache 的 cgi-bin 目錄：

```
cp /usr/share/doc/packages/squid/scripts/cachemgr.cgi /srv/www/cgi-bin/
```

41.6.2 在 /etc/squid/squid.conf 中的快取管理員 ACL

快取管理員需要原始檔案中某些預設的設定值。首先，定義兩個 ACL，然後 http_access 選項使用這些 ACL 來授予可從 CGI 程序檔存取 Squid 的權限。第一

個 ACL 最重要，因為快取管理員會嘗試透過 cache_object 通訊協定與 Squid 通訊。

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
```

下列規則會將 Squid 的存取權限授與 Apache：

```
http_access allow manager localhost
http_access deny manager
```

這些規則是假設網頁伺服器與 Squid 在相同的機器上執行。如果快取管理員與 Squid 之間的通訊是源自於另一部電腦上的網頁伺服器，請依 範例 41.2「存取規則」[725頁] 所述包含一個額外的 ACL。

範例 41.2 *存取規則*

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl webserver src 192.168.1.7/255.255.255.255 # webserver IP
```

然後新增 範例41.3「存取規則」[725頁] 中的規則，以允許從網頁伺服器存取。

範例 41.3 *存取規則*

```
http_access allow manager localhost
http_access allow manager webserver
http_access deny manager
```

設定管理員的密碼以存取更多的選項，如遠端關閉快取記憶體或是檢視快取記憶體的詳細資訊。為此，請以管理員的密碼以及要檢視的選項清單設定 `cachemgr_passwd` 項目。這個清單會顯示成 `/etc/squid/squid.conf` 中項目備註的一部份。

每次變更組態檔時，請重新啟動 Squid。使用 `rcsquid reload` 即可輕鬆執行重啟。

41.6.3 檢視統計資料

瀏覽對應的網站—`http://webserver.example.org/cgi-bin/cachemgr.cgi`。按「*繼續*」，然後瀏覽不同的統計資料。有關快取管理員所顯示之每個

項目的詳細資訊，可在 Squid FAQ 中找到，網址為：http://www.squid-cache.org/Doc/FAQ/FAQ-9.html。

41.7 squidGuard

本節目的不是說明 squidGuard 的廣泛組態，而是簡單介紹並給予使用上的一些建議。如需更深入的組態問題，請參閱 squidGuard 網站，網址為 http://www.squidguard.org。

squidGuard 屬於自由軟體(GPL)，是一個靈活而快速的過濾器，也是重新導向器以及 Squid 的存取控制器外掛程式。它可讓您在 Squid 快取上，針對不同的使用者群組，使用不同的限制來定義多重存取規則。squidGuard 使用 Squid 的標準重新導向器介面。squidGuard 可以執行下列動作：

- 將某些使用者的 Web 存取權，限制為已接受的清單或已知的網頁伺服器或 URL。
- 針對某些使用者，封鎖某些列示或列為黑名單的網頁伺服器或 URL 的存取權。
- 針對某些使用者，封鎖符合一般運算式或文字清單的 URL。
- 將封鎖的 URL，重新導向至「智慧型」的 CGI 資訊頁面。
- 將未註冊的使用者重新導向至註冊表單。
- 將橫幅重新導向至空白的 GIF。
- 根據時間、星期、日期等，使用不同的存取規則。
- 針對不同的使用者群組，使用不同的規則。

squidGuard 與 Squid 無法用於：

- 編輯、過濾或審查文件內的文字。
- 編輯、過濾或審查 HTML 內嵌的程序檔語言，例如 JavaScript 或 VBscript。

使用前，請先安裝 squidGuard。以 /etc/squidguard.conf 提供最小的組態檔。組態範例請見 http://www.squidguard.org/config/。稍後請使用較複雜的組態設定值來測試。

接著，如果用戶端要求已列為黑名單的網站，請建立虛擬的「拒絕存取」頁面，或建立一個有點複雜的 CGI 頁面以重新導向 Squid。強烈建議使用 Apache。

現在，請將 Squid 設定為使用 squidGuard。在 /etc/squid/squid.conf 檔案中使用下列項目：

```
redirect_program /usr/bin/squidGuard
```

另一個稱為 redirect_children 的選項會設定在機器上執行之「重新導向」(在此範例中為 squidGuard) 程序的數量。squidGuard 有足夠快的速度處理許多要求：在具有 5,900 個網域和 7,880 個 URL (共 13,780)，處理器為500 MHz Pentium 的情況下，它可以在 10 秒內處理 100,000 個要求。因此，不建議設定四個以上的程序，因為這些程序的配置將會消耗相當大的的記憶體量。

```
redirect_children 4
```

最後，執行 rcsquid reload，讓 Squid 載入新的組態。現在，請使用瀏覽器測試您的設定值。

41.8 使用 Calamaris 產生快取報告

Calamaris 是一種 Perl 程序檔，用來產生 ASCII 或 HTML 格式的快取記憶體活動報告。它使用原生的 Squid 存取記錄檔。Calamaris 的首頁網址為 http://Calamaris.Cord.de/。該程式非常容易使用。

以 root 的身份登入，然後輸入 cat access.log.files | calamaris *options* > reportfile。當要傳輸一個以上的記錄檔時，須依時間順序來排列記錄檔，較舊的檔案排在前面。以下是程式的某些選項：

-a
: 輸出所有可用的報告

-w
: 以 HTML 報告輸出

-l
在報告標題中包含訊息或標誌

在程式的手冊頁中，使用 man calamaris 可以找到各種選項的詳細資訊。

以下是典型的範例：

```
cat access.log.2 access.log.1 access.log | calamaris -a -w \
> /usr/local/httpd/htdocs/Squid/squidreport.html
```

這會將報告放在網頁伺服器的目錄中。需要有 Apache 才能檢視報告。

另一個強大的快取記憶體報告產生器工具為 SARG (Squid 分析報告產生器)。如需詳細資訊，請參閱 http://sarg.sourceforge.net/。

41.9 如需更多資訊

請瀏覽 Squid 的首頁，網址為 http://www.squid-cache.org/。此處可以找到「Squid 使用者指南」(Squid User Guide) 以及有關 Squid 常見問題集 (FAQ) 的豐富資訊。

安裝後，可以在 howtoenh 中找到有關透明代理之簡要操作資訊，檔案名稱為 /usr/share/doc/howto/en/txt/TransparentProxy.gz。除此之外，還可以在 squid-users@squid-cache.org 中找到 Squid 的郵件清單。這個的歸檔是位於 http://www.squid-cache.org/mail-archive/squid-users/。

V. 安全性

42 管理 X.509 憑證

愈來愈多的驗證機制基於加密程序構建。能將加密金鑰指派給其擁有者的數位憑證在上述技術中扮演重要的角色。舉例來說，除了用於通訊之外，這些憑證也可應用在公司ID卡上。憑證的產生與管理大部分是由「官方」機構處理，並以商業服務形式提供。然而在某些情況下，例如，若公司不願意將個人資料交給第三者，則必須由您自行完成這些工作。

YaST 針對憑證提供了兩種模組，可處理數位 X.509 憑證的基本管理功能。以下各節將說明數位憑證的基礎，以及如何使用 YaST 來建立和管理此類憑證。如需更多詳細資訊，請參閱 http://www.ietf.org/html.charters/pkix-charter.html。

42.1 數位憑證原理

數位憑證使用加密程序來加密資料，以防止未經授權的使用者存取資料。使用者資料是透過第二資料記錄或*金鑰*來加密的。金鑰會透過數學運算方式來套用到使用者資料，會產生變更過的資料記錄，其原始內容無法辨識。目前廣泛使用的是非對稱式加密 (*公用金鑰方法*)。金鑰必定成對出現：

私密金鑰
: 私密金鑰必須由金鑰擁有者妥善保管。意外地公開私密金鑰，會危及金鑰組合並使其失效。

公用金鑰
: 金鑰擁有者會發行公用金鑰，以供第三者使用。

42.1.1 金鑰驗證性

由於公用金鑰程序應用廣泛，有許多公用金鑰在流通。若要順利使用本系統，每個使用者都必須能確定公用金鑰確實為假定的擁有者所擁有。使用者的公用金鑰指派要由可信任的組織根據公用金鑰憑證來確認。這類的憑證含有金鑰擁有者的名稱，對應的公用金鑰，以及發行該憑證的個人電子簽名。

發行及簽署公用金鑰憑證的可信任組織，通常是憑證基礎結構的一份子，這些組織還需負責其他方面的憑證管理，例如，發行、撤銷和更新憑證。這類基礎結構一般稱為*公用金鑰基礎結構*或 *PKI*。常見的 PKI 是 *OpenPGP* 標準，其中不採用中央授權點，而是由使用者自行發行他們的憑證。透過「信任網」中的其他廠商簽署後，這些憑證會變成可信任的。

X.509 公用金鑰基礎結構 (PKIX) 是 *IETF* (網際網路工程任務推動小組) 定義的替代模型，此模型目前幾乎已成為所有公用 PKI 的典範。在此模型中，*認證機構* (CA) 會在階層樹狀結構中執行驗證。此樹狀結構的根目錄是根 CA，它會簽發所有子 CA 的憑證。子 CA 的最低層發行使用者憑證。使用者憑證的憑證可追蹤至根 CA，因此可以信任。

這類 PKI 的安全性有賴 CA 憑證的可靠性而定。為了讓 PKI 客戶瞭解憑證實施內容，PKI 操作人員定義了*憑證實施準則* (CPS) 來界定憑證管理程序。這可確保 PKI 只發行可信任憑證。

42.1.2 X.509 憑證

X.509 憑證是個含有多個固定欄位並可能含有其他延伸資料的資料結構。固定欄位內容主要包含金鑰擁有者名稱、公用金鑰以及發行 CA (名稱和簽章) 的相關資料。基於安全性考量，憑證有效性應有時間上的限制，因此另有一個欄位內含有效期限。CA 會保證憑證在特定期間當中的有效性。CPS 通常需要 PKI (發行 CA) 來建立憑證並在到期前分發新憑證。

延伸資料可以包含任何其他資訊。只有當延伸資料被辨識為*關鍵*資料時，應用程式才需要評估它。如果應用程式無法辨別關鍵的延伸資料，則該程式必須拒絕憑證。有些延伸資料只適用於特定應用程式，例如簽名或加密。

表格 42.1 顯示版本 3 中的基本 X.509 憑證欄位。

表格 42.1　X.509v3 憑證

欄位	內容
版本	憑證版本，例如 v3
序號	唯一憑證 ID (整數)
簽名	用來簽署憑證的演算法 ID。
發證者	發行機構 (CA) 的唯一名稱 (DN)
有效性	有效期間
標題	擁有者的唯一名稱 (DN)
主旨公用金鑰資訊	擁有者的公用金鑰和演算法的 ID
發行者唯一 ID	發行 CA 的唯一 ID (選擇性)
主旨唯一 ID	擁有者的唯一 ID (選擇性)
延伸	其他選擇性資訊，例如「KeyUsage」或「BasicConstraints」

42.1.3 封鎖 X.509 憑證

如果憑證在到期前就變成不可信任，此憑證應立即封鎖。舉例來說，意外公開私密金鑰時，就需要這麼做。如果此私密金鑰屬於 CA 而非使用者憑證，封鎖憑證這個動作就特別重要。在此情況下，透過相關 CA 所發行的全部使用者憑證必須立即加以封鎖。如果憑證遭封鎖，PKI (負責 CA) 必須使用*憑證撤銷清單* (CRL)，向所有相關人士通知此資訊。

這些清單由 CA 定期提供給公用 CRL 發布點 (CDP)。CDP 也可以命名為憑證中的延伸，好讓檢查程式擷取目前的 CRL 來進行驗證。完成此工作的方法是使用*線上憑證狀態通訊協定* (OCSP)。CRL 的真實性是根據發行 CA 的簽名來確認。表格 42.2「X.509 憑證撤銷清單 (CRL)」[734頁]顯示 X.509 CRL 的基本部分。

表格 42.2 X.509 憑證撤銷清單 (CRL)

欄位	內容
版本	CRL 版本，例如 v2
簽名	用來簽署 CRL 的演算法 ID。
發證者	CRL 發行者 (通常為發行 CA) 的唯一名稱 (DN)
這次更新	此 CRL 的發行時間 (日期、時間)
下一次更新	下一個 CRL 的發行時間 (日期、時間)
撤銷憑證清單	每個項目都包含憑證序號、撤銷時間以及選擇性的延伸 (CRL 項目延伸)
延伸	選擇性 CRL 延伸

42.1.4 憑證與 CRL 的儲存區

CA 的憑證與 CRL 必須使用*儲存庫*來提供公開存取。由於簽章能防止憑證與 CRL 被偽造，因此儲存庫本身並不需要採用特殊保護方式。相反地，它會盡可能提供簡單、快速的存取環境。因此，憑證通常會在 LDAP 或 HTTP 伺服器上提供。如需 LDAP 的說明，請參閱 第 36 章「*LDAP—一種目錄服務*」[605頁]。第 40 章「*Apache HTTP 伺服器*」[673頁] 包含 HTTP 伺服器的相關資訊。

42.1.5 專用 PKI

YaST 包含提供 X.509 憑證基本管理的模組。這部分主要與建立 CA、子 CA 以及其憑證有關。PKI 的服務不單只是建立和分發憑證與 CRL 而已。PKI 的操作需要設想完善的管理基礎結構，才能持續更新憑證與CRL。這個基礎結構由 PKI 商業產品提供，也可以部分自動化。YaST 提供建立和分發 CA 與憑證的工具，但目前無法提供此背景基礎結構。若要設定小型 PKI，您可以使用可用的 YaST 模組。不過，您應該使用商業產品來設定「正式」或商用 PKI。

42.2 適用於 CA 管理的 YaST 模組

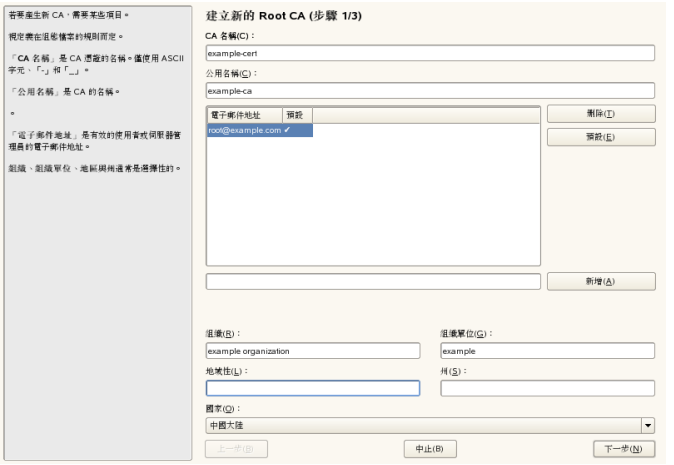
YaST 針對基本 CA 管理提供了兩種模組。此處將說明使用這些模組的主要管理工作。

42.2.1 建立根 CA

設定 PKI 時的首要步驟是建立一個根 CA。請進行下列幾項操作：

1. 啟動 YaST，並移至「*安全性與使用者*」>「*CA 管理*」。
2. 按一下「*建立根 CA*」。
3. 在第一個對話方塊輸入 CA 的基本資料，如圖形 42.1「YaST CA 模式 — 根 CA 的基本資料」[735頁]所示。文字欄位代表下列意義：

圖形 42.1　YaST CA 模式 — 根 CA 的基本資料



「*CA 名稱*」
輸入CA的技術名稱。包括目錄名稱在內的其他項目皆衍生自此名稱，這也是為何只能使用說明中列出的字元的原因。當啟動模組時，技術名稱也會顯示於概觀中。

「*公用名稱*」
輸入用來參照 CA 的名稱。

「*電子郵件地址*」
在此可輸入數個電子郵件地址，以便供 CA 使用者查看。這對查詢有所幫助。

「*國家*」
選取進行 CA 操作的國家。

「*組織*」、「*組織單位*」、「*地區*」、「*州*」
選擇性設定值

4 按一下「下一步」「」。

5 在第二個對話方塊中輸入密碼。當建立子CA或產生憑證時，若使用CA，都會要求輸入此密碼。文字欄位代表下列意義：

「*金鑰長度*」
「*金鑰長度*」包含有意義的預設值，除非應用程式無法處理此金鑰長度，否則通常無須變更金鑰長度。

「*有效期限 (天數)*」
CA 預設的「*有效期限*」是 3650 天 (約十年)。長時間是合理的，因為取代刪除的 CA，牽涉到大量的管理工作。

按一下「*進階選項*」可開啟對話方塊，以設定 X.509 延伸的不同屬性 (圖形 42.4「YaST CA 模組 — 延伸設定」 [741頁])。這些值具有合理的預設值。只有當您有相當自信時，才應變更此值。

6 YaST 顯示目前設定以供確認。按一下「*建立*」。根 CA 已建立並隨後顯示在概觀中。

提示

一般來說，最好不要允許根 CA 發行使用者憑證。最好至少建立一個子 CA，並從那裡建立使用者憑證。此方法的優點是可將根 CA 隔離開來以保護其安全，舉例來說，將根 CA 放在隔離的電腦上以加強安全性。在此狀況下，要攻擊根 CA 就變得非常困難。

42.2.2 建立子 CA 或撤銷子 CA

子 CA 的建立方式與根 CA 完全一樣。請進行下列幾項操作：

1. 啟動 YaST 並開啟 CA 模組。
2. 選取所需 CA 並按一下「*輸入 CA*」。

   注

   子 CA 的有效期必須完全在「上層」CA 的有效期之內。因為子 CA 一定是在「上層」CA 之後建立，因此預設值會產生錯誤訊息。若要避免，請輸入有效期限允許值。

3. 如果是第一次輸入 CA，請輸入密碼。YaST 會在索引標籤「*說明*」中顯示 CA 金鑰資訊 (請參閱圖形 42.2)。

圖形 42.2　YaST CA 模組 — 使用 CA

4. 按一下「*進階*」並選取「*建立子 CA*」。此步驟所開啟的對話方塊，和建立根 CA 一樣。
5. 依 第 42.2.1 節「建立根 CA」[735頁] 所述進行。

6 選取索引標籤「*憑證*」。使用「*撤銷*」以在此重新設定外洩的子 CA 或不需要的子 CA。單憑撤銷不足以停用子 CA。還要在 CRL 中發佈撤銷的子 CA。CRL 的建立步驟如 第 42.2.5 節「建立 CRL 」 [741頁] 所述。

7 結束時按一下「*確定*」。

42.2.3 建立或撤銷使用者憑證

建立用戶端與伺服器憑證與第 42.2.1 節「建立根 CA 」[735頁]中的建立 CA 程序十分相近。此處也適用相同的原則。在用於電子郵件簽章的憑證中應該包含寄件者 (私密金鑰擁有者) 的電子郵件地址，電子郵件程式才能指派正確的憑證。若要加密時進行憑證指派，憑證中必須包含收件人 (公用金鑰擁有者) 的電子郵件地址。如果是伺服器和用戶端憑證，則必須在「*公用名稱*」欄位中輸入伺服器的主機名稱。憑證預設有效期限是 365 天。

若要建立用戶端和伺服器憑證，請執行下列步驟：

1 啟動 YaST 並開啟 CA 模組。

2 選取所需 CA 並按一下「*輸入 CA*」。

3 如果是第一次輸入 CA，請輸入密碼。YaST 會在「*說明*」索引標籤中顯示 CA 金鑰資訊。

4 按一下「*憑證*」(請參閱 圖形 42.3「CA 憑證」 [739頁])。

圖形 42.3　CA 憑證

5 按一下「*新增*」>「*新增伺服器憑證*」，並建立伺服器憑證。

6 按一下「*新增*」>「*新增用戶端憑證*」，並建立用戶端憑證。請不要忘記輸入電子郵件地址。

7 結束時按一下「*確定*」。

若要撤銷外洩或不要的憑證，請執行下列步驟：

1 啟動 YaST 並開啟 CA 模組。

2 選取所需 CA 並按一下「*輸入 CA*」。

3 如果是第一次輸入 CA，請輸入密碼。YaST 會在「*說明*」索引標籤中顯示 CA 金鑰資訊。

4 按一下「*憑證*」(請參閱第 42.2.2 節「建立子 CA 或撤銷子 CA」[737頁])。

5 選取要撤銷的憑證，並按一下「*撤銷*」。

6 選擇要撤銷此憑證的原因。

7 結束時按一下「*確定*」。

注

單憑撤銷動作不足以停用憑證。還要在 CRL 中發佈撤銷憑證。第 42.2.5 節「建立 CRL 」 [741 頁]說明了如何建立 CRL。在 CRL 發佈撤銷憑證後，即可使用「*刪除*」將撤銷的憑證完全移除。

42.2.4 變更預設值

前幾節說明了如何建立子CA、用戶端憑證和伺服器憑證。特殊設定則用於X.509憑證的延伸。這些設定已賦予合理的預設值可供所有的憑證類型使用，通常這些設定是不需要變更的。不過，您也許對這些延伸有特定需求。在這種情況下，您可以調整預設值。否則，每次您都要從頭開始建立憑證。

1 啟動 YaST 並開啟 CA 模組。

2 輸入必要 CA，如 第 42.2.2 節「建立子 CA 或撤銷子 CA」 [737頁] 所述。

3 按一下「*進階*」>「*編輯預設值*」。

4 選擇要變更的設定類型。接著會開啟如 圖形 42.4「YaST CA 模組 — 延伸設定」 [741頁] 所示的對話方塊，以供變更預設值。

圖形 42.4　YaST CA 模組 — 延伸設定

5 在右邊變更相關值，並使用「*關鍵*」來設定或刪除關鍵設定。

6 按一下「*下一步*」來檢視簡短摘要。

7 完成變更後按一下「*儲存*」。

提示

所有的預設值變更只會影響此後所建立的物件。已存在的 CA 和 憑證保持不變。

42.2.5 建立 CRL

若有外洩憑證或不要的憑證應加以排除以免之後再次用到，首先必須將這類憑證撤銷。第 42.2.2 節「建立子 CA 或撤銷子 CA」[737頁] (適用於 CA) 與第 42.2.3 節「建立或撤銷使用者憑證」[738頁] (適用於使用者憑證) 中會詳細說明這個程序。之後，必須使用此資訊來建立及發佈 CRL。

系統只會為每個CA維護一個CRL。若要建立或更新此CRL，請執行下列步驟：

1 啟動 YaST 並開啟 CA 模組。

2 輸入必要 CA，如 第 42.2.2 節「建立子 CA 或撤銷子 CA」[737頁] 所述。

3 按一下「*CRL*」。開啟的對話方塊會顯示 此 CA 的上一個 CRL 的摘要。

4 如果您在 CRL 建立之後曾撤銷新的子 CA 或憑證，請使用「*產生 CRL*」來建立新的 CRL。

5 指定新 CRL 的有效期限 (預設值：30 天)。

6 按一下「*確定*」來建立和顯示 CRL。之後，您必須發佈此 CRL。

提示

如果無法取得 CRL 或 CRL 到期，評估 CRL 的應用程式將會拒絕每個憑證。身為 PKI 提供者，您有責任在目前的 CRL 到期之前 (有效期間內)，建立及發佈新的 CRL。YaST 未提供自動化這項程序的功能。

42.2.6 將 CA 物件輸出至 LDAP

執行中的電腦必須透過 YaST LDAP 用戶端來設定，以執行 LDAP 的輸出。此程序會提供執行階段之 LDAP 伺服器資訊供您填寫對話方塊欄位。否則，雖然可以採取輸出方式，所有 LDAP 資料還是必須以手動方式來輸入。您必須輸入數個密碼 (請參閱 表格 42.3「LDAP 輸出期間的密碼」[742頁])。

表格 42.3 *LDAP 輸出期間的密碼*

密碼	代表意義
LDAP 密碼	授權使用者在 LDAP 樹狀結構中建立項目。
憑證密碼	授權使用者輸出憑證。
新的憑證密碼	LDAP 輸出時會使用 PKCS12 格式。此格式會強制指定新的密碼給輸出的憑證。

憑證、 CA 和 CRL 都可輸出至 LDAP。

輸出 CA 到 LDAP

若要輸出 CA，請輸入 CA，如第 42.2.2 節「建立子 CA 或撤銷子 CA」[737頁]所述。在接下來的對話方塊中，選取「*延伸*」>「*輸出至 LDAP*」，這會開啟用來輸入 LDAP 資料的對話方塊。如果您的系統已透過 YaST LDAP 用戶端設定，則這些欄位中有一部分已填入資料。否則，請以手動方式輸入所有資料。利用「caCertificate」屬性，於個別樹的 LDAP 中建立項目。

將憑證輸出至 LDAP

輸入含有要輸出憑證的 CA，接著選取「*憑證*」。從對話方塊上方的憑證清單中，選取需要的憑證，接著選取「*輸出*」>「*輸出至 LDAP*」。使用和 CA 相同的方式，在此處輸入 LDAP 資料。此憑證會與 LDAP 樹狀結構中的相對應使用者物件一起儲存，並包含屬性「userCertificate」(PEM 格式) 和「userPKCS12」(PKCS12 格式)。

輸出 CRL 到 LDAP

輸入含有 CRL 的 CA 以便進行輸出，並選取「*CRL*」。若有需要，請建立新的 CRL 並按一下「*輸出*」。隨即開啟的對話方塊將顯示輸出參數。您可以一次性或在固定的時間間隔內輸出此 CA 的 CRL。透過選取「*輸出到 LDAP*」啟用輸出並分別輸入 LDAP 資料。若要在固定時間間隔內執行此動作，請選取「*重複的重新建立和輸出*」選項圓鈕並變更間隔 (如果適合的話)。

42.2.7 以檔案格式輸出 CA 物件

如果已在電腦上設定了供管理 CA 使用的儲存庫，您可以使用此選項，於正確的位置上直接將 CA 物件建立成一個檔案。可以使用多種不同的輸出格式，例如 PEM、DER 以及 PKCS12。如果是 PEM，也可能選擇輸出憑證時包含還是不包含金鑰，以及金鑰是否應加密。如果是 PKCS12，則您也可以將憑證路徑輸出。

以 LDAP 的相同方法輸出憑證、CA 的檔案，如第 42.2.6 節「將 CA 物件輸出至 LDAP」[742頁]所述，不同之處僅在於要選取「*輸出為檔案*」而非「*輸出到 LDAP*」。接著會出現一個對話方塊，讓您選取必要的輸出格式及輸入密碼和檔案名稱。按一下「*確定*」之後，該憑證會儲存到所需的位置。

對於 CRL，請按一下「*輸出*」，選取「*輸出到檔案*」，選擇輸出格式 (PEM 或 DER) 並輸入路徑。繼續按「*確定*」將其儲存到相應位置。

提示

您可以在任何檔案系統中選取任一儲存位置。這個選項也可以用來將 CA 物件儲存在傳輸媒體上，例如 USB 晶片組。/media目錄通常存放系統硬碟以外的任何磁碟類型。

42.2.8 輸入一般伺服器憑證

如果使用 YaST 將伺服器憑證輸出到獨立 CA 管理電腦上的媒體中，您就可以在伺服器上將此憑證輸入為*一般伺服器憑證*。您可以在安裝期間或稍後使用 YaST 來執行這個動作。

注

您需要使用其中一個 PKCS12 格式才能順利輸入憑證。

一般伺服器憑證儲存於 `/etc/ssl/servercerts`，並且可以透過任何 CA 支援的服務來使用此憑證。當憑證到期時，您可以使用相同的機制來輕易地更換憑證。若要開始使用更換的憑證，請重新啟動相關服務。

提示

如果在此選取「*輸入*」，便可以在檔案系統中選取來源。這個選項也可以用來從傳輸媒體輸入憑證，例如 USB 隨身碟。

若要輸入一般伺服器憑證，請執行下列步驟：

1. 啟動 YaST 並開啟「*安全性與使用者*」下的「*一般伺服器憑證*」。
2. YaST 啟動後，在說明欄位中檢視目前憑證的資料。
3. 選取「*輸入*」和憑證檔案。
4. 輸入密碼，並按一下「*下一步*」。憑證便會輸入並顯示在說明欄位中。
5. 按一下「*完成*」以關閉 YaST。

43 偽裝與防火牆

只要 Linux 用於網路環境，您可以使用允許操作網路封包的核心功能，讓內部和外部網路區域之間保持區隔。Linux netfilter 結構提供建立有效防火牆的方法，將不同的網路分開。利用 iptables (規則集定義的一般表格結構) 的幫助，準確地控制允許傳遞網路介面的封包。這類封包過濾器只要透過 SuSEfirewall2 及對應 YaST 模組的協助即可輕鬆設定。

43.1 使用 iptables 過濾封包

元件 netfilter 與 iptables 負責過濾及操作網路封包，以及網路位址轉譯 (NAT，Network Address Translation)。過濾準則和任何與其關聯的動作以鏈結方式儲存，在個別網路封包抵達時必須相互符合。符合的鏈結儲存於表格內。iptables 指令允許您變更這些表格和規則集。

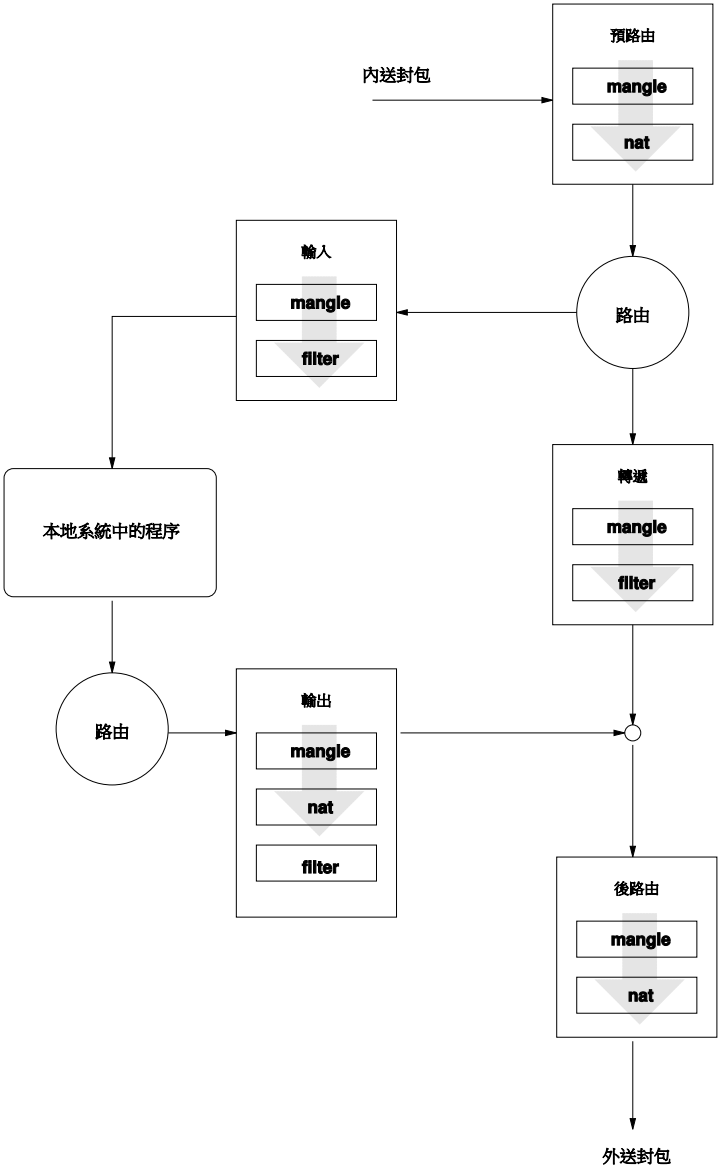
Linux 核心維護三個表格，每個針對封包過濾器的特定功能種類：

filter
: 此表格保存一組過濾器規則，因為它以更嚴格的方式執行*封包過濾 (packet filtering)*機制，例如，決定封包是否允許通過 (ACCEPT) 或放棄 (DROP)。

nat
: 此表格定義對封包之來源和目標位址的任何變更。使用這些功能也可以讓您執行「*偽裝*」*(masquerading)*，這是 NAT 用來連結私有網路與網際網路的一種特殊方式。

mangle
此表格中的規則可以操作儲存於 IP 標頭中的值 (如服務類型)。

*圖形 **43.1** iptables：封包的可能路徑*



這些表格包含數個符合封包的預先定義鏈結：

PREROUTING
: 此鏈結套用到內送封包。

INPUT
: 此鏈結套用到預定送到系統內部程序的封包。

FORWARD
: 此鏈結套用到僅透過系統路由的封包。

OUTPUT
: 此鏈結套用到來自於從系統本身的封包。

POSTROUTING
: 此鏈結套用到所有外送封包。

圖形 43.1「iptables：封包的可能路徑」[746頁]說明網路封包在指定系統上傳送的路徑。為了簡化的緣故，圖中將表格列為鏈結的各部份，但是實際上，這些鏈結是在表格本身內。

所有可能情況中最簡單的一種，系統本身預定的內送封包抵達 `eth0` 介面。封包先參照 `mangle` 表的 `PREROUTING` 鏈結，然後再參照 `nat` 表的 `PREROUTING` 鏈結。下一個步驟有關封包的路由，決定封包的實際目標是系統本身的程序。傳遞 `mangle` 與 `filter` 表的 `INPUT` 鏈結後，封包最後會到達其目標，也就是確定符合 `filter` 表的規則。

43.2 偽裝基本原則

偽裝是 NAT (網路位址轉譯) 的 Linux 專用形式。它可以用來連結小型 LAN (主機使用私用範圍的 IP 位址；請參閱第 30.1.2 節「網路遮罩與路由」[501頁]) 與網際網路 (使用正式 IP 位址)。若要讓 LAN 主機能夠連接網際網路，這些主機的私用位址會轉譯為正式位址。這是在路由器上完成的，路由器當作LAN與網際網路之間的閘道。其根本原則很簡單：路由器具備一個以上的網路介面，一般是網路卡與連接至網際網路的個別介面。後者連結路由器與外界，一或多個其他的則連結路由器與LAN主機。當區域網路中的這些主機連接到路由器的網路卡時(如`eth0`)，它們可以傳送任何不是預定給區域網路的封包到其預設的閘道或路由器。

重要：使用正確的網路遮罩

設定您的網路時，確定廣播位址及網路遮罩在所有本地主機上都是相同的。如果不相同，可能會造成封包無法正確的路由。

如所述，只要其中一個LAN主機傳送預定給網際網位位址的封包時，就會送到預設路由器。不過，路由器必須先設定才能轉遞這類封包。為了安全性起見，預設安裝不會起用此選項。若要啟用，將檔案 `/etc/sysconfig/sysctl` 中的變數 `IP_FORWARD` 設定為 `IP_FORWARD=yes`。

連接的目標主機可以看到您的路由器，但是並不知道您內部網路中產生封包的主機。這就是為什麼這個技術稱為偽裝。由於位址轉譯功能，所以路由器是任何回覆封包的第一個目的地。路由器必須識別這些內送封包並轉譯其目標位址，如此封包可轉遞到區域網路中的正確主機。

依據偽裝表格之內送交通的路由，是不可能從外面開啟對內部主機的連接。對於這類連接，表格中不會出現項目。此外，任何已經建立的連接在表格中都有指定給它的狀態項目，因此另一個連接無法使用此項目。

其結果是，您在一些應用程式通訊協定上可能會出現問題，如 ICQ、cucme、IRC (DCC、CTCP) 和 FTP (PORT 模式)。Web 瀏覽器、標準 FTP 程式，與許多其他程式，都使用 PASV 模式。就封包過濾和偽裝而言，這種被動模式的問題較少。

43.3 防火牆基本原則

「*防火牆*」大概是使用最為廣泛的字詞，用來描述提供及管理網路之間的連結機制，同時還能控制其間的資料流。嚴格來說，本節中描述的機制稱為「*封包過濾器*」。封包過濾器根據特定準則 (如通訊協定、通訊埠和 IP 位址) 規範資料流。這樣允許您根據封包的位址，阻斷不應該送到您網路上的封包。例如，若要允許公用存取您的網頁伺服器，請明確開啟對應連接埠。不過，封包過濾器不會掃描具有正常位址的封包內容，如導向您網頁伺服器的那些封包。例如，即使內送封包意圖危害您網頁伺服器上的 CGI 程式，封包過濾器仍會讓這些封包通過。

更有效但更複雜的機制是數種系統類型的組合，如與應用程式閘道或代理互動的封包過濾器。這個時候，封包過濾器會拒絕預定給停用連接埠的任何封包。只有導向到應用程式閘道的封包才會被接受。此閘道或代理會假裝是伺服器的

實際用戶端。以這種意義而言，這類代理會視為通訊協定層級上應用程式所使用的偽裝主機。這種代理的範例之一是 Squid，一種 HTTP 代理伺服器。若要使用 Squid，瀏覽器必須設定成透過代理進行通訊。要求的任何 HTTP 頁面會從代理快取取得，而快取中找不到的頁面會透過代理從網際網路抓取。另一個範例則是，SUSE proxy-suite (`proxy-suite`) 為 FTP 通訊協定提供 Proxy。

下節將焦點放在 SUSE Linux Enterprise 隨附的封包過濾器。如需有關封包過濾及防火牆的詳細資訊，請參閱 `howto` 套件中包含的「防火牆 HOWTO」內容。如果安裝了此套件，請閱讀 HOWTO

```
less /usr/share/doc/howto/en/txt/Firewall-HOWTO.gz
```

43.4 SUSEfirewall2

SuSEfirewall2 是讀取 `/etc/sysconfig/SuSEfirewall2` 中變數集的程序檔，可以產生一組 iptables 規則。它定義三個安全性區域，但是以下範例組態僅考慮第一個和第二個區域：

外部區域
: 假設沒有方法可以控制外部網路上發生的情況，因此需要保護主機。在大部分的情況中，外部網路就是網際網路，但是也可能是其他不安全的網路，例如 WLAN。

內部區域
: 這是指私用網路，通常是指 LAN。如果此網路上的主機使用私用範圍的 IP 位址 (請參閱第 30.1.2 節「網路遮罩與路由」[501 頁])，請啟用網路位址轉譯 (NAT)，這樣內部網路上的主機即可存取外部網路。

廢除區域 (DMZ)
: 儘管外部及內部網路可以連接位於此區域中的主機，但是這些主機本身無法存取內部網路。這種設定可以在內部網路前加上額外的防護線，因為 DMZ 系統與內部網路是隔離的。

過濾規則集未明確允許的任何網路流量類型會由 iptables 封鎖。因此，具有內送流量的每個介面必須放置在三個區域中的其中一個。對於每個區域，定義允許的服務或通訊協定。規則集僅套用到遠端主機產生的封包。本地產生的封包不會被防火牆攔截。

使用 YaST 可以執行組態 (請參閱第 43.4.1 節「以 YaST 設定防火牆」[750頁])。也可以在檔案 /etc/sysconfig/SuSEfirewall2 中手動進行，該檔案的註解完整。不僅如此，在 /usr/share/doc/packages/SuSEfirewall2/EXAMPLES 中還提供一些範例案例。

43.4.1 以 YaST 設定防火牆

重要：自動防火牆組態

在安裝後，YaST 在所有已設定的介面上會自動啟動防火牆。如果已設定系統並在系統上啟用，YaST 會使用伺服器組態模組中的「*開啟防火牆中選取介面的連接埠*」或「*開啟防火牆的連接埠*」選項，以修改自動產生的防火牆組態。有些伺服器模組對話方塊具有「*防火牆詳細資料*」按鈕，可以啟用其他服務和連接埠。YaST 防火牆組態模組僅能啟用、停用或獨立重新設定防火牆。

YaST 的圖形組態對話方塊可從「YaST 控制中心」存取。選取「*安全性和使用者*」>「*防火牆*」。該組態一共分成七個部份，可以在左手邊的三個結構直接存取。

啟動
: 在此對話方塊中設定啟動行為。預設的安裝中，會自動啟動 SUSEfirewall2。您也可以在此啟動和停止防火牆。若要在執行中的防火牆設定新的設定，請使用「*立即儲存設定並重新啟動防火牆*」。

介面
: 所有已知的網路介面都列在這裏。若要從區域移除介面，請選取介面，按「*變更*」後選擇「*沒有指定區域*」。若要新增介面至區域，請選取介面，按「*變更*」後選擇任何可用的區域。您也可以使用「*自定*」，以自己的設定建立特殊的介面。

允許的服務
: 您需要此選項以從系統提供服務至受保護的區域。按照預設值，只會保護系統不受外部區域的侵犯。明確的規範允許外部區域可使用的服務。在「*選定區域允許的服務*」中選取所需的區域後，啟用清單中的服務。

偽裝
: 偽裝可將內部網路隱身於外部網路，如網際網路，但允許內部網路的主機存取外部網路。從外部網路發出對內部網路的要求會遭到封鎖，但是從內部網

路發出的要求，從外部看起來會像是從偽裝伺服器發出。如果內部機器的特殊服務需要開放給外部網路使用，可以針對服務增加特殊的重新指向規則。

廣播
在此對話方塊中，已設定允許廣播的 UDP 埠。新增必需的連接埠號碼或服務到適當的區域，由空格分隔。請參閱 `/etc/services` 檔案。

不被接受的廣播紀錄可以在此啟動。這有可能會有問題，因為 Windows 主機使用廣播瞭解彼此，也因而產生許多不被接受的封包。

IPsec 支援
在此對話方塊設定 IPsec 服務是否可供外部網路使用。在「*詳細資料*」下，設定可信任的封包。

記錄層級
有兩種登入規則：接受與不接受封包。不被接受的封包包括 DROPPED 或 REJECTED。為這兩個封包選取「*記錄所有*」、「*只記錄關鍵*」或「*完全不記錄*」。

當完成防火牆組態時，請按「*下一步*」結束此對話方塊。開啟防火牆設定的區域導向摘要。在此，勾選所有設定。所有已允許的服務，埠和通訊協定都會列在此摘要中。若要修改組態，使用「*上一步*」。按「*接受*」可儲存您的組態。

43.4.2 手動設定

以下段落提供成功組態的逐步指示。每個組態項目會依據是否與防火牆或偽裝功能相關加以標示。無論是否合適，均使用埠範圍 (例如，`500:510`)。這裡並沒有涵蓋組態檔中所提與 DMZ (廢除區域) 相關的觀點。這些觀點僅適用較大型組織的更複雜網路基礎結構 (公司網路)，因為需要全面的組態以及對該主題的深入瞭解。

首先，使用 YaST 模組系統服務 (Runlevel) 以啟動 runlevel 中的 SUSEfirewall2 (最有可能是 3 或 5)。它在 `/etc/init.d/rc?.d/` 目錄中會設定 SUSEfirewall2_* 程序檔的符號連結。

`FW_DEV_EXT` (防火牆、偽裝)
連結到網際網路的設備。如果是數據機連接，請輸入 `ppp0`。如果是 ISDN 連結，請使用 `ippp0`。DSL 連接則使用 `dsl0`。指定 `auto` 使用與預設路由對應的介面。

FW_DEV_INT (防火牆、偽裝)
連結到內部私用網路的設備 (如 eth0)。如果沒有內部網路且防火牆僅保護執行的主機時，請留置空白。

FW_ROUTE (防火牆、偽裝)
如果需要偽裝功能，將此設定為 yes。外界將無法看到您的內部主機，因為網際網路路由器會忽略其私用網路位址 (例如，192.168.x.x)。

對於沒有偽裝的防火牆，如果想要允許存取內部網路，僅將此設定為 yes。在這種情況下，您的內部主機需要使用正式註冊的 IP 位址。不過，通常您「*不*」應該允許外界存取您的內部網路。

FW_MASQUERADE (偽裝)
如果需要偽裝功能，將此設定為 yes。這可以提供網際網路主機，建立虛擬/直接的連接到網際網路。在內部網路和網際網路的主機之間使用代理伺服器較為安全。對於代理伺服器所提供的服務並不需要偽裝。

FW_MASQ_NETS (偽裝)
指定要偽裝的主機或網路，在個別項目之間加上空格。例如：

```
FW_MASQ_NETS="192.168.0.0/24 192.168.10.1"
```

FW_PROTECT_FROM_INT (防火牆)
將此設定為 yes，保護您的防火牆主機免於來自內部網路的攻擊。如果確實啟用，只有內部網路才可以使用服務。另請參閱 FW_SERVICES_INT_TCP 與 FW_SERVICES_INT_UDP。

FW_SERVICES_EXT_TCP (防火牆)
輸入應該可用的 TCP 連接埠。對於不應該提供任何服務的一般家用工作站，請留置空白。

FW_SERVICES_EXT_UDP (防火牆)
除非執行 UDP 服務且希望讓外界使用，否則請留置空白。使用 UDP 的服務包括 DNS 伺服器、IPsec、TFTP、DHCP 以及其他。如果要讓外界使用，輸入要使用的 UDP 連接埠。

FW_SERVICES_INT_TCP (防火牆)
利用此變數，定義可讓內部網路使用的服務。其表示法與 FW_SERVICES_EXT_TCP 相同，但是設定是套用到「*內部*」網路。只有在 FW_PROTECT_FROM_INT 設定為 yes 時，才需要設定變數。

FW_SERVICES_INT_UDP (防火牆)
請參閱 FW_SERVICES_INT_TCP。

設定防火牆後，測試您的設定。防火牆規則集的建立是以 root 身份輸入 SUSEfirewall2 start。接著使用 telnet，例如，從外部主機查看是否已實際拒絕連接。之後，檢視 /var/log/messages，在其中應該會看到如下內容：

```
Mar 15 13:21:38 linux kernel: SFW2-INext-DROP-DEFLT IN=eth0
OUT= MAC=00:80:c8:94:c3:e7:00:a0:c9:4d:27:56:08:00 SRC=192.168.10.0
DST=192.168.10.1 LEN=60 TOS=0x10 PREC=0x00 TTL=64 ID=15330 DF PROTO=TCP
SPT=48091 DPT=23 WINDOW=5840 RES=0x00 SYN URGP=0
OPT (020405B40402080A061AFEBC0000000001030300)
```

其他測試防火牆設定的套件為 nmap 或 nessus。安裝各自的套件後，nmap 的文件可在 /usr/share/doc/packages/nmap 中找到，而 nessus 的文件則位於 /usr/share/doc/packages/nessus-core 目錄中。

43.5 如需更多資訊

有關 SUSEfirewall2 套件的最新資訊及其他文件請參閱 /usr/share/doc/packages/SuSEfirewall2。netfilter 與 iptables 專案的首頁 (http://www.netfilter.org) 提供許多語言的大量文件。

44 SSH：安全性網路作業

隨著有愈來愈多的電腦安裝在網路環境中，通常會需要從遠端位置存取主機。這通常是指使用者會傳送登入與密碼字串以供驗證。只要這些字串是以純文字傳送，它們就有可能被攔截和濫用以取得使用者帳戶的存取權，而授權的使用者卻完全未察覺。除了這將會把使用者的所有檔案暴露給攻擊者之外，不合法的帳戶有可能取得管理員或 `root` 存取權，或是滲透其他的系統。在過去，遠端連接是使用 telnet 建立，它並未提供對抗監聽加密形式或其他安全性機制的防護。有一些通訊通道是未防護的，像是傳統的 FTP 通訊協定以及某些遠端的複製程式。

SSH 套裝軟體可以提供必要的保護，即加密驗證字串 (通常是登入名稱與密碼) 以及所有在主機間的其他資料交換。有了 SSH，即使第三方記錄了資料流，但是內容已被加密，除非知道加密金鑰，否則無法將內容回復至純文字。因此 SSH 可以在非安全的網路上啟用安全通訊，例如網際網路。SUSE Linux Enterprise 隨附的 SSH 為 OpenSSH。

44.1 OpenSSH 套件

SUSE Linux Enterprise 預設會安裝 OpenSSH 套件。ssh、scp 以及 sftp 程式就變成了 telnet、rlogin、rsh、rcp 以及 ftp 的替代程式。在預設的組態中，只有使用 OpenSSH 公用程式且防火牆允許時，才能存取 SUSE Linux Enterprise 系統。

44.2 ssh 程式

使用 ssh 程式，就有可能登入遠端系統並互動式地工作。它會取代 telnet 與 rlogin。slogin 程式只是指向 ssh 的符號連結。例如，使用 `sshsun` 指令登入 sun 主機。該主機接著會提示輸入 sun 的密碼。

在成功地驗證後，即可使用遠端指令行，或使用互動式應用程式，例如 YaST。如果本地使用者名稱與遠端使用者名稱不同，您可以使用不同的登入名稱加上 `ssh -l augustine sun` 或 `ssh augustine@sun`。

此外，ssh 提供從遠端系統 rsh 上執行指令的可能性。在下列範例中，在 sun 主機上執行 `uptime` 指令，並建立名稱為 `tmp` 的目錄。程式輸出會顯示在 earth 主機的本地終端機上。

```
ssh otherplanet "uptime; mkdir tmp"
Password:
1:21pm  up  2:17,  9 users,  load average: 0.15, 0.04, 0.02
```

在這裏需要使用引號以使用一個指令傳送兩個指示。只有執行此動作，第二個指令才可以在 sun 上執行。

44.3 scp—安全複製

scp 會將檔案複製到遠端機器。它是 rcp 的安全與加密的替代指令。For example, `scp MyLetter.tex sun:` copies the file `MyLetter.tex` from the host earth to the host sun. 如果在 earth 上的使用者名稱與 sun 上的使用名稱不相同，請使用 `username@host` 格式指定後者的使用者名稱。此指令並沒有 `-l` 的選項。

在輸入正確的密碼後，scp 會啟動資料傳輸並顯示逐漸變長的星號列，以模擬進度列。除此之外，程式會永遠在進度列的右邊顯示到達的估計時間。指定 `-q` 選項以抑制所有的輸出。

scp 也會為整個目錄提供遞迴複製功能。`scp -r src/ sun:backup/` 指令會複製 `src` 目錄的整個內容 (包括所有的子目錄) 至 sun 主機上的 `backup` 目錄。如果這個子目錄不存在，就不會自動建立。

`-p` 選項會指示 scp 保留未變更檔案的時間戳記。`-C` 可以壓縮資料傳輸。這可能會減少要傳輸的資料量，但是會造成處理器更重的負擔。

44.4 sftp—安全檔案傳輸

可以使用 sftp 程式取代 scp，以進行安全的檔案傳輸。在 sftp 工作階段期間，可以使用許多已知的 ftp 指令。sftp 程式可能是比 scp 更好的選擇，特別是在傳輸未知檔名的資料時。

44.5 SSH 精靈 (sshd)—伺服器端

若要使用 SSH 用戶端程式 ssh 與 scp，伺服器 (SSH 精靈) 必須在背景執行，監聽 `TCP/IP port 22` 的連接。精靈會在第一次啟動時產生三個金鑰組合。每個金鑰組合都是由私密與公用金鑰所組成。為了保證透過 SSH 通訊的安全性，必須限制只有系統管理員可以存取私密金鑰檔。預設安裝會據此設定檔案權限。SSH 精靈只有在本地上需要私密金鑰，而且絕對不能將它給予任何人。公用金鑰元件 (以 `.pub` 的副檔名來識別) 會傳送至要求連接的用戶端。它們可讓所有的使用者讀取。

SSH 用戶端所啟始的連接。等待 SSH 精靈與要求 SSH 用戶端交換識別資料，以比較通訊協定與軟體版本，並防止透過錯誤埠的連接。因為原始 SSH 精靈的子處理序會回覆要求，所以可以同時建立數個 SSH 連接。

至於 SSH 伺服器與 SSH 用戶端之間的通訊，OpenSSH 可以支援 SSH 通訊協定的版本1 與 2。預設會使用 SSH 通訊協定的版本 2。使用 `-1` 參數即可覆寫此預設以使用通訊協定的版本 -1。若要在系統更新後繼續使用版本 1，請遵循 `/usr/share/doc/packages/openssh/README.SuSE` 中的指示。這個文件也描述如何只執行一些步驟，即將 SSH 1 環境轉換成工作 SSH 2 環境。

當使用 SSH 的版本 1 時，伺服器會傳送每小時由 SSH 精靈重新產生的公用主機金鑰與伺服器金鑰。這兩者都允許 SSH 用戶端加密自由選擇的工作階段金鑰，並且會將它傳送至 SSH 伺服器。SSH 用戶端也會指示伺服器要使用的加密方法 (密碼)。

SSH 通訊協定的版本 2 並不需要伺服器金鑰。兩邊都是使用根據 Diffie-Helman 的演算法來交換金鑰。

私密主機與伺服器金鑰絕對需要解密工作階段的金鑰，而且不能從公用部份產生。只有聯絡的 SSH 精靈可以使用私密金鑰解密工作階段金鑰 (請參閱

man/usr/share/doc/packages/openssh/RFC.nroff)。若要仔細監看此啟始連接階段，可以開啟 SSH 用戶端的 -v 詳細偵錯選項。

用戶端會在第一次與遠端主機連繫後，將所有的公用主機金鑰儲存在 ~/.ssh/known_hosts 中。這將可防止外來的 SSH 伺服器進行「在中間攔截的攻擊」(外部 SSH 伺服器使用假冒的名稱與 IP 位址)。這類的攻擊通常都是由下列方式所偵測出來：在 ~/.ssh/known_hosts 中未包含主機金鑰，或伺服器在缺少適當的私密對照組時無法解密工作階段的金鑰。

建議將儲存在 /etc/ssh/ 中的私密與公用金鑰備份在安全的外部位置。以此方式，就可以偵測出金鑰的修改，並且可以在重新安裝後，再度使用舊的金鑰。這可讓使用者避免收到任何擾人的警告。儘管出現警告，但是如果已確認它確實是正確的 SSH 伺服器，就必須從 ~/.ssh/known_hosts 移除關於此系統的現有項目。

44.6 SSH 驗證機制

目前以最簡單的形式所執行的實際驗證，是由上方所提及的輸入密碼所組成。SSH 的目標就是要引進安全且易於使用的軟體。因為它是用以取代 rsh 與 rlogin，SSH 也必須能夠提供適於平日使用的驗證方法。SSH 是藉由使用者所產生的另一個金鑰組合來完成此動作。SSH 套件為此提供了說明程式：ssh-keygen。在輸入 ssh-keygen -t rsa 或 ssh-keygen -t dsa 之後，就會產生金鑰組合，而且會提示您輸入儲存金鑰的基本檔案名稱。

確認預設值並回覆通關密語的要求。即使軟體建議空白的通關密語，一般建議輸入 10 至 30 個字元的文字以描述在此的程序。請勿使用簡短的文字或片語。重複通關密語以確認。接著，您將會看到儲存私密與公用金鑰的位置，在此例中，為 id_rsa 與 id_rsa.pub 檔案。

使用 ssh-keygen -p -t rsa 或 ssh-keygen -p -t dsa 以變更舊的通關密語。複製公用金鑰元件 (在範例中為 id_rsa.pub) 至遠端機器，並將它儲存至 ~/.ssh/authorized_keys。下次建立連接時，將會要求您驗證自己的通關密語。如果沒有要求您驗證，請驗證這些檔案的位置與內容。

就長期而言，此程序比每次給密碼更為麻煩。因此，SSH 套件提供另一個工具：ssh-agent。此工具可保留了 X 工作階段期間的私密金鑰。整個 X 工作階段是啟動成 ssh-agent 的子處理序。執行這個的最簡單的方法就是在 .xsession 檔案

的開頭將 usessh 變數設為 yes，並透過顯示管理員 (例如 KDM 或 XDM) 登入。或者，輸入 ssh-agentstartx。

現在您可以照常使用 ssh 或 scp。如果您已依照上方所述配送公用金鑰，就不會再提示您輸入密碼。請利用 xlock 之類的密碼保護應用程式，來終止或鎖定 X 工作階段。

在推出 SSH 通訊協定版本 2 後所造成的相關變更，也記載在 /usr/share/doc/packages/openssh/README.SuSE 檔案中。

44.7 X，驗證與轉寄機制

除了先前所述的安全性相關改善之外，SSH 也會簡化遠端 X 應用程式的使用。如果您以 -X 選項執行 ssh，就會自動在遠端機器上設定 DISPLAY 變數，而且所有的 X 輸出都會透過現有的 SSH 連接輸出至遠端機器。同時，未獲授權的個體將無法攔截以此方式檢視，並在遠端和本地上啟動的 X 應用程式。

透過新增 -A 選項，ssh-agent 驗證機制就會延續至下一部機器。以此方式，您不需輸入密碼就可以在不同的機器上工作，但僅限於您已配送公用金鑰至目的主機並將它適當地儲存在主機上。

兩種機制都是在預設值中停用，但是也可以隨時在整個系統的 /etc/ssh/sshd_config 組態檔中或使用者的 ~/.ssh/config 中啟用。

ssh 也可以用於重新導向 TCP/IP 連接。在下列範例中，設定 SSH 分別重新導向 SMTP 與 POP3 埠：

```
ssh -L 25:sun:25 earth
```

使用此指令後，任何導向 earth 埠 25 (SMTP) 的連接都會透過加密通道重新導向至 sun 上的 SMTP 埠。這對於那些使用 SMTP 伺服器但卻沒有 SMTP-AUTH 或 POP-before-SMTP 功能的連接特別有用。從任何連接至網路的位置，都可以將電子郵件傳送至「主」郵件伺服器以進行傳遞。同樣地，在 earth 上的所有 POP3 要求 (埠 110) 都可以使用此指令轉寄至 sun 的 POP3 埠：

```
ssh -L 110:sun:110 earth
```

兩個指令都必須以 root 的身份執行，因為該連接必須以有權限的本地埠建立。電子郵件是由一般的使用者在現有的 SSH 連接中傳送和擷取。SMTP 與 POP3 主機都必須設成 localhost 才能正常運作。上方所述的每個程式的其他資訊

都可以在手冊頁面中找到，也可以在 /usr/share/doc/packages/openssh 下的檔案中找到。

45 網路驗證—Kerberos

除了通用的密碼機制之外，開放式網路沒有其他方式可確保工作站能正確識別它的使用者。在一般安裝中，每次存取網路內的服務時，使用者都必須輸入密碼。Kerberos 提供一種驗證方法，讓使用者只需註冊一次，之後其餘的工作階段在整個網路中皆受到信任。要有安全網路必須符合下列需求：

- 要求所有使用者證明他們的識別身份之後，才能取得每項想要的服務，並確定他人沒有盜用該識別身份。
- 確定每台網頁伺服器也能證明其識別身份。否則攻擊者有可能會假扮成伺服器，並取得傳輸給伺服器的機密資訊。此概念稱為「*相互驗證*」，因為用戶端會驗證伺服器，而且反之亦然。

Kerberos 可提供強化的加密驗證，協助您達成這些需求。下列會顯示達成的方式。此處只會討論 Kerberos 的基本原則。如需詳細的技術說明，請參閱 Kerberos 執行方式所提供的文件。

45.1 Kerberos 術語

下列詞彙定義部分 Kerberos 術語。

身份證明
: 使用者或用戶端需要出示一些授權他們要求服務的身份證明。Kerberos 可辨識兩種身份證明 — 票證與授權者。

票證

票證是每一台伺服器的身份證明，用戶端會在向其要求服務的伺服器上使用以供驗證。它包含伺服器名稱、用戶端的名稱、用戶端的網際網路位址、時間戳記、存在時間以及隨機的工作階段金鑰。所有這類資料都會使用伺服器金鑰來加密。

授權者

授權者是結合票證，用來證明持有票證的用戶端，的確是其所宣稱的身份。授權者是由用戶端的名稱、工作站的 IP 位址與目前工作站的時間，全都以工作階段金鑰加密而建立的，該金鑰只有用戶端與向其要求服務的伺服器知道。不像票證，授權者只能使用一次。用戶端可以自行建立授權者。

原則

Kerberos 原則是可指定給票證的唯一實體 (使用者或服務)。原則是由下列元件所組成：

- **主要**—原則的第一部分，就使用者而言，此部分可以與使用者名稱相同。
- **例項**—一些可描述「主要」部分的選擇性資訊。此字串會以 / 來與「主要」部分分隔。
- **領域**—指定您的 Kerberos 領域。領域 (Realm) 通常就是以大寫字母書寫的網域 (domain) 名稱。

相互驗證

Kerberos 可確保用戶端與伺服器雙方都能確定彼此的識別身份。它們會共用工作階段金鑰，來與彼此安全通訊。

工作階段金鑰

工作階段金鑰是由 Kerberos 產生的暫時私密金鑰。用戶端會知道這些金鑰，而且用來加密用戶端與伺服器間的通訊，以供要求與接收票證。

重播

在網路中傳送的所有訊息幾乎會被偷聽、竊取和重送。如果攻擊者設法取得包含您票證與授權者的服務要求，在 Kerberos 內容中，這是最危險的情況。攻擊者之後可以重送該要求 (「*重播*」) 假扮成您的身份。不過，Kerberos 會執行數種機制來處理該問題。

伺服器或服務

「*服務*」是用來稱呼所執行的特定動作。此動作背後的程序則稱為*「伺服器」*。

45.2 Kerberos 的運作方式

Kerberos 通常稱為協力廠商信任驗證服務，表示所有用戶端都信任 Kerberos 對於另一個用戶端識別身份的判斷。Kerberos 會將所有使用者與他們的私密金鑰存放在資料庫。

若要確保 Kerberos 確實值得全然信任，請在專屬的機器上執行驗證與票證授予伺服器。請確定只有管理員能夠透過網路以及實際來存取此機器。將其上執行的 (網路) 服務儘可能減到最少—甚至不要執行 sshd。

45.2.1 一開始接觸

一開始接觸 Kerberos，會與一般網路系統的登入程序相當相似。輸入您的使用者名稱。這部分的資訊與票證授予服務的名稱會傳送到驗證伺服器 (Kerberos)。如果驗證伺服器知道您的存在，就會隨機產生工作階段金鑰，讓您之後可在用戶端與票證授予伺服器之間使用。現在驗證伺服器會為票證授予伺服器準備票證。票證會包含下列資訊—所有資訊都會以工作階段金鑰加以加密，而只有驗證伺服器與票證授予伺服器知道此工作階段金鑰：

- 用戶端與票證授予伺服器的名稱
- 目前的時間
- 指定給此票證的存在時間
- 用戶端的 IP 位址
- 新產生的工作階段金鑰

之後此票證會再次以加密的形式，連同工作階段金鑰傳回給用戶端，但這次會使用用戶端的私密金鑰。因為此私密金鑰是從您的使用者密碼衍生，所以只有 Kerberos 與用戶端知道。現在當用戶端收到此回應之後，就會提示您輸入密碼。此密碼會轉換成金鑰，而該金鑰可用來解密驗證伺服器所傳送的封包。「解開」封包之後，就會從工作站記憶體將密碼與金鑰刪除。只要指定給該票證用以取得其他票證的存在時間尚未過期，工作站就可以證明您的識別身份。

45.2.2 要求服務

若要向網路中的任何伺服器要求服務，用戶端應用程式就需要向該伺服器證明其識別身份。因此，該應用程式會產生授權者。授權者是由下列元件所組成：

- 用戶端的原則
- 用戶端的 IP 位址
- 目前的時間
- Checksum (由用戶端選擇)

用戶端已收到專供此特殊伺服器使用的工作階段金鑰，可用來加密所有這類資訊。伺服器的授權者與票證就會傳送給該伺服器。伺服器可使用它的工作階段金鑰複本來將授權者解密，提供有關用戶端要求服務所需的全部資訊，以與票證中包含的資訊相比較。伺服器會檢查該票證與授權者是否來自相同的用戶端。

伺服器端若沒有執行任何安全性方法，此程序階段會是重播攻擊的絕佳目標。嘗試傳送先前從網路上竊取的要求就可達到攻擊目的。為了避免這種情況，伺服器不會接受具有先前接收過的時間戳記與票證的任何要求。除此之外，要求的時間戳記若與接收要求當時的時間相距過遠的話，就會忽略該項要求。

45.2.3 相互驗證

Kerberos 驗證可以雙向使用。不僅用戶端必須是其所宣稱的身份。伺服器也應該要能夠向要求服務的用戶端驗證其身份。因此，伺服器會自行傳送一些授權者。它會在用戶端授權者接受的Checksum加入一個授權者，並使用與用戶端共用的工作階段金鑰來加密。用戶端會將此當作伺服器驗證的證明，而且雙方會開始協同運作。

45.2.4 票證授予—聯絡所有伺服器

票證的設計是只能在一台伺服器上使用一次。這意味著每次您要求另一項服務時，就必須取得新的票證。Kerberos 會執行機制以取得個別伺服器的票證。此服務稱為「票證授予服務」。票證授予服務與之前提過的其他服務一樣都是服務，因此請使用已略述的相同存取協定。只要應用程式需要尚未要求過的票證時，就會聯絡票證授予伺服器。此要求是由下列元件所組成：

- 要求的原則
- 票證授予票證
- 授權者

如同其他伺服器一樣，票證授予伺服器會檢查票證授予票證以及授權者。如果視為有效，票證授予伺服器會建立新的工作階段金鑰，以供原始的用戶端與新伺服器之間使用。所建立的新伺服器票證會包含下列資訊：

- 用戶端的原則
- 伺服器的原則
- 目前的時間
- 用戶端的 IP 位址
- 新產生的工作階段金鑰

指定給新票證的存在時間，會短於票證授予票證與服務預設值的剩餘存在時間用戶端會接收到票證授予服務傳送的此一票證與工作階段金鑰，但這次會以原始票證授予票證所隨附的工作階段金鑰來將回應加密。當聯絡到新服務時，用戶端無需取得使用者的密碼，即可將該回應解密。Kerberos 因而可為用戶端取得後續票證，而無需在登入時麻煩使用者多次。

45.2.5 與 Windows 2000 的相容性

Windows 2000 包含 Microsoft 的 Kerberos 5 執行方式。由於 SUSE Linux Enterprise® 使用 Kerberos 5 的 MIT 實做，因此可在 MIT 說明文件中找到實用的資訊與指南。請參閱第 45.4 節「如需更多資訊」[766頁]。

45.3 Kerberos 的使用者觀點

觀念上，唯一會與在工作站登入期間發生的 Kerberos 有所聯絡的只有使用者。登入程序包括取得票證授予票證。登出時，就會自動摧毀使用者的 Kerberos 票證，防止其他人在未登入的情形下假扮此位使用者。如果使用者的登入工作階段比指定給票證授予票證的期限還要長 (適當的設定為 10 小時)，自動摧毀票證

可能會產生些許不便。不過，使用者可以執行 `kinit` 來取得新的票證授予票證。請再次輸入密碼，然後 Kerberos 無需額外進行驗證即可存取想要的服務。執行 `klist`，可看見由 Kerberos 自動取得的所有票證清單。

此處有一些使用Kerberos驗證的應用程式清單。這些應用程式可在 `/usr/lib/mit/bin` 或 `/usr/lib/mit/sbin` 下找到。它們全都有一般 UNIX 與 Linux 同類程式的完整功能，另外加上由 Kerberos 管理的透明驗證：

- telnet、telnetd
- rlogin
- rsh、rcp、rshd
- ftp、ftpd
- ksu

使用這些應用程式無需再輸入您的密碼，因為Kerberos已經證明您的識別身份。如果使用Kerberos支援編譯ssh，甚至可將為工作站取得的所有票證轉送給另一台工作站。如果使用 ssh 登入另一台工作站，ssh 可確保能夠針對新的情況調整加密的票證內容。只是在工作站之間複製票證並不夠用，因為票證包含工作站特有的資訊(IP 位址)。XDM、GDM 和 KDM 亦提供 Kerberos 支援。如需 Kerberos 網路應用程式的更多資訊，請參閱 `http://web.mit.edu/kerberos` 的 *Kerberos V5 UNIX User's Guide*

45.4 如需更多資訊

MIT Kerberos 的官方網站為 `http://web.mit.edu/kerberos`。您可由此找到與 Kerberos 有關的其他資源連結，包括 Kerberos 安裝、使用者與管理員指南。

`ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS` 的論文相當詳盡地說明 Kerberos 基本原則，而不會過於難懂。也提供許多進一步調查與閱讀 Kerberos 的機會。

正式的 Kerberos 常見問題可在 `http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html` 取得。由 Brian Tung 所著的 *Kerberos—A Network Authentication System*一書 (ISBN 0-201-37924-4) 能提供廣泛的資訊。

46 安裝與管理 Kerberos

本小節涵蓋 MIT Kerberos 執行方式的安裝，以及部分的管理。本小節假設您已熟悉 Kerberos 的基本概念（另請參閱 第 45 章「*網路驗證—Kerberos*」[761頁]小節）。

46.1 選擇 Kerberos 領域

Kerberos 安裝的範圍稱為領域 (realm)，依其名稱來識別，如：`FOOBAR.COM` 或僅 `ACCOUNTING`。Kerberos 有大小寫之分，因此 `foobar.com` 和 `FOOBAR.COM` 實際上是不同領域。請使用您偏好的大小寫。但是，一般的方式是使用大寫領域名稱。

使用您的 DNS 網域名稱 (或子網域，如 `ACCOUNTING.FOOBAR.COM`) 也是好主意。如下所示，如果您設定您的 Kerberos 用戶端透過 DNS 找出 KDC 和其他 Kerberos 服務，則管理將非常容易。若要這樣做，如果您的領域名稱是您 DNS 網域名稱的子網域會很有用。

和 DNS 名稱空間不同，Kerberos 非階層式。您無法設定名為 `FOOBAR.COM` 的領域，其下含有兩個名為 `DEVELOPMENT` 和 `ACCOUNTING` 的「子領域」，並期望兩個次級領域繼承任何來自 `FOOBAR.COM` 的主體。相反地，您應該分別建立三領域，並為不同領域之間的使用者或伺服器，設定跨領域驗證。

為了簡化範例，假設您僅為整個組織設定一個領域。在本小節的其他部分，會在所有範例中使用 `EXAMPLE.COM` 做為領域名稱。

46.2 設定 KDC 硬體

使用Kerberos需要的第一件事是可做為金鑰配送中心 (簡稱KDC) 的機器。此機器包含整個 Kerberos 使用者資料庫，以及密碼與所有資訊。

KDC 是安全性基礎結構中最重要的部分 — 如果有人突破 KDE 的防線，所有使用者帳戶和您所有受到 Kerberos 保護的基礎結構都會遭到危害。具有 Kerberos 資料庫存取權的攻擊者可模擬資料庫中的任何主體。請儘可能加強使此機器的安全性：

1. 將伺服器機器放在實體安全的位置，例如只有少數人可進入的上鎖伺服器機房。

2. 請勿在其上執行 KDC 以外的任何網路應用程式。此原則適用於伺服器和用戶端 — 例如，KDC 不應透過 NFS 輸入任何檔案系統，或使用 DHCP 來擷取其網路組態。

3. 請先安裝最小系統，然後檢查安裝套件的清單，再移除不需要的套件。此原則適用於伺服器 (如 inetd、portmap 和 cups) 以及任何以 X 為基礎的套件。即使是安裝 SSH 伺服器也必須考量潛在的安全性風險。

4. 此機器不提供圖形登入，因為X伺服器具有潛在的安全性風險。Kerberos 提供自己的管理介面。

5. 設定 `/etc/nsswitch.conf` 僅使用本地檔案進行使用者和群組查詢。將 `passwd` 和 `group` 這一行變更成像這樣：

   ```
   passwd:         files
   group:          files
   ```

   編輯 `/etc` 中的 `passwd`、`group`、`shadow` 和 `gshadow` 檔案，並移除以 + 字元開頭的幾行（這些是用於 NIS 查詢）。

6. 編輯 `/etc/shadow` 並以 `*` 或 ! 字元取代雜湊密碼。

46.3 時鐘同步化

若要成功地使用 Kerberos，請確認您組織內的所有系統時間均同步於特定範圍中。這是很重要的，因為 Kerberos 可防護重播證件的攻擊。攻擊者可在網路上觀察 Kerberos 證件，並重新使用它們來攻擊伺服器。Kerberos 使用數種防禦來避免此攻擊。其中一個是在票證中放入時間戳記。接收到的票證之時間戳記若與目前時間不同，伺服器會拒絕票證。

Kerberos 在比較時間戳記時，允許特定的時間差。但是，電腦時鐘在準確性方面較為不足—PC 時鐘在一星期內多出或少了一小時，並不罕見。基於此因素，請以中央時間來源設定網路上所有主機，來同步化它們的時鐘。

簡單的方式是在一台機器上安裝 NTP 時間伺服器，並讓所有用戶端以此伺服器同步化它們的時鐘。若要如此，可在所有機器上以用戶端模式執行 NTP 精靈，或在所有用戶端一天執行一次 `ntpdate` (此解決方案可能僅適用於用戶端數量少者)。KDC 本身也必須與共同時間來源同步化。因為在此機器上執行 NTP 精靈會有安全性風險，透過 cron 項目執行 ntpdate 來完成可能是個好主意。若要將您的電腦設定為 NTP 用戶端，請執行 第 32.1 節「使用 YaST 設定 NTP 用戶端」[553頁] 中描述的程序。

在檢查時間戳記時，也可以調整 Kerberos 允許的最大誤差。此值 (稱為*時鐘偏移*) 可透過 `krb5.conf` 檔案設定，如第 46.5.3 節「調整時鐘偏移」[774頁] 中所述。

46.4 設定 KDC

本小節涵蓋 KDC 的起始設定與安裝，包括管理主體的建立。此程序包含多個步驟：

1. **安裝 RPM**　在專用為 KDC 的電腦上，安裝特殊軟體套件。如需詳細資料，請參閱 第 46.4.1 節「安裝 RPM」[770頁]。

2. **調整組態檔案**　您必須依您的案例調整組態檔案 `/etc/krb5.conf` 和 `/var/lib/kerberos/krb5kdc/kdc.conf`。這些檔案包含 KDC 上的所有資訊。

3. **建立 Kerberos 資料庫**　Kerberos 會維護一個資料庫，內有所有主體識別碼，以及驗證所有主體所需的秘密金鑰。

4 **調整 ACL 檔案：新增管理員**　KDC 上的 Kerberos 資料庫可遠端管理。為了防止未驗證的主體篡改資料庫，Kerberos 使用存取控制清單。您必須明確的啟用管理員主體的遠端權限，管理員才能對資料庫進行遠端管理。

5 **調整 Kerberos 資料庫：新增管理員**　您必須具備至少一個管理員主體，才能執行並管理 Kerberos。此主體必須在啟動 KDC 之前新增。

6 **啟動 Kerberos 精靈**　安裝 KDC 並正確設定之後，請啟動 Kerberos 精靈，為您的領域提供 Kerberos 服務。

7 **建立您自己的主體。**

46.4.1 安裝 RPM

在開始之前，請安裝 Kerberos 軟體。在 KDC 上，安裝套件 `krb5`、`krb5-server` 和 `krb5-client`。

46.4.2 設定資料庫

下一步是啟始資料庫，Kerberos 將所有關於主體的資料保存在其中。請設定資料庫主要金鑰，這是用來保護資料庫免於意外洩漏，特別是在備份到磁帶時。主要金鑰衍生自通行短語 (pass phrase)，儲存在名為貯藏檔 (stash file) 的檔案中。因此在您每次啟動 KDC 時，不須重新鍵入密碼。請確認您選擇良好的通行短語，例如來自書本中的句子。

當您在製作 Kerberos 資料庫 (`/var/lib/kerberos/krb5kdc/principal`) 的磁帶備份時，請不要備份貯藏檔 (在 `/var/lib/kerberos/krb5kdc/.k5.EXAMPLE.COM`)。否則，每一個可以讀取磁帶的人都可以解密資料庫。因此，最好將通行短語的副本保存在安全或其他受到安全的位置，因為在損毀之後，您將需要它才能從備份磁帶復原資料庫。

若要建立貯藏檔與資料庫，請執行：

```
$> kdb5_util create -r EXAMPLE.COM -s
Initializing database '/var/lib/kerberos/krb5kdc/principal' for realm
'EXAMPLE.COM',
master key name 'K/M@EXAMPLE.COM'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
```

```
Enter KDC database master key:  <= Type the master password.
Re-enter KDC database master key to verify:  <= Type it again.
$>
```

若要確定它所做的事，請使用 list 指令：

```
$>kadmin.local
kadmin> listprincs
K/M@EXAMPLE.COM
kadmin/admin@EXAMPLE.COM
kadmin/changepw@EXAMPLE.COM
krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

這將會顯示目前在資料庫的主體數目。這些均供 Kerberos 內部使用。

46.4.3 建立主體

下一步，建立兩個 Kerberos 主體：一個一般主體用於日常工作，另一個用於 Kerberos 相關的管理任務。假設您的登入名稱為 `newbie`，請繼續如下：

```
kadmin.local

kadmin> ank newbie
newbie@EXAMPLE.COM's Password: <type password here>
Verifying password: <re-type password here>
```

接著，在 `kadmin` 提示輸入 `anknewbie/admin`，建立名為 `newbie/admin` 的另一個主體。在您使用者名稱後的 `admin` 字尾是*角色*。稍後，會在管理 Kerberos 資料庫時使用此角色。使用者對於不同用途可有數個角色。角色基本上是有相似名稱的完全不同帳戶。

46.4.4 啟動 KDC

啟動 KDC 精靈與 kadmin 精靈。若要手動啟動精靈，請輸入 `rckrb5kdc start` 和 `rckadmind start`。亦請使用指令 `insserv krb5kdc` 和 `insserv kadmind`，確認 KDC 和 kadmind 預設在伺服器重新開機時會啟動。

46.5 手動設定 Kerberos 用戶端

設定 Kerberos 時，一般可以採用兩種方法 — 透過 /etc/krb5.conf 檔案靜態設定，或透過 DNS 動態設定。使用 DNS 組態時，Kerberos 應用程式會嘗試透過 DNS 記錄找出 KDC 服務。使用靜態組態時，請將 KDC 伺服器的主機名稱新增到 krb5.conf（並在移動KDC時更新檔案，或以其他方式重新設定領域）。

以 DNS 為基礎的組態通常較有彈性，每台機器的組態工作量比較少。但是，它要求您的領域名稱必須和 DNS 網域或它的子網域一樣。透過 DNS 設定 Kerberos 也會產生些微的安全性問題 — 攻擊者可透過 DNS 嚴重地干擾您的基礎結構 (藉由癱瘓名稱伺服器、欺騙 DNS 記錄等)。但是，最多僅能達成拒絕服務而已。除非您在 krb5.conf 中輸入 IP 位址取代主機名稱，否則在靜態組態案例中也會發生相似狀況。

46.5.1 靜態組態

設定 Kerberos 的其中一個方式是編輯 /etc/krb5.conf 組態檔。預設安裝的檔案包含各種範例項目。在啟動之前，請刪除所有這些項目。krb5.conf 由數個小節所組成，每一個由包括在括號中的小節名稱做為前導，如 [this]。

若要設定您的 Kerberos 用戶端，請將下列節新增到 krb5.conf (其中 kdc.example.com 是 KDC 的主機名稱)：

```
[libdefaults]
        default_realm = EXAMPLE.COM

[realms]
        EXAMPLE.COM = {
                kdc = kdc.example.com
                admin_server = kdc.example.com
        }
```

default_realm 行設定 Kerberos 應用程式的預設領域。如果您有數個領域，僅需將其他敘述句新增到 [realms] 小節。

同時新增敘述句到此檔案，告知應用程式如何將主機名稱對應到領域。例如，在連線到遠端主機時，Kerberos 程式庫必須知道此主機所處的領域。這必須在 [domain_realms] 小節中設定：

```
[domain_realm]
```

```
.example.com = EXAMPLE.COM
www.foobar.com = EXAMPLE.COM
```

這告訴程式庫 example.com DNS 網域中的所有主機均位於 EXAMPLE.COM Kerberos 領域。此外，名為 www.foobar.com 的外部主機應該被視為 EXAMPLE.COM 領域的成員。

46.5.2 以 DNS 為基礎的組態

以 DNS 為基礎的 Kerberos 組態大量使用 SRV 記錄。請參閱 *(RFC2052) A DNS RR 以指定服務的位置*，網址為 http://www.ietf.org。在 BIND 名稱伺服器的較早執行方式中不支援這些記錄。至少必須使用 BIND 版本 8 才可執行。

SRV 的名稱 (就 Kerberos 相關的部分) 格式永遠為 _service._proto.realm，此處領域為 Kerberos 領域。DNS 中的網域名稱沒有大小寫之分，因此有大小寫之分的 Kerberos 領域在使用此組態方式時會發出問題。_service 是服務名稱 (例如，在嘗試聯絡 KDC 或密碼服務時使用不同名稱)。_proto 可以是 _udp 或 _tcp，但是並非所有服務均支援兩種通訊協定。

SRV 來源記錄的資料部分由優先程度值、權重、連接埠號碼和主機名稱所組成。優先程度定義嘗試主機的順序 (較低的值表示較高的優先程度)。權重在此支援相同優先程度伺服器之間負載平衡的排序。您可能從不需要它，因此也可以設為零。

MIT Kerberos 目前在查詢服務時會查閱下列名稱：

_kerberos
: 定義 KDC 精靈 (驗證和票證授予伺服器) 的位置。典型記錄是像這樣：

```
_kerberos._udp.EXAMPLE.COM.   IN  SRV    0 0 88 kdc.example.com.
_kerberos._tcp.EXAMPLE.COM.   IN  SRV    0 0 88 kdc.example.com.
```

_kerberos-adm
: 描述遠端管理服務的位置。典型記錄是像這樣：

```
_kerberos-adm._tcp.EXAMPLE.COM. IN  SRV    0 0 749 kdc.example.com.
```

因為 kadmind 不支援 UDP，應該沒有 _udp 記錄。

和靜態組態檔相似，有一種機制可通知用戶端特定伺服器在 EXAMPLE.COM 領域中，即使它不是 example.com DNS 網域的一部分。可將 TXT 記錄附加到 _keberos.hostname 來完成，如此處所示：

```
_keberos.www.foobar.com.  IN TXT "EXAMPLE.COM"
```

46.5.3 調整時鐘偏移

時鐘偏移*是接受時間戳記不完全符合主機系統時鐘票證的之容錯度*。時鐘偏移通常設為 300 秒鐘 (五分鐘)。這表示票證的時間戳記可以比伺服器的時間超前或延後五分鐘。

在使用 NTP 同步化所有主機時，可將此值減少到一分鐘左右。時鐘偏移值可在 /etc/krb5.conf 中設定如下：

```
[libdefaults]
        clockskew = 120
```

46.6 以 YaST 設定 Kerberos 用戶端

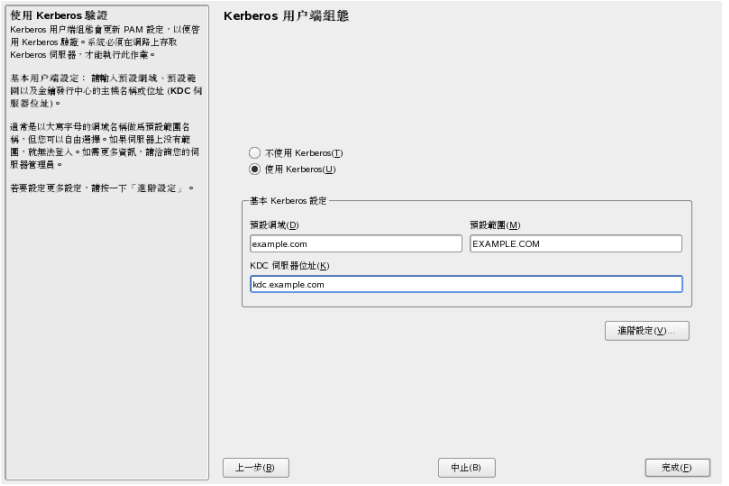
您也可以使用 YaST 來設定 Kerberos 用戶端，做為上述手動組態設定的替代方案。請執行下列步驟：

1. 以 root 使用者身份登入，並選取「*網路服務*」>「*Kerberos 用戶端*」。
2. 選擇「*使用 Kerberos*」。
3. 若要設定以 DNS 為基礎的 Kerberos 用戶端，請如下執行：
   - **3a** 確認已顯示「*基本 Kerberos 設定*」。
   - **3b** 按一下「*進階設定*」設定票證相關問題、OpenSSH 支援，和時間同步等詳細資訊。
4. 若要設定靜態 Kerberos 用戶端，請如下執行：

4a 將「*預設網域*」、「*預設領域*」和「*KDC 伺服器位址*」設定為符合您設定的數值。

4b 按一下「*進階設定*」設定票證相關問題、OpenSSH 支援，和時間同步等詳細資訊。

圖形 46.1 *YaST：Kerberos 用戶端的基本組態*



若要在「*進階設定*」中設定票證相關選項，請從下列選項中選擇：

- 以日、小時或分為單位 (使用度量單位 *d*、*h* 和 *m*，值與單位間不空格)，指定「*預設票證存在時間*」與「*預設可更新的存在時間*」。
- 若要轉寄您的完整識別以便在其他主機上使用票證，請選取「*可轉寄*」。
- 若要讓特定票證可轉送，請選取「*可代理*」。
- 以 PAM 模組可讓票證保持可使用，即使在啟用「*保留*」結束工作階段之後。
- 選取對應的核取方塊，啟用您 OpenSSH 用戶端的 Kerberos 驗證支援。然後用戶端使用 Kerberos 票證驗證 SSH 伺服器。

- 您可指定此功能的使用者必須具有的「*最小 UID*」值，以使用 Kerberos 驗證「排除」某個範圍的使用者帳戶。例如，您可能想要排除系統管理員 (`root`)。
- 使用「*時鐘偏移*」來設定時間戳記和您主機系統時間之間可容許的差異。
- 若要保持系統時間與 NTP 伺服器同步，也可以選取「*NTP 組態*」將主機設為 NTP 用戶端，以開啟第 32.1 節「使用 YaST 設定 NTP 用戶端」[553頁]中所述的 YaST NTP 用戶端。在完成組態之後，YaST 會執行所有需要的變更，而 Kerberos 用戶端便可以使用。

圖形 46.2 *YaST：Kerberos 用戶端的進階組態*

46.7 遠端 Kerberos 管理

若要新增或從 Kerberos 資料庫移除主體而不直接存取 KDC 的主控台，請告知 Kerberos 管理伺服器允許哪些主體執行何種動作。請編輯檔案 `/var/lib/kerberos/krb5kdc/kadm5.acl` 以完成此動作。ACL (存取控制檔案) 檔案讓您以良好的控制層次來指定權限。如需詳細資訊，請使用 `man 8 kadmind` 參考手冊頁面。

目前，將以下這一行放在檔案中，授與您自己對資料庫做任何想做的事情之權限：

```
newbie/admin              *
```

將 newbie 使用者名稱以您自己的名稱取代。重新啟動 kadmind 讓變更生效。

46.7.1 使用 kadmin 進行遠端管理

現在，您應該可以使用 kadmin 工具來遠端執行 Kerberos 管理任務。首先，取得您管理角色的票證，並在連線到 kadmin 伺服器時使用該票證：

```
kadmin -p newbie/admin
Authenticating as principal newbie/admin@EXAMPLE.COM with password.
Password for newbie/admin@EXAMPLE.COM:
kadmin:  getprivs
current privileges: GET ADD MODIFY DELETE
kadmin:
```

使用 getprivs 指令，確認您擁有的權限。上列清單是全部的權限。

如範例所示修改 newbie 主體：

```
kadmin -p newbie/admin
Authenticating as principal newbie/admin@EXAMPLE.COM with password.
Password for newbie/admin@EXAMPLE.COM:

kadmin:  getprinc newbie
Principal: newbie@EXAMPLE.COM
Expiration date: [never]
Last password change: Wed Jan 12 17:28:46 CET 2005
Password expiration date: [none]
Maximum ticket life: 0 days 10:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Wed Jan 12 17:47:17 CET 2005 (admin/admin@EXAMPLE.COM)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 2
Key: vno 1, Triple DES cbc mode with HMAC/sha1, no salt
Key: vno 1, DES cbc mode with CRC-32, no salt
Attributes:
Policy: [none]

kadmin:  modify_principal -maxlife "8 hours" newbie
Principal "newbie@EXAMPLE.COM" modified.
kadmin:  getprinc joe
Principal: newbie@EXAMPLE.COM
Expiration date: [never]
```

```
Last password change: Wed Jan 12 17:28:46 CET 2005
Password expiration date: [none]
Maximum ticket life: 0 days 08:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Wed Jan 12 17:59:49 CET 2005 (newbie/admin@EXAMPLE.COM)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 2
Key: vno 1, Triple DES cbc mode with HMAC/sha1, no salt
Key: vno 1, DES cbc mode with CRC-32, no salt
Attributes:
Policy: [none]
kadmin:
```

這樣會將最大票證存在時間變更為八小時。如需關於 kadmin 指令與可用選項的詳細資訊，請參閱 http://web.mit.edu/kerberos/www/krb5-1.4/krb5-1.4/doc/krb5-admin.html#Kadmin%20Options 或 man 8 kadmin。

46.8 建立 Kerberos 主機主體

除了確認您網路上的每一台機器都知道所處的 Kerberos 領域和所連絡的 KDC 之外，請為其建立*主機主體*。到目前為止，僅討論使用者證件。但是，Keberos 相容的服務通常也需要對用戶端使用者驗證自己。因此，在 Kerberos 資料庫中必須包含領域中每一台主機的主體。

主機主體的命名慣例為 host/<hostname>@<REALM>，其中 hostname 是主機的完整主機名稱。主機主體與使用者主體相似，但是有顯著的差異。使用者主體與主機主體的主要差異在於，前者的金鑰受到密碼保護—當使用者從 KDC 取得票證授予的票證時，必須輸入密碼，Kerberos 才能解密票證。如果系統管理員必須每八個小時取得 SSH 精靈的新票證，這對他而言很不方便。

替代方法是，解密主機主體的起始票證所需之金鑰由管理員從 KDC 解壓縮一次並儲存在名為 *keytab* 的本地檔案中。像 SSH 精靈之類的服務在需要時自動讀取並使用它來取得新票證。預設 keytab 檔案位於 /etc/krb5.keytab。

若要建立 test.example.com 的主機主體，請在您的 kadmin 工作階段中輸入下列指令：

```
kadmin -p newbie/admin
```

```
Authenticating as principal newbie/admin@EXAMPLE.COM with password.
Password for newbie/admin@EXAMPLE.COM:
kadmin:  addprinc -randkey host/test.example.com
WARNING: no policy specified for host/test.example.com@EXAMPLE.COM;
defaulting
to no policy
Principal "host/test.example.com@EXAMPLE.COM" created.
```

除了設定新主體的密碼之外，`-randkey` 旗標會告知 `kadmin` 產生隨機金鑰。在此處這樣用是因為此主體不需使用者互動。這是機器的伺服器帳戶。

最後，解壓縮金鑰並將它儲存在本地 keytab 檔案 `/etc/krb5.keytab` 中。此檔案是由超級使用者所擁有，因此您必須是 `root` 使用者身份，才能在 kadmin 外圍程序執行下一個指令：

```
kadmin:  ktadd host/test.example.com
Entry for principal host/test.example.com with kvno 3, encryption type Triple
DES cbc mode with HMAC/sha1 added to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/test.example.com with kvno 3, encryption type DES
cbc mode with CRC-32 added to keytab WRFILE:/etc/krb5.keytab.
kadmin:
```

在完成時，請確認您以 `kdestroy` 摧毀上面透過 kinit 所取得的管理票證。

46.9 啟用 Kerberos 的 PAM 支援

SUSE Linux Enterprise® 附帶名為 `pam_krb5` 的 PAM 模組，可支援 Kerberos 登入與密碼更新。此模組可由應用程式使用，如主控台登入、su 和 KDM 之類的圖形登入應用程式，其中使用者提供密碼，希望驗證中的應用程式為他取得起始 Kerberos 票證。

pam_unix模組也支援 Kerberos 驗證與密碼更新。若要啟用 `pam_unix2` 中的 Kerberos 支援，請編輯 `/etc/security/pam_unix2.conf` 檔案，讓它包含下列這幾行：

```
auth:       use_krb5 nullok
account:    use_krb5
password:   use_krb5 nullok
session:    none
```

接下來，所有評估此檔案中項目的程式均使用 Kerberos 進行使用者驗證。對於沒有 Kerberos 主體的使用者，pam_unix2 會回到一般密碼驗證機制。對於有主體的使用者，現在應該可以使用 passwd 指令，直接變更其 Kerberos 密碼。

若要微調使用 pam_krb5 方式，請編輯 /etc/krb5.conf 檔案並新增預設應用程式到 pam。如需詳細資訊，請使用 man 5 pam_krb5 參考手冊頁面。

pam_krb5模組不是特別為接受 Kerberos 票證做為使用者驗證的一部分之網路服務所設計。這是完全不一樣的事情，將於以下討論。

46.10 設定 Kerberos 驗證的 SSH

OpenSSH 同時支援通訊協定版本 1 和 2 的 Kerberos 驗證。在版本 1 中，會有特殊通訊協定訊息來傳輸 Kerberos 票證。版本 2 不再直接使用 Kerberos，而是依靠 GSSAPI (一般安全性 API，General Security Services API)。這是一種非 Kerberos 特定的程式設計介面 — 它設計用來隱藏基礎驗證系統 (如 Kerberos)、公開金鑰驗證系統 (如 SPKM) 或其他系統的特質。但是，包括在其中的 GSSAPI 程式庫僅支援 Kerberos。

若要以 Kerberos 驗證使用 sshd，請編輯 /etc/ssh/sshd_config 並設定下列選項：

```
# These are for protocol version 1
#
# KerberosAuthentication yes
# KerberosTicketCleanup yes

# These are for version 2 - better to use this
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
```

然後使用 rcsshd restart 重新啟動您的 SSH 精靈。

若要以通訊協定版本 2 使用 Kerberos，請同時在用戶端啟用它。可在 /etc/ssh/ssh_config 涵蓋整個系統的組態檔中編輯，或在 ~/.ssh/config 個別使用者層級來編輯。在兩種方式，都請新增選項 GSSAPIAuthentication yes。

現在您應該可以使用 Kerberos 驗證來連線。使用 `klist` 以確認您擁有有效票證，然後連線到 SSH 伺服器。若要強制 SSH 通訊協定版本 1，請在指令行指定選項 `-1`。

提示：其他資訊

`/usr/share/doc/packages/openssh/README.kerberos` 檔案討論 OpenSSH 和 Kerberos 互動的詳細資訊。

46.11 使用 LDAP 與 Kerberos

在使用 Kerberos 時，在本地網路配送使用者資訊 (如使用者 ID、群組、主目錄等) 的方式之一為使用 LDAP。必須要有強力的驗證機制以避免封包欺騙和其他攻擊。解決方案之一是也使用 Kerberos 進行 LDAP 通訊。

OpenLDAP 透過 SASL (簡單驗證工作階段層) 執行大部分驗證類別。SASL 基本上是為了驗證所設計的網路通訊協定。在 中使用的 SASL 執行方式是 cyrus-sasl，支援許多不同的驗證類別。Kerberos 驗證透過 GSSAPI (一般安全性 API，General Security Services API) 執行。根據預設，未安裝 GSSAPI 的 SASL 外掛程式。可使用 `rpm -ivh cyrus-sasl-gssapi-*.rpm` 手動安裝。

若要啟用 Kerberos 以繫結到 OpenLDAP 伺服器，請建立 `ldap/earth.sample.com` 主體，並將它新增到 keytab。

根據預設，LDAP 伺服器 slapd 以 `ldap` 使用者和群組身份執行，而只有 `root` 使用者可讀取 keytab 檔案。因此，可變更 LDAP 組態使伺服器以 `root` 使用者執行，或讓群組 `ldap` 可讀取 keytab 檔案。若將 `/etc/sysconfig/openldap` 中的 `OPENLDAP_KRB5_KEYTAB` 變數中指定 keytab 檔案，並將 `OPENLDAP_CHOWN_DIRS` 變數設為預設值 `yes`，則會由 OpenLDAP 的啟動程序檔 (`/etc/init.d/ldap`) 完成後者。若 `OPENLDAP_KRB5_KEYTAB` 留空，則會使用 `/etc/krb5.keytab` 中的 keytab，且您必須如下調整自己的權限。

若要以 `root` 使用者身份執行 slapd，請編輯 `/etc/sysconfig/openldap`。將註解字元放在 `OPENLDAP_USER` 和 `OPENLDAP_GROUP` 變數的前面可將變數停用。

若要讓群組 LDAP 可讀取 keytab 檔案，請執行

```
chgrp ldap /etc/krb5.keytab

chmod 640 /etc/krb5.keytab
```

第三或許也是最好的解決方案，就是讓 OpenLDAP 使用特殊的 keytab 檔案。若要這麼做，請啟動 kadmin，新增主體後輸入下列指令 ldap/earth.example.com:

```
ktadd -k /etc/openldap/ldap.keytab ldap/earth.example.com@EXAMPLE.COM
```

接著在外圍程序中執行：

```
chown ldap.ldap /etc/openldap/ldap.keytab
chmod 600 /etc/openldap/ldap.keytab
```

若要告知 OpenLDAP 使用不同的 keytab 檔案，要在 `/etc/sysconfig/openldap` 中變更下列變數：

```
OPENLDAP_KRB5_KEYTAB="/etc/openldap/ldap.keytab"
```

最後，使用 `rcldap restart` 重新啟動 LDAP 伺服器。

46.11.1 以 LDAP 使用 Kerberos 驗證

現在您應該可使用具 Kerberos 自動驗證的工具，如 ldapsearch。

```
ldapsearch -b ou=people,dc=example,dc=com '(uid=newbie)'

SASL/GSSAPI authentication started
SASL SSF: 56
SASL installing layers
[...]

# newbie, people, example.com
dn: uid=newbie,ou=people,dc=example,dc=com
uid: newbie
cn: Olaf Kirch
[...]
```

如您所見，ldapsearch 會列印訊息表示它啟動了 GSSAPI 驗證。下一個訊息相當難以理解，但是它顯示了*安全性強度因數* (簡稱 SSF) 是 56。(56 是個任意值。選擇它最可能的原因是它是 DES 加密金鑰的位元數。)它告訴您的是 GSSAPI 驗證成功，已使用加密來提供 LDAP 連線的完整性保護和機密保護。

在 Kerberos 中，驗證永遠為互相的。這表示您不僅要對 LDAP 伺服器驗證您自己，LDAP 伺服器也需要對您驗證。特別地，這代表與想要的 LDAP 伺服器通訊，而不是攻擊者所設定的假服務。

46.11.2 Kerberos 驗證與 LDAP 存取控制

現在，可允許每一個使用者修改其 LDAP 使用者記錄的登入外圍程式屬性。假設您有一個配置，在其中使用者 `joe` 的 LDAP 項目位於 `uid=joe,ou=people,dc=example,dc=com`，請在 `/etc/openldap/slapd.conf` 中設定下列存取控制：

```
# This is required for things to work _at all_
access to dn.base="" by * read
# Let each user change their login shell
access to dn="*,ou=people,dc=example,dc=com" attrs=loginShell
       by self write
# Every user can read everything
access to *
       by users read
```

第二個敘述句給授權的使用者對於所擁有的 LDAP 項目之 `loginShell` 屬性的寫入存取權。第三個敘述句給所有授權的使用者對整個 LDAP 目錄的讀取存取權。

整個機制還少一個步驟 — LDAP 伺服器如何確定 Kerberos 使用者 `joe@EXAMPLE.COM` 對應到 LDAP 可辨識名稱 `uid=joe,ou=people,dc=example,dc=com`。這種對應必須使用 `saslExpr` 指示詞手動設定。在此範例中，新增下列到 `slapd.conf`：

```
authz-regexp
   uid=(.*),cn=GSSAPI,cn=auth
   uid=$1,ou=people,dc=example,dc=com
```

若要了解其運作方式，您必須知道當 SASL 驗證使用者時，OpenLDAP 從 SASL 給它的名稱 (如 `joe`) 和 SASL 類別的名稱 (`GSSAPI`) 形成可辨識名稱。結果會是 `uid=joe,cn=GSSAPI,cn=auth`。

如果已經設定 `authz-regexp`，它會使用第一個引數做為一般表示式來檢查從 SASL 資訊形成的 DN。如果符合此一般表示式，名稱會以 `authz-regexp` 敘述句的第二個引數取代。`$1` 保留字元以符合 `(.*)` 表示式的子字串取代。

可能有更複雜的符合表示式。如果您的目錄結構更複雜，或者配置中的使用者名稱不是 DN 的一部分，甚至可以使用搜尋表示式來將 SASL DN 對應到使用者 DN。

47 加密分割區和檔案

每一位使用者都有一些第三方無法存取的機密資料。越依賴於行動計算，以及越依賴於在不同環境和網路中工作，處理資料時就越要多加小心。如果其他人對您的系統擁有網路或實體存取權，則建議對文件或整個分割區進行加密。筆記型電腦或抽取式媒體 (例如外接式硬碟或 USB 晶片組) 很容易遺失或被盜。因此，建議對存放機密資料的檔案部份進行加密。

有幾種方法可以藉由加密的方式來保護您的資料：

加密硬碟分割區
: 您可以在安裝期間或在已安裝的系統上，使用 YaST 建立加密的分割區。請參閱第 47.1.1 節「在安裝時建立加密分割區」 [786頁]與第 47.1.2 節「在執行系統上建立加密分割區」 [787頁]以取得詳細資料。這個選項也可用於抽取式媒體，如外接式硬碟 (如第 47.1.4 節「加密抽取式媒體內容」 [788頁]所述)。

建立加密檔案做為容器
: 您隨時都可使用 YaST 在硬碟或抽取式媒體上建立加密的檔案。然後使用這個加密檔案來*存放*其他檔案或資料夾。若需更多資訊，請參閱第 47.1.3 節「建立加密檔案做為容器」 [788頁]。

加密主目錄
: 您亦可透過 SUSE Linux Enterprise，為使用者建立加密的主目錄。使用者登入系統時，就會掛載加密的主目錄，且使用者可使用其中的內容。如需相關資訊，請參閱第 47.2 節「使用加密主目錄」 [788頁]。

加密單一 ASCII 文字檔案

如果只有少數ASCII文字檔案包含敏感或機密資料，您可以分別對其加密，並使用vi編輯器以密碼保護這些檔案。如需相關資訊，請參閱第 47.3 節「使用 vi 加密單一 ASCII 文字檔案」 [789頁]。

警告：加密媒體提供有限的保護

本章說明的方法只能提供有限的保護。您無法保護執行中的系統，使其避免資料洩露。成功裝載加密的媒體後，每個使用者都必須要有適當的權限才可存取該媒體。不過，加密的媒體對於電腦遺失或失竊等狀況很有用，也可以防止未經授權的個人讀取您的機密資料。

47.1 以 YaST 設定加密檔案系統

使用YaST可在安裝期間或在已安裝的系統上加密分割區或檔案系統的一部分。但是，在已安裝的系統上加密分割區會比較困難，因為您必須調整現有分割區的大小並對其進行變更。在這種情況下，建立指定大小的加密檔案來*存放*其他檔案或檔案系統的一部分或許會比較容易。若要加密整個分割區，請固定分割區以便在分割區配置進行加密。依預設，YaST提供的標準磁碟分割建議不包含加密分割區。在磁碟分割對話方塊中手動新增。

47.1.1 在安裝時建立加密分割區

警告：密碼輸入

請務必牢記加密分割區的密碼。沒有該密碼，您就無法存取或還原加密的資料。

YaST分割區進階對話方塊會提供建立加密分割區所需的選項。若要建立新的加密分割區，請依以下的說明繼續：

1. 透過「*系統*」>「*磁碟分割程式*」從 YaST 控制中心執行 YaST 磁碟分割程式。
2. 按一下「*建立*」並選取主要分割區或邏輯分割區。
3. 選取此分割區所需的檔案系統、大小及裝載點。

4 如果只在有需要時才裝載加密的檔案系統，請啟用「*Fstab 選項*」中的「*不要在系統啟動時裝載*」。

5 啟用「*將檔案系統加密*」核取方塊。

6 按一下「*確定*」。您會收到提示要求您輸入用於加密此分割區的密碼。此密碼不會顯示出來。若要避免輸入錯誤，請輸入該密碼兩次。

7 按一下「*確定*」完成該程序。這樣便建立了新的加密分割區。

除非選取了「*不要在系統啟動時裝載*」，否則在開機期間，作業系統會在裝載分割區之前要求輸入密碼。一旦分割區裝載完成，所有使用者都可以以使用它。

若要在開機期間跳過裝載加密分割區的程序，請在要求輸入密碼時按下 Enter。接著拒絕再次輸入密碼。在此情況下，將不會裝載加密的檔案系統，接著作業系統會繼續進行開機且不會存取您的資料。

若要存取開機時未掛載的加密分割區，請輸入 `mount` *`name_of_partition mount_point`*手動掛載分割區。收到提示時輸入密碼。處理完分割區後，請使用 `umount` *`name_of_partition`* 將其解下，以避免其他使用者存取該分割區。

在已經有許多分割區存在的機器上安裝系統時，您也可以決定在安裝期間加密現有分割區。在這種情況下，請依照第 47.1.2 節「在執行系統上建立加密分割區」 [787頁]中的說明執行，而且要知道這個動作將毀掉要加密的現有分割區上的所有資料。

47.1.2 在執行系統上建立加密分割區

警告：在執行系統中啟用加密

您也可以在執行中的系統上建立加密分割區。但是，加密現有分割區會毀掉其中的所有資料，而且必須調整現有分割區的大小，並重新設定結構。

在執行中的系統上，從 YaST 控制中心選取「*系統*」>「*磁碟分割*」。按一下「*是*」繼續進行。在「*進階分割程式*」中，選擇要加密的分割區，並按一下「*編輯*」。其餘程序與第 47.1.1 節「在安裝時建立加密分割區」 [786頁]中所述的一樣。

47.1.3 建立加密檔案做為容器

您可以不使用分割區，而改為建立特定大小的加密檔案，來存放包含機密資料的其他檔案或資料夾。此類容器檔案可從「YaST 進階磁碟分割程式」對話方塊中建立。選取「*加密檔案*」並輸入檔案的完整路徑及其大小。接受或變更建議的格式化設定和檔案系統類型。指定裝載點，並決定系統開機時是否裝載加密檔案系統。

加密分割區上的加密容器檔案有一個好處，即您無需重新分割硬碟便可新增加密容器檔案。他們會透過迴路設備進行裝載，且運作方式就跟一般分割區一樣。

47.1.4 加密抽取式媒體內容

YaST 處理抽取式媒體的方式，跟處理外接式硬碟或 USB 隨身碟等任何其他硬碟一樣。這類媒體上的容器檔案或分割區可依上述方式加密。但請啟用「*Fstab 選項*」對話中的「*開機時不掛載*」，因為抽取式媒體通常只會在系統執行時連接系統。

如果您已使用 YaST 對抽取式設備進行加密，KDE 與 GNOME 桌面在偵測到該設備時會自動辨識加密的分割區，並提示輸入密碼。如果您在執行 KDE 或 GNOME 時插入 FAT 格式化的抽取式設備，輸入密碼的桌面使用者將自動成為設備的擁有者，並且可以讀寫檔案。對於檔案系統不是 FAT 的設備，請明確變更除 `root` 之外使用者的擁有權，以讓這些使用者能夠讀寫設備上的檔案。

47.2 使用加密主目錄

要針對失竊與硬碟移除的情況來保護主目錄中的資料，可使用 YaST 使用者管理模組來啟用主目錄加密。您可為新使用者或現有使用者建立加密主目錄。要加密或解密現有使用者的主目錄，您需要知道他們的登入密碼。請參閱相關指示。

加密的主分割區建立於檔案容器中，如第 47.1.3 節「建立加密檔案做為容器」[788頁]中所述。系統會在 `/home` 下為每個加密的主目錄建立兩個檔案：

`LOGIN``.img`
存放目錄的影像

LOGIN.key
　影像金鑰，受使用者登入密碼的保護。

登入時，系統會自動解密主目錄。在系統內部則是透過 PAM 模組 pam_mount 來實現。如果需要新增其他提供加密主目錄的登入方法，必須將此模組新增到 /etc/pam.d/ 下各自的組態檔案中。如需詳細資訊，另請參閱第 27 章「*使用 PAM 驗證*」[453頁]和 pam_mount 的手冊頁。

警告：安全性限制

加密使用者的主目錄並不會強化對於其他使用者的安全性。若需要高度安全性，則不應共用實體系統。

若要增強安全性，請同時加密 swap 分割區、/tmp 與 /var/tmp 目錄，因為它們可能會儲存重要資料的暫存影像。您可依第 47.1.1 節「在安裝時建立加密分割區」[786頁]或第 47.1.3 節「建立加密檔案做為容器」[788頁]中的說明，使用 YaST 磁碟分割程式加密 swap、/tmp 與 /var/tmp。

47.3 使用 vi 加密單一 ASCII 文字檔案

使用加密分割區的缺點是在裝載分割區時，您至少要使用 root 身份登入才可存取資料。若要避免這種問題，您可以在加密模式中使用 vi。

請使用 vi -x*filename* 來編輯新檔案。接著 vi 會在加密檔案內容後提示您設定密碼。之後，每當您要存取該檔案時，vi 就會要求您輸入正確的密碼。

為了確保更高的安全性，您可以在加密分割區中放置加密的文字檔。因為 vi 中使用的加密功能不是非常強大，所以我們建議您建立一個加密文字檔。

48 藉由 AppArmor 限制權限

許多的安全性弱點通常源自於*可信賴*的程式。由於可信賴程式所具備的權限通常會成為一些攻擊者的目標，且一旦程式中出現錯誤而讓攻擊者趁機取得權限時，該程式將無法再獲得信任。

Novell® AppArmor 是一種應用程式安全性解決方案，特別針對不受信任的程式提供最低的權限限制。AppArmor 可讓管理員針對該應用程式研發一個安全性*設定檔*，指定程式可執行的作業領域，該設定檔會列出程式可存取的檔案以及可執行的作業。

若要有效提升電腦的安全性，則必須減少需調解權限的程式數，然後設定程式的最高安全性。若使用 Novell AppArmor，您只需要設定環境中可能會遭受攻擊的程式，以大幅減低安全電腦所需的工作量。Novell AppArmor 設定檔可強化原則以確保程式只會執行必要的作業。

管理員只需要關注容易受到攻擊的應用程式，並針對這些程式建立設定檔。因此，只需要建立和維護 AppArmor 設定檔集，並監控原則違反情形或 AppArmor 報告機制所記錄的例外，就足以達到保護系統的目的。

建立 AppArmor 設定檔以限制應用程式是一種簡單且方便的方法。與 AppArmor 一起提供的數種工具可協助建立設定檔。它不會要求您做任何程式設計或程序檔處理。管理員唯一需要做的就是決定最嚴格的存取原則，以及每個需強化的應用程式執行權限。

只有當軟體組態或想要的活動範圍變更時，才需要更新或修改應用程式設定檔。AppArmor 可提供直覺式工具來處理設定檔更新或修改。

使用者完全不會注意到 AppArmor 的存在。其將以「幕後」的方式執行且不需要任何使用者互動。AppArmor 不會對效能造成顯著的影響。如果 AppArmor 並未涵蓋一些應用程式活動，或者當 AppArmor 禁止執行一些應用程式活動時，管理員只需要調整此應用程式的設定檔以涵蓋這一類的行為。

本指南將概略說明 AppArmor 所需執行的基本任務以有效強化系統。如需更深入詳盡的資訊，請參閱 *Novell AppArmor 管理指南*。

48.1 安裝 Novell AppArmor

不論是哪一種安裝型式，只要是 SUSE Linux Enterprise® 安裝，都會預設安裝並執行 Novell AppArmor。若要完全發揮 AppArmor 的功能，需要下列套件：

- `apparmor-parser`
- `libapparmor`
- `apparmor-docs`
- `yast2-apparmor`
- `apparmor-profiles`
- `apparmor-utils`
- `稽核`

48.2 啟用和停用 Novell AppArmor

任何全新安裝的 SUSE Linux Enterprise 上，都會預設執行 Novell AppArmor。有兩種方式可以切換 AppArmor 的狀態：

使用 YaST 系統服務 (Runlevel)
: 若要停用或啟用 AppArmor，可將其開機程序檔加入或移出系統開機時執行的程序檔順序之中。狀態變更會在下一次系統開機時套用。

控制 Novell AppArmor 控制台
: 若要在執行的系統中切換 Novell AppArmor 的狀態，可使用 YaST Novell AppArmor 控制台將它開啟或關閉。所做的變更會立即套用。控制台會觸發 AppArmor 的停止或啟動事件，並將它的程序檔加入或移出系統的開機順序。

若要從系統開機時執行的程序檔順序中移除 AppArmor，以永久停用，請執行下列步驟：

1. 以 `root` 身份登入並啟動 YaST。
2. 選取「*系統*」>「*系統服務 (Runlevel)*」。
3. 選取「*進階模式*」。
4. 選取 `boot.apparmor`，再按一下「*設定/重設*」>「*停用服務*」。
5. 選取「*完成*」以結束 YaST Runlevel 工具。

下次系統開機時，AppArmor 將不會啟始，而且會保持停用狀態，直到您重新明確啟用為止。使用 YaST Runlevel 工具重新啟用服務的方式，與停用時的方式類似。

使用 AppArmor 控制台在執行系統中切換 AppArmor 的狀態。套用後的變更會立即生效，而且在系統重新開機後仍然有效。若要切換 AppArmor 的狀態，請按照下列步驟進行：

1. 以 `root` 身份登入並啟動 YaST。
2. 選擇「*Novell AppArmor*」>「*AppArmor 控制台*」。
3. 選取「*啟用 AppArmor*」。若要停用 AppArmor，請取消勾選這個選項。
4. 選取「*完成*」以結束 AppArmor 控制台。

48.3 設定應用程式入門

在系統上準備一個成功的 Novell AppArmor 部署，並謹慎考慮以下項目：

1. 決定要設定的應用程式。請參閱第 48.3.1 節「選擇要建立設定檔的應用程式」 [794頁]以深入瞭解相關資訊。
2. 依照 第 48.3.2 節「建立與修改設定檔」 [795頁] 中概述的說明建立需要的設定檔。檢查結果並視需要調整設定檔。

3 執行 AppArmor 報告以追蹤系統上發生的問題並處理安全性事件。請參閱 第 48.3.3 節「設定 Novell AppArmor 事件通知報告」[797頁]。

4 當環境改變，或者當您需要回應 AppArmor 報告工具所記錄的安全性事件時，請更新設定檔。請參閱 第 48.3.4 節「更新設定檔」[799頁]。

48.3.1 選擇要建立設定檔的應用程式

您只需要保護容易在特殊設定中遭受攻擊的程式，因此只需使用實際執行的應用程式設定檔。請使用以下的清單來決定最有可能的選項：

網路代辦程式
: 程式 (伺服器和用戶端) 具有開啟的網路埠。使用者用戶端 (如郵件用戶端和網頁瀏覽器) 擁有調解權限。這些程式執行時具有寫入使用者主目錄的權限，而且它們所處理的輸入來自有潛在風險的遠端來源，例如具潛在風險的網站和以電子郵件傳遞的惡意程式碼。

Web 應用程式
: 您可以透過網頁瀏覽器來呼叫 CGI Perl 程序檔、PHP 網頁和其他更複雜的網頁應用程式。

Cron 工作
: cron 精靈定期執行的程式會從各種來源讀取輸入。

若想知道目前有哪些程序透過開啟的網路連接埠執行，而且需要設定檔來加以限制時，請以 `root` 身份登入，執行 `aa-unconfined` 指令。

範例 48.1 *aa-unconfined 的輸出*

```
19848 /usr/sbin/cupsd not confined
19887 /usr/sbin/sshd not confined
19947 /usr/lib/postfix/master not confined
29205 /usr/sbin/sshd confined by '/usr/sbin/sshd (enforce)'
```

以上範例中每個標示 `not confined` 的程序可能都需要自定設定檔來加以限制。標示 `confined by` 的程序則已受到 AppArmor 的保護。

提示：如需更多資訊

如需更多關於選擇正確應用程式以建立設定檔的資訊，請參閱第 1.2 節「Determining Programs to Immunize」(第 1 章「*Immunizing Programs*」，↑*Novell AppArmor Administration Guide*)。

48.3.2 建立與修改設定檔

SUSE Linux Enterprise 平台的 Novell AppArmor 中也提供了預先設定的設定檔集，適用於大多數的重要應用程式。此外，您也可以使用 AppArmor 來為任何應用程式組建立您自己的設定檔。

共有兩種管理設定檔的方法。一種是使用 YaST Novell AppArmor 模組所提供的圖形前端，另一種是使用 Novell AppArmor 套件本身所提供的指令行工具。基本上兩種方法的運作方式都相同。

依照第 48.3.1 節「選擇要建立設定檔的應用程式」[794頁]中的說明執行 aa-unconfined，以辨識需要設定檔使其在安全模式中執行的應用程式清單。

請為每個應用程式執行以下步驟來建立設定檔：

1. 以 `root` 身份執行 `aa-genprof` *`programname`* 來讓 AppArmor 建立應用程式的設定檔大綱。

   或

   執行「*YaST*」>「*Novell AppArmor*」>「*新增設定檔精靈*」，並指定要建立設定檔的應用程式完整路徑，進而建立基本設定檔的大綱。

   系統會列出基本設定檔，並將 AppArmor 置於學習模式，這表示它會記錄您執行的任何程式活動，但還不會進行任何限制。

2. 執行完整的應用程式動作讓 AppArmor 能夠充分瞭解這些活動的運作。

3. 在 aa-genprof 中輸入 s，讓 AppArmor 分析步驟 2 [795頁]中產生的記錄檔案。

   或

在「*新增設定檔精靈*」中，按一下「*掃描 AppArmor 事件的系統記錄*」，並依照精靈提供的指示完成設定檔，來分析記錄。

AppArmor 會掃描在應用程式執行期間所進行的記錄，並要求您為每個記錄的事件設定存取權限。為每個檔案進行設定或使用 globbing。

4 視您應用程式的複雜度而定，可能有必要重複執行步驟 2 [795頁]和步驟 3 [795頁]。對應用程式進行限制，讓它在限制的條件下執行，並處理所有記錄事件。若要適當地限制應用程式功能的整體範圍，您可能需要時常重複執行這項程序。

5 在設定所有的存取權限之後，您的設定檔會被設為 enforce 模式。系統將套用此設定檔，且 AppArmor 會根據剛才建立的設定檔來限制應用程式。

如果您啟動 aa-genprof 的應用程式中，有個現有的設定檔正處於 complain 模式，則此設定檔在結束此學習循環之後將保持在學習模式。如需更多關於變更設定檔模式的資訊，請參閱章節「aa-complain—Entering Complain or Learning Mode」(第 4 章「*Building Profiles from the Command Line*」, ↑*Novell AppArmor Administration Guide*)和章節「aa-enforce—Entering Enforce Mode」(第 4 章「*Building Profiles from the Command Line*」, ↑*Novell AppArmor Administration Guide*)。

針對受限制之應用程式，執行每項所需任務來測試設定檔設定。正常情況下，封閉程式的執行會相當流暢，而您完全感覺不到 AppArmor 的活動。不過，當您發現應用程式出現一些不正常的行為時，請檢查系統記錄並查看 AppArmor 對於應用程式的限制是否過當。依系統使用的記錄機制而定，您可以在幾個位置尋找 AppArmor 記錄項目：

`/var/log/audit/audit.log`

如果已安裝 `audit` 套件，而且 auditd 正在執行中，則 AppArmor 事件會記錄在下列位置：

```
type=APPARMOR msg=audit(1140325305.502:1407): REJECTING w access to
/usr/lib/firefox/update.test (firefox-bin(9469) profile
/usr/lib/firefox/firefox-bin active /usr/lib/firefox/firefox-bin)
```

`/var/log/messages`

如果未使用 auditd，AppArmor 事件會記錄在 `/var/log/messages` 底下的標準系統記錄中。項目範例應該看起來如下：

```
Feb 22 18:29:14 dhcp-81 klogd: audit(1140661749.146:3): REJECTING w access
to /dev/console (mdnsd(3239) profile /usr/sbin/mdnsd active
/usr/sbin/mdnsd)
```

`dmesg`

如果 auditd 不在執行中，則也可以使用 `dmesg` 指令檢查 AppArmor 事件：

```
audit(1140661749.146:3): REJECTING w access to /dev/console (mdnsd(3239)
profile /usr/sbin/mdnsd active /usr/sbin/mdnsd)
```

若要調整設定檔，請依步驟 3 [795頁]所述，重新分析有關此應用程式的記錄訊息。在出現提示時，決定存取權限或限制。

提示：如需更多資訊

如需建立和修改設定檔的詳細資訊，請參閱第 2 章「*Profile Components and Syntax*」(↑*Novell AppArmor Administration Guide*)、第 3 章「*Building and Managing Profiles with YaST*」(↑*Novell AppArmor Administration Guide*)和第 4 章「*Building Profiles from the Command Line*」(↑*Novell AppArmor Administration Guide*)。

48.3.3 設定 Novell AppArmor 事件通知報告

您可以在 Novell AppArmor 中設定事件通知以檢閱安全性事件。事件通知是一種 Novell AppArmor 功能，可通知指定的電子郵件收件者何時會發生系統化 Novell AppArmor 活動 (針對選擇的嚴重程度)。您目前可在 YaST 介面中使用此功能。

若要在 YaST 中設定事件通知，請執行下列步驟：

1. 確認您可在系統中執行郵件伺服器以傳送事件通知。
2. 以 `root` 身份登入並啟動 YaST。然後選擇「*Novell AppArmor*」>「*AppArmor 控制台*」)。
3. 在「*啟用安全性事件通知*」中，選取「*設定*」。
4. 為每種記錄類型(「*精簡*」、「*摘要*」和「*詳細*」)設定報告頻率，然後輸入應接收報告的電子郵件位址並決定要記錄的事件嚴重程度。若要在事件報告中包含未知的事件，請勾選「*包含未知嚴重程度事件*」。

注：選取要記錄的事件

除非您對於 AppArmor 的事件分類非常熟悉，否則請選擇所有安全性層級的事件通知。

5 選擇「*確定*」>「*完成*」離開此對話方塊來套用設定。

透過 Novell AppArmor 報告，您可以閱讀記錄中所報告的重要 Novell AppArmor 安全性事件，而不需要手動切換只有 aa-logprof 工具才需要的繁瑣訊息。您可以依照日期範圍或程式名稱進行過濾，以縮減報告的大小。

若要設定 AppArmor 報告，請執行下列步驟：

1 以 `root` 身份登入並啟動 YaST。選取「*Novell AppArmor*」>「*AppArmor 報告*」。

2 在「*執行安全性摘要*」、「*應用程式稽核*」和「*安全性事件報告*」中選取要檢查或設定的報告類型。

3 選取「*編輯*」並提供要求的資料，來編輯報告產生頻率、電子郵件位址、輸出格式和報告的位置。

4 若要執行已選類型的報告，請按一下「*立即執行*」。

5 選取「*檢視歸檔*」並指定報告類型來瀏覽該類型的歸檔報告。

或

刪除不需要的報告或新增報告。

提示：如需更多資訊

如需在 Novell AppArmor 中設定事件通知的詳細資訊，請參閱第 6.2 節「Configuring Security Event Notification」(第 6 章「*Managing Profiled Applications*」, ↑*Novell AppArmor Administration Guide*)。如需更多有關報告組態的詳細資訊，請參閱第 6.3 節「Configuring Reports」(第 6 章「*Managing Profiled Applications*」, ↑*Novell AppArmor Administration Guide*)。

48.3.4 更新設定檔

軟體和系統組態會經常變更。因此，AppArmor的設定檔設定可能得不時進行微調。AppArmor 會檢查系統日誌以查看是否有違反原則或其他的 AppArmor 事件，並讓您隨之調整設定檔設定。您也可以使用「*更新設定檔精靈*」來處理任何設定檔定義以外的應用程式行為。

若要更新設定檔集，請執行下列步驟：

1. 以 `root` 身份登入並啟動 YaST。
2. 啟動「*Novell AppArmor*」>「*更新設定檔精靈*」。
3. 對任何資源或任何在提示時所記錄的執行檔，調整存取或執行權限。
4. 在回答所有問題後離開 YaST。您的變更會套用至對應的設定檔。

提示：如需更多資訊

如需有關從系統記錄中更新設定檔的詳細資訊，請參閱第 3.5 節「Updating Profiles from Log Entries」(第 3 章「*Building and Managing Profiles with YaST*」, ↑*Novell AppArmor Administration Guide*)。

49

安全性與機密性

Linux 或 UNIX 系統的其中一個主要特性是能夠同時 (多重使用者) 處理數個使用者，並允許這些使用者在同一部電腦上同時執行數個任務 (多任務業)。再者作業系統具有透明化的網路。使用者通常不知道他們正在使用的資料與應用程式是由其本地機器提供或是透過網路所提供。

使用多個使用者的功能，不同使用者的資料必須分開儲存。必須保證安全性與隱私。即使在電腦可以透過網路連結前，資料安全性就已經是一個重要的議題。就像今日，最重要的問題就是在遺失資料或損毀資料的媒體上(大部份是指硬碟而言)，仍然能夠提供資料。

本小節主要是將焦點放在機密議題以及保護使用者隱私的方法上，但是它並未著墨太多關於全面安全性概念應該永遠包含的一些程序，即在適當的位置具有定期更新、可運作以及已測試的備份。如果沒有這些，您可能很難將資料還原—不只是在某些硬體有缺陷的例子中，懷疑有人取得未授權的存取權並損毀檔案時也是如此。

49.1 本地安全性與網路安全性

存取資料有數種方式：

- 與具有所需資訊或具有電腦資料存取權的人員進行個人通訊
- 直接從電腦的主控台 (實體存取)
- 透過序列線
- 使用網路連結

在所有的這些例子中，在使用者存取資源或是有問題的資料前，應該先驗證其身份。網頁伺服器在這方面可能限制比較少，但是您仍然不應該對任何網友揭露所有的個人資料。

在上方的清單中，第一個例子是需要與人頻繁的互動，例如，當您正在與銀行人員聯繫時，需要證明您就是擁有該銀行帳戶的人員。接著會要求您提供簽章、PIN 或密碼以證明您就是您所聲稱的那個人。在某些狀況下，有可能從某個知道的人問出一些資訊，只要使用一些巧妙的措詞提及一些已知的片斷資訊，就有可能贏得該人員的信賴。這個受害者有可能被引導逐漸揭露更多的資訊，但卻不自知。在駭客之間，稱此為「*社會工程*」。您只能以教育人們的方式以及以謹慎的方式來處理語言與資訊才能對抗這樣的受害。在攻擊者侵入電腦系統前，他們通常會嘗試鎖定接待人員、公司的客服人員或甚至是家庭成員。在許多狀況下，通常很晚才能探查到利用社會工程的攻擊。

一個想要取得資料未授權存取權的人，也有可能使用傳統的方式，嘗試直接取得您的硬體。因此，應該善加保護機器，以防他人移動、替換或破壞其元件。這也適用於備份，甚至是任何的網路纜線或電線。另外也需要防護開機程序，因為有些眾所周知的按鍵組合，可能會引起不正常的行為。請設定 BIOS 與開機載入程式的密碼，以避免自己的電腦發生此狀況。

連接至序列埠的序列終端機仍然用於許多地方。不像網路介面，它們並不依賴網路通訊協定和主機通訊。它們使用簡單的纜線或紅外線埠，在設備之間來回傳送純文字字元。纜線本身就是此類系統的最弱點：連接較老舊的印表機時，要記錄所有透過線路傳輸的所有資料就變得相當容易。印表機可以達成的動作也可以由其他方式來完成，端視攻擊所進行的動作而定。

讀取某個主機本地上的檔案所需的存取規則，就是在不同的主機上開啟與伺服器的網路連線。本地安全性與網路安全性之間是有區別的。這條界線在於資料必須放入封包以傳送到別的地方。

49.1.1 本地安全性

本地安全性是從電腦執行位置的實體環境開始。在符合您的期望與需求的適當位置安裝機器。本地安全性的主要目標就是要將使用者彼此區隔開來，如此就不會有其他的使用者可以擅用另一個使用者的權限或身份。這是一般所遵守的規則，但是對於擁有系統最高權限的 `root` 使用者更是特別需要遵守。因為 `root` 可以使用任何本地使用者的身份，而且系統不會提示它輸入密碼，就可以讀取任何本地儲存的檔案。

49.1.2 密碼

在 Linux 系統上，密碼不會以純文字儲存，而且輸入的文字字串也不會與儲存的型式直接符合。如果是這樣，只要有人取得該對應檔案的存取權，系統上的所有帳戶都將受到侵害。實際上，會加密儲存的密碼，而且每次輸入它時，還會再加密一次，然後再比較這兩個加密的字串。只要加密的密碼無法反轉計算成原始的文字字串，就不允許登入，這提供了更多的安全性。

實際上，這是由一種特殊的演算法所達成，也稱為*「暗門演算法」*，因為它只能往一個方向運作。取得加密字串的攻擊者將無法以直接再次套用相同的演算法來取得密碼。因此,在找到一個組合與加密的密碼相似前，它將需要測試所有可能的字元組合。使用 8 個字元長的密碼，將需要計算相當多的可能組合。

在 70 年代，有人主張此方法比其他方法更為安全，因為使用了相當慢的演算法，它需要好幾秒才能加密一個密碼。然而，在另一方面，PC 已變得非常強大，可以在每秒內進行數十萬或甚至數百萬的加密。因此，加密的密碼不應該對一般的使用者顯示 (一般使用者無法讀取 `/etc/shadow`)。更重要的一點是密碼必須不易猜測，以防止密碼因為某些錯誤而變成可見。因此，將密碼如「tantalize」「翻譯」成「t@nt@1lz3」並沒有太大的用處。

以某些看起來與字母相似的數字取代一個字的某些字母並不夠安全。使用字典猜測的密碼破解程式也可以猜測到這樣的替換。最好的方法是使用一個沒有一般意義但只對您個人有意義的文字，像是某句子或某書名文字的第一個字母，例如「The Name of the Rose」by Umberto Eco。這將可以給予下列安全的密碼：

「TNotRbUE9」。相反的，像是「beerbuddy」或「jasmine76」，將很容易被對您只有一些瞭解的人士猜到。

49.1.3 開機程序

設定系統使其無法從軟碟或從 CD 開機，可以將設備整個移除或是設定 BIOS 的密碼，並將 BIOS 設成只允許從硬碟開機。一般而言，Linux 系統是由開機載入程式所啟動，允許您將額外的選項傳遞至開機核心。在 `/boot/grub/menu.lst` 中設定額外的密碼以防止其他人在開機時使用像是這樣的參數 (請參閱 第 21 章「*開機載入程式*」[371 頁])。這對於系統的安全性非常重要。不只是核心本身是以 `root` 權限執行，它也是第一個授權者可以在系統啟動時授予 `root` 權限。

49.1.4 檔案許可權

一般而言，儘可能以限制最多的權限來進行指定的任務。例如，閱讀或撰寫電子郵件絕對不需要使用 `root` 的身份。如果郵件程式有問題，而這個問題有可能造成在程式啟動時，利用它以相同的程式權限來進行攻擊。遵照上方原則，可以減少可能的損毀。

SUSE Linux Enterprise 套裝作業系統中包含之所有檔案的權限是經過精心選擇的。安裝其他軟體或其他檔案的系統管理員應該極為注意何時會這麼做，特別是在設定權限時。資深以及對安全性敏感的系統管理員，永遠會使用 `-l` 選項加上 `ls` 指令以取得龐大的檔案清單，這可允許它們立即偵測是否有任何不正確的檔案權限。不正確的檔案屬性並不只是代表檔案已經過變更或遭到刪除。這些修改過的檔案有可能是 `root` 曾經執行過它們，或是在組態檔中，程式可能以 `root` 的權限使用過這些檔案。這大幅地增加了攻擊者的機會。像這一類的攻擊稱為布穀鳥式借巢孵蛋，因為不同的使用者 (鳥) 執行 (孵化) 程式 (蛋)，就像布穀鳥誘使其他的鳥孵化牠自己的蛋。

SUSE Linux Enterprise 系統包含 `permissions`、`permissions.easy`、`permissions.secure` 以及 `permissions.paranoid` 檔案，全部都在 `/etc` 目錄中。這些檔案的目的就是用以定義特殊的權限，像是全球都可以寫入的目錄或是供檔案使用的 Setuser ID 位元 (具有 Setuser ID 位元集的程式並不是以啟動它的使用者權限執行，而是以檔案擁有者的權限執行，大部份是指 `root` 而言)。管理員可以使用 `/etc/permissions.local` 檔案以新增他自己的設定值。

若要定義上面哪些檔案要讓 SUSE Linux Enterprise 的組態程式用來做為設定許可權的根據，請選取 YaST「*安全性與使用者*」區段中的「*本地安全性*」。若要學習更多關於此主題的詳細資訊，請參閱 `/etc/permissions` 中的備註或是 `chmod` 的手冊頁 (`man chmod`)。

49.1.5 緩衝區溢位與格式字串問題

每當程式應該處理使用者可以或可能變更的資料時，就應該特別謹慎處理，不過這個問題大部份是針對應用程式設計人員而言，而非一般使用者。程式設計人員務必確保其應用程式以正確的方式解譯資料，而不需將它們寫入空間不足以儲存它們的記憶體區域中。另外，程式應該使用為該目的所定義的介面，以一致性的方式傳遞資料。

如果在寫入緩衝區時，未考量記憶體緩衝區的實際大小，就有可能發生*「緩衝區溢位」*。在某些例子中，此資料 (由使用者所產生) 使用了比緩衝區可用空間更多的空間。結果，所寫入的資料超過了緩衝區資料的結尾，因此在某些狀況下，儘可能讓程式執行使用者 (而非程式設計人員) 所設定的程式順序，而不只是處理使用者資料而已。這一類的問題有可能造成嚴重的後果，特別是如果使用特別的權限執行程式的話 (請參閱第 49.1.4 節「檔案許可權」 [804頁])。

*格式字串問題*的運作方式有些不同，但是它又是有可能導致程式出狀況的使用者輸入。大部份而言，是利用這些程式的錯誤以及特殊的權限 (Setuid 以及 Setgid 程式)，這也表示您可以移除對應的執行權限，以保護資料與系統免於這類的問題。此外，最好的方法是使用最低可能的權限來套用原則 (請參閱 第 49.1.4 節「檔案許可權」 [804頁])。

假使緩衝區溢出與格式字串問題是與使用者資料處理相關的問題，如果已指定存取權給本地帳戶，就不只可以利用它們。許多已報告的問題都有可能透過網路連結被利用。因此，應該將緩衝區溢位與格式字串問題，分類成與本地安全性以及與網路安全性相關。

49.1.6 病毒

與某些人的說法相反，有些病毒是在 Linux 上執行。然而，那些作者所發行的已知病毒是一種*「概念證明」*，以證明該技術如預期運作。目前為止沒有病毒已經*蔓延*。

如果沒有寄主得以寄生，病毒將無法生存和蔓延。在此例子中，宿主將會是一個程式或是系統的重要儲存區域，例如主要開機記錄，也就是病毒的程式碼可以寫入的。由於多個使用者的功能，Linux 可以將存取權限制為某些檔案，特別是某些重要的系統檔案。因此，如果您以 `root` 權限進行一般工作，將會增加系統受到病毒感染的機會。相反的，如果您遵照上方所提及的原則，儘可能使用最低權限，則感染病毒的機率便微乎其微。

除此之外，您絕對不要從網際網路上某些您不瞭解的網站執行程式。SUSE Linux Enterprise 的 RPM 套件帶有加密的簽章以做為數位標籤，必須小心謹慎才能建立它們。病毒通常代表管理員或使用者缺乏必要的安全性認知，造成原先應該非常安全的系統有感染的風險。

病毒不應該與蠕蟲混淆，它完全屬於網路世界。蠕蟲並不需要宿主就可以散播。

49.1.7 網路安全性

網路安全性對於防護從外面開始的攻擊非常重要。需要使用者名稱與密碼以驗證使用者的一般登入程序，仍然是本地的安全性問題。在透過網路登入的特定例子中，區別兩個安全性領域。一直到實際驗證所發生的事，都是網路安全性，而之後所發生的任何事則為本地安全性。

49.1.8 X 視窗系統以及 X 驗證

如一開始所提及，網路通透性是 UNIX 系統的其中一個主要特性。X 是 UNIX 作業系統的視窗系統，可以印象深刻的方式來利用此功能。使用 X 基本上就可以輕易地在遠端主機中登入並啟動圖形程式，接著就可以透過網路在您的電腦上顯示該程式。

應該使用 X 伺服器以遠端顯示 X 用戶端時，X 伺服器應該從未授權的存取來保護它 (即該顯示) 所管理的資源。更具體而言，某些權限必須指定給用戶端程式。使用 X Window System，就可以使用兩種方法來進行此動作，分別為以主機為基礎的存取控制以及以 Cookie 為基礎的存取控制。第一個是依賴用戶端應該執行的主機 IP 位址。負責控制的程式為 xhost。xhost 會將合法用戶端的 IP 位址輸入屬於 X 伺服器的小資料庫。然而，依賴 IP 位址的驗證並不安全。例如，如果有第二個使用者在傳送用戶端程式的主機上工作，則該使用者也可以存取 x 伺服器—就像有人在偷竊 IP 位址一樣。由於這些缺點，在此將不對此驗證方法做詳細的說明，但是您可以使用 `man xhost` 來學習它。

在以 Cookie 為基礎的存取控制例子中，所產生的字元字串只能讓 X 伺服器以及合法的使用者知道，就像某種 ID 卡。此 Cookie (這個字並不是指一般的餅乾，而是指包含警語的中國幸運籤餅) 是儲存在使用者主目錄的 `.Xauthority` 檔案中，並且提供給任何需要使用 X 伺服器的 X 用戶端來顯示視窗。使用者可以使用 xauth 工具來檢查 `.Xauthority` 檔案。如果您要重新命名 `.Xauthority` 或不小心從主目錄刪除檔案，您將無法開啟任何新的視窗或 X 用戶端。在 Xsecurity 的線上文件中，可以參閱更多關於 X Window System 安全性機制的詳細資訊 (`man Xsecurity`)。

SSH (安全外圍程序) 可用以完整地加密網路連線，並將它以背景作業方式轉遞至 X 伺服器，但是使用者不會察覺到該加密機制。這又可稱為 X 轉遞。X 轉遞是藉由模擬伺服器端的 X 伺服器並在遠端主機上設定外圍程序的 DISPLAY 變數來達成。您可以在 第 44 章「*SSH：安全性網路作業*」[755頁] 找到有關 SSH 的詳細資訊。

警告

如果您未考慮您所登入的主機是否為安全的主機，請勿使用 X 轉遞。啟用 X 轉遞後，攻擊者可以透過 SSH 連接進行驗證，以入侵您的 X 伺服器並查看您的鍵盤輸入。

49.1.9 緩衝區溢位與格式字串問題

如第 49.1.5 節「緩衝區溢位與格式字串問題」[805頁]中所討論，應該將緩衝區溢位與格式字串問題，分類成與本地與網路安全性相關的問題。隨著這類問題的本地變化、網路程式的緩衝溢位，當成功地利用這些瑕疵時，大部份都會使用它們來取得 `root` 權限。即使不是這樣，攻擊者也可能使用該問題來取得未授權的本地帳戶，以利用其他可能存在於系統上的弱點。

透過網路連結以利用緩衝區溢位與格式字串問題，絕對是遠端攻擊最常見的形式。對於這些程式的不當利用以開拓新發現的安全性漏洞，通常會張貼在安全性郵件清單上。使用它們就可以鎖定弱點，而不需知道程式碼的詳細資訊。經過這些年來，經驗顯示提供這些不當利用的程式碼對於更加安全的作業系統貢獻良多，很明顯地這是因為作業系統設計者被迫必須修正其軟體中的問題。有了免費軟體，每個人都可以存取原始程式碼 (SUSE Linux Enterprise 是隨附在所有可用的原始程式碼)，而且任何找到弱點及其不當利用程式碼的人，都可以提交修補程式以修正對應的問題。

49.1.10 拒絕服務

拒絕服務 (DoS) 的目的是為了封鎖某個伺服器程式或甚至整個系統，而這可以透過幾種方法達成：讓伺服器多載、藉由垃圾封包讓它保持忙碌，或利用遠端緩衝區溢位。通常 DoS 攻擊的唯一目的就是要使服務消失。然而，一旦特定的服務變成無法使用時，通訊就有可能變成「*攔截式攻擊*」(封包攔截、TCP 連接攔截、偽裝式攻擊) 以及 DNS 定址攻擊的弱點。

49.1.11 攔截式攻擊：嗅探、攔截、詐騙

一般而言，由攻擊者所執行的遠端攻擊會將自己放在通訊主機之間，可稱為*攔截式攻擊*。幾乎所有類型的攔截式攻擊共同點就是，受害者通常不會察覺正在進行的事情。這類的攻擊有許多可能的變化，例如，攻擊者有可能接收某個連接要求並將它轉遞給目標機器本身。現在受害者已無意識地建立與錯誤主機的連接，因為另一端假裝自己是合法的目標機器。

攔截式攻擊最簡單的形式稱為*嗅探器*—攻擊者「只是」監聽經過的網路流量。至於更複雜的攻擊，「攔截式攻擊」有可能嘗試接收已建立的連接 (攔截)。 如果要這麼做，攻擊者將需要一些時間分析封包，才能夠預測屬於該連接的 TCP 序號。當攻擊者最後奪取目標主機的角色時，受害者將會注意到此，因為他們會取得錯誤訊息說明連接因失敗而終止。有些通訊協定並沒有透過加密以防止攔截，只有在建立連接時執行簡單的驗證程序，使其更容易成為攻擊者攻擊的弱點。

*偽裝式攻擊*是一種將封包加以修改以包含假的來源資料的攻擊，通常是 IP 位址。大部份主動攻擊的形式是依賴傳送這樣的假封包—這在 Linux 機器上只能由進階使用者執行 (`root`)。

許多所提及的攻擊都是結合 DoS 來執行。如果攻擊者見到有機會可以將某個主機出其不意地擊倒，即使只是很短的時間，這將使其更易於做主動的攻擊，因為該主機將有一段時間無法妨礙其攻擊。

49.1.12 DNS 定址攻擊

DNS 定址攻擊是指攻擊者破壞 DNS 伺服器的快取，即使用偽裝的 DNS 回覆封包以回覆它、嘗試使伺服器傳送某種資料給正在要求伺服器資訊的受害者。許多伺服器以 IP 位址或主機名稱來維持與其他主機的信任關係。攻擊者必須非常

瞭解對主機之間信任關係的實際結構，以便將自己偽裝成其中一個信任的主機。通常，攻擊者會分析從伺服器所收到的一些封包以取得所需的資訊。攻擊者通常也需要在名稱伺服器鎖定適時的 DoS 攻擊。請使用加密連接，以便能夠驗證要連接的主機身份來保護自己。

49.1.13 蠕蟲

一般常會將蠕蟲與病毒混淆，但是在這兩者之間其實有一個明確的界限。不像病毒，蠕蟲並不需要感染某個宿主程式才能存活。它們相當擅長於在網路結構上儘可能迅速散播。過去所出現的蠕蟲 (例如 Ramen、Lion 或 Adore) 都是利用伺服器程式已知的安全性漏洞像是 bind8 或 lprNG。防止蠕蟲的入侵其實相當容易。假設在安全漏洞的探查以及蠕蟲攻擊伺服器之間有些間隔，這樣正是感染程式的更新版本準時提供的好機會。不過這只對管理員在有問題的系統上真正安裝了安全性更新才有用。

49.2 一些一般的安全性秘訣與技巧

為了能妥善處理安全性問題，則必須隨時使用新開發並隨時關心最新的安全性問題。保護系統以防發生各種問題的最好方法，就是儘快取得和安裝安全性公告所建議的更新套件。SUSE 安全性公告是以郵件清單的方式公佈，您可以至下列 http://en.opensuse.org/Communicate/Mailinglists 連結來訂閱。清單opensuse-security-announce@opensuse.org是第一手關於更新套件的資訊來源，並且包含 SUSE 主動貢獻者的安全性團隊成員。

郵件清單 opensuse-security@opensuse.org 是討論任何所關心的安全性問題的好地方。可在同一網頁訂閱。

bugtraq@securityfocus.com是全球其中一個最知名的安全性郵件清單。建議閱讀這個清單，它每天會收到 15 到 20 個張貼。在 http://www.securityfocus.com 可以找到更多的詳細資訊。

下列是關於處理基本安全性問題的有用規則清單：

- 根據每個工作儘可能都使用限制最多的權限集合原則，避免以 root 的身份執行一般工作。這將可減少得到布穀鳥式借巢孵蛋或病毒的風險，並防止自己犯錯。

- 如果有可能，請永遠嘗試使用加密的連接在遠端機器上工作。使用 `ssh` (安全外圍程式) 以取代 `telnet`、`ftp`、`rsh` 以及 `rlogin` 應該為標準的慣例。

- 避免使用僅依據 IP 位址的驗證方法。

- 請試著將最重要的網路相關套件保持在最新的狀態，並訂閱對應的郵件清單以接收這類程式新版的公告 (bind、postfix、ssh 等等)。同樣的原則也適用於與本地安全性相關的軟體。

- 變更 `/etc/permissions` 檔案以最佳化對系統安全性很重要的檔案權限。如果您從程式移除 Setuid 位元，它有可能再也無法依照所需的方式執行工作。另一方面，在大部份的情況下，請想想程式也將不再有潛在性的安全性風險。您可以採取全球可以寫入的目錄與檔案的相似方式。

- 停用所有非必要的網路服務，讓伺服器正常運作。這可讓您的系統較為安全。使用 `netstat` 程式可以找到插槽狀態為 LISTEN 的開啟連接埠。至於其選項，建議使用 `netstat -ap` 或 `netstat -anp`。`-p` 選項允許您查看哪些程序正在佔據哪個名稱下的埠。

  將 `netstat` 產生的結果與從主機外對連接埠進行全面掃描的結果相比較。進行此工作的最佳程式為 `nmap`，它不只檢查機器的連接埠，還會針對哪些服務正在這些埠後等待做出摘要。然而，埠掃描有可能被視為一種侵略行為，因此若無管理員的明確允許，請勿在主機上執行此動作。最後，請記得不只是要掃描 TCP 埠，另外也必須掃描 UDP 埠 (`-sS` 與 `-sU` 選項)。

- 如果要以可靠方式來監督系統檔案的完整性，請使用 SUSE Linux Enterprise 上提供的 `AIDE` (進階入侵偵測環境) 程式。加密 AIDE 所建立的資料庫以防有人篡改它。另外，在您的機器之外複製一份資料庫備份，將它儲存在未連接網路的外部資料媒體上。

- 在安裝協力廠商軟體時請謹慎小心。曾經有駭客將木馬程式建入安全軟體套件的 tar 歸檔中，所幸很快就探查到了。如果您要安裝二進位的套件，請確定您對所下載的網站沒有疑慮。

  SUSE 的 RPM 套件是以 GPG 簽署。SUSE 用以簽署的金鑰為：

```
ID:9C800ACA 2000-10-19 SUSE Package Signing Key <build@suse.de>
Key fingerprint = 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA
```

`rpm --checksig package.rpm` 指令顯示解除安裝的套件，其檢查總數與簽章是否正確。尋找該版本第一張CD上的金鑰以及全球大部份金鑰伺服器上的金鑰。

- 定期檢查使用者與系統檔案的備份。請考慮如果您未測試備份是否可以使用，它有可能沒有用處。
- 檢查記錄檔。請儘可能隨時撰寫小型的程序檔以搜尋可疑的項目。不可否認的，這不完全是一個鎖碎的任務。最後只有您才知道哪些項目是不尋常的，哪些才是正常的。
- 使用 `tcp_wrapper` 以限制存取在機器上所執行的個別服務，因此您對於哪個 IP 位址可以連接至服務具有明確的控制權。如需關於 `tcp_wrapper` 的進一步資訊，請參閱 tcpd 與 hosts_access (`man 8 tcpd`、`man hosts_access`) 的手冊頁面。
- 使用 SuSEfirewall 以增強 `tcpd` (`tcp_wrapper`) 所提供的安全性。
- 請盡量設計過多的安全措施：顯示兩次訊息比完全不顯示訊息來得好多了。

49.3 使用集中式安全性報告位址

如果您發現安全性相關的問題 (請先檢查可用的更新套件)，請撰寫電子郵件給 security@suse.de。請包含問題的詳細描述以及該套件的版本號碼。SUSE 將會儘快嘗試傳送回覆。您最好能以 PGP 加密電子郵件。SUSE 的 PGP 金鑰為：

```
ID:3D25D3D9 1999-03-06 SUSE Security Team <security@suse.de>
Key fingerprint = 73 5F 2E 99 DF DB 94 C4 8F 5A A3 AE AF 22 F2 D5
```

這個金鑰也可以從 http://www.novell.com/linux/security/securitysupport.html 下載。

VI. 疑難排解

50 說明和文件

SUSE Linux Enterprise® 隨附許多資訊和說明文件的來源資料。「SUSE 說明中心」提供系統最重要的文件資源，以可搜尋的形式提供。這些資源包括已安裝應用程式的線上說明、手冊頁面、info 頁面、硬體與軟體主題的資料庫，以及隨同產品一起提供的所有手冊。

50.1 使用 SUSE 說明中心

當您從主功能表 (「*SuSE 說明中心*」) 或在外圍程序使用 `susehelp` 指令第一次啟動「SUSE 說明中心」時，就會顯示如圖形 50.1「「SUSE 說明中心」的主要視窗」 [816頁]所示的視窗。對話方塊包含 3 個主要區域：

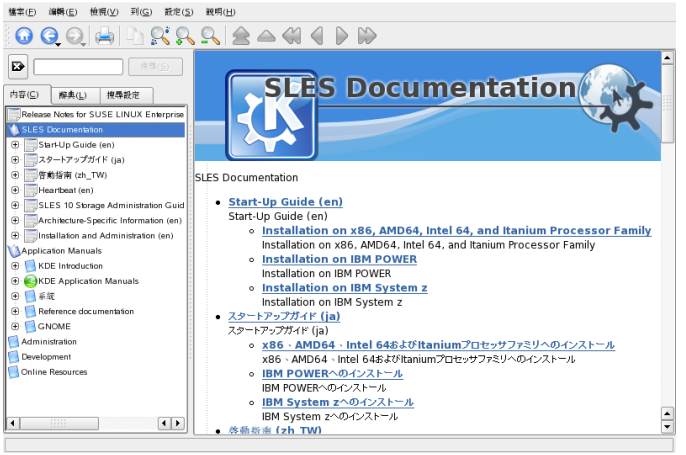
功能表列和工具列
: 功能表列提供主要編輯、瀏覽和組態選項。「*檔案*」包含用於列印目前顯示內容的指令。在「*編輯*」底下，可存取搜尋功能。「*開始*」包含了所有可能的導覽對象：「*目錄*」(說明中心的首頁)、「*上一步*」、「*下一步*」和「*上次搜尋結果*」。使用「*設定*」>「*建立搜尋索引*」，產生所有選取資訊來源的搜尋索引。工具列包含 3 個瀏覽圖示 (下一頁、上一頁和首頁) 以及一個印表機圖示，可列印目前的內容。

具有索引標籤的瀏覽區域
: 視窗左方的瀏覽區域提供輸入欄位，可快速搜尋選取的資訊來源。關於「*搜尋*」索引標籤的搜尋與搜尋功能組態的細節，請參閱 第 50.1.2 節「搜尋功能」 [817頁]。「*內容*」索引標籤顯示所有可用和目前安裝資訊來源的樹狀結構檢視。按一下書籍圖示，開啟和瀏覽個別類別。

檢視視窗

檢視視窗永遠會顯示目前選取的內容，例如線上手冊、搜尋結果或網頁。

圖形 50.1 「SUSE 說明中心」的主要視窗



注：語言選擇檢視

「SUSE 說明中心」將依目前語言提供文件。若要變更語言，請變更樹狀檢視。

50.1.1 內容

「SUSE 說明中心」提供來源的有用資訊。它包含 SUSE Linux Enterprise 的特殊文件 (*Start-Up*、*KDE User Guide*、*GNOME User Guide* 和 *Reference*)、工作站環境所有可用的資訊來源、安裝程式的線上說明，以及其他應用程式的說明內容。此外，「SUSE 說明中心」提供 SUSE 線上資料庫，包含 SUSE Linux Enterprise 相關的硬體和軟體問題。當已經產生搜尋索引後，就可以搜尋所有這些來源。

50.1.2 搜尋功能

搜尋 SUSE Linux Enterprise 所有已安裝的資訊來源、產生搜尋索引並設定一些搜尋參數。要這樣做，請使用 圖形 50.2「設定搜尋功能」[817頁] 中的「*搜尋*」索引標籤。

圖形 50.2 *設定搜尋功能*

如果尚未產生搜尋索引，當您按一下「*搜尋*」索引標籤或輸入搜尋字串後按一下「*搜尋*」時，系統會自動提示您進行。在用於產生搜尋索引的視窗 (如 圖形 50.3「產生搜尋索引」[818頁] 所示) 中，使用核取方塊來判斷要建立索引的資訊來源。當您按「*建立索引*」結束對話方塊時，便會產生索引。

圖形 50.3　產生搜尋索引

要儘可能精確限制搜尋基礎和相符清單，使用 3 個下拉功能表來決定顯示的相符數目以及要搜尋的來源選擇區域。以下選項可用來判斷選擇區域：

預設
　會搜尋預先定義的來源選擇。

全部
　會搜尋所有來源。

無
　未選取要搜尋的來源。

自定
　啟用綜覽中的相關核取方塊，決定搜尋的來源。

當您完成搜尋組態時，請按一下「*搜尋*」。相關項目會顯示在檢視視窗中，而且可以使用滑鼠，輕輕鬆鬆瀏覽。

50.2 線上文件

線上文件是 Linux 系統的重要部份。它們提供指令用法以及所有可用選項與參數的說明。如 表格 50.1「線上文件—類別和說明」[819頁] (取自 man 本身的線上文件) 所示，線上文件會依類別儲存。

表格 50.1　線上文件—類別和說明

數字	描述
1	執行程式或外圍程序指令
2	系統呼叫 (核心提供的函數)
3	程式庫呼叫 (程式庫中的函數)
4	特殊檔案 (通常位於 /dev)
5	檔案格式和慣例 (`/etc/fstab`)
6	遊戲
7	其他 (包括巨集套件與慣例)，例如，man(7)、groff(7)
8	系統管理指令 (通常僅適用於 root)
9	核心常式 (非標準)

線上文件通常和相關指令一起提供。可以從說明中心或直接在外圍程序進行瀏覽。要在外圍程序顯示線上文件，請使用 `man` 指令。例如，要顯示 `ls` 的線上文件，請輸入 `man ls`。每一個線上文件由標籤為 *NAME*、*SYNOPSIS*、*DESCRIPTION*、*SEE ALSO*、*LICENSING* 以及 *AUTHOR* 等的許多部份組成。視指令的類型，可能還包括其他可用區段。使用 Q 鍵，即可結束 man 頁面檢視器。

顯示線上文件的另一個方法是使用 Konqueror。請啟動 Konqueror 並進行輸入，例如，`man:/ls`。如果指令有多個類別，Konqueror 會顯示為連結。

50.3 Info 頁面

Info 頁面是您系統上另一個重要的資訊來源。其所含的資訊通常比 man 頁面更詳細。您可以使用 info 檢視器來瀏覽 info 頁面，以及顯示稱為「節點」的各個區段。請使用 `info` 指令來進行這個任務。例如，要檢視 `info` 本身的 Info 頁面，請在外圍程序輸入 `info info`。

如需更方便的操作方式，請使用「說明中心」或 Konqueror。請啟動 Konqueror 並輸入 `info:/` 來檢視最頂層。要顯示 `grep` 的 info 頁面，請輸入 `info:/grep`。

50.4 The Linux Documentation Project

The Linux Documentation Project (TLDP) 是由一個志願團體所營運，他們撰寫 Linux 及 Linux 相關文件 (請參閱 http://www.tldp.org)。這組文件包含入門者的教學課程，但主要是針對有經驗的使用者和專業系統管理員。TLDP 已免費公開發行 HOWTO、常見問題集以及指南 (手冊)。

50.4.1 HOWTO

HOWTO 通常是完成某個特定任務的簡短資訊性逐步指南。它是由進階使用者以程序性的方式針對非進階使用者所撰寫的。例如，如何設定 DHCP 伺服器。HOWTO 可以在套件 `howto` 中找到，並安裝在 `/usr/share/doc/howto`

50.4.2 常見問題解答

FAQ (常見問題集) 是一系列的問題與解答。源於 Usenet 新聞群組，目的是為了減少連續張貼相同的基本問題。

50.5 Wikipedia：免費線上百科全書

Wikipedia 指「設計成可供任何人閱讀與編輯的多語系百科全書」(請參閱 http://en.wikipedia.org)。Wikipedia 的內容是由它的使用者所建立並已免費公開發行 (GFDL)。任何訪客都可以編輯文章，儘管有遭到破壞的風險，但並不因此拒絕任何訪客。在超過四十萬篇的文章中，幾乎每一個主題都可以找到解答。

50.6 指南與書籍

關於 Linux 主題，有廣泛的指南和書籍可供閱讀。

50.6.1 SUSE 書籍

SUSE 提供詳細的參考書籍。我們提供不同語言的 HTML 和 PDF 版本的書籍。DVD 的 `docu` 目錄有提供 PDF 檔案。如需 HTML，請安裝 `opensuse-manual_LANG` 套件 (以您偏好的語言來取代 *LANG*)。安裝之後，您就可以在「SUSE 說明中心」找到這些資訊。

50.6.2 其他手冊

「SUSE 說明中心」針對各種主題與程式提供額外的手冊和指南。可以在 `http://www.tldp.org/guides.html` 找到更多詳細資訊。範圍從 *Bash Guide for Beginners (Bash 初學者指南)* 到 *Linux Filesystem Hierarchy (Linux 檔案系統階層)*、*Linux Administrator's Security Guide (Linux 管理員安全性指南)*。通常，指南比 HOWTO 或常見問題集擁有更詳細更豐富的資訊。它們通常是由進階使用者針對進階使用者而撰寫的。其中有些書籍雖然舊但仍有價值。請使用 YaST 來安裝書籍與指南。

50.7 套件文件

如果您在系統中安裝套件，就會建立 `/usr/share/doc/packages/`*`packagename`* 目錄。您也可以在套件維護員中，找到來自 SUSE 的檔案和其他資訊。有時也會有範例、組態檔、其他程序檔等可供使用。您通常可以找到下列檔案，但是這些檔案並不是標準檔案，有時也無法使用所有的檔案。

`AUTHORS`
: 此套件的主要開發人員清單，通常還包含他們的任務。

`BUGS`
: 此套件的已知問題或故障。通常也會包含 Bugzilla 網頁的連結，您可在此搜尋所有問題。

`CHANGES` , `ChangeLog`
: 版本之間的變更摘要。因為它非常詳盡，所以對於開發人員而言通常很有幫助。

COPYING , LICENSE
授權資訊。

FAQ
自郵寄清單或新聞群組所收集的問題與解答。

INSTALL
在系統中安裝此套件的程序。因為您已經安裝套件，所以通常不需要它。

README , README.*
使用方式、套件運用方法的一般資訊。

TODO
尚未執行但可能會在未來執行的項目。

MANIFEST
具有簡短摘要的檔案清單。

NEWS
說明此版本的新功能。

50.8 Usenet

Usenet 建立於網際網路興起之前的 1979 年，是至今仍在使用的最早電腦網路之一。Usenet 文章的格式與傳輸和電子郵件非常相似，但卻是針對多對多通訊而開發。

Usenet 分成七個主題類別：comp.* 針對電腦相關的討論、misc.* 針對各種主題、news.* 針對新聞群組的相關主題、rec.* 針對休閒娛樂、sci.* 針對科學相關的討論、soc.* 針對社會問題的討論，以及 talk.* 針對各種聊天主題。最上層分成許多子群組。例如，comp.os.linux.hardware 是 Linux 硬體相關問題的新聞群組。

您必須先將您的用戶端連接到新聞伺服器並訂閱特定新聞群組，才能夠張貼文章。新聞用戶端包含 Knode 或 Evolution。新聞伺服器彼此之間都互相通訊並交換文章。您的新聞伺服器並不一定提供所有的新聞群組。

Linux 使用者比較關心的新聞群組有 comp.os.linux.apps、comp.os.linux.questions 以及 comp.os.linux.hardware。如果您找不到某個特定的新聞群組，請前往 http://www.linux.org/docs/usenetlinux.html。請遵循 http://www.faqs.org/faqs/usenet/posting-rules/part1/ 線上提供的一般 Usenet 規則。

50.9 標準和規格

目前有各種來源，提供關於標準或規格的資訊。

http://www.linuxbase.org
: Free Standards Group 是非營利的獨立組織，促進發送免費軟體和開放原始碼軟體。該組織定義發送獨立標準，致力於達到此目標。多個標準的維護，例如重要 LSB (Linux Standard Base) 是由該組織監督的。

http://www.w3.org
: World Wide Web Consortium (W3C) 是特別知明的標準組織之一。它是在 1994 年 10 月由 Tim Berners-Lee 所創立，致力於網路技術的標準化。W3C 促進開放、免授權以及獨立於製造商規格的傳播，例如 HTML、XHTML 和 XML。這些網路標準分四階段在*工作小組*中開發，並以 *W3C 建議* (REC) 的形式，推向公眾。

http://www.oasis-open.org
: OASIS (Organization for the Advancement of Structured Information Standards) 是國際性論壇，專門開發網路安全、電子商務、商業交易、邏輯以及不同市場間的交互運作標準。

http://www.ietf.org
: (IETF) 是由研究者、網路設計人員、供應商和使用者等組成的一個活躍的國際性組織。它著重於網際網路結構的開發，以及透過通訊協定平穩地操作網際網路。

  每一個 IETF 標準會發行為 RFC (Request for Comments) 文件，而且是免費提供的。RFC 可分為六種類型：建議標準、草案標準、網際網路標準、實驗協定、資訊文件和歷史標準。只有前 3 個 (建議、草稿和完整) 較精細的 IETF 標準 (請參閱 http://www.ietf.org/rfc/rfc1796.txt)。

http://www.ieee.org

Institute of Electrical and Electronics Engineers (IEEE) 是草擬資訊技術、電訊、醫藥與保健、運輸和其他等等區域標準的組織。IEEE 標準需要付費。

http://www.iso.org

ISO Committee (International Organization for Standards) 是全球最大的標準開發商，它的網路含蓋 140 個國家的標準機構。ISO 標準需要付費。

http://www.din.de ， http://www.din.com

Deutsches Institut fur Normung (DIN) 是註冊的技術和科學協會。它成立於 1917。根據 DIN，該組織是「負責德國在全球和歐洲標準組織的利益。」

該協會讓製造商、客戶、貿易專家、服務公司、科學家和其他對標準建立感興趣的人聚集一起。該標準需要付費而且可以使用 DIN 首頁訂購。

51 一般問題和解決方案

本章提供一系列可能發生的常見問題，希望盡可能涵蓋各類的潛在問題。這樣一來，即使這裡沒有精準的列出您的狀況，也會有足夠類似的狀況，提供解決方案的提示。

51.1 尋找並收集資訊

Linux 會非常詳細的記錄事件。系統發生問題時，有幾個地方需要注意，大部分都是 Linux 系統的標準訊息，而有些則是特定 SUSE Linux Enterprise 系統的訊息。大部分記錄檔可透過 YaST (「*其他*」>「*啟動記錄*」) 檢視。

YaST 可讓您收集支援團隊所需的所有系統資訊。請使用「*其他*」>「*支援查詢*」。選取問題類別。在收集到所有資訊之後，將此份資訊連結到您的支援要求。

下面為最常被查看的記錄檔案清單，以及其一般內容。

表格 51.1 *記錄檔案*

Log File	描述
`/var/log/boot.msg`	開機程序期間，來自核心的訊息。
`/var/log/mail.*`	來自郵件系統的訊息。
`/var/log/messages`	執行時來自核心與系統記錄精靈的持續訊息。

Log File	描述
/var/log/NetworkManager	NetworkManager 中的記錄檔，用於收集網路連接性的問題。
/var/log/SaX.log	來自 SaX 顯示器與 KVM 系統的硬體訊息。
/home/*user*/.xsession-errors	來自目前執行中桌上應用程式的訊息。以實際的使用者名稱取代 *user*。
/var/log/warn	來自核心和系統記錄精靈的所有訊息，都會被指定為「警告」或更高的層級。
/var/log/wtmp	二進位檔案包含使用者對於目前機器工作階段的登入記錄。請以 last 檢視。
/var/log/Xorg.*.log	來自 X Window System 的多種啟動和執行時期記錄。對於 X 啟動失敗的除錯非常實用。
/var/log/YaST2/	目錄包含 YaST 的動作與結果。
/var/log/samba/	目錄包含 Samba 伺服器和用戶端記錄訊息。

與記錄檔不同的是，您的機器亦提供您執行中系統的資訊。請參閱表格 51.2: 系統資訊。

表格 51.2 *系統資訊*

檔案	描述
/proc/cpuinfo	這裡顯示處理器資訊，包括其類型、廠商、型號與效能。
/proc/dma	這裡顯示目前使用的 DMA 頻道。
/proc/interrupts	這裡顯示使用中的岔斷為何，以及使用中的數量。

檔案	描述
/proc/iomem	這裡顯示 I/O (輸入/輸出) 記憶體的狀態。
/proc/ioports	這裡顯示此時使用中的 I/O 埠。
/proc/meminfo	這裡顯示記憶體狀態。
/proc/modules	這裡顯示個別模組。
/proc/mounts	這裡顯示目前裝載的設備。
/proc/partitions	這裡顯示所有硬碟的分割區。
/proc/version	這裡顯示目前的 Linux 版本。

Linux 隨附多種工具可進行系統分析和監控。請參閱 第 17 章「*系統監視公用程式*」[299頁] 以取得用於系統診斷最重要的選項。

下列的各個狀況都以描述問題的標題為開頭，接著的一或兩個段落則會提供建議的解決方案、更詳細解決方案的可用參考資料，以及交叉參照到可能相關的其他狀況。

51.2 安裝問題

安裝問題是安裝機器時失敗的一個狀況。可能是徹底失敗，也可能是無法啟動圖形安裝程式。本節著重於您可能遭遇的一般問題，並對此類狀況提供可能的解決方案或處理方式。

51.2.1 檢查媒體

若您使用 SUSE Linux Enterprise 安裝媒體有任何問題，請使用「*軟體*」>「*媒體檢查*」檢查安裝媒體的完整性。媒體問題比較可能發生在您自己燒錄的媒體中。若要檢查 product; CD 或 DVD，請將媒體插入光碟機，並按一下「*開始*」，讓 YaST 檢查媒體的 MD5 檢查總數。這可能會花費幾分鐘。如果偵測到錯誤，請勿使用此媒體來進行安裝。

51.2.2 硬體資訊

使用「*硬體*」>「*硬體資訊*」顯示偵測到的硬碟和技術資料。按一下樹狀結構的任何節點，取得關於設備的更多資訊。例如當您需要有關硬體的資訊而希望提交支援要求時，這個模組就會特別實用。

按一下「*儲存至檔案*」將顯示的硬體資訊儲存至檔案。選取想要的目錄和檔案名稱，然後按一下「*儲存*」來建立檔案。

51.2.3 無可用的可開機 CD-ROM 光碟機

若您的電腦沒有可開機的 CD 或 DVD-ROM 光碟機，或若 Linux 不支援您的光碟機，有幾種選項可讓您不用內建 CD 或 DVD 光碟機即可安裝機器：

從磁片開機
: 建立開機磁片並從磁片開機，而不用 CD 或 DVD。

使用外接開機設備
: 若機器的 BIOS 和安裝核心支援的話，您可以從外接 CD 或 DVD 光碟機來開機。

透過 PXE 以網路開機
: 若機器沒有 CD 或 DVD 光碟機，但提供乙太網路連線作業，則可完全採用網路式安裝。請參閱第 4.1.3 節「透過 VNC 進行的遠端安裝—PXE 開機和網路喚醒功能」[46頁]和第 4.1.6 節「透過 SSH 進行的遠端安裝—PXE 開機和網路喚醒功能」[49頁]以取得詳細資料。

從磁片 (SYSLINUX) 開機

在某些舊型電腦上，沒有可用的可開機CD-ROM光碟機，只有軟碟機設備。若要在這類系統上安裝，請建立開機磁片並以此開機。

開機磁片包括 SYSLINUX 載入器與 linuxrc 程式。SYSLINUX 允許在啟動程序選取核心以及硬體需要的任何參數。linuxrc程式可以為您的硬體載入核心模組，然後開始安裝。

從開機磁片開機時，會由 SYSLINUX 載入器 (`syslinux` 套件) 起始開機程序。系統啟動時，SYSLINUX 會執行最小的硬體偵測，主要包括以下步驟：

1. 程式檢查 BIOS 是否提供 VESA 2.0–相容的框架緩衝區支援，然後啟動核心。

2. 讀取監視器資料 (DDC 資訊)。

3. 開機載入程式設定時，會讀取第一個硬碟的第一個區塊 (MBR)，將 BIOS ID 對應至 Linux 設備名稱。程式嘗試透過 BIOS 的 lba32 功能讀取區塊，判斷 BIOS 是否支援這些功能。

如果啟動 SYSLINUX 時，您一直按住 Shift，會略過所有這些步驟。基於疑難排解用途，請插入以下一行文字

```
verbose 1
```

於 `syslinux.cfg`，讓啟動載入器顯示目前執行的動作。

如果機器未從磁片開機，您可能需要將 BIOS 的啟動順序變更成 `A,C,CDROM`。

外接開機設備

支援大部份的 CD-ROM 光碟機。如果 CD-ROM 光碟機開機發生問題，請以 CD 組中的 CD 2 開機片來開機。

如果系統沒有 CD-ROM 光碟機也沒有軟碟機，仍然可以使用 USB、FireWire 或 SCSI 來連接外接式 CD-ROM 以啟動系統。這大部份是依賴 BIOS 及使用的硬體之間的互動。如果您遭遇到問題，有時更新 BIOS 可能會有幫助。

51.2.4 從安裝媒體開機失敗

機器無法開機進行安裝的原因可能有兩種：

CD 或 DVD-ROM 光碟機無法讀取開機影像檔
: 您的光碟機可能無法讀取 CD1 上的開機影像檔。如果是這種情況，請使用 CD2 來執行系統開機。CD2 含有傳統 2.88 MB 的開機影像檔，即使未支援的磁碟機也可以讀取，並且可以如 第 4 章「*遠端安裝*」[43頁] 所述，透過網路執行安裝。

BIOS 中的開機順序不正確
: BIOS 的開機順序中必須將 CD-ROM 設為開機的第一個項目。否則機器會嘗試從其他媒體開機，一般會從硬碟開機。您可在主機板提供的文件或下列段落中，找到變更 BIOS 開機順序的指導。

BIOS 是提供電腦最基本功能的軟體。主機板供應商會針對自己的硬體提供特製的 BIOS。通常，BIOS 設定只可在特定時間 (機器開機時) 進行存取。在這個啟始化階段，機器會執行一些硬體診斷測試。其中之一是記憶體檢查，由記憶體計數器指示。當計數器出現時，請尋找指示按下按鍵來存取 BIOS 設定的一行文字，通常在計數器下方或底端某個位置。該按鍵通常是按 Del、F1 或 Esc。請按住這個按鍵，直到 BIOS 設定畫面出現為止。

過程 51.1 *變更 BIOS 開機順序*

1. 使用開機常式所宣告的正確按鍵進入 BIOS，等待 BIOS 畫面出現。

2. 要變更 AWARD BIOS 中的開機順序，請尋找「*BIOS FEATURES SETUP*」(BIOS 功能設定) 項目。其他製造商可能使用不同的名稱，例如「*ADVANCED CMOS SETUP*」(進階 CMOS 設定)。當您找到該項目後，請選取並按 Enter 確認。

3. 在接著開啟的畫面，請尋找叫做「*BOOT SEQUENCE*」(開機順序) 的子項目。開機順序通常設成 `C,A` 或 `A,C`。在前一種情況，機器首先搜尋硬碟 (C)，然後是軟碟機 (A) 來尋找可開機媒體。請按 PgUp 或 PgDown 鍵來變更設定，直到順序改成 `A,CDROM,C` 為止。

4. 請按 Esc 來離開 BIOS 設定畫面。要儲存變更，請選取「*SAVE & EXIT SETUP*」(儲存並結束變更)，也可以按 F10。要確認儲存設定，請按 Y。

過程 51.2 *在 SCSI BIOS (Adaptec 主機介面卡) 中變更開機順序*

1. 按 Ctrl + A 開啟設定。

2. 選取「*Disk Utilities*」(磁碟公用程式)，來顯示已經連接的硬體元件。

   記下您 CD-ROM 光碟機的 SCSI ID。

3. 使用 Esc 離開功能表。

4. 開啟「*Configure Adapter Settings*」(設定介面卡設定)。在「*Additional Options*」(其他選項) 下，請選取「*Boot Device Options*」(開機設備選項)，然後按 Enter。

5. 請輸入光碟機的 ID，接著再按 Enter。

6. 按兩下 Esc 回到 SCSI BIOS 的開始畫面。

7 退出這個畫面，接著按「*Yes*」(是) 來啟動電腦。

不管您最終安裝使用的語言和鍵盤配置為何，大部分的BIOS組態均使用美國鍵盤配置，如下圖所示：

圖形 51.1 *美國鍵盤配置*

51.2.5 無法開機

有些硬體類型，多半是極舊或極新的機型，會無法安裝。在許多案例中，發生此狀況的原因是因為安裝核心不支援此類型的硬體，或由於此核心中包含的某些功能 (如 ACPI) 一直造成某些硬體發生問題。

若您的系統無法使用標準「*安裝*」模式，從第一個安裝開機畫面安裝的話，請嘗試下列方法：

1 使用仍位於 CD-ROM 光碟機中的 CD 或 DVD，以 Ctrl + Alt + Del，或硬體 reset 按鈕重新開機。

2 出現開機畫面時，使用鍵盤上的方向鍵瀏覽至「*Installation--ACPI Disabled*」，並按 Enter 啟動開機與安裝程序。此選項會停用 ACPI 電源管理技術的支援。

3 如 第 3 章「*使用 YaST安裝*」[17頁] 所述，繼續安裝。

若這樣失敗的話，請如上繼續，但改選「*Installation--Safe Settings*」。此選項會停用 ACPI 和 DMA 支援。大部分的硬體是以此選項開機。

若這些選項都失敗的話，請使用開機選項提示，將支援此類硬體所需的其他參數傳送到安裝核心。如需關於可做為開機選項之參數的詳細資訊，請參閱位於 `/usr/src/linux/Documentation/kernel-parameters.txt` 的核心文件。

提示：取得核心文件

安裝 `kernel-source` 套件以檢視核心文件。

其中有許多其他的ACPI相關參數，可讓您在開機前的開機提示中輸入，以進行安裝：

`acpi=off`
: 此參數會關閉電腦的所有 ACPI 子系統。如果您的電腦根本無法處理 ACPI 或者您認為電腦的 ACPI 造成問題，此參數會很有幫助。

`acpi=force`
: 永遠啟用 ACPI，即使電腦的 BIOS 出廠日期是在 2000 年以前。若沒有使用 `acpi=off`，設定此參數也會啟用 ACPI。

`acpi=noirq`
: 不將 ACPI 用於 IRQ 路由。

`acpi=ht`
: 只執行足夠啟用超執行緒的 ACPI。

`acpi=strict`
: 降低對不完全與 ACPI 規格相容之平台的容忍度。

`pci=noacpi`
: 停用新 ACPI 系統的 PCI IRQ 路由。

`pnpacpi=off`
: 當您的BIOS設定中包含錯誤的岔斷或連接埠時，可透過此選項檢查序列或平行問題。

`notsc`
: 停用時戳計數器。可使用此選項解決系統中的計時問題。這是一項新功能。如果您發現機器上出現效能衰退，特別是越到最後，效能越低，或者甚至完全停止，不妨嘗試此選項。

`nohz=off`
: 停用 nohz 功能。如果您的機器暫停，使用此選項可能會有所幫助。通常您不需要使用這個選項。

一旦您判斷出正確的參數組合，YaST 就會自動將其寫入開機載入程式組態，以確定系統下次可正確開機。

如果核心載入或者安裝時發生不明錯誤，選取開機功能表的「*記憶體測試*」，檢查記憶體。若「*記憶體測試*」傳回錯誤，則通常會是硬體錯誤。

51.2.6 無法啟動圖形安裝程式

將第一片 CD 或 DVD 插入光碟機並重新開機後，會出現安裝畫面，但在選取「*安裝*」之後，並未啟動圖形安裝程式。

有許多方法可以解決此狀況：

- 嘗試選取安裝對話方塊的其他螢幕解析度。
- 選取「*文字模式*」進行安裝。
- 透過 VNC，使用圖形安裝程式進行遠端安裝。

若要變更為其他的螢幕解析度，請如下執行：

1. 開機以進行安裝。
2. 按 F3 開啟功能表，從中選取較低解析度進行安裝。
3. 選取「*安裝*」並如 第 3 章「*使用 YaST安裝*」 [17頁] 所述繼續安裝。

若要以文字模式進行安裝，請如下執行：

1. 開機以進行安裝。
2. 按下 F3 並選取「*文字模式*」。
3. 選取「*安裝*」並如 第 3 章「*使用 YaST安裝*」 [17頁] 所述繼續安裝。

若要進行 VNC 安裝，請依照下列步驟執行：

1 開機以進行安裝。

2 在開機選項提示中輸入下列文字：

```
vnc=1 vncpassword=some_password
```

以安裝所用的密碼取代 *some_password*。

3 選取「*安裝*」，然後按 Enter 啟動安裝。

這樣不會直接啟動圖形安裝常式，系統反而會繼續以文字模式執行然後暫停，顯示含有 IP 位址與連接埠號碼的訊息，根據此訊息便可以透過瀏覽器介面或 VNC 檢視器應用程式來找到安裝程式。

4 如果使用瀏覽器來存取安裝程式，請啟動瀏覽器並輸入 SUSE Linux Enterprise 機器安裝程式所提供的位址資訊，並按一下 Enter：

```
http://ip_address_of_machine:5801
```

瀏覽器視窗中會開啟一個對話方塊，提示您輸入 VNC 密碼。輸入密碼，並如 第 3 章「*使用 YaST安裝*」[17頁] 所述繼續安裝。

重要

若要在任何作業系統下、使用任何瀏覽器、透過 VNC 工作，首先必須啟用 Java 支援。

若您在偏好的作業系統中使用 VNC 檢視器，請在出現提示時輸入 IP 位址和密碼。會開啟一個視窗，顯示安裝對話。請如一般方式繼續安裝。

51.2.7 只有極簡開機畫面被啟動

您將第一片 CD 或 DVD 插入光碟機後，BIOS 常式結束了，但系統未啟動圖形開機畫面，而是啟動一個非常簡化的文字介面。若機器無法提供足夠的圖形記憶體以轉譯圖形開機畫面，就可能會發生此現象。

雖然文字開機畫面看起來簡化，但其提供的功能幾乎與圖形介面一樣：

開機選項
: 與圖形介面不同的是，這裡無法以鍵盤游標選取開機選項。文字模式開機畫面的開機功能表，會在開機提示時提供一些可輸入的關鍵字。這些關鍵字對

映到圖形版本所提供的選項。輸入您的選擇並按一下 Enter 以啟動開機程序。

自定開機選項
: 選取開機選項之後，在開機提示中輸入適當的關鍵字，或如第 51.2.5 節「無法開機」[831 頁]所述輸入自定開機選項。若要啟動安裝程序，請按下 Enter。

螢幕解析度
: 使用 F 鍵決定安裝時的螢幕解析度。若您需要以文字模式開機，請選擇 F3。

51.3 開機問題

開機問題指的是您的系統無法正常開機的狀況 (無法開機到預期的 Runlevel 和登入畫面)。

51.3.1 無法載入 GRUB 開機載入程式

若硬體功能正常，則可能是開機載入程式損毀而 Linux 無法在機器上啟動。若是這樣的話，必須重新安裝開機載入程式。若要重新安裝開機載入程式，請執行下列步驟：

1. 將安裝媒體插入光碟機中。
2. 重新開機。
3. 從開機功能表選取「*安裝*」。
4. 選擇語言。
5. 接受授權合約。
6. 在「*安裝模式*」畫面中，選取「*其他*」，並將安裝模式設定為「*修復已安裝系統*」。
7. 進入 YaST 系統修復模組後，選取「*進階工具*」，然後選取「*安裝新開機載入程式*」。
8. 還原原始設定並重新安裝開機載入程式。

9 結束 YaST 系統修復並重新啟動系統。

如果由於某種原因，圖形介面未出現，或者您想要手動修復系統，請參閱章節「使用救援系統」[853頁]以取得相關指示。

另一個機器無法開機的原因可能跟 BIOS 有關：

BIOS 設定
: 檢查與您硬碟相關的 BIOS 設定。若在目前 BIOS 設定中找不到硬碟本身，則可能只是未啟動 GRUB。

BIOS 開機順序
: 檢查您系統的開機順序是否包含硬碟。若無法啟用硬碟選項的話，您的系統可能已正確安裝，但在需要存取硬碟時會無法開機。

51.3.2 沒有圖形登入

若可開機，但未開機到圖形登入管理員，可能的問題不是出在預設的 runlevel 選擇，就是在於 X Window System 組態。若要檢查 Runlevel 組態，請以根使用者的身份登入，檢查機器是否設定為開機至 Runlevel 5 (圖形桌面)。有個方法可以快速檢查此設定，就是檢驗 `/etc/inittab` 的內容，如下：

```
nld-machine:~ # grep "id:" /etc/inittab
id:5:initdefault:
nld-machine:~ #
```

傳回的行指示機器的預設 runlevel (`initdefault`) 設為 `5`，且應開機至圖形桌面。若 runlevel 設為其他數字，請使用 YaST Runlevel 編輯器模組將它設為 `5`。

重要

請勿手動編輯 runlevel 組態。否則 SUSEconfig (由 YaST 所執行) 會在下次執行時覆寫這些變更。若您需要在此進行手動變更，請在 `/etc/sysconfig/suseconfig` 中，將 *`CHECK_INITTAB`* 設定為 `no`。

若 runlevel 是設為 `5`，您的桌面或 X Windows 可能有損毀問題。請檢查 `/var/log/Xorg.*.log` 中的記錄檔，以瞭解 X 伺服器嘗試啟動時的訊息。若啟動時桌面故障，可能會將錯誤訊息記錄至 `/var/log/messages`。若這些錯誤訊息指出X伺服器中有組態問題，請嘗試修復這些問題。若仍然未出現圖形系統，請考慮重新安裝圖形桌面。

快速測試：若使用者目前已登入主控台中，則 `startx` 指令可強制 X Window System 以預設的組態啟動。若這樣沒有作用的話，會將錯誤記錄至主控台。如需 X Window System 組態的詳細資訊，請參閱 第 26 章「*X Window System*」[441頁]。

51.4 登入問題

登入問題是指您的機器已確實登入到歡迎畫面，或收到登入提示，但使用者名稱或密碼不被接受，或是接受後運作不正常 (無法啟動圖形桌面、產品錯誤、出現指令行等等)。

51.4.1 有效的使用者名稱和密碼組合失敗

這種情形常發生於系統設定為使用網路驗證或目錄服務時，且基於某些原因，會無法從其所設定的伺服器取得結果。身為唯一的本地使用者，根使用者是唯一仍可登入這些機器的使用者。下面是機器可能運作良好卻無法正確執行登入的一些常見原因：

- 網路未作用。如須對此情況的進一步指示，請參閱第 51.5 節「網路問題」[842頁]。
- DNS 此時未運作 (這樣會阻礙 GNOME 或 KDE 運作，也會妨礙系統驗證安全伺服器的要求)。若機器花費過久的時間回應任何動作的話，表示可能是這種情況。如需此主題的詳細資訊，請參閱第 51.5 節「網路問題」[842頁]。
- 若系統設定為使用 Kerberos，則系統的本地時間有可能超過了 Kerberos 伺服器時間所容許的變異 (一般為 300 秒)。若 NTP (網路時間協定) 未正確運作，或本地 NTP 伺服器未運作，則 Kerberos 驗證會停止作用，因為它必須仰賴網路上同步的共同時脈才可運作。
- 系統的驗證組態設定錯誤。請檢查 PAM 組態檔案是否有錯字或指示詞順序錯誤。如需關於 PAM 和所包含組態檔案語法的其他背景資料，請參閱第 27 章「*使用 PAM 驗證*」[453頁]。

對於所有非外部網路造成的問題，解決方案就是重新開機進入單一使用者模式，並修復組態後再次開機進入操作模式，以嘗試重新登入。若要開機進入單一使用者模式：

1 重新啟動系統。會出現開機畫面及提示。

2 在開機提示中輸入 1，讓系統開機進入單一使用者模式。

3 輸入根的使用者名稱與密碼。

4 進行必要的所有變更。

5 在指令行中輸入 `telinit 5`，開機進入完整多使用者及網路模式。

51.4.2 有效的使用者名稱和密碼不被接受

這顯然是使用者最常遇到的問題，其發生的原因有很多。根據您使用本地使用者管理和驗證，或使用網路驗證，會有不同原因造成登入失敗。

本地使用者管理可能因為下列原因而失敗：

- 使用者輸入的密碼錯誤。
- 使用者包含桌面組態檔的主目錄損毀或有防寫保護。
- X Window System 可能無法驗證此特定使用者，尤其是在安裝目前版本之前，此使用者的主目錄由其他 Linux 版本所使用的話。

若要找出本地登入失敗的原因，請執行下列步驟：

1 開始進行整個驗證機制的偵錯之前，先確認使用者可以正確記住密碼。若使用者可能記錯密碼，請使用 YaST 使用者管理模組變更使用者的密碼。

2 以根使用者登入，並檢查 `/var/log/messages` 中有沒有登入程序和 PAM 的錯誤訊息。

3 嘗試從主控台登入 (使用 Ctrl + Alt + F1)。如果成功，表示問題不在 PAM，因為它能夠在此機器上驗證此使用者。嘗試找出 X Window System 或桌面系統 (GNOME 或 KDE) 的任何問題。若需更多資訊，請參閱第 51.4.3 節「登入成功但 GNOME 桌面失敗 」 [840頁]和第 51.4.4 節「登入成功但 KDE 桌面失敗」 [841頁]。

4 若使用者的主目錄已由其他 Linux 版本使用，請移除使用者主目錄中的 `Xauthority` 檔案。使用主控台透過 Ctrl + Alt + F1 登入，並以此使用者身份執行 `rm .Xauthority`。這樣應可排除此使用者的 X 驗證問題。重新嘗試圖形登入。

5 若仍無法進行圖形登入，請以 Ctrl + Alt + F1 執行主控台登入。嘗試在其他畫面啟動 X 工作階段—第一個 (:0) 已被使用：

```
startx -- :1
```

這樣應可出現圖形畫面與您的桌面。若沒有的話，請檢查 X Window System 的記錄檔案 (/var/log/Xorg.*displaynumber*.log)，或您桌面應用程式的登入檔案 (位於使用者主目錄中的 .xsession-errors)，以得知是否有任何異常。

6 若由於組態檔案損毀導致桌面無法啟動，請繼續執行 第 51.4.3 節「登入成功但 GNOME 桌面失敗」[840頁] 或 第 51.4.4 節「登入成功但 KDE 桌面失敗」[841頁]。

下列為特定使用者在特定機器上網路驗證失敗的一些常見原因：

- 使用者輸入的密碼錯誤。
- 機器的本地驗證檔案中已存在使用者名稱，但網路驗證系統也提供了，兩者產生了衝突。
- 主目錄是存在的，但損毀或無法使用。或許此目錄設為防止寫入，或位於此時無法存取的伺服器上。
- 使用者沒有登入驗證系統特定主機的許可。
- 機器的主機名稱已因某種原因而變更，而使用者沒有登入該主機的許可。
- 機器無法聯繫驗證伺服器，或是含有使用者資訊的目錄伺服器。
- X Window System 可能無法驗證此特定使用者，尤其是若此使用者的主目錄由其他 Linux 版本所使用，而此版本優於目前所安裝版本的話。

若要找出登入發生網路驗證失敗的原因，請執行下列步驟：

1 開始進行整個驗證機制的偵錯之前，先確認使用者可以正確記住密碼。

2 決定機器賴以進行驗證的目錄伺服器，並確定該伺服器已啟動且在執行中，而且能夠正常與其他機器進行通訊。

3 確定使用者的使用者名稱和密碼可以在其他機器上使用，以確定其驗證資料存在，而且已正確配送。

4 再看看在運作不正常的機器上，可否讓其他使用者登入。若其他使用者可以正常登入，或根使用者可以登入的話，請登入並檢驗 `/var/log/messages` 檔案。找出嘗試登入所對應的時間戳記，並判斷 PAM 是否已產生任何錯誤訊息。

5 嘗試從主控台登入 (使用 Ctrl + Alt + F1)。若這樣成功的話，問題就不是出在 PAM 或使用者主目錄所在的目錄伺服器，因為能夠在此機器上驗證此使用者。嘗試找出 X Window System 或桌面系統 (GNOME 或 KDE) 的任何問題。若需更多資訊，請參閱第 51.4.3 節「登入成功但 GNOME 桌面失敗」[840 頁]和第 51.4.4 節「登入成功但 KDE 桌面失敗」[841 頁]。

6 若使用者的主目錄已由其他 Linux 版本使用，請移除使用者主目錄中的 `Xauthority` 檔案。使用主控台透過 Ctrl + Alt + F1 登入，並以此使用者身份執行 `rm .Xauthority`。這樣應可排除此使用者的 X 驗證問題。重新嘗試圖形登入。

7 若仍無法進行圖形登入，請以 Ctrl + Alt + F1 執行主控台登入。嘗試在其他畫面啟動 X 工作階段—第一個 (`:0`) 已被使用：

```
startx -- :1
```

這樣應可出現圖形畫面與您的桌面。若沒有的話，請檢查 X Window System 的記錄檔案 (`/var/log/Xorg.displaynumber.log`)，或您桌面應用程式的登入檔案 (位於使用者主目錄中的 `.xsession-errors`)，以得知是否有任何異常。

8 若由於組態檔案損毀導致桌面無法啟動，請繼續執行第 51.4.3 節「登入成功但 GNOME 桌面失敗」[840 頁] 或 第 51.4.4 節「登入成功但 KDE 桌面失敗」[841 頁]。

51.4.3 登入成功但 GNOME 桌面失敗

若這是特定使用者的問題，很可能是該使用者的 GNOME 組態檔案已損毀。某些症狀還會包括鍵盤無法運作、螢幕幾何錯亂，甚至螢幕只呈現空白的灰色區塊。此問題最重要的區隔是，若其他使用者可以登入，則此機器是正常運作的。若這樣的話，問題很可能可以快速解決，只要將使用者的 GNOME 組態目錄移到新的位置，使 GNOME 啟始化新的組態目錄即可。雖然這樣算是強制使用者重新設定 GNOME，但並沒有資料因此遺失。

1 按Ctrl + Alt + F1切換到文字主控台。

2 使用您的使用者名稱登入。

3 將使用者的 GNOME 組態目錄移到一個暫時位置：

```
mv .gconf  .gconf-ORIG-RECOVER
mv .gnome2 .gnome2-ORIG-RECOVER
```

4 登出。

5 再次登入，勿執行任何應用程式。

6 依照下列方式將 ~/.gconf-ORIG-RECOVER/apps/ 目錄複製回新的 ~/.gconf 目錄，修復您的個別應用程式組態資料 (包括 Evolution 電子郵件用戶端資料)：

```
cp -a .gconf-ORIG-RECOVER/apps .gconf/
```

若這樣導致登入問題，請嘗試僅修復重要的應用程式資料，並重新設定其他的應用程式。

51.4.4 登入成功但 KDE 桌面失敗

KDE 桌面無法讓使用者登入有幾個原因。快取資料損毀以及 KDE 桌面組態檔案損毀，均可能導致登入問題。

快取檔案可用於增進桌面啟動的效能。若此資料損毀的話，啟動速度會變慢或完全故障。請將其移除，並強制桌面啟動常式重新開始啟動。這樣會比正常啟動的時間久，但將不會損害資料的保存並讓使用者登入。

若要移除 KDE 桌面的快取檔案，請以*根使用者*身份發出以下指令：

```
rm -rf /tmp/kde-user /tmp/socket-user
```

以實際的使用者名稱取代 *user*。移除這兩個目錄只會移除損毀的快取檔案，過程中並不會傷害到真正的資料。

我們永遠可以使用初始組態檔案取代損毀的桌面組態檔案。若您希望修復使用者的調整值，請在以預設組態值復原組態之後，小心地將其從暫存位置複製回原來位置。

若要以初始組態值取代損毀的桌面組態，請執行下列步驟：

1 按Ctrl + Alt + F1切換到文字主控台。

2 使用您的使用者名稱登入。

3 將 KDE 組態目錄和 `.skel` 檔案移動到暫時位置：

```
mv .kde  .kde-ORIG-RECOVER
mv .skel .skel-ORIG-RECOVER
```

4 登出。

5 再次登入。

6 成功啟動桌面後，將使用者自己的組態複製回原位：

```
cp -a .kde-ORIG-RECOVER/share .kde/share
```

重要

若使用者自己的調整值導致登入失敗，且一再重複，請重做以上步驟，但不要複製 `.kde/share` 目錄。

51.5 網路問題

您系統的許多問題可能都與網路有關，但可能一開始看不出來。例如，系統不允許使用者登入，可能就是某種網路問題所致。此節會介紹簡單的檢查清單，可讓您用來辨識所遇到網路問題的原因。

檢查機器網路連線時，請如下執行：

1 若使用乙太網路連線，請先檢查硬體。確認您的網路線正確插在電腦上。若有控制燈的話，乙太網路接頭旁的控制燈應均呈現作用中的狀態。

若連接失敗，請檢查網路線在其他機器上可否使用。若可以的話，就是您的網路卡造成的問題。若您的網路設定中包含了集線器或切換器，請檢查這些設備是否有問題。

2 若使用無線連接的話，請檢查可否由其他機器建立無線連結。若不是此狀況的話，請聯絡您的無線網路管理員。

3 檢查完基本網路連線之後，請嘗試找出未回應的服務為何。收集您設定中所需所有網路伺服器的位址資訊。您可在適當的 YaST 模組中查詢，或詢問您的系統管理員。下列清單提供了設定中所包含的一些基本的網路伺服器，以及其故障的症狀。

DNS (名稱服務)
: 名稱服務損壞或故障會在許多方面影響網路的功能。若本地網路仰賴網路伺服器進行驗證，而因為名稱解析問題而找不到這些伺服器的話，使用者就無法登入了。故障名稱伺服器所管理的機器將無法「看到」彼此並進行通訊。

NTP (時間服務)
: NTP 服務的損壞或完全故障會影響 Kerberos 驗證以及 X 伺服器的功能。

NFS (檔案服務)
: 若應用程式所需的資料儲存於裝載 NFS 的目錄中，萬一此服務關閉或設定錯誤，該應用程式會無法啟動或無法正常運作。最糟糕的情況是，如果由於 NFS 伺服器損耗，而找不到包含 `.gconf` 或 `.kde` 子目錄的主目錄，則使用者的個人桌面組態將無法出現。

Samba (檔案服務)
: 若應用程式所需的資料儲存於 Samba 伺服器的目錄中，則若此服務損壞的話，該應用程式會無法啟動或無法正常運作。

NIS (使用者管理)
: 若您的 SUSE Linux Enterprise 系統仰賴 NIS 伺服器提供使用者資料，萬一 NIS 服務關閉，使用者就無法登入這個機器。

LDAP (使用者管理)
: 若您的 SUSE Linux Enterprise 系統仰賴 LDAP 伺服器提供使用者資料，萬一 LDAP 服務關閉，使用者就無法登入這個機器。

Kerberos (驗證)
: 無法進行驗證，且無法登入任何機器。

CUPS (網路列印)
: 使用者無法列印。

4 請檢查網路伺服器是否運作，且您的網路設定可否讓您建立連接：

重要

下述偵錯程序只適用於不涉及任何內部路由的簡易網路伺服器/用戶端設定。假設伺服器和用戶端都是相同子網路的成員，不需要其他路由。

4a 使用 `ping hostname` (將 *hostname* 取代為伺服器的主機名稱)，檢查各部機器是否開啟，且能否回應網路。若此指令成功的話，就會告知您的主機您正在尋找並執行它，且您網路的名稱服務設定是正確的。

若 ping 的結果失敗且傳回 `destination host unreachable` (無法聯繫目的地主機)，則您的系統或想找的伺服器可能設定錯誤或故障。請從其他機器執行 `ping your_hostname`指令。如果您可以從另一台機器存取您的機器，可能是伺服器完全未執行或設定錯誤。

若 ping 失敗且傳回 `unknown host` (未知主機)，則是名稱服務設定錯誤，或使用的主機名稱不正確。請使用 `ping -nipaddress` 指令，以名稱以外的方式連接此主機。若這樣成功的話，請檢查主機名稱的拼法是否正確，以及您網路上的名稱服務設定是否正確。如須對此問題做進一步檢查，請參閱步驟 4b [844頁]。若 ping 仍然失敗，則是您的網路卡未設定正確，或網路硬體故障。請參閱步驟 4c [845頁]來取得相關資訊。

4b 使用`host hostname`指令，檢查您嘗試連接的伺服器主機名稱是否正確轉譯為 IP 位址，反之亦然。若此指令傳回主機的 IP 位址，則名稱服務是啟動且執行中的。如果 `host` 指令失敗，請在您的主機上檢查所有與名稱及位址解析有關的網路組態檔案：

`/etc/resolv.conf`

此檔案用於追蹤您目前使用的名稱伺服器與領域。您可以手動修改此檔案，或以 YaST 或 DHCP 自動調整。建議您採用自動調整。然而，請確定此檔案的結構如下，且所有的網路位址與領域名稱均正確：

```
search fully_qualified_domain_name
nameserver ipaddress_of_nameserver
```

此檔案會包含多個名稱伺服器位址，其中至少有一個必須是正確的，才能為您的主機提供名稱解析。需要的話，請使用 YaST DNS 和主機名稱模組調整此檔案。

若您是透過 DHCP 處理網路連線，請在 YaST DNS 和主機名稱模組中選取「*透過 DHCP 變更主機名稱*」和「*透過 DHCP 更新名稱伺服器和搜尋清單*」，以啟用 DHCP 來變更主機名稱與名稱服務資訊。

`/etc/nsswitch.conf`

此檔案會告知 Linux 何處可找到名稱服務資訊。檔案外觀如下：

```
 ...
hosts: files dns
networks: files dns
...
```

`dns` 項目是必備的。這會告訴 Linux 使用外部名稱伺服器。正常情況下，YaST 會自動建立這些項目，而檢查並無損這些項目。

若主機上所有相關的項目都正確的話，請要求您的系統管理員檢查 DNS 伺服器組態是否具備正確的時區資訊。如需關於 DNS 的詳細資訊，請參閱 第 33 章「*網域名稱系統*」[559頁]。若您確定主機和 DNS 伺服器的 DNS 組態正確物誤，請繼續檢查網路組態和網路設備。

4c 若您的系統無法建立與網路伺服器的連線，且您已經從問題可能原因清單中排除名稱服務的問題，則請檢查網路卡的組態。

使用指令 `ifconfig network_device`(以根使用者身份執行) 來檢查設備是否正確設定。請確認 `inet address` 和 `Mask` 均已設定正確。IP 位址有錯誤或網路遮罩有位元遺失的話，都可能造成網路組態無法使用。必要的話，請一併於伺服器上執行此檢查。

4d 如果名稱服務與網路硬體均設定正確且在執行中，但仍有些外部網路連線逾時或完全失敗，請使用 `traceroute` *`fully_qualified_domain_name`* (以根使用者身份執行) 來追蹤這些要求所採用的網路路由。此指令會列出要求從您機器傳送到其目的地所經的所有閘道 (躍程)。其會列出各躍程的回應時間，以及是否能取得此躍程。請使用 traceroute 加上 ping 找出問題的原因，並告知管理員。

一旦您辨識出網路問題的原因，就可以自己解決(若問題發生在您機器上的話)，或將您的發現告訴網路系統管理員，讓他們可以重新設定服務，或修復必要的系統。

51.5.1 NetworkManager 問題

若您有網路連線的問題，請依 [842頁]所述將範圍調窄。若 NetworkManager 似乎有問題，請執行下列步驟，取得 NetworkManager 故障原因的提示記錄：

1. 開啟外圍程序並以 `root` 身份登入。
2. 重新啟動 NetworkManager：

   ```
   rcnetwork restart -o nm
   ```

3. 以一般使用者身份開啟網頁，例如 `http://www.opensuse.org`，看看是否可以連接。
4. 在 `/var/log/NetworkManager` 中收集 NetworkManager 狀態的所有相關資訊。

如需關於 NetworkManager 的詳細資訊，請參閱第 30.6 節「使用 NetworkManager 管理網路連線」 [529頁]。

51.6 資料問題

資料問題是指，機器可能可以 (或無法) 正確開機，但另一方面，系統上有著明顯的資料損毀，且必須修復系統。遇到這些情況的話，就需要用到您重要資料的備份檔案，以讓您的系統回復到故障前的狀態。SUSE Linux Enterprise 提供了

專用的 YaST 模組，進行系統備份與還原，並可從外部對損毀的系統提供救援並予以修復。

51.6.1 備份重要資料

使用 YaST 系統備份模組，可輕鬆管理系統備份：

1. 以根使用者身份啟動 YaST，並選取「*系統*、」>「*系統備份*」。
2. 建立持有備份所需所有細節、歸檔檔名、範圍和備份類型等資訊的備份設定檔：
   - **2a** 選取「*設定檔管理*」>「*新增*」。
   - **2b** 輸入歸檔的名稱。
   - **2c** 若您希望將備份保存於本地，請輸入備份的路徑與位置。若您要將備份歸檔於網路伺服器 (透過 NFS)，請輸入要存放歸檔的 IP 位址或伺服器名稱和目錄。
   - **2d** 決定歸檔類型，並按一下「*下一步*」。
   - **2e** 決定要使用的備份選項，如不隸屬任何套件的檔案是否要備份，以及建立歸檔前是否要先顯示檔案清單。同時也要決定是否要使用耗時的 MD5 機制，辨識檔案是否已變更過。

     使用「*進階*」進入備份整個硬碟區域的對話。此選項目前只適用於 Ext2 檔案系統。
   - **2f** 最後，請設定搜尋限制，以將不需備份的特定系統區域從備份區域中排除，如所定的檔案或是快取檔案。新增、編輯或刪除項目，直到符合您的需求為止，再按「*確定*」離開。
3. 完成設定檔設定之後，您就可以立即使用「*開始*」開始備份，或設定自動備份。您亦可針對其他不同目的，建立其他設定檔。

若要為已知設定檔設定自動備份，請如下操作：

1. 從「*設定檔管理*」中選取「*自動備份*」功能表。

2 選取「*開始自動備份*」。

3 決定備份的頻率。選擇「*每日*」、「*每週*」或「*每月*」。

4 決定備份開始時間。這些設定會根據所選取的備份頻率而定。

5 決定是否保留舊的備份，以及要保留多少。若要收到自動產生的備份程序狀態訊息，請按一下「*傳送摘要郵件給使用者 root*」。

6 按一下「*確定*」套用您的設定，並於指定的時間啟動第一次備份。

51.6.2 還原系統備份

使用 YaST 系統還原模組，從備份還原系統組態。您可還原整個備份，或選取特定損毀而需要重設置舊狀態的元件。

1 啟動「*YaST*」>「*系統*」>「*系統還原*」。

2 輸入備份檔案的位置。這可能是本地檔案、掛載於網路的檔案，或軟碟或 CD 等抽取式設備中的檔案。然後按「*下一步*」。

   下列對話方塊會顯示歸檔內容的摘要，如檔案名稱、建立日期、備份類型和選用的註解。

3 按一下「*歸檔內容*」檢視歸檔的內容。按一下「*確定*」返回「*歸檔內容*」對話方塊。

4 會開啟「*進階選項*」對話方塊，讓您微調還原程序。按一下「*確定*」即可回到「*歸檔內容*」對話。

5 按一下「*下一步*」開啟要還原的套件檢視畫面。按「*接受*」還原歸檔中的所有檔案，或使用各式各樣的「*全選*」、「*取消全選*」和「*選取檔案*」按鈕，微調您的選擇。只有當 RPM 資料庫損毀或已刪除，而且此檔案包含於備份中時，才使用「*還原 RPM 資料庫*」選項。

6 按一下「*接受*」之後，就會還原備份。備份程序完成後，請按一下「*結束*」離開模組。

51.6.3 修復損毀的系統

系統無法啟動並正常運作的原因可能有幾種。系統當機後檔案損毀、組態檔案損毀，或最常見的是，開機載入程式組態損毀。

SUSE Linux Enterprise 提供兩種不同的方法來處理這種狀況。您可以使用 YaST 系統修復功能，或啟動救援系統。下列章節會討論這兩種系統修復方式：

使用 YaST 系統修復

啟動「YaST 系統修復」模組前，請判定要使用哪種模式執行，以符合您的需求。根據精確程度、您系統故障的原因，以及您的專精程度，共有三種模式可選：

自動修復
: 若您的系統會因為未知的原因故障，而您基本上不知道系統的哪個部分導致故障，請使用「*自動修復*」。這會對所安裝系統的所有元件進行大規模的自動檢查。如需此程序的詳細描述，請參閱章節「自動修復」[849頁]。

自定修復
: 若您的系統故障，且您已經知道哪個元件造成故障，請將系統分析的範圍限制在那些元件上，停止「*自動修復*」冗長的系統檢查。例如，若故障前的系統訊息指示套件資料庫有錯誤，則您可將分析與修復程序限定於檢查並復原系統的這一部份。如需此程序的詳細描述，請參閱章節「自定修復」[851頁]。

進階工具
: 若您已經對於造成系統故障的元件有清楚的概念，且知道如何修復，您可略過分析執行，並直接套用修復此部份元件所需的工具。如需詳細資訊，請參閱章節「進階工具」[852頁]。

如上所述選擇修復模式之一，並如以下章節所述繼續進行系統修復。

自動修復

若要啟動「YaST 系統修復」的自動修復模式，請如下執行：

1. 將 SUSE Linux Enterprise 的安裝媒體插入光碟機或 DVD 光碟機中。
2. 重新啟動系統。

3 在開機畫面中，選取「*安裝*」。

4 選取語言並按一下「*下一步*」。

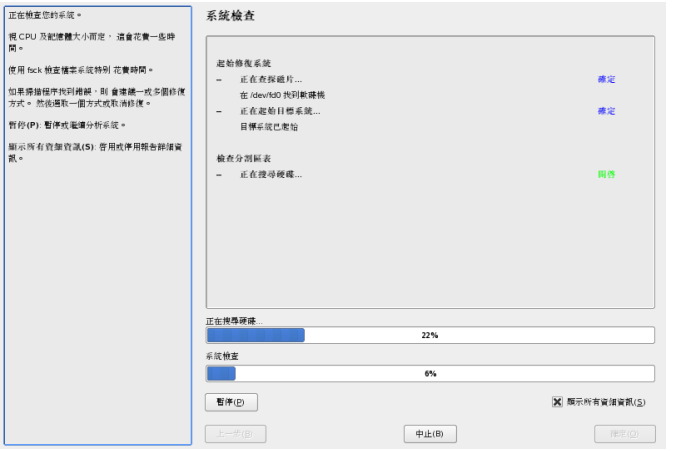
5 確認授權合約並按一下「*下一步*」。

6 在「*系統分析*」中，選取「*其他*、」>「*修復已安裝系統*」。

7 選擇「*自動修復*」。

YaST 會立即啟動已安裝系統的大規模分析。程序的進度會以畫面底部的兩個進度列顯示。上面的進度列顯示目前執行測試的進度。下面的進度列顯示分析程序的整體進度。上方區段的記錄視窗會追蹤目前執行中的測試及其結果。請參閱圖形 51.2「自動修復模式」[850頁]。每次執行時都會執行下列主測試執行。它們包括一些個別子測試。

圖形 51.2 自動修復模式



所有硬碟的分割區表
: 對所有偵測到之硬碟的分割區表，進行有效性和連貫性的檢查。

交換分割區
: 會針對已安裝系統的交換分割區進行偵測、測試，並在適用的情況下提議啟用。為達更高的系統修復速度就會接受這種提議。

檔案系統
所有偵測到的檔案系統都會進行檔案系統的特定檢查。

檔案 `/etc/fstab` 中的項目
針對檔案中的項目進行完整性和一致性的檢查。所有有效分割區都會進行裝載。

開機載入程式組態
針對已安裝系統 (GRUB 或 LILO) 的開機載入程式組態，檢查其完整性和一致性。這時會檢查開機和 root 設備，並檢查 initrd 模組的可用性。

套件資料庫
檢查是否具有最小安裝作業所需的所有套件。另外一個可用選項就是分析其基本套件。這很耗時，因為套件的數量十分龐大。

8 發生錯誤時，程序會停止，並開啟對話方塊，說明詳細資訊及可能的解決方法。

請仔細閱讀畫面訊息，再接受建議的修復動作。若您決定拒絕建議的解決方案，系統會保留不變。

9 修復程序成功完成之後，請按一下「*確定*」和「*完成*」，並移除安裝媒體。這樣系統就會自動重新開機。

自定修復

若要啟動「*自定修復*」模式，並選擇性檢查已安裝系統的特定元件，請依照下列步驟執行：

1 將 SUSE Linux Enterprise 的安裝媒體插入光碟機或 DVD 光碟機中。

2 重新啟動系統。

3 在開機畫面中，選取「*安裝*」。

4 選取語言並按一下「*下一步*」。

5 確認授權合約並按一下「*下一步*」。

6 在「*系統分析*」中，選取「*其他*、」>「*修復已安裝系統*」。

7 選擇「*自定修復*」。

選擇「*自定修復*」會顯示測試執行清單，這些測試執行最初都會標記為執行。測試的整個範圍會與自動修復相符。如果您已經知道系統沒有受損，請取消相對應測試的標記。按一下「*下一步*」，然後啟動範圍較小的測試程序，它所需的執行時間可能短的多。

不是所有測試群組都可以個別套用。檔案系統的檢驗一定會結合 fstab 項目的分析，包括現有的交換分割區。YaST 會選擇必要測試的最小執行數量，自動解析這樣的相依性。

8 發生錯誤時，程序會停止，並開啟對話方塊，說明詳細資訊及可能的解決方法。

請仔細閱讀畫面訊息，再接受建議的修復動作。若您決定拒絕建議的解決方案，系統會保留不變。

9 修復程序成功完成之後，請按一下「*確定*」和「*完成*」，並移除安裝媒體。這樣系統就會自動重新開機。

進階工具

如果您瞭解 SUSE Linux Enterprise而且清楚知道需要修復系統中的什麼項目，請略過系統分析，直接套用工具。

若要使用「YaST 系統修復」模組的「*進階工具*」功能，請依照下列步驟執行：

1 使用最初安裝的原始安裝媒體來為系統開機 (詳細資訊請參閱第 3 章「*使用 YaST安裝*」[17頁])。

2 在「*系統分析*」中，選取「*其他*」>「*修復已安裝系統*」。

3 選取「*進階工具*」並選擇一或多個修復選項。

4 修復程序成功完成之後，請按一下「*確定*」和「*完成*」，並移除安裝媒體。這樣系統就會自動重新開機。

進階工具可提供下列選項來修復故障的系統：

安裝新的開機載入程式
: 這會啟動 YaST 開機載入程式組態模組。如需詳細資訊，請參閱第 21.3 節「使用 YaST 設定開機載入程式」[380頁]。

啟動磁碟分割工具
: 這會啟動 YaST 裡的進階分割工具。

修復檔案系統
: 這會檢查您已安裝系統中的檔案系統。系統會先提供所有偵測到之分割區的選單，供您選擇一個來進行檢查。

復原遺失的分割區
: 您可以嘗試重建受損的分割區表。首先會提供所偵測到的硬碟清單，供您選擇。按一下「*確定*」，啟動檢查。所需時間視處理能力和硬碟大小而定。

重要：重建分割區表

重建分割區表有些麻煩。YaST 會分析硬碟的資料磁區，嘗試辨識遺失的分割區。辨識之後，會將遺失的分割區新增至重新建好的分割區表。不過，這個方法並不一定適用於每個所想得到的例子。

將系統設定儲存至軟碟
: 這個選項將重要的系統檔案儲存至軟碟。如果這些檔案中任何一個受損時，就可以由磁片還原。

驗證安裝的軟體
: 這會檢查套件資料庫的一致性，和最重要套件的可用性。任何已安裝套件在受損時都可以使用這個工具重新安裝。

使用救援系統

SUSE Linux Enterprise 包含一個救援系統。救援系統是一個小型的 Linux 系統，可以載入到 RAM 磁碟上並裝載為根目錄檔案系統，好讓您從外部存取 Linux 分割區。藉由此救援系統，您可以復原或修改任何重要的系統項目：

- 操作任何類型的組態檔案。
- 檢查檔案系統有無缺失並啟動自動修復程序。
- 在「變更根目錄」環境中存取已安裝的系統。

- 檢查、修改和重新安裝開機載入程式設定。
- 使用 Parted 指令來調整分割區大小。如需此工具的詳細資訊，請參閱 GNU Parted 網站 (http://www.gnu.org/software/parted/parted.html)。

救援系統可以從各種來源與位置載入。最簡單的方法是原始安裝光碟或DVD啟動救援系統：

1. 將安裝媒體插入光碟機或 DVD 光碟機中。
2. 重新啟動系統。
3. 在開機畫面中，選擇「*救援系統*」選項。
4. 在 `Rescue:` 提示輸入 `root`。無須輸入密碼。

如果硬體安裝不包括 CD 或 DVD 光碟機，您可以從網路來源啟動救援系統。下列範例將套用至遠端開機方案—如果使用其他開機媒體(例如軟碟機)，請據此修改 `info` 檔案，並按照一般安裝方式開機。

1. 輸入 PXE 開機設定的組態，將 `install=`*`protocol`*`://`*`instsource`* 取代為 `rescue=`*`protocol`*`://`*`instsource`*。正如一般安裝的情況，*`protocol`* 代表任何支援的網路通訊協定 (NFS、HTTP、FTP 等等)，而 *`instsource`* 代表網路安裝來源的路徑。
2. 依照第 4.3.7 節「區域網路喚醒」 [68頁]中的說明，使用「網路喚醒」啟動系統。
3. 在 `Rescue:` 提示輸入 `root`。無須輸入密碼。

一旦進入救援系統後，您可以利用 Alt + F1 至 Alt + F6 來使用虛擬主控台。

`/bin` 目錄中提供外圍程序以及其他許多有用的公用程式，例如 mount 程式。`sbin`目錄中包含重要的檔案與網路公用程式，以便檢視及修復檔案系統。此目錄中也包含最重要的二進位系統維護程式，例如 fdisk、mkfs、mkswap、mount、mount、init 及 shutdown，還有維護網路的 ifconfig、ip、route 以及 netstat。目錄 `/usr/bin` 包含 vi 編輯器、find、less 以及 ssh。

若要檢視系統訊息，請使用指令 `dmesg` 或檢視檔案 `/var/log/messages`。

檢查和操作組態檔案

為了舉例說明使用救援系統如何修正組態檔案，請想像系統由於組態檔案損毀而無法正常開機。您可以使用救援系統來解決這個問題。

若要操作組態檔案，請執行下列步驟：

1. 使用上述的其中一個方法啟動救援系統。
2. 若要將 /dev/sda6 下的開機檔案系統裝載到救援系統，請使用下列指令：

   ```
   mount /dev/sda6 /mnt
   ```

   系統的所有目錄現在都存放在 /mnt 中
3. 將此目錄變更到裝載的開機檔案系統中：

   ```
   cd /mnt
   ```
4. 在 vi 編輯器中開啟有問題的組態檔案。調整並儲存設定。
5. 從救援系統解除裝載開機檔案系統：

   ```
   umount /mnt
   ```
6. 重新開機。

修復和檢查檔案系統

一般而言，檔案系統無法在執行中的系統上修復。如果發生了嚴重的問題，您可能甚至無法裝載開機檔案，而且系統可能會因為核心異常而無法開機。在此情況下，唯一的方法就是從外部修復系統。強烈建議您使用 YaST 系統修復來執行這項任務 (請參閱 章節「使用 YaST 系統修復」 [849頁] 以取得詳細資訊)。不過，如果您需要執行手動檔案系統檢查或修復，請啟動救援系統。其中包含的公用程式可檢查和修復 ext2、ext3、reiserfs、xfs、、dosfs 以及 vfat 檔案系統。

存取已安裝的系統

若您需要從救援系統存取已安裝的系統(例如，您要修改開機載入程式組態或執行硬體組態公用程式，您必須在「變更根目錄」環境下執行這個動作。

若要根據已安裝的系統設定「變更根目錄」環境，請執行下列步驟：

1 首先，從已安裝的系統和設備檔案系統裝載根分割區：

```
mount /dev/sda6 /mnt
mount --bind /dev /mnt/dev
```

2 現在您可以「變更根目錄」至新環境：

```
chroot /mnt
```

3 接著裝載 `/proc` and `/sys`：

```
mount /proc
mount /sys
```

4 最後，從已安裝的系統裝載其餘分割區：

```
mount -a
```

5 現在您可以存取已安裝的系統。重新啟動系統之前，請先使用 `umount -a` 來解除裝載分割區，並以 `exit` 離開「變更根」目錄環境。

警告：限制

雖然您可以完全存取已安裝系統的檔案和應用程式，但必須遵守某些限制。所執行的核心是以救援系統啟動的核心。它只支援基本硬體，而且無法從已安裝系統新增核心模組，除非核心版本完全一致 (這種機率很低)。因此舉例而言，您將無法存取音效卡。您也無法啟動圖形使用者介面。

另外請注意，當您使用 Alt + F1 至 Alt + F6 來切換主控台時，將會離開「變更根」目錄環境。

修改和重新安裝開機載入程式

有時候系統無法開機是因為開機載入程式組態已損毀。譬如說，開機載入程式若未執行，啟動常式便無法將實體磁碟機轉譯為Linux檔案系統中的實際位置。

若要檢查開機載入程式組態和重新安裝開機載入程式，請執行下列步驟：

1. 執行存取已安裝系統所需的必要步驟，如 章節「存取已安裝的系統」[856頁] 所述。
2. 根據 第 21 章「*開機載入程式*」[371頁] 中概述的 GRUB 組態原則，檢查下列檔案是否正確設定。

   - `/etc/grub.conf`
   - `/boot/grub/device.map`
   - `/boot/grub/menu.lst`

   必要時，針對設備映射(`device.map`)或開機分割區與組態檔案的位置套用修正程式。
3. 依序使用下列指令來重新安裝開機載入程式：

   ```
   grub --batch < /etc/grub.conf
   ```
4. 解除裝載分割區，從「變更根目錄」環境登出，並重新啟動系統：

   ```
   umount -a
   exit
   reboot
   ```

51.7 IBM System z：使用 initrd 做為救援系統

如果升級或修改 IBM System z 的 SUSE® Linux Enterprise Server 核心，可能會意外以不一致的狀態重新啟動系統，使得對已安裝系統執行 IPL 的標準程序失敗。在已經安裝新的或更新的 SUSE Linux Enterprise Server 核心，但尚未執行

zipl程式來更新IPL記錄時，最容易發生這種意外。在此狀況下，請使用標準安裝套件做為救援系統，可在其中執行 zipl 程式以更新 IPL 記錄。

51.7.1 IPL 救援系統

重要：提供使用安裝資料

若要使用這種方法，必須取得適用於 IBM System z 的 SUSE Linux Enterprise Server 的安裝資料。如需詳細資訊，請參閱 *Architecture-Specific Information* 中的第 2.1 節「Making the Installation Data Available」(第 2 章「*Preparing for Installation*」, ↑*Architecture-Specific Information*)。除此之外，您需要設備的通道號碼和包含 SUSE Linux Enterprise Server 安裝的根目錄檔案系統的設備內分割區數目。

首先，對適用於 IBM System z 的 SUSE Linux Enterprise Server 安裝系統執行 IPL，如*Architecture-Specific Information*手冊中所述。然後會出現使用的網路介面卡選項清單。

選取「*啟動安裝或系統*」，然後選取「*啟動救援系統*」以啟動救援系統。視安裝環境而定，您現在必須指定網路介面卡的參數和安裝來源。救援系統便會載入，且結尾會顯示下列登入提示：

```
Skipped services in runlevel 3:  nfs nfsboot

Rescue login:
```

現在可以使用 `root` 身份登入，無需密碼。

51.7.2 設定磁碟

在此狀態下，尚未設定任何磁碟。您必須先設定磁碟，才能繼續。

過程 51.3 *設定 DASD*

1 使用下列指令來設定 DASD：

```
dasd_configure 0.0.0150 1 0
```

0.0.0150 是連接 DASD 的通道。1 表示啟用磁碟 (此處的 0 會停用磁碟)。0 表示磁碟的「無 DIAG 模式」(此處的 1 會啟用磁碟的 DAIG 存取)。

2 現在 DASD 已經上線 (請使用 `cat /proc/partitions` 來檢查)，而且可以用於後續指令。

*過程 **51.4** 設定 zFCP 磁碟*

1 若要設定 zFCP 磁碟，您必須先設定 zFCP 介面卡。使用下列指令來執行此動作：

```
zfcp_host_configure 0.0.4000 1
```

0.0.4000 是連接介面卡的通道，而 1 表示啟動 (此處的 0 會停用介面卡)。

2 啟用介面卡之後，便可以設定磁碟。使用下列指令來執行此動作：

```
zfcp_disk_configure 0.0.4000 1234567887654321 8765432100000000  1
```

0.0.4000 是之前使用的通道 ID，1234567887654321 是 WWPN (全球連接埠號碼)，而 8765432100000000 則是 LUN (邏輯單元編號)。1 表示啟用磁碟 (此處的 0 會停用磁碟)。

3 現在 zFCP 磁碟已經上線 (請使用 `cat /proc/partitions` 來檢查)，而且可以用於後續指令。

51.7.3 裝載根設備

如果所有所需磁碟均已上線，您現在就應該能夠裝載根設備。假設根設備位於 DASD 設備的第二分割區 (/dev/dasda2)，對應的指令為 `mount /dev/dasda2 /mnt`。

重要：檔案系統一致性

如果安裝的系統未正確關閉，建議在裝載之前檢查檔案系統一致性。如此可避免意外遺失資料。使用此範例，發出指令 `fsck /dev/dasda2` 以確定檔案系統處於一致的狀態。

僅需發出指令 mount，便可以檢查檔案系統是否已正確裝載。

範例 51.1 *裝載指令的輸出*

```
SuSE Instsys suse:/ # mount
shmfs on /newroot type shm (rw,nr_inodes=10240)
devpts on /dev/pts type devpts (rw)
virtual-proc-filesystem on /proc type proc (rw)
/dev/dasda2 on /mnt type reiserfs (rw)
```

51.7.4 變更至已裝載的檔案系統

zipl 指令可從安裝的系統之根設備讀取組態檔，而非從救援設備，可使用 chroot 指令來變更安裝系統的根設備：

範例 51.2 *chroot 至已裝載的檔案系統*

```
SuSE Instsys suse:/ # cd /mnt
SuSE Instsys suse:/mnt # chroot /mnt
```

51.7.5 執行 zipl

現在執行 zipl 將正確值重新寫入 IPL 記錄：

範例 51.3 *使用 zipl 安裝 IPL 記錄*

```
sh-2.05b# zipl
building bootmap : /boot/zipl/bootmap
adding Kernel Image : /boot/kernel/image located at 0x00010000
adding Ramdisk : /boot/initrd located at 0x00800000
adding Parmline : /boot/zipl/parmfile located at 0x00001000
Bootloader for ECKD type devices with z/OS compatible layout installed.
Syncing disks....
...done
```

51.7.6 結束救援系統

若要結束救援系統，首先以 `chroot` 指令和 `exit` 讓外圍程序保持開啟。若要防止資料遺失，請將所有未寫入的緩衝區以 `sync` 指令沖洗至磁碟。現在變更救援系統的根目錄，並解除裝載適用於 IBM System z 的 SUSE Linux Enterprise Server 安裝的根設備。

範例 51.4 *解除裝載檔案系統*

```
SuSE Instsys suse:/mnt # cd /
SuSE Instsys suse:/ # umount /mnt
```

最後，以 `halt` 指令停止救援系統。現在可以如 第 3.13.1 節「IBM System z：對安裝的系統執行 IPL」[32頁] 所述對 SUSE Linux Enterprise Server 系統執行 IPL。