

Kommentarer till utgåvan Debian GNU/Linux 5.0 (lenny), ARM EABI

The Debian Documentation Project (<http://www.debian.org/doc/>)

19 september 2009

Kommentarer till utgåvan Debian GNU/Linux 5.0 (lenny), ARM EABI

Published 2009-08-23

Detta dokument är fri mjukvara; du kan vidare distribuera det och/eller modifiera det i enlighet med villkoren i Free Software Foundations GNU General Public License version 2.

Detta program är distribuerat med förhoppning att det ska vara användbart men HELT UTAN GARANTIER; inte ens underförstådd garanti om SÄLJBARHET eller att PASSA ETT SÄRSKILT SYFTE. Läs mer i GNU General Public License för djupare detaljer.

Du borde ha fått en kopia av GNU General Public License tillsammans med det här programmet; om inte, skriv till Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA, 02110-1301 USA.

Licenstaget kan också hämtas på <http://www.gnu.org/copyleft/gpl.html> (<http://www.gnu.org/copyleft/gpl.html>) och `/usr/share/common-licenses/GPL-2` på Debian GNU/Linux.

Innehåll

1	Introduktion	3
1.1	Rapportera fel i det här dokumentet	3
1.2	Bidra med uppgraderingsrapporter	3
1.3	Källor för det här dokumentet	4
2	Vad är nytt i Debian GNU/Linux 5.0	5
2.1	Vad är nytt för ARM?	5
2.2	Vad är nytt i distributionen?	6
2.2.1	Pakethantering	6
2.2.2	Om proposed-updates (föreslagna förändringar)	7
2.3	Systemförbättringar	7
2.4	Större kärnrelaterade ändringar	8
2.4.1	Ändringar i paketering av kärnan	8
2.5	Emdebian 1.0 (baserad på Debian GNU/Linux lenny 5.0)	8
2.6	Netbook-stöd	8
2.7	Java, nu i Debian	8
3	Installationssystem	9
3.1	Vad är nytt i installationssystemet?	9
3.1.1	Stora ändringar	9
3.1.2	Automatisk installation	10
4	Uppgraderingar från tidigare utgåvor	11
4.1	Förberedelse inför uppgraderingen	11
4.1.1	Säkerhetskopiera all data eller konfigurationsinformation	11
4.1.1.1	Kontrollera att du har korrekt kärna	11
4.1.2	Informera användarna i förväg	11
4.1.3	Förbered för återställning	12
4.1.3.1	Felsökningsskal under uppstart med hjälp av initrd	12
4.1.4	Förbered en säker miljö för uppgraderingen	12
4.2	Kontrollera systemets status	13
4.2.1	Kontrollera kommande åtgärder i pakethanteraren	13
4.2.2	Inaktivera APT-nålning	13
4.2.3	Kontrollera paketstatus	13
4.2.4	Avsnittet proposed-updates	14
4.2.5	Inofficiella källor och bakåtporteringar	14
4.2.5.1	Att använda paket från <code>backports.org</code>	14
4.3	Avmarkera paket manuellt	14
4.4	Förbered källor för APT	15
4.4.1	Lägg till APT-källor på Internet	15
4.4.2	Lägg till APT-källor för en lokal spegelservr	16
4.4.3	Lägg till APT-källa från cd-rom eller dvd	16
4.5	Uppgradering av paket	16
4.5.1	Spela in sessionen	17
4.5.2	Uppdatering av paketlistan	17
4.5.3	Se till att du har tillräckligt med utrymme för uppgraderingen	17
4.5.4	Uppgradera apt och/eller aptitude först	19
4.5.5	Använda aptitudes lista över automatiskt installerade paket i apt	19
4.5.6	Minimal systemuppgradering	19
4.5.7	Uppgradering av resten av systemet	20
4.5.8	Möjliga problem under uppgraderingen	20
4.6	Uppgradering av kärna och relaterade paket	21
4.6.1	Installera metapaketet för kärnan	21
4.6.2	Ny ordning för enhetsnumrering	21

4.6.3	Tidsproblem vid uppstart	22
4.7	Saker att göra före omstart	22
4.7.1	Kör lilo igen	22
4.8	Systemets uppstart stannar vid Väntar på rotfilssystem	23
4.8.1	Att undvika problemet före uppgradering	23
4.8.2	För att återhämta sig från problemet efter uppgraderingen	25
4.8.2.1	Lösning 1	25
4.8.2.2	Lösning 2	25
4.8.2.3	Lösning 3	26
4.9	Förberedelse inför nästa utgåva	26
4.10	Föråldrade paket	26
4.10.1	Dummy-paket	27
4.11	Inför nästa utgåva av Debian	27
4.11.1	Stöd för ARM ABI-porteringen avslutas till förmån för ARM EABI-porteringen	27
5	Problemområden att känna till för utgåvan lenny	29
5.1	Potentiella problem	29
5.1.1	Problem med enheter på grund av udev	29
5.1.2	Vissa applikationer fungerar inte längre med kärna 2.4	29
5.1.3	Vissa nätverksplatser kan inte anslutas till via TCP	29
5.1.4	Automatisk avstängning slutar fungera	29
5.1.5	Asynkron initiering av nätverksinstallation kan orsaka oförutsett beteende	30
5.1.6	Problem med att använda WPA-säkrade trådlösa nätverk	30
5.1.7	Problem med filnamn med tecken utanför ASCII-tabellen	30
5.1.8	Ljudet slutar fungera	30
5.2	NFS-monteringar hanteras av nfs-common	31
5.3	Förändringar i rumänskt (ro) tangentbordsupplägg	31
5.4	Uppgradera apache2	31
5.5	NIS och nätverkshanteraren	31
5.6	Säkerhetsstatus för Mozillaprodukter	31
5.7	KDE skrivbordet	32
5.8	Stöd och ändringar i Gnome-skrivbordet	32
5.9	Inget stöd för Unicode i emacs21*	32
5.10	slurpd/replica kommer inte längre att fungera	32
5.11	Skrivbordet visar inte helskrämsläge	32
5.12	Problem med DHCP-överlämning vid fel	32
5.13	VServer Disk Limit	33
6	Mer information om Debian GNU/Linux	35
6.1	Ytterligare läsning	35
6.2	Få hjälp	35
6.2.1	Sändlistor	35
6.2.2	Internet Relay Chat	35
6.3	Rapportera fel	35
6.4	Att bidra till Debian	36
A	Hantera ditt etch-system	37
A.1	Uppgradering av ditt etch-system	37
A.2	Kontrollera dina källistor	37
B	Bidrag till Kommentarer till utgåvan	39
C	Till minne av Thiemo Seufer	41
D	Gloslista	43
	Sakregister	45

The Debian Documentation Project (<http://www.debian.org/doc/>)

Kapitel 1

Introduktion

Detta dokument upplyser användarna av Debian GNU/Linux-distributionen om större förändringar i version 5.0 (kodnamn 'lenny').

Dokumentet kommer att förklara hur man på ett säkert sätt uppgraderar från utgåvan 4.0 (kodnamn etch) till den aktuella utgåvan och informerar om kända potentiella problem som kan uppstå i den processen.

Du kan få tag på senaste versionen av detta dokument på <http://www.debian.org/releases/lenny/releasenotes>. Du kan säkerställa att du läser den senaste versionen av dokumentet genom jämföra datumen på första sidan.

OBSERVERA



Observera att det är omöjligt att lista alla kända problem och därför har ett urval gjorts baserat på en kombination av den allmänna förekomsten och problemets inverkan.

Observera att vi endast ger stöd för och dokumenterar uppgraderingen från den tidigare utgåvan av Debian (i det här fallet, uppgradering från 4.0). Om du behöver uppgradera från äldre utgåvor föreslår vi att du läser tidigare versioner av kommentarerna för utgåvan och uppgraderar till 4.0 först.

1.1 Rapportera fel i det här dokumentet

Vi har försökt att testa alla steg i uppgraderingen som beskrivs i det här dokumentet, vi har också försökt förutse alla möjliga problem som kan inträffa för våra användare.

Om du tror att du hittat fel i dokumentationen (felaktig information eller information som saknas) var vänlig och lämna en felrapport via [felrapporteringssystemet](http://bugs.debian.org/) (<http://bugs.debian.org/>), ange `release-notes` som paket. Fel eller idéer rörande själva översättningen av dokumentet ska ställas till debian-l10n-swedish@lists.debian.org.

1.2 Bidra med uppgraderingsrapporter

Vi välkomnar all information från användare som relaterar till uppgraderingar från etch till lenny. Om du vill ge oss information kan du skicka in den genom en felrapport via [felrapporteringssystemet](http://bugs.debian.org/) (<http://bugs.debian.org/>) mot paketet `upgrade-reports` med dina erfarenheter. Vi önskar att du komprimerar eventuella bilagor som inkluderats (med `gzip`).

Inkludera följande information när du skickar in din uppgraderingsrapport:

- Status för din paketdatabas före och efter uppgraderingen: `dpkg`:s statusdatabas tillgänglig i `/var/lib/dpkg/status` och `aptitude`:s pakettillståndsinformation, tillgänglig i `/var/lib/aptitude/pkgstates`. Du bör även ha gjort en säkerhetskopiera innan uppgraderingen, som beskrivs i Avsnitt [4.1.1](#) men du kan även hitta säkerhetskopior av den här informationen i `/var/backups`.

- Sessionsloggar från **script**, läs mer om detta i Avsnitt **4.5.1**.
- Dina apt-loggar, tillgängliga i `/var/log/apt/term.log` eller dina **aptitude**-loggar, tillgängliga i `/var/log/aptitude`.

NOTERA



Du bör ta dig tid att granska och ta bort eventuellt känslig och/eller konfidentiell information från loggfilerna innan de inkluderas i en felrapport eftersom informationen kommer att publiceras i en publik databas.

1.3 Källor för det här dokumentet

Källan till detta dokument är formaterat med DocBook XML. HTML-versionen skapas med `docbook-xsl` och `xsltproc`. PDF-versionen skapas med `dblatex` eller `xmlroff`. Källor för Kommentarer till utgåvan finns tillgängliga i SVN-förrådet för *Debian Documentation Project*. Du kan använda **webbgränssnittet** (<http://svn.debian.org/viewsvn/ddp/manuals/trunk/release-notes/>) för att komma åt dess filer individuellt via webben och se ändringar i dem. För mer information om hur man kommer åt SVN-förrådet läs **SVN-sidorna för Debian Documentation Project** (<http://www.debian.org/doc/cvs>).

Kapitel 2

Vad är nytt i Debian GNU/Linux 5.0

Wikin (<http://wiki.debian.org/NewInLenny>) har närmare information om detta ämne.

Denna utgåva lägger till officiellt stöd för ARM EABI (armel).

Följande arkitekturer stöds officiellt av Debian GNU/Linux lenny:

- Intel x86 ('i386')
- Alpha ('alpha')
- SPARC ('sparc')
- PowerPC ('powerpc')
- ARM ('arm')
- MIPS ('mips' (rak byteordning) och 'mipsel' (omvänd byteordning))
- Intel Itanium ('ia64')
- HP PA-RISC ('hppa')
- S/390 ('s390')
- AMD64 ('amd64')
- ARM EABI ('armel')

Du kan läsa mer om porteringsstatus och porteringsspecifik information för din arkitektur på **Debianians webbsidor för porteringar** (<http://www.debian.org/ports/>).

2.1 Vad är nytt för ARM?

Stöd för Marvells Orionplattform. Debian GNU/Linux 5.0 har stöd för följande enheter baserade på Orionplattformen; QNAP Turbo Station (**TS-109** (<http://www.cyrius.com/debian/orion/qnap/ts-109/>)), **TS-209** (<http://www.cyrius.com/debian/orion/qnap/ts-209/>), **TS-409** (<http://www.cyrius.com/debian/orion/qnap/ts-409/>)), **HP mv2120** (<http://www.cyrius.com/debian/orion/hp/mv2120/>), och **Buffalo Kurobox Pro** (<http://www.cyrius.com/debian/orion/buffalo/kuroboxpro/>).

Stöd för Versatile-plattformen har lagts till via emulering i QEMU.

En ethernet-drivrutin för IXP4xx-plattformen (ex Linksys NSLU2) har nyligen integrerats i standardkärnan, Debiankärnan i lenny använder denna istället för den inofficiella som använts i tidigare utgåvor av Debian.

Den proprietära mikrokoden för IXP4xx som behövs för att använda det inbyggda ethernetgränssnittet finns numera i paketet `ixp4xx-microcode` i arkivet `non-free`. Installationsavbildningar för Debian som innehåller denna mikrokod kommer att finnas tillgänglig via slug-firmware.net (`slug-firmware.net`).

2.2 Vad är nytt i distributionen?

Den nya utgåvan av Debian kommer med ännu fler programvaror än dess föregångare etch; distributionen inkluderar över 7700 nya paket och innehåller totalt 23200 paket. Större delen av programvaran i distributionen har uppdaterats: över 13400 programvarupaket (det är 72% av alla paket i etch). Ett stort antal paket (över 3100, 17 % av paketen i etch) har av olika anledningar tagits bort från distributionen. Du kommer inte att se några uppdateringar för dessa paket och de kommer att markeras som föråldrade¹ i din pakethanterare.

I den här utgåvan av Debian GNU/Linux uppdateras X.Org från 7.1 till 7.3.

Debian GNU/Linux skickar än en gång med flera olika skrivbordsprogram och -miljöer. Bland annat inkluderas skrivbordsmiljön Gnome 2.22¹, KDE 3.5.10, Xfce 4.4.2 och LXDE 0.3.2.1+svn20080509. Produktivitetsprogrammen har även uppgraderats, inklusive kontorssviterna OpenOffice.org 2.4.1 och KOffice 1.6.3 såväl som GNUcash 2.2.6, GNUMeric 1.8.3 och Abiword 2.6.4.

Uppdateringar av andra skrivbordsprogram inkluderar uppgradering till Evolution 2.22.3 och Pidgin 2.4.3 (tidigare under namnet Gaim). Mozilla-sviten har även uppdaterats: iceweasel (version 3.0.6) är webbläsaren Firefox utan varumärkesanknytning och icedove (version 2.0.0.19) är e-postklienten Thunderbird utan varumärkesanknytning.

Den här utgåvan inkluderar även, bland annat, följande programvaruuppdateringar:

Paket	Version i 4.0 (etch)	Version i 5.0 (lenny)
Apache	2.2.3	2.2.9
BIND DNS Server	9.3.4	9.5.0
Cherokee webbserver	0.5.5	0.7.2
Courier MTA	0.53.3	0.60.0
Dia	0.95.0	0.96.1
Ekiga VoIP Client	2.0.3	2.0.12
Exim default email server	4.63	4.69
GNU Compiler Collection som standardkompilator	4.1.1	4.3.2
GIMP	2.2.13	2.4.7
Biblioteket GNU C	2.3.6	2.7
lighttpd	1.4.13	1.4.19
maradns	1.2.12.04	1.3.07.09
MySQL	5.0.32	5.0.51a
OpenLDAP	2.3.30	2.4.11
OpenSSH	4.3	5.1p1
PHP	5.2.0	5.2.6
Postfix MTA	2.3.8	2.5.5
PostgreSQL	8.1.15	8.3.5
Python	2.4.4	2.5.2
Tomcat	5.5.20	5.5.26

Den officiella Debian GNU/Linux-distributionen levereras på 4 till 5 binär-dvd-skivor eller 28 till 32 binär-cd-skivor (beroende på arkitektur) och 4 källkods-dvd-skivor eller 28 källkods-cd-skivor. Dessutom finns en *multiarkitektur* dvd-skiva med delar av arkitekturerna amd64 och i386 och deras källkod. För första gången levereras Debian GNU/Linux som en Blu-ray-avbildning, även denna för arkitekturerna amd64 och i386 med tillhörande källkod.

Debian stödjer numera Linux Standards Base (LSB) version 3.2. Debian 4.0 hade stöd för version 3.1.

2.2.1 Pakethantering

aptitude är det rekommenderade programmet för pakethantering i kommandoradsläge. **Aptitude** stödjer de flesta kommandoradsflaggor från **apt-get** och har visat sig vara bättre på beroendelösning. Om du fortfarande använder **dselect** bör du byta till **aptitude** som gränssnitt för pakethantering.

¹ Med vissa moduler från Gnome 2.20

För lenny har en avancerad konfliktlösningsmekanism implementerats i **aptitude** som kommer att försöka hitta den bästa lösningen om en konflikt uppstår på grund av beroenden mellan paket.

2.2.2 Om proposed-updates (föreslagna förändringar)

Alla förändringar i färdiga utgåvor (stabil samt gammal stabil), genomgår en noggrann testperiod innan förändringen upptas i arkivet. Varje sådan uppdatering av stabil (eller gammal stabil) utgåva kallas för en punktutgåva. Dessa punktutgåvor sker genom mekanismen `proposed-updates`.

Paket kan hamna i `proposed-updates` på två sätt. Säkerhetsuppdateringar till paket via `security.debian.org` hamnar automatiskt i `proposed-updates`. För det andra kan Debian GNU/Linux-utvecklare lägga till paket direkt till `proposed-updates`. Den aktuella listan med paket kan hittas på <http://ftp-master.debian.org/proposed-updates.html> (`http://ftp-master.debian.org/proposed-updates.html`).

Om du vill hjälpa till och prova uppdateringar av paket innan de formellt läggs till i en punktutgåva kan du göra detta genom att lägga till `proposed-updates`-avdelningen i din `sources.list`:

```
deb http://mirrors.kernel.org/debian lenny-proposed-updates main contrib
deb-src http://mirrors.kernel.org/debian lenny-proposed-updates main contrib
```

Nästa gång du kör **aptitude update** kommer systemet att känna av paketen i `proposed-updates`-avdelningen och överväga att installera dessa när den letar efter uppgraderingar.

Detta är inte någon helt ny funktion i Debian men en funktion som tidigare inte talats så högt om.

2.3 Systemförbättringar

Det har skett ett antal ändringar i distributionen som nya installationer av lenny har nytta av men som inte automatiskt kan tillämpas på uppgraderingar från etch. Det här avsnittet ger en överblick för de mest relevanta ändringarna.

SELinux-prioriteten är standard men inte aktiverad som standard Paketet som behövs för SELinux-stödet har höjts till prioriteten *standard*. Det betyder att de kommer att installeras som standard under nya installationer. För befintliga system kan du installera SELinux genom att köra:

```
# aptitude install selinux-basics
```

Observera att SELinux-stödet *inte* är aktiverat som standard. Information om hur man konfigurerar och aktiverar SELinux kan hittas på [Debians wiki](http://wiki.debian.org/SELinux) (`http://wiki.debian.org/SELinux`).

Ny standard syslog-demon Standard `syslog-demon` för lenny är `rsyslog` istället för `syslogd` och `klogd`. `rsyslog` är nästan helt kompatibel med `syslogd` och kan användas utan större besvär. Om du har särskilda loggregler ska du migrera dessa till den nya inställningsfilen `/etc/rsyslog.conf`.

Användare som uppgraderar från etch måste installera `rsyslog` och ta bort `sysklogd` manuellt. Standard `syslog-demon` byts inte ut automatiskt vid uppgradering till lenny.

Bättre stöd för UTF-8 Ett antal ytterligare applikationer kommer att ställas in att använda UTF-8 som standard eller har bättre UTF-8-stöd än tidigare. Läs mer på <http://wiki.debian.org/UTF8BrokenApps> (`http://wiki.debian.org/UTF8BrokenApps`) om applikationer som fortfarande har svårt att hantera UTF-8.

Identification of the release's revision Starting from Lenny, `/etc/debian_version` will indicate the revision number of the debian release (5.0, then 5.0.1, etc.)

This also means that you should not expect this file to be constant through the release lifetime.

[Debians wiki](http://wiki.debian.org/Etch2LennyUpgrade) (`http://wiki.debian.org/Etch2LennyUpgrade`) har ytterligare information om förändringar mellan etch och lenny.

2.4 Större kärnrelaterade ändringar

Debian GNU/Linux 5.0 levereras med kärnversionen 2.6.26 för alla arkitekturer.

Det har skett stora ändringar både i själva kärnan och i paketeringen av kärnan för Debian. Några av dessa ändringar gör uppgraderingsprocessen komplicerad och kan möjligen resultera i problem vid omstart av systemet efter uppgraderingen till lenny. Det här avsnittet ger en överblick för de viktigaste ändringarna, möjliga problem och information om hur man kommer runt dessa problem inkluderas i senare kapitel.

2.4.1 Ändringar i paketering av kärnan

Binary firmware for some drivers moved to non-free Some drivers load binary firmware into the device they are supporting at run time. While this firmware was included in the stock kernel in previous releases, it has now be separately packaged in the non-free section. If you want to continue to use these devices after reboot, make sure the required firmware is present on the installed system. See section 6.4 of the [Installation Manual](http://www.debian.org/releases/stable/installmanual) (<http://www.debian.org/releases/stable/installmanual>) for details.

Nya OpenVZ-kärntyper Debian GNU/Linux 5.0 tillhandahåller kärnavbildningar för OpenVZ en virtualiseringslösning utöver Linux-VServer som introducerades i etch. OpenVZ erbjuder stöd för migrering i körande system till priset av lite extra overhead.

Enhetlighet i kärnpaket för x86 I tidigare utgåvor fanns en speciell kärnvariant (`-k7`) för 32-bitars AMD Athlon/Duron/Sempron-processorer. Denna kärntyp har nu blivit ersatt av en gemensam variant, `-686` som tar hand om alla AMD/Intel/ViA-processorer av 686-variant.

Där det är möjligt har övergångspaket med korrekta beroende för de nya paketen placerats istället för de gamla.

2.5 Emdebian 1.0 (baserad på Debian GNU/Linux lenny 5.0)

Lenny innehåller de verktyg som behövs för att bygga Emdebian, vilket möjliggör för Debian att skapa paket som är nedbantade för att passa inbäddade ARM-system.

Utgåvan Emdebian 1.0 innehåller i sig självt på förhand skapade ARM-paket som räcker för att skapa rotfilssystem som kan anpassas för specifika maskiner och maskintyper. Kärnor och kärnmoduler måste tillhandahållas separat. Stöd för armel och i386 håller på att utvecklas. Läs mer på [webplatsen för Emdebian](http://www.emdebian.org/) (<http://www.emdebian.org/>).

2.6 Netbook-stöd

Netbookdatorer, som Eee PC från Asus, stöds numera av Debian. Läs mer om paketet `eeepc-acpi-scripts` för Eee PC. Dessutom har Debian stöd för Lightweight X11 Desktop Environment, `lxde`, som är särskilt anpassat för netbookdatorer och datorer med låg prestanda.

2.7 Java, nu i Debian

OpenJDK Java Runtime Environment `openjdk-6-jre` och Development Kit `openjdk-6-jdk` som behövs för att köra Java GUI och webbstartprogram eller att bygga sådana program finns nu tillgängliga i Debian. Paketet skapas med stöd för byggsystemet IcedTea och med vissa förändringar från IcedTea-projektet.

Kapitel 3

Installationssystem

Debian Installer är det officiella installationssystemet för Debian. Det tillhandahåller en mängd installationsmetoder, vilka som fungerar på ditt system beror på dess arkitektur.

Avbilder av installeraren för lenny kan hittas tillsammans med installationsguiden på [Debians webbplats](http://www.debian.org/releases/stable/debian-installer/) (<http://www.debian.org/releases/stable/debian-installer/>).

Installationsguiden finns också inkluderad på den första cd-/dvd-skivan av de officiella cd-/dvd-skivorna:

```
/doc/install/manual/åsprk/index.html
```

Du kanske även vill läsa igenom [erratan](http://www.debian.org/releases/stable/debian-installer/index#errata) (<http://www.debian.org/releases/stable/debian-installer/index#errata>) för debian-installer där en lista över kända problem finns.

3.1 Vad är nytt i installationssystemet?

Det har skett en hel del utveckling av Debians Installerare sedan dess första officiella utgivning med Debian GNU/Linux 3.1 (sarge) som resulterat i både förbättrat hårdvarustöd och ett antal nya spännande funktioner.

I dessa Kommentarer till utgåvan kommer vi endast att lista de större ändringarna i installeraren. Om du är intresserad av en översikt över detaljerade ändringar sedan etch se utgåveinformationen för beta- och RC-utgåvorna i [nyhetsarkivet](http://www.debian.org/devel/debian-installer/News/) (<http://www.debian.org/devel/debian-installer/News/>) för Debians Installerare.

3.1.1 Stora ändringar

Stöd för att läsa in inbäddad mjukvara under installationen. Det är nu möjligt att läsa in inbäddad mjukvara i binär form från ett flyttbart media som tillhandahålls utanför Debians installationsmedia.

Stöd för installation från Microsoft Windows Installationsmediet levereras med ett program som gör det möjligt att förbereda systemet för att installera Debian direkt från Microsoft Windows.

Stöd för SATA RAID

tidig uppgradering av paket med säkerhetslagningar. När en fungerande nätverksanslutning tillhandahålls kommer installeraren att vilja uppgradera alla paket som har blivit uppgraderade sedan den första utgåvan av lenny. Denna uppgradering sker i installationssteget, före systemet startas.

Genom detta är det nyligen installerade systemet mindre troligen utsatt för säkerhetshot som har upptäckts och lagats mellan första utgåvan av lenny och installationsmediet.

Stöd för *volatile* (ung, ombytlig) Installeraren kan nu som alternativ ställa in det installerade systemet att uppgradera paket från volatile.debian.org. Arkivet hanterar paket som tillhandahåller data som behöver uppdateras regelbundet, exempelvis definitioner av tidszoner och signaturfiler för antivirus mm.

Nya porteringar Arkitekturen `armel` ingår numera i floran med arkitekturer som Debian fungerar på. Avbildningar för i386 Xen-gäster tillhandahålls också.

Stöd för hårdvarubaserade talsyntesenheter. Ett flertal enheter för hårdvarubaserad talsyntes kan nu användas tillsammans med installeraren, detta förbättrar tillgängligheten för användare med synnedsättning.

Stöd för monteringsalternativet `relatime`. Installeraren kan nu ställa in partitioner med monteringsalternativet `relatime`, detta innebär att tidsstämpeln för åtkomsttid för filer och kataloger bara uppdateras om den tidigare angivna tidsstämpeln var i förfluten tid relativt till den som ska anges med modifieringskommandot.

Klocksynchronisering med NTP vid installationstillfället Datorns klocka synkroniserad med NTP-servrar över nätverket vid installationen så att det installerade systemet omedelbart har korrekt tid angiven.

Nya språk Tack vare en enorm insats av översättare kan Debian nu bli installerad i 63 olika språk (50 via det textbaserade gränssnittet och 13 endast via det grafiska gränssnittet). Fem språk fler än i `etch`, nya språk i den här utgåvan är Amhariska, Marathi, Iriska, Nordsamiska och Serbiska. På grund av otillräcklig uppdatering har ett språk fått tas bort i denna utgåva: Estniska. Ett språk har återaktiverats sedan 4.0 Kymriska (Walesiska).

Språk som bara kan väljas via den grafiska installationsmetoden på grund av att deras tecken inte kan visas i en ickegrafisk miljö är: Amhariska, Bengali, Dzongkha, Gujarati, Hindi, Georgiska, Khmer, Malayalam, Marathi, Nepali, Punjabi, Tamil och Thai.

Förenklat landsval Listan med valbara länder är grupperad efter kontinenter vilket gör det enklare att välja land om användaren inte vill välja ett land som kopplats till ett särskilt språk.

3.1.2 Automatisk installation

Några ändringar som nämns ovan innebär också förändringar i stödet för automatisk installation med förinställda filer. Detta innebär att om du har förinställda filer som fungerat för installeraren för `etch` så kan du inte förvänta dig att dessa fungerar med den nya installeraren utan att redigeras.

Installationsguiden (<http://www.debian.org/releases/stable/installmanual>) har en separat bilaga med omfattande dokumentation om hur förinställningar ska användas.

Kapitel 4

Uppgraderingar från tidigare utgåvor

4.1 Förberedelse inför uppgraderingen

Du bör läsa informationen i Kapitel 5 innan du uppgraderar. Det kapitlet täcker in möjliga problem som inte direkt relaterar till uppgraderingsprocessen men som fortfarande kan vara viktigt att känna till innan du påbörjar arbetet.

4.1.1 Säkerhetskopiera all data eller konfigurationsinformation

Innan uppgradering av ditt system rekommenderas det starkt att du gör en fullständig säkerhetskopia, eller åtminstone en säkerhetskopia av data eller konfigurationsinformation som du inte vill riskera att förlora. Uppgraderingsverktygen och -processen är tillförlitlig men ett hårdvarufel mitt i en uppgradering kan resultera i ett allvarligt skadat system.

De huvudsakliga delar du vill säkerhetskopiera är innehållet i `/etc`, `/var/lib/dpkg`, `/var/lib/aptitude/pkgstates` och utdata från `dpkg --get-selections "*" (citationstecknen är viktiga)`.

Själva uppgraderingsprocessen ändrar ingenting i katalogen `/home`. Dock är det känt att vissa program (exempelvis delar av Mozilla-sviten och skrivbordsmiljöerna GNOME och KDE) skriver över befintliga användarinställningar med nya standardvärden när en ny version av programmet startas för första gången av en användare. Som en försiktighetsåtgärd bör du göra en säkerhetskopia av de dolda filerna och katalogerna (så kallade 'punktfiler') i användarnas hemkataloger. Denna säkerhetskopia kan hjälpa till att återställa eller återskapa de gamla inställningarna. Du kanske även vill informera dina användare om det här.

Alla paketinstallationsåtgärder måste köras med superanvändarens rättigheter, logga in som root, använd **su** eller **sudo** för att få de nödvändiga åtkomsträttigheterna.

Uppgraderingen innebär att vissa förutsättningar måste mötas; du bör kontrollera dem innan den faktiska uppgraderingen påbörjas.

4.1.1.1 Kontrollera att du har korrekt kärna

`glibc`-versionen i lenny fungerar inte på en kärna äldre än 2.6.8 oberoende vilken arkitektur som används. Vissa arkitekturer har till och med högre krav. Vi rekommenderar starkt att du först uppgraderar till en kärna från `etch`, version 2.6.18 eller 2.6.24. Eller en alternativ kärna av minst version 2.6.18 innan du påbörjar uppgraderingsprocessen.

4.1.2 Informera användarna i förväg

Det är klokt att informera alla användare i förväg angående de uppgraderingar som du planerar att göra, även om användarna som kommer åt ditt system via en ssh-anslutning knappast kommer att märka det under uppgraderingen, och bör kunna fortsätta att arbeta som vanligt.

Om du vill vidta extra försiktighetsåtgärder bör du säkerhetskopiera eller avmontera `/home` före uppgradering.

Du kommer antagligen behöva göra en kärnuppgradering vid uppgradering till lenny, så en omstart kommer troligen vara nödvändig. Vanligtvis kommer omstarten att göras efter att uppgraderingen är färdig.

4.1.3 Förbered för återställning

På grund av många ändringar i kärnan mellan etch och lenny gällande drivrutiner, identifiering av hårdvara, namnstandarder och ordning på enhetsfiler, finns det en risk att du kan uppleva problem vid omstart av ditt system efter uppgraderingen. En del kända tänkbara problem finns dokumenterade i det här och de nästkommande kapitlen av Kommentarer till utgåvan.

Av den anledningen är det klokt att försäkra sig om att du kan återställa om ditt system skulle misslyckas att starta om eller, för fjärrhanterade system, misslyckas att komma åt nätverket.

Om du fjärruppgraderar via en `ssh`-länk är det starkt rekommenderat att du vidtar nödvändiga säkerhetsåtgärder för att kunna komma åt servern genom en fjärrserieterminal. Det finns en chans att, efter uppgraderingen av kärnan och omstart, vissa enheter kommer att få nya namn (som beskrivs i Avsnitt 4.6.2) och du kommer att behöva rätta till systemkonfigurationen genom en lokal konsoll. Om systemet av misstag startas om mitt i en uppgradering finns det en chans att du behöver återställa systemet med hjälp av en lokal konsoll.

Det självklara är att först försöka starta om med din gamla kärna. Av olika anledningar, dokumenterade på annan plats i det här dokumentet, är det inte garanterat att det fungerar.

Om det misslyckas behöver du ett alternativt sätt att starta upp ditt system på så att du kan komma åt och reparera det. Ett alternativ är att använda en speciell räddningsavbild eller en cd-skiva med ett körbart Linuxsystem på. Efter att du har startat upp från en sådan skiva bör du kunna montera ditt rotfilssystem och använda `chroot` in i det för att undersöka och rätta till problemet.

Ett annat alternativ som vi rekommenderar är att använda *räddningsläget* i Debian-installeraren för lenny. Fördelen med att använda installeraren är att du kan välja bland dess många installationsmetoder för att hitta en som bäst passar din situation. För mer information, konsultera avsnittet 'Återställning av ett trasigt system' i kapitel 8 av *Installationsguiden* (<http://www.debian.org/releases/stable/installmanual>) och *Debian Installer FAQ* (<http://wiki.debian.org/DebianInstaller/FAQ>).

4.1.3.1 Felsökningsskal under uppstart med hjälp av `initrd`

Paketet `initramfs-tools` inkluderar ett felsökningsskal¹ i de `initrd`-filer som den genererar. Om, till exempel, `initrd`-filen inte kan montera ditt rotfilssystem, kommer du att bli förflyttad in i det här felsökningskalet vilket har grundläggande kommandon tillgängliga för att hjälpa till att spåra problemet och möjligen rätta till det.

Grundläggande saker att kontrollera är: närvaron av korrekta enhetsfiler i `/dev`; vilka moduler som läses in (`cat /proc/modules`); utdata för `dmesg` efter fel vid inläsning av drivrutiner. Utdata för `dmesg` kommer även att visa vilka enhetsfiler som har tilldelats till vilka diskar; du bör kontrollera det här mot utdata för `echo $ROOT` för att försäkra dig om att rotfilssystemet finns på den förväntade enheten.

Om du lyckas rätta till problemet, skriv `exit` för att avsluta felsökningskalet och fortsätta uppstartsprocessen där felet inträffade. Så klart behöver du även rätta till det underliggande problemet och generera om `initrd`-filen så att nästa uppstart inte misslyckas igen.

4.1.4 Förbered en säker miljö för uppgraderingen

Uppgradering av distributionen bör göras antingen lokalt från en virtuell textkonsoll (eller en direktanslutet serieterminal), eller från ett fjärrsystem via en `ssh`-anslutning.

För att öka säkerhetsmarginalen vid en fjärruppgradering föreslår vi att du kör uppgraderingsprocesser i den virtuella konsollen som tillhandahålls av programmet `screen`, vilket gör att man säkert kan återansluta till sessionen och försäkra sig om att uppgraderingsprocessen inte avbryts även om fjärranslutningen avbryts.

¹ Den här funktionen kan inaktiveras genom att lägga till parametern `panic=0` till dina uppstartparametrar.

VIKTIGT

Viktigt! Du bör inte uppdatera via **telnet**, **rlogin**, **rsh** eller från en X-session som hanteras av **xdm**, **gdm** eller **kdm** etc på maskinen som du uppdaterar. Då processer som hanterar dessa tjänster kan avslutas under uppdateringen vilket kan resultera i ett *oåtkomligt* system som endast är halvt uppdaterat.

4.2 Kontrollera systemets status

Uppgraderingsprocessen som beskrivs i detta kapitel har tagits fram med uppdatering från ett 'rent' 4.0-system, utan några tredjeparts paket, i åtanke. För störst tillförlitlighet i uppdateringsprocessen bör du ta bort eventuella tredjepartsprogram från ditt system innan uppdateringen påbörjas.

Processen förutsätter även att ditt system har uppdaterats till den senaste punktutgåvan av 4.0. Om du inte har gjort detta eller är osäker följ instruktionerna i Avsnitt [A.1](#).

4.2.1 Kontrollera kommande åtgärder i pakethanteraren

I vissa fall kan användandet av **apt-get** för installation av paket istället för **aptitude** orsaka att **aptitude** anser att ett paket är 'oanvänt' och markera det för radering. Tillse att ditt system är helt uppdaterat och 'rent' innan du fortsätter med uppdateringen.

På grund av detta bör du kontrollera och det finns några kommande åtgärder i pakethanteraren **aptitude**. Om ett paket är markerat för radering eller uppdatering i pakethanteraren kan det innebära att uppdateringen drabbas negativt. Kom ihåg att detta endast kan åtgärdas om din `sources.list` fortfarande pekar på *etch* och inte på *stable* eller *lenny*, läs mer i Avsnitt [A.2](#).

För att genomföra denna granskning ska du köra **aptitude** i 'visuellt läge' och trycka **g** ('Gå'). Om det indikerar att det finns åtgärder att utföra kontrollera vad det är och lös dem eller kör föreslagen åtgärd. Om inga åtgärder föreslås visas ett meddelande, 'Inga paket är schemalagda för installation, borttagning eller uppdatering'.

4.2.2 Inaktivera APT-nålning

Om du har konfigurerat APT att installera vissa paket från en annan distribution än den stabila (exempelvis från *testing*), kan du ändra din konfiguration för pakethålning i APT (lagrad i `/etc/apt/preferences`) för att tillåta uppdateringen av paket till versionerna i den nya stabila utgåvan. Ytterligare information om APT-nålning kan hittas i `apt_preferences(5)`.

4.2.3 Kontrollera paketstatus

Oavsett vilken metod som används för uppdatering, rekommenderas det att du kontrollerar statusen på paketen först och verifierar att alla paket är möjliga att uppdatera. Följande kommando kommer att visa de paket som har statusen *Half-Installed* eller *Failed-Config*, och de som har någon form av felstatus.

```
# dpkg --audit
```

Du kan även inspektera tillståndet för alla paket på ditt system med **dselect**, **aptitude**, eller med kommandon som

```
# dpkg -l | pager
```

eller

```
# dpkg --get-selections "*" > ~/curr-pkgs.txt
```

Det är önskvärt att ta bort eventuella tillbakahållna paket innan uppdatering. Om något paket är systemkritiskt och hålls tillbaka för uppdateringen, kommer uppdateringen att misslyckas.

Observera att **aptitude** använder en annan metod för att registrera paket som hålls tillbaka än **apt-get** och **dselect**. Du kan identifiera paket som hålls tillbaka med **aptitude** med

```
# aptitude search "~ahold" | grep "^h"
```

Om du vill kontrollera vilka paket som hålls tillbaka vid användning av **apt-get**, ska du använda

```
# dpkg --get-selections | grep hold
```

Om du ändrat och byggt om ett paket lokalt, och inte bytte namn på det eller la in ett datum i versionen, måste du hålla tillbaka det för att förhindra att det uppgraderas.

Pakettillståndet 'hold' för **aptitude** kan ändras med:

```
# aptitude hold package_name
```

Ersätt `hold` med `unhold` för att ändra 'hold'-tillståndet.

Om det är någonting du behöver rätta till är det bäst att se till att din `sources.list` fortfarande refererar till etch vilket förklaras i Avsnitt [A.2](#).

4.2.4 Avsnittet `proposed-updates`

Om du har `proposed-updates` i din `/etc/apt/sources.list` ska du ta bort det innan du försöker uppdatera ditt system. Detta är en försiktighetsåtgärd för att minska risken att konflikter uppstår.

4.2.5 Inofficiella källor och bakåtporteringar

Om du har några icke-Debianpaket på ditt system, bör du tänka på att dessa kan tas bort under uppgraderingen på grund av beroendekonflikter. Om dessa paket blev installerade genom att lägga till extra paketarkiv i din `/etc/apt/sources.list`, bör du kontrollera om det arkivet även erbjuder paket som är byggda för lenny och ändra källraden på lämpligt sätt samtidigt som dina källrader för Debian-paket.

Vissa användare kan ha inofficiella bakåtporterade versioner av paket, som är 'nyare' än de som finns i Debian, installerade på sina etch-system. Sådana paket kommer med stor sannolikhet att orsaka problem under en uppgradering eftersom de kan resultera i filkonflikter². Avsnitt [4.5.8](#) innehåller information om hur man hanterar filkonflikter om de skulle inträffa.

4.2.5.1 Att använda paket från `backports.org`

`backports.org` är ett semiofficiell förråd som tillhandahålls av Debian GNU/Linux-utvecklare. Förrådet innehåller nyare paket för den stabila utgåvan, dessa paket är baserade på de som finns i 'testing'-arkivet.

Paketen i `backports.org`-förrådet kommer från 'testing' med versionsnummer som justerats för att bevara uppgraderingsgången från etch backports till lenny. Det finns dock några backports-paket som hämtas från paket som finns i unstable-arkivet nämligen säkerhetsuppdateringar och följande undantag: Firefox, Linuxkärnan, OpenOffice.org samt X.Org.

If you do not use one of these exceptions, you can safely upgrade to lenny. If you use one of these exceptions, set the `Pin-Priority` (see `apt_preferences(5)`) temporarily to 1001 for all packages from lenny, and you should be able to do a safe dist-upgrade too.

4.3 Avmarkera paket manuellt

För att förhindra att **aptitude** tar bort vissa paket som installerades som beroenden för andra paket behöver du manuellt avmarkera dem som *auto*-paket. Det inkluderar OpenOffice och Vim för skrivbordsinstallationer:

```
# aptitude unmarkauto openoffice.org vim
```

Och 2.6-kärnavbilder om du har installerat dem med ett kärnmetapakets:

```
# aptitude unmarkauto $(dpkg-query -W 'linux-image-2.6.*' | cut -f1)
```

² Debians pakethanteringssystem tillåter vanligtvis inte att ett paket tar bort eller ersätta en fil som ägs av ett annat paket såvida det inte har definierats att ersätta det paketet.

NOTERA



Observera: Du kan granska vilka paket som är markerade som *auto* i aptitude genom att köra:

```
# aptitude search '~i~M'
```

4.4 Förbered källor för APT

Innan du påbörjar uppgraderingen måste du redigera konfigurationsfilen för paketlistor i `apt`, `/etc/apt/sources.list`.

Apt kommer att överväga att alla paket som kan hittas via någon 'deb'-rad, och installera paketet med högsta versionsnumret, där prioritet ges till de förstnämnda raderna (om du nyttjar flera redundanta speglar, skulle du vanligtvis först namnge en lokal hårddisk, sedan cd-skivor, och sedan HTTP/FTP-speglar).

TIPS



Du kan behöva lägga till ett undantag från GPG-kontrollen för dvd- och cd-rom-avbildningarna. Lägg till följande rad i `/etc/apt/apt.conf` om det inte redan finns i `/etc/apt/apt.conf.d/00trustcdrom`:

```
APT::Authentication::TrustCDROM "true";
```

Detta fungerar dock inte med dvd-/cd-rom-avbildningar.

En utgåva kan ofta refereras till både dess kodnamn (t.ex. `etch`, `lenny`) och efter dess statusnamn (alltså `oldstable`, `stable`, `testing`, `unstable`). Att referera till en utgåva efter dess kodnamn har fördelen att du aldrig blir överraskad av en ny utgåva och av den anledningen används den här metoden här. Det kan naturligtvis betyda att du själv måste hålla utkik efter nya utgåvor. Om du istället använder statusnamnet kommer systemet automatiskt att uppgraderas utan förvarning genom att uppdatera en mängd paket så snart en utgivning har skett.

4.4.1 Lägg till APT-källor på Internet

Standardkonfigurationen är inställd för installation från Debians huvudservrar på Internet, men du kanske önskar ändra `/etc/apt/sources.list` till att använda andra speglar, föredragsvis en spegel som är nätverksmässigt närmare dig.

Adresserna till Debians HTTP- eller FTP-speglar kan hittas på <http://www.debian.org/distrib/ftplist> (se avsnittet 'Listan över Debianspeglingar'). HTTP-speglar är vanligtvis snabbare än FTP-speglar.

Till exempel, anta att din närmaste Debian-spegel är `http://mirrors.kernel.org`. När den spegeln inspekteras med en webbläsare eller FTP-program, kommer du att märka att huvudkatalogerna är organiserade så här:

```
http://mirrors.kernel.org/debian/dists/lenny/main/binary-armel/...
http://mirrors.kernel.org/debian/dists/lenny/contrib/binary-armel/...
```

Lägg till den här raden till din `sources.list` för att använda den här spegelservern med apt:

```
deb http://mirrors.kernel.org/debian lenny main contrib
```

Observera att 'dists' läggs till automatiskt och argumenten efter utgåvans namn används för att utöka sökvägen till flera kataloger.

Efter att du har lagt till dina nya källor ska du inaktivera de tidigare befintliga 'deb'-raderna i `sources.list` genom att placera ett hash-tecken (#) framför dem.

4.4.2 Lägg till APT-källor för en lokal spegelservare

Istället för att använda HTTP- eller FTP-paketspeglar, kanske du önskar ändra `/etc/apt/sources.list` till att använda en spegel på en lokal hårddisk (möjligen monterad över NFS).

Till exempel, din paketspegel kan finnas under `/var/ftp/debian/` och innehåller huvudkataloger som dessa:

```
/var/ftp/debian/dists/lenny/main/binary-armel/...
/var/ftp/debian/dists/lenny/contrib/binary-armel/...
```

Lägg till den här raden till din `sources.list` för att använda den här med apt:

```
deb file:/var/ftp/debian lenny main contrib
```

Observera att 'dists' läggs till automatiskt och argumenten efter utgåvans namn används för att utöka sökvägen till flera kataloger.

Efter att du har lagt till dina nya källor ska du inaktivera de tidigare befintliga 'deb'-raderna i `sources.list` genom att placera ett hash-tecken (#) framför dem.

4.4.3 Lägg till APT-källa från cd-rom eller dvd

Om du *endast* vill använda cd-skivor, kommentera ut de befintliga 'deb'-raderna i `/etc/apt/sources.list` genom att placera ett hash-tecken (#) framför dem.

Se till att det finns en rad i `/etc/fstab` som aktiverar montering av din cd-rom-enhet på monteringspunkten `/cdrom` (den exakta monteringspunkten `/cdrom` krävs för apt-cdrom). Till exempel, om `/dev/hdc` är din cd-rom-enhet, ska `/etc/fstab` innehålla en rad som denna:

```
/dev/hdc /cdrom auto defaults,noauto,ro 0 0
```

Observera att det *inte får finnas några blanksteg* mellan orden `defaults, noauto, ro` i det fjärde fältet.

För att verifiera att det fungerar, mata in en cd och försök köra

```
# mount /cdrom      # det ähr monterar cd-skivan åp monteringspunkten
# ls -alF /cdrom    # det åhr ska visa cd-skivans rotkatalog
# umount /cdrom     # det åhr kommer att avmontera cd-skivan
```

Kör sedan:

```
# apt-cdrom add
```

för varje Debian cd-rom med binärer som du har tillgång till för att lägga till data om varje cd till APT:s databas.

4.5 Uppgradering av paket

Det rekommenderade sättet att uppdatera mellan utgåvor av Debian GNU/Linux är att använda pakethanteringsverktyget **aptitude**. Det här programmet gör säkrare val angående paketinstallationer än att köra **apt-get** direkt.

Glöm inte att montera alla nödvändiga partitioner (speciellt rot- och `/usr`-partitionerna) läs- och skrivbara, med ett kommando som det här:

```
# mount -o remount,rw /mountpoint
```

Efter det ska du kontrollera att källraderna för APT (i `/etc/apt/sources.list`) refererar antingen till 'lenny' eller till 'stable'. Det ska inte finnas några källrader som pekar till etch.

NOTERA



Observera: källrader för en cd-skiva kommer ofta att referera till 'unstable', även om det här är konstigt ska du *inte* ändra dem.

4.5.1 Spela in sessionen

Det rekommenderas starkt att du använder programmet `/usr/bin/script` för att spela in en utskrift av uppgraderingssessionen. Om problem uppstår har du en logg på vad som hände och, om det behövs, kan tillhandahålla exakt information i en felrapport. För att påbörja inspelningen, kör:

```
# script -t 2>>/upgrade-lenny.time -a ~/upgrade-lenny.script
```

eller liknande. Lägg inte typescript-filen i en temporär katalog såsom `/tmp` eller `/var/tmp` (filer i dessa kataloger kan tas bort under uppgraderingen eller under en omstart).

Typescript kommer även att låta dig granska informationen som har rullat ut från skärmen. Växla helt enkelt till VT2 (med `Alt+F2`) och, efter inloggning, använd `less -R ~root/uppgradering-till-lenny.script` för att visa filen.

Efter att du har färdigställt uppgraderingen, kan du stoppa `script` genom att ange `exit` vid prompten.

Om du har använt flaggan `-t` för `script` kan du använda programmet `scriptreplay` för att spela upp hela sessionen:

```
# scriptreplay ~/upgrade-lenny.time ~/upgrade-lenny.script
```

4.5.2 Uppdatering av paketlistan

Först behöver listan över tillgängliga paket för den nya utgåvan hämtas. Det görs genom att köra:

```
# aptitude update
```

Körning av det här första gången nya källor uppdateras kommer att skriva ut några varningar som relaterar till tillgängligheten för källorna. Dessa varningar är harmlösa och kommer inte att visas om du kör kommandot igen.

4.5.3 Se till att du har tillräckligt med utrymme för uppgraderingen

Du måste kontrollera att ditt system har tillräckligt mycket ledigt hårddiskutrymme innan du påbörjar en fullständig systemuppgradering, som beskrivs i Avsnitt 4.5.7. Alla paket som behöver hämtas för installation kommer att hämtas från nätverket och lagras i `/var/cache/apt/archives` (och underkatalogen `partial/` under hämtningen) så du måste se till att du har tillräckligt utrymme på filsystemspartitionen som innehåller `/var/` för temporär hämtning av paketen som ska installeras på ditt system. Efter hämtningen kommer du antagligen behöva mer utrymme på de andra filsystemspartitionerna för att både installera de uppgraderade paketen (som kan innehålla större binärfiler eller mer data) och de nya paketen som kommer att inkluderas i uppgraderingen. Om ditt system inte har tillräckligt med utrymme kan det resultera i en ofullständig uppgradering som kan vara svår att rätta till.

Både `aptitude` och `apt` kommer att visa dig detaljerad information om det diskutrymme som behövs för installationen. Du kan se det här estimatet innan den faktiska uppgraderingen påbörjas genom att köra:

```
# aptitude -y -s -f --with-recommends dist-upgrade
[ ... ]
XXX uppgraderade, XXX nyinstallerade, XXX att ta bort och XXX inte uppgraderade.ö
Behver ähmta xx.xMB/yyyMB arkiv. Efter uppäckning kommer AAAMB diskplats att ←
  äänvndas.
Kommer att ähmta/installera/ta bort paket.
```

NOTERA



Körning av det här kommandot i början av uppgraderingsprocessen kan ge felaktigheter. Anledningarna beskrivs i nästkommande avsnitt. I det fallet behöver du vänta tills du har gjort en minimal systemuppgradering enligt Avsnitt 4.5.6 och uppgraderat din kärna enligt Avsnitt 4.1.1.1 innan du kör det här kommandot för att uppskatta diskutrymme.

Om du inte har tillräckligt med utrymme kan du se till att frigöra utrymme innan uppgraderingen. Du kan till exempel:

- Ta bort paket som tidigare har hämtats ner för installation (i `/var/cache/apt/archive`). Rensa upp paketcachen genom att köra **apt-get clean** eller **aptitude clean** vilket kommer att ta bort alla tidigare hämtade paketfiler.
- Ta bort gamla paket som du inte längre använder. Om du har paketet `popularity-contest` installerat kan du använda **popcon-largest-unused** för att lista de paket som du inte använder i systemet och som tar upp mest utrymme. Du kan även använda **deborphan** eller **debfooster** för att hitta föråldrade paket (se Avsnitt 4.10). Alternativt kan du starta **aptitude** i 'visuellt läge' och hitta föråldrade paket under 'Föråldrade och lokalt skapade paket'.
- Ta bort paket som tar upp för mycket utrymme och som du inte har ett omedelbart behov av (du kan alltid installera om dem efter uppgraderingen). Du kan lista paketen som tar upp mest utrymme genom att köra **dpigs** (tillgängligt i paketet `debian-goodies`) eller med **wajig** (kör `wajig size`).

You can list packages that take up most of the disk space with `aptitude`. Start **aptitude** into 'visual mode', select Views → New Flat Package List (this menu entry is available only after `etch` version), press **I** and enter `~i`, press **S** and enter `~installsize`, then it will give you nice list to work with. Doing this after upgrading `aptitude` should give you access to this new feature.

- Ta bort översättningar och lokalanpassade filer för systemet om de inte behövs. Du kan installera paketet `localepurge` och ställa in det så att endast de lokalanpassningar som du vill ha sparas på systemet. Detta kommer att minska mängden hårddiskutrymme som används i `/usr/share/locale`.
- Flytta systemloggar från `/var/log/` till ett annat system, eller ta bort permanent.
- Använd en temporär `/var/cache/apt/archives`: Du kan använda en temporär cachekatalog på ett annat filsystem (USB-disk, temporär hårddisk, filsystem som redan används, ...)

NOTERA



Använd inte en NFS-montering eftersom nätverksanslutningen kan avbrytas under uppgraderingen.

Till exempel, om du har en USB-disk monterad på `/media/usbkey`:

1. ta bort paket som tidigare hämtats för installation:

```
# apt-get clean
```

2. kopiera katalogen `/var/cache/apt/archives` till USB-diskheten:

```
# cp -ax /var/cache/apt/archives /media/usbkey/
```

3. montera den temporära cachekatalogen ovanpå den nuvarande:

```
# mount --bind /media/usbkey/archives /var/cache/apt/archives
```

4. efter uppgraderingen återställer du original katalogen `/var/cache/apt/archives`:

```
# umount /media/usbkey/archives
```

5. radera det som lämnats kvar i `/media/usbkey/archives`.

Du kan skapa den temporära cachekatalogen på vilket filsystem som helst som helst som finns monterat på ditt system.

Observera att för att ta bort paket på ett säkert sätt, rekommenderas det att växla tillbaka din `sources.list` till etch vilket förklaras i Avsnitt [A.2](#).

4.5.4 Uppgradera apt och/eller aptitude först

Många felrapporter har indikerat att paketen `aptitude` och `apt` i `etch` ofta misslyckas med att hantera uppgraderingen till `lenny`. `apt` i `lenny` är bättre på att hantera komplexa kedjor av paket som omedelbart behöver ställas in och `aptitude` hittar smartare lösningar för att uppfylla beroenden. I och med att dessa funktioner används oerhört mycket vid `dist-upgrade` till `lenny` måste dessa paket först uppgraderas före allting annat. För `apt` kör:

```
# apt-get install apt
```

och för `aptitude` (om du har det installerat) kör:

```
# aptitude install aptitude
```

Detta steg kommer automatiskt att uppgradera `libc6` och `locales` samt lägga till stödbibliotek för SELinux (`libselinux1`). Därefter kommer några tjänster att startas om, ex. `xdm`, `gdm` och `kdm` vilket medför att lokala X11-sessioner kommer att avslutas.

4.5.5 Använda aptitudes lista över automatiskt installerade paket i apt

`aptitude` underhåller en lista med paket som installerats automatiskt (exempelvis som beroenden av ett annat paket). I `lenny` finns denna funktion även i `apt`.

När `lenny`-versionen av `aptitude` körs kommer den att läsa in listan med automatiskt installerade paket och konvertera den så att den kan användas tillsammans med `lenny`-versionen av `apt`. Om du har `aptitude` installerat ska du åtminstone köra ett av **aptitudes** kommandon för att genomföra konverteringen av listan, exempelvis kan du söka efter ett paket som inte finns.

```
# aptitude search "?false"
```

4.5.6 Minimal systemuppgradering

På grund av vissa nödvändiga paketkonflikter mellan `etch` och `lenny` kommer en direkt körning av `aptitude dist-upgrade` ofta att ta bort ett stort antal paket som du vill behålla. Vi rekommenderar därför en tvådelad uppgraderingsprocess, först en minimal uppgradering för att komma runt dessa konflikter och sedan en fullständig `dist-upgrade`.

Kör först:

```
# aptitude safe-upgrade
```

Det här innebär att endast de paket som kan uppgraderas utan att kräva att några andra paket tas bort eller installeras uppgraderas.

Nästa steg beror på uppsättningen paket som du har installerade. Dessa Kommentarer till utgåvan ger allmänna råd om vilken metod som ska användas, men om du är osäker är det rekommenderat att du undersöker de paketborttagningar som föreslås av varje metod innan du fortsätter.

Några vanliga paket som förväntas bli borttagna är bland annat `base-config`, `hotplug`, `xlibs`, `netkit-inetd`, `python2.3`, `xfree86-common` och `xserver-common`. För en mer komplett lista över föråldrade paket i `lenny`, se Avsnitt [4.10](#).

4.5.7 Uppgradering av resten av systemet

Du är nu redo att fortsätta med huvuddelen av uppgraderingen. Kör:

```
# aptitude dist-upgrade
```

Det här kommer att genomföra en fullständig uppgradering av systemet, alltså installera de senaste tillgängliga versionerna av samtliga paket och lösa alla tänkbara beroendeändringar mellan paketen i olika utgåvor. Om det är nödvändigt kommer det även att installera några nya paket (vanligtvis nya versioner av bibliotek eller paket som fått nya namn) samt ta bort eventuella föråldrade paket som står i konflikt med varandra.

Vid uppgradering från en uppsättning cd-skivor (eller dvd-skivor), kommer du att bli uppmanad att mata in specifika cd-skivor vid olika tillfällen under uppgraderingen. Du kanske måste mata in samma cd-skiva flera gånger; detta beror på att sammankopplade paket har blivit utspridda över cd-skivorna.

Nya versioner av installerade paket, som inte kan uppgraderas utan att ändra installationsstatus för ett annat paket, kommer att lämnas kvar vid deras nuvarande version (visas som 'återhållna'). Det kan lösas genom att antingen använda **aptitude** för att välja dessa paket för installation eller genom att prova `aptitude -f install paket`.

4.5.8 Möjliga problem under uppgraderingen

Om en åtgärd med **aptitude**, **apt-get** eller **dpkg** misslyckas med felet

```
F: Dynamisk MMap fick slut på utrymme
E: Dynamic MMap ran out of room
```

räcker inte standardutrymmet för cachen till. Du kan lösa det här genom att antingen ta bort eller kommentera rader som du inte behöver i `/etc/apt/sources.list` eller genom att öka storleken på cachen. Storleken på cachen kan ökas genom att ställa in `APT::Cache-Limit` i `/etc/apt/apt.conf`. Följande kommando kommer att ställa in den till ett värde som ska vara tillräckligt för uppgraderingen:

```
# echo 'APT::Cache-Limit "12500000";' >> /etc/apt/apt.conf
```

Det här förutsätter att du inte redan har ställt in variabeln i den filen.

Ibland är det nödvändigt att aktivera alternativet `APT::Force-LoopBreak` i `APT` för att temporärt ta bort ett systemkritiskt paket på grund av en Konflikt/Förberoende-slinga. **Aptitude** kommer att varna dig om det här och avbryta uppgraderingen. Du kan lösa det genom att ange alternativet `-o APT::Force-LoopBreak=1` på kommandoraden för **aptitude**.

Det är möjligt att beroendestrukturen för ett system kan vara så skadat att det kräver handpåläggning. Vanligtvis innebär det att använda **aptitude** eller

```
# dpkg --remove package_name
```

för att plocka bort några av de störande paketen, eller

```
# aptitude -f install
# dpkg --configure --pending
```

I extrema fall kan du behöva tvinga fram en ominstallation med ett kommando som detta

```
# dpkg --install /öaskvg/till/paketnamn.deb
```

Filkonflikter bör inte inträffa om du uppgraderar från ett 'rent' etch-system, men kan inträffa om du har inofficiella bakåtporteringar installerade. En filkonflikt resulterar i ett fel som:

```
Packar upp <paket-foo> (åfrn <paket-fil-foo>) ...
dpkg: fel vid hantering av <paket-foo> (--install):öö
  frsker skriva över "<ångot-fil-namn">,
  som åocks finns i paketet <paket-bar>
dpkg-deb: underprocessen paste öddad av signal (Brutet örr)
Fel uppstod vid hantering:
<paket-foo>
```

Du kan försöka lösa en filkonflikt genom att tvinga igenom borttagning av paketet som nämns på *sista* raden i felmeddelandet:

```
# dpkg -r --force-depends paketnamn
```

Efter att problemen har lösts, bör du kunna återuppta uppgraderingen genom att upprepa tidigare beskrivna **aptitude**-kommandon.

Under uppgraderingen kommer det att ställas frågor om konfiguration eller omkonfigurationen av flera paket. När du blir tillfrågad om någon fil i katalogerna `/etc/init.d` eller `/etc/terminfo`, eller filen `/etc/manpath.config` ska ersättas av paketansvariges version, är det oftast nödvändigt att svara 'ja' för att upprätthålla systemets tillstånd. Du kan alltid återgå till de gamla versionerna, eftersom de kommer att sparas med en `.dpkg-old`-ändelse.

Om du inte är säker på vad som behöver göras, skriv ner namnet på paketet eller filen och red ut saker och ting senare. Du kan söka i typescript-filen för att granska informationen som visades på skärmen under uppgraderingen.

4.6 Uppgradering av kärna och relaterade paket

Det här avsnittet förklarar hur man uppgraderar sin kärna och identifierar tänkbara problem relaterade till den här uppgraderingen. Du kan antingen installera ett av paketen `linux-image-*` som tillhandahålls av Debian, eller bygga en anpassad kärna från källkod.

Observera att en hel del information i det här avsnittet är baserad på antagelsen att du kommer att använda en av de modulära Debian-kärnorna tillsammans med `initramfs-tools` och `udev`. Om du har valt att använda en anpassad kärna som inte kräver en `initrd` eller om du använder en annan `initrd`-generator kan delar av den här informationen vara irrelevant för dig.

4.6.1 Installera metapaketet för kärnan

När du kör `dist-upgrade` från `etch` till `lenny`, rekommenderas det starkt att du installerar ett nytt `linux-image-2.6-*`-metapaket. Det här paketet kan installeras automatiskt av `dist-upgrade`-processen. Du kan verifiera det genom att köra:

```
# dpkg -l "linux-image*" | grep ^ii
```

Om du inte ser något utdata, behöver du installera ett nytt `linux-image`-paket för hand. Kör följande kommando för att se en lista över tillgängliga `linux-image-2.6`-metapaket:

```
# apt-cache search linux-image-2.6- | grep -v transition
```

Om du är osäker på vilket paket du ska välja, kör `uname -r` och leta efter ett paket med liknande namn. Om du till exempel ser `'2.6.18-6-686'` rekommenderas det att du installerar `linux-image-2.6-686` (observera att varianten `k7` inte längre finns, om du för tillfället kör en kärna av `k7`-variant bör du installera varianten `686` istället.). Du kan även använda **apt-cache** för att se en lång beskrivning för varje paket för att hjälpa dig att välja den bästa möjliga. Till exempel:

```
# apt-cache show linux-image-2.6-686
```

Du bör sedan använda `aptitude install` för att installera den. När den här nya kärnan har installerats bör du starta om vid nästa möjliga tillfälle för att dra nytta av den nya kärnversionen.

För den mer äventyrlige finns det ett enkelt sätt att bygga sin egna, anpassade, kärna på Debian GNU/Linux. Installera verktyget `kernel-package` och läs dokumentationen i `/usr/share/doc/kernel-package`.

Om möjligt är det till din fördel att uppgradera kärnpaketet separat från själva `dist-upgrade` för att minska chanserna för ett temporärt icke-startbart system. Observera att det här bör endast göras efter den minimala uppgraderingsprocessen, beskriven i Avsnitt [4.5.6](#).

4.6.2 Ny ordning för enhetsnumrering

`lenny` har en mer robust mekanism för att identifiera hårdvara än tidigare utgåvor. Dock kan den här orsaka ändringar på den ordning som enheter identifieras på ditt system vilket påverkar ordningen i vilken enhetsnamnen tilldelas. Om du till exempel har två nätverkskort som associeras med två olika drivrutiner, kan de enheter som `eth0` och `eth1` refererar till ändras. Observera att den nya mekanismen

betyder att om du till exempel byter Ethernet-kort på ett körande lenny-system, kommer det nya kortet även att få ett nytt gränssnittsnamn.

För nätverksenheter kan du undvika den här omnumreringen genom att använda regler i `udev`, mer specifikt genom definitionerna i `/etc/udev/rules.d/70_persistent-net.rules`³. Alternativt kan du använda verktyget `ifrename` för att binda fysiska enheter till specifika namn vid uppstarten. Se `ifrename(8)` och `iftab(5)` för mer information. De två alternativen (`udev` och `ifrename`) bör inte användas samtidigt.

För lagringsenheter kan du undvika den här omsorteringen genom att använda `initramfs-tools` och konfigurera det att läsa in drivrutinsmoduler för lagringsenheter i samma ordning som de för närvarande läses in i. För att göra detta, identifiera ordningen på lagringsmodulerna på ditt system genom att se på utdatat från `lsmod`. `lsmod` listar moduler i omvänd ordning än den de lästes in i, alltså den första modulen i listan är den som sist lästes in. Observera att det här endast fungerar för enheter som kärnan numrerar i en stabil ordning (som PCI-enheter).

Dock kommer borttagning och omläsning av moduler efter initial uppstart att påverka ordningen. Din kärna kan även ha några drivrutiner som är statiskt länkade, och dessa namn kommer inte att visas i utdatat från `lsmod`. Du kan möjligen läsa ut dessa drivrutinsnamn och inläsningsordning genom att se i `/var/log/kern.log`, eller utdatat från `dmesg`.

Lägg till dessa modulnamn till `/etc/initramfs-tools/modules` i den ordning som de ska läsas in i vid uppstarten. Vissa modulnamn kan ha ändrats mellan `etch` och `lenny`. Till exempel har `sym53c8xx_2` blivit `sym53c8xx`.

Du kommer att behöva generera om dina `initramfs`-avbild(er) genom att köra `update-initramfs -u -k all`.

När du väl kör en lenny-kärna tillsammans med `udev` kan du konfigurera om ditt system till att komma åt diskarna genom ett alias som inte är beroende av inläsningsordningen. Dessa alias finns under `/dev/disk/-hierarkin`.

4.6.3 Tidsproblem vid uppstart

Om en `initrd` som skapats med `initramfs-tools` används för att starta upp systemet, kan i vissa fall skapandet av enhetsfiler av `udev` ske för sent för uppstartsskripten att agera på.

De vanligaste symptomen är att uppstarten misslyckas på grund av att rotfilsystemet inte kan monteras och du hamnar i ett felsökningsskal, men när du kontrollerar senare så finns alla nödvändiga enheter i `/dev`. Det har observerats i fall där rotfilsystemet finns på en USB-disk eller på RAID, speciellt om LILO används.

Ett sätt att komma runt det här problemet på är att använda uppstartsparametern `rootdelay=9`. Värdet för tidsgränsen (i sekunder) kan behöva justeras.

4.7 Saker att göra före omstart

När `aptitude dist-upgrade` har kört färdigt är den 'formella' uppgraderingen färdig, men det finns vissa andra saker som bör tas om hand före nästa omstart.

4.7.1 Kör lilo igen

Om du använder `lilo` som uppstartshanterare (det är standardvalet för uppstartshantering i vissa installationer av `etch`) rekommenderas det starkt att du åter kör `lilo` efter uppdateringarna.

```
# /sbin/lilo
```

Observer att dett är ett nödvändigt steg även om du inte har uppdaterat din kärna i och med att andra steget av `lilo` har förändrats i och med uppdateringen av paket.

Kontrollera även innehållet i din `/etc/kernel-img.conf` och säkerställ att du har `do_bootloader = Yes` i den. På det viset kommer uppstartshanteraren alltid att köras efter en uppgradering av kärnan.

Skulle några problem uppstå när du kör `lilo` kontrollera de symboliska länkarna i `/` till `vmlinuz` och `initrd` samt att innehållet i `/etc/lilo.conf` inte avviker från detta.

³ Reglerna där genereras automatiskt av skriptet `/etc/udev/rules.d/75_persistent-net-generator.rules` för att få statiska namn för nätverksgränssnitten. Ta bort den här symboliska länken för att inaktivera den statiska enhetsnamngivningen för nätverkskort genom `udev`.

Om du glömmer att köra **lilo** innan omstart av systemet eller om systemet startas om av misstag innan du gjort detta manuellt kan det innebära att systemet inte kan startas normalt. Isättlet för den vanliga lilo-bilden kommer endast LI visas när systemet ska startas⁴. Se även Avsnitt 4.1.3 för mer information om att återhämta sig från detta.

4.8 Systemets uppstart stannar vid Väntar på rotfilssystem

Procedur för att återhämta sig från att /dev/hda blev /dev/sda Några användare har rapporterat att en uppgradering kan innebära att kärnan inte hittar systemets rotpartition efter att systemet startats om.

I sådana fall kommer systemet att stanna med följande meddelande:

```
ä
Vntar på rotfilssystem... (engelska: Waiting for root file system ...)
```

och efter några sekunder visas ett enkelt inmatningsställe för busybox.

Detta problem uppstår när kärnan introducerar en ny typ av namnskapande för IDE-enheter. IDE-enheternas namnkonvention var tidigare hda, hdb, hdc och hdd. Det nya sättet kommer att ge samma diskar följande namn sda, sdb, sdc och sdd. Problemet härstammar från att uppgraderingen inte skapar en ny /boot/grub/menu.lst som tar de nya namnen i beaktan. Vid uppstarten kommer Grub att skicka en rotpartition till kärnan som kärnan inte kan hitta.

Har du drabbats av detta problem efter uppgraderingen, läs i Avsnitt 4.8.2. För att undvika detta problem före uppgradering läs vidare nedan.

4.8.1 Att undvika problemet före uppgradering

Problemet kan undvikas helt genom att använda en identitet för rotfilssystemet som inte förändras mellan olika uppstarter. Det finns två möjliga metoder för att uppnå detta. Att sätta etiketter på filsystem eller genom att använda ett filsystems universella unika identitet (UUID). Dessa metoder har funnits i Debian sedan utgåvan 'etch'.

Metoderna har fördelar och nackdelar. Etikett-metoden är mer läsbar men det kan finnas problem med den om ett annat filsystem på din maskin har samma etikett. UUID-metoden är fulare men att det skulle skapas två filsystem med samma UUID är mycket osannolikt.

I exemplet nedan antas att rotfilssystemet finns på /dev/hda6. Vi antar dessutom att systemet har en fungerande udev-installation och ett ext2- eller ext3-filsystem.

För att använda etikettmetoden:

1. Sätt etikett på filsystemet (namnet måste vara < 16 tecken) genom att köra följande kommando:
e2label /dev/hda6 rootfilesys
2. Redigera /boot/grub/menu.lst och ändra följande rad:

```
# kopt=root=/dev/hda6 ro
```

till

```
# kopt=root=LABEL=rootfilesys ro
```

NOTERA



Ta inte bort # i början på raden, den måste vara där.

3. Uppdatera kernel-raderna i menu.lst genom att köra kommandot **update-grub**.
4. Redigera /etc/fstab och ändra raden som monterar /-partition, ex:

⁴ För mer information om lilos felkoder i uppstarten se [The Linux Bootdisk HOWTO](http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html) (<http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html>).

```
/dev/hda6 / ext3 defaults,errors=remount-ro 0 1
```

till

```
LABEL=rootfilesystem / ext3 defaults,errors=remount-ro 0 1
```

Förändringen som spelar roll finns i första kolumnen, du behöver inte ändra de övriga kolumnerna på denna rad.

För att använda UUID-metoden:

1. Ta reda på den universella unika identiteten (UUID) på ditt filsystem genom: **ls -l /dev/disk/by-uuid | grep hda6**

Din rad bör se ut ungefär så här:

```
lrwxrwxrwx 1 root root 24 2009-01-30 08:16 d0dfcc8a-417a-41e3-ad2e-9736317f2d8a -> ../../hda6
```

UUID är namnet på den symboliska länk som pekar på /dev/hda6 exempelvis: d0dfcc8a-417a-41e3-ad2e-9736317f2d8a.

NOTERA



Ditt filsystem kommer ha en annan UUID som sträng.

2. Redigera /boot/grub/menu.lst och ändra följande rad:

```
# kopt=root=/dev/hda6 ro
```

till

```
# kopt=root=UUID=d0dfcc8a-417a-41e3-ad2e-9736317f2d8 ro
```

NOTERA



Ta inte bort # i början på raden, den måste vara där.

3. Uppdatera kernel-raderna i menu.lst genom att köra kommandot **update-grub**.

4. Redigera /etc/fstab och ändra raden som monterar /-partition, ex:

```
/dev/hda6 / ext3 defaults,errors=remount-ro 0 1
```

till

```
UUID=d0dfcc8a-417a-41e3-ad2e-9736317f2d8 / ext3 defaults,errors=remount-ro 0 1
```

Förändringen som spelar roll finns i första kolumnen, du behöver inte ändra de övriga kolumnerna på denna rad.

4.8.2 För att återhämta sig från problemet efter uppgraderingen

4.8.2.1 Lösning 1

Detta fungerar om Grub visar sin grafiska meny där du kan välja vilket alternativ som ska startas. Om denna meny inte visas försök trycka **Esc**-tangenter innan kärnan startar för att visa menyn. Om du inte kan visa menyn försök med Avsnitt [4.8.2.2](#) eller Avsnitt [4.8.2.3](#).

1. I Grub-menyn, markera alternativet du vill använda för att starta systemet. tryck **e**-tangenter för att redigera alternativet. Du kommer nu se något liknande detta:

```
root (hd0,0)
kernel /vmlinuz-2.6.26-1-686 root=/dev/hda6 ro
initrd /initrd.img-2.6.26-1-686
```

2. Markera raden

```
kernel /vmlinuz-2.6.26-1-686 root=/dev/hda6 ro
```

tryck **e**-tangenter och ersätt `hdX` med `sdX` (där X är bokstäverna a, b, c eller d beroende på ditt system). I vårt exempel blir raden:

```
kernel /vmlinuz-2.6.26-1-686 root=/dev/sda6 ro
```

Tryck sedan **vagnretur** för att spara ändringen. Om andra rader visar `hdX` uppdatera dessa också. Ändra inte delen som ser ut ungefär så här `root (hd0,0)`. När alla ändringar är klara tryck **b**-tangenter så bör systemet starta som vanligt.

3. När ditt system har startat upp behöver detta problemet lösas permanent. Hoppa till Avsnitt [4.8.1](#) och använd en av metoderna.

4.8.2.2 Lösning 2

Starta systemet på ett Debian GNU/Linux-installationsmedia (cd/dvd), ange `rescue` vid inmatningsstället för att starta återhämtningssläget. Välj språk, plats, tangentbordsupplägg och låt systemet försöka ställa in nätverket (det spelar ingen roll om det lyckas eller ej). Strax ska du ange vilken partition som ska användas som rotfilssystem. Alternativen bör se ut ungefär så här:

```
/dev/ide/host0/bus0/target0/lun0/part1
/dev/ide/host0/bus0/target0/lun0/part2
/dev/ide/host0/bus0/target0/lun0/part5
/dev/ide/host0/bus0/target0/lun0/part6
```

Om du vet vilken partition som är ditt rotfilssystem väljer du detta. Om du inte vet välj den första, systemet kommer tala om att det är ett ickefungerande rotfilssystem om partitionen inte kan användas, fortsätt att prova med nästa och så vidare. Att prova sig fram så här ska inte vara någon fara, har du bara ett rotfilssystem installerat bör det vara ganska enkelt att hitta det. Har du flera system installerade på dina diskar är det bättre att veta exakt vilken partition som är rätt.

När du väl har valt rätt partition får du välja mellan ett antal alternativ. Välj alternativet att starta ett skal på den valda partitionen, kan detta inte utföras prova med en annan partition.

Nu bör du ha skaltillgång som rotanvändaren på ditt rotfilssystem monterat på `/target`. Du behöver ha tillgång till innehållet i katalogerna `/boot`, `/sbin` och `/usr` på din hårddisk, dessa bör finnas i sökvägarna `/target/boot`, `/target/sbin` och `/target/usr`. Om dessa kataloger måste monteras från andra partitioner så ska du göra det först. (Konsultera `/etc/fstab` om du inte vet vilka partitioner som ska monteras).

Hoppa till Avsnitt [4.8.1](#) och genomför en av metoderna för att permanent laga felet. Ange sedan `exit` för att lämna återhämtningsskalet och välj `omstart` (`reboot` på engelska) för att starta om systemet som vanligt (glöm inte att avlägsna det startbara mediet).

4.8.2.3 Lösning 3

1. Starta systemet från en cd-skiva med ett körbart system, exempelvis Debian Live, Knoppix eller Ubuntu Live.
2. Montera partitionen där katalogen `/boot` finns. Om du inte vet vilken det är kan du använda kommandot **dmesg** för att se om din disk kallas `hda`, `hdb`, `hdc`, `hdd` eller `sda`, `sdb`, `sdc`, `sdd`. När du har kommit fram till vilken disk du ska använda, exempelvis `sdb`, kör följande kommando för att visa partitionstabellen och hitta rätt partition: **fdisk -l /dev/sdb**
3. Så länge du har monterat rätt partition i `/mnt` och att den partitionen innehåller katalogen `/boot` med rätt innehåll så ska du redigera filen `/mnt/boot/grub/menu.lst`.

Hitta avsnittet som liknar följande:

```
## ## End Default Options ##

title          Debian GNU/Linux, kernel 2.6.26-1-686
root           (hd0,0)
kernel         /vmlinuz-2.6.26-1-686 root=/dev/hda6 ro
initrd        /initrd.img-2.6.26-1-686

title          Debian GNU/Linux, kernel 2.6.26-1-686 (single-user mode)
root           (hd0,0)
kernel         /vmlinuz-2.6.26-1-686 root=/dev/hda6 ro single
initrd        /initrd.img-2.6.26-1-686

### END DEBIAN AUTOMAGIC KERNELS LIST
```

och ersätt alla `hda`, `hdb`, `hdc` och `hdd` med `sda`, `sdb`, `sdc` respektive `sdd`. Ersätt inte rader som liknar:

```
root           (hd0,0)
```

4. Starta om systemet, ta ut den körbara cd-skivan så startar ditt system som det ska.
5. När systemet har startat genomför du en av metoderna som beskrivs i Avsnitt 4.8.1 för att laga felet permanent.

4.9 Förberedelse inför nästa utgåva

Efter uppgraderingen finns det flera saker som du kan göra för att förbereda inför nästa utgåva.

- Om den nya kärnavbildens metapakete drogs in som ett beroende till den gamla, kommer det att markeras som automatiskt installerat, vilket bör korrigeras:

```
# aptitude unmarkauto $(dpkg-query -W 'linux-image-2.6-*' | cut -f1)
```

- Ta bort föråldrade och oanvända paket som beskrivs i Avsnitt 4.10. Du bör granska vilka konfigurationsfiler som de använder och överväga att avinstallera paketen fullständigt för att ta bort deras konfigurationsfiler

4.10 Föråldrade paket

lenny introducerar tusentals nya paket men pensionerar och utelämnar mer än två tusen gamla paket som fanns i etch. Det tillhandahålls inget uppgraderingssätt för dessa föråldrade paket. Ingenting hindrar dig från att fortsätta att använda ett föråldrat paket om så önskas men Debian-projektet kommer vanligtvis att sluta ge säkerhetsstöd för dessa ett år efter utgivningen av lenny⁵ och kommer normalt sett inte att tillhandahålla annat stöd under den tiden. Att ersätta dem med tillgängliga alternativ, om det finns några, rekommenderas.

⁵ Eller så länge som ingen annan utgivning sker i den tidsperioden. Normalt sett stöds endast två stabila utgåvor åt gången.

Det finns många anledningar till varför paket kan ha tagits bort från distributionen: de underhålls inte längre av upphovsmännen; det finns inte längre någon Debian-utvecklare som är intresserad i att underhålla paketet; funktionaliteten de tillhandahåller har ersatts av en annan programvara (eller en ny version); eller så anses de inte längre vara lämpliga för lenny på grund av fel i dem. I det senare fallet kan paket fortfarande finnas i 'unstable'-distributionen.

Att identifiera vilka paket på ett uppdaterat system som är 'föråldrade' är enkelt eftersom pakethanteringsvertygen markerar dem så. Om du använder **aptitude**, kommer du att se en lista över dessa paket under 'Föråldrade och lokalt skapade paket'. **dselect** tillhandahåller en liknande sektion men listan som visas kan skilja sig.

Om du har använt **aptitude** för att manuellt installera paket i etch kommer den att hålla kontroll på de paket som du manuellt installerat och kan markera de paket som själva har hämtats in via beroenden, vilka inte längre behövs om ett paket har tagits bort. **Aptitude**, till skillnad från **deborphan**, kommer inte markera de manuellt installerade paketen som föråldrade, i motsats till de som blev automatiskt installerade genom beroenden.

Det finns ytterligare verktyg som du kan använda för att hitta föråldrade paket, såsom **deborphan**, **debfooster** eller **cruft**. **deborphan** rekommenderas starkt, även om det endast kommer (i standardläget) rapportera om föråldrade bibliotek: paket i sektionerna 'libs' eller 'oldlibs' som inte används av några andra paket. Ta inte helt blint bort de paket som dessa verktyg presenterar, speciellt om du använder aggressiva ickestandardalternativ som är benägna att producera felaktigheter. Det rekommenderas starkt att du manuellt granskar paketen som föreslås för borttagning (alltså deras innehåll, storlek och beskrivning) innan du tar bort dem.

Debian Bug Tracking System (<http://bugs.debian.org/>) tillhandahåller ofta ytterligare information om varför paketet blev borttaget. Du bör granska både de arkiverade felrapporterna för själva paketet och de arkiverade felrapporterna för pseudopaketet på [ftp.debian.org](http://ftp.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes) (<http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes>).

The list of obsolete packages includes:

- apache (1.x), ersätts av apache2
- bind (8), successor is bind9
- php4, successor is php5
- postgresql-7.4, successor is postgresql-8.1
- exim (3), successor is exim4

4.10.1 Dummy-paket

Vissa paket från etch har delats upp i flera paket i lenny, ofta för att förbättra systemunderhållet. För att göra uppgraderingssättet enklare i sådana fall, tillhandahåller lenny ofta så kallade 'dummy'-paket: tomma paket som har samma namn som det gamla paketet i etch med beroenden som gör att de nya paketen blir installerade. Dessa 'dummy'-paket anses som föråldrade paket efter uppgraderingen och kan med säkerhet tas bort.

Beskrivningarna för de flesta (men inte alla) dummy-paket indikerar deras syfte. Paketbeskrivningar för dummy-paket är inte enhetliga, dock kan **deborphan** med flaggan `--guess` vara användbar för att identifiera dem på ditt system. Observera att vissa dummy-paket inte är tänkta att tas bort efter en uppgradering men används istället för att hålla kontroll på den för närvarande tillgängliga versionen av ett program över tid.

4.11 Inför nästa utgåva av Debian

4.11.1 Stöd för ARM ABI-porteringen avslutas till förmån för ARM EABI-porteringen

Debian lenny har två olika och inkompatibla ARM-porteringar, den äldre ABI-porteringen (arm) och den nyare EABI-porteringen (armel). Debian lenny är den sista utgåvan med stöd för ABI-porteringen och framtida utgåvor kommer bara att ha stöd för ARM EABI eller armel-porteringen. Därför bör armel användas för nya installationer av lenny.

Bortsett från Netwinder så finns det ARM-installationsavbildningar för både arm och armel i lenny. Netwinder-stöd finns bara för arm och kommer att försvinna i och med att arm-porteringen försvinner efter lenny.

Läs mer på [Debians wiki](http://wiki.debian.org/ArmEabiPort) (<http://wiki.debian.org/ArmEabiPort>) om ARM EABI-porteringen (armel).

Kapitel 5

Problemområden att känna till för utgåvan lenny

5.1 Potentiella problem

Ibland innebär förändringar att oväntade sidoeffekter uppstår och i vissa fall skapas nya fel. Här dokumenterar vi problem som vi känner till. Vänligen läs även erratan, dokumentationen för aktuellt paket, felrapporter och annan information som nämns i Avsnitt [6.1](#).

5.1.1 Problem med enheter på grund av udev

Trots att `udev` testats noga och mycket kan du uppleva mindre problem med några enheter. Det vanligaste felet är ändrade rättigheter och/eller äganderätten på en enhet. I vissa fall kan en enhet inte skapas (ex. `/dev/video` och `/dev/radio`).

`udev` tillhandahåller mekanismer för att åtgärda dessa problem. Läs mer i `udev(8)` och `/etc/udev`.

5.1.2 Vissa applikationer fungerar inte längre med kärna 2.4

Vissa applikationer i `lenny` fungerar inte längre med kärna 2.4, exempelvis för att de kräver stöd för `epoll()` vilket inte finns i kärna 2.4. Dessa applikationer kan inte köras alls eller fungerar inte som de ska förrrens systemet startas om med en 2.6-baserad kärna.

Ett exempel på detta är HTTP-proxyn `squid`.

5.1.3 Vissa nätverksplatser kan inte anslutas till via TCP

I och med 2.6.17 använder Linux fönsterstorleken i TCP aggressivt enligt RFC 1323. Vissa servrar är felaktigt inställda och ber om felaktiga fönsterstorlekar. Läs mer i felrapporterna [381262](http://bugs.debian.org/381262) (<http://bugs.debian.org/381262>), [395066](http://bugs.debian.org/395066) (<http://bugs.debian.org/395066>) och [401435](http://bugs.debian.org/401435) (<http://bugs.debian.org/401435>).

Det finns vanligen två vägar runt dessa problem; ändra den största tillåtna fönsterstorleken i TCP till ett mindre värde (rekomenderat) eller stäng av TCP:s föränderliga fönsterstorlek helt och hållet (detta fasas ut och ska inte användas). Exempel finns i [debian-installer errata](http://www.debian.org/devel/debian-installer/errata) (<http://www.debian.org/devel/debian-installer/errata>).

5.1.4 Automatisk avstängning slutar fungera

På en del äldre system kan inte kommandot `shutdown -h` stänga av systemet längre (systemet stannar bara). Detta händer på grund av att APM måste användas på dessa system. Genom att lägga till `acpi=off apm=power_off` i kärnans kommandorad i inställningarna för `grub` eller `lilo` ska lösa detta problem. Läs mer i felrapport [390547](http://bugs.debian.org/390547) (<http://bugs.debian.org/390547>).

5.1.5 Asynkron initiering av nätverksinstallation kan orsaka oförutsett beteende

På system som använder `udev` för att ladda drivrutiner för nätverksgränssnitt är det möjligt att den asynkrona naturen för `udev` medför att nätverksdrivrutinen inte laddas före `/etc/init.d/networking` utförs vid systemets uppstart. Inte ens genom att lägga till `allow-hotplug` till `/etc/network/interfaces` (tillsammans med `auto`) kan säkerställa att nätverksgränssnittet är aktiverat innan det börjar användas i uppstartssekvensen och detta kan i sin tur påverka resten av uppstarten.

5.1.6 Problem med att använda WPA-säkrade trådlösa nätverk

I `etch` var paketet `wpasupplicant` installerat som en systemservice med inställningarna lagrade i `/etc/default/wpasupplicant` och med användarspecifika inställningar lagrade i `/etc/wpasupplicant.conf`.

I `lenny` har `/etc/init.d/wpasupplicant` försvunnit och Debianpaketet är numera integrerat med `/etc/network/interfaces` i enlighet med hur andra paket, som exempelvis `wireless-tools`, fungerar. Detta innebär att `wpasupplicant` inte längre tillhandahåller en systemservice.

För information om hur inställningar för `wpasupplicant` görs läs mer i `/usr/share/doc/wpasupplicant/README.modes.gz` som ger exempel på olika `/etc/network/interfaces`. Uppdaterad information om användning av paketet `wpasupplicant` i Debian kan hittas på [Debian wikin](http://wiki.debian.org/WPA) (<http://wiki.debian.org/WPA>).

5.1.7 Problem med filnamn med tecken utanför ASCII-tabellen

Montering av filsystemen `vfat`, `ntfs` eller `iso9660` som innehåller filnamn med tecken utanför ASCII-tabellen leder till att fel uppstår när dessa filer ska användas, om alternativet `utf8` inte används. Följande felmeddelande kan vara en varningsklocka 'Felaktig eller ickekomplett multibyte eller brett tecken'. En lösning är att använda monteringsalternativet `defaults.utf8` för filsystemen `vfat`, `ntfs` och `iso9660` om de innehåller filnamn med tecken utanför ASCII-tabellen.

OBS! Linuxkärnan har inte stöd för skiftlägesökänslighet vid filnamnshantering för `vfat` om alternativet `utf8` används.

5.1.8 Ljudet slutar fungera

I vissa, ovanliga, fall kan det hända att ljudet slutar fungera efter en uppgradering. Om detta händer ska du gå igenom kontrollistan för ALSA:

- kör `alsaconf` som `root`-användare,
- lägg till din användare till gruppen `audio`,
- använd `alsamixer` och kontrollera så att alla ljudkanaler är aktiverade och volymen inte står på noll,
- kontrollera så att `arts` och `esound` är avstängda,
- kontrollera så att inga OSS-modulen inte är inlästa,
- kontrollera så att högtalarna faktiskt är igång samt
- klargör att

```
cat /dev/urandom > /dev/audio
```

or the command

```
speaker-test
```

fungerar för `root`.

5.2 NFS-monteringar hanteras av nfs-common

I och med `util-linux` version 2.13 hanteras NFS-monteringar inte längre av `util-linux` självt, det sköter `nfs-common`. Då inte alla system har nytta av att kunna montera NFS-utdelningar och för att undvika standardinstallation av portmapper är paketet `util-linux` bara inställt på att föreslå paketet `nfs-common`. Om du behöver kunna montera NFS-utdelningar måste du säkerställa att paketet `nfs-common` är isntallerat på ditt system. Förinstallationsskriptet för `mount`-paketet kontrollerar om NFS-monteringar existerar och avbryter om `/usr/sbin/mount.nfs` från `nfs-common` inte existerar eller om `nfs-common` inte är uppdaterat. Antingen uppgraderar du `nfs-common` eller så avmonterar du alla NFS-monteringar före uppgraderingen av `mount`.

5.3 Förändringar i rumänskt (ro) tangentbordsupplägg

På grund av uppgraderingar i `xkb-data` version 1.3 i `lenny` är standard upplägget för rumänska (ro) tangentbord att `ş` skrivs ut korrekt (med komma under) istället för `ș` (med cedilj under). Vissa varianter har dessutom fått nya namn, även om de gamla namnen fungerar uppmanas användare att redigera sin `/etc/X11/xorg.conf`. Mer information och möjliga sideeffekter på grund av denna förändringen finns beskrivna på [wikin \(endast på rumänska\)](http://wiki.debian.org/L10N/Romanian/Lenny/Notes) (<http://wiki.debian.org/L10N/Romanian/Lenny/Notes>).

5.4 Uppgradera apache2

Standardinställningarna i `apache2` är något uppdaterade och kan behöva vissa manuella justeringar. De viktigaste är:

`NameVirtualHost *` har ändrats till `NameVirtualHost *:80`. Om du har lagt till flera namnbaserade virtuella värdar måste du ändra `<VirtualHost *>` till `<VirtualHost *:80>` vid samtliga.

Användaren, grupp och sökvägen till `PidFile` anges numera i `/etc/apache2/envvars`. Om du har ändrat dessa inställningar från standardvärderna måste dessa ändras i den filen. Det innebär dessutom att man inte längre kan starta `apache2` med `apache2 -k start`, du måste använda `/etc/init.d/apache2` eller `apache2ctl`.

Hjälpprogrammet `suexec` som behövs för `mod_suexec` levereras numera som ett separat paket, `apache2-suexec`, detta installeras inte i standardutförandet.

Modulspecifika inställningar har flyttats från `/etc/apache2/apache2.conf` till `/etc/apache2/mods-available/*.conf`.

Läs mer om specifika detaljer i `/usr/share/doc/apache2.2-common/NEWS.Debian.gz` och `/usr/share/doc/apache2.2-common/README.Debian.gz`.

5.5 NIS och nätverkshanteraren

`ypbind`-versionen som finns i `nis` för `lenny` innehåller stöd för nätverkshanteraren. Detta stöd medför att `ypbind` avaktiverar NIS-klient-funktioner när nätverkshanteraren anger att datorn är nedkopplad från nätet. I och med att nätverkshanteraren ibland anger att datorn inte är uppkopplad när nätet inte används bör NIS-användare med NIS-klientsystem kontrollera att stödet för nätverkshanteraren är avaktiverat i systemet.

Detta kan göras antingen genom att avinstallera paketet `network-manager` eller genom att redigera `/etc/default/nis` för att lägga till `-no-dbus` till `YPBINDARGS`.

Användningen av `-no-dbus` är standard för nya installationer av Debian men var inte standardinställning i tidigare utgåvor.

5.6 Säkerhetsstatus för Mozillaprodukter

Mozillaprogrammen `firefox`, `thunderbird` och `sunbird` (i Debian modifierade och omdöpta till `iceweasel`, `icedove` respektive `iceowl`) är viktiga verktyg för många användare. Tyvärr är utvecklarens säkerhetspolicy att uppmantra användarna att uppgradera till nya versioner från utgivaren, detta står i konflikt till Debians policy om att inte skicka stora funktionella förändringar som säkerhetsuppdateringar. Vi kan inte förutse det idag men under livscykeln för `lenny` kan Debians säkerhetsgrupp ställas

inför faktumet att det är omöjligt att tillhandahålla Mozillas produkter på ett säkert sätt och därmed måste säkerhetsstödet för dessa avslutas. Du måste överväga detta när du börjar använda Mozillaprodukter och överväga alternativen som finns i Debian om faktumet att säkerhetsstödet försvinner är ett problem för dig.

`iceape`, den omdöpta och något modifierade versionen av Internetsviten `seamonkey` har tagits bort från `lenny` (med undantag för några få bibliotekpaket).

5.7 KDE skrivbordet

Det har inte gjorts några stora förändringar i skrivbordsmiljön KDE sedan versionen som följde med `etch`. `Lenny` innehåller en serviceutgåva som bland annat uppdaterar översättningar och vissa delar av KDE 3.5. Utgåvan är en blandning mellan KDE 3.5.9 och KDE 3.5.10. Några moduler är märkta 3.5.9 men har fått särskilda uppdateringar för att stämma överrens med förändringar till 3.5.10. Allt som allt kan man säga att `lenny` innehåller KDE 3.5.10 bortsett från förändringarna i `kicker` i `kdebase` samt några fellagningar i `kdepim`.

`Lenny` kommer vara den sista stabila utgåvan med KDE3-miljön.

5.8 Stöd och ändringar i Gnome-skrivbordet

Det har skett många förändringar i skrivbordsmiljön Gnome sedan versionen som följde med `etch` fram till den version som följer `md lenny`, närmare information om detta finns i [Kommentarer till utgåvan GNOME 2.22](#) (<http://library.gnome.org/misc/release-notes/2.22/index.html.sv>).

5.9 Inget stöd för Unicode i emacs21*

`Emacs21` and `emacs21-nox` are not configured to use Unicode by default. For more information and a workaround please see bug [#419490](#) (<http://bugs.debian.org/419490>). Consider switching to `emacs22`, `emacs22-gtk`, or `emacs22-nox`.

5.10 slurpd/replica kommer inte längre att fungera

OpenLDAP har inte längre stöd för LDAP-replikering via servicen `slurp` i och med version 2.4.7. Existerande installationer måste ställas om för att använda LDAP Sync Replication engine (`syncrepl`). Utförliga instruktioner finn på <http://www.openldap.org/doc/admin24/replication.html> (<http://www.openldap.org/doc/admin24/replication.html>).

5.11 Skrivbordet visar inte helskärmsläge

drivrutinen för Intel Mobile GM965 kan felaktigt upptäcka en VGA-utdata och ställa in storleken till ett lägre värde för att kompensera för detta. Symptomet av felet är att fönsterhanteraren bara kommer att visa en mycket liten fel av skärmen. Man kan tvinga fram rätt beteende genom att lägga till följande rader i `/etc/X11/xorg.conf`.

```
Section "Monitor"
    Identifier "VGA"
    Option "Ignore" "true"
EndSection
```

Please refer to the bug [#496169](#) (<http://bugs.debian.org/496169>) for more informations.

5.12 Problem med DHCP-överlämning vid fel

När serverpar med en DHCP-tjänst körs måste namnen vara konsekventa annars kommer DHCP att krascha.

Please see bug [#513506](#) (<http://bugs.debian.org/513506>) and <https://lists.isc.org/pipermail/dhcp-users/2007-September/004538.html> for more information.

5.13 VServer Disk Limit

To use the disk limit feature of vserver in lenny, you should use the **mount** option `tag` (instead of `tagxid` in etch).

You should manually update `/etc/fstab` and/or any script which uses `tagxid`. Otherwise, the partition will not be mounted and thus the vservers will not start.

Kapitel 6

Mer information om Debian GNU/Linux

6.1 Ytterligare läsning

Förutom dessa kommentarer till utgåvan och installationsguiden finns ytterligare dokumentation för Debian GNU/Linux tillgänglig från Debian Documentation Project (DDP), som har som mål att skapa högkvalitativ dokumentation för Debian-användare och utvecklare. Dokumentation som till exempel Debian Reference, Debian New Maintainers Guide, och Debian FAQ finns tillgängliga tillsammans med många andra dokument. För fullständiga detaljer om tillgängliga resurser, se webbplatsen för [DDP](http://www.debian.org/doc/ddp) (<http://www.debian.org/doc/ddp>).

Dokumentation för individuella paket installeras i `/usr/share/doc/paket`. Den kan inkludera information om upphövsrätt, Debian-specifika detaljer och dokumentation från upphovsmännen.

6.2 Få hjälp

Det finns många källor för hjälp, råd och stöd för Debian-användare, men dessa bör endast användas om dokumentationen inte har hjälpt till att lösa problemet. Det här kapitlet tillhandahåller en kort introduktion till dessa, vilka kan vara till hjälp för nya Debian-användare.

6.2.1 Sändlistor

De sändlistor som är mest intressanta för Debian-användarna är listan `debian-user` (engelsk) och andra `debian-user-språk`-listor (för andra språk, `swedish` för svenska). För information om dessa listor och detaljer om hur man prenumererar, se <http://lists.debian.org/>. Leta i arkiven efter svar på dina frågor innan du postar en ny fråga och därmed inte bryter listetiketten.

6.2.2 Internet Relay Chat

Debian har en IRC-kanal vars ändamål är att ge stöd och hjälp till Debian-användare. Kanalen finns på IRC-nätverket OFTC. För att komma åt kanalen, peka din favorit-IRC-klient till `irc.debian.org` och gå in i kanalen `#debian`.

Följ kanalens riktlinjer, respektera andra användare. Riktlinjerna finns tillgängliga på [Debians wiki](http://wiki.debian.org/DebianIRC) (<http://wiki.debian.org/DebianIRC>).

För mer information om OFTC, besök dess [webbplats](http://www.oftc.net/) (<http://www.oftc.net/>).

6.3 Rapportera fel

Vi strävar mot att göra Debian GNU/Linux till ett högkvalitativt operativsystem, det betyder dock inte att paketen som vi tillhandahåller är helt felfria. I enlighet med Debians filosofi om 'öppen utveckling' och som en tjänst till våra användare, tillhandahåller vi all information om rapporterade fel på vårt eget felhanteringssystem (BTS). BTS är bläddringsbar på <http://bugs.debian.org/>.

Om du hittar ett fel i distributionen eller i paketerad programvara som är en del av den, vänligen rapportera felet så att det kan rättas till i framtida utgåvor. Felrapportering kräver att du har en giltig e-postadress. Vi frågar efter den så att vi kan spåra fel och utvecklarerna kan komma i kontakt med de som rapporterat felet om de skulle behöva ytterligare information.

Du kan skicka in en felrapport med programmet **reportbug** eller manuellt via e-post. Du kan läsa mer om felhanteringssystemet och hur man använder det genom att läsa referenskorten (tillgängliga i `/usr/share/doc/debian` om du har paketet `doc-debian` installerat) eller i **felhanteringssystemet** (<http://bugs.debian.org/>).

6.4 Att bidra till Debian

Du behöver inte vara en expert för att bidra till Debian. Genom att hjälpa användare med problem på de olika **sändlistorna** (<http://lists.debian.org/>) för användarstöd bidrar du till gemenskapen. Identifiering (och lösning) av problem relaterade till utveckling av distributionen genom att delta i **sändlistorna** (<http://lists.debian.org/>) för utveckling är också mycket uppskattat. För att underhålla Debians högkvalitativa distribution ska du **skicka in felrapporter** (<http://bugs.debian.org/>) och hjälp utvecklarerna att spåra upp och rätta till felet. Om du är bra på att sätta ihop ord kanske du vill bidra mer aktivt genom att hjälpa till att skriva **dokumentation** (<http://www.debian.org/doc/ddp>) eller **översätta** (<http://www.debian.org/international/>) befintlig dokumentation till ditt eget språk.

Om du kan avsätta mer tid, skulle du kunna ansvara för en del av den fria programvaran i Debian. Speciellt behjälpligt är det om personer adopterar eller ansvarar för saker som folk har frågat efter om att inkluderas i Debian. Databasen **Work Needing and Prospective Packages** (<http://www.debian.org/devel/wnpp/>) har detaljer om den här informationen. Om du har intresse av mer specifika grupper kan du finna glädje i att bidra till några av Debians underprojekt vilka inkluderar porteringar till specifika arkitekturer, **Debian Jr.** (<http://www.debian.org/devel/debian-jr/>) och **Debian Med** (<http://www.debian.org/devel/debian-med/>).

I vilket fall som helst, om du arbetar i den fria programvarugemenskapen på något sätt, som en användare, programmerare, författare eller översättare hjälper du redan den fria programvaran. Att bidra är belönande och roligt, såväl som att det låter dig träffa nya människor som att det ger dig den där varma känslan inom dig.

Bilaga A

Hantera ditt etch-system

Denna bilaga innehåller information om hur du kontrollerar att du kan installera eller uppgradera paket från etch innan du uppgraderar till lenny. Det här bör endast vara nödvändigt i specifika situationer.

A.1 Uppgradering av ditt etch-system

Det är inga grundläggande skillnader mot någon annan uppgradering av etch som du gjort. Den enda skillnaden är att du först behöver se till att din paketlista fortfarande innehåller paket från etch vilket förklaras i Avsnitt [A.2](#).

Om du uppgraderar ditt system med en Debian-spegel kommer den automatiskt att uppgraderas till den senaste punktutgåvan av etch.

A.2 Kontrollera dina källistor

Om någon av raderna i din `/etc/apt/sources.list` refererar till 'stable', 'använder' du faktiskt redan lenny. Om du redan har kört `apt-get update`, kan du fortfarande komma tillbaka utan problem om du följer nedanstående procedur.

Om du även har installerat paket från lenny, är det antagligen inte så stor mening att installera paket från etch längre. I det fallet måste du bestämma dig för om du vill fortsätta eller inte. Det är möjligt att nedgradera paket, men det beskrivs inte här.

Öppna filen `/etc/apt/sources.list` med din favoritredigerare (som `root`) och kontrollera alla rader som börjar med `deb http:` eller `deb ftp:` efter en referens till 'stable'. Om du hittar någon, ändra `stable` till `etch`.

Om du har vissa rader som börjar med `deb file:` måste du själv kontrollera om platsen som de refererar till innehåller ett arkiv för etch eller lenny.

VIKTIGT



Ändra inte några rader som börjar med `deb cdrom:`. Om du gör det så ogiltigförklaras raden och du måste köra **apt-cdrom** igen. Bli inte rädd om en 'cdrom'-källrad refererar till 'unstable'. Även om det är förvirrande så är det normalt.

Om du har gjort några ändringar, spara filen och kör

```
# apt-get update
```

för att uppdatera paketlistan.

Bilaga B

Bidrag till Kommentarer till utgåvan

Många har hjälpt till med Kommentarer till utgåvan, bland andra

Adam Di Carlo, Andreas Barth, Andrei Popescu, Anne Bezemer, Bob Hilliard, Charles Plessy, Christian Perrier, Daniel Baumann, Eddy Petrișor, Emmanuel Kasper, Esko Arajärvi, Frans Pop, Giovanni Ragnani, Gordon Farquharson, Javier Fernández-Sanguino Peña, Jens Seidel, Jonas Meurer, Josip Rodin, Justin B Rye, LaMont Jones, Luk Claes, Martin Michlmayr, Michael Biebl, Moritz Mühlenhoff, Noah Meyerhans, Noritada Kobayashi, Osamu Aoki, Peter Green, Rob Bradford, Samuel Thibault, Simon Bienlein, Simon Paillard, Stefan Fritsch, Steve Langasek, Tobias Scherer, Vincent McIntyre, och W. Martin Borgert.

Detta dokument har översatts till många språk. Ett stort tack till alla översättare!

Svensk översättning av Martin Bagge och Daniel Nylander

Bilaga C

Till minne av Thiemo Seufer

Debianprojeketet har förlorat en aktiv medlem. Thiemo Seufer omkom i en tragisk bilolycka den 26 december 2008.

Thiemo bidrog till utvecklingen av Debian på många sätt. Han var ansvarig för många paket och var huvudansvarig för Debian-porteringen till MIPS-arkitekturen. Dessutom var Thiemo medlem av Debians utvecklingsgrupper för Linux-kärnan och Debians installerare. Hans bidrag sträckte sig dock utanför Debianprojeketet, han arbetade också med MIPS-porteringen av Linux-kärnan, MIPS-emulering i qemu och massor med mindre projekt som vi inte har plats att nämna här.

Thiemos arbete, hängivenhet och breda tekniska kunskap och förmåga att dela detta med andra kommer att saknas. Hans bidrag kommer inte att glömmas. Den höga nivån på Thiemos arbete kommer att bli svår att ta över.

För att hedra Thiemos bidrag till Debian tillägnas denna utgåva av Debian GNU/Linux 5.0 'lenny' till honom.

Bilaga D

Gloslista

ACPI

Advanced Configuration and Power Interface

ALSA

Advanced Linux Sound Architecture

APM

Advanced Power Management

cd

Compact Disc

cd-rom

Compact Disc Read Only Memory

DHCP

Dynamic Host Configuration Protocol

DNS

Domain Name System

dvd

Digital Versatile Disc

GIMP

GNU Image Manipulation Program

GNU

GNU's Not Unix

GPG

GNU Privacy Guard

IDE

Integrated Drive Electronics

LDAP

Lightweight Directory Access Protocol

LILO

Linux LOader

LSB

Linux Standards Base

LVM

Logical Volume Manager

MTA

Mail Transport Agent

NFS

Network File System

NIC

Network Interface Card

NIS

Network Information Service

OSS

Open Sound System

RAID

Redundant Array of Independent Disks

RPC

Remote Procedure Call

SATA

Serial Advanced Technology Attachment

USB

Universal Serial Bus

UUID

Universally Unique Identifier

VGA

Video Graphics Array

WPA

Wi-Fi Protected Access

Sakregister

A

Abiword, 6
Apache, 6

B

BIND, 6
Blu-ray, 6

C

CD, 6
Cherokee, 6
Courier, 6

D

Dia, 6
DocBook XML, 4
DVD, 6

E

Ekiga, 6
Emdebian, 8
Evolution, 6
Exim, 6

F

Firefox, 6

G

Gaim, 6
GCC, 6
GIMP, 6
GNOME, 6
GNUCash, 6
GNUmeric, 6

I

icedove, 6
IcedTea, 8
iceweasel, 6

J

Java, 8

K

KDE, 6
KOffice, 6

L

LILO, 22
Linux Standards Base, 6
LXDE, 6

M

Microsoft Windows, 9
Mozilla, 6, 31
MySQL, 6

N

Netwinder, 28
Network Manager, 31
NIS, 31

O

OpenJDK, 8
OpenOffice.org, 6
OpenSSH, 6
OpenVZ, 8

P

packages
 apache, 27
 apache2, 27
 apache2-suexec, 31
 Apt, 15, 19
 apt, 4, 15–17, 19
 Aptitude, 19
 aptitude, 6, 18, 19
 base-config, 19
 bind, 27
 bind9, 27
 dblatex, 4
 debian-goodies, 18
 doc-debian, 36
 docbook-xsl, 4
 eeepc-acpi-scripts, 8
 emacs22, 32
 emacs22-gtk, 32
 emacs22-nox, 32
 exim, 27
 exim4, 27
 firefox, 31
 glibc, 11
 grub, 29
 hotplug, 19
 iceape, 32
 icedove, 31
 icedoveicedove, 6
 iceowl, 31
 iceweasel, 31
 iceweaseliceweasel, 6
 initramfs-tools, 12, 21, 22
 ixp4xx-microcode, 5
 kernel-package, 21
 libc6, 19
 libselinux1, 19
 lilo, 22, 29
 linux-image-*, 21
 linux-image-2.6-686, 21
 localepurge, 18
 locales, 19
 lxde, 8
 mount, 31
 netkit-inetd, 19
 network-manager, 31

nfs-common, 31
nis, 31
php4, 27
php5, 27
popularity-contest, 18
postgresql-7.4, 27
postgresql-8.1, 27
python2.3, 19
release-notes, 3
rsyslog, 7
seamonkey, 32
squid, 29
sunbird, 31
sysklogd, 7
thunderbird, 31
udev, 21, 22, 29, 30
upgrade-reports, 3
util-linux, 31
wireless-tools, 30
wpa_supplicant, 30
xfree86-common, 19
xkb-data, 31
xlibs, 19
xmlroff, 4
xserver-common, 19
xsltproc, 4

PHP, 6

Pidgin, 6

Postfix, 6

PostgreSQL, 6

S

SELinux, 7, 19

synnedsättning, 10

T

Thunderbird, 6

Tomcat, 6

U

Unicode, 32

V

virtualization, 8

VServer, 8

W

WPA, 30

X

Xfce, 6