

Utgivelsesnotater for Debian GNU/Linux 5.0 (lenny), ARM EABI

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

September 19, 2009

Utgivelsesnotater for Debian GNU/Linux 5.0 (lenny), ARM EABI

Published 2009-02-14

This document is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License, version 2, as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Lisensteksten kan også hentes fra <http://www.gnu.org/copyleft/gpl.html> og `/usr/share/common-licenses/GPL-2` på ditt Debian GNU/Linux system.

Contents

1	Innledning	3
1.1	Rapportere feil i dette dokumentet	3
1.2	Bidra med oppgraderingsrapporter	3
1.3	Kilder brukt i dette dokumentet	4
2	Hva er nytt i Debian GNU/Linux 5.0	5
2.1	Hva er nytt for ARM?	5
2.2	Hva er nytt i distribusjonen?	6
2.2.1	Pakkehåndtering	6
2.2.2	Arkivet med proposed-updates	7
2.3	Systemforbedringer	7
2.4	Større endringer relatert til kjernen	8
2.4.1	Endringer i hvordan kjernen pakkes	8
2.5	Emdebian 1.0 (basert på Debian GNU/Linux lenny 5.0)	8
2.6	Støtte for Netbooks	8
2.7	Java er nå i Debian	8
3	Installasjonssystem	9
3.1	Hva er nytt i installasjonssystemet?	9
3.1.1	Større endringer	9
3.1.2	Automatisk installasjon	10
4	Oppgradering fra tidligere versjoner	11
4.1	Forberedelser før selve oppgraderingen	11
4.1.1	Ta sikkerhetskopi av alle dine filer og konfigurasjoner	11
4.1.1.1	Pass på at du bruker korrekt kjerne	11
4.1.2	Gi dine bruker beskjed på forhånd	11
4.1.3	Forberedelse til oppgradering	12
4.1.3.1	Feilsøkingsskall under oppstart ved hjelp av initrd	12
4.1.4	Forbered et sikkert miljø for oppgradering	12
4.2	Kontroller systemets status	13
4.2.1	Kontroller handlinger som er satt på vent i pakkebehandleren	13
4.2.2	Skrue av APT pinning	13
4.2.3	Kontroller pakkestatus	13
4.2.4	Avsnittet med proposed-updates	14
4.2.5	Uoffisielle kilder og backports	14
4.2.5.1	Å bruke pakker fra backports.org	14
4.3	Manuell fjerning av pakker markert som auto pakker	14
4.4	Klargjør APT sine kilder	15
4.4.1	Legge til APT-kilder for installering fra Internet	15
4.4.2	Legge til APT-kilder for installering fra lokalt speil	16
4.4.3	Legge til APT-kilde for installering fra CD-ROM eller DVD	16
4.5	Oppgradering av pakker	16
4.5.1	Ta opp hele oppgraderingsrutinen	17
4.5.2	Oppdatering av pakkelisten	17
4.5.3	Pass på at du har nok plass til oppgraderingen	17
4.5.4	Oppgrader apt og/eller aptitude først	19
4.5.5	Hvordan bruke aptitude sin liste over automatiske installerte pakker sammen med apt	19
4.5.6	Minimal systemoppgradering	19
4.5.7	Oppgradering av resten av systemet	19
4.5.8	Mulige problem under oppgraderingen	20
4.6	Oppgrader din kjerne og tilhørende pakker	21
4.6.1	Installere en metapakke for kjernen	21

4.6.2	Nytt system for nummerering av enheter	21
4.6.3	Tidsmessige problem under oppstart	22
4.7	Ting å huske før omstart	22
4.7.1	Kjør lilo igjen	22
4.8	Systemet stopper under oppstart med <code>Waiting for root file system</code>	22
4.8.1	Hvordan man unngår problemet før oppgradering	23
4.8.2	Hvordan løse problemer etter oppgraderingen	24
4.8.2.1	Løsning 1	24
4.8.2.2	Løsning 2	25
4.8.2.3	Løsning 3	25
4.9	Forberedelser for neste versjon	26
4.10	Avleggse pakker	26
4.10.1	Dummy-pakker	27
4.11	Planer for neste Debian versjon	27
4.11.1	Støtte for ARM EABI-porteringen istedenfor ARM-ABI-porteringen	27
5	Problemer du bør kjenne til vedrørende lenny	29
5.1	Potensielle problemer	29
5.1.1	Problemer med enheter i tillknytting til udev	29
5.1.2	Noen program vil ikke fungere med kjerne 2.4	29
5.1.3	Noen nettverkssteder kan ikke nås via TCP	29
5.1.4	Automatisk avskruing slutter å virke	29
5.1.5	Asynkron initiering av nettverket kan forårsake uforutsette konsekvenser	30
5.1.6	Problemer med å bruke trådløse nettverk beskyttet med WPA	30
5.1.7	Problemer i filnavn med tegn som ikke er ASCII	30
5.1.8	Lyd stopper å virke	30
5.2	NFS monteringer håndteres nå av <code>nfs-common</code>	31
5.3	Endringer i rumensk(ro) tastaturopsett	31
5.4	Oppgradere <code>apache2</code>	31
5.5	NIS og Network Manager	31
5.6	Sikkerhetsstatus for Mozillaprodukter	31
5.7	KDE skrivbordet	32
5.8	Støtte og endringer i Gnome skrivebordet	32
5.9	Ingen støtte for Unicode som standard i <code>emacs21*</code>	32
5.10	<code>slurpd/replica</code> vil ikke lenger fungere	32
5.11	Skrivebordet bruker ikke hele skjermen	32
5.12	DHCP problemer med feiltoleranse	32
5.13	VServer Disk Limit	33
6	Mer informasjon om Debian GNU/Linux	35
6.1	Ytterligere lesing	35
6.2	Få hjelp	35
6.2.1	E-postlister	35
6.2.2	Internet Relay Chat	35
6.3	Rapportere feil	35
6.4	Bidra til Debian	36
A	Forberedelser for oppgradering av et etch system	37
A.1	Oppgradering av ditt etch system	37
A.2	Kontroller dine arkivlister	37
B	Bidragstere til Utgivelsenotatene	39
C	Til minne om Thiemo Seufer	41
D	Glossary	43
	Index	45

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

Chapter 1

Innledning

Dette dokumentet informerer brukerne av Debian GNU/Linux distribusjonen om de største endringene i versjonen 5.0 (kodenavn 'lenny').

Dette dokumentet forklarer hvordan man på en sikker måte oppgraderer fra versjon 4.0 (kodenavn etch) til den aktuelle versjonen, og informerer også om de vanligste problemene man kan støte på.

Du finner den siste aktuelle versjonen av disse notatene på <http://www.debian.org/releases/lenny/releasenotes>. Hvis du er usikker på om du har den siste aktuelle versjonen, så sjekk datoen på forsiden.

CAUTION



Det er umulig for oss å ta med alle ting som kan gå galt, så derfor har vi bare tatt med et utvalg basert på kombinasjonen av hyppigst forekommende og alvorlighet.

Merk at vi kun støtter og dokumenterer hvordan man oppgraderer fra den forrige stabile versjonen av Debian (i dette tilfellet er det 4.0). Hvis du vil oppgradere fra en enda eldre versjoner så foreslår vi at du leser tidligere dokumentasjon på hvordan man oppgraderer versjonene i mellom, og begynner med å oppgradere til 4.0 først

1.1 Rapportere feil i dette dokumentet

Vi har forsøkt å teste alle skritt i dette dokumentet, vi har også forsøkt å beskrive alle mulige feil som kan oppstå når man oppgraderer.

Hvis du finner en feil (enten feil informasjon eller manglende informasjon) i dette dokumentet, så vær vennlig å rapporter dette som en feil i [feilrapportssystemet](http://bugs.debian.org/) (<http://bugs.debian.org/>) mot pakken som heter `release-notes`.

1.2 Bidra med oppgraderingsrapporter

Vi tar gjerne i mot informasjon fra dere brukere angående hvordan det gikk når dere oppgraderte fra etch til lenny. Hvis dere er villige til å dele denne informasjonen, så rapporter dette i [feilrapportssystemet](http://bugs.debian.org/) (<http://bugs.debian.org/>) mot pakken `upgrade-reports`. Vi ber deg å komprimere eventuelle vedlegg med kommandoen `gzip`.

Vennligst inkluder følgende informasjon når du sender inn en oppgraderingsrapport:

- Tilstanden på din pakkedatabase før og etter oppgraderingen: `dpkg` sin status er tilgjengelig i `/var/lib/dpkg/status` og `aptitude` sin status er tilgjengelig i `/var/lib/aptitude/pkgstates`. Du bør ta en sikkerhets kopi før oppgraderingen som beskrevet i Section 4.1.1, det skal også ligge en kopi i `/var/backups`.
- Sesjonsloger fra `script`, som beskrevet i Section 4.5.1.

- Ditt system sine apt logger, tilgjengelig i `/var/log/apt/term.log` eller loggene til **aptitude**, tilgjengelig i `/var/log/aptitude`.

NOTE

Du bør passe på å ikke inkludere sensitiv eller konfidensiell informasjon i dine loggfiler i en slik feilrapport, ettersom slike feilrapporter blir gjort offentlig tilgjengelig.

1.3 Kilder brukt i dette dokumentet

Kilden til dette dokumentet er i DocBook XML format. HTML versjonen er lagd med `docbook-xsl` og `xsltproc`. PDF versjonen er lagd med `dblatex` eller `xmlroff`. Kilden til Release Notes er tilgjengelig i SVN hos *Debian Documentation Project*. Du kan bruke **web interface** (<http://svn.debian.org/viewsvn/ddp/manuals/trunk/release-notes/>) for å få tilgang til enkeltfilene på nettet for å se på endringer i dem. For mer informasjon om hvordan du bruker SVN ta en titt på **Debian Documentation Project SVN information pages** (<http://www.debian.org/doc/cvs>).

Chapter 2

Hva er nytt i Debian GNU/Linux 5.0

Debian sin [Wiki](http://wiki.debian.org/NewInLenny) (<http://wiki.debian.org/NewInLenny>) inneholder mer informasjon om hva som er nytt.

Denne utgivelsen tilbyr nå offisiell støtte for ARM EABI (armel).

Følgende er de offisielt støttede prosessorvariantene for Debian GNU/Linux lenny:

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

Du kan lese om fremdriften på porteringsstatusen og spesifikk informasjon om din prosessorvariant på [Debian port nettsidene](http://www.debian.org/ports/) (<http://www.debian.org/ports/>).

2.1 Hva er nytt for ARM?

Støtte for Marvell's Orionplattform. Debian GNU/Linux 5.0 har støtte for følgende enheter basert 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/>), og **Buffalo Kurobox Pro** (<http://www.cyrius.com/debian/orion/buffalo/kuroboxpro/>).

Støtte for Versatileplattformen har blitt lagt til via emulering i QEMU.

Nettverksdriveren for IXP4xx plattformen (brukes f.eks Linksys NSLU2) har blitt integrert i standardkjernen, så kjernen i Debian lenny bruker nå denne, istedenfor den tidligere brukte uoffisielle driveren.

Den proprietære mikrokoden for IXP4xx som trengs for å bruke det innebygde nettverkskortet finnes nå i pakken `ixp4xx-microcode` i non-free arkivet. Installasjonsmedier for Debian som inneholder denne mikrokoden kommer til å finnes tilgjengelig på slug-firmware.net (slug-firmware.net).

2.2 Hva er nytt i distribusjonen?

Denne utgaven av Debian kommer med enda flere pakker enn forgjengeren etch; distribusjonen inneholder mer enn 7700 nye pakker, totalt følger det med 23200 pakker. Flesteparten av pakkene har blitt oppgradert: mer enn 13400 har blitt oppgradert (dette er 72 % av alle pakker i etch). En del pakker har av forskjellige årsaker blitt fjernet (mer enn 3100, det er 17 % av pakkene som var i etch). De pakkene som har blitt fjernet vil ikke få noen oppgradering, og de vil bli markert i pakkebehandlerene som 'obsolete'.

I denne utgaven av Debian GNU/Linux så er X.org oppgradert fra 7.1 til 7.3.

Det følger også denne gangen med i Debian GNU/Linux flere forskjellige typer skrivebordsprogram og -miljøer. Blant annet så følger skrivebordsmiljøet Gnome 2.22¹, KDE 3.5.10, Xfce 4.4.2, og LXDE 0.3.2.1+svn20080509. Produksjonspakkene har også blitt oppgradert, blant annet kontorpakkene OpenOffice.org 2.4.1 og KOffice 1.6.3, samt GNUcash 2.2.6, GNUMeric 1.8.3 og Abiword 2.6.4.

Andre skrivebordsprogram som har blitt oppgradert inkluderer Evolution 2.22.3 og Pidgin 2.4.3 (tidligere kalt Gaim). Mozilla programpakken har også blitt oppgradert: *iceweasel* (versjon 3.0.6) er Firefox nettleseren uten varemerker, og *icedove* (versjon 2.0.0.19) er Thunderbird e-postklienten men uten varemerker.

Denne utgaven inneholder også, blant annet, følgende programpakke oppdateringer:

Pakke	Versjon i 4.0 (etch)	Versjon i 5.0 (lenny)
Apache	2.2.3	2.2.9
BIND DNS Server	9.3.4	9.5.0
Cherokee web server	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 standard e-postserver	4.63	4.69
GNU Compiler Collection as default compiler	4.1.1	4.3.2
GIMP	2.2.13	2.4.7
GNU C library	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 offisielle Debian GNU/Linux distribusjonen leveres nå på mellom 4 og 5 binær DVD'er, eller på mellom 28 og 32 binær CD'er (antallet varierer med prosessorvariant) og 4 DVD'er eller 28 CD'er med kildekode. I tillegg så finnes det såkalte *multi-prosessorvarianter* DVD, som gjør det mulig å installere med samme DVD på prosessorvariantene amd64 og i386, kildekode ligger også på DVD'en. For første gang så leveres Debian GNU/Linux nå også som Blu-ray avbildninger, for prosessorvariantene amd64 og i386, kildekode følger også med.

Debian støtter nå Linux Standards Base (LSB) versjon 3.2. Debian 4.0 på sin side støttet versjon 3.1.

2.2.1 Pakkehåndtering

aptitude er den anbefalte pakkebehandleren når man jobber på kommandolinja. **aptitude** støtter de fleste kommandoflagg som **apt-get** gjør, og har vist seg å være bedre på å håndtere pakkeavhengigheter. Om du fortsatt bruker **dselect**, så bør du begynne å bruke **aptitude** som grensesnitt for pakkehåndteringen.

¹ Med noen deler fra GNOME 2.20.

For lenny så har det blitt implementert en avansert mekanisme i **aptitude** for å løse problemer som kan oppstå som følge av avhengigheter mellom pakker.

2.2.2 Arkivet med proposed-updates

Alle forandringer i den stabile versjonen (samt i old-stable versjonen), gjennomgår utstrakt testing før de blir akseptert inn i arkivet. Hver slik oppdatering i den stabile versjonen (samt i old-stable versjonen) kalles for en punktversjon. Forberedelsen til disse punktversjonene skjer gjennom mekanismen `proposed-updates`.

Pakker kan havne i `proposed-updates` på to forskjellige måter. Først, sikkerhetsoppgraderinger til en pakke som havner i `security.debian.org` blir automatisk lagt til i `proposed-updates`. Den andre måten er når en Debian GNU/Linux-utvikler laster en ny pakke opp direkte i `proposed-updates`. Den til enhver tid gjeldende listen over pakker kan sees på <http://ftp-master.debian.org/proposed-updates.html> (<http://ftp-master.debian.org/proposed-updates.html>).

Om du vil hjelpe med å teste pakkeoppdateringer før de formelt blir lagt inn i en punktversjon, så kan du gjøre det ved å legge til `proposed-updates` arkivet til 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
```

Neste gang du bruker kommandoen **aptitude update**, så vil ditt system begynne hente inn pakker fra `proposed-updates` når det skal oppgradere pakker.

Dette er egentlig ikke noen ny funksjon i Debian, men en funksjon som ikke har vært så godt kjent tidligere.

2.3 Systemforbedringer

Det har skjedd mange endringer i distribusjonen som vil komme nye installasjoner av lenny til gode, men ikke automatisk for systemer som blir oppgradert fra `etch`, i motsetning til systemer som blir installert helt fra bunnen av. Dette avsnittet gir et overblikk over de mest relevante endringene.

SELinux-prioriteten er standard, men den er ikke aktivert som standard. Pakkene som trengs for SELinux (Security-Enhanced Linux) har blitt gitt prioritet *standard*, det betyr at de kommer til å bli installert på nye installasjoner. For et system som oppgraderes, så kan disse pakkene installeres med kommandoen:

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

Merk at støtten for SELinux *ikke* er aktivert som standard. Informasjon om hvordan man aktiverer og konfigurerer SELinux finner du på [Debian Wiki](http://wiki.debian.org/SELinux) (<http://wiki.debian.org/SELinux>).

Ny standard syslog-nisse Standard syslog-nisse for Debian 5.0 er `rsyslog`, den erstatter `syslogd` og `klogd`. Om du ikke har noen særskilte loggingsregler konfigurert, så kan du begynne å bruke `rsyslog` uten noen endringer. Om du har særskilte loggingsregler, så må du migrere dem til den nye konfigurasjonsfilen `/etc/rsyslog.conf`.

Hvis du oppgraderer fra `etch`, så må du selv fjerne `sysklogd` og installere `rsyslog`. Standard syslog-nisse byttes ikke automatisk ut ved oppgradering til lenny.

Bedre støtte for UTF-8 Mange applikasjoner vil begynne å bruke UTF-8 som standard, eller de har bedre UTF-8 støtte enn tidligere. Les mer på <http://wiki.debian.org/UTF8BrokenApps> (<http://wiki.debian.org/UTF8BrokenApps>) om applikasjoner som fremdeles har problemer med å håndtere 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.

[Debian Wiki](http://wiki.debian.org/Etch2LennyUpgrade) (<http://wiki.debian.org/Etch2LennyUpgrade>) har mer informasjon om endringene mellom `etch` og lenny.

2.4 Større endringer relatert til kjernen

Debian GNU/Linux 5.0 leveres med kjerneversjonen 2.6.26 for alle prosessorvarianter.

Det har skjedd store endringer i både selve kjernen og i måten den blir pakket på i Debian. Noen av disse endringene gjør oppgraderingen mer komplisert, og kan også resultere i problem når det nye systemet blir startet for første gang etter oppgraderingen til lenny. Dette avsnittet gir en oversikt over de viktigste endringene. Mulige problem og informasjon om hvordan man kommer rundt disse problemene inkluderes i senere kapitler.

2.4.1 Endringer i hvordan kjernen pakkes

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.

Nye OpenVZ-kjernevarianter Debian GNU/Linux 5.0 tilbyr ferdig bygde kjerneversjoner for OpenVZ, en virtualiseringsløsning som tilbys side om side med Linux-VServer som ble introdusert i etch. OpenVZ inkluderer støtte for migrering av systemer mens de er i bruk, med eneste ulempe at de gir litt ekstra belastning.

En eneste pakke for alle x86 varianter. I tidligere versjoner fantes det en spesiell kjernevariant (-k7) for 32-bits AMD Athlon/Duron/Sempron-prosessorer. Denne varianten har blitt erstattet av en variant -686 som håndterer alle AMD/Intel/ViA-prosessorer av 686-varianten.

Der det er mulig så har overgangspakker (dummy) med de korrekte avhengigheter blitt satt inn istedenfor de som har blitt avlegs.

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

Lenny inneholder de nødvendige verktøy som trengs for å bygge Emdebian, noe som gjør det mulig å lage pakker som er små i størrelse og tilpasset ARM-systemer.

Distribusjonen Emdebian 1.0 inneholder ferdig bygde pakker for ARM tilstrekkelig for å lage root-filssystemer som kan tilpasses spesifikke maskiner og maskintyper. Kjerner og kjernemoduler må skaffes separat. Støtte for armel og i386 er under utvikling. Les mer på [Emdebian webpage](http://www.emdebian.org/) (<http://www.emdebian.org/>).

2.6 Støtte for Netbooks

Netbooks, slik som Eee PC fra Asus, støttes nå av Debian. For Eee PC, så les mer om pakken `eeepc-acpi-scripts`. Dessuten, Debian støtter nå også Lightweight X11 Desktop Environment, `lxde`, som er spesielt tilpasset for netbooks og datamaskiner med lav ytelse.

2.7 Java er nå i Debian

OpenJDK Java Runtime Environment `openjdk-6-jre` og Development Kit `openjdk-6-jdk` som trengs for å kjøre Java GUI og webbstartprogram eller for å bygge slike program, finnes nå tilgjengelig i Debian. Pakken lages med støtte for byggsystemet IcedTea og med noen forandringer fra IcedTea-prosjektet.

Chapter 3

Installasjonssystem

Debian Installer er det offisielle installasjonssystemet for Debian. Debian Installer støtter flere forskjellige måter å installere på, hvilke som er tilgjengelig for ditt system, er avhengig av din prosessorvariant.

Installasjonsmedier for lenny ligger sammen med Installasjonsguiden på [nettsiden til Debian](http://www.debian.org/releases/stable/debian-installer/) (<http://www.debian.org/releases/stable/debian-installer/>).

Installasjonsguiden finnes også på den første cd/dvd-platen, i katalogen

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

Du burde sjekke listen over kjente skrivefeil på [errata](http://www.debian.org/releases/stable/debian-installer/index#errata) (<http://www.debian.org/releases/stable/debian-installer/index#errata>)

3.1 Hva er nytt i installasjonssystemet?

Siden den første offisielle utgivelsen av Debian Installer i Debian GNU/Linux 3.1 (sarge) så har det blitt lagt ned et betydelig arbeid med Debian Installer, noe som har resultert i bedre hardware støtte samt en rekke nye spennende egenskaper.

I disse utgivelsesnotatene kommer vi bare til å liste opp de største endringene i Debian Installer. Hvis du er interessert i en mer detaljer oversikt over endringer siden etch, så les pressemeldingene for lenny beta og RC utgivelsene som er tilgjengelige fra Debian Installer sitt [nyhets arkiv](http://www.debian.org/devel/debian-installer/News/) (<http://www.debian.org/devel/debian-installer/News/>).

3.1.1 Større endringer

Støtte for innlasting av fastvare(firmware) under installasjonen Det er nå mulig å laste inn binære fastvare filer fra flyttbare media tilgjengelig utenfor Debian Installer installasjonsmedia.

Støtte for å installere direkte fra Microsoft Windows Installasjonsmedia inkluderer nå et program som gjør det mulig å starte installasjonen av Debian direkte fra Microsoft Windows.

Støtte for SATA RAID

Tidlig oppgradering av pakker med sikkerhetsfixs. Hvis et fungerende nettverk er tilgjengelig under installasjonen, så vil alle pakker som har blitt oppdatert siden utgivelsen av lenny bli oppgradert. Dette skjer under selve installasjonen, før systemet starter opp for første gang.

Som en konsekvens av dette, så er det ferdige installerte systemet mye tryggere, ettersom alle feilfixer som har kommet siden lenny, nå kommer med i ditt nye system.

Støtte for *volatile* Du kan under installasjonen velge om du vil at ditt nye system skal settes opp til å også bruke pakker fra volatile.debian.org. Dette pakkearkivet inneholder pakker som det er naturlig å oppdatere ofte, slik som pakker som inneholder informasjon om definisjoner av tidssoner, antivirus signatur filer, osv.

Nye prosessorvarianter Prosessorvarianten armel er nå støttet. Installasjonsmedier for i386 Xen gjest er også støttet.

Støtte for hardware baserte talesynteser Flere enheter som sørger for hardware basert talesyntese er nå støttet for bruk sammen med Debian Installer, noe som bedrer støtten for bruker med nedsatt syn.

Støtte for monteringsalternativet `relatime` Debian Installer kan nå sette opp partisjoner med monteringsalternativet `relatime`, dette innebærer at tidsstempelet for sist bruk av filen kun oppdateres hvis tidligere tidstempel er tidligere enn gjeldende tid for endring eller modifikasjon.

Klokkesynkronisering med NTP under installasjon Datamaskinens klokke blir nå synkronisert med en NTP server over nettverk under installasjonen, slik at det nye systemet har helt nøyaktig klokke.

Nye språk Takket være den enorme innsatsen til oversetterene, så kan Debian nå installeres på 63 forskjellige språk (50 via det tekstbaserte grensesnittet, og 13 kun via det grafiske grensesnittet). Dette er 5 språk flere enn det som var støttet i etch. Nye språk som nå er støttet inkluderer amharisk, marathi, irsk, nordsamisk og serbisk. På grunn av utdaterte oversettelser så er støtten for å installere med estisk fjernet. Et språk som falt ut i etch pga utdaterte oversettelse er nå tilbake: kymrisk (walisisk).

Språk som kun kan brukes sammen med det grafiske grensesnittet, ettersom de har et tegnsatt som ikke kan representeres i et rent tekstbasert grensesnitt er amharisk, bengalsk, dzongkha, gujarati, hindi, georgisk, khmer, malayalam, marathi, nepali, punjabi, tamil og thai.

Enklere valg av land Listen med valgbare land er gruppert etter kontinenter, noe som gjør det enklere å velge ett annet land enn det som er naturlig utifra valgt språk.

3.1.2 Automatisk installasjon

Noen av endringene som er nevnt ovenfor innebærer også endringer i støtten for automatisk installasjon med prekonfigurerte filer. Dette betyr at hvis du bruker prekonfigurerte filer som tidligere virket med etch Installer, så kan du ikke automatisk forvente at disse virker med den nye Installeren uten å først ha gjort de nødvendige endringene.

Installasjonsguiden (<http://www.debian.org/releases/stable/installmanual>) har et tillegg med utfyllende dokumentasjon på hvordan prekonfigurasjon kan brukes.

Chapter 4

Oppgradering fra tidligere versjoner

4.1 Forberedelser før selve oppgraderingen

Vi foreslår at du før du begynner oppgraderingen, leser informasjonen i Chapter 5. Det kapitelet dekker potensielle problemer som ikke er direkte tilknyttet selve oppgraderingen, men som det allikevel kan være lurt å kjenne til.

4.1.1 Ta sikkerhetskopi av alle dine filer og konfigurasjoner

Før du begynner på oppgraderingen så bør du ta en full sikkerhetskopi av alle dine filer, eller i det minste ta en sikkerhetskopi av alle filer du absolutt ikke kan miste. Oppgraderingsverktøyet og rutinen er til å stole på, men f.eks en hardware feil som oppstår midt i oppgraderingsrutinen kan etterlate systemet ditt i en ikke-fungerende tilstand.

De viktigste filene å ta sikkerhetskopi av er de som ligger i `/etc`, `/var/lib/dpkg`, `/var/lib/aptitude/pkgstates` og resultatet av kommandoen `dpkg --get-selections "*" (hermetegnene er viktige)`

Oppgraderingsrutinen endrer ikke noe på innholdet i katalogen `/home`. Men noen programmer (f.eks noen Mozilla-program og skrivebordsmiljøene KDE og GNOME) er kjent for å overskrive enkelte brukerinnstillinger når en ny versjon av disse programmene startes første gang. For sikkerhetskyld så bør du ta en sikkerhetskopi av alle skjulte filer og kataloger (såkalte 'punktfiler') i alle hjemmekataloger til dine brukere. Denne sikkerhetskopien kan du bruke for å gjenskape gamle innstillinger. Du vil kanskje også informere dine bruker om dette.

Om du ønsker å bruke pakkebehandleren, så må du gjøre dette med superbruker rettigheter, enten må du være logget på som `root`, eller så må du bruke kommandoen `su` eller `sudo` for å oppnå de nødvendige rettighetene.

Selve oppgraderingen har noen betingelser som må oppfylles, du bør se til at de faktisk er oppfylt før du begynner å oppgradere.

4.1.1.1 Pass på at du bruker korrekt kjerne

Versjonen av `glibc` som følger med `lenny` fungerer ikke sammen med kjerner eldre enn 2.6.8, uavhengig av hvilken prosessorvariant du bruker. Noen prosessorvarianter har til og med høyere krav. Vi anbefaler at du først oppgraderer kjernen med en kjerne som følger med `etch`, 2.6.18 eller 2.6.24. Eventuelt så kan du selv bygge din egen kjerne, men det må minst være versjon 2.6.18.

4.1.2 Gi dine bruker beskjed på forhånd

Det er lurt å informere dine bruker før du begynner på selve oppgraderingen, selv om brukere som er logget på via `ssh` oppkoblinger knapt kommer til å merke noe til at du oppgraderer, og burde kunne jobbe uavbrutt.

Hvis du vil være ekstra påpasselig, så kan du ta sikkerhetskopi av hjemmeområdet til dine brukere, eller avmontere partisjonen hvor de ligger (`/home`) før du oppgraderer.

Du må sannsynligvis også oppgradere kjernen din når du oppgraderer til `lenny`, da må du også ta en omstart på ditt system. Dette vil typisk skje helt til slutt når selve oppgraderingen er gjennomført.

4.1.3 Forberedelse til oppgradering

På grunn av de mange endringene i kjernen mellom etch og lenny, spesielt mht til drivere, gjenkjenning av hardware og navngiving, rekkefølgen på enhetsfiler, så finnes det en risiko for at du vil oppleve problemer når du starter opp ditt system første gang etter oppgraderingen. En del av disse problemene er kjent, og er dokumentert her i Utgivelsesnotatet.

Av den grunn er det lurt å forsikre seg om at du kan gjenopprette ditt system om det ikke klarer å starte opp, eller hvis ditt nettverk ikke virker etter oppstart og du er avhengig av nettverk for å komme inn på systemet.

Om du fjernoppgraderer via `ssh` så er det sterkt anbefalt at du har et opplegg på plass for å kunne komme inn på systemet via en fjernserielterminal. Det finnes en risiko for at du etter å ha oppgradert kjernen vil oppleve at noen enheter har byttet navn, som beskrevet her i Section 4.6.2, du vil da måtte bruke en lokal konsoll for å fikse dette. Om systemet plutselig omstartes midt i oppgraderingen, så må du kanskje også bruke lokal konsoll for å komme inn igjen.

Det første du bør gjøre er å forsøke å starte med din gamle kjerne. Men av ulike grunner dokumentert her i dette dokumentet, så er det ikke garantert å virke.

Hvis det ikke virker, så er et alternativ å starte systemet på en slik måte at du har en sjanse til å kunne reparere det. En måte å gjøre det på er å starte systemet fra en spesiell CD, enten en rednings-cd eller en såkalt Linux Live CD. Etter at du har startet opp ditt system fra en slik CD, så bør du kunne montere ditt rot-filsystem og deretter benytte deg av `chroot` for å komme inn i det og undersøke og forhåpentligvis reparere det.

En annen mulighet som vi anbefaler er å bruke det spesielle *rescue mode* valget som finnes i lenny Debian Installer. Fordelen med dette er at du her kan velge blant flere installasjonsmetoder, og lettere finne den som passer din situasjon best. For mer informasjon, se avsnittet 'Recovering a Broken System' i kapittel 8 i *Installation Guide* (<http://www.debian.org/releases/stable/installmanual>) og *Debian Installer FAQ* (<http://wiki.debian.org/DebianInstaller/FAQ>).

4.1.3.1 Feilsøkingsskall under oppstart ved hjelp av `initrd`

Pakken `initramfs-tools` inkluderer et feilsøkingsskall¹ i de `initrd` som den lager. Hvis f.eks din `initrd` ikke klarer å montere ditt rot-filsystem, så vil du bli tilbudt dette feilsøkingsskallet, som har alle nødvendige kommandoer tilgjengelig for å hjelpe deg med å reparere feilen.

Det første du bør se på er: er de nødvendige enhetsfilene tilstede i `/dev`; hvilke kjernemoduler er lastet (`cat /proc/modules`); resultatet av kommandoen `dmesg` vil vise mulige feilmeldinger ved lasting av drivere. Resultatet av kommandoen `dmesg` vil også fortelle deg hvilke enhetsfiler som har blitt tildelt dine harddisker, dette bør du kontrollere opp mot resultatet av kommandoen `echo $ROOT` for å være sikker på at rot-filsystemet er på den riktige harddisken.

Om du klarer å løse dine problem, så skriver du `exit` for å komme ut av feilsøkingsskallet og fortsette med oppstartsrutinen fra der hvor det feilet. Du må huske på å reparere problemet når du senere har fått startet opp ditt system, husk å lage ny `initrd`, ellers får du bare problemet igjen neste gang du starter systemet.

4.1.4 Forbered et sikkert miljø for oppgradering

Oppgraderingen av distribusjonen bør gjøres enten direkte fra en tekstbasert virtuell konsoll (eller en direkte tilkoblet seriell terminal), eller via fjernpålogging fra en `ssh` oppkobling.

Som en ekstra sikkerhet, så anbefaler vi at du benytter deg av programmet `screen` når du oppgraderer via fjernpålogging, da er du sikret at oppgraderingsrutinen ikke blir avbrutt selv om forbindelsen blir borte.

¹ Denne funksjonen kan skrues av med parameteren `panic=0` til dine oppstartsparametere.

IMPORTANT

Viktig! Du bør ikke oppgradere via **telnet**, **rlogin**, **rsh**, eller fra en X-sesjon som håndteres av **xdm**, **gdm** eller **kdm** på den maskinen du utfører oppgraderingen på. Dette er fordi prosessene som håndteres av disse tjenestene kan stoppe under selve oppgraderingen, noe som kan resultere i et *utilgjengelig* system som er bare halvveis oppgradert.

4.2 Kontroller systemets status

Oppgraderingsrutinen som er beskrevet her tar utgangspunkt i et 'rent' etch system som ikke inneholder tredjepartspakker. For best resultat, så bør du vurdere å fjerne slike eventuelle tredjepartspakker før du begynner å oppgradere.

Prosessen forutsetter at du først har oppgradert til siste punktversjonen av etch. Hvis du ikke allerede har gjort det, eller om du er usikker, så følg instruksjonene i Section [A.1](#).

4.2.1 Kontroller handlinger som er satt på vent i pakkebehandleren

I noen tilfeller så vil bruken av **apt-get** istedenfor bruken av **aptitude** for å installere en pakke, føre til at **aptitude** anser denne pakken for å ikke være i bruk av systemet, og derfor foreslå at den skal fjernes. Pass på at ditt system er helt oppdatert og 'rent' før du fortsetter med oppgraderingen.

Kontrollere om det finnes noen handlinger som er satt på vent i pakkebehandleren **aptitude**. Hvis en pakke er markert for å bli fjernet eller å oppdateres i pakkebehandleren, så kan dette ha negative konsekvenser for selve oppgraderingen av systemet. Merk at dette kun kan rettes på om din `sources.list` fremdeles forholder seg til *etch*; og ikke *stable* eller *lenny*; se Section [A.2](#).

For å gjøre dette så må du kjøre **aptitude** i såkalt 'visuelt modus' og der trykke på **g** ('Go'). Om det der finnes noen indikasjon på at noen handlinger ligger på vent, så bør du fikse dem, eller følge eventuelle anbefalinger. Hvis det ikke ligger noe på vent, så vil du få en beskjed som sier 'No packages are scheduled to be installed, removed, or upgraded'.

4.2.2 Skru av APT pinning

Hvis du har konfigurert APT til å installere enkelte pakker fra en et annet arkiv enn *stable* (f.eks fra *testing*), så kan det hende at du må endre denne konfigurasjonen (se i `/etc/apt/preferences`) for å få disse pakkene oppgradert til den versjonen som finnes i den nye stabile distribusjonen. For mer informasjon om APT pinning, så kan du lese manualsiden til `apt_preferences(5)`.

4.2.3 Kontroller pakkestatus

Uansett hvordan du planlegger å oppgradere, så lønner det seg å sjekke status på alle installerte pakker, og passer på at alle lar seg oppgradere. De følgende kommandoene vil finne pakker som har status som halvveis-installerte, eller som ikke er korrekt konfigurert.

```
# dpkg --audit
```

Du kan også sjekke status på alle installerte pakker på ditt system med kommandoene **dselect**, **aptitude**, eller kommandoer som

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

eller

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

Det er lurt å ta bort eventuelle merknader om at en pakke skal være på vent før man begynner på oppgraderingen. Hvis en pakke som er viktig for selve oppgraderingen er satt på vent, så kan hele oppgraderingen stoppe opp.

Merk at **aptitude** bruker en annen metode for å registrere at en pakke er satt på vent enn **apt-get** og **dselect**. Du kan finne ut hvilke som er på vent med kommandoen **aptitude** med

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

Om du med **apt-get** vil finne hvilke pakker som er satt på vent, så kan du bruke

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

Hvis du selv har bygd en pakke, og ikke gitt den et nytt navn, eller har lagt inn en dato i pakkenavnet, så må du sette den på vent for å forhindre at den blir oppgradert.

Pakkestatusen 'hold' for en pakke kan med **aptitude** endres med:

```
# aptitude hold pakkenavn
```

Bytt ut `hold` med `unhold` for å endre 'hold' status på en pakke.

Om det er noe du trenger å rette på, så lønner det seg å passe på at `sources.list` fortsatt refererer til `etch`, som forklart i Section [A.2](#).

4.2.4 Avsnittet med proposed-updates

Hvis du har et avsnitt med `proposed-updates` i din `/etc/apt/sources.list` fil, så bør du for sikkerhets skyld fjerne disse før du begynner på oppgraderingen.

4.2.5 Uoffisielle kilder og backports

Om du har noen ikke-Debian pakker installert på ditt system, så bør du være klar over at disse kan komme til å bli fjernet under oppgraderingen pga mulige konflikter. Hvis du har installert disse pakkene via et tilhørende arkiv nevnt i `/etc/apt/sources.list`, så bør du sjekke om dette arkivet også tilbyr disse pakkene for lenny, og endre de respektive linjene.

Hvis du har uoffisielle backportede pakker installert på ditt `etch` system som er 'nyere' enn de som finnes i Debian, så vil disse trolig skape konflikter under en oppgradering². Section [4.5.8](#) har mer informasjon om dette problemet skulle oppstå.

4.2.5.1 Å bruker pakker fra backports.org

`backports.org` er et semi-offisielt arkiv som vedlikeholdes av Debian GNU/Linux-utviklere. Dette arkivet inneholder nyere pakker enn de som finnes i det stabile arkivet, disse pakkene er basert på versjoner som finnes i 'testing'-arkivet.

Pakker i `backports.org` arkivet har pakker fra 'testing' men med et lavere versjonsnummer for å sikre at oppgraderingen fra `etch` backports til lenny går greit. Det finnes også pakker i backports-arkivet som kommer fra `unstable`-arkivet (sikkerhetsoppgraderinger og følgende unntak; Firefox, Linux kjernen, OpenOffice.org og 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 Manuell fjerning av pakker markert som auto pakker

For å hindre **aptitude** fra å fjerne pakker som har blitt installert gjennom en avhengighet i en annen pakke, så må du manuelt fjerne merkingen de har som `auto` pakker. Dette inkluderer OpenOffice.org og Vim for Skrivebordssystemer:

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

Dette gjelder også 2.6 kjerner om de er installert via en kjerne metapakke:

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

² Debian sitt pakkesystem tillater normalt ikke en pakke å fjerne eller overskrive en fil som tilhører en annen pakke, med mindre den er satt til på bytte ut denne pakken.

NOTE



Du kan se hvilke pakker som er merket som *auto* pakker i aptitude ved å kjøre:

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

4.4 Klargjør APT sine kilder

Før du starter på oppgraderingen, så må du konfigurere apt sin konfigurasjonsfil, `/etc/apt/sources.list`.

apt vil gå igjennom alle pakker som finnes via 'deb' linjer i konfigurasjonsfilen, og installere de med høyest versjonsnummer, linjene gis prioritet etter rekkefølge (slik at om du har flere speil oppført, så vil du typisk først føre opp lokal harddisk, så CD-ROM, og deretter HTTP/FTP-speil)

TIP



Det kan hende at du trenger å sette et GPG unntak for sjekking av DVD'er og CD-ROM'er. Legg til følgende linje i `/etc/apt/apt.conf`, hvis du ikke har en slik linje allerede i fila `/etc/apt/apt.conf.d/00trustedrom`:

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

Merk at dette vil ikke virke med DVD/CD-ROM avbildinger.

En utgivelse blir ofte referert til enten ved sitt kodenavn (f.eks `etch`, `lenny`), eller ved sitt statusnavn (f.eks `oldstable`, `stable`, `testing`, `unstable`). Hvis man holder seg til tradisjonen med å referere til kodenavn, så slipper man å bli overrasket når en ny versjon kommer, det er denne metoden vi bruker her. Det betyr at du selv må holde deg oppdatert på når det kommer en ny versjon. Om du bruker statusnavnet, så vil du kanskje først oppdage at det har kommet en ny versjon når alle pakker du har installert blir oppdaterte.

4.4.1 Legge til APT-kilder for installering fra Internet

Standardkonfigurasjonen er å laste ned pakker fra Debian sin hovedserver på Internet, men du vil kanskje endre `/etc/apt/sources.list` til å bruke en server som står nærmere deg, nettverksmessig.

Adressene til de forskjellige speilene til Debian HTTP eller FTP finner du på <http://www.debian.org/distrib/ftplist> (se på 'list of Debian mirrors'). HTTP-speil er generelt raskere enn FTP-speilene.

For eksempel, la oss anta at det nærmeste Debian speilet er `http://mirrors.kernel.org`. Når du bruker en nettleser for å se på innholdet i dette speilet, så vil du se at katalogene er organisert slik:

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

Legg til denne linjen i din `sources.list` fil om du vil bruke dette speilet med apt:

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

Merk at 'dists' legges til automatisk, og at argumentet etter versjonens kodenavn brukes for å utvide til å inkludere flere kataloger

Etter at du har lagt til en ny kilde, så kan du gjøre de tidligere brukte kildene inaktive ved å plassere en skigard (#) foran de tilhørende 'deb' linjene i `sources.list`

4.4.2 Legge til APT-kilder for installering fra lokalt speil

Istedenfor å bruke et HTTP- eller FTP-speil for å installere pakker, så kan du endre på `/etc/apt/sources.list` slik at det er et lokalt speil på din harddisk som brukes (eventuelt en nettverksdisk montert med NFS).

For eksempel, ditt lokale speil kan befinne seg på `/var/ftp/debian/`, og ha underkataloger som inneholder dette:

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

Legg til denne linjen i `sources.list` for å bruke den med `apt`:

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

Merk at 'dists' legges til automatisk, og at argumentet etter versjonens kodenavn brukes for å utvide til å inkludere flere kataloger

Etter at du har lagt til en ny kilde, så kan du gjøre de tidligere brukte kildene inaktive ved å plassere en skigard (#) foran de tilhørende 'deb' linjene i `sources.list`

4.4.3 Legge til APT-kilde for installering fra CD-ROM eller DVD

Hvis du *kun* vil bruke CD'er, så kommenter bort de eksisterende 'deb' linjene i `/etc/apt/sources.list` ved å plassere en skigard (#) foran dem.

Pass på at det finnes en linje i `/etc/fstab` som muliggjør for montering av CD-ROM på monteringspunktet `/cdrom` (**apt-cdrom** trenger at CD-ROM er montert på `/cdrom`). Hvis `/dev/hdc` er der din CD-ROM er, så vil den tilhørende linja i `/etc/fstab` se slik ut:

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

Merk at det *ikke må være noen mellomrom* mellom ordene `defaults, noauto, ro` i det fjerde feltet i denne linja.

For å sjekke om dette fungerer som det skal, så prøv

```
# mount /cdrom # dette vil montere CD-ROM åp monteringspunktet
# ls -alF /cdrom # dette vil liste opp innholdet åp roten til CD-ROM
# umount /cdrom # dette vil avmontere CD-ROM
```

Så kan du kjøre:

```
# apt-cdrom add
```

for hver Debian Binær CD-ROM som du har, for å legge de inn i APT sin database over tilgjengelige pakker.

4.5 Oppgradering av pakker

Den anbefalte metoden for å oppgradere til en nyere Debian GNU/Linux versjonen er å bruke pakkebehandleren **aptitude**. Denne pakkebehandleren gjør sikrere valg angående installasjon av pakker enn å bruke **apt-get** direkte.

Ikke glem å montere alle partisjoner (spesielt rot- og `/usr`-partisjonen) som skrivbare med en kommando som f.eks dette:

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

Deretter må du dobbeltsjekke at APT-kildene (i `/etc/apt/sources.list`) enten referer til 'lenny' eller til 'stable'. Det må ikke være noen referanser til `etch`.

NOTE



Merk at linjer for CD-ROM ofte referer til 'unstable', selv om dette tilsynelatende er feil, så er det faktisk oftest korrekt, du skal derfor *ikke* endre disse.

4.5.1 Ta opp hele oppgraderingsrutinen

Det anbefales på det sterkeste å lage et opptak av alt som skjer under oppgraderingen, bruk programmet `/usr/bin/script` for å gjøre dette. Så, om noe går galt under oppgraderingen, så har du en detaljer loggfil over hva som har hendt, og kan da gi eksakt informasjon i tilknytting til en feilrapport. For å starte opptaket, så skriv:

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

eller noe lignende. Ikke plasser loggfilene i en midlertidig katalog som `/tmp` eller `/var/tmp` (filer i disse katalogene blir ofte slettet ved en omstart eller ved oppgradering)

Loggfilene lar deg også se det som har rullet forbi på skjermen. Bytt til VT2 (bruk `Alt+F2`) deretter kan du bruke `less -R ~root/upgrade-lenny.script` for å se på innholdet i fila.

Etter at du er ferdig med oppgraderingen, så kan du stoppe **script** ved å skrive `exit` ved prompten.

Hvis du brukte opsjonen `-t` sammen med **script**, så kan du bruke **scriptreplay** for å spille av hele sesjonen.

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

4.5.2 Oppdatering av pakkelisten

Først må du hente den nye oversikten over pakker som finnes i den nye versjonen. Dette gjør du med:

```
# aptitude update
```

Første gang du kjører denne kommandoen etter at du har lagt til nye kilder, så vil du se en del advarsler relatert til tilgjengeligheten til disse nye arkivene. Disse advarslene er harmløse, og vil ikke dukke opp neste gang du kjører kommandoen.

4.5.3 Pass på at du har nok plass til oppgraderingen

Du må sjekke at du har nok plass på din harddisk før du starter oppgraderingen som beskrevet i Section 4.5.7. Alle pakker som skal installeres blir først lastet ned og plassert i `/var/cache/apt/archives` (under selve nedlastingen av pakken så blir de midlertidig plassert i `partial/` katalogen), så du må passe på at du har nok plass på partisjonen som har `/var/`. Etter at pakkene er lastet ned, så vil du trenge plass andre steder for selve installasjonen av pakkene. Hvis systemet ditt ikke har nok plass, så kan du risikere å ende opp med et ødelagt system.

Både **aptitude** og **apt** vil vise deg detaljert hvor mye diskplass de trenger for installasjonen. Før du starter oppgraderingen, så kan du se estimatet for hvor mye diskplass du trenger ved å kjøre:

```
# aptitude -y -s -f --with-recommends dist-upgrade
[ ... ]
XXX pakker oppgradert, XXX nylig installert, XXX skal fjernes og XXX skal ikke ←
  oppgraderes.
Trenger å hente XXXXMB i installasjonspakker. Etter utpakking vil XXXMB bli ←
brukt.
Vil laste ned, installere og/eller fjerne pakker.
```

NOTE



Hvis du forsøker å kjøre denne kommandoen på starten av oppgraderingsrutinen, så kan den gi mange feilmeldinger, av årsaker beskrevet i kommende avsnitt. I så fall så må du vente med denne kommandoen til du har fått unnagjort en minimal systemoppgradering som beskrevet i Section 4.5.6 og fått oppgradert kjernen.

Hvis du ikke har nok plass til å kunne oppgradere, så må du frigjøre plass. Du kan:

- Fjerne pakker som tidligere har blitt lastet ned (`/var/cache/apt/archives`) og installert. Disse pakkene slettes trygt med kommandoen **apt-get clean** eller **aptitude clean**.

- Fjerne gamle pakker du ikke lenger bruker. Hvis du har `popularity-contest` installert, så kan du bruke kommandoen **popcon-largest-unused** for å få en liste over pakker som ikke er mye i bruk, rangert etter installert størrelse. Du kan også bruke **deborphan** eller **debfooster** for å finne pakker som ikke lenger trengs (se Section 4.10). Alternativt så kan du starte **aptitude** i 'visual mode' og der finne utgåtte pakker under 'Obsolete and Locally Created Packages'.
- Fjerne pakker som bruker for mye plass, og som ikke trengs akkurat nå (du kan installere dem igjen når du er ferdig med å oppgradere). Du kan finne de pakkene som tar størst plass med kommandoen **dpigs** (denne finner du i pakken `debian-goodies`) eller du kan bruke kommandoen **wajig** (bruk da `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 oversettelser og språktilpassninger som du ikke trenger. Du kan installere pakken `localepurge` og sette den opp slik at bare et fåtall språk og språktilpassninger beholdes. Dette vil minske plassen som brukes på `/usr/share/locale`.
- Flytt systemets loggfiler midlertidig til et annet system, eller slett dem, disse filene finner du under `/var/log/`.
- Bruk en midlertidig `/var/cache/apt/archives` katalog: Du kan bruke en midlertidig cachekatalog på et annet filsystem (USB-enhet, midlertidig harddisk, et eksisterende filsystem, ...)

NOTE



Ikke bruk NFS-montering, ettersom nettverket kan bli forstyrret under oppgraderingen

For eksempel, hvis du har en USB-diskenheter montert på `/media/usbkey`:

1. Fjern pakker som har blitt lastet ned og allerede installert:

```
# apt-get clean
```

2. Kopier katalogen `/var/cache/apt/archives` over til USB-diskenheten

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

3. Monter den midlertidige cachekatalogen over den nåværende:

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

4. Etter at du er ferdig med oppgraderingen, så kan du legge tilbake den originale katalogen `/var/cache/apt/archives`:

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

5. Fjern gjenværende `/media/usbkey/archives`.

Du kan lage midlertidige cachekataloger hvor du vil på ditt monterte filsystem.

Merk at for å fjerne pakker på en trygg måte, så bør du bruke referanser til `etch` i fila `sources.list`, som beskrevet i Section A.2.

4.5.4 Oppgrader apt og/eller aptitude først

Mange feilrapporter indikerer at pakkene `aptitude` og `apt` i `etch` versjonen mange ganger feiler ved oppgraderingen til `lenny`. I `lenny` så håndterer `apt` best kompliserte situasjoner hvor mange pakker trenger å konfigureres samtidig, mens `aptitude` er best på å finne løsninger på pakkeavhengigheter. Ettersom disse to pakkene er helt sentrale ved en `dist-upgrade` til `lenny` så må disse pakkene oppgraderes før noe annet. For å oppgradere `apt` så gjør dette:

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

og for å oppgradere `aptitude` (hvis du har den installert), så gjør du:

```
# aptitude install aptitude
```

Dette vil automatisk oppgradere `libc6` og `locales`, og vil også installere støttebibliotekene til SELinux (`libselinux1`). Deretter kommer en del tjenester til på å bli restartet, blant annet `xdm`, `gdm` og `kdm`, dette medfører at lokale X-sesjoner kan komme til å bli avbrutt.

4.5.5 Hvordan bruke aptitude sin liste over automatiske installerte pakker sammen med apt

`aptitude` vedlikeholder en liste med pakker som har blitt installert automatisk (for eksempel som en avhengighet til en annen pakke). I `lenny` så finnes denne funksjonen også i `apt`.

Når du bruker `lenny` versjonen av `aptitude` første gang, så vil den lese inn listen med automatiske installerte pakker, og konvertere den for bruk sammen med `lenny` versjonen av `apt`. Hvis du har `aptitude` installert, så bør du i det minste kjøre kommandoen **`aptitude`** en gang for å gjennomføre denne konverteringen, for eksempel så kan du søke etter en pakke som ikke finnes.

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

4.5.6 Minimal systemoppgradering

På grunn av noen helt nødvendige konflikter mellom noen pakker som finnes i `etch` og `lenny`, så vil en direkte kjøring av kommandoen `aptitude dist-upgrade` ofte fjerne mange pakker som du helst vil ha installert. Vi anbefaler derfor en todelt oppgradering for å unngå dette problemet. Først foretaes en minimal oppgradering, deretter en full `dist-upgrade`.

Kjør først:

```
# aptitude safe-upgrade
```

Dette vil oppgradere de pakkene som kan oppgraderes uten at noen andre pakker blir fjernet eller installert.

Neste skritt avhenger av hvilke pakker du har installert på ditt system. Du finner råd og tips i Utgivelsesnotatet om hvilke løsning du burde bruke, hvis du er i tvil, så anbefaler vi at du nøye undersøker hvilke pakker som foreslås fjernet av hver metode før du fortsetter.

Pakker som du kan forvente vil bli fjernet inkluderer blant annet `base-config`, `hotplug`, `xlibs`, `netkit-inetd`, `python2.3`, `xfree86-common` og `xserver-common`. For en komplett liste over pakker som har blitt gjort avleggse i `lenny`, se Section 4.10.

4.5.7 Oppgradering av resten av systemet

Du er nå klar for å fortsette med hoveddelen av oppgraderingen. Kjør:

```
# aptitude dist-upgrade
```

Dette kommer til å utføre en fullstendig oppgradering av ditt system, altså kommer den nyeste versjonen av alle installerte pakker til å bli installert, samt løse alle avhengigheter mellom pakker i forskjellige versjoner. Om det er nødvendig, så kommer også nye pakker til å bli installert (vanligvis dreier det seg om nye versjoner av bibliotek eller pakker som har fått nytt navn), avleggse pakker som står i konflikt med nyere pakker vil bli fjernet.

Ved oppgradering fra et sett av cd'er (eller dvd'er), så vil du bli bedt om å sette inn spesifikke cd'er flere ganger i løpet av oppgraderingen. Det kan hende at du blir bedt om å sette inn den samme cd'en flere ganger, dette er fordi en pakke kan ha avhengighet på en pakke som befinner seg på en annen cd.

Nye versjoner av installerte pakker som ikke kan oppgraderes uten å endre installasjonsstatus på andre pakker, vil ikke bli oppgradert (de vises som 'held back'). Disse kan man tvinge gjennom en oppgradering ved å bruke **aptitude** for å installere disse enkeltvis, eller ved å bruke `aptitude -f install pakkenavn`.

4.5.8 Mulige problem under oppgraderingen

Hvis du ser denne feilen når du bruker **aptitude**, **apt-get** eller **dpkg**:

```
E: Dynamic MMap ran out of room
```

så skyldes det at standard størrelsen på cachen er for liten. Du kan løse dette problemet ved å enten fjerne unødvendige linjer i `/etc/apt/sources.list`, eller ved å øke størrelsen på cachen. Størrelsen på cachen kan du øke med `APT::Cache-Limit` i `/etc/apt/apt.conf`. Denne kommandoen burde øke størrelsen til noe som er stort nok for oppgraderingen:

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

Dette forutsetter at du ikke allerede har denne variabelen i fila.

Noen ganger kan det være nødvendig å aktivere alternativet `APT::Force-LoopBreak` i APT for å kunne midlertidig fjerne en systemkritisk pakke på grunn av en `Conflicts/Pre-Depends`-løkke. **aptitude** vil advare deg om dette, og avbryte oppgraderingen. Du kan komme rundt dette ved å angi alternativet `-o APT::Force-LoopBreak=1` sammen med kommandoen **aptitude**.

Det er mulig at avhengigsstrukturen for et system kan være så ødelagt at det krever manuell inngripen. Vanligvis betyr dette at du må bruke **aptitude** eller

```
# dpkg --remove pakkenavn
```

for å fjerne noen av de pakkene som er i veien, eller

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

I helt ekstreme tilfeller så kan det hende at du må reinstallere pakken med en kommando som

```
# dpkg --install /sti/til/pakkenavn.deb
```

Filkonflikter bør ikke oppstå om du forsøker å oppgradere fra et 'rent' etch system, men kan skje om du har uoffisielle backports installert. En filkonflikt kan resultere i en feilmelding som dette:

```
Unpacking <package-foo> (from <package-foo-file>) ...
dpkg: error processing <package-foo> (--install):
 trying to overwrite '<some-file-name>',
 which is also in package <package-bar>
dpkg-deb: subprocess paste killed by signal (Broken pipe)
Errors were encountered while processing:
<package-foo>
```

Du kan forsøke å løse denne filkonflikten ved å tvinge igjennom fjerningen av pakken som nevnes på den *siste* linjen i feilmeldingen:

```
# dpkg -r --force-depends pakkenavn
```

Etter at dette problemet er løst, så burde du kunne forsette med oppgraderingen ved å fortsette med **aptitude** kommandoene som tidligere nevnt.

Under oppgraderingen så kan du komme til å bli stilt en rekke spørsmål om hvordan du vil at en pakke skal konfigureres, eller omkonfigureres. Når du blir spurt om hva som skal gjøres med en fil som befinner seg i enten `/etc/init.d` eller `/etc/terminfo` katalogene, eller `/etc/manpath.config` fila, om du vil bytte ut en fil med pakkevedlikeholderens nye versjon, så er vanligvis det nødvendige svaret "ja", for å beholde et korrekt fungerende system. Du kan alltid angre dette valget, ettersom de gamle versjonene blir tatt vare på med en `.dpkg-old` filendelse.

Hvis du er usikker på hva du burde gjøre, så skriv ned navnet på den aktuelle fila, eller pakken, så kan du ordne opp i det senere. Du kan se i typeskript-fila for å nærmere undersøke informasjonen som ble vist på skjermen.

4.6 Oppgrader din kjerne og tilhørende pakker

Dette avsnittet forklarer hvordan du oppgraderer kjernen, og tar for seg mulige problem som kan oppstå når du oppgraderer kjernen. Du kan enten installere en av `linux-image-*` pakkene som følger med Debian, eller lage dine egne kjerner direkte fra kildekoden.

Merk at mye av informasjonen i dette avsnittet tar utgangspunkt i at du bruker en av kjernepakken som følger med Debian, sammen med `initramfs-tools` og `udev`. Hvis du bygger dine egne kjerner som ikke trenger en `initrd`, eller om du bruker en annen `initrd`-generator, så kan det hende at informasjonen her ikke gjelder deg.

4.6.1 Installere en metapakke for kjernen

Når du kjører `dist-upgrade` fra `etch` til `lenny`, så anbefales det på det sterkeste at du installerer en ny `linux-image-2.6-*` metapakke. Denne pakken kan installeres automatisk av `dist-upgrade` prosessen. Du kan kontrollere dette ved å kjøre:

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

Om du ikke får noe svar, så må du installere en ny `linux-image` metapakke manuelt. Kjør denne kommandoen for å få en oversikt over tilgjengelige `linux-image-2.6` metapakker:

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

Om du er usikker på hvilken pakke du skal velge, så kjør `uname -r` og let etter en pakke med et lignende navn. For eksempel, hvis du ser `'2.6.18-6-686'`, så anbefales det at du installerer `linux-image-2.6-686`. (Merk at varianten `k7` ikke lenger finnes, hvis du bruker `k7`, så skal du nå velge `686` istedenfor). Du kan også bruke `apt-cache` for å se en lengre beskrivelse for hver pakke, slik at du lettere kan velge riktig pakke. For eksempel:

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

Du bør så bruke `aptitude install` for å installere den. Så snart den nye kjernen er installert, så bør du ved første anledning starte systemet pånytt, slik at du kan begynne å bruke den nye forbedrede kjernen.

For de mer eventyrlystne så finnes det en enkel måte å bygge dine egne spesialtilpassede kjerner i Debian GNU/Linux. Installer pakken `kernel-package` og les dokumentasjonen i `/usr/share/doc/kernel-package`.

Hvis det er mulig, så er det i til din fordel om du kan oppgradere kjernen separat fra selve hoved `dist-upgrade` prosessen, da reduserer du sjansen for at du ender opp med et system som ikke vil starte. Merk at dette ikke bør forsøkes før du har utført en minimal oppgradering, som beskrevet i Section 4.5.6.

4.6.2 Nytt system for nummerering av enheter

`lenny` har en mer robust mekanisme for å identifisere hardware enn tidligere versjoner hadde. Men, det kan hende at denne nye mere robuste mekanismen vil påvirke rekkefølgen på hvordan enhetsnavn tildeles. Om du f.eks har to nettverkskort som bruker forskjellige drivere, så kan de enheter som `eth0` og `eth1` referer til endres. Merk at om du bytter ut et nettverkskort på et `lenny` system, så vil det nye nettverkskortet få et nytt navn.

For nettverksenheter, så kan du unngå dette ved å bruke regler i `udev`, mer spesifikt gjennom definisjoner i `/etc/udev/rules.d/70-persistent-net.rules`³. Alternativt så kan du bruke kommandoen `ifrename` for å sette opp en bestemt sammenheng mellom en fysisk enhet og et spesifikt navn. `Seifrename(8)` og `iftab(5)` for mer informasjon. De to alternativene (`udev` og `ifrename`) bør ikke brukes samtidig.

For lagringsenheter, så kan du unngå dette ved å bruke `initramfs-tools`, og konfigurere den slik at den laster drivermoduler i samme rekkefølge som de nå blir lastet. For å finne ut i hvilken rekkefølge dine nåværende drivermoduler har blitt lastet, så kan du se på resultatet fra kommandoen `lsmod`. `lsmod` lister opp drivermoduler i omvendt rekkefølge av når de ble lastet, dvs at den første

³ Reglene der genereres automatisk av skriptet `/etc/udev/rules.d/75-persistent-net-generator.rules` for å ha statiske navn på nettverksenhetene. Slett den symbolske lenken for å gjøre den statiske enhetsnavngivningen via NIC via `udev` inaktiv.

drivermodulen som listes er den siste som ble lastet. Merk at dette kun gjelder for enheter som kjernen nummerer på en fast måte (som PCI enheter).

Merk at om moduler fjernes og lastes pånytt etter oppstart, så vil dette påvirke rekkefølgen. Merk også at din kjerne kan ha moduler statisk lenket, og disse vil da ikke dukke opp på listen fra `lsmod`. Du kan kanskje klare å finne ut av dette ved å inspisere fila `/var/log/kern.log`, eller ved å se på kommandoen `dmesg`.

Legg til modulene i `/etc/initramfs-tools/modules` i den rekkefølgen de skal lastes automatisk ved oppstart. Noen moduler har byttet navn mellom versjonene `etch` og `lenny`. Feks så har `sym53c8xx_2` byttet navn til `sym53c8xx`.

Du blir nødt til å lage dine `initramfs`-avbildninger på nytt ved å kjøre `update-initramfs -u -k all`.

Når du har begynt å bruke kjerne og `udev` fra `lenny` så kan du konfigurere systemet ditt til å akseptere diskene gjennom et alias på en slik måte at rekkefølgen de oppdages i ikke betyr noe. Disse aliasene finner du under `/dev/disk/` strukturen.

4.6.3 Tidsmessige problem under oppstart

Hvis en `initrd` som er lagd med `initramfs-tools` brukes for å starte systemet, så kan det i noen tilfeller hende at opprettingen av enhetsfiler av `udev` skjer for sent for noen oppstartsskripts.

De vanlige symptom på dette er at oppstarten misslykkes fordi rotfilsystemet ikke kan monteres, og du kommer bare til et feilsøkingsskall. Når du senere sjekker om de nødvendige enhetsfilene er på plass i `/dev`, så er de der alle sammen. Dette har blitt observert på systemer hvor rotfilsystemet ligger på en USB-enhet, eller på et RAID, spesielt hvis `LILO` brukes.

En måte å komme rundt dette problemet på er å bruke oppstartsparameteren `rootdelay=9`. Verdien for tidsgrensen (i sekunder) kan behøves å endres.

4.7 Ting å huske før omstart

Når `aptitude dist-upgrade` er ferdig med å kjøre, så er den 'formelle' oppgraderingen ferdig, men det er visse andre ting som bør taes hånd om *før* neste omstart.

4.7.1 Kjør lilo igjen

Hvis du bruker `lilo` som din oppstartslaster (den var brukt som standard oppstartslaster på enkelte installasjoner av `etch`) så anbefales det at du kjører `lilo` etter oppgraderingen:

```
# /sbin/lilo
```

Merk at dette er nødvendig selv om du ikke har oppgradert kjernen, ettersom oppgraderingen av `lilo` i seg selv har endret andre steget i `lilo`.

Husk også å kontrollere innholdet i `/etc/kernel-img.conf`, pass på at du der har `do_bootloader = Yes`, da vil oppstartslasteren alltid bli kjørt etter en oppgradering av kjernen.

Hvis det oppstår problemer når du kjører `lilo`, så kontrollert symlinkene fra `/` til `vmlinuz` og `initrd`, og kontroller at innholdet i `/etc/lilo.conf` stemmer.

Hvis du glemmer å kjøre `lilo` før du restarter, eller hvis systemet ved et uhell restarter, så kan det hende at ditt system ikke starter. Isteden for å se det vanlige `lilo`-bildet, så ser du bare `LI` når du forsøker å starte systemet⁴. Se Section 4.1.3 for hvordan du løser dette problemet.

4.8 Systemet stopper under oppstart med Waiting for root file system

Prosedyre for å rette opp problem hvor `/dev/hda` ble til `/dev/sda` Noen brukere har rapportert at oppgraderingen har ført til at kjernen ikke har klart å finne systemets rofilsystem etter opprtart.

I slike tilfeller så kommer systemet til å stoppe opp med feilmeldingen:

⁴ For mer informasjon om `lilo` sine feilkoder, se [The Linux Bootdisk HOWTO](http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html) (<http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html>).

```
Venter åp rotfilssystem... (engelsk: Waiting for root file system ...)
```

og etter noen sekunder så dukker en enkel busybox terminalprompt opp.

Dette problemet kan oppstå når en oppgradering av kjernen medfører at en ny type IDE drivere. Konvensjonen for navngivning med de gamle driverene var å bruke `hda`, `hdb`, `hdc`, `hdd`. De nye driverene vil kalle de samme diskene for `sda`, `sdb`, `sdc`, `sdd`. Problemet oppstår når disse nye navnene ikke blir inkludert i `/boot/grub/menu.lst`. Under oppstart så vil Grub gi kjernen utdatert informasjon om hvor rotfilssystemet befinner seg.

Hvis du har støtt på dette problemet etter oppgraderingen, så les Section 4.8.2. Hvis du ønsker å unngå dette problemet, så les videre.

4.8.1 Hvordan man unngår problemet før oppgradering

Problemet kan unngås ved å bruke en identitet for rotfilssystemet som ikke forander seg mellom de forskjellige oppstartene. Det finnes to mulige metoder for å oppnå dette. En måte er å sette en slags etikett på filsystemet, eller ved å bruke en såkalt universiell unik identitet (UUID). Begge disse metodene har vært støttet i Debian siden 'etch' utgaven.

Disse metodene har fordeler og ulemper. Metoden med etiketter er lettere å lese, men sjansen er tilstede for at et annet filsystemt kan ha samme etikett. Metoden med UUID er ikke fullt så enkel å lese, men det er veldig lite sannsynlig at to filsystem kan ha samme UUID.

I dette eksempelet så antar vi at rotfilssystemet ligger på `/dev/hda6`. Vi antar også at systemet har fungerende `udev`, og er et `ext2` eller `ext3` filsystem.

For å bruke etikettmetoden:

1. Sett en etikett på filsystemet (navnet må være på (mindre) < 16 tegn) ved å bruke kommandoen:
e2label /dev/hda6 rootfilesys
2. Rediger `/boot/grub/menu.lst` og endre linjen:

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

til

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

NOTE



Ikke ta bort # fra begynnelsen av linja, den må være der.

3. Oppdatert `kernel`-linjene i fila `menu.lst` ved å bruke kommandoen **update-grub**.
4. Rediger fila `/etc/fstab` og endre linja som er ansvarlig for montering av / (rootfilssystemet), for eksempel:

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

til

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

Endringen som betyr noe her er den som utføres i første kolonne, du trenger ikke å endre på noe annet på linja.

For å bruke UUID-metoden:

1. For å finne universally unique identifier til ditt filsystem, så bruk: **ls -l /dev/disk/by-uuid | grep hda6**

Din linje bør se ut som dette:

```
lrwxrwxrwx 1 root root 24 2008-09-25 08:16 d0dfcc8a-417a-41e3-ad2e-9736317f2d8a -> ../../hda6
```

UUID er navnet på den symbolske lenken som peker til `/dev/hda6` for eksempel: `d0dfcc8a-417a-41e3-ad2e-9736317f2d8a`.

NOTE



Ditt filsystem komme til å ha en annen UUID-streng.

2. Rediger `/boot/grub/menu.lst` og endre linjen:

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

til

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

NOTE



Ikke ta bort # fra begynnelsen av linja, den må være der.

3. Oppdatert kernel-linjene i fila `menu.lst` ved å bruke kommandoen **update-grub**.

4. Rediger fila `/etc/fstab` og endre linja som er ansvarlig for montering av `/` (rootfilsystemet), for eksempel:

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

til

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

Endringen som betyr noe her er den som utføres i første kolonne, du trenger ikke å endre på noe annet på linja.

4.8.2 Hvordan løse problemer etter oppgraderingen

4.8.2.1 Løsning 1

Dette fungerer om Grub viser sin grafiske meny der du kan velge hvilke valg du vil starte opp fra. Hvis du ikke ser denne menyen, så prøv å trykke på **Esc** før kjernen starter for å vise menyen. Hvis du ikke kommer inn i menyen, så forsøk Section 4.8.2.2 eller Section 4.8.2.3.

1. Når du er inne i Grub menyen, så marker alternativet du vil starte. Trykk **e** (e for engelsk edit) for å redigere dette alternativet. Da vil du se noe slikt som dette:

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

2. Marker linjen

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

trykk **e** og bytt ut `hdX` med `sdX` (der X er bokstaven a, b, c or d avhengig av ditt system). I dette eksempelet blir linjen:

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

Trykk så **Enter** for å lagre endringene. Om andre linjer inneholder `hdX`, så endrer du disse også. Ikke rediger den linja som inneholder `root (hd0,0)`. Når du er ferdig med alle dine endringer, så trykk **b** (b for engelsk boot), og ditt system burde starte som normalt.

3. Når ditt system har startet opp, så må du løse problemet permanent. Gå til Section 4.8.1, og velg en av de to foreslåtte løsningene.

4.8.2.2 Løsning 2

Start systemet fra et Debian GNU/Linux installasjonsmedium (CD/DVD) og velg `rescue` for å starte opp i redningsmodus. Velg språk, sted og tastaturopsett og la systemet selv sette opp nettverket (det spiller ingen rolle om det klarer det eller ikke). Etter en stund blir du spurt om hvilken partisjon du ønsker å bruke som rotfilssystem. De forskjellige alternativene burde se om som dette:

```
/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
```

Hvis du vet hvilken partisjon som er ditt rotfilssystem, så velg den. Hvis du ikke vet hvilken som er ditt rotfilssystem, så prøver du den første på lista, om den ikke fungerer, så går du til neste på lista, helt til du finner den riktige. Det skader ikke å prøve seg fram på denne måten, og om du bare har et rotfilssystem, så burde dette gå raskt. Om du har flere rotfilssystem installert på dine harddisker, så er det nok best å vite nøyaktig hvilken partisjon som er korrekt.

Når du har valgt korrekt partisjon så får du valget mellom en del alternativ. Velg det alternativet som innebærer å starte et skall på den valgte partisjonen, hvis dette ikke går, så velg en annen partisjon.

Nå bør du ha fått opp et skall som brukeren `root` på ditt rotfilssystem montert på `/target`. Du trenger tilgang til innholdet i `/boot`, `/sbin` og `/usr`, som nå burde befinne seg montert under `/target/boot`, `/target/sbin` og `/target/usr`. Hvis disse katalogene trenger å bli montert fra andre partisjoner, så gjør det (hvis du ikke vet hvilke partisjoner det er snakk om, så ta en titt i `/etc/fstab`).

Gå til avsnittet Section 4.8.1 og bruk en av de to foreslåtte løsningene for å fikse problemet permanent. Når du er ferdig, så skriver du `exit` for å kommet ut av skallet, deretter velger du `Start systemet på nytt` for å starte opp på vanlig måte (ikke glem å fjerne `cd'en`).

4.8.2.3 Løsning 3

1. Start opp ditt system med din favoritt Live-cd distribusjon, slik som Debian Live, Knoppix eller Ubuntu Live.
2. Monter partisjonen der `/boot` katalogen finnes. Hvis du ikke vet hvilken det er, så bruk kommandoen **dmesg** for å finne ut om disken din heter `hda`, `hdb`, `hdc`, `hdd` eller `sda`, `sdb`, `sdc`, `sdd`. Når du vet hva disken din heter, for eksempel `sdb`, så kan du bruke denne kommandoen for å finne ut hva slags partisjoner som er på den, og finne den riktige partisjonen: **fdisk -l /dev/sdb**
3. Hvis du har montert riktig partisjon på `/mnt`, og denne partisjonen faktisk inneholder `/boot` katalogen, så skal du redigere filen `/mnt/boot/grub/menu.lst`.

Finn avsnittet som ligner på dette:

```
## ## 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

```

og bytt ut alle hda, hdb, hdc, hdd med sda, sdb, sdc, sdd, der det passer seg. Ikke bytt ut linjer som ligner på:

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

4. Start systemet omigjen, ta ut den kjørklare Live-cd'en, og ditt system burde starte opp på normal måte.
5. Når systemet ditt har startet opp, så gjennomfører du en av de foreslåtte metodene på Section 4.8.1 for å løse problemet permanent.

4.9 Forberedelser for neste versjon

Etter oppgraderingen er det flere ting du kan gjøre for å forberede deg for neste versjon, den etter lenny.

- Om den nye kjernepakken metapakke ble dratt inn som en avhengighet på den gamle, så vil den bli markert som automatisk installert, dette bør fikses med:

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

- Fjern avleggse og ubrukte pakker, som beskrevet i Section 4.10. Du bør granske hvilke konfigurasjonsfiler de bruker, og vurderer å fjerne pakken fullstendig for å også fjerne konfigurasjonsfilene

4.10 Avleggse pakker

lenny introduserer flere tusen nye pakker, men over to tusen pakker som var med i etch har blitt pensjonert eller gjort avleggs og har blitt fjernet. Det finnes ingen støtte for å oppgradere disse pakkene. Selv om ingenting hindrer deg i å bruke pakker som har blitt gjort avleggs, så kommer Debian til å slutte med å tilby sikkerhetsoppgraderinger av disse pakkene normalt et år etter utgivelsen av lenny⁵, og vil normalt ikke tilby annen støtte i denne tiden. Å erstatte dem med tilgjengelige alternativ, om det finnes, anbefales på det sterkeste.

Det finnes mange grunner for at en pakke har blitt fjernet fra distribusjonen: de vedlikeholdes ikke lenger av original utvikleren; det finnes ingen Debian-utvikler som er interessert i å vedlikeholde pakken; funksjonaliteten de tilbyr har blitt erstattet av en annen pakke (eller en nyere versjon); eller pakken ansees ikke for å være passende for lenny. I det siste tilfellet, så kan den fremdeles finnes i distribusjonen, i 'unstable'-distribusjonen.

Det er lett å identifisere hvilke pakker som er 'avleggse' ettersom pakkebehandlerne vil merke dem som nettopp det. Hvis du bruker **aptitude**, så ser du disse pakkene under 'Foreldede pakker og pakker som er laget lokalt'. **dselect** tilbyr en lignende seksjon, men listen som vises kan være annerledes.

Om du har brukt **aptitude** for å manuelt installere pakker i etch, så vil den holde orden på disse pakkene og markere de som blir avleggse når en pakke som trengte dem blir fjernet. **aptitude**, til forskjell fra **debtorphan**, vil ikke markere manuelt installerte pakker som avleggse, i motsetning til de pakker som ble installert gjennom en avhengighet.

Det finnes flere verktøy for å finne avleggse pakker, noen av dem er **debtorphan**, **debfooster** eller **cruft**. **debtorphan** anbefales, selv om den (som standard) bare rapporterer avleggse biblioteker: pakker i 'libs' eller 'oldlibs' som ikke lenger brukes av noen pakker. Ikke stol blindt på resultatet fra disse

⁵ Eller så lenge det ikke kommer noen ny utgivelse i den tidsperioden. Normalt så støttes bare to stabile versjoner samtidig.

verktøyene, spesielt hvis du benytter deg av mye ikke-standard valg. Det anbefales sterkt at du manuelt går igjennom listen med pakker som foreslås fjernet.

Debian Bug Tracking System (<http://bugs.debian.org/>) inneholder ofte mer informasjon om hvorfor en pakke har blitt fjernet. Du bør lese de arkiverte feilrapportene for selve pakken, samt de for pseudopakken på [ftp.debian.org pseudo-package](http://ftp.debian.org/pseudo-package) (<http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes>).

The list of obsolete packages includes:

- apache (1.x), etterfølgeren er 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-pakker

Noen pakker fra etch har blitt delt opp i flere pakker i lenny, ofte har det blitt gjort for å gjøre vedlikeholdet av dem lettere. For å gjøre oppgraderingen lettere for slike tilfeller, så brukers det såkalte 'dummy'-pakker: tomme pakker som har samme navn som pakken i etch hadde, men med de riktige avhengighetene slik at de nye pakkene blir installert. Disse 'dummy'-pakkene kan ansees som overflødige, og kan etter oppgraderingen trygt fjernes.

Beskrivelsen for de fleste (men ikke alle) dummy-pakkene inneholder deres formål. Pakkebeskrivelsene for dummy-pakker er ikke enhetlige, du kan bruke **deborphan** med `--guess` valget for å finne slike pakker på ditt system. Merk at noen dummy-pakker ikke skal fjernes etter oppgraderingen, men at de har som funksjon å holde orden på versjonsnummerene på pakker over tid.

4.11 Planer for neste Debian versjon

4.11.1 Støtte for ARM EABI-porteringen istedenfor ARM-ABI-porteringen

Debian Lenny har to forskjellige og inkompatible ARM-porteringer, den eldre ABI-porteringen (arm) og den nyere EABI-porteringen (armel). Debian Lenny kommer til å bli den siste Debian versjonen med støtte for ARM ABI-porteringen, de kommende versjonen vil kun ha støtte for ARM EABI, eller armel. Derfor bør man bruke armel for nye installasjoner.

Bortsett fra for Netwinder, så finnes det ARM-installasjonsmedia for både arm og armel i Lenny. Det er bare støtte for Netwinder i arm, og den kommer til å bli borte ettersom arm-porteringen forsvinner etter Lenny.

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

Chapter 5

Problemer du bør kjenne til vedrørende lenny

5.1 Potensielle problemer

Noen ganger så vil endringer ha uønskede sideeffekter, eller de vil avdekke feil andre steder. Vi dokumenterer her alle slike problemer som vi kjenner til. Vennligst les avsnittet om skrivefeil, dokumentasjon for den aktuelle pakken, feilrapporter og annen informasjon nevnt i Section 6.1.

5.1.1 Problemer med enheter i tilknytting til udev

Selv om `udev` har blitt testet grundig, så kan du oppleve mindre problemer med noen enheter som må fikses. Det vanligste er å måtte endre rettigheter og/eller eierskap på en enhet. I noen tilfeller så vil en enhet som standard ikke lages, f.eks `/dev/video` og `/dev/radio`

`udev` tilbyr mekanismer for å enkelt utføre disse konfigurasjonene. Se `udev(8)` og `/etc/udev` for mer informasjon.

5.1.2 Noen program vil ikke fungere med kjerne 2.4

Noen program som følger med lenny vil ikke virke med kjerne 2.4, ofte fordi de krever `epoll()` støtte, som ikke er tilgjengelig i kjerne 2.4. Slike program vil enten bare delvis virke, eller ikke virke i det hele tatt før systemet har blitt startet med en 2.6 kjerne.

Et slikt eksempel er HTTP proxy `squid`.

5.1.3 Noen nettverkssteder kan ikke nås via TCP

Siden 2.6.17 kjernen så brukes vindusstørrelsen i TCP veldig aktivt i henhold til RFC 1323. Noen servere har feil innstillinger, og ber om feil vindusstørrelsen. Les mer om dette i [#381262](http://bugs.debian.org/381262) (<http://bugs.debian.org/381262>), [#395066](http://bugs.debian.org/395066) (<http://bugs.debian.org/395066>), [#401435](http://bugs.debian.org/401435) (<http://bugs.debian.org/401435>).

Det fins vanligvis to måter å komme seg rundt dette problemet: enten ved å endre den største tillatte vindusstørrelsen til en lavere verdi (anbefalt måte), eller ved å skru av TCP vindusstørrelse skallering (fases ut, bør ikke brukes). Se eksempel på kommandoer i [debian-installer skrivefeil](http://www.debian.org/devel/debian-installer/errata) (<http://www.debian.org/devel/debian-installer/errata>).

5.1.4 Automatisk avskruing slutter å virke

På en del eldre systemer så kan ikke kommandoen `shutdown -h` brukes for å skru av systemet lengre (systemet stopper bare opp). Dette skjer fordi APM må brukes på disse systemene. Ved å legge til `acpi=off apm=power_off` til kjernens kommandolinje i innstillingene i pakken `grub` eller `lilo` så vil de skru seg ordentlig av. Se feilrapport [#390547](http://bugs.debian.org/390547) (<http://bugs.debian.org/390547>) for mer informasjon.

5.1.5 Asynkron initiering av nettverket kan forårsake uforutsette konsekvenser

På system som bruker `udev` for å laste drivere for nettverkskort, så er det mulig at den asynkrone måten som `udev` kjører på, gjør at nettverkskortet ikke har blitt aktivert før `/etc/init.d/networking` starter under oppstart. Selv om det å legge til `allow-hotplug` i fila `/etc/network/interfaces` (i tillegg til `auto`) vil sørge for at nettverket kommer opp så snart nettverkskortet er aktivert, så er det ingen garanti for at dette skjer før oppstartsekvensen er over, en del tjenester som starter under oppstartsekvensen kan bli påvirket av dette.

5.1.6 Problemer med å bruke trådløse nettverk beskyttet med WPA

I `etch` var pakken `wpa_supplicant` installert som en systemtjeneste med innstillinger lagret i `/etc/default/wpa_supplicant` og med brukerspesifikke innstillinger lagret i `/etc/wpa_supplicant.conf`.

I `lenny`, har `/etc/init.d/wpa_supplicant` forsvunnet, og Debian pakken er nå integrert med `/etc/network/interfaces`, på samme måte som f.eks pakken `wireless-tools`. Dette betyr at `wpa_supplicant` ikke lenger tilbyr noen systemtjeneste.

For mer informasjon om hvordan man skal konfigurere `wpa_supplicant`, så les `/usr/share/doc/wpa_supplicant/README.modes.gz`, der finner du eksempler på ulike oppsett av `/etc/network/interfaces`. Oppdatert informasjon på hvordan du skal bruke `wpa_supplicant` pakken i Debian finner du på [Debian Wiki](http://wiki.debian.org/WPA) (<http://wiki.debian.org/WPA>).

5.1.7 Problemer i filnavn med tegn som ikke er ASCII

Montering av filsystemtypene `vfat`, `ntfs` eller `iso9660` som inneholder tegn utenfor ASCII-tabellen fører til at feil oppstår når disse filene brukes, om man ikke har montert filsystemet med opsjonen `utf8`. Følgende melding kan være et slikt eksempel: 'Invalid or incomplete multibyte or wide character'. En mulig løsning er å bruke `defaults,utf8` som en opsjon ved montering av filsystemtypene `vfat`, `ntfs` eller `iso9660` om disse inneholder filnavn med tegn fra utenfor ASCII-tabellen.

Merk at Linux-kjernen ikke har støtte for å ikke ta hensyn til store eller små bokstaver i filnavn for `vfat` filsystem når alternativet `utf8` brukes.

5.1.8 Lyd stopper å virke

I noen tilfeller så kan det hende at lydsystemet ikke virker etter oppgraderingen. Hvis dette er tilfellet, så se om denne sjekklisten for ALSA løser ditt problem:

- Kjør `alsacnf` som brukeren `root`
- Legg brukeren til i gruppa `audio`
- Husk å skru opp volumet på lydkortet, og aktiver det ved å ta bort `mute` (bruk `alsamixer`)
- Pass på at `arts` og `esound` ikke er aktive
- Pass på at ingen OSS moduler er lastet
- Pass på at høyttalerene faktisk er skrudd på
- Sjekk om kommandoen

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

or the command

```
speaker-test
```

virker for brukeren `root`.

5.2 NFS monteringer håndters nå av nfs-common

Fra og med versjon 2.13 av `util-linux` så håndteres ikke lenger NFS-monteringer av `util-linux`, men av `nfs-common`. Fordi ikke alle systemer har NFS-monteringer, og for å unngå at portmap installeres unødig, så bare anbefaler pakken `util-linux` at pakken `nfs-common` skal installeres. Hvis du trenger å montere NFS-monteringer, så pass på at du har pakken `nfs-common` installert. Installasjonsskriptet til pakken `mount` sjekker hvorvidt NFS-monteringer finnes, og avbryter hvis fila `/usr/sbin/mount.nfs` fra `nfs-common` ikke er tilstede, eller hvis pakken `nfs-common` er utdatert. Enten så må du oppgradere `nfs-common` eller avmontere alle NFS-monteringer før du oppgraderer pakken `mount`.

5.3 Endringer i rumensk(ro) tastaturoppsett

På grunn av oppgraderingen av `xkb-data` til versjon 1.3 i lenny så vil standard rumensk (ro) tastatur oppsett nå lage det korrekte tegnet `șț` (med komma under) istedenfor `șț` (med cedilj under). Noen konfigurasjonsvarianter har fått nytt navn, selv om de gamle navnene fortsatt virker, så oppfordres det å oppdatere `/etc/X11/xorg.conf` med de nye navnene. Mer info om mulig problem på grunn av denne endringen finnes på [wiki \(kun på rumensk\)](http://wiki.debian.org/L10N/Romanian/Lenny/Notes) (<http://wiki.debian.org/L10N/Romanian/Lenny/Notes>).

5.4 Oppgradere apache2

Standardinnstillingene til `apache2` har endret seg litt, noe som kan medføre at du manuelt på gjøre noen endringer i din konfigurasjon. De viktigste endringene er:

`NameVirtualHost *` har blitt endret til `NameVirtualHost *:80`. Om du har lagt til flere navnbaserte virtuelle verter, så må du endre `<VirtualHost *>` til `<VirtualHost *:80>` for dem alle.

`Apache` sin `User` og `Group` og stien til `PidFile` er nå konfigurert i `/etc/apache2/envvars`. Hvis du har endret disse verdiene fra sin standardverdi, så må du endre denne fila. Dette betyr at du ikke kan starte `apache2` omigjen med kommandoen `apache2 -k start`, men at du nå må bruke `/etc/init.d/apache2` eller `apache2ctl`.

Hjelpeprogrammet `suexec` som trengs for `mod_suexec` er nå pakket i en egen pakke, `apache2-suexec`, denne blir som standard ikke installert.

Modulspesifikke innstillinger har blitt flyttet fra `/etc/apache2/apache2.conf` til `/etc/apache2/mods-available/*.conf`.

For mer informasjon se `/usr/share/doc/apache2.2-common/NEWS.Debian.gz` og `/usr/share/doc/apache2.2-common/README.Debian.gz`.

5.5 NIS og Network Manager

Versjonen av `yppbind` som finnes i `nis` i lenny inneholder støtte for Network Manager. Denne støtten for Network Manager innebærer at `yppbind` skruer av NIS-klientfunksjonalitet når Network Manager rapporterer at datamaskinen er uten nettverk. Siden Network Manager ofte rapporterer at datamaskinen er uten nettverk når nettverket ikke brukes, så bør NIS-brukere med NIS-klientsystemer sørge for at støtten for Network Manager er skrudd av.

Dette kan gjøres enten ved å fjerne pakken `network-manager` eller ved å redigere fila `/etc/default/nis` ved å legge til `-no-dbus` til `YPBINDARGS`.

Bruken av `-no-dbus` er standard for nye installasjoner av Debian, men det var ikke slik i tidligere versjoner.

5.6 Sikkerhetsstatus for Mozillaprodukter

Mozillaprogrammene `firefox`, `thunderbird`, og `sunbird` (som har fått andre navn i Debian, henholdsvis `iceweasel`, `icedove`, og `iceowl`,) er viktige for mange brukere. Dessverre så er hovedforfatterenes holdning angående sikkerhetsoppdateringer, at brukerne hele tiden skal oppgradere til siste versjon av disse produktene, denne holdningen er ikke forenelig med Debian sine regler for sikkerhetsoppdateringer, som sier at man ikke skal blande sammen sikkerhetsoppdateringer med større

funksjonsendringer i programvaren. Vi kan ikke si med sikkerhet i dag, men det kan komme til å skje i løpet av levetiden til lenny at Debian Security Team kommer til den konklusjon at det ikke lenger er mulig å tilby sikkerhetsoppgraderinger for Mozilla produktene. Du bør ta høyde for dette når du implementerer bruken av disse Mozilla produktene, og vurdere andre alternativer som er tilgjengelige i Debian, om du er opptatt av å kunne ha sikkerhetsoppgraderinger.

`iceape`, versjonen av `seamonkey` uten varemerker, har blitt tatt ut av lenny (med unntak av noen nødvendige bibliotekspakker).

5.7 KDE skrivbordet

Det er ikke noen større endringer i skrivebordsmiljøet KDE. Lenny inneholder en serviceversjon av KDE, som blant annet inneholder oppdaterte oversettelser, versjonen er en blanding av KDE 3.5.9 og KDE 3.5.10. Noen moduler er merket som versjon 3.5.9, men de har endringer inkludert for å passe sammen med versjon 3.5.10. I all hovedsak så er det versjon 3.5.10 som følger med Lenny, men uten endringene i `kdebase` for `kicker`, og noen feilfikser i `kdepim`.

Lenny kommer til å være den siste stabile utgaven som inneholder KDE3.

5.8 Støtte og endringer i Gnome skrivebordet

Det har skjedd mange forandringer i skrivebordsmiljøet GNOME side den versjonen som ble levert med `etch`. Du finner mer informasjon på [GNOME 2.22 Release Notes](http://library.gnome.org/misc/release-notes/2.22/) (<http://library.gnome.org/misc/release-notes/2.22/>).

5.9 Ingen støtte for Unicode som standard 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) (<http://bugs.debian.org/419490>). Consider switching to `emacs22`, `emacs22-gtk`, or `emacs22-nox`.

5.10 slurpd/replica vil ikke lenger fungere

OpenLDAP har fra versjone 2.4.7 ikke lenger støtte for LDAP replikering via `slurpd`. Eksisterende konfigurasjoner må tilpasses den nye LDAP Sync Replication engine (`syncrepl`). Mer informasjon finnes på <http://www.openldap.org/doc/admin24/replication.html> (<http://www.openldap.org/doc/admin24/replication.html>).

5.11 Skrivebordet bruker ikke hele skjermen

Driveren for Intel Mobile GM965 kan feilaktig oppdage en VGA utgang, og derfor sette for lav størrelse på skjermen for å kompensere for dette. Symptomet på dette er at bare deler av skjermen blir brukt av vindusbehandleren. Du kan rette opp denne feilen ved å legge til disse linjene i fila `/etc/X11/xorg.conf`:

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

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

5.12 DHCP problemer med feiltoleranse

Når et par med servere kjører som DHCP-servere, så må servernavnene være konsekvente, ellers krasjer DHCP

Please see bug [#513506](http://bugs.debian.org/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.

Chapter 6

Mer informasjon om Debian GNU/Linux

6.1 Ytterligere lesing

I tillegg til disse Utgivelsesnotatene og Installasjonsguiden, så er mer dokumentasjon tilgjengelig om Debian GNU/Linux fra Debian Documentation Project (DDP), som har som mål å lage dokumentasjon av høy kvalitet for brukere og utviklere av Debian. Dokumentasjon, som inkluderer Debian Reference, Debian New Maintainers Guide, og Debian FAQ er tilgjengelig. Besøk siden til DDP på [DDP nettside](http://www.debian.org/doc/ddp) (<http://www.debian.org/doc/ddp>).

Dokumentasjonen for de enkelte pakkene blir installert sammen med pakken, og plassert på `/usr/share/doc/pakkenavn`. Denne kan inneholde informasjon om opphavsrett, spesifikke Debian detaljer og eventuell dokumentasjon fra hovedutviklerene.

6.2 Få hjelp

Det finnes mange kilder for å få hjelp, råd og støtte for brukere av Debian, men disse bør først brukes etter at man har forsøkt å finne hjelp i dokumentasjonen som finnes. Dette avsnittet nevner kort noen av disse kildene som nye brukere av Debian kan gå til for å få hjelp, råd og støtte.

6.2.1 E-postlister

De e-postlistene som er mest interessante for Debian brukere er nok `debian-user list` (engelsk) og de andre `debian-user-language` e-postlistene. For mer informasjon om disse e-postlistene og hvordan man abonnerer på dem gå til <http://lists.debian.org/>. Sjekk alltid e-postliste arkivet for eventuelle svar på dine problemer, og husk å følge e-postlist etiketten.

6.2.2 Internet Relay Chat

Debian har en egen IRC kanal på OFTC IRC nettverket. Bruk din favoritt IRC-klient og gå til `irc.debian.org` og inn i `#debian`.

Følg kanalens retningslinjer. Retningslinjene er tilgjengelige på [Debian Wiki](http://wiki.debian.org/DebianIRC) (<http://wiki.debian.org/DebianIRC>).

For mer informasjon om OFTC, gå til [website](http://www.oftc.net/) (<http://www.oftc.net/>).

6.3 Rapportere feil

Vi jobber hardt for at Debian GNU/Linux skal være et operativsystem med høy kvalitet, det betyr ikke at alle våre pakker er feilfrie. I tråd med Debian sin 'åpne utviklingsmodell', og som en tjeneste til alle våre brukere, så ligger alle feilrapporter åpent tilgjengelig i vårt eget feilrapporteringsssystem (BTS). BTS er søkbar på <http://bugs.debian.org/>.

Hvis du finner en feil i distribusjonen, eller i en av pakkene, så vennligst lag en feilrapport så dette kan rettes opp i fremtidige versjoner. Du trenger en gyldig e-postadresse for å sende inn feilrapporter.

Vi trenger din e-postadresse så våre utviklere kan komme i kontakt med deg for mulige oppfølgende spørsmål.

Du kan enkelt sende inn feilrapporter ved å bruke programmet **reportbug**, eller manuelt som en vanlig e-post. Du kan lese mer om feilrapporteringsystemet og hvordan du kan bruke det ved å lese `/usr/share/doc/debian`, hvis du har pakken `doc-debian` installert. Eller du kan lese om det på **Bug Tracking System** (<http://bugs.debian.org/>).

6.4 Bidra til Debian

Alle kan bidra til Debian. Ved å hjelpe andre bruker med å svar på de forskjellige e-postlistene på **e-postlistene** (<http://lists.debian.org/>) så er yter du et viktig bidrag. Ved å sende inn feilrapporter til **send inn feilrapporter** (<http://bugs.debian.org/>) og bistå utviklerene med å rette dem opp, så er du med på å opprettholde den høye kvaliteten på Debian. Hvis du er flink til å skrive, så vil du kanskje vurdere å skrive **dokumentasjon** (<http://www.debian.org/doc/ddp>), eller kanskje hjelpe til med å **oversette** (<http://www.debian.org/international/>) den dokumentasjon som allerede finnes til ditt eget språk.

Hvis du kan sette av litt mer tid, så kunne du f.eks vedlikeholde en Debian-pakke. Spesielt hjelpsomt er det om du kunne adoptere, eller ta ansvar for en pakke som noen ønsker skal inkluderes inn i Debian. Nettsiden **Work Needing and Prospective Packages database** (<http://www.debian.org/devel/wnpp/>) inneholder en oversikt over slike pakker. Hvis du har en spesiell interesse, så kan du bidra til en av Debians mange underprosjekter, som **Debian Jr.** (<http://www.debian.org/devel/debian-jr/>) og **Debian Med** (<http://www.debian.org/devel/debian-med/>).

Uansett, om du arbeider med fri programvare på en eller annen måte, som en vanlig bruker, som programmerer, forfatter eller oversetter, så er du allerede en viktig bidragsyter. Å kunne bidra er belønnende og morsomt, samtidig som det lar deg møte nye mennesker og den gir deg den gode varme følelsen i kroppen.

Appendix A

Forberedelser for oppgradering av et etch system

Dette tillegget inneholder informasjon om hvordan du kontrollerer at du faktisk kan installere eller oppgradere pakker på etch versjonen, før du oppgraderer til lenny. Dette er kun nødvendig i helt spesifikke situasjoner.

A.1 Oppgradering av ditt etch system

Det er ingen større forskjeller i forhold til tidligere oppgraderinger du har gjort på etch. Den eneste forskjellen er at du nå må passe på at dine pakkearkiv refererer til etch, som beskrevet i Section [A.2](#).

Hvis du benytter deg av et Debian speil, så vil du automatisk bli oppgradert til den siste punktutgivelsen av etch.

A.2 Kontroller dine arkivlister

Hvis noen av linjene i `/etc/apt/sources.list` refererer til 'stable', så bruker du i praksis 'allerede' lenny. Hvis du allerede har brukt `apt-get update` så kan du fortsatt rette det opp igjen uten problemer om du følger oppskriften som kommer her.

Om du allerede har begynt å installere pakker fra lenny, så er det antakelig ingen grunn lenger til å installere pakker fra etch. Du må selv bestemme om du vil fortsette eller ikke. Det er mulig å nedgradere pakker, men det er ikke beskrevet her.

Åpne fila `/etc/apt/sources.list` med din favoritt tekstbehandler (som `root`) og kontroller alle linjer som begynner med enten `deb http:` eller `deb ftp:`, hvis noen av dem refererer til 'stable', så bytt ut `stable` med `etch`.

Hvis du har noen linjer som begynner med `deb file:`, så må du selv finne ut om det sted til referer til inneholder et arkiv for etch eller lenny.

IMPORTANT



Ikke endre noen linjer som begynner med `deb cdrom:`. Hvis du gjør det, så blir det en feil på den linja, og du må kjøre kommandoen **apt-cdrom** omigjen for å rette det opp. Bli ikke forvirret om du ser en 'cdrom'-linje som referer til 'unstable', der er ofte slik for CD-ROM.

Om du har gjort noen endringer, så lagrer og avslutter du, og kjører deretter:

```
# apt-get update
```

for å oppdatere pakkelisten

Appendix B

Bidragstere til Utgivelsenotatene

Mange har hjulpet med å skrive dette Utgivelsenotat, blant andre

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 Bilenlein, Simon Paillard, Stefan Fritsch, Steve Langasek, Tobias Scherer, Vincent McIntyre, og W. Martin Borgert.

Dette dokument har blitt oversatt til flere språk. Mange takk til oversetterene.

Norsk bokmål oversettelse: Klaus Ade Johnstad

Appendix C

Til minne om Thiemo Seufer

Debianprosjektet har mistet et viktig medlem. Thiemo Seufer, døde i en bilulykke 26. desember 2008.

Thiemo var involvert i Debian på mange måter, han vedlikeholdte flere pakker, og var hovedsponsoren for porteringen av Debian til MIPS prosessorvarianter. Han var også medlem av linuxkjerne-laget til Debian, samt medlem av Debian Installer laget. Hans bidrag strak seg lengre enn til bare Debian prosjektet. Han jobbet også med en portering av Linux kjernen til MIPS, emuleringen av MIPS i qemu, samt mange andre mindre prosjekter.

Thiemos arbeid, hans hengivelse, hans brede tekniske innsikt og hans evne til å dele dette med andre vil bli sårt savnet. Thiemos bidrag vil ikke bli glemt. Den høye standarden på hans arbeid gjør det hardt å fortsette dette arbeidet.

For å hedre Thiemos bidrag til Debian dedikeres denne utgaven av Debian GNU/Linux 5.0 'lenny' til han.

Appendix D

Glossary

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

Index

A

Abiword, 6
Apache, 6

B

BIND, 6
Blu-ray, 6
brukere med nedsatt syn, 10

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

IcedTea, 8

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, 27
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, 4, 15–17, 19
 aptitude, 6, 18, 19
 base-config, 19
 bind, 27
 bind9, 27
 dlatex, 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, 6, 31
 iceowl, 31
 iceweasel, 6, 31
 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

T

Thunderbird, 6

Tomcat, 6

U

Unicode, 32

V

virtualization, 8

VServer, 8

W

WPA, 30

X

Xfce, 6