

Debian-pikareferenssi

Osamu Aoki <osamu\#at\#debian.org>
Suomenkielinen käännös: Esko Arajärvi <edu\#at\#iki.fi>
'Tekijät' sivulla **23**

CVS, to 18.1.2007, 11:54:40 UTC

Tiivistelmä

Debian-pikareferenssi (<http://qref.sourceforge.net/>) on tarkoitettu lyhyeksi johdatukseksi, **pikaohjeeksi**, Debian-järjestelmään. Se on lyhennelmä dokumentista Debian-referenssi (<http://qref.sourceforge.net/>).

Tekijänoikeuksista

Copyright © 2001–2005 by Osamu Aoki <osamu#at#debian.org>.

This document may be used under the terms of the GNU General Public License version 2 or higher. (<http://www.gnu.org/copyleft/gpl.html>) (Tätä dokumenttia saa käyttää GNU GPL -lisenssin version 2 tai uudemman ehtojen mukaisesti. Katso epävirallinen suomenos (http://www.turre.com/licenses/gpl_fi.php).

Permission is granted to make and distribute verbatim copies of this document provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this document under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this document into another language, under the above conditions for modified versions, except that this permission notice may be included in translations approved by the Free Software Foundation instead of in the original English.

Sisältö

1 Johdanto	1
1.1 Dokumentin käytännöt	1
1.2 Debian-jakeluiden perusteet	1
2 Päivittäminen jakeluun <i>stable</i>, <i>testing</i> tai <i>unstable</i>	3
2.1 Päivittäminen Potatosta Woodyyn	3
2.2 Päivitykseen valmistautuminen	3
2.3 Päivittäminen	4
2.3.1 <i>dselect</i> -ohjelman käyttö	4
3 Debianin pakettienhallinta	5
3.1 Esittely	5
3.1.1 Tärkeimmät pakettienhallintatyökalut	6
3.1.2 Aputyökalut	6
3.2 Debianin pakettien hallinnan aloitus	6
3.2.1 APT:n laittaminen käyttövalmiiksi	6
3.2.2 Tehtävien asennus	7
3.2.3 <i>aptitude</i>	7
3.2.4 <i>dselect</i>	8
3.2.5 Jakeluiden seuraaminen APT:lla	9
3.2.6 Komennot <i>aptitude</i> , <i>apt-get</i> ja <i>apt-cache</i>	9
3.3 Debianin selviytymiskomennot	11
3.3.1 Tarkista Debianin virheet ja etsi apua	11
3.3.2 APT upgraden vianetsintä	11

3.3.3	Pelastus käyttäen ohjelmaa <code>dpkg</code>	12
3.3.4	Pakettien valintatietojen palauttaminen	12
3.3.5	Järjestelmän pelastaminen hakemiston <code>/var</code> tuhoutumisen jälkeen	13
3.3.6	Paketin asentaminen käynnistyskelvottomana järjestelmään	13
3.3.7	Mitä tehdä, jos <code>dpkg</code> -komento on hajalla	14
3.4	Debianin nirvana-komennot	14
3.4.1	Tietoa tiedostosta	14
3.4.2	Tietoa paketista	15
3.4.3	Valvottoman asennus APT:lla	16
3.4.4	Asennettujen pakettien asetusten päivittäminen	16
3.4.5	Pakettien poistaminen ja siivoaminen	17
3.4.6	Vanhempien pakettien säilyttäminen	17
3.4.7	<code>stable/testing/unstable</code> -sekajärjestelmä	17
3.4.8	Pakettitiedostojen karsiminen välimuistista	18
3.4.9	Järjestelmäasetusten tallennus/kopiointi	18
3.4.10	Paketin sovittaminen <code>stable</code> -järjestelmään	18
3.4.11	Paikallinen pakettiarkisto	19
3.4.12	Vieraiden binääripakettien muuntaminen tai asentaminen	20
3.4.13	Automaattiasennus-komento	20
3.4.14	Asennettujen pakettitiedostojen varmennus	21
3.5	Muita Debianin omituisuuksia	21
3.5.1	<code>dpkg-divert</code> -komento	21
3.5.2	<code>equivs</code> -paketti	21
3.5.3	Vaihtoehtoiset komennot	22
3.5.4	Ajotasojen käyttö	22
3.5.5	Pois päältä kytketyt taustapalvelut	22
A	Liite	23
A.1	Tekijät	23
A.2	Takuut	25
A.3	Palaute	25

Luku 1

Johdanto

Tämä dokumentti oli alun perin “pikareferenssi”, mutta se kasvoi. Joka tapauksessa johtavia ajatuksia ovat lyhyys ja yksinkertaisuus.

1.1 Dokumentin käytännöt

Tämä Debian-pikareferenssi esittää ohjeet lyhyinä `bash`-komentotulkin komentoina.

Viitteet

- UNIX-tyylisiin **ohjesivuihin** on esitetty muodossa: `bash(1)` ja
- GNU **TEXINFO-sivuihin** muodossa: `info libc`.

1.2 Debian-jakeluiden perusteet

Debian ylläpitää kolmea eri jakelua yhtäaikaisesti. Nämä ovat:

- `stable` — Hyödyllisin tuotantopalvelimilla, koska sitä päivitetään vain tietoturvapäivityksillä.
- `testing` — Suositeltu jakelu työasemille, koska se sisältää työpöytäohjelmien tuoreita, mutta jonkin verran testattuja versioita.
- `unstable` — Uusinta uutta. Debianin kehittäjien valinta.

Kun `unstable`-jakelun paketeista ei ensimmäisen noin viikon aikana löydetä julkaisukriittisiä virheitä, ne siirretään automaattisesti `testing`-jakeluun.

Debian-jakeluilla on myös koodinimet. Ennen Sargen julkaisua kesäkuussa 2005, kolme jakelua olivat Woody (`stable`), Sarge (`testing`) ja Sid (`unstable`). Sargen julkaisun jälkeen kolme jakelua olivat vastaavasti Sarge, Etch ja Sid. Kun Etch julkaistaan, `stable` ja `unstable` julkaisut

ovat Etch ja Sid. Tällöin luodaan uusi `testing`-jakelu (aluksi se on kopio `stable`:sta) ja sille annetaan uusi koodinimi.

Jos haluat sähköpostia Debiania koskevista tärkeistä tiedotuksista, liity hiljaiselle sähköpostilistalle `debian-devel-announce@lists.debian.org`.

Jos haluat käyttää paketeista uudempia versioita kuin niitä, jotka julkaistiin julkaistaessa käyttämäsi jakelua, voit joko päivittää järjestelmäsi uudempaan jakeluun kappaleen 'Päivittäminen jakeluun `stable`, `testing` tai `unstable`' seuraavalla sivulla ohjeiden mukaan tai päivittää vain valitsemasi paketit. Jos pakettia ei saa helposti päivitettyä, saatat haluta rakentaa paketin uudelleen (backport) kappaleen 'Paketin sovittaminen `stable`-järjestelmään' sivulla [18](#) ohjeiden mukaan.

Luku 2

Päivittäminen jakeluun `stable`, `testing` tai `unstable`

2.1 Päivittäminen Potatosta Woodyyn

Tämä prosessi on kuvattu erikseen, koska Potaton APT-ohjelmassa ei ole kaikkia ohjesivulla `apt_preferences(5)` kuvattuja toiminnallisuuksia.

Lisättyäsi Woodyyn lähteet tiedostoon `/etc/apt/sources.list`, päivitä APT ja vaaditut peruspaketit Woody-versioihin seuraavilla komennoilla:

```
# apt-get update
# apt-get install libc6 perl libdb2 debconf
# apt-get install apt apt-utils dselect dpkg
```

Päivitä tämän jälkeen järjestelmän loppuosa Woodyyn.

```
# apt-get upgrade
# apt-get dist-upgrade
```

2.2 Päivitykseen valmistautuminen

Voit päivittää jakelusta toiseen hakemalla paketit verkon yli seuraavasti.

Hanki puhdas lista `stabilen:n` säilytyspaikoista:

```
# cd /etc/apt
# cp -f sources.list sources.list.old
# :>sources.list
# apt-setup noprobe
```

Jos haluat päivittää `testing`-jakeluun, lisää `testing`:in lähteet tähän uuteen listaan. Jos haluat päivittää `unstable`-jakeluun, lisää myös `unstable:n` lähteet listaan.

```
# cd /etc/apt
# grep -e "^deb " sources.list >srcs
# :>sources.list
# cp -f srcs sources.list
# sed -e "s/stable/testing/" srcs >>sources.list
# sed -e "s/stable/unstable/" srcs >>sources.list
# apt-get update
# apt-get install apt apt-utils
```

Kappale ‘Debianin pakettien hallinnan aloitus’ sivulla 6 kertoo tiedostojen `/etc/apt/sources.list` ja `/etc/apt/preferences` virittämisestä.

2.3 Päivittäminen

Muokattuasi tiedostoja `/etc/apt/sources.list` ja `/etc/apt/preferences` ylläolevien ohjeiden mukaan voit aloittaa päivityksen.

Huomaa, että Debianin `testing`-jakelun käyttäminen saattaa sivuvaikutuksena aiheuttaa sen, että tietoturvapäivityksiä sisältävien pakettien asennus viivästyy, koska tällaiset paketit ladataan jakeluun `unstable` ja ne vasta myöhemmin siirtyvät jakeluun `testing`.

Katso kappaleesta ‘Debianin pakettienhallinta’ viereisellä sivulla perusteet ja katso kappaletta ‘APT upgrade:n vianetsintä’ sivulla 11, jos törmäät ongelmiin.

2.3.1 `dselect`-ohjelman käyttö

Jos järjestelmässä on useita paketteja, mukaan lukien `-dev`-paketteja jne., on suositeltavaa käyttää ohjelmaa `dselect` seuraavasti pakettien tarkempaan hallintaan.

```
# dselect update # Aja tämä aina ennen päivitystä
# dselect select # Valitse lisää paketteja
```

Kaikki nykyiset paketit valitaan, kun `dselect` käynnistyy. `dselect` saattaa kysyä lisäpakettien asentamisesta perustuen asennettujen pakettien `Depends-`, `Suggests-` ja `Recommends-` riippuvuuksiin. Jos et halua asentaa lisää paketteja, kirjoita `Q` poistuaksesi ohjelmasta.

```
# dselect install
```

Sinun täytyy vastata joihin pakettien asetuksiin liittyviin kysymyksiin tämän prosessin aikana, joten pidä muistiinpanosi lähellä ja varaa aikaa tämän osan suorittamiseen. Katso ‘`dselect`’ sivulla 8.

Käytä ohjelmaa `dselect`. **Se toimii aina. :)**

Luku 3

Debianin pakettienhallinta

`aptitude` on nykyään suositeltu tekstikäyttöliittymä ohjelman APT (Advanced Package Tool) käyttöön. Se pitää erikseen kirjaa tarkoituksella asennetuista paketeista ja riippuvuuksien takia asennetuista paketeista. Jälkimmäiset `aptitude` poistaa automaattisesti, kun mikään tarkoituksella asennettu paketti ei enää riipu niistä. Se sisältää kehittyneitä pakettien suodatusominaisuuksia, mutta niiden käyttäminen voi olla vaikeaa.

`synaptic` on nykyään suositeltu graafinen käyttöliittymä ohjelman APT käyttöön. Sen pakettien suodatusominaisuuksia on helpompi käyttää kuin `aptituden`. Se tukee koeluontoisesti Debianin pakettimerkkejä (<http://debtags.alioth.debian.org/>).

Debianin palvelimien verkkokuorman vähentämiseksi ja latausten nopeuttamiseksi tulisi paketit hakea Debianin peilipalvelimilta.

Jos sama paketti on tarpeen asentaa useammalle koneelle paikallisessa verkossa, voit ohjelmalla `squid` perustaa paikallisen HTTP-välityspalvelimen APT:n kautta ladatuille paketeille. Tarvittaessa aseta arvo ympäristömuuttujalle `http_proxy` tai aseta arvo `http` tiedostossa `/etc/apt/apt.conf`.

Vaikka osiossa `apt_preferences` (5) kuvattu APT:n valintajärjestelmä on tehokas, sen vaikutukset saattavat olla vaikeasti ymmärrettäviä. Se on luonteeltaan edistynyt ominaisuus.

Ohjelman `chroot` käyttö on suositeltavaa The use of `chroot` is desirable for järjestelmän vakauden turvaamiseksi ja uusimpien ohjelmaversioiden saamiseksi samanaikaisesti.

Tämä luku pohjaa Woodyn jälkeiseen järjestelmään. Jotkin ominaisuudet saattavat vaatia Sarge-järjestelmän tai myöhemmän.

3.1 Esittely

Jos kaiken kehittäjille tarkoitetun dokumentaation lukeminen tuntuu liian paljolta, lue tämä luku ja pääset nauttimaan Debianin `testing/unstable`-jakeluiden täydestä voimasta. :-)

3.1.1 Tärkeimmät pakettienhallintatyökalut

<code>dpkg</code>	- Debian pakettitiedostojen asennusohjelma
<code>apt-get</code>	- Komentorivikäyttöliittymä APT-ohjelmalle
<code>aptitude</code>	- Kehittynyt teksti- ja komentorivikäyttöliittymä APT:lle
<code>synaptic</code>	- Graafinen käyttöliittymä APT:lle
<code>dselect</code>	- Menupohjainen pakettienhallintaohjelma
<code>tasksel</code>	- Tehtävienasennusohjelma

Nämä ohjelmat eivät kaikki ole toistensa vaihtoehtoja. Esimerkiksi `dselect` käyttää sekä APT:ia että `dpkg:tä`.

APT käyttää tiedostoja `/var/lib/apt/lists/*` saatavilla olevien pakettien seuraamiseen kun taas `dpkg` käyttää tiedostoa `/var/lib/dpkg/available`. Jos paketteja on asennettu `aptitude`:lla tai muilla APT:n käyttöliittymillä ja halutaan asentaa paketteja `dselect`:llä, tulee ensin päivittää tiedosto `/var/lib/dpkg/available` valitsemalla `[U]pdate dselect:n` valikoista (tai ajaa "`dselect update`").

`apt-get` automaattisesti asentaa kaikki paketit, joista valittu paketti riippuu. Se ei asenna paketteja, joita ainoastaan suositellaan tai ehdotetaan.

`aptituden` asetuksissa voidaan puolestaan määrätä paketin suosittelemat tai ehdottamat paketit asennettaviksi.

`dselect` Näyttää käyttäjälle listan paketeista, joita valittu paketti suosittelee tai ehdottaa ja antaa käyttäjän valita asennettavat näistä vapaasti.

3.1.2 Aputyökalut

<code>dpkg-reconfigure</code>	- muuttaa jo asennetun paketin asetuksia (jos se käyttää <code>debconf:ia</code>)
<code>dpkg-source</code>	- hallinnoi lähdekoodipaketteja
<code>dpkg-buildpackage</code>	- automatisoi pakettien rakentamisen
<code>apt-cache</code>	- tutkii paikallista pakettiarkistoa

3.2 Debianin pakettien hallinnan aloitus

3.2.1 APT:n laittaminen käyttövalmiiksi

Muokkaa tiedostoa `sources.list` kappaleen 'Päivitykseen valmistautuminen' sivulla 3 ohjeiden mukaan.¹

¹Jos seuraat `testing-` tai `unstable-`jakeluita Voit poistaa viittaukset jakeluun `stable` tiedostoista `/etc/apt/sources.list` ja `/etc/apt/preferences` koska `testing` on aluksi `stable:n` kopio.

3.2.2 Tehtävien asennus

Paketteja on mahdollista asentaa ryhmittäin siten, että yhden ryhmän paketteja tyypillisesti tarvitaan Debian-järjestelmän tietynlaiseen käyttöön. Tällaisia pakettiryhmiä kutsutaan "tehtäviksi".

Helpoin tapa asentaa tehtäviä on käyttää alkuasennuksen aikana ohjelmaa `tasksel`. Komento

```
dselect update
```

on ajettava ennen sen käyttöä.

Ohjelmalla `aptitude` voi myös asentaa tehtäviä ja se on suositeltava työkalu. Sitä käytettäessä on mahdollista poistaa yksittäisiä paketteja tehtävistä ennen niiden asennusta.

3.2.3 `aptitude`

`aptitude` on uusi menu-pohjainen pakettien asennusohjelma. Se on samankaltainen kuin `dselect`, mutta tehty alusta asti uudelleen APT:n päälle. Sitä voidaan käyttää `apt-get:n` sijaan useimpien komentojen osalta. Katso `aptitude(1)` ja `/usr/share/doc/aptitude/README`.

Kun `aptitude` otetaan käyttöön, on parempi jatkaa sen käyttämistä pakettien asentamiseen vaihtoehtoisten metodien sijaan. Muuten menetetään hyöty, joka saadaan `aptitude:n` kyvystä pitää kirjaa tarkoituksella asennetuista paketeista.

Käytettäessä `aptitude:a` kokoruututilassa, sille voidaan antaa yhden kirjaimen komentoja, jotka yleensä ovat pieniä kirjaimia. Tärkeimmät komennot ovat:

Näppäily	Toiminto
F10	Valikko
?	Apua näppäinkomennoissa (täydellinen listaus)
u	Päivitä pakettiarkiston tiedot
+	Merkitse paketti päivitettäväksi tai asennettavaksi
-	Merkitse paketti poistettavaksi (säilytä asetukset)
_	Merkitse paketti hävitettäväksi (poista asetukset)
=	Merkitset paketti säilytettäväksi
U	Merkitset päivitettäväksi kaikki paketit, jotka voidaan päivittää
g	Lataa ja asenna merkityt paketit
q	Poistu nykyisestä näkymästä ja tallenna muutokset
x	Poistu nykyisestä näkymästä ja hylkää muutokset
Enter	Näytä tietoa paketista
C	Näytä paketin muutosloki
l	Vaihda näytettävien pakettien rajoitusta
/	Hae seuraava osuma
\	Toista edellinen haku

Kuten `apt-get`, `aptitude` asentaa paketit, joista valittu paketti on riippuvainen. `aptitude` tarjoaa myös mahdollisuuden suositeltujen ja ehdotettujen pakettien lataamiseen ja asentamiseen. Voit vaihtaa oletustoimintaa valitsemalla valikosta `F10 -> Options -> Dependency handling`.

Muita `aptituden` etuja ovat:

- `aptitude` tarjoaa asennettavaksi paketin kaikki versiot.
- `aptitude` pitää lokia toiminnastaan tiedostossa `/var/log/aptitude`.
- `aptitude` helpottaa vanhentuneiden ohjelmien löytämistä listaamalla ne osiossa "Obsolete and Locally Created Packages".
- `aptitude` tarjoaa tehokkaan järjestelmän tiettyjen pakettien etsimiseen ja näytettyjen pakettien rajoittamiseen. Ohjelman `mutt` tuntevat käyttäjät oppivat nämä nopeasti, koska `mutt` inspiroi käytettyä lausekesyntaksia. Katso osiota "SEARCHING, LIMITING, AND EXPRESSIONS" dokumentissa `/usr/share/doc/aptitude/README`.
- `aptitude:n` kokoruututilassa on ohjelman `su` toiminnallisuus upotettuna ja ohjelmaa voidaan ajaa tavallisena käyttäjänä kunnes todella tarvitaan ylläpitäjän oikeuksia.

3.2.4 `dselect`

Vakaissa julkaisuissa Potatoon asti `dselect` oli pääasiallinen pakettienhallintatyökalu. Sargessa kannattaisi sen sijaan käyttää ohjelmaa `aptitude`.

Käynnistettäessä `dselect` automaattisesti valitsee "Required"-, "Important"- ja "Standard"-paketit (vaadittavat, tärkeät ja vakiopaketit).

`dselect:ssä` on jossain määrin outo käyttöliittymä. Useimmat käyttäjät kuitenkin tottuvat siihen. Siinä on neljä komentoa (Iso kirjain tarkoittaa ISOA!):

Näppäily	Toiminto
Q	Lopeta. Vahvista nykyiset valinnat ja lopeta joka tapauksessa. (kumoa riippuvuudet)
R	Palauta! En tarkoittanut sitä.
D	Toimi! <code>dselectin</code> mielipiteellä ei ole väliä.
U	Laita kaikki suositeltuun tilaan.

Komentojen `D` ja `Q` kanssa on mahdollista tehdä ristiriitaisia valintoja omalla vastuulla. Näitä komentoja tulisi käyttää varoen.

Rivin "expert" lisääminen tiedostoon `/etc/dpkg/dselect.cfg` vähentää ohjelman tuloksia.

Jos `dselect` toimii hitaasti jollain koneella, ohjelma voidaan ajaa nopeammalla koneella asennettavien pakettien määrittämiseksi ja käyttää sen jälkeen komentoa `apt-get install hitaalla koneella pakettien asentamiseen`.

3.2.5 Jakeluiden seuraaminen APT:lla

testing-jakelun seuraaminen sen muuttuessa tapahtuu muuttamalla tiedoston `/etc/apt/preferences` sisältö seuraavanlaiseksi:

```
Package: *
Pin: release a=testing
Pin-Priority: 800
```

```
Package: *
Pin: release a=stable
Pin-Priority: 600
```

testing-jakelun seuraamisen sivuvaikutuksena tietoturvapäivityksiä sisältävien pakettien asennus saattaa viivästyä. Tämä johtuu siitä, että paketit ladataan ensin jakeluun `unstable` ja ne siirtyvät testing-jakeluun viiveellä.

Ohjesivu `apt_preferences(5)` sisältää monimutkaisempia esimerkkejä esimerkiksi siitä kuinka seurata testing:iä, mutta samalla asentaa valittuja paketteja `unstable:sta`.

Esimerkkejä tiettyjen pakettien lukitsemisesta tiettyyn versioon, samalla kun muita paketteja päivitetään, löytyy `examples`-alihakemistosta (<http://www.debian.org/doc/manuals/debian-reference/examples/>) tiedostoista `preferences.testing` ja `preferences.unstable`.

Jos paketteja käytetään sekaisin eri jakeluista esim. `testing:stä` ja `stable:sta` tai `unstable:sta` ja `stable:sta`, päivittyvät peruspaketit kuten `libc6` lähes pakolla jossain vaiheessa `testing-` tai `unstable-`versioihin, jotka saattavat sisältää virheitä. Varoitus on annettu.

Toisena esimerkkinä `preferences.stable` pakottaa kaikki paketit palaamaan `stable-`versioon.

Paketin myöhemmästä versiosta aiempaan palaaminen ei ole virallisesti tuettua Debianissa. Saattaa kuitenkin olla, että aikaisempaan versioon palaaminen on tarpeellista, jos uudempi versio toimii virheellisesti. Aiemmat pakettiversiot löytyvät paikallisesti hakemistosta `/var/cache/apt/archives/` verkosta osoitteesta <http://snapshot.debian.net/>. Katso myös 'Pelastus käyttäen ohjelmaa `dpkg`' sivulla [12](#).

Palaaminen aiempaan **jakeluun** ei myöskään ole tuettua ja aiheuttaa hyvin todennäköisesti ongelmia. Äärimäisessä hätässä tätä voi kuitenkin yrittää viimeisenä ratkaisuna.

3.2.6 Komennot `aptitude`, `apt-get` ja `apt-cache`

Seurattaessa testing:iä yllä olevan esimerkin mukaisesti, voidaan järjestelmää päivittää seuraavilla komennoilla:

- `aptitude update` (tai `apt-get update`)
Nämä päivittävät listan pakettivarastoista saatavilla olevista paketeista.
- `aptitude upgrade` (tai `apt-get upgrade` tai `aptitude dist-upgrade` tai `apt-get dist-upgrade`)
Nämä seuraavat `testing`-jakelua — ne päivittävät vuorollaan järjestelmän jokaisen paketin asennettuaan `testing`-jakelusta muista paketeista sellaiset versiot, joista kyseinen paketti on riippuvainen.²
- `apt-get dselect-upgrade`
Tämä seuraa `testing`-jakelua — se päivittää kunkin järjestelmän paketin ohjelman `dselect` valintojen mukaisesti.
- `aptitude install paketti/unstable`
Tämä asentaa *paketin* `unstable`-jakelusta, mutta paketit, joista se on riippuvainen, `testing`-jakelusta.
- `aptitude install -t unstable paketti`
Tämä asentaa *paketin* `unstable`-jakelusta samoin kuin paketit, joista se on riippuvainen. Tämä saadaan aikaan asettamalla `unstable`-jakelun Pin-Priority arvoon 990.
- `apt-cache policy foo bar ...`
Tämä tarkistaa pakettien *foo bar ...* tilan.
- `aptitude show foo bar ... | less` (tai `apt-cache show foo bar ... | less`)
Tämä näyttää pakettien *foo bar ...* tiedot.
- `aptitude install foo=2.2.4-1`
Tämä asentaa paketin *foo* version `2.2.4-1`.
- `aptitude install foo bar-`
Tämä asentaa paketin *foo* ja poistaa paketin *bar*.
- `aptitude remove bar`
Tämä poistaa paketin *bar*, mutta ei sen asetustiedostoja.
- `aptitude purge bar`
Tämä poistaa paketin *bar* ja kaikki sen asetustiedostot.

Jos yllä olevissa esimerkeissä ohjelmalle `apt-get` annetaan valitsin `-u`, se listaa kaikki päivitettävät paketit ja pyytää käyttäjältä varmistuksen ennen päivitystä. `aptitude` tekee näin oletuksena. Tämän voi asettaa `apt-get`:n oletukseksi seuraavasti:

²Komentojen `upgrade` ja `dist-upgrade` erot näkyvät vain kun pakettien uusien versioiden riippuvuussuhteet ovat eri kuin pakettien vanhempien versioiden. Yksityiskohdat löytyvät ohjesivulta `apt-get(8)`. Käskyt `aptitude upgrade` ja `aptitude dist-upgrade` käynnistävät `aptitude`:n komentorivivilassa. Voit vaihtaa kokoruututilaan painamalla `e`-näppäintä.

```
$ cat >> /etc/apt/apt.conf << .
// Näytä aina lista päivitettävistä paketeista (-u)
APT::Get::Show-Upgraded "true";
.
```

Valitsimen `--no-act` avulla on mahdollista simuloida toimintaa asentamatta, poistamatta, jne. mitään paketteja oikeasti.

3.3 Debianin selviytymiskomennot

Tämä tieto tuo mukanaan ikuisten päivitysten onnen. :-)

3.3.1 Tarkista Debianin virheet ja etsi apua

Jos tietyn paketin kanssa ilmenee ongemia, tulisi ennen avun etsimistä tai virheen ilmoittamista tarkistaa seuraavat sivut. (`lynx`, `links` ja `w3m` toimivat kaikki):

```
$ lynx http://bugs.debian.org/
$ lynx http://bugs.debian.org/paketti # jos paketin nimi on tiedossa
$ lynx http://bugs.debian.org/virhenumero # jos virhenumero on tiedossa
```

Etsi Googlessa (www.google.com) hakusanoilla, joihin sisältyy "site:debian.org".

Epävarmassa tilanteessa kannattaa lukea ohjeita. Aseta `CDPATH` seuraavasti:

```
export CDPATH=./usr/local:/usr/share/doc
```

ja kirjoita

```
$ cd paketti
$ pager README.Debian # jos tiedosto on olemassa
$ mc
```

3.3.2 APT upgrade:n vianetsintä

Päivitettäessä paketteja `unstable:ssa` tai `testing:ssä` kappaleen 'Päivittäminen' sivulla 4 ohjeiden mukaan, saattaa riippuvuuksissa ilmetä ongelmia. Useimmiten tämän johtuu siitä, että päivitettävä paketti on riippuvainen paketista, joka ei ole vielä saatavilla. Ongelman voi korjata ajamalla komennon:

```
# aptitude dist-upgrade
```

Jos tämä ei auta, toista jompaa kumpaa seuraavista komennoista kunnes ongelma ratkeaa:

```
# aptitude -f upgrade          # jatka päivittämistä myös virheistä huolimatta
... tai
# aptitude -f dist-upgrade     # jatka päivittämistä ja tarvittaessa poistamista
```

Jotkin pahasti hajalla olevat päivitysskriptit saattavat aiheuttaa jatkuvia ongelmia. Tällöin kannattaa yleensä tutkia ja korjata kyseisen paketin skriptejä `/var/lib/dpkg/info/paketti.{post,pre}{inst,rm}` ja tämän jälkeen ajaa:

```
# dpkg --configure -a         # tekee kaikkien osittain asennettujen pakettien asennus
```

Jos skripti valittaa puuttuvasta asetustiedostosta, etsi hakemistosta `/etc/` vastaavaa tiedostoa. Jos hakemistosta löytyy muuten samanniminen tiedosto päätteellä `.dpkg-new` (tai jotain vastaavaa), poista päätte siirtämällä tiedosto toiselle nimelle komennolla `mv`.

Pakettien riippuvuussuhteissa voi ilmetä ongelmia asennettaessa niitä `unstable`:ssa tai `testing`:ssä. Riippuvuuksia voi kiertää seuraavasti.

```
# aptitude -f install paketti # unohda puuttuvat riippuvuudet
```

Vaihtoehtoisesti tilanteen voi korjata käyttämällä pakettia `equivs`. Katso `/usr/share/doc/equivs/README.Debian`.

3.3.3 Pelastus käyttäen ohjelmaa dpkg

Jos APT:n käyttäminen johtaa umpikujaan, paketit voi ladata Debianin peilipalvelimilta ja asentaa ohjelmalla `dpkg`. Jos käytettävissä ei ole verkkoyhteyttä, pakettien väliaikaisia kopioita voi etsiä hakemistosta `/var/cache/apt/archives/`.

```
# dpkg -i fetchmail_6.2.5-4_i386.deb
```

Jos todella tarvitun paketin asennus tällä tavalla epäonnistuu riippuvuussuhdeongelmien takia, niiden tarkistusta voi ohjata mm. valitsimilla `--ignore-depends` ja `--force-depends`. Katso yksityiskohtat kappaleesta `dpkg(8)`.

3.3.4 Pakettien valintatietojen palauttaminen

Jos tiedosto `/var/lib/dpkg/status` korruptoituu mistään syystä, Debian-järjestelmä menettää tiedon valituista paketeista ja kärsii pahasti. Vanhempi versio tiedostosta `/var/lib/dpkg/status` voi löytyvä nimellä `/var/lib/dpkg/status-old` tai `/var/backups/dpkg.status.*`.

Hakemiston `/var/backups/` pitäminen eri levyosiolla voi olla hyvä idea, koska tämä hakemisto sisältää paljon tärkeää järjestelmätietoa.

Jos vanhaa versiota tiedostosta `/var/lib/dpkg/status` ei ole saatavilla, voidaan tieto palauttaa hakemistossa `/usr/share/doc/` olevista hakemistoista.

```
# ls /usr/share/doc | \
  grep -v [A-Z] | \
  grep -v '^texmf$' | \
  grep -v '^debian$' | \
  awk '{print $1 " install"}' | \
  dpkg --set-selections
# dselect --expert # asentaa järjestelmän uudelleen, poista tarpeettomat
```

3.3.5 Järjestelmän pelastaminen hakemiston `/var` tuhoutumisen jälkeen

Koska hakemisto `/var` sisältää usein päivitettävää tietoa, kuten sähköposteja, se korruptoituu helpommin kuin esim. `/usr/`. Hakemiston `/var` sijoittaminen omalle levyosiolleen vähentää riskejä. Jos tapahtuu katastrofi, on `/var/` ehkä rakennettava uudelleen, jotta Debian-järjestelmä voidaan pelastaa.

Hajonneeseen järjestelmään voidaan kopioida toisesta toimivasta, minimaalisesta Debian-järjestelmästä hakemisto `/var`, esimerkiksi `var.tar.gz` (<http://people.debian.org/~osamu/pub/>), kunhan lähdehakemisto on samasta tai vanhemmasta Debianin versiosta. Tämän jälkeen

```
# cd /
# mv var var-old      # jos jäljellä on mitään käyttökelpoista
# tar xvzf var.tar.gz # puretaan hakemisto Woody-jakelusta
# aptitude           # tai dselect
```

Tämän jälkeen järjestelmän pitäisi toimia. Pakettitietojen palauttamista voidaan edistää kappaleen 'Pakettien valintatietojen palauttaminen' viereisellä sivulla ohjeiden avulla. ([FIXME]: This procedure needs more experiments to verify.)

3.3.6 Paketin asentaminen käynnistyskelvottomana järjestelmään

Linux käynnistetään Debianin pelastuslevykkeeltä tai `-CD:`ltä tai toiselta osiolta Linuxissa, jossa on monikäynnistys. Käynnistyskelvoton järjestelmä liitetään hakemistoon `/target` ja asennetaan paketti käyttäen `dpkg:n` `chroot-asennusmoodia`.

```
# dpkg --root /target -i pakettitiedosto.deb
```

Tämän jälkeen tehdään asetukset ja korjataan ongelmat.

Jos käynnistysongelmat johtuvat vain hajonneesta `lilo`:sta, voi järjestelmän käynnistää Debianin standardilta pelastuslevykkeeltä. Jos Linux-asennuksen juuriosio on `/dev/hda12` ja halutaan käynnistää ajotasolle 3, kirjoitetaan käynnistyskehotteessa:

```
boot: rescue root=/dev/hda12 3
```

Käynnistyvä järjestelmä on lähes täysin toimiva ja käyttää ytimenään levykkeellä olevaa ydintä. (Pieniä häiriöitä voi esiintyä johtuen puuttuvista ytimen ominaisuuksista tai moduleista.)

3.3.7 Mitä tehdä, jos `dpkg`-komento on hajalla

Minkään `.deb`-paketin asentaminen saattaa olla mahdotonta, jos `dpkg` on hajonnut. Tilanteen voi korjata seuraavalla proseduurilla. (Ensimmäisellä rivillä voit korvata komennon "links" haluamallasi selainkomennolla.)

```
$ links http://http.us.debian.org/debian/pool/main/d/dpkg/  
... lataa toimiva dpkg_versio_ark.deb  
$ su  
password: *****  
# ar x dpkg_versio_ark.deb  
# mv data.tar.gz /data.tar.gz  
# cd /  
# tar xzfv data.tar.gz
```

Alustalla `i386` voidaan käyttää myös URL:ia `http://packages.debian.org/dpkg`.

3.4 Debianin nirvana-komennot

Näiden komentojen ymmärtämistä seuraava **valaistuminen** päästää käyttäjän ikuisesta pahan karman päivityshelvetistä ja päästää hänet Debianin **nirvanaan**. :-)

3.4.1 Tietoa tiedostosta

Asennetut paketit, joihin liittyy tiedosto, jonka nimi on tiettyä muotoa, voi löytää komennolla:

```
$ dpkg {-S|--search} lauseke
```

Tai vastaavasti voi etsiä Debianin arkistoista:

```
$ wget http://ftp.us.debian.org/debian/dists/sarge/Contents-i386.gz
$ zgrep -e lauseke Contents-i386.gz
```

Tai voi käyttää erikoistuneita pakettikomentoja:

```
# aptitude install dlocate
$ dlocate tiedostonimi # Komentojen dpkg -L ja dpkg -S nopea vaihtoehto
...
# aptitude install auto-apt # työkalu, joka asentaa paketteja tarvittaessa
# auto-apt update          # luo tietokannan auto-apt:lle
$ auto-apt search lauseke # etsii lauseketta sekä asennetuista että asentamat
```

3.4.2 Tietoa paketista

Tietoja voidaan etsiä pakettiarkistoista. Ennen tätä tulee varmistaa, että APT käyttää oikeita arkistoja muokkaamalla tiedostoa `/etc/apt/sources.list`. Paketin nykyisen version vertailu jakeluiden `testing` ja `unstable` versioihin onnistuu komennolla `apt-cache policy`

```
# apt-get check          # Päivitä välimuisti ja etsi rikkoutuneita pakett
$ apt-cache search lauseke # Etsi paketin tekstikuvauksesta
$ apt-cache policy paketti # Paketin tärkeys-/jakelutiedot
$ apt-cache show -a paketti # Näytä paketin kaikkien jakeluiden kuvaukset
$ apt-cache showsrc paketti # Näytä vastaavan lähdekoodipaketin kuvaus
$ apt-cache showpkg paketti # Pakettitietoja virheiden korjaukseen
# dpkg --audit|-C       # Etsi osittain asennettuja paketteja
$ dpkg {-s|--status} paketti ... # Asennetun paketin kuvaus
$ dpkg -l paketti ...     # Asennetun paketin satuts status (1 rivi/paketti
$ dpkg -L paketti ...    # Listaa paketin asentamat tiedostot
```

`apt-cache showsrc` -komento is ole Woody-jakelusta lähtien ole ollut dokumentoitu, mutta se toimii. :-)

Pakettitietoja löytyy myös seuraavista tiedostoista (näitä voi selata esim ohjelman `mc` avulla):

```
/var/lib/apt/lists/*
/var/lib/dpkg/available
```

Seuraavat tiedostot kertovat mitä oikeastaan on tapahtunut muutaman edellisen asennusession aikana.

```
/var/lib/dpkg/status
/var/backups/dpkg.status*
```

3.4.3 Valvomaton asennus APT:lla

Valvomattoman asennuksen voi tehdä lisäämällä seuraavan rivin tiedostoon `/etc/apt/apt.conf`:

```
Dpkg::Options {"--force-confold";}
```

Tämä vastaa komennon `aptitude -y install paketti` tai `apt-get -q -y install paketti` ajamista. Koska kaikkiin kysymyksiin vastataan automaattisesti "kyllä", tämä voi aiheuttaa ongelmia ja sitä pitäisi käyttää varoen. Katso `apt.conf(5)` and `dpkg(1)`.

Minkä tahansa paketin asetuksia voidaan myöhemmin muuttaa kappaleen 'Asennettujen pakettien asetusten päivittäminen' tällä sivulla ohjeiden mukaan.

3.4.4 Asennettujen pakettien asetusten päivittäminen

Asennettujen pakettien asetuksia voidaan muuttaa seuraavilla komennoilla

```
# dpkg-reconfigure --priority=medium paketti [...]
# dpkg-reconfigure --all # päivitä kaikkien pakettien asetukset
# dpkg-reconfigure locales # luo mahdolliset uudet maa-asetukset
# dpkg-reconfigure --p=low xserver-xfree86 # päivitä X serverin asetukset
```

Jos on tarvetta muuttaa ohjelman `debconf` toimintatilaa pysyvästi, se voidaan tehdä tällä.

Joissain ohjelmissa on erillisiä asetusten tekoskriptejä.³

```
apt-setup      - luo /etc/apt/sources.list
install-mbr    - asenna pääkäynnistyslohkon hallinnoija
tzconfig       - aseta paikallinen aikavyöhyke
gpmconfig      - muokkaa gpm-hiiritaustaohjelman asetuksia
eximconfig     - muokkaa Exim (MTA):n asetuksia
texconfig      - muokkaa teTeX:n asetuksia
apacheconfig   - muokkaa Apache:n (httpd) asetuksia
cvsconfig      - muokkaa CVS:n asetuksia
sndconfig      - muokkaa äänentoistojärjestelmän asetuksia
...
update-alternatives - aseta oletuskomentoja esim., vim on vi
update-rc.d      - System-V init skriptien hallinta
update-menus     - Debian-menujärjestelmä
...
```

³Jotkin `*config` skriptit ovat katoamassa uudemmassa Sarge-julkaisussa ja toiminnallisuudet siirretään `debconf-järjestelmään`.

3.4.5 Pakettien poistaminen ja siivoaminen

Paketin poisto niin, että sen asetustiedostot säilytetään:

```
# aptitude remove paketti ...
# dpkg --remove paketti ...
```

Paketin ja asetustiedostojen poistaminen:

```
# aptitude purge paketti ...
# dpkg --purge paketti ...
```

3.4.6 Vanhempien pakettien säilyttäminen

Esimerkiksi pakettien `libc6` ja `libc6-dev` nykyisten versioiden säilyttäminen ajettaessa `dselect` tai `aptitude install paketti` voidaan tehdä seuraavasti:

```
# echo -e "libc6 hold\nlibc6-dev hold" | dpkg --set-selections
```

`aptitude install paketti`-komentoa ei tämä "säilytys" haittaa. Säilyttääksesi paketin pakottamalla automaattisen paluun aikaisempaan versioon ajettaessa `aptitude upgrade paketti` tai `aptitude dist-upgrade`, lisää seuraava tiedostoon `/etc/apt/preferences`:

```
Package: libc6
Pin: release a=stable
Pin-Priority: 2000
```

"Package:"-kohdassa ei voi olla "`libc6*`"-tyylisiä määritelmiä. Jos kaikki lähdekoodipakettiin `glibc` liittyvät binääripaketit halutaan pitää samassa versiossa, ne on listattava kaikki erikseen.

Seuraava komento listaa tällä hetkellä säilytettävät paketit:

```
dpkg --get-selections "*" | grep -e "hold$"
```

3.4.7 `stable/testing/unstable`-sekajärjestelmä

`apt-show-versions`-komennolla voidaan listata paketin saataville olevat versiot jakeluittain.

```
$ apt-show-versions | fgrep /testing | wc
... montako pakettia järjestelmässä on testing:stä
$ apt-show-versions -u
... päivitettävissä olevat paketit
$ aptitude install `apt-show-versions -u -b | fgrep /unstable`
... päivitä kaikki unstable-paketit uusimpiin versioihin
```

3.4.8 Pakettitiedostojen karsiminen välimuistista

APT jättää paketteja asennettaessa pakettitiedostot välimuistiin hakemistoon `/var/cache/apt/archives/` ja se täytyy puhdistaa.

```
# aptitude autoclean # poistaa hyödyttömät pakettitiedostot
# aptitude clean     # poistaa kaikki talletetut pakettitiedostot
```

3.4.9 Järjestelmäasetusten tallennus/kopiointi

Paikallisen kopion tekeminen pakettien valintatiloista:

```
# dpkg --get-selections "*" >valinnat # tai käytä \*
# debconf-get-selections > debconfsel.txt
```

"*" sisällyttää tiedostoon `valinnat` myös "siivotut" paketit.

Tämä tiedosto voidaan siirtää toiselle tietokoneella ja käyttää asennukseen siellä:

```
# dselect update
# debconf-set-selections < debconfsel.txt
# dpkg --set-selections <valinnat
# apt-get -u dselect-upgrade # tai dselect install
```

3.4.10 Paketin sovittaminen stable-järjestelmään

Tehtäessä `stable`-järjestelmän osittaista päivitystä, paketin rakentaminen uudelleen lähdekoodipaketista nykyisessä ympäristössä on suositeltavaa. Tällä vältetään riippuvuuksista johdettava massiivinen pakettien päivitys. Seuraavat rivit on ensin lisättävä tiedostoon `/etc/apt/sources.list`:

```
deb-src http://http.us.debian.org/debian testing \
main contrib non-free
deb-src http://http.us.debian.org/debian unstable \
main contrib non-free
```

Tulostussyistä kukin `deb-src`-rivi on jaettu kahdelle riville, mutta tiedostossa `sources.list` kaiken tulisi olla yhdellä rivillä.

Tämän jälkeen haetaan lähdekoodi ja tehdään paikallinen paketti:

```
$ apt-get update # päivitä lista lähdekoodipaketeista
$ apt-get source paketti
$ dpkg-source -x paketti.dsc
$ cd paketti-versio
... myös paketin vaatimat paketit (Build-Depends .dsc-tiedostossa)
    tulee asentaa. Myös paketti "fakeroot" tarvitaan.

$ dpkg-buildpackage -rfakeroot

...tai (ei allekirjoitusta)
$ dpkg-buildpackage -rfakeroot -us -uc # käytä myöhemmin "debsign"-ohjelmaa,

...Sitten asennus
$ su -c "dpkg -i pakettitiedosto.deb"
```

Yleensä vain muutama "-dev"-päätteinen paketti tarvitaan riippuvuussuhteiden tyydyttämiseen. `debsign` on paketissa `devscripts`. `auto-apt` voi auttaa riippuvuuksien tyydyttämisessä. `fakeroot`-ohjelman käyttö välttää tarpeetonta pääkäyttäjätunnuksen käyttöä.

In Woody, these dependency issues can be simplified. For example, to compile a source-only pine package:

```
# apt-get build-dep pine
# apt-get source -b pine
```

3.4.11 Paikallinen pakettiarkisto

APT:n ja `dselect`-järjestelmän kanssa yhteensopivan paikallisen pakettiarkiston luominen vaatii, että luodaan `Packages`-tiedosto ja että pakettitiedostot talletetaan tietynlaiseen hakemistopuuhun.

Paikallinen, virallisten Debian-arkistojen kaltainen `deb`-varasto voidaan tehdä seuraavasti:

```
# aptitude install dpkg-dev
# cd /usr/local
# install -d pool # fyysiset paketit sijaitsevat täällä
# install -d dists/unstable/main/binary-i386
# ls -l pool | sed 's/_.*$/ priority section/' | uniq > override
# editor override # määritä priority ja section
# dpkg-scanpackages pool override /usr/local/ \
```

```
> dists/unstable/main/binary-i386/Packages
# cat > dists/unstable/main/Release << EOF
Archive: unstable
Version: 3.0
Component: main
Origin: Local
Label: Local
Architecture: i386
EOF
# echo "deb file:/usr/local unstable main" \
  >> /etc/apt/sources.list
```

Vaihtoehtoisesti, nopea karkea versio paikallisesta deb-varastosta voidaan tehdä seuraavasti:

```
# aptitude install dpkg-dev
# mkdir /usr/local/debian
# mv /jos/ssain/paketti.deb /usr/local/debian
# dpkg-scanpackages /usr/local/debian /dev/null | \
  gzip - > /usr/local/debian/Packages.gz
# echo "deb file:/usr/local/debian ./" >> /etc/apt/sources.list
```

Näitä arkistoja voidaan etäkäyttää tarjoamalla pääsy hakemistoihin joko HTTP:llä tai FTP:llä ja vaihtamalla rivejä tiedostossa `/etc/apt/sources.list` tämän mukaisesti.

3.4.12 Vieraiden binääripakettien muuntaminen tai asentaminen

`alien` mahdollistaa Red Hat:n `rpm`-, Stampeden `slp`-, Slackwaren `tgz`- ja Solariksen `pkg`-tiedostomuotojen muuntamisen Debianin `deb`-paketeiksi. Näin ollen on mahdollista käyttää `alien`-ohjelmaa toisesta Linux-jakelusta peräisin olevan paketin muuntamiseen haluamaasi pakettiformaattiin ja paketin asentamiseen järjestelmään. `alien` tukee myös LSB-paketteja.

3.4.13 Automaattiasennus-komento

`auto-apt` on tarvittaessa paketteja asentava työkalu.

```
$ sudo auto-apt update
... päivitä tietokanta
$ auto-apt -x -y run
Entering auto-apt mode: /bin/bash
Exit the command to leave auto-apt mode.
$ less /usr/share/doc/med-bio/copyright # käytä tiedostoa, jota ei ole
... Asennetaan tiedoston tarjoava paketti.
... Asennetaan myös riippuvuudet
```


3.4.14 Asennettujen pakettitiedostojen varmennus

debsums mahdollistaa asennettujen pakettitiedostojen varmennuksen MD5-tarkistussummia vastaan. Joillekin paketeille ei ole saatavissa MD5-tarkistussummia. Mahdollinen väliaikainen korjaus ylläpitäjille:

```
# cat >>/etc/apt/apt.conf.d/90debsums
DPkg::Post-Install-Pkgs {"xargs /usr/bin/debsums -sg";};
^D
```

tekijänä Joerg Wendland <joergland@debian.org> (testaamaton).

3.5 Muita Debianin omituisuuksia

3.5.1 dpkg-divert -komento

Tiedostojen **ohjaukset** ovat tapa pakottaa dpkg asentamaan tiedosto muualle kuin sen oletus sijaintiin. Ohjauksia voidaan käyttää Debianin pakettiskripteissä tiedoston siirtämiseen, kun se aiheuttaa törmäyksen. Järjestelmän ylläpitäjät voivat myös käyttää ohjauksia paketin asetustiedoston syrjäyttämiseen tai kun dpkg:n halutaan säilyttävän joitain tiedostoja (joita ei ole merkitty asetustiedostoiksi) asennettaessa uudempaa versiota paketista, joka sisältää kyseiset tiedostot. .

```
# dpkg-divert [--add] filename # lisää "ohjaus"
# dpkg-divert --remove filename # poista "ohjaus"
```

Yleensä ohjelman dpkg-divert käyttöä kannattaa välttää, jos se ei ole aivan välttämätöntä.

3.5.2 equivs-paketti

Käännettäessä ohjelmaa lähdekoodista, on parasta tehdä siitä oikea paikallinen Debian-paketti (*.deb). Käytä equivs-pakettia viimeisenä toivona.

```
Package: equivs
Priority: extra
Section: admin
Description: Circumventing Debian package dependencies
 This is a dummy package which can be used to create Debian
 packages, which only contain dependency information.
 (Debianin pakettiriippuvuuksien kiertäminen
 Tätä valepakettia voidaan käyttää sellaisten Debian-pakettien
 luomiseen, jotka sisältävät vain riippuvuussuhdetietoa.)
```

3.5.3 Vaihtoehdotiset komennot

Komennon `vi` voi määrätä ajamaan ohjelman `vim` ohjelmalla `update-alternatives`:

```
# update-alternatives --display vi
...
# update-alternatives --config vi
  Selection      Command
-----
          1      /usr/bin/elvis-tiny
          2      /usr/bin/vim
*+       3      /usr/bin/nvi
```

Enter to keep the default[*], or type selection number: 2

Debianin vaihtoehtojärjestelmän vaihtoehdot listataan hakemistossa `/etc/alternatives/` symbolisina linkkeinä.

X Window -ympäristön voi asettaa käyttämällä `update-alternatives`:ia tiedostoihin `/usr/bin/x-session-manager` ja `/usr/bin/x-window-manager`.

`/bin/sh` on suora symbolinen linkki tiedostoon `/bin/bash` tai `/bin/dash`. On turvallisempaa käyttää `/bin/bash`:ia, joka on yhteensopiva vanhojen Bashismin saastuttamien skriptien kanssa, mutta POSIX-yhteensopivuuteen vaativan `/bin/dash`:n käyttö pakottaa kurinalaisuuteen. Linuxin 2.4-ytimeen päivittäminen tyypillisesti asettaa linkin tiedostoon `/bin/dash`.

3.5.4 Ajotasojen käyttö

Asennettaessa useimmat Debian-paketit asettavat palvelunsa ajettaviksi ajotasoilla 2-5. Näin ollen ajotasojen 2, 3, 4 ja 5 välillä ei ole eroja mukauttamattomassa Debian-järjestelmässä. Debian jättää paikallisen ylläpitäjän vastuulle ajotasojen mukauttamisen. Tämä poikkeaa tavasta, jolla ajotasoja käytetään joissain muissa suosituista GNU/Linux-jakeluissa. Yksi mahdollisesti haluttu muutos on `xdm`:n tai `gdm`:n poistaminen ajotasolta 2, jolloin X-ikkunointijärjestelmä ei käynnisty käynnistyssekvenssin lopuksi. Se voidaan käynnistää vaihtamalla ajotasolle 3.

3.5.5 Pois päältä kytketyt taustapalvelut

Debianin kehittäjät suhtautuvat vakavasti järjestelmän turvallisuuteen. Monet taustapalvelut asennetaan niin, että vain osa palveluista ja ominaisuuksista ovat käytössä.

Komento `ps aux` ja tiedostojen `/etc/init.d/*` ja `/etc/inetd.conf` sisältö on tarkistettava, jos on mitään epäilyksiä (Exim:stä, DHCP:stä, ...). Myös `/etc/hosts.deny` kannattaa tarkistaa. Myös komento `pidof` on hyödyllinen (katso `pidof(8)`).

X11 ei oletuksena salli (etä-) TCP/IP-yhteyksiä uusimmissa Debianin versioissa. SSH:n X-forwarding (X-yhteyksien edelleenvälittäminen) on myös estetty.

Liite A

Liite

A.1 Tekijät

Debian-pikareferenssin aloitti Osamu Aoki <osamu\#at\#debian.org> henkilökohtaisena asennusmuistiona, jota alettiin lopulta kutsua nimellä "Quick Reference...". Sisällöstä suuri osa on peräisin "debian-user"-sähköpostilistan arkistoista. Lähteinä käytettiin myös dokumentteja "Debian Installation Manual" ja "Debian Release Notes".

Dokumentin kutsumista nimellä "Debian-referenssi" ehdotti Josip Rodin, joka on hyvin aktiivinen Debian-dokumentointiprojektissa (<http://www.debian.org/doc/ddp>) (DDP) ja ylläpitää dokumenttia "The Debian FAQ". Debian-referenssiin liitettiin muutamia dokumentin "The Debian FAQ" lukuja, joiden sisältö on referenssimäinen. Sitten "Debian-pikareferenssi" syntyi dokumentin lyhennelmänä.

Tätä dokumenttia ovat toimittaneet, kääntäneet ja laajentaneet seuraavat QREF-tiimin jäsenet:

- Alkuperäisen "Quick Reference..."-oppaan englanninkieliset originaalit
 - Osamu Aoki <osamu\#at\#debian.org> (johtaja: kaikki sisältö)
- Englannin oikoluku ja muu panostus
 - Esko Arajärvi <edu\#at\#iki.fi> (etch-päivityksiä)
 - Thomas Hood <jdthood\#at\#yahoo.co.uk> (verkkoihin liittyvää)
 - Brian Nelson <nelson\#at\#bignachos.com> (erityisesti X:ään liittyvää)
 - David Sewell <dsewell\#at\#virginia.edu> (lopettanut)
 - Jan Michael C Alonzo <jmalonzo\#at\#spaceants.net>
 - Daniel Webb <webb\#at\#robust.colorado.edu>
 - Palautetta kaikilta kääntäjiltä
- Ranskankielinen käännös
 - Guillaume Erbs <gerbs\#at\#free.fr> (johtaja: fr)
 - Rénaud Casagraude <rcasagraude\#at\#interfaces.fr>
 - Jean-Pierre Delange <adeimantos\#at\#free.fr>
 - Daniel Desages <daniel\#at\#desages.com>
- Italiankielinen käännös
 - Davide Di Lazzaro <mc0315\#at\#mclink.it> (johtaja: it)
- Portugalinkielinen (Brasilia) käännös

- Paulo Rogério Ormenese <pormenese\#at\#uol.com.br> (johtaja: pt-br)
- Andre Luis Lopes <andrelap\#at\#ig.com.br>
- Marcio Roberto Teixeira <marciotex\#at\#pop.com.br>
- Rildo Taveira de Oliveira <to_rei\#at\#yahoo.com>
- Raphael Bittencourt Simoes Costa <raphael-bsc\#at\#bol.com.br>
- Gustavo Noronha Silva <kov\#at\#debian.org> (koordinaattori)
- Espanjankielinen käännös
 - Walter Echarri <wecharri\#at\#infovia.com.ar> (johtaja: es)
 - José Carreiro <ffx\#at\#urbanet.ch>
- Saksankielinen käännös
 - Jens Seidel <tux-master\#at\#web.de> (johtaja: de)
 - Willi Dyck <wdyck\#at\#gmx.net>
 - Stefan Schröder <stefan\#at\#fkip.uni-hannover.de>
 - Agon S. Buchholz <asb\#at\#kefk.net>
- Puolankielinen käännös — PDDP (<http://debian.linux.org.pl>):n seuraavat jäsenet:
 - Marcin Andruszkiewicz
 - Mariusz Centka <mariusz.centka\#at\#debian.linux.org.pl>
 - Bartosz Feński <fenio\#at\#debian.linux.org.pl> (johtaja: pl)
 - Radosław Grzanka <radekg\#at\#debian.linux.org.pl>
 - Bartosz 'Xebord' Janowski
 - Jacek Lachowicz
 - Rafał Michaluk
 - Leonard Milcin, Jr.
 - Tomasz Z. Napierała <zen\#at\#debian.linux.org.pl>
 - Oskar Ostafin <cx\#at\#debian.linux.org.pl>
 - Tomasz Piękoś
 - Jacek Politowski
 - Mateusz Prichacz <mateusz\#at\#debian.linux.org.pl>
 - Marcin Rogowski
 - Paweł Różański
 - Mariusz Strzelecki
 - Krzysztof Ścierański
 - Przemysław Adam Śmiejek <tristan\#at\#debian.linux.org.pl>
 - Krzysztof Szynter
 - Mateusz Tryka <uszek\#at\#debian.linux.org.pl>
 - Cezary Uchto
 - Krzysztof Witkowski <tjup\#at\#debian.linux.org.pl>
 - Bartosz Zapałowski <zapal\#at\#debian.linux.org.pl>
- Kiinankielinen (yksinkertaistettu) käännös
 - Hao "Lyoo" LIU <iamlyoo\#at\#163.net>
 - Ming Hua <minghua\#at\#rice.edu>
 - Xiao Sheng Wen <atzlinux\#at\#163.com> (johtaja: zh-cn)
 - Haifeng Chen <optical.dlz\#at\#gmail.com>
 - Xie Yanbo <xieyanbo\#at\#gmail.com>
 - easthero <easthero\#at\#gmail.com>

- Kiinankielinen (perinteinen) käännös
 - Asho Yeh <asho\#at\#debian.org.tw> (johtaja: zh-tw)
 - Tang Wei Ching <wctang\#at\#csie.nctu.edu.tw> (ex-johtaja: zh-tw)
- Japaninkielinen käännös
 - Shinichi Tsunoda <tsuno\#at\#ngy.1st.ne.jp> (leader: ja)
 - Osamu Aoki <osamu\#at\#debian.org>
- Suomenkielinen käännös
 - Esko Arajärvi <edu\#at\#iki.fi> (johtaja: fi)

A.2 Takuut

Koska en ole ammattilainen, en teeskentele tuntevani täydellisesti Debiania tai Linuxia yleensä. Käyttämäni tietoturvatoinenpiteet saattavat sopia vain kotikäyttöön.

Tämä dokumentti ei korvaa mitään virallisia oppaita.

Kaikesta vastuusta sanoudutaan irti. Kaikki tavaramerkit ovat merkkien omistajien omaisuutta.

A.3 Palaute

Dokumenttia koskevat kommentit ja lisäykset ovat aina tervetulleita. Ole hyvä ja lähetä sähköpostia Debianin virheidenhallintajärjestelmään (<http://bugs.debian.org/>) liittyen pakettiin `debian-reference` tai vastaaviin käännöspaketteihin. Ohjelma `reportbug` tekee perusteellisen virheraportin lähettämisestä helppoa. Voit myös lähettää sähköpostia englanniksi Osamu Aokille (<http://people.debian.org/~osamu/>) osoitteeseen <osamu\#at\#debian.org> tai kullekin kääntäjälle heidän omalla kielellään.