

Slackware -current CHANGES_AND_HINTS.txt

This file documents the instructions for upgrading to Slackware -next, the packages added, removed, renamed, and/or split during the development cycle from Slackware 13.37 through -next, and some potential “gotchas” that users can avoid by arming themselves with a little knowledge.

INSTRUCTIONS FOR UPGRADING FROM 13.37

Follow the instructions detailed in the UPGRADE.TXT located in this directory.

Note that upgrading from a Slackware version earlier than 13.37 is NOT supported at all and will most likely not work.

PACKAGE ADDITIONS SINCE 13.37

- a/gptfdisk
- a/kmod
- a/udisks
- a/udisks2
- a/upower
- ap/htop
- ap/soma
- d/gcc-go
- d/llvm
- kde/analitza
- kde/ark
- kde/blinken
- kde/bluedevil
- kde/calligra
- kde/cantor
- kde/filelight
- kde/gwenview
- kde/kaccessible
- kde/kactivities
- kde/kalgebra
- kde/kalzium
- kde/kamera
- kde/kanagram
- kde/kate
- kde/kbruch
- kde/kcalc
- kde/kcharselect
- kde/kcolorchooser
- kde/kde-baseapps
- kde/kde-runtime
- kde/kde-wallpapers
- kde/kde-workspace
- kde/kdegraphics-mobipocket

- kde/kdegraphics-strigi-analyzer
- kde/kdegraphics-thumbnailers
- kde/kdevelop-pg-qt
- kde/kdf
- kde/kfloppy
- kde/kgamma
- kde/kgeography
- kde/kgpg
- kde/khangman
- kde/kig
- kde/kiten
- kde/klettres
- kde/kmag
- kde/kmousetool
- kde/kmouth
- kde/kmplot
- kde/kolourpaint
- kde/konsole
- kde/korundum
- kde/kplayer
- kde/kremotecontrol
- kde/kruler
- kde/ksaneplugin
- kde/ksecrets
- kde/ksnapshot
- kde/kstars
- kde/ktimer
- kde/ktouch
- kde/kturtle
- kde/kwallet
- kde/kwebkitpart
- kde/kwordquiz
- kde/libkdcraw
- kde/libkdeedu
- kde/libkexiv2
- kde/libkipi
- kde/libksane
- kde/marble
- kde/networkmanagement
- kde/okular
- kde/oxygen-gtk2
- kde/parley
- kde/perl kde
- kde/perlqt
- kde/polkit-kde-agent-1
- kde/polkit-kde-kcmodules
- kde/printer-applet
- kde/pykde4
- kde/qtruby
- kde/quanta
- kde/rocs

- kde/smokegen
- kde/smokekde
- kde/smokeqt
- kde/step
- kde/superkaramba
- kde/svgpart
- kde/sweeper
- kde/wicd-kde
- kdei/calligra-l10n-*
- l/GConf
- l/gcr
- l/glade3
- l/glib-networking
- l/gnome-keyring
- l/gobject-introspection
- l/grantlee
- l/gsettings-desktop-schemas
- l/gtk+3
- l/gvfs
- l/herqq
- l/icu4c
- l/js185
- l/keybinder
- l/keyutils
- l/lcms2
- l/libatasmart
- l/libbluedevil
- l/libcanberra
- l/libcroco
- l/libffi
- l/libgnome-keyring
- l/libnl3
- l/libproxy
- l/libsoup
- l/libssh
- l/libtasn1
- l/libusb-compat
- l/libyami
- l/media-player-info
- l/mozilla-nss
- l/phonon-gstreamer
- l/phonon-xine
- l/pycurl
- l/raptor2
- l/sg3_utils
- l/sound-theme-freedesktop
- n/ModemManager
- n/NetworkManager
- n/cifs-utils
- n/iftop
- n/mobile-broadband-provider-info

- n/nettle
- n/p11-kit
- x/freelut
- x/libXaw3d
- x/libXcm
- x/mtdev
- x/transset
- x/xcursor-image
- x/xcursor-keymaps
- x/xcursor-renderutil
- x/xcursor-wm
- xap/network-manager-applet
- xfce/*

PACKAGE REMOVALS SINCE 13.37

- a/gdisk - *renamed to gptfdisk*
- a/module-init-tools - *replaced by kmod*
- ap/ntfsprogs - *merged into ntfs-3g*

KDE packages have been split up - notice all of the new packages in the "Added" list...

- kde/kdeaccessibility
- kde/kdebase
- kde/kdebase-runtime
- kde/kdebase-workspace
- kde/kdebindings
- kde/kdeedu
- kde/kdegraphics
- kde/kdeutils
- kde/konq-plugins
- kde/koffice
- kde/polkit-kde-1
- l/hal-info - *obsolete; replaced by various udev components*
- l/hal - *obsolete; replaced by upower, udisks, various others*
- l/jre - *removed due to licensing issues*
- l/wv2 - *no longer used by anything in Slackware*
- l/raptor - *no longer used (everything links raptor2 instead)*

*All of these firmwares (except some *old* ones are now included in the kernel-firmware package - if you need an older firmware, use an older Slackware.*

- n/rt2860-firmware
- n/rt2870-firmware
- n/rt61-firmware
- n/rt71w-firmware
- n/iwlwifi-100-ucode
- n/iwlwifi-1000-ucode
- n/iwlwifi-3945-ucode
- n/iwlwifi-4965-ucode

- n/iwlwifi-5000-ucode
- n/iwlwifi-5150-ucode
- n/iwlwifi-6000-ucode
- n/iwlwifi-6xxx-ucode
- x/radeon_ucode

All of these are obsolete and/or will not build with recent Xorg and/or toolchains:

- x/xf86-video-sisusb
- x/xf86-video-tseng
- x/xf86-video-xgi
- x/xf86-video-radeonhd
- x/scripts
- x/xaw3d

All of these are moved to the new xfce/ set:

- xap/thunar-volman
- xap/xfce4-notifyd
- xap/xfce4-power-manager
- xap/xfce
- extra/jdk - removed due to licensing issues

OTHER NOTABLE CHANGES AND HINTS

Use one of the provided generic kernels for daily use. Do not report bugs until/unless you have reproduced them using one of the stock generic kernels. You will need to create an initrd in order to boot the generic kernels - see /boot/README.initrd for instructions. The huge kernels are primarily intended as “installer” and “emergency” kernels in case you forget to make an initrd. For most systems, you should use the generic SMP kernel if it will run, even if your system is not SMP-capable. Some newer hardware needs the local APIC enabled in the SMP kernel, and theoretically there should not be a performance penalty with using the SMP-capable kernel on a uniprocessor machine, as the SMP kernel tests for this and makes necessary adjustments. Furthermore, the kernel sources shipped with Slackware are configured for SMP usage, so you won't have to modify those to build external modules (such as NVidia or ATI proprietary drivers) if you use the SMP kernel.

If you decide to use one of the non-SMP kernels, you will need to follow the instructions in /extra/linux-3.2.21-nosmp-sdk/README.TXT to modify your kernel sources for non-SMP usage. Note that this only applies if you are using the Slackware-provided non-SMP kernel - if you build a custom kernel, the symlinks at /lib/modules/\$(uname -r)/{build,source} will point to the correct kernel source so long as you don't (re)move it.

As with 13.37, the system udev rules now reside in /lib/udev/rules.d/ instead of /etc/udev/rules.d/ in older versions. There should never be a reason to edit anything in /lib/udev/rules.d/, so if you think you have a case where this is required, either you're wrong or it needs to be addressed in the upstream source. However, you can override default rules by placing one with an identical name inside /etc/udev/rules.d/ The rules files in /etc/udev/rules.d/ are still intended to (maybe) be edited as needed by local system administrators, and as such, the rules for optical and network devices will still be placed there.

Speaking of udev, pay particular attention to 70-persistent-net.rules and 70-persistent-cd.rules in /etc/udev/rules.d/ – these two are automatically generated by the system. If you remove, add, and/or replace some hardware (specifically network cards and/or optical drives) in a machine, you will probably need to edit one or both of the rules files mentioned above.

If you are upgrading (as opposed to a new install), there are some stale files in /etc/modprobe.d/ that you will need to remove: blacklist.conf, bluetooth.conf, isapnp.conf, psmouse.conf, and usb-controller.conf See /etc/modprobe.d/README for a more detailed explanation.

Both printing and scanning require that your user account be a member of the “lp” group (membership in the “scanner” group is no longer needed); we had to configure sane to use the “lp” group or else multifunction devices (e.g. print/scan/copy units) would only do one or the other (depending on whether the group ownership was “lp” or “scanner”).

Xorg no longer uses /etc/X11/xorg.conf by default (and in most cases, there is absolutely no need for it). You can still create an xorg.conf file if you wish, or you can create some minimal xorg.conf snippets with only the specific contents that you wish to override (as an example, to use a binary video driver) as separate files in the /etc/X11/xorg.conf.d/ directory.

/usr/share/X11/xorg.conf.d/ is the “packaged” configuration directory; all files ending with “.conf” in this directory are used by the X server unless there is an identically-named file in the local sysadmin directory. The local sysadmin config directory is /etc/X11/xorg.conf.d/ - all files ending with “.conf” in this directory are parsed.

There are several default config files in /usr/share/X11/xorg.conf.d/:

- 10-evdev.conf - a “catchall” file for input devices using the evdev driver; this should work for most hardware in the absence of a better driver
- 50-synaptics.conf - overrides the earlier 10-evdev.conf file and uses the synaptics driver for all touchpads
- 50-wacom.conf - overrides the earlier 10-evdev.conf file and uses the wacom driver for Wacom tablets
- 90-keyboard-layout.conf - this sample (“normal” en layout) keeps the “old” default of allowing Zap'ing the Xserver.

If you need to modify any of these defaults, then copy the relevant file from /usr/share/X11/xorg.conf.d/ to /etc/X11/xorg.conf.d/ and edit the copy.

Regardless of your chipset (though it seems more common with intel), if KDE crashes on startup, try disabling the Composite extension (which will also disable all of the fancy desktop effects). Place the following content in a file at /etc/X11/xorg.conf.d/disable-composite.conf:

```
Section "Extensions"
    Option "Composite" "Disable"
EndSection
```

Now that KMS (Kernel Mode Setting) for graphics cards has (mostly) stabilized, it is enabled by default for intel, ati, and nvidia graphics chipsets. It is possible to disable it use “nomodeset” as a kernel append in lilo.conf, but Xorg will not work at all on intel (and some ati) chips if you do that.

If you want to change the resolution of the KMS console, that can be done with something like this as a kernel append in lilo.conf:

```
append="video=1024x768"
```

Speaking of lilo.conf and KMS, make sure you use either vga=normal or vga=extended - some of the framebuffer drivers don't like KMS very much...

If you see some log messages like this:

```
mdadm: sending ioctl 1261 to a partition!
```

They're completely harmless - do not be alarmed. Here's why:

<http://lists.debian.org/debian-kernel/2012/03/msg00446.html>

The version of logrotate in Slackware 14.0 added an "su" option to define user/group for rotation. Logrotate now skips directories which are world writable or writable by a group which is not "root" unless the "su" directive is used. This has ramifications for third party packages that installed e.g. /var/log/somedir owned by a non-root user and/or group.

If you are using a KVM switch, you might experience problems with the mouse when switching from one system to another. If so, you probably need to be using the imps protocol for the psmouse driver, and that's a simple fix: Copy /lib/modprobe.d/psmouse.conf to /etc/modprobe.d/psmouse.conf, then uncomment the following line in /etc/modprobe.d/psmouse.conf:

```
#options psmouse proto=imps
```

Next, unload and reload the psmouse module (do this as root):

```
modprobe -r psmouse ; modprobe psmouse
```

If you have set up an encrypted root partition, you will need to have access to your keyboard in order to type the passphrase. This may require you to add the uhci-hcd and usbhid modules to your initrd image if you have a USB keyboard. Also note that if you are using a non-US keyboard, you can use the '-l' parameter to the 'mkinitrd' command in order to add support for this keyboard to your initrd.

If you have permission errors when attempting to burn a cdrom or dvd image, such as the following:

```
/usr/bin/cdrecord: Operation not permitted. Cannot send SCSI cmd via ioctl
```

then cdrecord almost certainly needs root privileges to work correctly. One potential solution is to make the cdrecord and cdrdao binaries suid root, but this has possible security implications. The safest way to do that is to make those binaries suid root, owned by a specific group, and executable by only root and members of that group. For most people, the example below will be sufficient (but adjust as desired depending on your specific needs):

```
chown root:cdrom /usr/bin/cdrecord /usr/bin/cdrdao
chmod 4750 /usr/bin/cdrecord /usr/bin/cdrdao
```

If you don't want all members of the 'cdrom' group to be able to execute the two suid binaries, then create a special group (such as 'burning' which is recommended by k3b), use it instead of 'cdrom' in the line above, and add to it only the users you wish to have access to cdrecord and cdrdao.

Input methods for complex characters (CJK, which is shorthand for Chinese, Japanese, Korean) and

other non-latin character sets have been added. These input methods use the SCIM (Smart Common Input Method) platform. The environment variables for SCIM support are set in `/etc/profile.d/scim.sh`. The requirements for getting SCIM input methods to work in your X session are as follows:

1. Use a UTF-8 locale. Look in `/etc/profile.d/lang.sh` for setting your language to (for instance) `en_US.UTF-8`. As a word of warning: maybe you should leave root with a non-UTF-8 locale because you don't want root's commands to be misinterpreted. You can add the following line to your `~/.profile` file to enable UTF-8 just for yourself:

```
export LANG=en_US.UTF-8
```

2. Make the scim profile scripts executable. These will setup your environment correctly for the use of scim with X applications. Run:

```
chmod +x /etc/profile.d/scim.*
```

3. Start the scim daemon as soon as your X session starts. The scim daemon must be active before any of your X applications. In KDE, you can add a shell script to the `~/.kde/Autostart` folder that runs the command "scim -d". In XFCE you can add "scim -d" to the Autostarted Applications. If you boot your computer in runlevel 4 (the graphical XDM/KDM login) you can simply add the line "scim -d" to your `~/.xprofile` file. This gives you a Desktop Environment independent way of starting scim.

When scim is running, you will see a small keyboard icon in your system tray. Right-click it to enter SCIM Setup. In 'Global Setup' select your keyboard layout, and you are ready to start entering just about any language characters you wish! Press the magical key combo `<Control><Space>` in order to activate or deactivate SCIM input. The SCIM taskbar in the desktop's corner allows you to select a language. As you type, SCIM will show an overview of applicable character glyphs (if you are inputting complex characters like Japanese).

If you have an older machine (with a BIOS released prior to 2001) and it will not power off on shutdown, try adding this to your kernel's lilo stanza:

```
append = "acpi=force"
```

If you have a Thinkpad T400 or T500, you probably want to append "pci=reboot" to the kernel boot parameters. For more information about this issue, see

<https://encrypted.google.com/search?hl=&q=t400+%22pci%3Dreboot%22>

TODO:

Not sure if/where these belong yet:

<http://git.gnome.org/browse/gvfs/tree/monitor/udisks2/what-is-shown.txt>

From:

<https://docs.slackware.com/> - **SlackDocs**

Permanent link:

<https://docs.slackware.com/playground:changeshints>

Last update: **2012/08/19 19:31 (UTC)**

