Slackware - 13.37 UPGRADE.TXT

Slackware 13.1 to 13.37 Upgrade HOWTO <volkerdi AT slackware.com>

This document explains how to upgrade from Slackware 13.1 to Slackware 13.37.

For details of important changes from Slackware 13.1 to 13.37, see the file 'CHANGES_AND_HINTS.TXT'. Thanks to Robby Workman for help with this. If you have partitions in /etc/fstab named /dev/hd*, please take special note of the instructions for the changeover to /dev/sd* devices or your machine will not reboot correctly.

Before you begin, I would strongly recommend making a backup of your system, or, if not the entire system, at least the /etc directory. You might find that you need to refer to a few things after the upgrade process is complete. Back it up, or take your chances.

OK, now that everything is safely backed up, let's proceed. 🙂

To do this, you'll need the Slackware 13.37 packages. If these are on a CD, create a new directory to mount the CD on so that it doesn't get in the way during the upgrade:

```
mkdir /packages
mount /dev/cdrom /packages
```

The packages don't have to be on a CD-ROM, as an alternative you could copy the slackware directory (the one with the various package subdirectories in it, basically the "slackware" or "slackware64" directory from the install disc) to someplace like /root/slackware/. The important thing is that you know where the slackware packages directory is. We'll use /root/slackware in the following examples.

1. Put your machine in single-user mode:

   ```
   telinit 1
   ```

   Note that this is _not_ strictly required, and there have been reports of success remotely upgrading machines that are still in multiuser mode. However, more things can go wrong in multiuser, so especially if you're considering a remote upgrade in multiuser mode, you might want to clone the machine locally so that you can do a test run to uncover any problem areas and come up with workarounds for them.

2. Upgrade your glibc shared libraries. This is important, or things might go haywire during the next part of the upgrade:

   ```
   upgradepkg /root/slackware/a/glibc-solibs-*.t?z
   ```

3. Upgrade your package utilities and related tools:

```
upgradepkg /root/slackware/a/pkgtools-*.tgz
    upgradepkg /root/slackware/a/tar-*.tgz
    upgradepkg /root/slackware/a/xz-*.tgz
    upgradepkg /root/slackware/a/findutils-*.txz
```

4. Upgrade everything else (and install new packages):

```
upgradepkg --install-new /root/slackware/*/*.t?z
```

If you wish to upgrade everything except for the KDEI language packs for KDE (these take a lot of space and can be dealt with after the main upgrade more quickly and easily), running this script in the "slackware" directory will do the trick:

```
#!/bin/sh
    for dir in a ap d e f k kde l n t tcl x xap xfce y ; do
      ( cd $dir ; upgradepkg --install-new *.t?z )
    done
```

5. Remove obsolete packages. The CHANGES_AND_HINTS.TXT file should have a list of these. You may also wish to go into /var/log/packages and take a look at the package list:

```
ls -lt | less
```

You may spot some old, obsolete, or discontinued packages. If so, you can remove these using 'removepkg'. This command will get rid of the packages which became obsolete since Slackware 13.1:

```
removepkg util-linux-ng guidance-power-manager eggdbus iptraf \
    libXTrap libXprintAppUtil libXprintUtil libxkbui rstart trapproto \
    xf86rushproto xfindproxy xfwp xplsprinters xprehashprinterlist \
    xproxymanagementprotocol xsetmode xsetpointer xtrap
```

Note the removal of util-linux-ng! You better make sure that the renamed util-linux package is installed before removing that. 🙂

6. Fix your config files. Some of the config files in /etc are going to need your attention. You'll find the new incoming config files on your system with the ".new" extension. You may need to fill these in with information from your old config files and then move them over. Feel brave? You can use this little script to install most of the .new config files in /etc. If you've made any local changes you'll need to add them to the newly installed files. Your old config files will be copied to *.bak. Anyway, it might be an easier starting point. Here it is:

```
#!/bin/sh
    cd /etc
    find . -name "*.new" | while read configfile ; do
      if [ ! "$configfile" = "./rc.d/rc.inet1.conf.new" \
        -a ! "$configfile" = "./rc.d/rc.local.new" \
        -a ! "$configfile" = "./group.new" \
        -a ! "$configfile" = "./passwd.new" \
```

```
                -a ! "$configfile" = "./shadow.new" ]; then
            cp -a $(echo $configfile | rev | cut -f 2- -d . | rev) \
              $(echo $configfile | rev | cut -f 2- -d . | rev).bak 2>
/dev/null
            mv $configfile $(echo $configfile | rev | cut -f 2- -d . |
rev)
        fi
      done
```

You might also wish to move these config files over:

```
/usr/man/man.conf.new
    /usr/share/vim/vimrc.new
```

7. If you use a non-en_US language pack for KDE and you already have it installed, then you may upgrade it by moving into the slackware/kdei directory and using this command:

```
upgradepkg --install-new k*<your KDE locale>*t?z
```

To have upgradepkg cycle through all of the available packages, and see which ones need to be upgraded, use this in slackware/kdei:

```
upgradepkg *t?z
```

If your language has been added to KDE since Slackware 13.1, you'll need to install it using installpkg, or upgradepkg –install-new. Typically you'll need to make sure that you have installed the slackware/kdei packages for kde, koffice, and k3b (if you use those).

8. IMPORTANT! *Before* attempting to reboot your system, you will need to make sure that the bootloader has been updated for the new kernel! First, be sure your initrd is up to date (if you use one). If you use LILO, make sure the paths in /etc/lilo.conf point to a valid kernel and then type 'lilo' to reinstall LILO. If you use a USB memory stick to boot, copy the new kernel to it in place of the old one. If you were using devices such as /dev/hda (IDE hard drive) with Slackware 13.1, you will need to consider how to switch over to the new /dev/sda type device names. See CHANGES_AND_HINTS.TXT for complete instructions on how to do this. If you have not read this yet, now is a great time to go look at that. Forget about step 8 and 9 and just go complete that part of the upgrade and you'll be good to go.

9. Return to multi-user mode:

```
telinit 3
```

10. Reboot to start using the new kernel.

At this point you should be running Slackware 13.37. 🙂

I wish everyone good luck with this!

— Patrick Volkerding volkerdi AT slackware.com

From:

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

Permanent link:

https://docs.slackware.com/playground:upgrade

Last update: **2012/08/19 19:54 (UTC)**