An Introduction to Slackware

Presentation to Linux Users of Victoria, Inc.
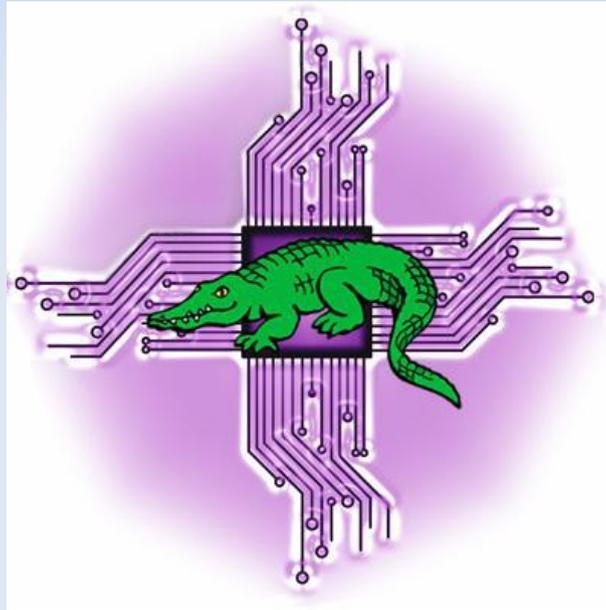
February 15th, 2014

http://levlafayette.com

What Is Slackware?

Slackware is venerable GNU/Linux distribution; an operating system consisting of a specified collection utilities and applications built on top of the Linux kernel. According to which metric used, it is somewhere around 10-20 in terms of distribution popularity and use.

The name "Slackware" derives from the edict of Slack (always capitalised) from the parody religion, the Church of the Subgenius, which has been described "the ability to effortlessly achieve your goals". It also represents that the initiator, Patrick Volkerding, never intended it to be a serious project.

"It's what I started calling it back when it was really just a hacked version of SLS and I had no intention of putting it up for public retrieval. When I finally did put it up for FTP, I kept the name. I think I named it 'Slackware' because I didn't want people to take it all that seriously at first." (Linux Journal #2 Interview, April 1st, 1994)

Historically Slackware was only available on IA-32 architectures. It wasn't until v13.0 that the x86_64 version was officially supported, although prior to that there was a port, slamd64. Slackware is also available for the ARM architecture in the form of Slackware ARM (originally known as 'ARMedslack') and for IBM S/390 mainframe (known as Slack/390). Until 2008 there was an unofficial port for PowerPC systems (Slackintosh).

History of Slackware

The earliest Linux distributions all date back from 1992; MCC Intermin, followed by TAMU, and SLS, and Yggdrasil. SLS was the first release to offer a fairly comprehensive distribution containing more than the Linux kernel and basic utilities; for example it included TCP/IP and the X Window System.

Slackware was a cleanup and fork of SLS (Softlanding Linux System) initiated by Patrick Volkerding, initiated when SLS tried to claim ownership of the scripts that Patrick had written. The first public release as its own distribution was on July 17, 1993; it was supplied on 24 3.5" inch floppy disks. This expanded to 73 floppy disks by v2.1 (October 1994). V

Version 3.0 released in November 1995. was available on a CD-ROM based installation, and v3.1 (July 1996) came out with the name "Slackware 96" (in contrast to Windows95). This was the first version of Slackware to ship with the 2.x kernel. In 1999, Slackware's release number jumped from 4 to 7 as a marketing ploy. Another marketing ploy was v13.37 in April 2011. The current release is 14.1 (4 November 2013).

Slackware is conservative in its source additions. The transition form XFree86 to X.org did not occur until 2004. The Hardware Abstraction Layer wasn't added until 2007 (and then merged into the udev device manager in 2012). A stable 64-bit version was not available until 2009.

History of Slackware

What Makes Slackware Different?

For those more familiar with major distributions there are several features of Slackware that may seem somewhat different to the downright perplexing. Much of this has to do with particular design decisions which are characteristic to Slackware, based on a philosophy of "purity", "simplicity", and "stability". By "purity" what is meant is lack of changes from upstream software sources. By "simplicity" what is mean is system design, not usability. By "stability" what is meant that new versions of Slackware are released when they are tested and ready, not according to the newness of software or a fixed schedule.

As part of the installation process it does not come with a guided hard disk partitioning tool, instead relying on tools like cfdisk and fdisk. Once the partition table is created, unlike most distributions Slackware does not come with a graphic installer. Rather it makes use of the ncurses library for a range of text-based user interfaces which are "GUI-like" for the program "setup". Setup manages software through a letter system of associated applications, a legacy from its floppy disk days. After Setup, Slackware uses LILO as a boot manager and the menu-driven configuration utility liloconfig instead of the newer, and more well known, GRUB.

Slackware does not boot directly into an X-windows system by default, and requires a modification of the /etc/inittab file to change this. Also the 64 bit install of Slackware is "pure"; it will not run or compile 32-bit binaries. Adding multilib capability requires adding multilib versions of glibc (to support running both 32 and 64 bit binaries), and gcc (to compile 32 and 64 bit binaries), and installing 32 bit system libraries.

What Makes Slackware Different?

In terms of development, Slackware does not have a formal bug tracking facility, no official procedure to become a contributor or developer, and no official repository. Contributions are informal and final decisions about what is included in the distribution are determined by Patrick Volkerding. Version 14 noted assistance from Eric Hameleers, Robby Workman, Piter Punk, Stuart Winter, Mark Post, Vincent Batts, Heinz Wiesinger, Amritpal Bath, Alan Hicks, Erik Jan Tromp, Karl Magnus Kolstø, Fred Emmott, and NetrixTardis.

Slackware doesn't ship with GNOME because (a) it's too painful to build from source (b) there's a number of community projects which support GNOME on Slackware. The official Slackware documentation recommends XFCE if one wants to use a GTK based desktop. For an example of the joys of building GNOME from source see: http://linuxgazette.net/164/laycock.html . If one really wants GNOME on Slackware see: http://www.droplinegnome.org or https://github.com/gnomeslackbuild/gsb-build-system/wiki

Package Management and Dependencies

One difference that really puts people off using Slackware is the package management system doesn't do dependency checking. Again, simplicity as in system design, not in usability. You're expected to know your system (and if you don't, you will learn a lot about dependencies). The two main package managers are installpkg (along with removepkg and upgradepk), which operates on local packages, and slackpkg, which can install packages over a network. The latter is recommended for keeping track of updates to packages through adding a mirror location to the system. This can also be used for searching and installing packages. c.f., http://docs.slackware.com/slackware:slackpkg

"Personnaly, i've given up on automatic dependencies management the day that, after installing a small, KDE-based debian and wanting to install Cups to print easily, apt-get downloaded me the full Gnome install just because there was a small dependency of the Cups package on gnome-print." rvdboom on linuxquestions.org, Nov 6, 2009

Slackware packages are tarballs using the LZMA algorithm, indicated by the .txz extension. The package contains the software being installed and metadata files for the Slackware package manager. These are in a directory structure that mirrors their respective locations in the root directory of the destination system. The metadata install/ directory typically contains slack-desc and doinst.sh. The first is describes the package being installed. The doinst.sh file is a shell script which is run at the end of the installation of a package.

Package Management and Dependencies

Another option is source tarballs from http://slackbuilds.org which has an automated and consistant compile process which will include the source and the slackbuild script. Note the descriptions on Slackbuilds and pkgs.org also makes mention of the immediate dependencies required. For the latest repository go to: http://slackbuilds.org/repository/14.1/ or http://pkgs.org/slackware-14.1/slackware-x86_64/

There are community developed tools for Slackware that carry out automated dependency resolution. These are controversial in the Slackware world as the are contrary to what is considered to be the a core design principle, but they are easier from a users perspective. These include Swaret (http://swaret.sourceforge.net/) and slapt-get (http://software.jaos.org/#slapt-get).

Installation

The installation process begins with a simple reboot with the Slackware DVD or CD in the drive and with the boot order set correctly. The first screen calls for special kernel parameters and in most case this can be ignored. Ater this is the option to set non-US keyboards with a list of mappings if this is chosen. The next step is to login as root and determine partitioning of the drive using fdisk or cfdisk; the latter is recommended with partitions for /boot, /home, and a swap partition.

The next step involves the setup environment. This again asks for the keyboard mapping and confirmation of the detected swap partition. After that select the root partition as the target for the install and select the source (i.e., the DVD/CD). An option then follows for selecting particular groups of software. Slackware officially recommends a "full install" for those who are unfamiliar with the system; this is another item of contention among those who think this is agains the Slackware way. A variety of configuration options are then offered, including the creation of a bootable USB stick, setting up LILO, assigning the default character encoding (Slackware, playing safe, assumes en_US; UTF-8 is an option worth considering in preference), assigning an IP address (static or dynamic), and choosing the default window manager.

Installation

Installation

PACKAGE SERIES SELECTION

Now it's time to select which general categories of software to install on your system. Use the spacebar to select or unselect the software you wish to install. You can use the up and down arrows to see all the possible choices. Recommended choices have been preselected. Press the ENTER key when you are finished.

```
[*] A      Base Linux system
[*] AP     Various Applications that do not need X
[*] D      Program Development (C, C++, Lisp, Perl, etc.)
[*] E      GNU Emacs
[*] F      FAQ lists, HOWTO documentation
[*] K      Linux kernel source
[*] KDE    Qt and the K Desktop Environment for X
[ ] KDEI   International language support for KDE
[*] L      System Libraries (needed by KDE, GNOME, X, and more)
    ↓(+)                                                    56%
```

        <   OK   >   <Cancel>

Post-Installation and Conclusion

After installation many users wish to boot directly into their preferred window manager. This is achieved by editing, as root, the default runlevel in /etc/inittab from 3 to 4. If multilib capability is desired follow the instructions at http://docs.slackware.com/slackware:multilib . As always with any system there may be other minor bugs as well. Fortunately with Slackware these are usually well documented. For example in my system alsa was sending sound to the wrong device - the following acted as a workaround http://docs.slackware.com/howtos:hardware:sound_problems_on_slackware

Finally, it is not recommended to work full time as root; create a new user account with the adduser command.

In conclusion, Slackware is not for everyone. It does assume either knowledge or a desire to understand the Linux system, and especially packages, in some depth. It does come with excellent documentation to help you, it does recommend a bit of forethought and planning on what exactly a system is going to be used for and sticking to it.