



## Installing distros on USB sticks with Unetbootin

# LITTLE BOOTS

Unetbootin lets users set up almost any distribution on a bootable USB stick. **BY KRISTIAN KISSLING**

**V**ampire hunter Van Helsing always has some holy water along for emergencies; Daywalker Blade relies on silver stakes. Experienced Linux users are more likely to resort to Linux on a USB stick to exorcise viruses from infected Windows computers. Thanks to Unetbootin, setting up an electronic emergency tool on a stick does not take much time. The installer lets you install various Linux distributions quickly and securely on an empty USB stick. Then you can use the stick to boot other computers.

Unetbootin will roll out live versions as well as installation and netboot variants. If you are not happy with the de-

fault selection of distributions supported by Unetbootin, you can feed the tool an ISO file for your favorite distro. Time-consuming workarounds necessitated by alternative tools such as Pendrivelinux [1] are thus a thing of the past.

To install Unetbootin, copy *unetbootin-linux-323* (from the DVD that came with this issue or from the website [2]) to your home directory and make it executable:

```
$chmod u+x unetbootin-linux-323
```

The “Installing Unetbootin” box describes the steps for setting up Unetbootin in openSUSE 11.1 and Ubuntu 9.04.

These steps should work for other distributions as well.

### Preparing the USB Stick

Before you dive into Unetbootin, you might want to prepare your USB stick for installation by deleting the existing data and reformatting the stick. On openSUSE, type *su root* to become root; on Ubuntu, add *sudo* to the start of the command line.

To start, type *sudo umount /dev/sdb1* to mount the stick, replacing */dev/sdb1* used in this example with the correct mount point. To discover the mount point ID, launch a terminal and type *tail -f /var/log/messages* shortly before unplugging the stick. The *sdb: sdb1* line tells you that your system has assigned a device file of */dev/sdb* to the stick, which has a partition called */dev/sdb1*.

## Installing Unetbootin

In openSUSE 11.1, make sure you have administrative privileges by typing `su root`, then type `./unetbootin-linux-323` to launch the file. For the sake of completeness, you might prefer running YaST and installing the `p7zip` package before you start, although the software will run without this dependency.

On Ubuntu 9.04, working with a normal user account, type your administrative password after launching Unetbootin with the `./unetbootin-linux-323` command. Additionally, you can use Synaptic to install the `p7zip-full` package, which the software requires.

Now, working as root, type `fdisk /dev/sdb` to launch the Fdisk partitioning tool; the tool is controlled from the keyboard. Type `P` to discover how many partitions are on the stick. Then press `D` and type the partition number to delete the partitions. If the stick only contains one partition, you do not need to specify a number. After deleting the partitions, press `N` to create a new partition; it needs to be a primary partition (`P`). To use the first partition, select `-1` then press Enter twice to use all the sectors on the stick. Pressing `A` and `-1` enables the partition you created by setting the boot flag; now finish the preparation work by pressing `W` (for “write”). Fdisk now writes all the changes to the stick; you can cancel at any time before this.

The next step is to format the empty – and still unmounted – partition with the ext3 filesystem. To do so, become root and type `mkfs.ext3 -b 4096 /dev/sdb1`, again replacing `sdb` with the device file for your USB stick. Linux will now write a fresh filesystem to the stick; this can

take a couple of minutes. Linux writes blocks of 4KB (as defined by the `-b 4096` option) to speed up the process. If you want to store Windows files on the stick, you need to run the `mkfs.vfat -F 32 /dev/sdb1` command to create a FAT32 partition. The drawback to this option is that FAT32 does not support files of more than 4GB.

After completing the format, unplug the stick, plug it back in to the USB port, and make sure the system mounts it again. In the file manager, you should now see a *Lost + Found* folder – but nothing else.

## Unetbootin in Action

With the empty, mounted stick prepared, it is now time to launch Unetbootin. In the *Distribution* pull-down menu select your favorite distribution and the matching version number on the right (Figure 1). The small text box below this tells you something about your choice of distribution. At the bottom, select *USB drive* as the *Type*, and select the newly created `sdb1` partition from the drop-down menu to the right. If you choose *Show all drives*, Unetbootin will list other drives – this is useful if you are not sure you have found your USB stick. Pressing *OK* launches Unetbootin – sink or swim time (Figure 2).

If everything goes as expected, the software will download an ISO from an FTP server, and generate a configuration file for the Syslinux bootloader. Unetbootin then finishes the transaction with a message and presents the buttons *Reboot now* and *Finish*. That’s all there is to it. As your final trick, you can check to see whether the image really does boot properly from your stick.

## Fast and Manual

Downloading the files can take some time. To



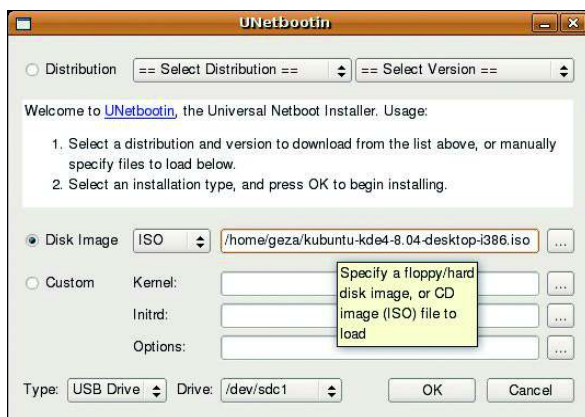
**Figure 2:** If you select a distribution the software supports, Unetbootin will retrieve it from an FTP server and install the distribution on your memory stick.

speed up the process, pass in the ISO file manually. To do so quickly, download the ISO via BitTorrent and, in the Unetbootin GUI, specify the path to the ISO file as your *Image*. If you want, you can even specify the kernel and initrd file versions. This makes sense if you use a non-standard kernel with a specific feature or one that works better with your hardware.

Often, it is quite useful to add a couple of *Options* to the boot process to make sure that Linux works with your hardware. Favorite kernel parameters include `ide = nodma` (for problematic IDE drives) and `acpi = off` (to avoid trouble with non-functional or missing power management).

## Conclusions

All told, Unetbootin is not just a convenient piece of software but a pretty mature one, too. However, nothing is perfect. For example, you still might not be able to install certain distributions. As a case in point, in our lab, we had trouble installing the SliTaz operating system, although Puppy Linux, Elive, and a KDE 4 Live version based on openSUSE worked fine. If the automated installation does not work, you can always try downloading the image manually from the web page and following the steps for the manual install. ■



**Figure 1:** Configuring Unetbootin is fairly intuitive thanks to the easy user interface.

## INFO

[1] Pendrivelinux: <http://www.pendrivelinux.com>

[2] Unetbootin: <http://unetbootin.sf.net/>