GRUB: Changing the Default Kernel

Serge Y. Stroobandt

Introduction

GNU GRUB (short for GNU GRand Unified Bootloader) is the boot loader of most GNU/Linux distributions. It gives a user the choice to boot one of multiple operating systems or select a specific kernel configuration available on a disk partition of the computer system.

Now, suppose you just upgraded your rolling release GNU/Linux distribution. The upgrade also included a new version of the Linux kernel. By default, GRUB will choose the latest kernel to boot after its menu time-out.

Occasionally, a new kernel might break a custom-installed package that does not belong to the standard repositories. Furthermore, it might be that you cannot or do no want to solve this issue straight away. More importantly, you do not want to disrupt the daily routine of other users whilst you are figuring things out. As long as you have not removed the older version of the kernel, you could configure GRUB to boot with that older kernel by default. Here is how it is done.

Start counting from zero

Edit `/etc/default/grub` with administrative privileges:

```
$ sudo nano /etc/default/grub
```

Look for the line containing `GRUB_DEFAULT=0`. Change the number knowing that computers always count starting from zero. In this case, zero corresponds to the latest kernel.

Don't forget to update-grub

After editing the above configuration file, do not forget to run `update-grub` to cause an update of `/boot/grub/grub.cfg`.

```
$ sudo update-grub
```

That is all. Now you can sit back and solve the broken package issue in a relaxed way, perhaps even remotely via OpenSSH. By the way, full details of the options that are available in the /etc/default/grub configuration file can be obtained by typing:

```
$ info -f grub -n 'Simple configuration'
```

or, more conveniently, by clicking this link.