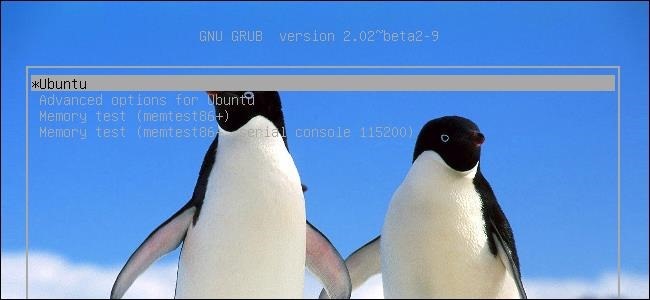
**[How to Configure the GRUB2 Boot Loader’s Settings](http://www.howtogeek.com/196655/how-to-configure-the-grub2-boot-loaders-settings/" \o "How to Configure the GRUB2 Boot Loader’s Settings)**



Ubuntu and most other Linux distributions now use [the GRUB2 boot loader](http://www.howtogeek.com/196520/grub2-101-how-to-access-and-use-your-linux-distributions-boot-loader/). You can change its settings to select a default operating system, set a background image, and choose how long GRUB counts down before automatically booting the default OS.

We configured GRUB2 on Ubuntu 14.04 here, but the process should be similar for other Linux distributions. You may have customized the original GRUB’s settings by editing its menu.lst file in the past, but the process is now different.

**GRUB2 Configuration Basics**

RELATED ARTICLE

**[](http://www.howtogeek.com/196520/grub2-101-how-to-access-and-use-your-linux-distributions-boot-loader/)**

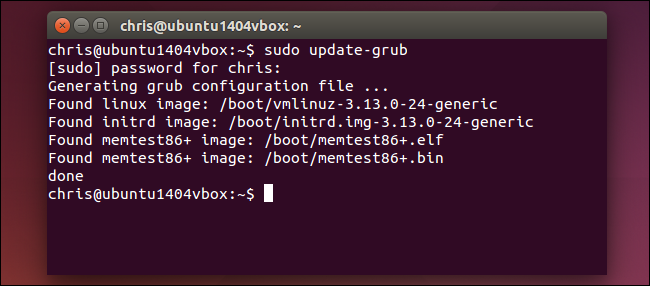
[**GRUB2 101: How to Access and Use Your Linux Distribution’s Boot Loader**](http://www.howtogeek.com/196520/grub2-101-how-to-access-and-use-your-linux-distributions-boot-loader/)  
Ubuntu and practically every other Linux distribution use the GRUB2 boot loader. Unless you have multiple operating systems installed, this... [[Read Article]](http://www.howtogeek.com/196520/grub2-101-how-to-access-and-use-your-linux-distributions-boot-loader/)

GRUB2 doesn’t use a menu.lst file. Instead, its main configuration file is the /boot/grub/grub.cfg file. However, you shouldn’t edit this file by hand! This file is just for GRUB2’s own usage. It’s automatically created by running the **update-grub** command as root — in other words, by running **sudo update-grub** on Ubuntu.

Your own GRUB settings are stored in the /etc/default/grub file. Edit this file to change GRUB2’s settings. Scripts are also located in the /etc/grub.d/ directory. For example, on Ubuntu, there are scripts here that configure the default theme. There’s also an os-prober script that checks the system’s internal hard drives for other installed operating systems — Windows, other Linux distributions, Mac OS X, and so on — and automatically adds them to GRUB2’s menu.

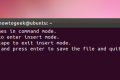
When you run the update-grub command, GRUB automatically combines the settings from the /etc/default/grub file, the scripts from the /etc/grub.d/ directory, and everything else, creating a /boot/grub/grub.cfg file that’s read at boot.

In other words, to customize your GRUB2 settings, you’ll have to edit the /etc/default/grub file and then run the **sudo update-grub** command.



**Edit the GRUB Configuration File**

RELATED ARTICLE

**[](http://www.howtogeek.com/102468/a-beginners-guide-to-editing-text-files-with-vi/)**

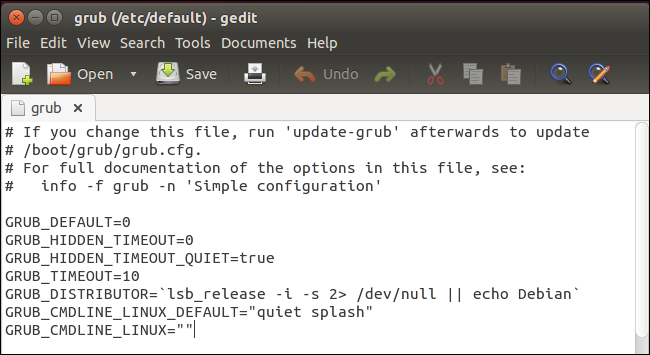
[**A Beginner’s Guide to Editing Text Files With Vi**](http://www.howtogeek.com/102468/a-beginners-guide-to-editing-text-files-with-vi/)  
Vi is a powerful text editor included with most Linux systems, even embedded ones. Sometimes you’ll have to edit a... [[Read Article]](http://www.howtogeek.com/102468/a-beginners-guide-to-editing-text-files-with-vi/)

Open the /etc/default/grub file for editing in a standard text editor. If you want to use a graphical text editor, open a terminal — or press Alt+F2 — and run the following command:

gksu gedit /etc/default/grub

For [an easy-to-use terminal-based editor — Nano](http://www.howtogeek.com/howto/42980/the-beginners-guide-to-nano-the-linux-command-line-text-editor/) — use the following command. You can use any text editor you like, of course — including [the standard vi text](http://www.howtogeek.com/102468/a-beginners-guide-to-editing-text-files-with-vi/) editor.

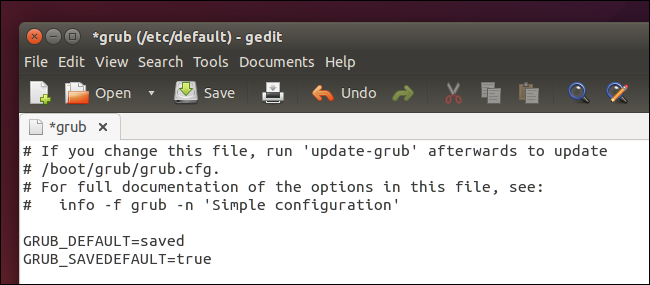
sudo nano /etc/default/grub



The /etc/default/grub file is short and should be easy to edit. As with any other configuration file, you need to edit the options to your desired state and then change the file. If any of the options below doesn’t already appear in the file, add it on a new line. If it does, edit the existing line instead of adding a duplicate one.

**Choose the Default OS**: Change the **GRUB\_DEFAULT=** line. By default, **GRUB\_DEFAULT=0** uses the first entry as the default — change the number to 1 to use the second entry, 2 to use the third entry, or so on. You could also use **GRUB\_DEFAULT=saved** and GRUB would automatically boot the last operating system you chose each time you boot. You can also specify a label in quotes. For example, if you had an operating system named Windows 7 (loader) in your OS list, you could use **GRUB\_DEFAULT=”Windows 7 (loader)”**

**Save a Default Operating System**: If you choose **GRUB\_DEFAULT=saved**, you also need to add a **GRUB\_SAVEDEFAULT=true** line — otherwise it won’t work.

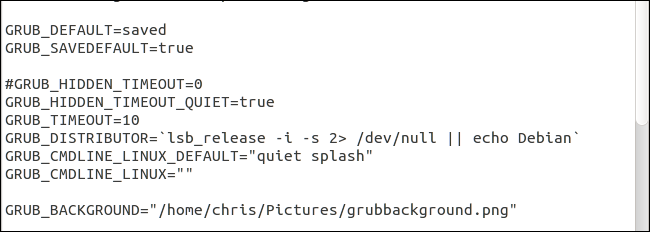


**Choose Whether GRUB is Hidden**: With only one operating system installed, Ubuntu defaults GRUB to automatically boot to the default OS with the GRUB\_HIDDEN\_TIMEOUT=0 option. This option specifies GRUB will be hidden and it will automatically boot to the default OS after 0 seconds –immediately, in other words. You can still access the menu by holding Shift as your computer boots. To set a higher timeout, use something like **GRUB\_HIDDEN\_TIMEOUT=5** — GRUB will display an empty screen or splash screen for five seconds, during which you can press any key to view the menu. To prevent GRUB from being automatically hidden, [comment the line out](http://www.howtogeek.com/118389/how-to-comment-out-and-uncomment-lines-in-a-configuration-file/) — just add a # before it so that it reads **#GRUB\_HIDDEN\_TIMEOUT=0** .

**Control GRUB’s Menu Timeout**: If GRUB isn’t automatically hidden, you’ll see the menu each time your computer boots. GRUB will automatically bot the default operating system after a period of time, usually ten seconds. During that time, you can choose another OS or leave it be to automatically boot. To change the timeout period, edit the **GRUB\_TIMEOUT=10** line and enter any number of seconds you like. (Remember, this is only used if GRUB isn’t hidden.) To prevent GRUB from booting automatically and always wait for you to choose an OS, change the line to **GRUB\_TIMEOUT=-1**

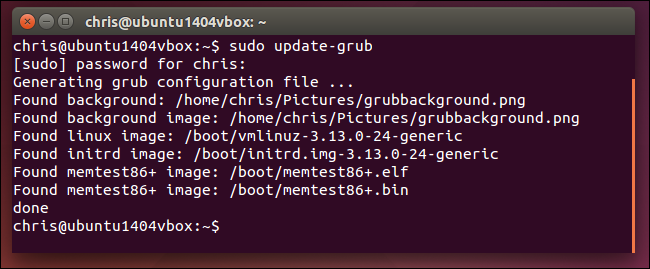
**Choose a Background Image**: The **GRUB\_BACKGROUND** line controls whether a background image is used — by default, GRUB uses a white-on-black monochrome look. You can add a line like **GRUB\_BACKGROUND=”/home/user/Pictures/background.png”** to specify an image file GRUB will use.

The image file must meet various specifications. GRUB supports JPG/JPEG images, but these are limited to 256 colors — so you probably won’t want to use a JPG image. Instead, you’ll probably want to use a PNG image that can have any number of colors. You could also use a TGA image file.



**Make Your Changes Take Effect**

To have your changes take effect, simply save the text file — File > Save in Gedit or Ctrl + O and then Enter to save the file in Nano — and then run the **sudo update-grub** command. Your changes will become part of the grub.cfg file and will be used each time you boot your computer.



These aren’t all of GRUB’s settings, but they are some of the most commonly changed. Other settings can be customized in the /etc/default/grub file, or by editing the scripts in the /etc/grub.d directory.

If you don’t want to edit the files by hand, you may be able to find graphical tools for customizing GRUB2 in your Linux distribution’s [software repositories](http://www.howtogeek.com/117579/htg-explains-how-software-installation-package-managers-work-on-linux/). The above method should work even on Linux distros where such tools aren’t easily available, or if you just have command-line access and want to do it by hand.