How to Compile the Linux Kernel

The linux kernel is the heart of any linux system. It handles user input/output, hardware, and controlling the power in the computer. While the kernel that comes with your linux distribution is usually sufficient, this allows you to make your owns specialized kernel!

Steps

1 **Download the latest version of the linux kernel from http** ://www.kernel.org .

2 **Be sure to download the full source by clicking on the "F" where it says "The latest stable version is**..." otherwise youll just download the patch, which is used when your current kernel is a patch number lower. An example of this would be 3.4.4.1>>3.4.4.2

3 **Make sure you have downloaded the complete source code, and not a patch or change log**.

4 **Once downloaded, open a terminal**.

Now we need to extract the kernel. Use these commands.

5

- tar xjvf kernel (Here -j option is for bz2 compression)

6 **Once extracted, move into the directory (in the terminal) that was created**.

7 **Configure the kernel**. There are 3 common ways to do this.

- make oldconfig -- asks you questions on what the kernel should support one by one, very time consuming.
- make menuconfig -- creates a menu where you can browse options on what the kernel supports. Requires ncurses library, but that is likely already on your computer.
- make qconfig/xconfig/gconfig -- same as menuconfig, except that now the configuration menu is graphics based."qconfig" Requires the QT library.

8 **Once the configuration window is opened, you will see that a specific type of configuration is already selected like support for essential drivers like Broadcom wireless support/EXT4 filesystem etc**. Further, you may customize the options like adding support for your specific type of device/controller/driver like you may add support for NTFS file system from "Filesystem >> DOS/FAT/NT/ >> select NTFS file system support, thereby taking full advantage of custom kernel.

9 **NOTE**: While configuring the kernel, you will see a section known as kernel hacking(by hacking we mean exploring into it), where different types of options are given for hacking into kernel and learning it. If you want to use it then you may add further options, otherwise you may disable the option "kernel debugging", as it makes the kernel a lot heavier and and may be improper to use in the production environment.

10 **Once configured, it is time to compile and install the kernel**. You will need to execute these commands in order. This may take a long time.

- make (-j option may also be added to fork additional processes for compiling kernel,

syntax would be "make -j 3" 3 here represents the number of processes to be created)

- make modules_install
- make install

11 **now go grab some coffee cause this is going to take a while**. on a modern (-5 years old) computer this will take at least 20 minutes. you can make your life easier by entering your commands all at once. this is done by using the ampersand (&) symnbol

- make && make modules_install && make install

12 **So the kernel is installed, but you need to make it bootable**.

13 **Go to /boot**

14 **Run this command "mkinitrd -o initrd**.img-<kernelversion> <kernelversion>" (For Redhat based distros, you need not create initrd, as it is created by default) remembering to replace <kernelverison> with the version number of the kernel you build.

15 **Point the boot loader at the new kernel so it can be started**. Use the tool that came with your distro to configure your bootloader. Add a new entry for the new kernel.

16 **Reboot and enjoy your customized kernel**!

Tips

- menuconfig is usually the best option when configuring the kernel.
- not all computers require an initrd to be created, but it is safe to do so just in case yours does.

Warnings

- If the new kernel isn't configured properly, you may not support all your hardware and the kernel may crash.

Related wikiHows

- How to Get Your Nvidia Graphics Card Working on Linux
- How to Boot a Floppy Image Using the Grub Bootloader
- How to Choose Your First GNU/Linux Distribution
- How to Switch from Linux/UNIX to Windows