

**Configuring and Compiling the Linux Kernel**

***Objective: Configure and compile a Linux Kernel into an installable RPM*.**

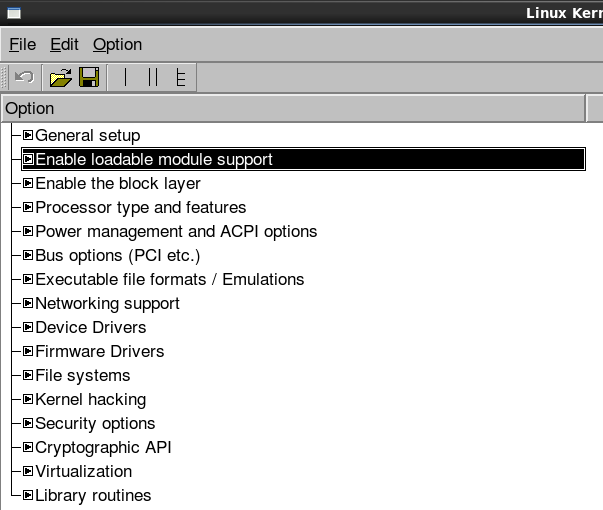
1. With the build root correctly set up, it's time to modify the kernel configuration (optional).
   * cd /home/student/rpmbuild
   * cd cd BUILD/kernel-3.10.0-123.el7/linux-`uname -r`(-r stands for Kernel release)

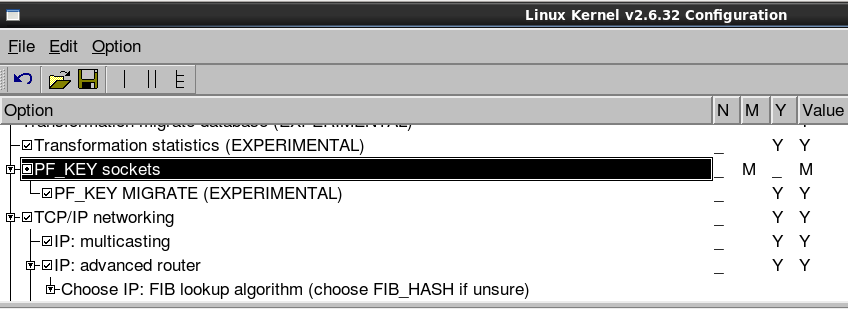
****

The XConfig menu system requires qt3 and qt3-devel packages:

* + sudo yum install qt3 qt3-devel libXi-devel
  + make xconfig   
    (make gconfig or make menuconfig)

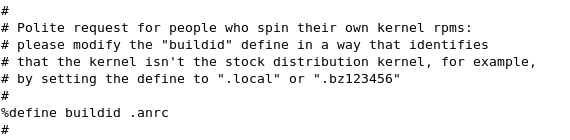
1. Alternatively, you can use the current running kernel’s configuration file (for this walkthrough use this step).
   * cp /boot/config-`uname -r` .config
   * make xconfig  
     to load and edit that files configuration



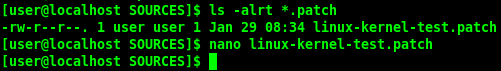


Spend a few minutes and navigate and get familiar with the basic structure and options available for CentOS Kernel customization.

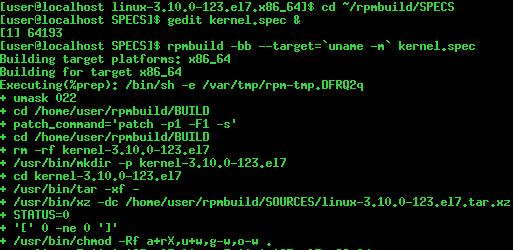
1. Copy the entire contents of the “configs/” directory to the “/student/src/redhat/SOURCES/“ directory.
   * sudo cp configs/\* /home/student/rpmbuild/SOURCES
2. Navigate to “/home/student/rpmbuild/SPECS/” and edit the file **kernel.spec** (nano, gedit, vi, ..etc.)
3. Search for “buildid” (nano CTRL+W, vi :/buildid). The definition of buildid is commented out. For custom kernels this must be uncommented and given a value to avoid a conflict with your currently installed kernel. Change the line in similar manner to the example below:



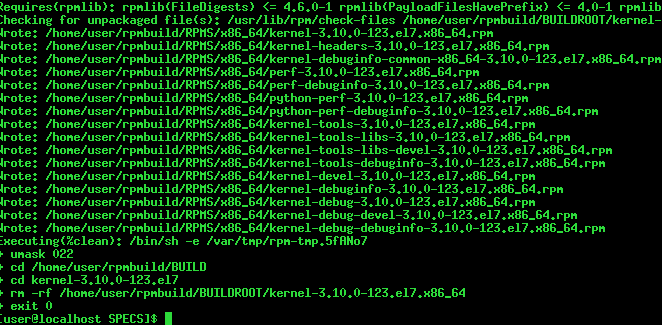
Since we are not building a custom kernel we can leave this commented out.

****

1. Now we are ready to build the kernel based on our configuration and patch options. Change directory to the /home/student/rpmbuild/SPECS folder and and execute the command:
   * rpmbuild -bb --target=`uname -m` kernel.spec



1. When the build completes (this could be a couple of hours depending on your computer), your custom kernel rpm files will be found in the **“**/home/student/rpmbuild/RPMS/`uname -m`/**”** directory.



*Note: If you have built a kernel version that is older than a currently installed version you will also have to use the --oldpackage flag with the rpm command.*

**Warning!** **UNDER NO CIRCUMSTANCES** use an rpm -Uvh command to install your kernel as this will update (overwrite) the currently installed version. Hence if you have a problem with your custom kernel, you will not be able to revert to the previous, working, version.