Instructions to Download, Compile and Install Linux Kernel.

Step 0: Get the Kernel Source.

- Go to https://www.kernel.org/
- Download the tarball of the **latest/stable** version into any directory within your home directory.
- Read more about distinguishing between a stable and development version here:
  - http://www.makelinux.net/books/lkd2/ch01lev1sec4
- Extract the tarball using
  - **tar -xvfJ linux-x.x.x.tar.xz**
- **cd** into the **linux-x.x.x** directory thus created.
- Lets call this your build directory.

Step 1: Configure, Build and Install Kernel.

- From the build directory configure the Kernel your are going to build using one of:
  - **make defconfig**
  - **make config**
  - **make menuconfig**
  - Read more about configuration commands here:
    - http://www.linux.org/threads/the-linux-kernel-configuring-the-kernel-part-1.4274/
    - http://www.linuxchix.org/content/courses/kernel_hacking/lesson2)
- Keeping the default options while configuring is fine.
- Now compile the Kernel using:
  - **make**

    or
  - **make -jx** (Where **x** = 2*no.of cores in your system) to speed up the build.
- Install the Kernel Modules
  - **make modules_install**
- Install the new Kernel onto the system using:
  - **make install**

- This creates:
  - kernel image/binary file by the name
    - **vmlinuz-x.x.x***
  - the **initramfs**(initial RAM file system) or **initrd**(initial RAM Disk) by the name
    - **initramfs-x.x.x***

      or
    - **initrd.img-x.x.x***
  - in your **/boot** directory.

- Here is a list of files of importance created in the **/boot** directory
  - vmlinuz-x.x.x* – The actual kernel

- System.map-x.x.x* – The symbols exported by the kernel
- initrd.img-x.x.x*/initramfs-x.x.x* – temporary root file system used during boot process
- config-x.x.x* – The kernel configuration file

- The **make install** command also executes a **update-grub** command which will make the grub (the **Gr**and **U**nified **B**oot Loader) aware of the new kernel image available
- and updates **/boot/grub/grub.cfg** file so that during the next boot the new kernel image shows up in the **GRUB menu.**

Step 2: Making your new Kernel the default entry in your GRUB menu.

- In the scenario where you are logging into a remote machine using, for eg., **ssh** or **telnet** you won't be presented with a grub menu. You'll be just logging into an already booted up machine.
- In order to ensure that you boot into the Kernel that we just built we need to make its entry the one that is selected by default in the grub menu.
- Open **/boot/grub/grub.cfg** in any of your favorite text editor.
- Locate the **menuentry** that has our **vmlinuz-x.x.x*** image file listed against the **linux** field inside the menuentry.
- Our aim is to boot into this menuentry by default.
- For example in the sample grub.cfg, we wish to make the menuentry
  - '**Ubuntu GNU/Linux, with Xen hypervisor**' the default one.
- Note that it is the **3rd** main menuentry item in the grub.cfg. Out of all these:
  - (0) menuentry 'Ubuntu'
  - (1) submenu 'Advanced options for Ubuntu'
  - **(2) menuentry 'Ubuntu GNU/Linux, with Xen hypervisor'**
  - (3) submenu 'Advanced options for Ubuntu GNU/Linux (with Xen hypervisor)'
- To achieve this open **/etc/default/grub** in a text editor and update the value of **GRUB_DEFAULT** to **2**
- That is make the following change:
  - **GRUB_DEFAULT=”2”**
- Then issue an **update-grub** command.
- Another example:
- We wish to make the menuentry
  - **'Ubuntu, with Linux 3.13.0-32-generic'**

- within the submenu item
  - '**Advanced options for Ubuntu**' the default;
- **submenu 'Advanced options for Ubuntu'** is the **2nd** main menuentry item in grub.cfg
- And **menuentry 'Ubuntu, with Linux 3.13.0-32-generic'** is the **3rd** sub-menuentry within that
  - (0) menuentry 'Ubuntu'
  - **(1) submenu 'Advanced options for Ubuntu'**
    - (0) menuentry 'Ubuntu, with Linux 3.13.0-39-generic'
    - (1) menuentry 'Ubuntu, with Linux 3.13.0-39-generic (recovery mode)'
    - **(2) menuentry 'Ubuntu, with Linux 3.13.0-32-generic'**
    - (3) menuentry 'Ubuntu, with Linux 3.13.0-32-generic (recovery mode)'
    - (4) menuentry 'Ubuntu, with Linux 3.3.1-mortar'
    - (5) menuentry 'Ubuntu, with Linux 3.3.1-mortar (recovery mode)'
  - (2) menuentry 'Ubuntu GNU/Linux, with Xen hypervisor'
  - (3) submenu 'Advanced options for Ubuntu GNU/Linux (with Xen hypervisor)'

- In order to boot into this we need to make the following change in **/etc/defaults/grub**
  - **GRUB_DEFAULT="1>2"**
- Follow it up by an **update-grub** command.
- For additional info look up https://help.ubuntu.com/community/Grub2/Submenus#Setting_a_Submenu_entry_as_the_default
- Once you've made all the changes reboot.
- Note that you won't be able to log into the remote system until it reboots.
- Once you've logged into the re-booted system check whether you've booted into the latest compiled kernel with
  - **uname -r** command.