Compiling the Kernel

1 Setting up the VM

1. Downloaded the latest version of virtualbox for Ubuntu as a debian package and installed it.
2. A virtual machine was created, with a vdi-image as hard-disk. Dynamic partition was used with a maximum size of 30 GB.
3. Downloaded Lubuntu 14.10 ISO image, and installed it in the virtual machine.
4. Inside the virtual machine lubuntu os, the package 'virtualbox-guest-dkms' to correct the screen size.

2 Compiling the Kernel

The following steps were done inside the virtual machine

1. Downloaded the latest linux kernel: **linux-3.18.1.tar.xz**, and moved it to `/usr/src`. The kernel source was extracted there to the subfolder "linux-3.18.1".
2. In that subfolder, the following commands were executed with root permission.
   (a) `make menuconfig`
   (To configure the kernel. Default settings were used without any alteration)
   (b) `make`
   (c) `make modules`
   (d) `make modules_install`
   (e) `make install`

   Approximately 20 GB HDD space was used for the entire process.
3. While rebooting, the system boots into the new kernel by default. If booting into an old kernel is necessary, pressing 'shift' key while booting displays the GRUB.

3 Setting up a Shared Folder

1. In the host os, create a directory to be shared. For my system, it is '/media/pp/MM/vboxshare'

2. Install the 'Virtualbox Guest Additions' into the VM using the following procedure.

   (a) In the VM, click on "Devices" menu and select "Insert Guest additions CD image"

   (b) Go to that CD directory, and execute the following command
   `sudo sh ./VBoxLinuxAdditions.run`
   this will install all the necessary packages in the guest os.

3. In the VM, click on "Devices" menu, and select "Shared Folders Settings". In that, select the option to add a new folder. Select the previously set HOST COMPUTER's folder to create a named share. The name will be same as the directory name by default - for me it is 'vboxshare'.
   'Automount' and 'make permanent' should be checked for future convenience.

4. In the VM, add the current user to the group 'vboxsf'. This will take care of the permissions of current user to access that share. Command for my system was:
   `sudo adduser pp vboxsf`
   On every boot, the folder will be mounted at '`/media/sf_<share-name>`'
   For my system, it was at '`/media/sf_vboxshare`'

   Alternatively,
   a shared folder can be mounted using the following command:
   `sudo mount -t vboxsf <share-name><mount-point>`
   example:
   `sudo mount -t vboxsf vboxshare  /shared_folder`