Experiment Number: 15

TITLE: Linux Kernel Compilation.

OBJECTIVE:

1. To study the Linux kernel compilation process.

Theory:

Step # 1 Get Latest Linux kernel code

Visit http://kernel.org/ and download the latest source code. File name would be linux-x.y.z.tar.bz2, where x.y.z is actual version number. For example file inux-2.6.25.tar.bz2 represents 2.6.25 kernel version. Use wget command to download kernel source code:

```
$ cd /tmp
$ wget http://www.kernel.org/pub/linux/kernel/v2.6/linux-
x.y.z.tar.bz2
```

Note: Replace x.y.z with actual version number.

Step # 2 Extract tar (.tar.bz3) file

Type the following command:

```
# tar -xjvf linux-2.6.25.tar.bz2 -C /usr/src
# cd /usr/src
```

Step # 3 Configure kernel

Before we configure kernel make sure we have development tools (gcc compilers and related tools) are installed on our system. If gcc compiler and tools are not installed then use apt-get command under Debian Linux to install development tools.

```
# apt-get install gcc
```

Now we can start kernel configuration by typing any one of the command:
`$ make menuconfig` - Text based color menus, radiolists & dialogs. This option also useful on remote server if you wanna compile kernel remotely.
`$ make xconfig` - X windows (Qt) based configuration tool, works best under KDE desktop
`$ make gconfig` - X windows (Gtk) based configuration tool, works best under Gnome Dekstop.
For example make menuconfig command launches following screen:
`$ make menuconfig`

We have to select different options as per your need. Each configuration option has HELP button associated with it so select help button to get help.

Step # 4 Compile kernel

Start compiling to create a compressed kernel image, enter:

```
$ make
```

Start compiling to kernel modules:

```
$ make modules
```

Install kernel modules (become a root user, use su command):

```
$ su -
# make modules_install
```

Step # 5 Install kernel

So far we have compiled kernel and installed kernel modules. It is time to install kernel itself.

```
# make install
```

It will install three files into /boot directory as well as modification to your kernel grub configuration file:

- System.map-2.6.25
- config-2.6.25
- vmlinuz-2.6.25

Step # 6: Create an initrd image

Type the following command at a shell prompt:

```
# cd /boot
# mkinitrd -o initrd.img-2.6.25 2.6.25
```

initrd images contains device driver which needed to load rest of the operating system later on. Not all computer requires initrd, but it is safe to create one.

Step # 7 Modify Grub configuration file - /boot/grub/menu.lst

Open file using vi:

```
# vi /boot/grub/menu.lst
title           Debian GNU/Linux, kernel 2.6.25 Default
root            (hd0,0)
kernel          /boot/vmlinuz root=/dev/hdb1 ro
initrd          /boot/initrd.img-2.6.25
savedefault
boot
```

Remember to setup correct root=/dev/hdXX device. Save and close the file. If you think editing and writing all lines by hand is too much for us, try out update-grub command to update the lines for each kernel in /boot/grub/menu.lst file. Just type the command:

```
# update-grub
```

Step # 8 : Reboot computer and boot into your new kernel

Just issue a reboot command:

```
# reboot
```

References:

1. www.cyberciti.biz/tips/compiling-linux-kernel-26.html
2. www.sysdesign.ca/guides/linux_kernel.html

Tushar B Kute
(Subject Teacher)