**Steps used to add "This is My Shell" system call in our kernel.**

1. Download the kernel Source.

I used linux kernel-3.16

2. Extract the kernel source code

Extract the kernel source code from the linux-3.16.tar.xz file in /usr/src/directory

sudo tar -xvf linux-3.16.tar.xz -C/usr/src/

/\*sudo will gain access And tar will store and extract files from a disk archive.\*/

cd/usr/src/linux-3.16/

/\*After extraction changing to kernel source directory\*/

3. Defining a new system call sys\_shell()

mkdir MyShell

cd MyShell

/\*Create a directory Myshell in the kernel source directory and change into this directory\*/

A)Create a "MyShell.c" file in this folder and add the definition of the system call.

gedit MyShell.c

#include<linux/kernel.h>

asmlinkage long sys\_shell(void)

{

printk("This is MyShell"); //printk prints the kernel's log file.

return 0;

}

B) Create a "Makefile' in the MyShell folder and add the following line

gedit Makefile

obj-y := MyShell.o /\* This is to ensure that the MyShell.c file is compiled and included in the kernel source code \*/

4. Add the MyShell directory to the kernel's Makefile

change back into the linux-3.16 folder and open Makefile

gedit Makefile

//change the following line.

"core-y +=kernel/mm/fs/ipc/security/crypto/block/" //line 842

//change to

"core-y +=kernel/mm/fs/ipc/security/crypto/block/MyShell"

//line 842

/\* This is to tell the compiler that the source files of our new system call(sys\_shell()) are in present in the MyShell directory.

5. Now the new system call (sys\_shell()) needs to be added into teh system call table(syscall\_64.tbl file)

cd arch/x64/syscalls

gedit syscall\_64.tbl

//add the following at the end of the file.

355 i364 MyShell sys\_shell

/\*Number of system calls now present in this file is 355\*/

6. Add the new system call (sys\_shell()) in the system call header file.

cd include/linux/

gedit syscalls.h

asmlinkage long sys\_shell(void);

//prototype of the function of our system

7. Now we need to compile this kernel on the system.

sudo apt-get install gcc

sudo apt-get install libncurses5-dev

sudo apt-get update

sudo apt-get upgrade

/\*the above commands are used to install the latest version of gcc, install ncurses development package and to update the system packages. \*/

sudo make menuconfig //configures our kernel...

/\* now to compile the kernel do make\*/

cd/usr/src/linux-3.16/

make

8. To install or update the kernel

sudo make modules\_install install

/\* The above command will install the linux kernel 3.16 into the system. It will create some files under /boot/directory and it will automatically make a entry in the grub.cfg. To check whether it made correct entry, we need to check the files under /boot/directory .

/\* Now to update the kernel reboot the system. \*/

shutdown -r now

uname -r //verifying kernel version.

9. TEST THE SYSTEM CALL.

Creating a "test.c" program in our home folder and type the following:

#include<stdio.h>

#include<linux/kernel.h>

#include<sys/syscall.h>

#include<unistd.h>

int main()

{

long int val=syscall(355);

printf("System call sys\_shell returned %ld\n",val);

return 0;

}

Now compile the above program and the message will be displayed.

gcc test.c

./a.out

OUTPUT

System call sys\_shell returned 0"

Now to check the message of the kernel we run the following command.

dmesg