The following steps are on how to add a new syscall to 32-bit linux kernel (3.0)

1) open terminal and login as root (*su*)
2) *apt-get update*
3) *cd /usr/src*
4) *wget http://www.kernel.org/pub/linux/kernel/v3.0/linux-3.0.1.tar.bz2*
5) *tar xjf linux-3.0.1.tar.bz2*
6) *ln -s linux-3.0.1.tar.bz2 linux*
7) *cd /usr/src/linux*
8) *cp /boot/config-`uname-r` ./.config*
9) *make menuconfig* -> load an alternate configuration file -> ./.config -> exit
10) add the following method to /usr/src/linux/kernel/module.c:

```
asmlinkage long sys_modcount(void)
{
        int count = 0;
        struct module *mod;
        list_for_each_entry(mod, &modules, list) {
                count++;
        }
        return count;
}
```

11) add the following line to the end of the file /usr/src/linux/include/linux/syscalls.h:

```
asmlinkage long sys_modcount(void);
```

12) add the following in **bold** to /usr/src/linux/arch/x86/include/asm/unistd_32.h:

```
#define __NR_modcount 347
#define NR_syscalls 348
```

13) add the following line to the end of the file /usr/src/linux/arch/x86/kernel/syscall_table_32.S:

```
.long sys_modcount
```

14) build the kernel (this may take a while)

make-kpkg clean
fakeroot make-kpkg --initrd --append-to-version=-custom kernel_image kernel_headers

15) install the new kernel

cd /usr/src
dpkg -i linux-image-3.0.1-custom_3.0.1-custom-10.00.Custom_i386.deb
dpkg -i linux-headers-3.0.1-custom_3.0.1-custom-10.00.Custom_i386.deb

16) reboot the system

shutdown -r now

17) test the new system call

a) create a test program (*vi test.c*) and add the following:

```
#include <stdio.h>
long modcount(void)
{
        asm("movl $347, %eax");
        asm("int $0x80");
}
int main(void)
{
        int count = modcount();
        printf("\nNumber Modules=%d", count);
}
```

12) open a new terminal and monitor /var/log/syslog

tail -f /var/log/syslog

13) compile test program and run it

gcc test.c -o test
./test