首先在终端执行

sudo apt-get install nautilus-open-terminal

sudo apt-get install nautilus-gksu

然后就可以右键在终端打开和右键以管理员打开文件了，如果习惯用终端打开文件的话也可以跳过这一步

同时建议使用 VMWare 的童鞋安装 VMWare tools 以便在宿主机与虚拟机之间复制数据

中途如果出现以下错误

greb: /boot/config-XX.XX.XX : no such file or directory

请无视，没什么影响

进行下面的操作前可以先执行 sudo su 获取权限

1、下载内核

apt-get install linux-source

我下载到的内核文件是 linux-source-2.6.35.tar.bz2，在/usr/src 下

2、解压内核

cd /usr/src

tar -jxvf linux-source-2.6.35.tar.bz2

将内核解压到/usr/src

3、修改文件

a）/usr/src/linux-source-2.6.35/kernel/sys.c

可以用右键管理员打开或者

gedit /usr/src/linux-source-2.6.35/kernel/sys.c

然后简单起见按照第一篇文章的函数修改 sys.c 文件

在文件最后添加

```
asmlinkage int sys_mycall(int number)
{
 printk("这是我添加的第一个系统调用");
 return number;
}
asmlinkage int sys_addtotal(int number)
{
 int i=0,enddate=0;
 printk("这是我添加的第二个系统调用");
 while(i<=number)
  enddate+=i++;
 return enddate;
}
asmlinkage int sys_three()
{
 printk("这是我添加的第三个系统调用");
```

```
 return 0;
}
```

b）/usr/src/linux-source-2.6.35/arch/x86/kernel/syscall_table_32.S
可以用右键管理员打开或者
gedit /usr/src/linux-source-2.6.35/arch/x86/kernel/syscall_table_32.S

然后在一列 .long sys_XXXX 的下一行添加
.long sys_mycall
.long sys_addtotal
.long sys_three
并记住他们分别是第几个.long sys_XXXX

c）/usr/src/linux-source-2.6.35/arch/x86/include/asm/unistd_32.h
可以用右键管理员打开或者
gedit /usr/src/linux-source-2.6.35/arch/x86/include/asm/unistd_32.h

在一列#define __NR_XXXX NNN 后添加几行
#define __NR_mycall 338
#define __NR_addtotal 339
#define __NR_three 340
注意后面那个数字是接着上面那几行下来的

d）/usr/src/linux-source-2.6.35/ubuntu/omnibook/Makefile
这个是教程里面没有的，不一定要改，如果在后面编译模块时出现
ld: /ubuntu/omnibook/sections.lds: No such file: No such file or directory
make[2]: *** [ubuntu/omnibook/omnibook.o] Error 1
make[1]: *** [ubuntu/omnibook] Error 2
make: *** [ubuntu] Error 2
以上错误，则需要修改，不过我建议还是先改了再说

可以用右键管理员打开或者
gedit /usr/src/linux-source-2.6.35/ubuntu/omnibook/Makefile

找到下面两行
#EXTRA_LDFLAGS += $(src)/sections.lds
EXTRA_LDFLAGS += $(PWD)/ubuntu/omnibook/sections.lds

调换一下‘#’的位置，如下
EXTRA_LDFLAGS += $(src)/sections.lds
#EXTRA_LDFLAGS += $(PWD)/ubuntu/omnibook/sections.lds

4、编译内核
这里也是按照教程，首先进入解压目录

cd /usr/src/linux-source-2.6.35

a）make mrproper //清除内核中不稳定的目标文件，附属文件及内核配置文件

b）make clean //清除以前生成的目标文件和其他文件

c）make oldconfig// 采用默认的内核配置，如果这里出现选项，选择默认的选项，就是方括号内的第一个字母，不过我这里没出现选项

d）make bzImage //编译内核，大概需要半小时

e）make modules //编译模块，大概需要两小时，如果出现错误，看看是不是因为上面的第3步的 d）没做

f）make modules_install// 安装模块，比较快

5、复制内核

首先查看一下编译好的内核版本，以便命名

打开 /lib/modules

里面应该多了一个纯数字不带"generic"的文件夹，那就是新内核版本号，我的是2.6.35.11

然后复制内核

cp /usr/src/linux-source-2.6.35/arch/i386/boot/bzImage /boot/vmlinuz-2.6.35.11-mykernel

6、创建 initrd 文件

在创建之前先安装必要的程序

apt-get install bootcd-mkinitramfs

mkinitramfs -o /boot/initrd.img-2.6.35.11

耐心等待创建完成

7、更新配置 GRUB 引导列表

可以用右键管理员打开/boot/grub/grub.cfg 或者

gedit /boot/grub/grub.cfg

找到下面这种结构

```
menuentry 'Ubuntu, with Linux 2.6.35-27-generic' --class ubuntu --class gnu-linux --class gnu --class os {
    recordfail
    insmod part_msdos
    insmod ext2
    set root='(hd1,msdos1)'
    search --no-floppy --fs-uuid --set 71a50d19-caef-4dff-9a7a-57cb1bbfe0c2
    linux    /boot/vmlinuz-2.6.35-27-generic root=UUID=71a50d19-caef-4dff-9a7a-57cb1bbfe0c2 ro    quiet
splash
    initrd    /boot/initrd.img-2.6.35-27-generic
}
menuentry 'Ubuntu, with Linux 2.6.35-27-generic (recovery mode)' --class ubuntu --class gnu-linux --class
gnu --class os {
    recordfail
    insmod part_msdos
    insmod ext2
    set root='(hd1,msdos1)'
    search --no-floppy --fs-uuid --set 71a50d19-caef-4dff-9a7a-57cb1bbfe0c2
```

```
    echo    'Loading Linux 2.6.35-27-generic ...'
    linux    /boot/vmlinuz-2.6.35-27-generic root=UUID=71a50d19-caef-4dff-9a7a-57cb1bbfe0c2 ro single
    echo    'Loading initial ramdisk ...'
    initrd    /boot/initrd.img-2.6.35-27-generic
}
```

复制一份在这些结构前粘贴，注意必须贴在

```
### BEGIN /etc/grub.d/10_linux ###
……
### END /etc/grub.d/10_linux ###
```

里面，并将粘贴出来的结构中的

```
linux    /boot/vmlinuz-2.6.35-27-generic
initrd    /boot/initrd.img-2.6.35-27-generic
```

改成你的内核文件地址和 initrd 文件地址

```
linux    /boot/vmlinuz-2.6.35.11-mykernel
initrd    /boot/initrd.img-2.6.35.11
```

最好把其他所有版本相关信息号改成2.6.35.11（新编译的版本号），以便在 Grub 菜单选择

比如改成

```
menuentry 'Ubuntu, with Linux 2.6.35.11' --class ubuntu --class gnu-linux --class gnu --class os {
    recordfail
    insmod part_msdos
    insmod ext2
    set root='(hd1,msdos1)'
    search --no-floppy --fs-uuid --set 71a50d19-caef-4dff-9a7a-57cb1bbfe0c2
    linux    /boot/vmlinuz-2.6.35.11-mykernel root=UUID=71a50d19-caef-4dff-9a7a-57cb1bbfe0c2 ro    quiet
splash
    initrd    /boot/initrd.img-2.6.35.11
}
menuentry 'Ubuntu, with Linux 2.6.35.11 (recovery mode)' --class ubuntu --class gnu-linux --class gnu --
class os {
    recordfail
    insmod part_msdos
    insmod ext2
    set root='(hd1,msdos1)'
    search --no-floppy --fs-uuid --set 71a50d19-caef-4dff-9a7a-57cb1bbfe0c2
    echo    'Loading Linux 2.6.35.11 ...'
    linux    /boot/vmlinuz-2.6.35.11-mykernel root=UUID=71a50d19-caef-4dff-9a7a-57cb1bbfe0c2 ro single
    echo    'Loading initial ramdisk ...'
    initrd    /boot/initrd.img-2.6.35.11
}
```

注意检查一下/boot/目录下是否存在上面这两个文件，如果没有，证明上面的几部还没成功

还有就是这些 menuentry 的顺序，有些系统启动引导时会直接进入第一个 menuentry，如果第一个 menuentry 不是你想进的内核，则需要在开机时按 Shift 进入 GRUB 引导菜单选择内核。如果你的系统开机是直接打开 GRUB 引导菜单，则无所谓，选择你想进的内核就行

8、最后的一点工作
到这里就差不多了，但是如果你现在就重启很可能会出错
warning:can't open directory /lib/modules/2.6.35/modules.dep
如果按照第一篇文章所说的执行 update-grub2，重启后极有可能出现 kernel panic
实际上不用执行这个命令

继续在终端执行以下命令（参考最后一篇文章）

```
cd /boot
cp initrd.img-2.6.35.11 initrd-2.6.35.11.old
```

以上是备份 initrid，下面是修改

```
depmod -a
update-initramfs -k 2.6.35.11 -c
cd /tmp
gzip -dc /boot/initrd.img-2.6.35.11| cpio -id
touch lib/modules/2.6.35.11/modules.dep
find ./ | cpio -H newc -o > /boot/initrd.img-2.6.35.11.new
gzip /boot/initrd.img-2.6.35.11.new
cd /boot
mv initrd.img-2.6.35.11.new.gz initrd.img-2.6.35.11
```

9、重启
重启 ubuntu，如果能进入系统，证明基本上没问题了。
打开终端输入

```
uname -a
```

查看版本号，如果是2.6.35.11就可以安心了

10、检查系统调用
打开编译器，新建工程来测试
在 main 函数中使用 syscall(函数号，参数)测试系统调用
比如

```
syscall(338,1);
syscall(339,10);
syscall(340);
```

编译运行，然后打开终端输入

```
dmesg -c
```

查看是否有添加系统调用成功信息