# OS Project 1

Project for Computer Architecture & Operating Systems by Chentao Wu, 2016 Autumn Semester

## Project Step:

### Part1 Preparation

1. Build Ubuntu 12.04

   The version of Ubuntu 12.04 kernel is 3.2.0-23.

2. Download linux-2.6.38

   Download kernel from http://www.kernel.org. I used the 2.6.38 version because I have heard that it is easier to manipulate.

3. Open termination, input command in termination, get root power.

```
sudo su
```

### Part2 Add a system call to the linux kernel

1. Unzip the kernel

```
mv linux-2.6.38.tar.bz2 /usr/src/
cd /usr/src
tar -jxvf linux-2.6.38.tar.bz2
```

   Move the zip file to the "/usr/src"directory, where needs the root power. Then unzip it.

2. Add the system call

```
cd linux-2.6.38/kernel
gedit sys.c
```

   open sys.c file, add header at the beginning of the file:

```
#include<linux/linkage.h>
#include<linux/kernel.h>
```

   Add the code at the end of the file

```
asmlinkage int sys_helloworld(){
        printk(KERN_EMERG "hello world!(by Yaowei Huang,Nov.2016)");
        return 1;
}
```

   And there I have made a small mistake..It's October now...

3. Modify the pointer list

```
cd /usr/src/linux-2.6.38/arch/x86/kernel
gedit syscall_table_32.S
```

open syscall_table_32.S file, add the code at the end of it.

```
.long sys_helloworld
```

```
cd /usr/src/linux-2.6.38/arch/x86/include/asm
gedit unistd_32.h
```

open unistd_32.h file, at the end of it add the code:

```
#define __NR_helloworld        341
```

And change the code:

```
#define __NR_syscalls        341
```

to

```
#define __NR_syscalls        342
```

### Part 3 Compile the kernel

1. Preparaion

```
cd /usr/src/linux-2.6.38
apt-get install build-essential kernel-package libncurses5-
    dev fakeroot
```

2. Compile

```
make mrproper
//it is alternative, to clean the compiler history
make menuconfig
//to generate a .config file
make -j4
//the command of compiling
```

And then it was a long time to wait, I waited about two and a half hours.

### Part 4 Install the kernel

1. Install kernel

```
make modules_install
make install
```

2. Make the new kernel to be loaded

```
sudo mkinitramfs -o /book/initrd.img-2.6.38
sudo update-initramfs -c -k 2.6.38
sudo update-grub2
```

The first command will create files like initrd.img-2.6.38 etc. under /boot list.

The second command will update the files corresponding to /lib/modules/2.6.32.63 files.

3. Add chice to the grup list

Change the "GRUB_HIDDEN_TIMEOUT=0" in the file of /etc/default/grub to "GRUB_HIDDEN

Change all "set timeout=0" in the file of /etc/grub.d/30_os-prober to "set timeout=10".

Update grup2

```
sudo update-grub2
```

**Part 5 Reboot**

Reboot the ubuntu and choose to enter the 2.6.38 version kernel.

## Validate

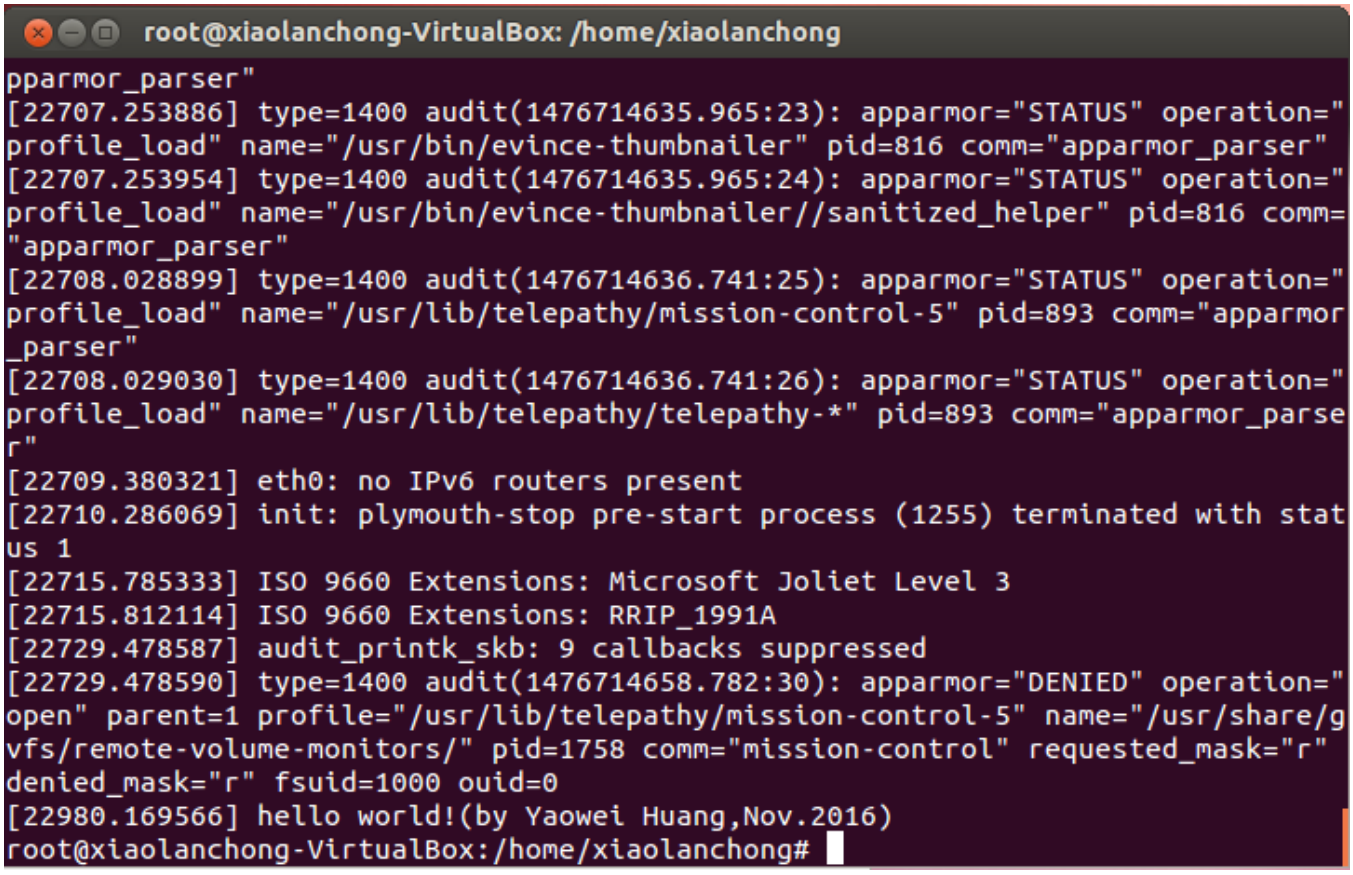
```c
#include<stdio.h>
#include<unistd.h>
#include<sys/syscall.h>

#define SYS_helloworld    341
//same as defined before

 int main(){
        int tmp;
        tmp=syscall(341);
        printf("\n");
        if(tmp==1){
                printf("success!\n");
        }
        return 0;
}
```

Input command in termination

```
gcc a.c
ll a.c a.out
./a.out
dmesg -d
```

## Result

The pictrue of result is as follows:



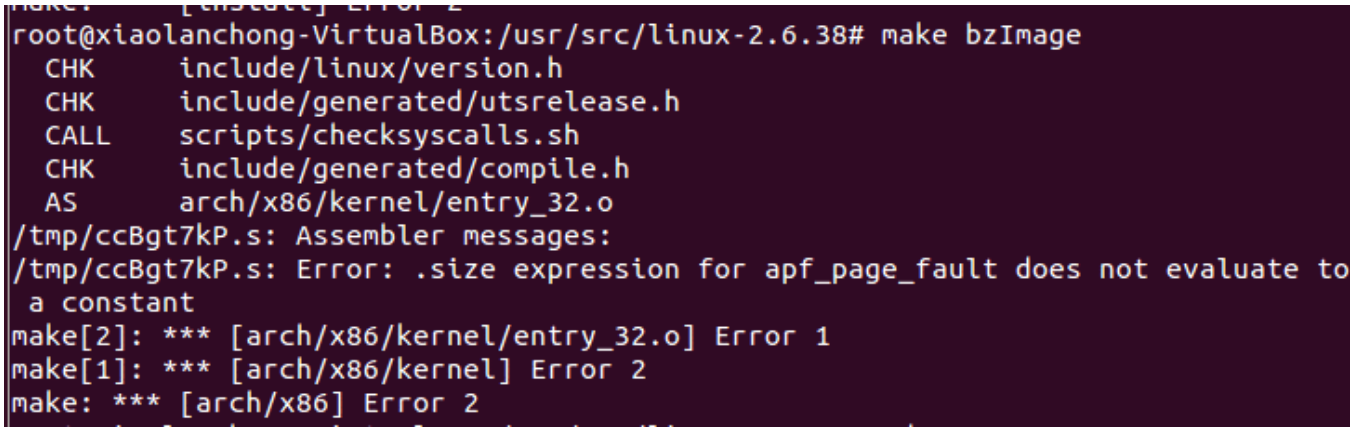
```
root@xiaolanchong-VirtualBox: /home/xiaolanchong
pparmor_parser"
[22707.253886] type=1400 audit(1476714635.965:23): apparmor="STATUS" operation="
profile_load" name="/usr/bin/evince-thumbnailer" pid=816 comm="apparmor_parser"
[22707.253954] type=1400 audit(1476714635.965:24): apparmor="STATUS" operation="
profile_load" name="/usr/bin/evince-thumbnailer//sanitized_helper" pid=816 comm=
"apparmor_parser"
[22708.028899] type=1400 audit(1476714636.741:25): apparmor="STATUS" operation="
profile_load" name="/usr/lib/telepathy/mission-control-5" pid=893 comm="apparmor
_parser"
[22708.029030] type=1400 audit(1476714636.741:26): apparmor="STATUS" operation="
profile_load" name="/usr/lib/telepathy/telepathy-*" pid=893 comm="apparmor_parse
r"
[22709.380321] eth0: no IPv6 routers present
[22710.286069] init: plymouth-stop pre-start process (1255) terminated with stat
us 1
[22715.785333] ISO 9660 Extensions: Microsoft Joliet Level 3
[22715.812114] ISO 9660 Extensions: RRIP_1991A
[22729.478587] audit_printk_skb: 9 callbacks suppressed
[22729.478590] type=1400 audit(1476714658.782:30): apparmor="DENIED" operation="
open" parent=1 profile="/usr/lib/telepathy/mission-control-5" name="/usr/share/g
vfs/remote-volume-monitors/" pid=1758 comm="mission-control" requested_mask="r"
denied_mask="r" fsuid=1000 ouid=0
[22980.169566] hello world!(by Yaowei Huang,Nov.2016)
root@xiaolanchong-VirtualBox:/home/xiaolanchong#
```

图 1: Result

## The problem I have met

The most difficult problem is the following error:

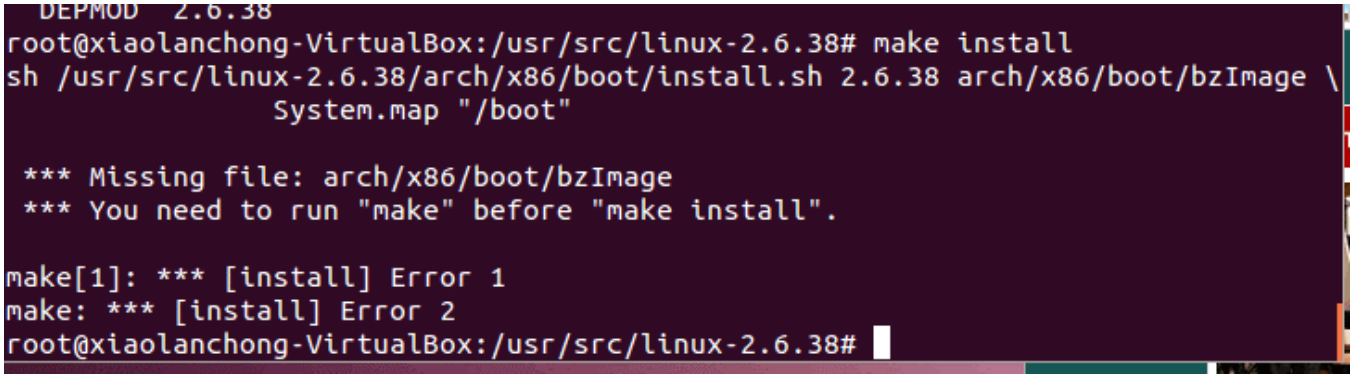
.size expression for apf_fault does not evaluate to a constant



```
root@xiaolanchong-VirtualBox:/usr/src/linux-2.6.38# make bzImage
  CHK     include/linux/version.h
  CHK     include/generated/utsrelease.h
  CALL    scripts/checksyscalls.sh
  CHK     include/generated/compile.h
  AS      arch/x86/kernel/entry_32.o
/tmp/ccBgt7kP.s: Assembler messages:
/tmp/ccBgt7kP.s: Error: .size expression for apf_page_fault does not evaluate to
 a constant
make[2]: *** [arch/x86/kernel/entry_32.o] Error 1
make[1]: *** [arch/x86/kernel] Error 2
make: *** [arch/x86] Error 2
```

图 2: Problem

And at first I didn't find this problem, because I was not using the command "make bzImage", then I found that the command "make -j4" runs very fast, which only needed about 10 minutes, much shorter than my roommates. Later I found I can't "make install".



```
  DEPMOD  2.6.38
root@xiaolanchong-VirtualBox:/usr/src/linux-2.6.38# make install
sh /usr/src/linux-2.6.38/arch/x86/boot/install.sh 2.6.38 arch/x86/boot/bzImage \
                System.map "/boot"

 *** Missing file: arch/x86/boot/bzImage
 *** You need to run "make" before "make install".

make[1]: *** [install] Error 1
make: *** [install] Error 2
root@xiaolanchong-VirtualBox:/usr/src/linux-2.6.38#
```

图 3: Problem

So I think there must be something wrong, and I searched the website to find more command to try, until I tried the "make bzImage" command to find the problem.

Fortunately, there's someone having solved this problem. I entered the website

http://blog.csdn.net/baozhb/article/details/9005150

The solution is to modify two lines of codes (code mistake) in /.../arch/x86/kernel/entry_32.S. And then the problem is solved quickly.

What's more, I also have met a stupid problem that I didn't allocate enough virtual disks space to my Ubuntu...

## Harvest

Through this project I learned a lot knowledge about the kernel and how to add a system call. Although I haven't done as the same as what the book recommend, but I think my way is as good too, which is more simple. The operating system is a very fun thing to manipulate, and solving problems makes me very proudable.