1、 编译内核

1） make mrproper（使用的内核版本是 3.16.51）

2） make config

3） make deep

4） make bzImage

make modules

```
  IHEX    firmware/kaweth/trigger_code_fix.bin
  IHEX    firmware/ti_3410.fw
  IHEX    firmware/ti_5052.fw
  IHEX    firmware/mts_cdma.fw
  IHEX    firmware/mts_edge.fw
  IHEX    firmware/mts_gsm.fw
  H16TOFW firmware/edgeport/boot.fw
  H16TOFW firmware/edgeport/down.fw
  H16TOFW firmware/edgeport/boot2.fw
  IHEX    firmware/edgeport/down3.bin
  H16TOFW firmware/edgeport/down2.fw
  IHEX2FW firmware/whiteheat_loader.fw
  IHEX2FW firmware/whiteheat.fw
  IHEX2FW firmware/keyspan_pda/keyspan_pda.fw
  IHEX2FW firmware/keyspan_pda/xircom_pgs.fw
  H16TOFW firmware/matrox/g200_warp.fw
  H16TOFW firmware/matrox/g400_warp.fw
[root@localhost linux-3.16.51]#
```

make modules_install

```
  INSTALL /lib/firmware/kaweth/new_code.bin
  INSTALL /lib/firmware/kaweth/trigger_code.bin
  INSTALL /lib/firmware/kaweth/new_code_fix.bin
  INSTALL /lib/firmware/kaweth/trigger_code_fix.bin
  INSTALL /lib/firmware/ti_5052.fw
  INSTALL /lib/firmware/mts_cdma.fw
  INSTALL /lib/firmware/ti_3410.fw
  INSTALL /lib/firmware/mts_gsm.fw
  INSTALL /lib/firmware/mts_edge.fw
  INSTALL /lib/firmware/edgeport/boot.fw
  INSTALL /lib/firmware/edgeport/boot2.fw
  INSTALL /lib/firmware/edgeport/down.fw
  INSTALL /lib/firmware/edgeport/down3.bin
  INSTALL /lib/firmware/edgeport/down2.fw
  INSTALL /lib/firmware/whiteheat_loader.fw
  INSTALL /lib/firmware/whiteheat.fw
  INSTALL /lib/firmware/keyspan_pda/keyspan_pda.fw
  INSTALL /lib/firmware/keyspan_pda/xircom_pgs.fw
  DEPMOD  3.16.51
[root@localhost linux-3.16.51]#
```

depmod –a

```
  INSTALL /lib/firmware/whiteheat_loader.fw
  INSTALL /lib/firmware/whiteheat.fw
  INSTALL /lib/firmware/keyspan_pda/keyspan_pda.fw
  INSTALL /lib/firmware/keyspan_pda/xircom_pgs.fw
  DEPMOD  3.16.51
[root@localhost linux-3.16.51]# depmod -a
```

部署编译好的内核

```
[root@localhost linux-3.16.51]# cp System.map /boot/
[root@localhost linux-3.16.51]# cp arch/x86/boot/bzImage /boot/
[root@localhost linux-3.16.51]#
```

在/lib/modules 文件夹下有与编译内核相应的文件夹

```
[root@localhost 151020216]# cd /lib/modules/
[root@localhost modules]# ls
2.6.32-358.el6.x86_64  3.16.51
[root@localhost modules]#
```

建立 ramdisk 映像文件

```
initrd-2.6.32-358.el6.x86_64kdump.img
[root@localhost boot]# mkinitrd initrd-3.16.51.img 3.16.51
[root@localhost boot]#
```

boot 目录下的文件

```
[root@localhost 151020216]# ls /boot/
bzImage                                 initrd-3.16.51.img
config-2.6.32-358.el6.x86_64            lost+found
efi                                     symvers-2.6.32-358.el6.x86_64.gz
grub                                    System.map
initramfs-2.6.32-358.el6.x86_64.img     System.map-2.6.32-358.el6.x86_64
initrd-2.6.32-358.el6.x86_64kdump.img   vmlinuz-2.6.32-358.el6.x86_64
[root@localhost 151020216]#
```

grub 配置文件

```
    21  kernel /bzImage ro root=/dev/mapper/VolGroup-lv_root rd_NO_LUKS LANG=en_
US.UTF-8 rd_NO_MD rd_LVM_LV=VolGroup/lv_swap SYSFONT=latarcyrheb-sun16 crashkern
el=128M rd_LVM_LV=VolGroup/lv_root  KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb qu
iet
    22  myparam=1
    23  initrd /initrd-3.16.51.img
[root@localhost 151020216]#
```

从新内核启动成功

```
File  Edit  View  Search  Terminal  Help
[151020216@localhost ~]$ uname -a
Linux localhost.localdomain 3.16.51 #1 SMP Tue Dec 5 20:52:55 EST 2017 x86_64 x8
6_64 x86_64 GNU/Linux
[151020216@localhost ~]$
```

3、添加系统调用

1) 用 Vim 打开*/kernel/sys.c* 文件,添加头文件

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

添加调用实现

```
asmlinkage int sys_ourcall(void)
{
        printk("I am zhanghaijie,151020216");
        return 100;
}
#endif /* CONFIG_COMPAT */
~
~
```

2）添加函数声明

151020216@localhost:/usr/src/kernels/linux-3.16.51/arch/x86/include/asm

File Edit View Search Terminal Help

```
/*
 * syscalls.h - Linux syscall interfaces (arch-specific)
 *
 * Copyright (c) 2008 Jaswinder Singh Rajput
 *
 * This file is released under the GPLv2.
 * See the file COPYING for more details.
 */

#ifndef _ASM_X86_SYSCALLS_H
#define _ASM_X86_SYSCALLS_H

#include <linux/compiler.h>
#include <linux/linkage.h>
#include <linux/signal.h>
#include <linux/types.h>

/* Common in X86_32 and X86_64 */
/* kernel/ioport.c */
asmlinkage long sys_ioperm(unsigned long, unsigned long, int);
asmlinkage long sys_iopl(unsigned int);
asmlinkage int sys_ourcall(void);
/* kernel/ldt.c */
```

3）加入系统调用号

151020216@localhost:/usr/src/kernels/linux-3.16.51/arch/x86/syscalls

File Edit View Search Terminal Help

```
306     common  syncfs                  sys_syncfs
307     64      sendmmsg                sys_sendmmsg
308     common  setns                   sys_setns
309     common  getcpu                  sys_getcpu
310     64      process_vm_readv        sys_process_vm_readv
311     64      process_vm_writev       sys_process_vm_writev
312     common  kcmp                    sys_kcmp
313     common  finit_module            sys_finit_module
314     common  sched_setattr           sys_sched_setattr
315     common  sched_getattr           sys_sched_getattr
316     common  renameat2               sys_renameat2
317     64      ourcall                 sys_ourcall
#
```

重新编译后安装重启进行测试，测试程序如下

```c
#include<stdio.h>
#include<unistd.h>
#include<sys/syscall.h>
int main(void)
{
        int i=syscall(317);
        printf("%d\n",i);
        return 0;
}
~
~
~
~
```

测试程序输出

```
[root@localhost 151020216]# gcc -o ourcall ourcall.c
[root@localhost 151020216]# ./ourcall
100
[root@localhost 151020216]#
```

内核输出：

```
[root@localhost 151020216]# ./ourcall
100
[root@localhost 151020216]# dmesg
[root@localhost 151020216]# dmesg
[root@localhost 151020216]# dmesg  -c
[root@localhost 151020216]# ./ourcall
100
[root@localhost 151020216]# dmesg  -c
[ 1631.389397] I am zhanghaijie,151020216
[ 1658.887750] I am zhanghaijie,151020216
[root@localhost 151020216]#
```

添加系统调用完成