3.修改"/home/shiyanlou/oslab/linux-0.11/include/liunx"目录下的"sys.h"文件,使0x80号中断能调用相应的系统函数: sys.h (~/oslab/oslab/linux-0.11/include/linux) - gedit File Edit View Tools Search Documents • sys.h 💥 EALERT THE SYS_SECUTIONAL extern int sys_ssetmask(); extern int sys_setreuid(); extern int sys setregid(); extern int sys_iam(); extern int sys_whoami(); fn_ptr sys_call_table[] = { sys_setup, sys_exit, sys_fork, sys_read, sys_write, sys_open, sys_close, sys_waitpid, sys_creat, sys_link, sys_unlink, sys_execve, sys_chdir, sys_time, sys_mknod, sys_chmod, sys_chown, sys_break, sys_stat, sys_lseek, sys_getpid, sys_mount, sys_umount, sys_setuid, sys_getuid, sys_stime, sys_ptrace, sys_alarm, sys_fstat, sys_pause, sys_utime, sys_stty, sys_gtty, sys_access, sys_nice, sys_ftime, sys_sync, sys_kill, sys_rename, sys_mkdir, h sys_rmdir, sys_dup, sys_pipe, sys_times, sys_prof, sys_brk, sys_setgid, sys_getgid, sys_signal, sys_geteuid, sys_getegid, sys_acct, sys_phys, sys_lock, sys_ioctl, sys_fcntl, sys_mpx, sys_setpgid, sys_ulimit, sys_uname, sys_umask, sys_chroot, sys_ustat, sys_dup2, sys_getppid, sys_getpgrp, sys_setsid, sys_sigaction, sys_sgetmask, sys_ssetmask, al sys_setreuid,sys_setregid_sys_iam,sys_whoami C/C++/ObjC Header ▼ Tab Width: 8 ▼ Ln 75, Col 1 INS shiyanlou@5ele030b6323:~/oslab/oslab/linux-0.11/kernel\$ cd ../include/li shiyanlou@5e1e030b6323:~/oslab/oslab/linux-0.11/include/linux\$ gedit sys .h 4."/home/shiyanlou/oslab/linux-0.11/kernel"目录, 创建"who.c"文件 who.c #define LIBRARY #include <unistd.h> #include <errno.h> #include <asm/segment.h> char usnm[64]={ 0}; sys_iam const char int result= 0; int cnt; while (get_fs_byte(name+result) != "\0" && result< 64) result++; if (result> 23) return - EINVAL; else for (cnt= 0;cnt<=result;cnt++){usnm[cnt]=get_fs_byte(name+cnt);}</pre> return result; sys_whoami char unsigned int int int result= 0; int cnt; while (usnm[result] != "\0" && result< 64) result++; if (result>size) return -1; else for (cnt= 0;cnt<=result;cnt++){put_fs_byte(usnm[cnt],(name+cnt));} return result;

打开"/home/shiyanlou/oslab/linux-0.11/kemel"目录,然后用"gedit"软件打开"Makefile",将文件第29行修改为:

6. 挂载虚拟机硬盘(在oslab目录下执行命令\$./mount-hdc),打开目录"/home/shiyanlou/oslab/hdc/usr/include",

who.s who.o:who.c ../include/unistd.h ../include/errno.h \ ../include/asm/segment.h

syscall2(int ,whoami, const char *,name, unsigned int ,size)

8. 在"/home/shiyanlou/oslab/hdc/usr/root"目录下编写程序"iam.c"和"whoami.c":

if (argc> 1) { if (iam(argv[1])< 0) return -1;}

-1;

if (whoami(str, 24)<0) return -1;

9. 从"/home/teacher"拷贝文件"testlab2.c"和"testlab.sh"至以下路径:

Bochs x86 emulator, http://bochs.sourceforge.net/

linux-0.00

linux0.tgz

shiyanlou@5ele030b6323:~/oslab/oslab/linux-0.11\$ cd ...

shoe

mtools.howto

shoelace.tar.Z whoami

whoami.c

whoami.c~

testlab2

_____ Bochs x86 Emulator 2.3.7 Build from CVS snapshot, on June 3, 2008 ______

] installing x module as the Bochs GUI

] using log file ./bochsout.txt

testlab2.c

testlabZ.sh

] reading configuration from ./bochs/bochsrc.bxrc

lou/os

it sys

../ cd

sniyaniou.com

\$Revision: 1.194 \$ \$Date: 2007/12/23 19:46:27 \$ Options: apmbios pcibios eltorito rombios32

3454 buffers = 3536896 bytes buffer space

hello.o

iam

umount hdc first

00000000000i[

00000000001[00000000001[

iam.c

iam.c

ataO master: Generic 1234 ATA-6 Hard-Disk (60 MBytes)

CTRL + 3rd button enables nouse R: HD:0-HINUM CAPS SCRL

shiyanlou@5ele030b6323:~/oslab/oslab\$./run

else printf ("%s\n", str);

2.修改"/home/shiyanlou/oslab/linux-0.11/kernel"目录下的"system_call.s"文件,扩大系统调用函数的数量:

<u>V</u>iew

Search

Edit

system_call.s 💥

offsets within sigaction

shiyanlou@5e1e030b6323:~/oslab/oslab/linux-0.11/include\$ cd ../kernel/

shiyanlou@5e1e030b6323:-/oslab/oslab/linux-0.11/kernel\$ gedit system_cal

system call.s-

shiyanlou@5ele030b6323:~/oslab/oslab/linux-0.11/kernel\$ gedit sys

aigirai

sigaction = 16

sa_handler = 0

sa_mask = 4

shiyanlou@5e1e030b6323:-/os .globl device_not_available, coprocessor_error

blocked = (33*16)

shiyanlou@5e1e0

make clean

shiyanlou@5e1e030b6323:~/os

shiyanlou@5e1e030b6323:-/o:

shiyanlou@5e1e030b6323:~/o:

shiyanlou@5ele030b6323:~/os

shiyanlou@5e1e030b6323:~/os

Firefishiyanlou@5ele030b6323:~/o: sa_flags = 8 sa_restorer = 12

ctype.h

dirent.h

errno.h

fcntl.h

float.h

shiyanlou@5ele030b6323:-/o: align 2

getopt.h

gnu-stabs.h

system call.o

system call.s

shiyanlou@5e1e030b6323:~/o: nr_system_calls = 74

make all

./run

247 history

Home

Term

242

243

244

246

ansidecl.h

a.out.h

assert.h

const.h

sys.c

sys.o

1.5

か 应用程序菜单

5. 修改"Make file"文件:

signal.o mktime.o who.o

添加头文件"usname.h":

#define __LIBRARY__ #include <unistd.h> #include <errno.h>

#include <asm/segment.h>

_syscall1(int ,iam, const char *,name)

"/home/shiyanlou/oslab/linux -0.11/include/unistd.h"

"/home/shiyanlou/oslab/linux -0.11/include/linux/sys.h"

"/home/shiyanlou/oslab/hdc/usr/include/linux";

<usname.h>

else return

return 0;

int main int char

"/home/shiyanlou/oslab/hdc/usr/include";

usname.h

7. 将文件:

将文件:

复制到以下路径:

复制到以下路径:

iam.c

#include

whoami.c

#include <usname.h> #include <stdio.h> int main void

char str[128];

"/home/shiyanlou/oslab/hdc/usr/root"

(1)键入以下三个命令:

gcc -o iam iam.c -Wall

(2)键入以下两个命令:

./whoami 之后键入:

(3) 键入以下两个命令:

chmod +x testlab2.sh

./iam aa

./testlab2

./testlab2.sh

实验结果截图:

Booting from Floppy...

free mem: 12582912 bytes

[/usr/root]# ./iam aa

Loading system ...

Partition table ok. 38985/62000 free blocks 19509/20666 free inodes

[/usr/root]# ls

0k.

README

hello

aa

hello.c

gcclib140

[/usr/root]#

♦ 应用程序菜单

gcc -o testlab2 testlab2.c

gcc -o whoami whoami.c -Wall

10. 取消虚拟机硬盘挂载,运行虚拟机:

在原第50行和第51行之间增加如下代码:

blocksize.h

ar.h

245 cd ..

system_call.s (~/oslab/oslab/linux-0.11/kernel) - gedit

Tab Width: 8 ▼

Ln 61, Col 1

INS

sniyaniou.com

Documents

Tools

MUST be 16 (=len of sigaction)

* Ok, I get parallel printer interrupts while using the floppy for some

* strange reason. Urgel. Now I just ignore them.

globl system_call,sys_fork,timer_interrupt,sys_execve

.globl hd_interrupt,floppy_interrupt,parallel_interrupt