Lab 2 Report

Christina Pavlopoulou
cpavl001@ucr.edu

Niloufar Hosseini Pour
nhoss003@ucr.edu

Andres Calderon
acald013@ucr.edu

February 22, 2016

1 Null pointer at command line.

```c
// Load program into memory.
sz = PGSIZE;
for(i=0, off=elf.phoff; i<elf.phnum; i++, off+=sizeof(ph)){
  if(readi(ip, (char*)&ph, off, sizeof(ph)) != sizeof(ph))
    goto bad;
  if(ph.type != ELF_PROG_LOAD)
    continue;
  if(ph.memsz < ph.filesz)
    goto bad;
  if((sz = allocuvm(pgdir, sz, ph.vaddr + ph.memsz)) == 0)
    goto bad;
  if(loaduvm(pgdir, (char*)ph.vaddr, ip, ph.off, ph.filesz) < 0)
    goto bad;
}
```

Listing 1: Changes in exec.c file.

```c
// Given a parent process's page table, create a copy
// of it for a child.
pde_t*
copyuvm(pde_t *pgdir, uint sz)
{
  pde_t *d;
  pte_t *pte;
  uint pa, i, flags;
  char *mem;

  if((d = setupkvm()) == 0)
    return 0;
  for(i = PGSIZE; i < sz; i += PGSIZE){
    if((pte = walkpgdir(pgdir, (void *) i, 0)) == 0)
      panic("copyuvm: pte should exist");
    if(!(*pte & PTE_P))
      panic("copyuvm: page not present");
    pa = PTE_ADDR(*pte);
    flags = PTE_FLAGS(*pte);
    if((mem = kalloc()) == 0)
      goto bad;
    memmove(mem, (char*)p2v(pa), PGSIZE);
    if(mappages(d, (void*)i, PGSIZE, v2p(mem), flags) < 0)
      goto bad;
  }
  return d;

bad:
  freevm(d);
  return 0;
}
```

Listing 2: Changes in vm.c file.

```makefile
_%: %.o $(ULIB)
  $(LD) $(LDFLAGS) -N -e main -Ttext 0x1000 -o $@ $^
  $(OBJDUMP) -S $@ > $*.asm
  $(OBJDUMP) -t $@ | sed '1,/SYMBOL TABLE/d; s/ .* / /; /^$$/d' > $*.sym

_forktest: forktest.o $(ULIB)
  # forktest has less library code linked in - needs to be small
  # in order to be able to max out the proc table.
  $(LD) $(LDFLAGS) -N -e main -Ttext 0x1000 -o _forktest forktest.o ulib.o usys.o
  $(OBJDUMP) -S _forktest > forktest.asm
```

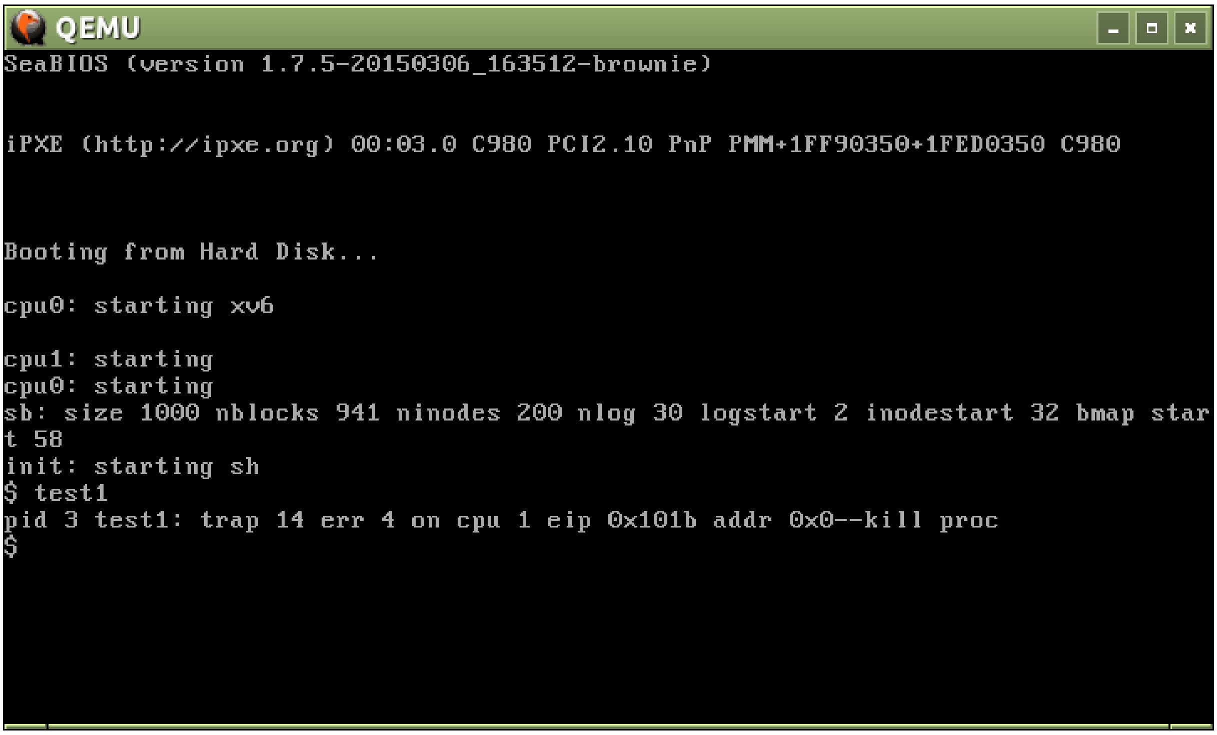
Listing 3: Changes in Makefile.

```c
#include "types.h"
#include "user.h"
#include "syscall.h"

int main(){
  int *p = 0;

  printf(1,"%d\n", *p);
  exit();
}
```

Listing 4: Test for null pointer catching at command line (test1.c file).



```
SeaBIOS (version 1.7.5-20150306_163512-brownie)


iPXE (http://ipxe.org) 00:03.0 C980 PCI2.10 PnP PMM+1FF90350+1FED0350 C980



Booting from Hard Disk...

cpu0: starting xv6

cpu1: starting
cpu0: starting
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2 inodestart 32 bmap star
t 58
init: starting sh
$ test1
pid 3 test1: trap 14 err 4 on cpu 1 eip 0x101b addr 0x0--kill proc
$
```

Figure 1: Output of test 1.

2 Null pointer at system call.

```c
// Fetch the nth word-sized system call argument as a pointer
// to a block of memory of size n bytes.  Check that the pointer
// lies within the process address space.
int
argptr(int n, char **pp, int size)
{
  int i;

  if(argint(n, &i) < 0)
    return -1;
  if((uint)i >= proc->sz || (uint)i+size > proc->sz)
    return -1;
  *pp = (char*)i;
  if(*pp == 0){
    cprintf("Null Pointer Exception!!!\n");
    return -1;
  }
  return 0;
}
```

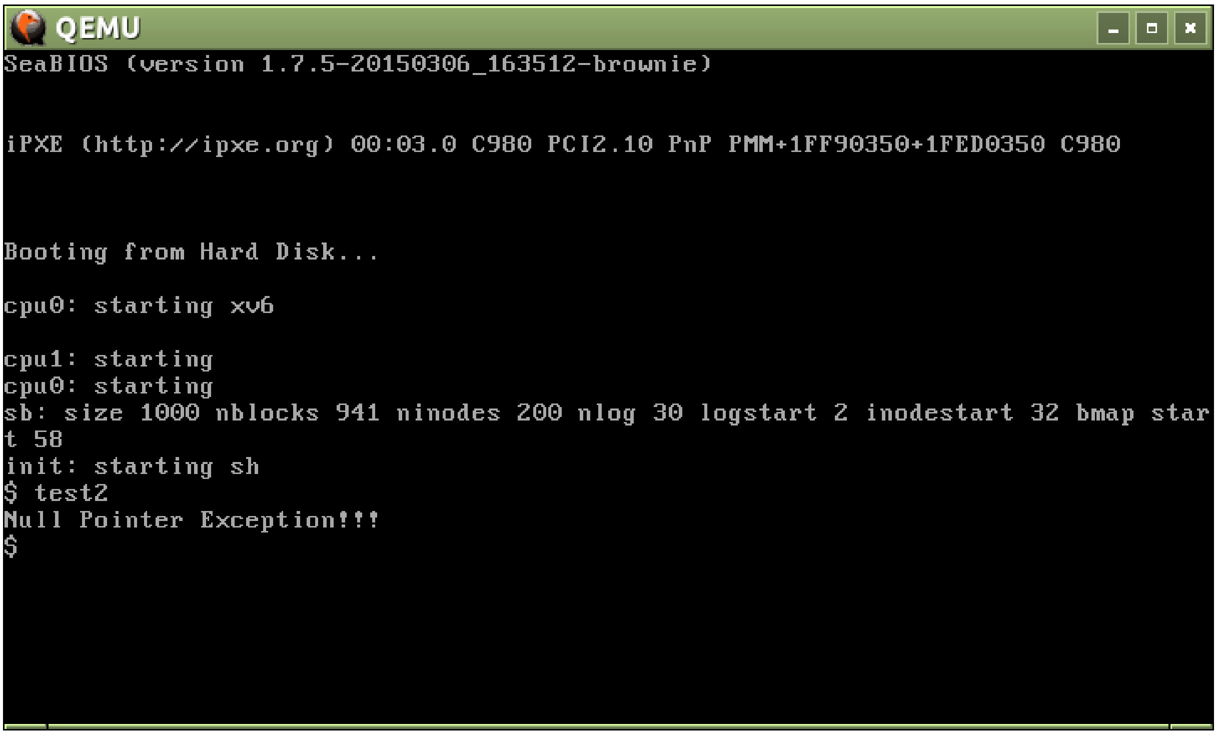
Listing 5: Changes in syscall.c file.

```c
#include "types.h"
#include "user.h"
#include "syscall.h"

int main(){
  int *p = 0;

  null(p);
  exit();
}
```

Listing 6: Test for null pointer catching at system call (test2.c file).



```
SeaBIOS (version 1.7.5-20150306_163512-brownie)


iPXE (http://ipxe.org) 00:03.0 C980 PCI2.10 PnP PMM+1FF90350+1FED0350 C980



Booting from Hard Disk...

cpu0: starting xv6

cpu1: starting
cpu0: starting
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2 inodestart 32 bmap star
t 58
init: starting sh
$ test2
Null Pointer Exception!!!
$
```

Figure 2: Output of test 2.