1.

#0 sys_read () at sysfile.c:71

#1 0x80104837 in syscall () at syscall.c:139

#2 0x801058b9 in trap (tf=0x8dffefb4) at trap.c:43

#3 0x8010561f in alltraps () at trapasm.S:20

#4 0x8dffefb4 in ?? ()

2.

کد های داخل sys_read را در یک حلقه ی نامتناهی قرار می دهیم((for(;;). یک برک پوینت روی sys_read میگذاریم و با دستورات next، step و finish شروع به دیباگ میکنیم. متوجه خواهیم شد که در :

```
if(argfd(0,
0, &f) < 0
||
argint(2,
&n) < 0 ||
argptr(1,
&p, n) < 0)
                  return -1;
                return fileread(f, p, n);
```

هیچگاه در صورت براورده نشدن شرط if خارج از حلقه نمیتوانیم برویم.

4.

s[tep]i

Execute a single instruction and then return to the command line interpreter.

n[ext]i

Like stepi, except that if the instruction is a subroutine call, the entire subrou-tine is executed before control returns to the interpreter.

5.

(gdb) bt

#0 sys_write () at sysfile.c:83

#1 0x80104837 in syscall () at syscall.c:139

#2 0x801058b9 in trap (tf=0x8dffffb4) at trap.c:43

#3 0x8010561f in alltraps () at trapasm.S:20

#4 0x8dffffb4 in ?? ()

6.

layout asm:

```
B+>|0x80104b50 <sys_write>        push   %ebp                          |
   |0x80104b51 <sys_write+1>      xor    %eax,%eax                     |
   |0x80104b53 <sys_write+3>      mov    %esp,%ebp                     |
   |0x80104b55 <sys_write+5>      sub    $0x18,%esp                    |
   |0x80104b58 <sys_write+8>      lea    -0x14(%ebp),%edx              |
   |0x80104b5b <sys_write+11>     call   0x80104a10 <argfd>            |
   |0x80104b60 <sys_write+16>     test   %eax,%eax                     |
   |0x80104b62 <sys_write+18>     js     0x80104bb0 <sys_write+96>     |
   |0x80104b64 <sys_write+20>     lea    -0x10(%ebp),%eax              |
   |0x80104b67 <sys_write+23>     sub    $0x8,%esp                     |
   |0x80104b6a <sys_write+26>     push   %eax                          |
   |0x80104b6b <sys_write+27>     push   $0x2                          |
   |0x80104b6d <sys_write+29>     call   0x80104720 <argint>
```

layout src

```
┌──────────────────────────────────────────────────────────────────────┐
B+>|83    {                                                           |
   |84      struct file *f;                                           |
   |85      int n;                                                    |
   |86      char *p;                                                  |
   |87                                                               |
   |88      if(argfd(0, 0, &f) < 0 || argint(2, &n) < 0 || argptr(1, &p, n) <|
```

```
89        return -1;
90      return filewrite(f, p, n);
91    }
92
93    int
94    sys_close(void)
95    {
```