

OSTEP Chapter 39

ECE 3600, Fall 2022

Table of Contents

- [1. Files and Directories](#)
- [2. strace](#)
- [3. Shared Files](#)
- [4. open, read, offset examples](#)
- [5. Inode Metadata](#)
- [6. Reading Directories](#)
- [7. Permission Bits](#)
- [8. Mounted File Systems](#)

1. Files and Directories

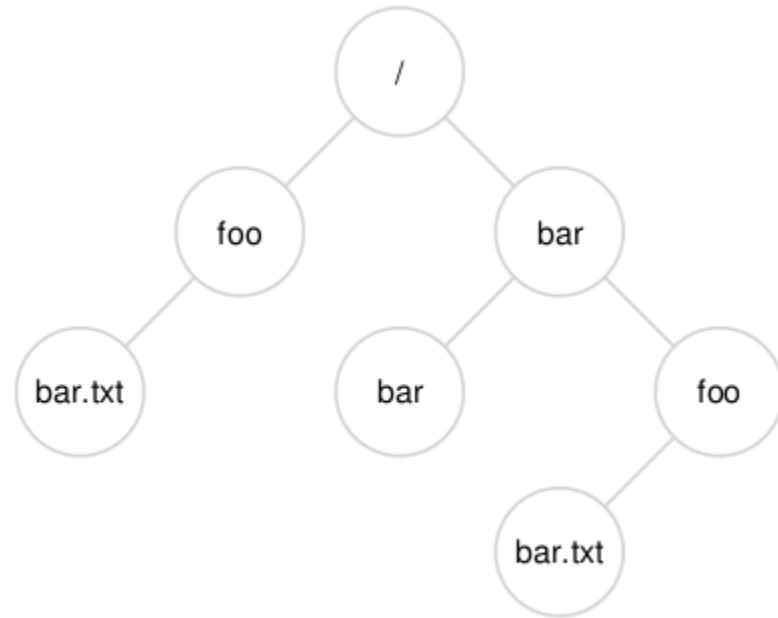


Figure 39.1: An Example Directory Tree

API:

[open\(\)](#), [read\(\)](#), [write\(\)](#), [dup\(\)](#), [lseek\(\)](#), [fsync\(\)](#), [close\(\)](#)

[stat\(\)](#), [link\(\)](#), [symlink\(\)](#), [unlink\(\)](#), [rename\(\)](#), [chmod\(\)](#)

[mkdir\(\)](#), [rmdir\(\)](#), [opendir\(\)](#), [readdir\(\)](#), [closedir\(\)](#)

2. strace

[strace](#) - trace system calls and signals

```
$ echo hi > j
$ strace -e trace=open,openat,read,close cat j
openat(AT_FDCWD, "/etc/ld.so.cache", 0_RDONLY|0_CLOEXEC) = 3
close(3) = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", 0_RDONLY|0_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\260\34\2\0\0\0\0\0"... , 832) = 832
close(3) = 0
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", 0_RDONLY|0_CLOEXEC) = 3
close(3) = 0
openat(AT_FDCWD, "j", 0_RDONLY) = 3
read(3, "hi\n", 131072) = 3
hi
read(3, "", 131072) = 0
close(3) = 0
close(1) = 0
close(2) = 0
+++ exited with 0 +++
$
```

3. Shared Files

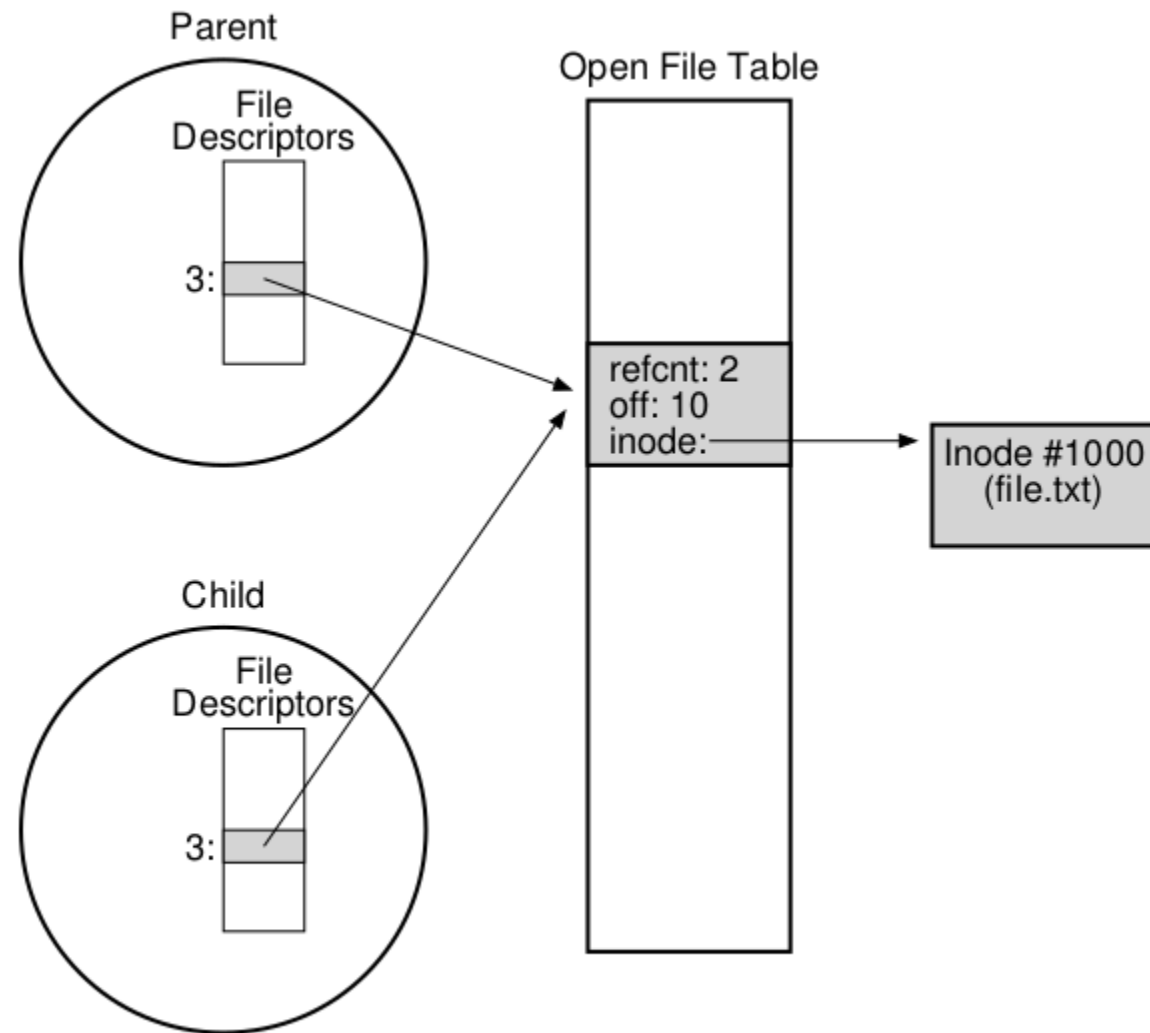


Figure 39.3: Processes Sharing An Open File Table Entry

4. open, read, offset examples

Assume *file* contains 300 bytes:

System Calls	Return Code	Current Offset
<code>fd = open("file", O_RDONLY);</code>	3	0
<code>read(fd, buffer, 100);</code>	100	100
<code>read(fd, buffer, 100);</code>	100	200
<code>read(fd, buffer, 100);</code>	100	300
<code>read(fd, buffer, 100);</code>	0	300
<code>close(fd);</code>	0	-

open more than once:

System Calls	Return Code	OFT[10] Current Offset	OFT[11] Current Offset
<code>fd1 = open("file", O_RDONLY);</code>	3	0	-
<code>fd2 = open("file", O_RDONLY);</code>	4	0	0
<code>read(fd1, buffer1, 100);</code>	100	100	0
<code>read(fd2, buffer2, 100);</code>	100	100	100
<code>close(fd1);</code>	0	-	100
<code>close(fd2);</code>	0	-	-

open and reposition:

System Calls	Return Code	Current Offset
<code>fd = open("file", O_RDONLY);</code>	3	0
<code>lseek(fd, 200, SEEK_SET);</code>	200	200
<code>read(fd, buffer, 50);</code>	50	250
<code>close(fd);</code>	0	-

5. Inode Metadata

inode = **index node** (information node)

inode number is index into file system inode table

```
struct stat {
    dev_t      st_dev;      // ID of device containing file
    ino_t      st_ino;      // inode number
    mode_t     st_mode;     // protection
    nlink_t    st_nlink;    // number of hard links
    uid_t      st_uid;      // user ID of owner
    gid_t      st_gid;      // group ID of owner
    dev_t      st_rdev;     // device ID (if special file)
    off_t      st_size;     // total size, in bytes
    blksize_t  st_blksize;  // blocksize for filesystem I/O
    blkcnt_t   st_blocks;   // number of blocks allocated
    time_t     st_atime;    // time of last access
    time_t     st_mtime;    // time of last modification
    time_t     st_ctime;    // time of last status change
};
```

Figure 39.5: The **stat** structure.

[stat\(\)](#) system call, [stat](#) command

```
$ stat j
  File: j
  Size: 3          Blocks: 8          IO Block: 4096   regular file
Device: 902h/2306d  Inode: 9981236   Links: 1
Access: (0644/-rw-r--r--)  Uid: ( 1000/   perry)   Gid: ( 1000/   perry)
Access: 2020-06-17 11:06:52.799937997 -0400
Modify: 2020-06-17 11:06:42.512146065 -0400
Change: 2020-06-17 11:06:42.512146065 -0400
  Birth: -
$
```

6. Reading Directories

The only standard fields in a directory entry structure are `d_ino` and `d_name`:

```
struct dirent
{
    ino_t  d_ino;      // Inode number
    char   d_name[256]; // Null-terminated filename
    //...
};
```

Example program from chapter 39 page 16:

```
#include <stdio.h>
#include <dirent.h>
#include <assert.h>

int main(int argc, char *argv[]) {
    DIR *dp = opendir(".");
    assert(dp != NULL);
    struct dirent *d;
    while ((d = readdir(dp)) != NULL) {
        printf("%lu %s\n", (unsigned long) d->d_ino, d->d_name);
    }
    closedir(dp);
    return 0;
}
```

7. Permission Bits

[chmod\(\)](#) system call, [chmod](#) command

```
$ ls -ld . notes.html /bin/ls
drwxr-xr-x 2 perry perry 4096 Jun 17 14:20 .
-rwxr-xr-x 1 root  root 133792 Jan 18 2018 /bin/ls
-rw-r--r-- 1 perry perry 1118 Jun 17 14:18 notes.html
$
```

user, group, other; read, write, execute:

```
-rwxr-xr-x
uuugggooo
111101101 = 0755
```

```
-rw-r--r--
uuugggooo
110100100 = 0644
```


8. Mounted File Systems

```
$ df | egrep '/dev/md|File'
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/md0         49687632 13045168 34088764 28% /
/dev/md3         960249880 514337568 397064492 57% /a
/dev/md2         402626372 13579452 368524324 4% /home
$ mount | grep /dev/md
/dev/md0 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)
/dev/md3 on /a type ext4 (rw,relatime,data=ordered)
/dev/md2 on /home type ext4 (rw,relatime,data=ordered)
$ grep md2 /proc/mdstat
md2 : active raid1 sda3[0] sdb3[1]
$ sudo fdisk -l | grep '^Disk /dev/sd'
Disk /dev/sdb: 447.1 GiB, 480103981056 bytes, 937703088 sectors
Disk /dev/sda: 447.1 GiB, 480103981056 bytes, 937703088 sectors
Disk /dev/sdc: 931.5 GiB, 1000204886016 bytes, 1953525168 sectors
Disk /dev/sdd: 931.5 GiB, 1000204886016 bytes, 1953525168 sectors
$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
udev            8060648          0    8060648  0% /dev
tmpfs           1618832         2088    1616744  1% /run
/dev/md0        49687632 13045168 34088764 28% /
tmpfs           8094148         40092    8054056  1% /dev/shm
tmpfs           5120            4         5116  1% /run/lock
tmpfs           8094148          0    8094148  0% /sys/fs/cgroup
/dev/loop0      99456           99456          0 100% /snap/core/9289
/dev/loop1      96256           96256          0 100% /snap/core/9066
/dev/loop2      18304           18304          0 100% /snap/pdftk/9
/dev/md3        960249880 514337568 397064492 57% /a
/dev/md2        402626372 13579452 368524324 4% /home
tmpfs           1618828          16    1618812  1% /run/user/125
tmpfs           1618828          96    1618732  1% /run/user/1000
tmpfs           1618828          0    1618828  0% /run/user/0
$
```