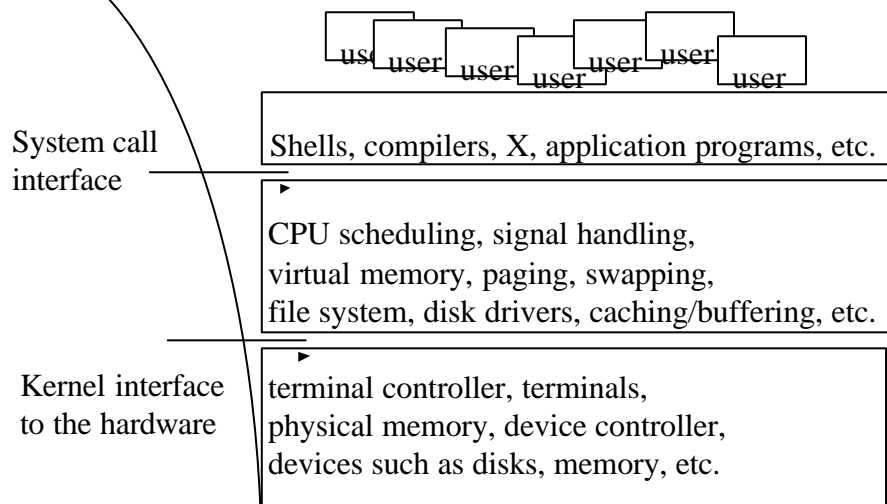


UNIX

- ✍ Introduction
- ✍ Programmer Interface
- ✍ User Interface
- ✍ Process Management
- ✍ Memory Management
- ✍ File System
- ✍ I/O System
- ✍ Interprocess Communication



UNIX Architecture



UNIX Architecture

☞ A Layer Architecture

☞ System Calls

☞ Programmer Interface to UNIX

☞ Trap 40 – VAX BSD4.2

☞ R0 – error code

☞ Categories

☞ File Manipulation

☞ Devices are special files under “/dev”!

☞ Process Control

☞ Information Manipulation

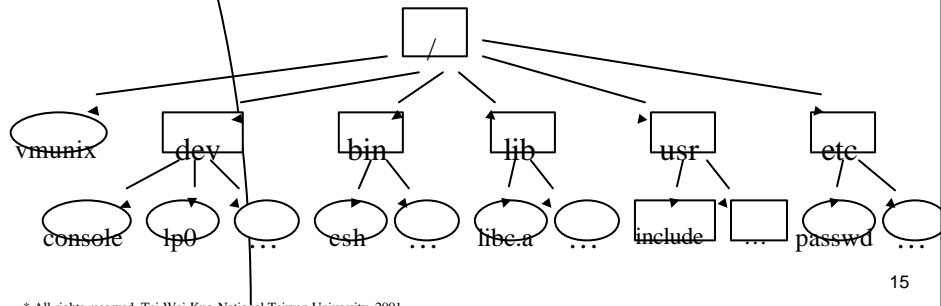
System Calls - File Manipulation

☞ File

☞ A sequence of bytes

☞ Directory

☞ A file that includes info on how to find other files.



System Calls - File Manipulation

✎ Path name

✎ Absolute path name

✎ Start at the root / of the file system

✎ /user/john/fileA

✎ Relative path name

✎ Start at the "current directory" which is an attribute of the process accessing the path name.

✎ ./dirA/fileB

✎ Links

✎ Symbolic Link – 4.3BSD

✎ A file containing the path name of another file can across file-system boundaries.

✎ Hard Link

16

* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

System Calls - File Manipulation

✎ Directories

✎ /vmunix - binary root image of UNIX

✎ /dev - device special files, e.g., /dev/console

✎ /bin - binaries of UNIX system programs

✎ /usr/ucb - written by Berkley instead of AT&T

✎ /usr/local/bin - written at the local site

✎ /lib - library files, e.g., those for C

✎ /user - directories for users, e.g., /user/john

✎ /etc - administrative files and programs,
e.g., passwd

✎ /tmp - temporary files

17

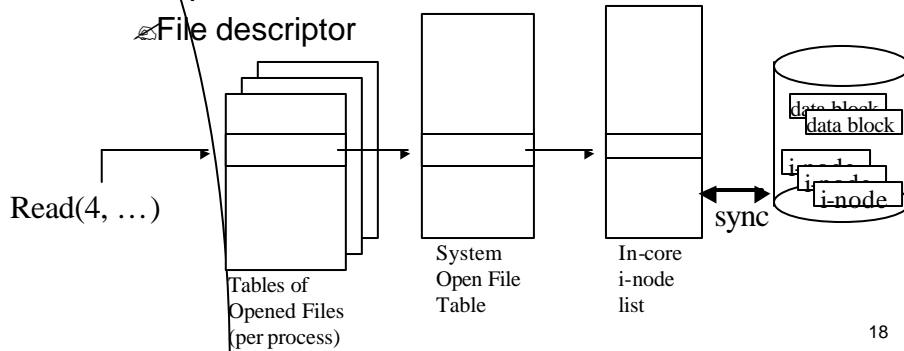
* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

System Calls - File Manipulation

Operations

open, close, read, write, trunc, lseek, dup, rename, chmod, chown, fcntl, ioctl, mkdir, cd, opendir, readdir, closedir, etc.

File descriptor



* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

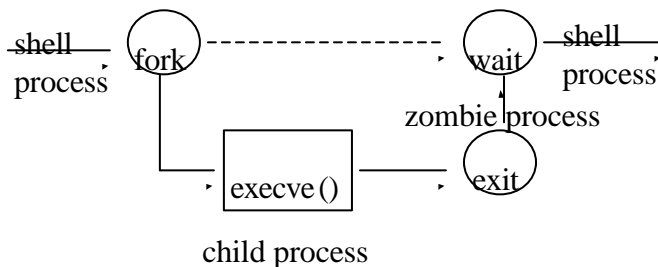
18

System Calls – Process Control

A process

A program in execution

```
if (pid = fork()) {
...
wait();
} else {
execve("a.out");
}
```



* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

19

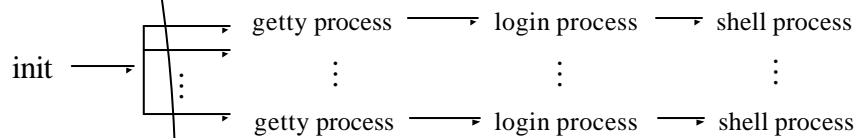
System Calls – Process Control

✍ User ID

- ✍ Effective UID, real UID
- ✍ With setuid bit in the inode of a file!

✍ Operations

- ✍ setuid, getuid, geteuid, getgid, getegid, getgroups, etc.



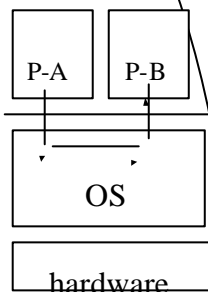
* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

20

System Calls - Signals

✍ Signal

- ✍ A facility for handling exceptional conditions similar to software interrupts
- ✍ Generated by errors, keyboard, events, etc, e.g., kill -9 pid
- ✍ ~ 20 signals, such as SIGSEGV, SIGINT (^C) – Multiple instances could be lost!
- ✍ POSIX – reliable signals
- ✍ User-level signal handlers, masking, etc.
- ✍ Except SIGKILL, SIGINT, SIGQUIT, which is for the killing of run-away processes.



* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

21

System Calls – Process Control

- ✍ Job (or process group)
 - ✍ A group of processes frequently cooperate to accomplish a common goal.
 - ✍ setpgrp()
 - ✍ One job can use a terminal I/O at any time – foreground job!
- ✍ SIGINT, SIGSTOP, or SIGTTOU vs SIGCONT
 - ✍ freeze vs resume
- ✍ Useful for X – each window as a terminal

22

* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.


System Calls – Information Manipulation & Library Routine

- ✍ Information Manipulation
 - ✍ gettimeofday/gettimeofday (us), gethostname, getpid, getgid, etc.
- ✍ Library routines
 - ✍ With header files, e.g., stdio.h, math.h, additional program support is provided.
 - ✍ Over 150 system calls for 4.3BSD, over 300 library functions for C
 - ✍ Eventually result in system calls!
 - ✍ Getchar() -> read()

23

* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

UNIX

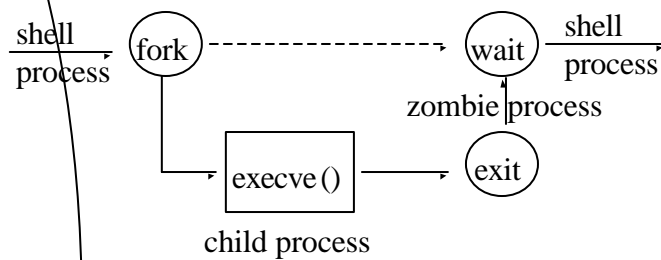
- ✍ Introduction
- ✍ Programmer Interface
- ✍  User Interface
- ✍ Process Management
- ✍ Memory Management
- ✍ File System
- ✍ I/O System
- ✍ Interprocess Communication

User Interface

- ✍ Interface of system programs
 - ✍ File/directory-oriented
 - ✍ Mkdir, rmdir, pwd, cd, ls, cp, mv, rm, cat, more, head, tail, diff, grep, etc.
 - ✍ Editors
 - ✍ Emacs, vi, ed, etc.
 - ✍ Compilers
 - ✍ C, Pascal, FORTRAN, etc.
 - ✍ Others
 - ✍ Mail, X, etc.

User Interface – Shells and Commands

- Shell – commander interpreter
- /bin/sh (Bourne shell by Steve Bourne), /bin/csh (C shell by Bill Joy), etc.

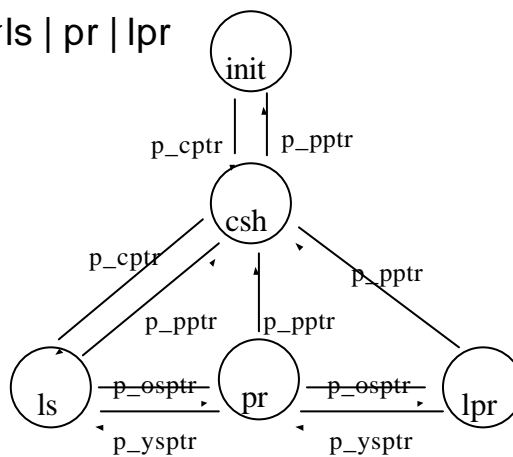


* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

26

User Interface – Shells

ls | pr | lpr



Siblings!

* All rights reserved, Tei-Wei Kuo National Taiwan University, 2001.

27

User Interface – Shells and Commands

Commands

- Shell-built-in commands, e.g., cd, etc.

- System programs, e.g., ls, ps, etc.

- Search Path (e.g. those in .cshrc or .login)

- set path=(/usr/sbin /usr/ucb /usr/bin /usr/local/bin /etc /usr/etc .)

- setenv LD_LIBRARY_PATH /usr/local/lib:/usr/lib:/usr/openwin/lib

Foreground vs Background Cmds

- ps& vs fg

User Interface – Standard I/O

Standard file descriptors

- 0, 1, 2 for stdin, stdout, stderr

I/O redirection

- pipe: carry data from one process to another!

- ls | pr > filea

- pr < filea

- make program >& err

