

PROCESS CONTROL

ISA 563: Fundamentals of Systems Programming

Internal Vs. External



- Your C code controls the internal behavior of your program: which variables to process and in what order
- As a unit of execution, how do you control your program's interaction with the OS?
 - Process control: tonight's topic
 - The Process Lifecycle

Overview: Processes

- A process is a program in execution
 - An OS abstraction
 - Important OS data structures represent and keep track of a process's state
 - PCB: Process Control Block (in Linux, an instance of type `task_struct`)
 - Process ID
 - Various process lists (running, sleeping/waiting)
 - Process Address Space
 - Process Sections
 - Re-read APUE Chapter 7 for reference

Process Metadata



- The Process ID
 - Unique but reused
 - The `getpid(2)` system call
 - Parent process ID: `getppid(2)`
 - How do you get PID of your children?
 - No built-in support
 - you need to keep track
- Real & effective user IDs, group IDs

The Process Lifecycle

How do processes begin and end?

What do they inherit?

APUE: Figure 7.2

User-level Process Control Utilities



- ps
- top
- jobs
- pstree
- bg, fg (shell built-in)
- “program&” (shell command to start in background:
i.e., detached from the terminal)
- kill, killall

Job Control Signals



- Ctrl-C (interrupt: SIGINT)
- Ctrl-D (end of input:)
- Ctrl-Z (suspend: SIGTSTP)

Interfacing with the Kernel



- Access (get/set) kernel state
 - sysctl(8): command line interface
 - sysctl(3): C library interface
- ptrace(2)
- strace

C Library Process Control



- `signal(3)`, `signal(7)`
- `abort(3)`
- `exit(3)`, `_Exit(3)`
- `atexit(3)`: APUe: Figure 7.3

OS Process Control



- `fork(2)`, `vfork(2)`, `execve(2)`, `wait(2)`,
- `kill(2)`
- `_exit(2)`
- `ptrace(2)`

The Environment

- NULL-terminated set of name=value pairs
- `envp*[]`
 - `int main(int argc, char* argv[], char* envp[]);`
 - Deprecated

```
extern char **environ;
```

Keeping Track of the Kids

Create a program that randomly forks off a child. Keep track of all children in dynamic storage.