Address Spaces

CSC 213 – September 5, 2022
Agenda for Today

1. Assignment Q&A
2. New Assignment
3. Process API (from Wednesday)
4. Limited Direct Execution (from Wednesday)
5. Address Spaces
6. Wrap Up
Q&A
Assignment Q&A

Do we need to comment our code?

Yes, please! You should see our discussion about good code from the first week for details on what makes for good commenting practices.

Hint: newline is not the same as end of file
New Assignment: Sorted List
Processes
Processes: Basics

What is a process?
A running program. Multiple processes can run at the same moment. There is a parent-child relationship between processes. The process holds the machine instructions that run the program, its current state, and an address space where the process runs.

Why does the OS provide this abstraction?
A process is a self-contained representation of a running program that we can pause and resume for time sharing/multitasking. Running in processes isolates them from each other. Good for security/privacy and reliability.
What does this function do?

Fork produces a child process that is a copy of the calling process. The process that called fork() is the parent of the new child process.

In the child process, fork() returns 0. In the parent, fork() returns the ID of the child.
What does this function do?
Calling wait in the parent process will wait for any child process to exit. The wait function returns the id of the child that exited.
The caller of wait() should pass in a pointer to an integer where values will be written by wait. Those values tell us the exit code from the child, how the process exited, and a few other details.
int execv(const char* pathname,
char* const argv[])  

What does this function do?
Stops running the program in the current process, and replaces it with a new program.
The argv array holds the string arguments accessible in the new program's main function. Index zero holds the name of program itself. The final element in the array is NULL, which marks the end.
Demo: fork, wait, and exec
Limited Direct Execution
Limited Direct Execution: Basics

What does the book mean by "limited direct execution?"

The program executes directly on the hardware, but the program cannot do some lower-level operations. The only way to do those is to ask the operating system to do them on behalf of the program.
Limited Direct Execution: Restricted Operations

What are some operations that are restricted under limited direct execution? Why are they restricted?

File Operations: the OS checks permissions for all file accesses. The OS also abstracts away the details of file operations.

Network Operations: the OS can hide the complexity of network hardware. Security concerns make it important to check network operations.

Managing Address Spaces: if programs could grab whatever memory they want, we could corrupt data or leak it.

Managing Processes: processes are isolated from each other, so the OS has to be involved in creating/interacting with them.
Limited Direct Execution: Restricted Operations

How can a process perform a restricted operation?
   by issuing a system call

How does this mechanism work?
   The program is running in user mode until it wants to issue a system call.
   It uses a trap to jump to kernel mode.
   The trap is received in kernel mode, and the OS kernel will check and perform
   the operation (assuming it's allowed).
   After the system call finishes, the OS returns to user mode.
Limited Direct Execution: Alternatives

Consider an OS that uses *Unlimited* Direct Execution. What would that mean? There could be an OS that does anything a program asks, with no checking. There could even be no OS at all, so programs can do anything they want.

**What about an OS that uses Limited *Indirect* Execution?**

There would still be restricted operations (just like limited direct execution). In this case, the program wouldn't run directly on the hardware. Instead, the code is interpreted or run on a virtual machine.
Address Spaces
Address Spaces: Basics

What are the main pieces of a running program's address space?

What is the difference between a virtual and physical address?
Address Spaces: Motivation

Why do we run processes in their own address spaces?
Whiteboard: A Typical Address Space
Wrap Up
Reminders

Reading
Read through Wednesday's lab before class

Assignment
The ngram assignment is due tonight. The sorted-list assignment is available now, and is due next Monday.

Need help?
I will post full office hours later today.
I am available today from 1:30–3:30pm. No appointment necessary, but you can reserve a time with calendly on the course website.