

Operating Systems: Overview

Shankar

January 24, 2022

What is an OS

- Software: runs directly on the hardware, always running
- Provides a more convenient virtual machine
 - processes and threads
 - process: executing instance of a program
 - protection for each process
 - synchronization and communication
 - virtual structured address space
 - filesystem, high-level IO
 - users
- Shares (virtualizes) hardware among processes and OS

Processes + Threads

- Process: executing instance of a program
 - Life: start, execute, terminate (perhaps)
 - Address space: text (code), data, stack segments
 - IO resources: open files, sockets, ...
 - Threads: each executes code; has its own stack
- Traditional programs: process has only one (main) thread
 - address space: text, data, stack
- Multi-threaded programs: one or more threads per process
 - address space: text, data, stack₁, stack₂, ...
- OS makes all threads execute concurrently
 - gives each process/thread a share of the hardware resources
 - sharing done in time and/or space (depends on resource)

Virtual Address Space

- Address space of a process
- Structured into segments/pages
 - attributes: size, allowed access, ...
 - checked during execution
- OS maps each virtual address to
 - address in physical memory (accessible to processor)
 - location in disk (processor access → exception)
- Mapping: segment/page tables, associative maps, ...

Filesystem

- Non-volatile structure of directories and files
- Tree/acyclic structure
- Each node is a directory or a file
 - file: holds data; variable size
 - directory: pointers to directories and files
 - attributes: owner, access rights, creation time, ...
- Processes can create/delete/read/modify/execute nodes
- Executable file: code + data segments, loading/linking info
- OS implements filesystem on block devices (disks, ...)
 - each node is mapped to one or more blocks
 - pointer structure to locate blocks of any node
 - use free blocks to expand nodes

System Calls

- sw-syscall n : like a function call except
 - function (“syscall handler”) is in kernel
 - n is not address but an index to a kernel table of addresses
 - cpu switches from user mode to kernel mode
- Classes of system calls
 - Process management
 - create/terminate a process/thread (including self)
 - Filesystem and IO
 - create, delete, open, read, write, close, modify attributes
 - Information
 - time, process information, hardware, IO devices, ...
 - Communication
 - connect, send, receive, terminate