Operating Systems | Monday- October 15, 2012

CSC 227 | Fall 2012/2013

Homework 2: Sample Solution | Points: 45

Due: Wednesday - November 7, 2012 at 2:00 pm

1.

Program execution:

The operating system loads the contents (or sections) of a file into memory and begins its execution. A user-level program could not be trusted to properly allocate CPU time, and memory space.

I/O operations:

I/O devices such as disks, tapes, printers must be communicated with at a very low level. The user need only specify the device and the operation to perform on it, while the system converts that request into device- or controller-specific commands. If user-level programs are allowed, they could want acquire them even when they haven't needed them. Programming them would require detailed knowledge of these devices.

Error detection:

Error detection occurs at both the hardware and software levels.

At the hardware level, all data transfers must be inspected to ensure that data have not been corrupted in transit after the data have been written to the media.

At the software level, media must be checked for data consistency; for instance, whether the number of allocated and unallocated blocks of storage matches the total number on the device.

There are numerous types of errors; hence there must be a global program (the operating system) to handle all types of errors. By having errors processed by the operating system, processes need not contain code to catch and correct all the possible errors on a system.

2.

Types	Examples
Process control	create/terminate/end/abort process
	wait for time, wait for event, etc
File management	create/delete a file or directory
	open/close a file, get/set file attributes
Device management	request/release device
	get/set device attributes
Information maintenance	get/set time or date
	get/set system data
	get/set process, file, or device attributes

Communications	create/delete communication connection send/receive messages, attach/detach remote devices

3.

a), c), and d) should be executed only in kernel mode.

a. This operation must be under the kernel's strict control; otherwise, a user program can execute a privileged instruction without kernel control.

b. Reading the time-of-day clock register without changing, doesn't interfere with the operating system's control of the machine.

c. Changing the time-of-day clock interferes with the operating system's control of the clock. This operation requires a kernel privilege; otherwise, the user can change the clock maliciously.

d. Because a user process can read or write the memory allocated to it freely, in order to protect the operating system from user process, the memory not allocated to a certain process is not allowed to be visited by this process. Therefore, the memory allocation or reallocation can only be handled by kernel

4.

Mechanism and policy must be separate to ensure that systems are easy to modify. No two system installations are the same, so each installation may want to tune the operating system to suit its needs. With mechanism and policy separate, the policy may be changed at will while the mechanism stays unchanged. This arrangement provides a more flexible system.

5.

Benefits typically include the following (i) adding a new service does not require modifying the kernel, (ii) it is more secure as more operations are done in user mode than in kernel mode, (iii) a simpler kernel design and functionality typically results in a more reliable operating system, and (iv) easier to port the operating system to new architecture.

6.

Performance decreases due to increased system function overhead because communication is provided by message passing.

7.

The modular kernel approach requires subsystems to interact with each other through carefully constructed interfaces that are typically narrow (in terms of the functionality that is exposed to external modules). The layered kernel approach is similar in that respect. However, the layered kernel imposes a strict ordering of subsystems such that subsystems at the lower layers are not allowed to invoke operations corresponding to the upper-layer subsystems. There are no such restrictions in the modular-kernel approach, wherein modules are free to invoke each other without any constraints.

8.

9.

Answer: The virtual memory subsystem and the storage subsystem are typically tightly-coupled and requires careful design in a layered system due to the following interactions. Many systems allow files to be mapped into the virtual memory space of an executing process. On the other hand, the virtual memory subsystem typically uses the storage system to provide the backing store for pages that do not currently reside in memory. Also, updates to the file system are sometimes buffered in physical memory before it is flushed to disk, thereby requiring careful coordination of the usage of memory between the virtual memory subsystem and the file system.

10.

Answer

- SYSGEN program obtains information concerning the specific configuration of the hardware system
- Bootstrap program – code stored in ROM that is able to locate the kernel, load it into memory, and start its execution