

Branch: master ▼

Find file

Copy path

Operating-Sysem / Feb26.md



Lapland-Stark Update Feb26.md

be27bde 11 seconds ago

1 contributor

Raw Blame History



52 lines (35 sloc) | 2.54 KB

# 操作系统作业1

PB18000227 艾语晨

## 1. 从系统的角度理解，操作系统主要负责哪两大功能

answer

- OS is a control program
- OS is a resource allocator
- 即：控制程序和资源分配者

## 2. 什么是系统调用？阐述系统调用与API的区别和逻辑关系

answer

- System calls are the programming interface between processes and the OS kernel
  - 系统调用是在进程与操作系统内核之间的接口
- Mostly a system call is accessed by programs via a high-level API rather than direct system call use
  - 大多数情况下系统调用是通过一个高层次的API实现，而不是直接做系统调用

### 3. 阐述Dual Mode的工作机制，以及采用Dual Mode的原因

answer

#### 工作机制

- 在进程执行时，分为 User mode 和 Kernel mode，由一个 mode bit 区分。
- Some instructions designed as privileged, only executable in kernel mode
- System call changes mode to kernal, return from call that resets it to user
- 总体来说，在执行一些必须在 kernel 里面执行的命令时，进入 kernel mode，执行完毕后返回 user mode

#### 原因

Dual-mode operation allows OS to protect itself and other system components .

Dual mode 可以更好的保护操作系统，因为在执行一些有危险错误（如非法访问并尝试修改越界数据，修改操作系统只读文件等）或恶意程序时，由于需要进入 kernel mode 而需要额外的权限，故可以避免一部分错误的发生

### 4. 分析Monolithic结构，层次化结构，模块化结构和微内核结构的优劣

answer

	Monolithic	Layered	MicroKernel
merits	更好的性能	易于建构和 debug；将 DS（数据结构？）、Ops（操作）、硬件对于上层结构隐藏起来	易于扩展：新的 services 加到用户空间，不需要修改内核；易于将操作系统移植到新的结构；更可靠安全（在 kernel 里面跑的代码更少）
weaknesses	难于实现和维护	不好定义不同层次; 效率问题	性能取决于用户空间到内核空间的传输效率

### 5. 举例说明采用机制与策略分离的设计原因

answer

- Timer mechanism 计时器 (for CPU protection) Policy decision: How long the timer is to be set? Mechanism 和 Policy 分开的原因是在具体使用时，或者不同版本，要根据使用环境、要求、不同需求的优先级的不同来改变设计，这时只需改变 Policy策略，就可以，而不需要修改 mechanism