Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心


# 操作系统结构与内核安全基础

博士研究生　刘望桐

2017年08月27日

# 内容提要

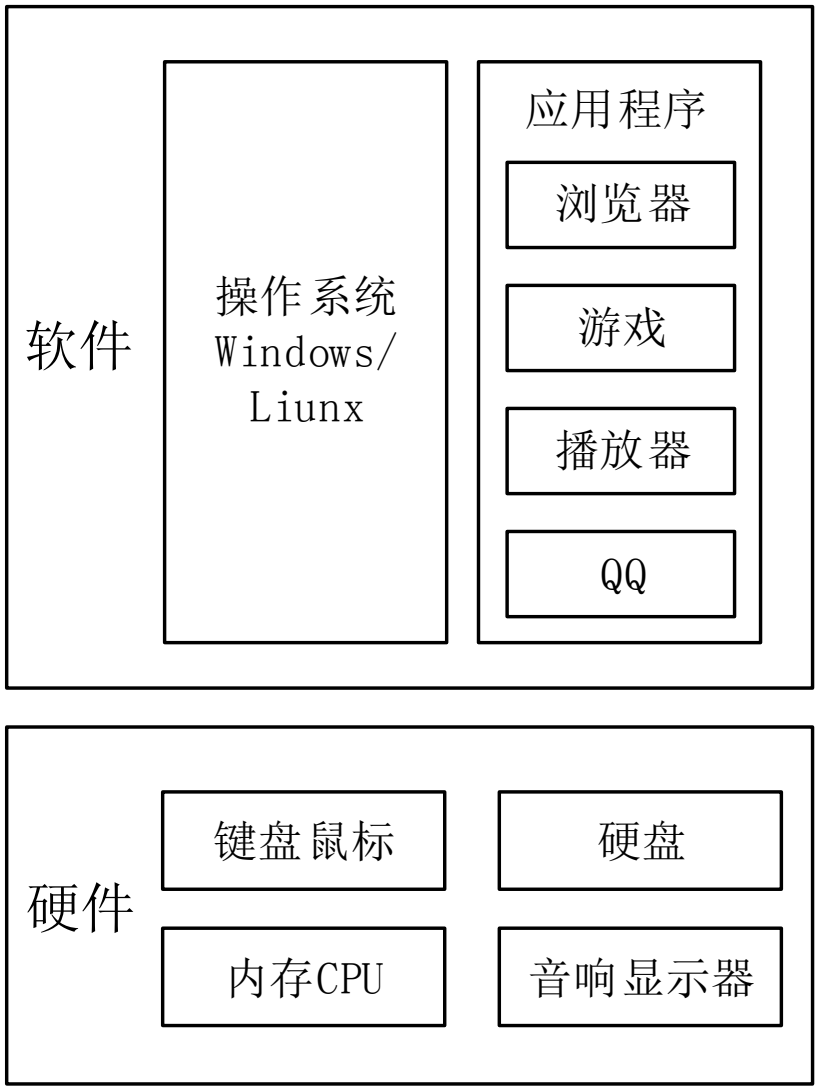
- 基本概念
  - 计算机层次结构
- 操作系统与操作系统内核
  - 操作系统与操作系统内核的关系
  - 操作系统内核功能
- 用户层与内核层关系
- 内核安全
- 其他

# 基本概念

# 基本概念

- 我们对计算机结构的认识可能是这样的

- 那么，其中操作系统的作用到底是什么？

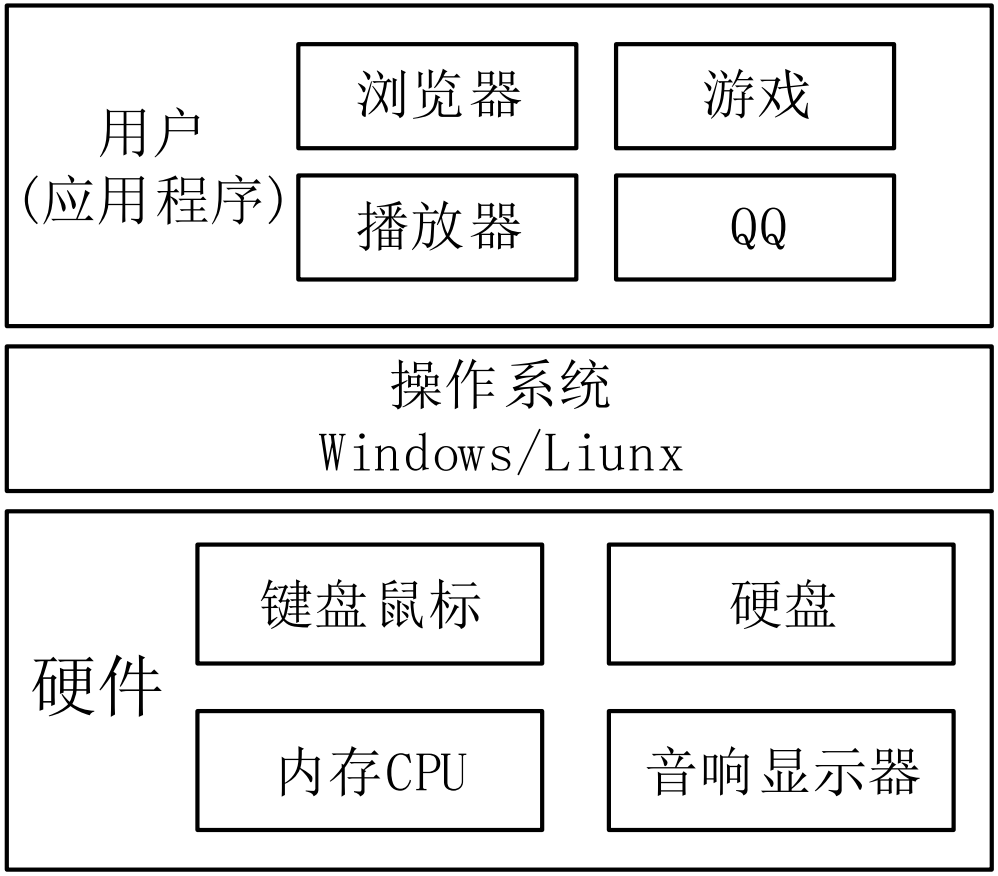


# 基本概念

- 操作系统产生的核心目的，是为了解决3个问题
  - 硬件差异
    - 不同的物理硬件在交互过程、数据结构等很多方面都有差异，需要一个统一的平台来进行规范，从而减轻应用程序的开发难度
    - 操作系统所提供的统一所有底层硬件的概念叫“硬件抽象”
  - 资源管理
    - 商用计算机需要能够“同时”运行多个程序，需要有一个管理者来分配和管理硬件资源，保证每个程序都正常运行
      - 参考“Windows多线程基础-刘望桐-2017-02-12”
  - 任务调度
    - “同时”运行多个程序时，对硬件的时分复用、不同特殊事件的中断处理等

# 基本概念

- 实际上计算机应该分为
  - 硬件
  - 操作系统
  - 应用程序
- 在计算机层次划分上称为
  - 硬件层
  - 内核层
  - 用户层



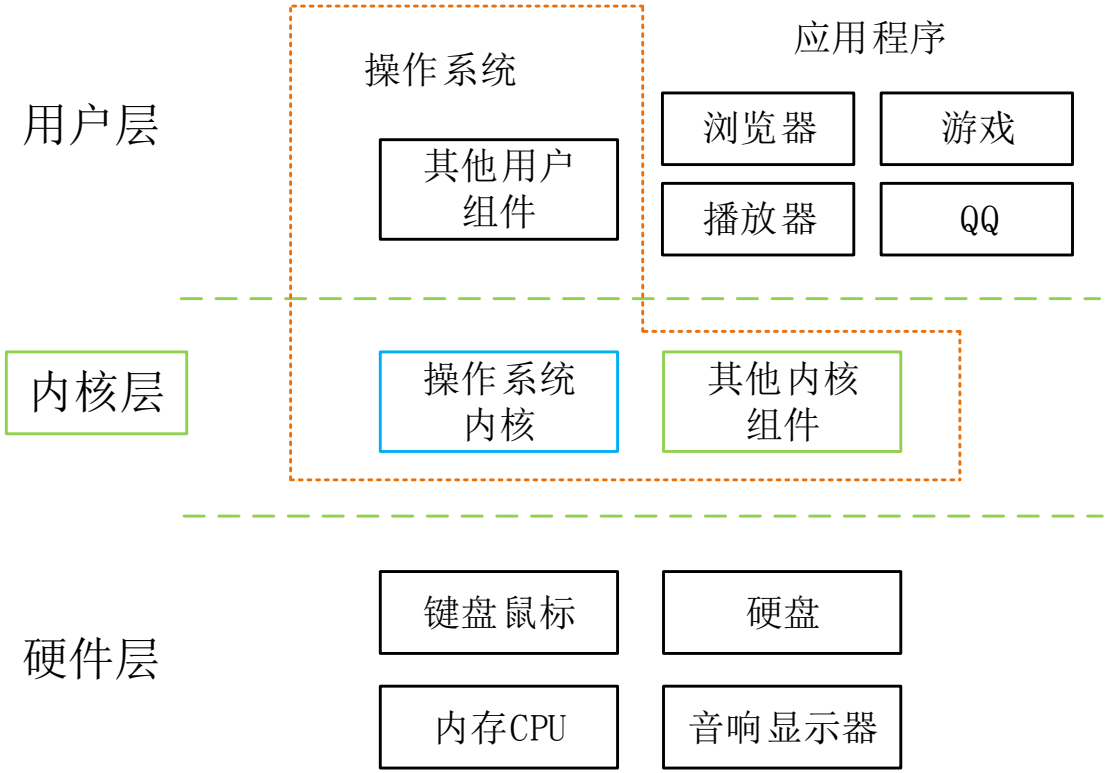
# 操作系统与操作系统内核

# 操作系统与操作系统内核

- 操作系统与操作系统内核的关系
  - 最初，操作系统=操作系统内核，即实现硬件抽象和资源管理的一套程序/代码
  - 但是，为了进一步丰富功能，为上层用户、应用程序提供便利，操作系统逐渐在操作系统内核的基础上增加了许多功能
    - 文件系统
    - 桌面
    - 一堆DLL、API、库
    - 系统工具（cmd/bash、ls、cat、rm……）
    - 其他（磁盘整理、画图、纸牌……）

# 操作系统与操作系统内核

- 在结构上，操作系统包含操作系统内核与其他组件
  - 其中操作系统内核完全处于“内核层”
  - 其他组件中部分内容处于“内核层”，部分内容处于“用户层”
- 在计算机安全领域，当提到“内核”时指的是操作系统中处于内核层中的全部内容，而不仅仅只是操作系统内核



# 操作系统与操作系统内核

- 操作系统内核的核心功能是硬件抽象、资源管理和任务调度
- 硬件抽象
  - 操作系统内核将底层的具体的、有差异的真实硬件设备，抽象成为一个抽象的、相同的虚拟硬件设备，并规范出一套统一的交互接口
  - 上层的用户、应用程序不需要关注真实硬件设备的具体型号、交互方式等细节问题，只需要通过这套统一的交互接口访问硬件即可
  - CPU、硬盘、内存……
  - 硬件差异特别大时这种抽象也会失败……

# 操作系统与操作系统内核

- 硬件抽象的实现不仅仅是靠内核完成的，还需要硬件的配合——驱动程序
  - 驱动程序即添加到操作系统中的一小块代码，其中包含有关硬件设备的信息，操作系统以此与设备进行通信
- 内核的工作就是管理这些硬件的驱动程序，做一个上层程序与底层硬件的之间的“翻译官”

# 操作系统与操作系统内核

- 资源管理
  - 操作系统内核统筹规划整个计算机的硬件使用情况，为上层的应用程序分配和调度资源
  - 典型的资源有
    - 寄存器
    - 内存
    - I/O设备
  - 寄存器管理
    - 上层不同的应用程序对应不同的进程、线程，每个线程运行过程中寄存器值不同
    - 因此在线程切换的过程中需要保存上一个线程的寄存器状态、载入下一个线程的寄存器状态
    - 程序计数器（PC），堆栈指针（SP），通用寄存器以及MMU(Memory Management Uinit)页表等

# 操作系统与操作系统内核

- 资源管理
  - 内存管理
    - 多线程环境下，每个进程/线程有自己独立的虚拟内存空间，操作系统内核需要将虚拟内存空间映射到实际的物理内存中
    - 对内存的管理在分页式内存管理模式下体现为对内存页的管理，包括
      - 在线程申请新的内存、访问到未分配的内存空间时新建页
      - 线程结束、释放内存时回收页
      - 物理内存不足时将部分内存页交换到硬盘空间

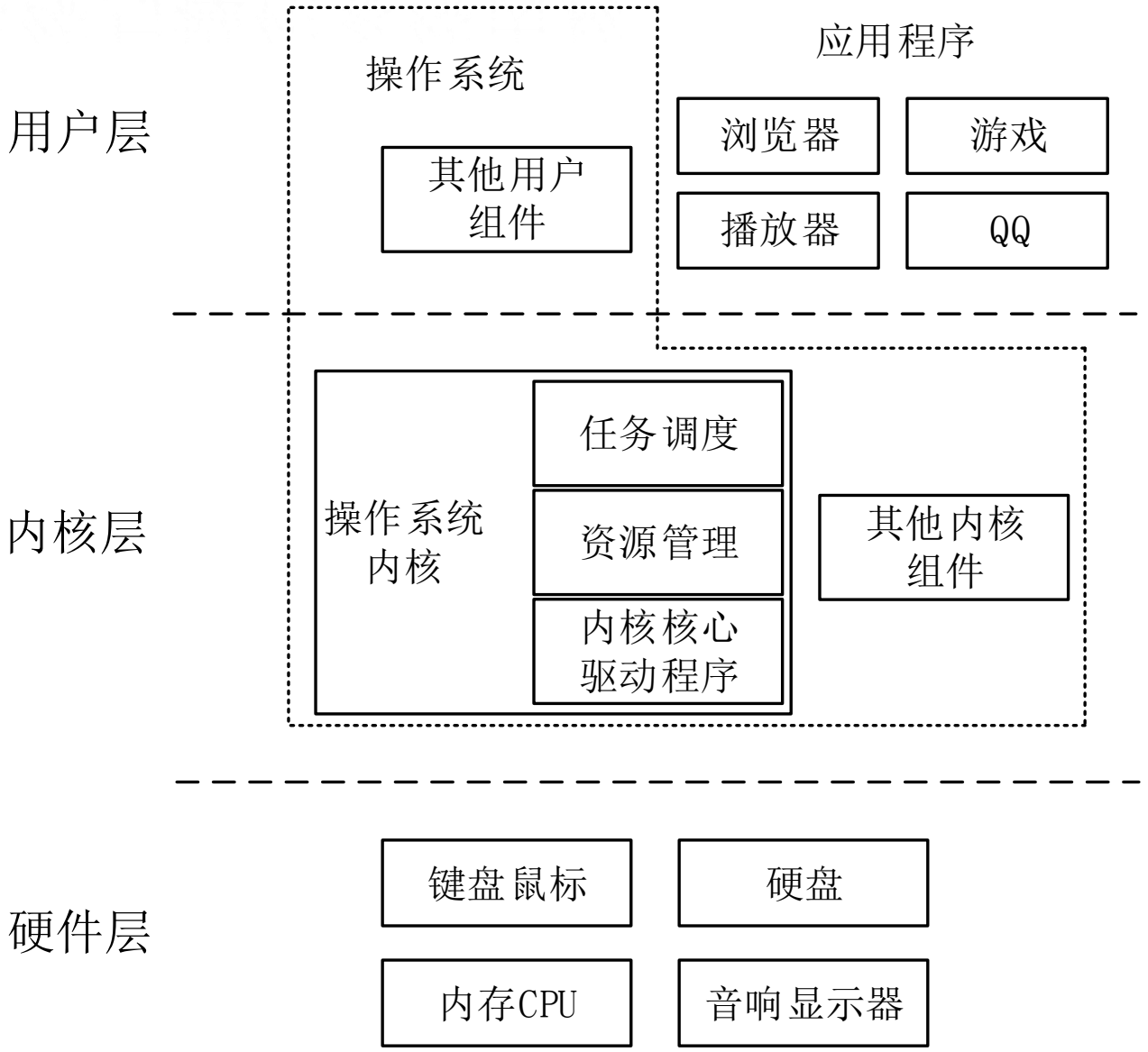
# 操作系统与操作系统内核

- 资源管理
  - I/O设备管理
    - 计算机I/O设备众多，部分设备具有独占性（硬盘、打印机）、部分具有公用性（鼠标、键盘）
    - 对独占I/O设备，操作系统内核需要管理I/O请求队列，根据物理设备的状态调度上层的线程
    - 对公用I/O设备，操作系统需要管理I/O设备的数输入出，转换为通知、请求、响应通过钩子、事件等方式传递给上层的线程
    - I/O设备的资源管理十分重要，在保证高效的基础上还需要避免各种死锁、竞争、饥饿等资源调度问题

# 操作系统与操作系统内核

- 任务调度
  - 对于单核操作系统，同一时间只能运行一个线程，因此需要通过时分复用的方式将CPU轮流分配给不同线程以实现结果上的“并行”
  - 操作系统的任务调度包括进程/线程切换、环境保存与恢复、中断/事件处理等
  - 其中涉及到很多任务调度算法，以保证最为合理的任务调度结果，以及特殊事件的处理机制
  - 任务调度和资源管理密不可分，部分时候在介绍操作系统功能和结构时会将2者直接合并

# 操作系统与操作系统内核



# 用户层与内核层关系

# 用户层与内核层关系

- 操作系统内核为什么要处于内核层？
  - CPU厂商在设计CPU时使用了分层式的架构
    - 例如X86架构的CPU工作模式有四层：RING0、RING1、RING2、RING3
    - RING0是内核层， RING3是用户层
    - 当CPU工作在RING0模式下时，称为内核模式
    - 当CPU工作在RING3模式下时，称为用户模式
  - 层次越低，能执行指令权限越高，通过这种方式能够实现计算机访问控制，从而保证计算机的安全

# 用户层与内核层关系

- 内核模式与用户模式有什么区别？
  - 用户模式下的进程/线程使用独立内存空间，内核模式下的线程使用公用统一的内存空间
    - 用户模式下，每个程序/进程有单独的内存空间（32位Windows中为4G内存空间中的高2G），无法直接跨进程访问其他进程的内存空间，也无法访问属于内核模式的内存空间
    - 内核模式下，所有线程统一使用同一段内存空间（ 32位Windows中为4G内存空间中的低2G ），同时可以访问属于用户层模式内存空间
  - 内核模式下能够执行更加高权限的指令
    - 一些涉及到内核相关工作指令在用户模式下直接执行的，典型的例如文件的读写的指令

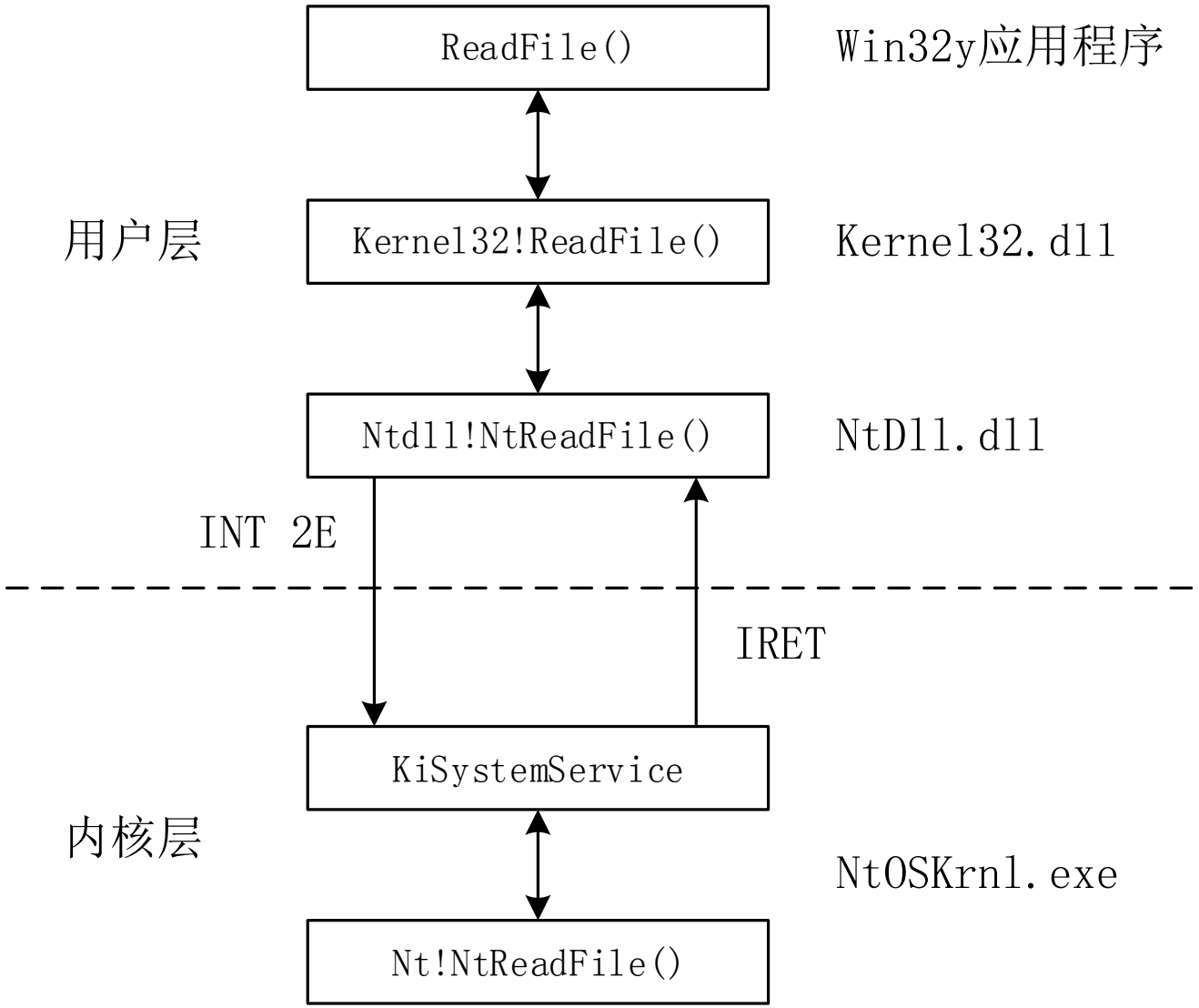
# 用户层与内核层关系

- 为了方便用户层的程序能够顺利使用一些需要在内核模式下才能完成的功能，操作内核提供了“系统调用（system call）”
- 系统调用通常通过中断的方式实现，用户层的程序通过发送中断请求，使得CPU由用户模式切换到内核模式并响应该请求，在完成请求内容后返回
  - 例如在用户模式下执行INT 2E中断，会调用内核模式下的nt!KiSystemService
- 操作系统内核提供了一张中断向量描述表(IDT）来保存各种中断对应的内核模式处理函数，用户模式中的程序可以按需使用

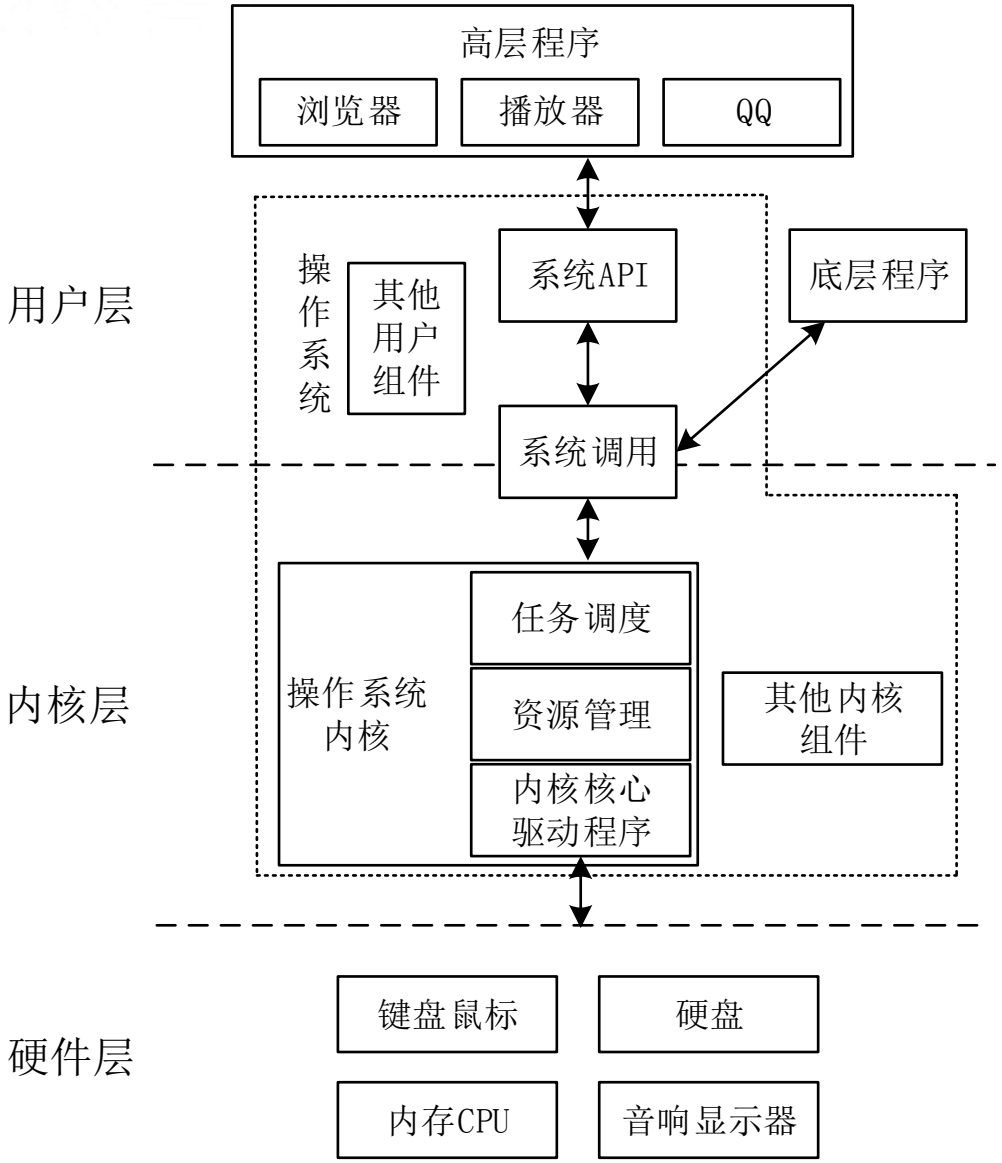
# 用户层与内核层关系

- 但这还是太麻烦了……于是操作系统在此基础上进一步包装，将常用的功能分门别类提前设计好，形成一套应用程序编程接口（API）供应用程序使用
  - 例如ntdll.dll中有各种windows提供的从用户模式切换到内核模式的API接口函数，并以此为基础进一步封装形成了kernel32.dll和user32.dll
  - 当然也有不使用API直接通过系统调用来实现内核模式切换的程序

# 用户层与内核层关系


ReadFile()
Win32y应用程序
用户层
Kernel32!ReadFile()
Kernel32.dll
Ntdll!NtReadFile()
NtDll.dll
INT 2E
IRET
KiSystemService
内核层
NtOSKrnl.exe
Nt!NtReadFile()

# 用户层与内核层关系


高层程序
浏览器
播放器
QQ
用户层
操作系统
其他用户组件
系统API
底层程序
系统调用
内核层
操作系统内核
任务调度
资源管理
内核核心驱动程序
其他内核组件
硬件层
键盘鼠标
硬盘
内存CPU
音响显示器

# 基本概念

## 内核安全

# 用户层与内核层关系

- 由于内核模式下的高权限（可以执行特殊指令）和自由性（共享内存、可以访问全部内存空间），在内核模式下完成一些事情会更加有效率，因此安全软件和电脑病毒都十分青睐于“入侵”到内核空间
  - 恶意行为拦截、保护文件与程序
  - 键盘记录、隐藏木马
- 在内核层中增加一个自定义的功能最常用的方式是“加载内核驱动”，即将自己的代码放到内核空间中并在操作系统内核里注册
  - 实际中，内核层里有大量并非操作系统自身的驱动
  - 这些驱动程序通常来自于可信赖的第三方，例如硬件厂商、网络服务供应商等等

# 用户层与内核层关系

- 但是如果有恶意程序进入到了内核空间，会带来极大的安全问题
  - 由于任意内核线程都可以访问全部内核空间，因此恶意程序可造成的破坏后果很严重
  - 所有内核线程的权限相等，恶意程序与反恶意程序可以互相攻击，胜负全凭谁先手
  - 受限于内核的敏感性以及对效率的追求，反恶意程序在扫描、查杀的幅度上受到的束缚，极容易出现漏查漏杀

# 用户层与内核层关系

- 目前，对操作系统内核安全的保障依赖于2个方面
  - 入口把关，通过修复漏洞、访问控制、特征识别等方式防止恶意程序进入内核空间
  - 底层查杀，通过底层的硬件或虚拟化技术获得更高的权限从而有效对抗恶意程序
- 与此同时，恶意代码的攻击方式也向着2个方面发展
  - 深度伪装，通过各种漏洞利用、反签名、ROP等方式绕过入口把关进入内核
  - 底层攻击，利用Bootkit技术或虚拟化逃逸的方式进入更底层的空间对抗反恶意软件

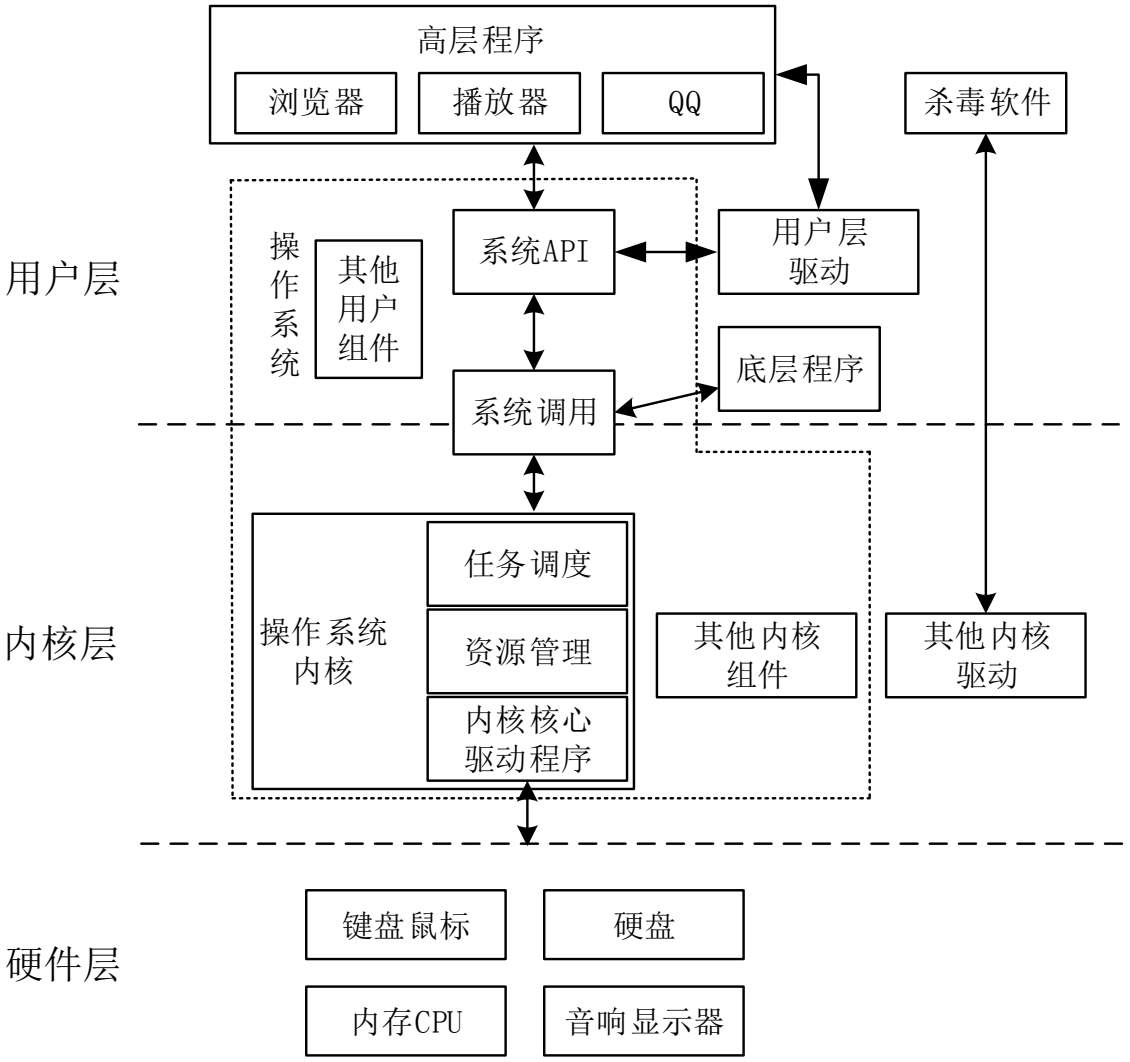
# 基本概念

其他

# 其他

- 操作系统中的任务调度、资源管理、硬件抽象都有很深的学问
- 为避免加载内核层驱动带来的安全隐患同时保证内核稳定，Windows倡导多使用用户层驱动
- 内核模式具有的高权限和管理员账户具有的高权限存在本质区别，前者是由CPU架构决定的，后者是由操作系统决定的
- “其他内核组件”有文件系统、电源管理、防火墙等等

# 其他


高层程序
浏览器
播放器
QQ
杀毒软件
用户层
操作系统
其他用户组件
系统API
用户层驱动
底层程序
系统调用
内核层
操作系统内核
任务调度
资源管理
内核核心驱动程序
其他内核组件
其他内核驱动
硬件层
键盘鼠标
硬盘
内存CPU
音响显示器

# 谢谢！

大成若缺，其用不弊。大盈若冲，其用不穷。大直若屈。大巧若拙。大辩若讷。静胜躁，寒胜热。清静为天下正。