

## Linux Kernel 阅读 1

### 1. 目录结构（自顶向下列举名称及功能）

Arch 文件夹：architecture 的缩写，子目录下有各种体系结构的子目录，子目录下又包含开机引导程序和配置文件、库文件以及 makefile 等；

Block 目录：管理块设备的代码；

Crypto 目录：加密和校验算法；

Documentation 目录：文档目录；

Drivers 目录：驱动目录，列举了 Linux 内核支持的硬件设备的驱动源代码；

Firmware 目录：固件目录；

Fs 目录：filesystem 的简写，文件系统的实现代码；

Include 目录：内核所需要的头文件。与平台无关的头文件在 include/linux 子目录下，平台相关的头文件则放在相应的子目录；

Init 目录：内核初始化代码；

Ipc 目录：进程间通信代码实现；

Kernel 目录：操作系统内核的核心功能代码，进程调度等；

Lib 目录：库文件代码；

Mm 目录：memory management 的缩写，用于实现内存管理中与体系结构无关的部分；

Net 目录：网络协议实现的代码；

Samples 目录：内核编程的范例；

Scripts 目录：配置内核的脚本；

Security 目录：安全相关的代码；

Sound 目录：音频设备的驱动程序；

Tools 目录：Linux 下一些有用的工具；

Utr 目录：Cpio 命令实现，Cpio 命令用来建立、还原备份档的程序；

Virt 目录：内核虚拟机相关的代码；

### 2. 核心模块叙述（有选择）

Bios 启动、内核加载：

在 Arch 目录下包含了支持的各种体系结构的开机引导程序，以 X86 为例，开机引导程序位于 Arch/x86/boot

Arch/x86/boot/compressed

Arch/x86/kernel

主要文件是：Arch/x86/boot/header.S、Arch/x86/boot/compressed/head.S

arch/x86/boot/compressed/misc.c

过程描述：当内核镜像被调用后，从 boot 目录下的 header.S 的 start 入口开始，之后调用 compressed 目录下的 startup\_32。Start\_up 设置基本的环境。然后调用 misc.c 中的 decompress\_kernel 函数解压内核镜像。之后再调用 x86/kernel 目录下的 startup\_32 函数。最后调用 init/main.中的 start\_kernel 函数。

### 进程管理：

内核在 `include/linux/sched.h` 定义了 `struct task_struct` 结构体，`task_struct` 是进程描述符，提供了内核需要了解的进程信息，例如进程的状态等；`include/linux/pid.h` 定义了 `pid`。`arch\powerpc\platforms\cell\spufs\sched.c` 中实现了进程调度的算法。`Kernel/fork.c` 定义了课上讲的 Linux 系统的 `fork` 函数；`kernel` 目录下的 `exit.c` 和 `exec_domain.c` 定义了 `wait`、`exit`、`exec` 相关函数。

### 内存管理：

Linux 采用页作为内存管理的基本单位，标准页框大小为 4kb，内核需要记录每个页框的状态，所以内核使用了页描述符 `struct page`，定义在 `include/linux/mm_types.h` 中；`include/linux/gtp.h` 中定义了内存分配接口。

### 虚拟文件系统：

与平台有关的内容定义在 `Fs` 目录下，`Fs/nfsd/vfs.h` 定义了虚拟文件系统相关的内容。

### 网络子系统：

`Net` 模块定义了网络系统需要的相关内容

### 进程间通信：

`mqueue.c`、`msg.c`、`shm.c`、`sem.c` 以及 `socket.c` 分别定义了 Linux 系统中进程间通信的管道、消息、信号量、共享内存和套接字模型的代码。`Util.c` 中定义了进程间通信的接口。

### 参考文献格式：

[1] Understanding the Linux Kernel 3<sup>rd</sup> Edition