

创建显示系统进程信息的 proc 模块

1. 实验背景

Linux 遵循现代操作的普遍原则：使程序员和内核、硬件等系统资源隔离开，普通用户无法看到内核空间中发生了什么，系统调用是操作系统提供给应用程序使用操作系统服务的重要接口，但同时也屏蔽了用户直接访问操作系统内核的可能性。Linux 提供了 LKM 机制可以使我们在内核空间工作。Linux 提供的 LKM 机制中一个重要的组成部分就是 proc 伪文件系统。

Proc 包含了当前计算机中重要的资源信息及系统信息。

2. 实验目的

我们要实现一个内核模块，该模块创建 `/proc/tasklist` 文件，并且提取系统中所有进程的 pid、state 和名称进行显示。

3. 实验流程

1. 首先可以先看一下系统中 `proc` 目录下是否有 `tasklist` 这个文件夹
2. 编写模块以及 `makefile` 文件，并运行 `make` 进行编译
3. 内核模块添加 `$sudo insmod tasklist.ko`
4. 添加内核模块后读取并信息 `tasklist` 内核信息： `$ cat /proc/tasklist`

4. 实现效果

```
amos@ubuntu:~/Desktop/src$ sudo insmod tasklist.ko
amos@ubuntu:~/Desktop/src$ cat /proc/tasklist
#0      0      0      swapper/0
#1      1      1      systemd
#2      2      1      kthreadd
#3      3      1026   rcu_gp
#4      4      1026   rcu_par_gp
#5      6      1026   kworker/0:0H
#6      9      1026   mm_percpu_wq
#7      10     1      ksoftirqd/0
#8      11     1026   rcu_sched
#9      12     1      migration/0
#10     13     1      idle_inject/0
#11     14     1      cpuhp/0
#12     15     1      cpuhp/1
#13     16     1      idle_inject/1
#14     17     1      migration/1
#15     18     1      ksoftirqd/1
```

1. 阅读并填充 route_struct.c 代码，完成热身小实验。
2. 阅读并填充 tasklist.c 代码，然后完成实验。
3. 并尝试在 pid、state、comm 后边添加打印出你感兴趣的 task_struct 中的字段。