## 实验六 进程与任务管理

## 实验目的

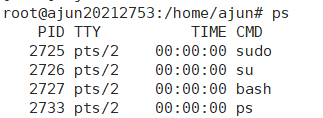
（1）熟悉进程管理常用命令

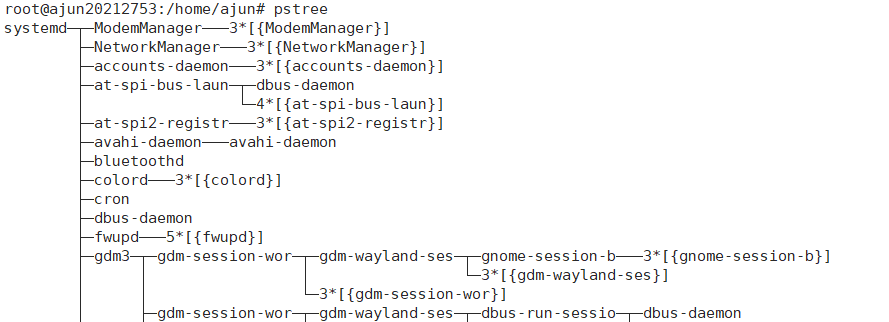
（2）熟悉定时任务与任务调度的设置

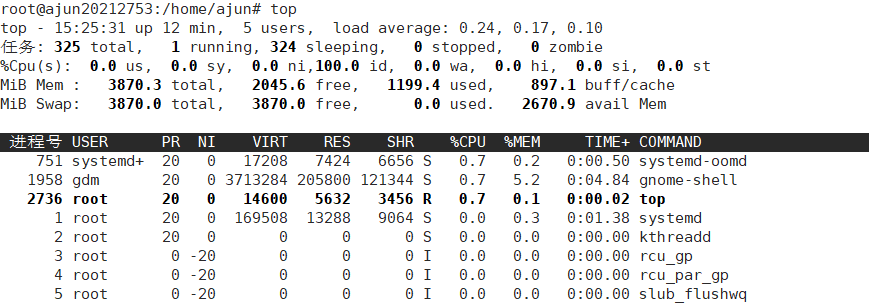
## 实验内容

**1. 进程管理**

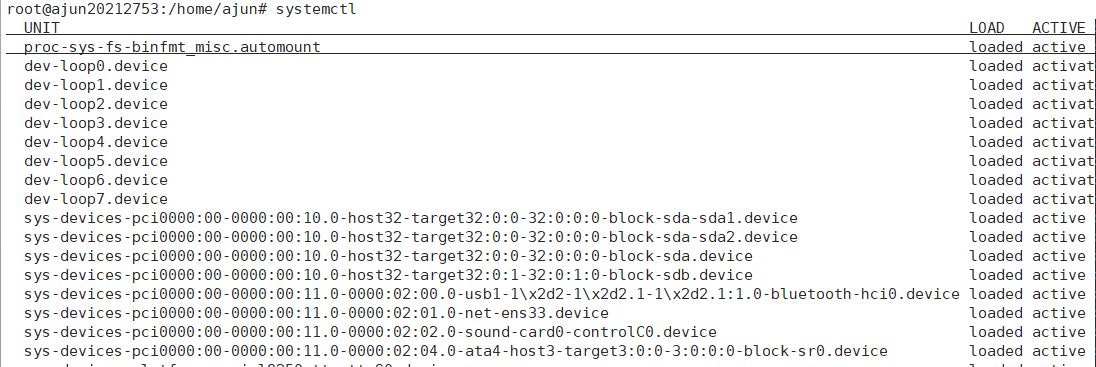
（1）练习进程状态查询命令ps、pstree、top

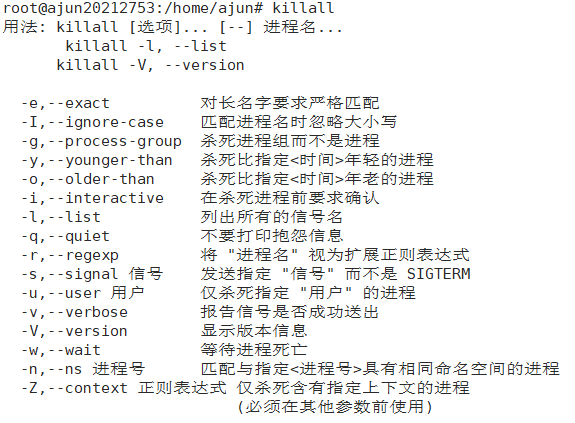




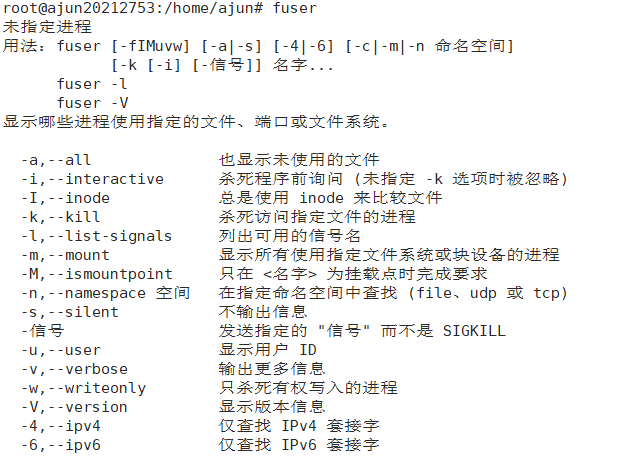


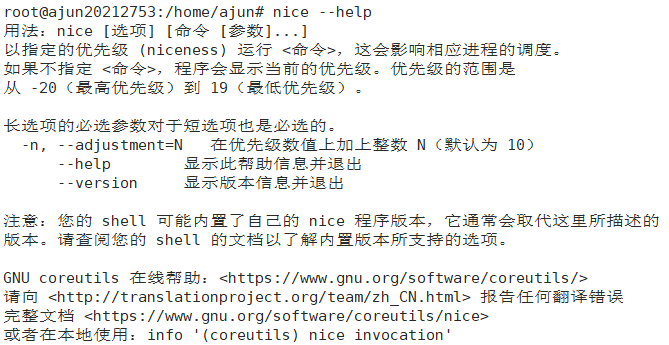
（2）练习进程管理命令systemctl、killall、pgrep/pkill、fuser、nice











（3）练习可执行文件的特殊属性操作

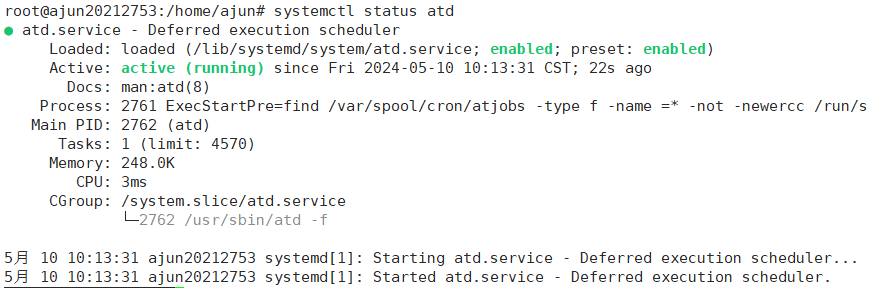
1）新建文件

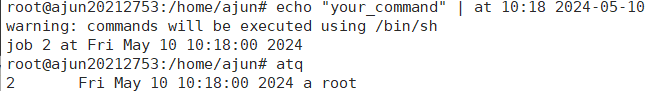
2）查看文件权限

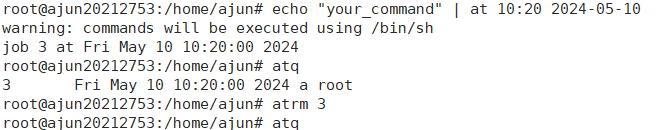
3）设置、修改、查看文件特殊权限suid、sgid及sticky

**2. 任务管理**

（1）利用at命令完成定时任务设置、查看并删除任务。注意：执行at时，需要先保证atd已启动，可执行systemctl命令重启atd，并观察atd的状态。



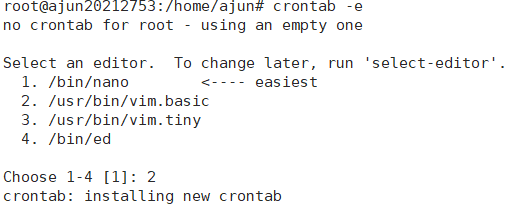




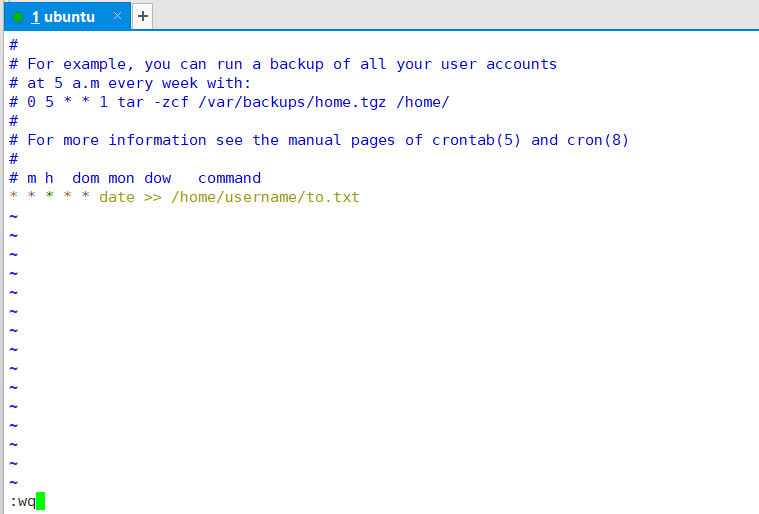
（2）利用 crontab完成任务调度设置：

* 设置任务调度文件/etc/crontab，具步骤如下：

1. 执行命令$ crontab -e，若是首次使用crontab，需选择系统已经安装的editors，建议选vi/vim；



1. 输入任务到调度文件：每隔一分钟，将当前日期追加到/home/username/to.txt；

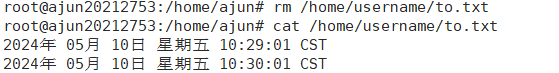


1. 保存退出，提示任务创建成功“crontab: installing new crontab”；



1. cat /home/username/to.txt，观察输出内容；



1. 删除/home/username/to.txt；
2. 一分钟后执行ls命令，查看是否有/home/username/to.txt重新生成。

* 有时候需要调度多个任务，可用crontab -e 增加多条任务，也可以将这多个任务写在一个脚本文件中，**以crond调用脚本的方式实现多个任务的提交**，但这种方式需要注意脚本文件有**执行权限**。请完成以下步骤：

1. 执行命令$ vim /home/username/my.sh，新建脚本文件并写入内容：

date >> /home/username/mycal

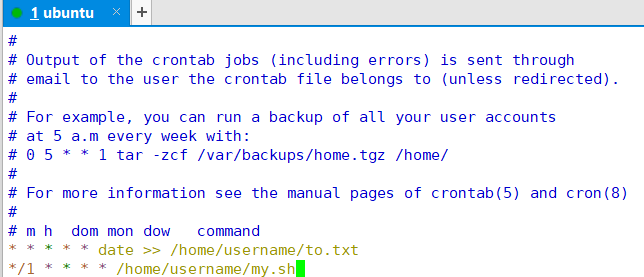
cal >> /home/username/mycal



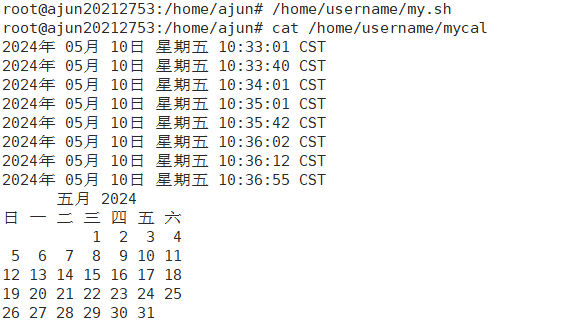
1. 给my.sh增加执行权限



1. 执行命令$ crontab -e ，增加任务：\*/1 \* \* \* \* /home/username/my.sh；保存退出。



1. 执行命令$ ./my.sh，执行脚本文件，观察输出；执行命令$ cat /home/username/mycal 查看mycal文件的内容，是否有日期和日历，且每隔一分钟，是否有追加内容。



1. 执行命令$ crontab -l ，列出当前定时任务



1. 执行命令$ crontab-r，删除定时任务



1. 重启crond。



## 实验总结

本次实验涉及了 at 和 crontab 两种常用的定时任务调度工具。通过 at 命令，能够设置一次性任务，对未来某一时刻执行操作十分便捷。而通过 crontab 命令，则可以设置周期性任务，例如每天、每周或每月执行特定任务。通过学习实验，我掌握了使用这两种工具的方法，能够更有效地管理系统任务，提高工作效率。