

# Linux 掌握的如何？

我能够使用一些命令和 shell 脚本对 linux 系统熟练操作，对系统中的文件进行管理，修改配置文件，查找日志等。管理系统中的用户和文件权限管理。查看系统运行状态和进程、服务，维护系统运行。还有网络，磁盘等操作。

## 文件管理

我掌握基本的文件操作。使用 touch 创建一个文件。对文件内容查看。文件内容较少可以使用 cat 输出全部内容。内容过多可以使用 less, more 分屏查看内容。head 可以看文件前几行。tail 也能动态的查看文件新增的内容用于日志追踪。

在项目中会进行**查找**.log 文件的操作，可以使用 grep 命令。ls /log | grep "\*.log"。查找文件可以用 find 命令，find /log -name "\*.log"。根据文件名查找-name，还有其它参数：-type, -user, -size 根据实际情况选择。

我们在项目中也会有**修改文本内容**的操作。例如替换配置文件中的域名。可以使用 sed 命令，sed "s/aa.com/bb.com/g config.ini。能使用正则表达式，删除匹配的行,也能添加，在 hostname 后面添加了一条配置：sed "hostname/a database=mysql"。或者使用 tee 命令 echo "port=3306" | tee -a config.ini。我还掌握 vim 的使用，在命令模式中使用 dd 删除光标所在行，yy 和 p 可以复制一行和粘贴，数字加 G 可以跳转到多少行。进入插入模式就可以任意修改内容。ESC 回到命令模式:wq 就保存退出。

**对文本内容截取**，在我们项目中也是常用的操作。可以使用 cut。以一个字符或多个字符进行分割行,获取需要的某列。比如日志文件以冒号分割的,并获取第一列:cat "access.log" | cut -d: -f 1。awk 也能完成这样的操作，它还能做一些运算。比如统计目录大小，先 cd 进入到目录中，使用 ll | awk '{sum+= \$5; print sum}'

对文件的**解压缩**也是常用操作，比如数据备份和还原等，可使用 zip 或 tar 命令。例如对文件和目录进行压缩：tar -zcvf backup.tar.gz file1 file2 dir1；解压文件操作 tar -zxvf filename;

## 用户管理

我还掌握用户和组的管理,用户创建和删除:useradd, userdel。组的创建,删除 groupadd, groupdel。设置用户和组的密码用 passwd uname 和 gpasswd gname。还有管理组中的成员，用 gpasswd -a uname gname，删除用-d 参数。-M 重新设置组员。查看组成员：getent group 组名。查看用户所属的组使用 id 命令：id 用户名；

## 权限管理

我们在项目还会遇到权限问题，使用 chmod 为文件的拥有者，组成员和其他用户设置读写执行的权限。例如为 shell 脚本添加可执行权限：chmod ugo+x clearLog.sh。如果要对整个目录中的内容修改权限就加上-R 参数。还有其它修改权限的命令 chown username file 更改文件的拥有者，chgrp gname filename 更改文件所属组。

## 进程管理

我还能够管理系统中的进程和服务。使用 ps -ef 能查看当前系统的进程。使用-aux 参数查看所有用户执行的进程。还可以使用 top 命令实时的系统状态，cpu 使用率，内存占用比和进程信息等。根据查看到进程的 PID，可以使用 kill 命令结束进程。

## 服务管理

我们在项目中会管理服务程序，httpd，mysql 等，在安装之后我们需要启动它，使用 systemctl 查看服务的状态，开启关闭等。设置服务自启或关闭自启。例如要查看 httpd 服务的状态：systemctl status httpd。开启或关闭：systemctl start 或 stop httpd。设置开机自动或关闭自启动：systemctl enable 或 disable httpd。

## 网络管理

我还掌握一些关于网络操作。使用 ifconfig 或者 ip addr 查看本机的 ip 地址。使用 ping ip 或者域名。查看是否与目标主机网络联通。还有使用 netstat 命令查看端口的使用情况。比如：那个进程使用了 80 端口：netstat -tunlp | grep ":80"。也可以使用简单的命令 lsof -i :80 查看。还有防火墙的开启关闭等。还有使用 firewall-cmd 命令加 --add-port=80/tcp，对 80 端口放行，这是临时的，需要永久的加上 --permanent 参数。然后重载防火墙 --reload。

## 磁盘管理

对于磁盘的管理我也有一些掌握。使用 du -ah 查看当前路径下每个文件的大小。df -h 查看整个系统磁盘使用的情况。还有当外部的硬盘连接时，使用 mount /dev/sad4 /data 命令将它挂载到 /data 目录下，硬盘的数据就可以从这个目录访问。卸载硬盘使用 umount /data。还有使用 fdisk 对磁盘进行分区。

## 软件安装

对于软件的安装，我掌握多种方式：使用 yum 安装 httpd：yum -y install httpd。或者下载二进制的安装包，使用 rpm -ivh 加文件名安装。还有就是下载软件的源码，例如使用 wget 或 curl 下载 lrzsz 源码压缩包，解压后可以修改安装路径，使用 make 命令进行编译，编译完成后使用 make install 命令安装软件。

## Shell 脚本编写

关于 shell 脚本的编写，我也掌握很多。定义一个用户变量：hostname=aa.com；使用 set | grep hostname 查看变量。要创建系统变量，可以使用 export。例如：export uname=tom；env 查看系统变量；用户变量和系统变量清除可以使用 unset；

在 shell 脚本中，可以直接使用一些特殊的变量：\$\*：接收运行脚本时传入的所有参数；\$#：传输脚本参数的个数；\$?：记录上一条命令是否成功执行；\$\$：当前脚本的执行的进程 id；还有位置变量；脚本最大支持 9 个参数，分别用 \$1-\$9 接收。

在脚本中掌握流程控制语句的使用：if 条件判断，case 条件选择，循环 for，while，until。其中最重要的就是 if 条件判断。使用 test 命令做出判断；比如判断一个目录或文件是否存在，可以使用 test -e 路径；或者用中括号的形式表达；[ -e 路径 ]；判断是否是文件或者目录，分别用 -f 和 -d 参数。还有数字判断：比如判断 age 变量是否大于 18：[ \$age -gt 18 ]；-lt 判断是否小于，-eq 和 -ne 判断是否等于或者不等于。多条件的可以使用 -a 或 -o 连接，表示逻辑与和逻辑或；也可以使用 && 和 || 更直观；

在脚本中也会进行重定向的操作；命令的结果可能需要保存下来；就可以在命令的后面加上重定向符 > 加上文件路径；也可以双 >> 追加内容到文件中；还有错误重定向符号，只负责把错误信息保存下来；比如 find / -name "\*.log" > info.log 2> error.log，分别把正常输出和错误输出保存到 info.log 和 error.log 中；

## 定时任务

我们在项目中会经常使用到定时任务 `crontab`，临时任务 `at`。查看定时任务 `crontab` `-l`，`-e` 参数进行任务编辑。任务分为六个部分，前五个都是时间代表：分钟,小时，日，月，星期，最后是 shell 脚本绝对路径。例如每天晚上 3 点执行备份脚本：`0 3 * * * /shell/backup.sh`。还有临时任务的使用：`at` 命令，比如在今天 9:30 要执行数据处理脚本，`at -f` 脚本绝对路径 9:30。明天 9:30 可以再加个 `tomorrow`。使用 `-l` 参数可以查看到已添加的临时任务，`-r` 参数加上任务 `id` 可以删除定时任务。

以上就是我对 linux 掌握的情况，内容可能过多，可能存在介绍遗漏的部分。还能进行其它部分补充。