



# 一、文件与目录操作

## （一）基础命令

1. **ls**：用于查看目录内容。
   * 常用选项：
     + -a：显示所有文件，包括隐藏文件（以.开头的文件）。例如ls -a，可列出当前目录下所有文件，让原本隐藏的配置文件等也能展示出来，方便用户全面了解目录构成。
     + -l：以列表形式展示详细信息，包括文件权限、所有者、大小、修改时间等。如ls -l，输出类似-rw-r--r-- 1 user group 1024 Jan 1 12:00 file.txt，从这些信息中，用户能清晰知晓文件的访问权限、归属以及大致内容量等关键信息。
     + -h：与-l结合使用时，以更易读的方式显示文件大小单位（如 KB/MB/GB 等）。如ls -lh，对于大文件，会显示为1.2M等形式，而不是一串难以直观理解的数字。
   * 可指定路径查看特定目录内容，如ls /home/user/Documents，将列出/home/user/Documents目录下的文件和子目录。
2. **cd**：用于切换工作目录。
   * 可使用绝对路径（以/开头），如cd /usr/local，直接切换到系统的/usr/local目录，这在需要精确指定系统特定位置时很有用。
   * 也可使用相对路径，如./表示当前目录，../表示上级目录。cd ../可返回上一级目录；cd ./subdir可进入当前目录下的subdir子目录。
   * cd ~可快速切换到当前用户的 HOME 目录，每个用户都有自己特定的 HOME 目录，存放个人配置文件等，此命令方便用户快速回到个人专属工作空间。
3. **pwd**：用于显示当前工作目录路径。执行pwd命令，会输出当前所在的完整路径，例如/home/user/Downloads，让用户随时明确自己在文件系统中的位置，避免操作失误。
4. **mkdir**：用于创建目录。
   * 基本用法为mkdir 路径，如mkdir new\_dir，在当前目录下创建名为new\_dir的新目录。
   * 若要创建多级目录，可使用-p选项，如mkdir -p a/b/c，会依次创建a目录，在a目录下创建b目录，再在b目录下创建c目录，无需用户手动逐级创建，方便构建复杂目录结构。
5. **touch**：用于创建空文件。例如touch new\_file.txt，会在当前目录下创建一个名为new\_file.txt的空文件，常用来初始化一些临时文件或用于后续编辑的文件。

## （二）文件操作

1. **cp**：用于复制文件或文件夹。
   * 复制单个文件：cp 源文件 目标路径，如cp file.txt /home/user/Documents/，将当前目录下的file.txt文件复制到/home/user/Documents目录中。
   * 复制文件夹时，需使用-r选项（递归复制），如cp -r my\_folder /backup/，会将my\_folder文件夹及其包含的所有文件和子文件夹完整复制到/backup目录下。
2. **mv**：用于移动或重命名文件 / 文件夹。
   * 移动文件：mv 源文件 目标路径，若目标路径存在，则将源文件移动到该路径下。如mv file.txt /home/user/Pictures/，将file.txt文件从当前位置移动到/home/user/Pictures目录。
   * 重命名：当目标路径与源文件在同一目录且文件名不同时，可实现重命名。如mv old\_name.txt new\_name.txt，在当前目录下将old\_name.txt重命名为new\_name.txt。
3. **rm**：用于删除文件 / 文件夹。
   * 删除文件：rm 文件，如rm file.txt，删除当前目录下的file.txt文件，系统会提示确认删除操作，以防止误删。
   * 删除文件夹需使用-r选项（递归删除），如rm -r my\_folder，将删除my\_folder文件夹及其所有内容。
   * -f选项用于强制删除，不提示确认，如rm -rf file\_or\_dir，但此操作非常危险，特别是rm -rf /命令，会删除根目录下所有文件，导致系统崩溃，务必谨慎使用。

# 二、文件查看与编辑

## （一）查看文件内容

1. **cat**：用于显示文件全部内容。如cat file.txt，会将file.txt文件的所有内容一次性全部输出在终端上，适合查看内容较少的文件。
2. **more**：用于分页查看文件。执行more file.txt后，文件内容会以一页一页的形式展示，按空格键可翻页，按q键可退出查看。对于内容较多的文件，使用more能更方便地逐页阅读，避免内容过多在终端一闪而过。
3. **tail**：用于查看文件尾部内容。
   * 基本用法tail 文件名，默认显示文件的最后 10 行内容。
   * -f选项可实时追踪文件更新，常用于查看日志文件，如tail -f app.log，当app.log文件有新内容写入时，终端会实时显示新增的内容，方便运维人员监控程序运行状态。
   * -num选项可指定显示最后num行，如tail -20 file.txt，将显示file.txt文件的最后 20 行内容。

## （二）文件搜索

1. **find**：用于按名称或大小等多种条件搜索文件。
   * 按名称搜索：find 起始路径 -name "文件名"，如find /home -name "\*.txt"，会在/home目录及其所有子目录中搜索所有扩展名为.txt的文件，可帮助用户快速定位特定类型文件。
   * 按大小搜索：如find / -size +10M，会在整个系统（从根目录/开始）中查找大于 10MB 的文件，对于磁盘空间管理，查找大文件以进行清理等操作很有帮助。
2. **grep**：用于按关键词过滤文件行。grep (-n) "关键词" 文件名，如grep "error" log.txt，会在log.txt文件中搜索包含error关键词的行并输出；grep -n "error" log.txt则会在输出结果中同时显示行号，方便用户快速定位关键词所在位置，常用于日志分析，查找错误信息等。

## （三）文本编辑（以 vim 为例）

1. **基础操作**：
   * 打开文件编辑：vim 文件名，如vim config.txt，进入vim编辑器界面。
   * 进入编辑模式：按i键，此时左下角会显示-- INSERT --，表示进入编辑状态，可对文件内容进行插入、修改等操作。
   * 返回命令模式：按ESC键，从编辑模式回到命令模式，可执行保存、退出等命令。
   * 保存并退出：在命令模式下输入:wq，即写入（w）并退出（q），文件修改内容会保存并退出vim编辑器。
   * 强制退出不保存：输入:q!，可在不保存修改内容的情况下强制退出vim。
2. **快捷操作**：
   * dd：在命令模式下，可删除当前行。如将光标移动到要删除的行，按dd键即可删除该行。
   * yy：用于复制当前行。将光标置于要复制的行，按yy键复制该行内容到缓冲区。
   * p：用于粘贴，在复制（如使用yy）后，将光标移动到要粘贴的位置，按p键可将缓冲区内容粘贴到当前位置。
   * /关键词：在命令模式下输入，可在文件中搜索指定关键词，输入后按回车键，光标会定位到第一个匹配关键词的位置。
   * n：在搜索后，按n键可跳转到下一个匹配项，方便依次查看文件中所有匹配关键词的位置。

# 三、权限与用户管理

## （一）权限控制

1. **权限格式**：文件或目录的权限以drwxr - xr - x这样的形式表示。
   * 第一个字符d表示该对象是目录（如果是-则表示普通文件）。
   * 接下来的三个字符rwx表示文件所有者的权限，r（读，4）、w（写，2）、x（执行，1），这里rwx表示所有者有读、写、执行权限，数值计算为4 + 2 + 1 = 7。
   * 再接下来三个字符r - x表示所属组的权限，这里表示组内成员有读和执行权限，数值为4 + 0 + 1 = 5。
   * 最后三个字符r - x表示其他用户的权限，同样是读和执行权限，数值为5。所以755是一种常见的权限设置表示方式。
2. **chmod**：用于修改权限，有数字法和符号法，数字法较为常用。
   * 数字法：chmod 权限数字 文件名，如chmod 755 file.txt，将file.txt文件的权限设置为所有者有读、写、执行权限，所属组和其他用户有读和执行权限。权限数字通过r = 4，w = 2，x = 1的组合计算得出，如6表示r（4）+ w（2）+ 0（x）即读写权限。
3. **chown**：用于修改文件所有者。chown 用户:组 文件名，如chown new\_user:new\_group file.txt，将file.txt文件的所有者改为new\_user，所属组改为new\_group，在多人协作或调整文件归属时常用。

## （二）用户与组

1. **创建用户**：useradd -g 组名 -d /home/用户名 用户名，如useradd -g users -d /home/newuser newuser，创建一个名为newuser的用户，该用户属于users组，家目录设置在/home/newuser。
2. **切换用户**：
   * su - root：可切换到root用户，root用户拥有最高权限，可进行系统级别的管理操作，但使用时需谨慎，避免误操作带来系统风险。
   * exit：用于退出当前用户，返回上一级用户环境。

# 四、系统监控与管理

## （一）进程管理

1. **ps**：用于查看进程信息。
   * ps -ef：可显示所有进程的详细信息，输出结果包含进程的所有者、进程 ID（PID）、父进程 ID（PPID）、启动时间等，如UID PID PPID C STIME TTY TIME CMD这样的表头下，列出各个进程的对应信息，方便管理员全面了解系统中正在运行的进程情况。
2. **kill**：用于终止进程。
   * kill 进程ID：可正常终止指定进程 ID 的进程，如kill 1234，尝试正常关闭进程号为1234的进程。
   * kill -9 进程ID：-9选项为强制终止，对于一些无法正常关闭的进程，可使用此命令强制结束，但可能导致数据丢失等问题，一般作为最后的手段，如kill -9 1234，强制关闭进程号为1234的进程。
3. **top**：用于实时监控系统资源。执行top命令后，会实时显示系统的 CPU 使用情况、内存使用情况、进程状态等信息。
   * 按P键可按 CPU 使用率对进程进行排序，方便查找占用 CPU 资源高的进程，排查系统性能瓶颈是否由某些进程过度占用 CPU 引起。
   * 按M键可按内存使用率排序，帮助发现占用大量内存的进程，以便进行优化或调整。

## （二）网络管理

1. **ping**：用于测试网络连通性。ping -c 次数 目标地址，如ping -c 3 baidu.com，表示向baidu.com发送 3 次 ICMP 请求包，通过返回的结果可判断网络是否通畅，以及网络延迟等情况，-c选项指定发送请求的次数。
2. **netstat**：用于查看端口占用。netstat -anp | grep 端口号，如netstat -anp | grep 80，可查看占用 80 端口的进程信息，其中-a显示所有连接和监听端口，-n以数字形式显示地址和端口号，-p显示使用该端口的进程，通过grep过滤出指定端口的相关信息，方便排查端口被哪个程序占用，解决端口冲突等网络问题。

# 五、压缩与解压

## （一）tar

1. **打包与压缩**：tar -zcvf 压缩包名.tar.gz 文件1 文件2...，如tar -zcvf backup.tar.gz /home/user/Documents /home/user/Pictures，将/home/user/Documents和/home/user/Pictures目录及其内容打包并压缩成backup.tar.gz文件，-z表示使用 gzip 压缩算法，-c表示创建新的归档文件，-v显示详细过程，-f指定归档文件名。
2. **解压**：tar -zxvf 压缩包名.tar.gz -C 目标路径，如tar -zxvf backup.tar.gz -C /backup，将backup.tar.gz压缩包解压到/backup目录下，-x表示解压缩，-C指定解压的目标路径。

## （二）zip/unzip

1. **压缩**：zip -r 压缩包名.zip 文件夹，如zip -r my\_folder.zip my\_folder，将my\_folder文件夹及其内容压缩成my\_folder.zip文件，-r表示递归压缩，即包含文件夹内的所有子文件和子文件夹。
2. **解压**：unzip 压缩包名.zip -d 目标路径，如unzip my\_folder.zip -d /extract，将my\_folder.zip压缩包解压到/extract目录下，-d指定解压的目标目录。

# 六、注意事项

1. **高危操作**：rm -rf /命令极其危险，会删除系统根目录下的所有文件和文件夹，导致系统完全崩溃且难以恢复，在任何情况下都要避免执行此命令，务必谨慎操作。
2. **权限控制**：修改系统关键文件或执行系统管理操作通常需要root权限，但直接使用root用户操作存在风险，建议普通用户在需要时使用sudo命令，如sudo apt update，以临时获取root权限执行命令，同时sudo命令会记录操作日志，便于追溯和管理。
3. **环境变量**：修改系统环境变量文件（如/etc/profile）后，需执行source /etc/profile命令使修改生效，新的环境变量设置才能在当前终端会话中被识别和应用，否则新设置可能不会立即起作用。