# 常用快捷键

## ctrl + l

清屏

## ctrl + c

终止命令

## ctrl + a

光标移到行首

## ctrl + e

光标移到行尾

## !!

上一条命令

## ctrl+insert

复制

## shift+insert

粘贴

## ctrl+shift+F

查找

## ctrl+b

屏幕往“后”移动一页

## ctrl+f

屏幕往“前”移动一页

## ctrl+u

屏幕往“后”移动半页

## ctrl+d

屏幕往“前”移动半页

## 按数字0

移到行的开头

## 按「$」

移动到光标所在行的“行尾”

## 按「G]

移动到文件的最后一行

# 基本概念

Linux 是一套免费使用和自由传播的似 Unix 操作系统，一般的 WEB 项目都是部都是放在 Linux 操作系统上面。 Linux 是一个基于 POSIX 和 Unix 的多用户多任务、支持多线程和多 CPU 的操作系统。它能运行主要的 Unix 工具软件、应用程序和网络协议。它支持 32 位和 64 位硬件。Linux 继承了 Unix 以网络为核心的设计思想，是一个性能稳定的多用户网络操作系统。

# Windows和Linux的区别

1、Windows 是微软开发的操作系统，民用操作系统，可用于娱乐、影音、上网。Windows 操作系统具有强大的日志记录系统和强大的桌面应用。好处是它可以帮我们实现非常多绚丽多彩的效果，可以非常方便去进行娱乐、影音、上网。

2、 Linux 的应用相对单纯很多，没有什么绚丽多彩的效果，因此 Linux 的性能是非常出色的，可以完全针对机器的配置有针对性的优化

3、简单来说 Windows 适合普通用户进行娱乐办公使用，Linux 适合软件开发部署。

# Unix和Linux有什么区别

Linux 和 Unix 都是功能强大的操作系统，都是应用广泛的服务器操作系统，有很多相似之处，甚至有一部分人错误地认为 Unix 和 Linux 提作系统是一样的，然而事实并非如此，以下是两者的区别。

## 1、开源性

Linux 是一款开源操作系统，不需要付费即可使用，Unix 是一款对源码实行知识产权保护的传统商业软件，使用需要付费授权使用。

## 2、跨平台性

Linux 操作系统具有良好的跨平台性能，可运行在多种硬件平台上;Unix 操作系统跨平台性能较弱，大多需与硬件配套使用。

## 3、可视化界面

Linux 除了进行命令行操作，还有窗体管理系统; Unix 只是命令行下的系统

## 4、硬件环境

Linux 操作系统对硬件的要求较低，安装方法更易掌握;Unix 对硬件要求比较苛刻，按照难度较大。

## 5、用户群体

Linux 的用户群体很广泛，个人和企业均可使用; Unix 的用户群体比较窄，多是安全性要求高的大型企业使用，如银行、电信部门等，或者 Unix 硬件厂商使用，如 Sun 等。相比于 Unix 操作系统，Linux 操作系统更受广大计算机爱好者的喜爱，主要原因是 Linux 操作系统具有 Unix 操作系统的全部功能，并且能够在普通 PC 计算机上实现全部的 Unix 特性，开源免费的特性，更容易普及使用。

# CentOS和Ubuntu的区别

## CentOS

CentOS 是一个开源的 Linux 发行版。你会发现非常多的商业公司部署在生产环境上的服务器都是使用的 CentOS 系统，CentOS 是从 RHEL 源代码编译的社区重新发布版，它被认为是企业IT世界中使用最广泛的。CentOS 是社区支持的企业级操作系统，于2004年发布。与 RHEL 非常相似，虽然不单独提供商业支持，但往往可以从 Redhat 中找到一丝线索。CentOS 简约，命令行下的人性化做得比较好，它具有高度可定制性，安全性和稳定性。与 RHEL 的密切联系使 CentOS 能够拥有相当多的企业级安全更新，从而使其成为很多用户的安全选择。

## [Ubuntu](https://so.csdn.net/so/search?q=Ubuntu&spm=1001.2101.3001.7020)

基于Debian 架构，自2004年首次发布以来，Ubuntu 一直是个人和专业用户的热门人选。Ubuntu 近些年的粉丝越来越多，Ubuntu 有着漂亮的用户界面，完善的包管理系统，强大的软件源支持，丰富的技术社区，Ubuntu 还对大多数硬件有着良好的兼容性，包括最新的图形显卡等等。这一切让 Ubuntu 越来越向大众化方向发展。Ubuntu 的图形界面固然漂亮，但这也决定了它最佳的应用领域是桌面操作系统而非服务器操作系统。如果你需要在服务器端使用图形界面，Ubuntu 是一个不错的选择，但是图形界面占用的内存非常大，而内存越大的 VPS 价格也越高。从这一点来讲，Ubuntu 并没有在 VPS 安装的操作系统选择之列。

## 主要差异

两个Linux发行版之间的最大区别是Ubuntu基于Debian 架构，而 CentOS 则来自Red Hat Enterprise Linux。

在Ubuntu中，您可以使用apt-get包管理器下载DEB包。同时在 CentOS 中，您必须使用 yum 命令从中央存储库下载和安装 RPM 软件包。



# ****Rocky Linux 9****

## 基本概念

Rocky Linux 是一个开源企业操作系统，设计为与企业级Linux 100% bug-for-bug 兼容 [Rocky Linux](https://rockylinux.org/)。它是Red Hat Enterprise Linux (RHEL) 的免费替代品，旨在提供与RHEL完全兼容的企业级Linux发行版。

## 软件包更新

Rocky Linux 9 包含了多个重要的软件包更新：

### 开发工具

动态编程语言的新版本

版本控制系统的更新

Web相关应用套件的升级

### 系统管理

增强的容器支持

改进的安全功能

更新的系统管理工具

## 优势

1、成本效益：免费替代RHEL

2、兼容性：与RHEL完全兼容

3、稳定性：企业级稳定性和可靠性

4、长期支持：10年支持周期

5、社区支持：活跃的开源社区

## 使用场景

1、服务器部署

2、数据中心应用

3、云计算平台

4、容器化环境

## 与CentOS的关系

Rocky Linux填补了CentOS停止开发后的空白，提供了相同的功能和兼容性。

## 总结

Rocky Linux 9 是一个成熟的企业级Linux发行版，为需要RHEL兼容性但不想支付商业许可费用的组织提供了理想的解决方案。它结合了开源的灵活性、企业级的稳定性和长期的支持承诺，使其成为服务器和企业环境的优秀选择。

# 什么是Linux内核

Linux 系统的核心是内核。内核控制着计算机系统上的所有硬件和软件，在必要时分配硬件，并根据需要执行软件。

1、系统内存管理。

2、应用程序管理

3、硬件设备管理

4、文件系统管理。

# 怎么修改linux登录的密码

## 1、使用 sudo 或 su 获取管理员权限

如果你当前是普通用户，并且该用户有 sudo 权限，可以使用 sudo 来修改 root 密码。

如果你已经以 root 用户登录，直接跳到第 2 步。

使用 sudo（推荐方式）

sudo passwd root

或者使用 su（如果你已经是 root 用户或有权限切换到 root）

su -

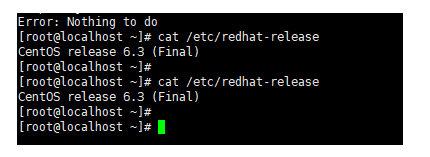
passwd root

2、

# 怎么查看linux的版本

## 方法一（使用这个）

通过命令cat /etc/[redhat](https://so.csdn.net/so/search?q=redhat&spm=1001.2101.3001.7020)-release，主要针对redhat系列，redhat,centos都可以通过此命令查看。

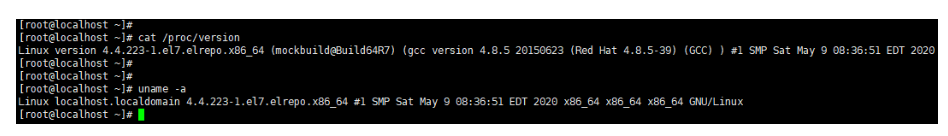


## 方法二

如下方法即可查看操作系统版本，也可查看内核版本。

cat /proc/version  #查看版本信息

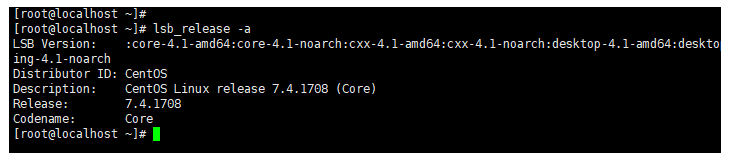
uname -a   #查看版本和内核



## 方法三

lsb\_release -a

使用命令时提示command not found，需要安装yum install redhat-lsb –y

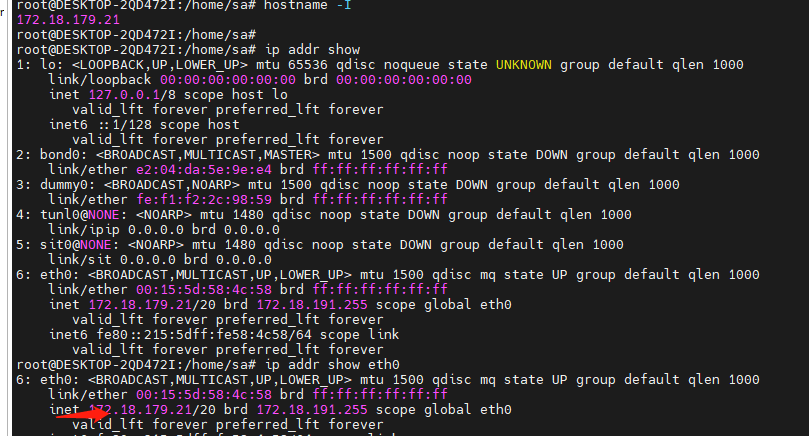


# 怎么查看IP地址（内网IP地址）

## 第一种方法（使用这个）

ip addr show

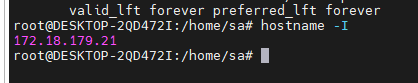
或者ifconfig



如果你不确定是哪个设备，可以查看最常用的、接入“eth0”端口的设备，也就是系统识别的第一个网络适配器。如果你只想查看“eth0”的信息，请输入**ip addr show eth0** 命令。

## 第二种方法

hostname –I



# 怎么查看Liunx的公网IP地址

使用外部服务查看公网 IP, 因为公网 IP 是分配给你的网络接口（通常是 NAT 后的公共地址），你通常可以通过访问外部服务来查看。

使用 curl 或 wget 查询公共 IP, 你可以通过向外部网站发送请求，来获取你的公网 IP 地址。这些网站会返回访问它们的客户端的公网 IP。

curl ifconfig.me

这些命令会返回你当前机器的公网 IP 地址。例如，返回：

203.0.113.1

# Linux基本组件

就像任何其他典型的操作系统一样，Linux拥有所有这些组件:内核，shell 和 GUl系统实用程序和应用程序。Linux 比其他操作系统更具优势的是每个方面都附带其他功能，所有代码都可以免费下载。

# BASH和DOS之间的基本区别

1、BASH命令区分大小写，而DOS命令则不区分。

2、在BASH下，/ character 是目录分隔符，\作为转义字符。在 DOS 下，/用作命令参数分隔符，是目录分隔符。

3、DOS 遵循命名文件中的约定，即 8 个字符的文件名后跟一个点，扩展名为 3个字符。BASH 没有遵循这样的惯例。

# Linux开机启动过程

1、主机加电自检，加载BIOS硬件信息。

2、 读取 MBR 的引导文件(GRUB、LILO).

3、引导 Linux 内核。

4、运行第一个进程 init (进程号永远为 1)。

5、进入相应的运行级别。

6、运行终端，输入用户名和密码。

# 安装git

## 方法一

### 1、更新系统包列表

sudo dnf update

### 2、安装git

sudo dnf install git

### 3、验证安装

git --version

## 方法二

### 1、安装开发工具组

sudo dnf groupinstall "Development Tools"

这会安装包括Git在内的完整开发工具套件。

### 2、验证git

git --version

which git

# Bash

## 基本概念

1、Bash 是一种 Shell（命令行解释器），是 Linux/Unix 系统的默认交互界面。

2、它是早期 **Bourne Shell（sh）** 的增强版，支持：

（1）命令执行（如 ls、cd）。

（2）脚本编程（写 .sh 文件自动化任务）。

（3）环境变量管理（如 PATH）。

（4）历史命令、自动补全等便捷功能。

## 如何判断自己是 Bash 用户

运行以下命令查看当前 Shell：

echo $SHELL

如果返回 /bin/bash，说明你是 Bash 用户。

## Bash用户的配置文件

Bash 在启动时会加载特定配置文件来初始化环境（如别名、环境变量等）

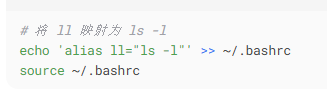


## Bash 用户常用操作

### 1、设置环境变量

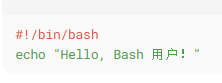


### 2、定义命令别名（快捷方式）



### 3、编写 Bash 脚本

创建一个文件（如 hello.sh）



运行脚本：

chmod +x hello.sh # 添加执行权限

./hello.sh # 执行

## 总结

**1、Bash 用户** = 使用 Bash 作为命令行解释器的用户。

2、通过 ~/.bashrc 和 ~/.bash\_profile 自定义环境。

3、Bash 是 Linux 生态的核心工具，掌握它能大幅提升效率

# 如何切换到root用户

可使用su命令切换到root，输入su命令并回车，此时会要求输入root用户的密码，如果密码正确即可切换成功。

# 文件目录结构

## /bin

bin 是 Binaries (二进制文件) 的缩写, 这个目录存放着最经常使用的命令。

## /boot

这里存放的是启动 Linux 时使用的一些核心文件，包括一些连接文件以及镜像文件。

## /dev

dev 是 Device(设备) 的缩写, 该目录下存放的是 Linux 的外部设备，在 Linux 中访问设备的方式和访问文件的方式是相同的。

## /etc

etc 是 Etcetera(等等) 的缩写,这个目录用来存放所有的系统管理所需要的配置文件和子目录。

## /home

用户的主目录，在 Linux 中，每个用户都有一个自己的目录，一般该目录名是以用户的账号命名的，如上图中的 alice、bob 和 eve。

## /lib

lib 是 Library(库) 的缩写，这个目录里存放着系统最基本的动态链接共享库，其作用类似于 Windows 里的 DLL 文件。几乎所有的应用程序都需要用到这些共享库。

## /lost+found

这个目录一般情况下是空的，当系统非法关机后，这里就存放了一些文件。

## usr

在 Linux 系统中，/usr 目录通常用于存放用户相关的应用程序、库文件、文档、资源文件以及其他共享数据。

1、/usr/bin：这个目录包含了大部分用户级的可执行程序。当我们安装新的应用程序时，它们的可执行文件通常会被放置在这个目录下。

2、/usr/sbin：这个目录包含了系统管理员使用的系统级可执行程序。这些程序需要具有 root 用户权限才能执行，一般用于系统管理和配置。

3、/usr/lib 和 /usr/lib64：这两个目录包含了共享的库文件（或者称为动态链接库），这些库文件由许多应用程序共享使用。/usr/lib 是用于 32 位系统，而 /usr/lib64 是用于 64 位系统。

4、/usr/include：此目录包含了头文件，它们用于编译应用程序时引用库函数和结构的定义。

5、/usr/share：这个目录用于存放特定应用程序共享的数据文件，如图标、桌面文件、配置文件、帮助文档等。

6、/usr/local：这个目录用于存放手动安装的本地软件，通常是用户自己安装的软件，而不是通过发行版的包管理器安装的软件。

7、/usr/src：这个目录经常用于存放系统内核源代码，以及其他一些软件的源代码。

8、/usr/man：此目录包含了系统手册页（man page）文件，这些文件提供了对各种命令、函数和配置文件的详细说明。

9、/usr/local/bin 和 /usr/local/sbin：这两个目录用于存放本地安装的可执行程序，类似于 /usr/bin 和 /usr/sbin，但是这些程序不是通过发行版的包管理器安装的，而是由用户自行安装的。

此外，/usr 目录下可能还有其他子目录和文件，具体内容可能因系统配置和安装的软件而有所不同。

需要注意的是，/usr 目录一般用于存放系统级的共享数据和应用程序文件，它们与操作系统的正常运行和用户环境息息相关。因此，在使用或更改这些文件时需要谨慎，最好遵循系统管理员的指导或者参考相关文档。

# 文件颜色

## 基本概念

在 Linux 终端中，文件或目录的颜色是有特殊含义的。通常情况下，这些颜色用于提示用户该文件或目录的类型、权限等信息。

## 红色的文件

表示该文件为压缩文件或归档文件，或者该文件的权限设置了不允许普通用户访问。例如，\*.gz、\*.tar.gz、\*.zip 等压缩文件和归档文件，以及 /etc/shadow 这种只有管理员才有权限访问的文件，在终端中都会以红色字体显示。

## 蓝色的文件

表示该文件为目录文件（文件夹），例如，/usr/bin/ 这个目录，在终端中会以蓝色字体显示。

## 绿色的文件

表示该文件是可执行文件或者说是有执行权限的文件，如二进制文件、脚本文件等。

## 黄色的文件

表示该文件是设备文件，即硬件设备的文件接口，如 /dev/sda

## 灰色的文件

表示该文件是系统默认定义的文件类型或者是未知文件类型。

# 压缩文件和归档文件的区别

# .tar文件

## 基本概念

.tar 文件是在 Linux 和类 Unix 系统中常见的归档文件格式，它通常用于将多个文件和目录组合成一个单个的文件。

# .gz文件

## 基本概念

.gz 文件是一种常见的压缩文件格式，在 Linux 系统中被广泛使用。.gz 文件是经过 GNU Zip 压缩的文件，通常用于将一个或多个文件进行压缩以减小文件的大小。它使用 DEFLATE 算法进行压缩，可以显著减少文件的体积，从而方便传输和存储。

## 主要的操作

1、解压缩文件：使用 gzip -d <file.gz> 或 gunzip <file.gz> 命令解压缩指定的 .gz 文件。该命令会还原压缩前的原始文件，并将其保存到当前目录下。

# .tgz文件

1、.tgz是一种文件扩展名，代表一个经过压缩的归档文件。它是由两个不同的扩展名组成：.tar和.gz。

2、.tar：表示该文件使用tar（tape archive）工具创建。tar是一种用于将多个文件和目录打包成单个文件的工具。它不进行压缩，只是将文件按照一定的格式打包到一起。

3、.gz：表示该文件经过gzip（GNU zip）压缩。gzip是一种常用的文件压缩工具，它可以有效地减小文件的大小，节省存储空间。

当这两个扩展名结合在一起时，表示一个.tar文件通过gzip压缩生成了一个.tgz文件。也就是说，.tgz文件实际上是.tar文件的压缩版本。使用.tgz文件可以方便地将多个文件和目录组合成一个文件，并且通过压缩减小文件的大小。对于需要在不同系统之间传输或存储大量文件的情况，.tgz文件是非常常见和实用的文件格式。在Linux和类Unix系统中，通常使用tar命令和gzip命令来创建和解压缩.tgz文件。

# .rpm文件

## 基本概念

.rpm 是一种常见的软件包文件格式，用于在 Linux 系统上方便地安装、升级、卸载软件。.rpm 文件是 RPM（Red Hat Package Manager）软件包的扩展名。它是一种二进制软件包，通常包含一个或多个预编译的可执行文件、库文件、配置文件、文档等软件组件。.rpm 文件采用标准的 CPIO（Copy In and Out）归档格式进行封装，同时通过一些特定的元数据信息描述软件包的内容和依赖关系。

需要注意的是，.rpm 文件通常用于基于 RPM 包管理系统的 Linux 发行版（如 Fedora、CentOS、openSUSE 等），在其他发行版中可能使用不同的包管理系统和包格式。

## 主要操作

1、安装软件包：使用 rpm -i <package.rpm> 命令安装指定的软件包。该命令会解压 .rpm 文件并将其中的文件拷贝到系统的相应位置。

2、升级软件包：使用 rpm -U <package.rpm> 命令将一个已安装的软件包升级到指定的 .rpm 文件对应的版本。

3、卸载软件包：使用 rpm -e <package> 命令卸载指定的软件包。该命令会将软件包相关的文件和配置进行清理。

4、查询软件包信息：使用 rpm -q <package> 命令查询已安装的软件包信息，包括软件包的名称、版本等。可以使用不同的选项来获取不同级别的信息。

5、查询文件所属的软件包：使用 rpm -qf <file> 命令查询指定文件所属的软件包。

# ./

在 Linux 中，./ 是一个特殊的路径表示法，它用于执行当前目录下的可执行文件。

1、点号（.）：代表当前目录。在文件系统中，点号通常用来表示当前目录的路径。

2、斜杠（/）：表示目录路径的分隔符。在 Linux 中，斜杠被用作路径分隔符来指示目录的层次结构。

当你使用 ./ 前缀来执行一个命令或者程序时，它告诉 Shell 在当前目录中查找可执行文件并执行。这种方式是为了避免与系统的环境变量冲突或误解。

# #!/bin/sh

## 基本说明

#!/bin/sh 是一个在类 Unix 系统中使用的特殊注释格式，这个注释告诉操作系统执行这个脚本时要使用哪个解释器来解释脚本的内容。

# 是注释符号，表示接下来的内容是注释

! 后面紧跟着一个路径或命令，用于指定解释器的类型

#!/bin/sh 表示要使用 /bin/sh 解释器来执行这个脚本。

因此，#!/bin/sh 告诉操作系统在执行这个脚本时要使用 /bin/sh 解释器。这个注释通常作为脚本文件的第一行，并且后面紧跟着脚本的实际内容。

# linux命令

## ss

### 基本概念

ss (Socket Statistics) 是 Linux 系统中用于查看套接字统计信息的现代工具，它比传统的 netstat 命令更快、更高效，支持更丰富的过滤功能。

### 语法

ss [options] [ FILTER ]

#### 常用选项

-t 显示TCP套接字

-u 显示UDP套接字

-l 仅显示监听状态的套接字

-a 显示所有套接字（监听和非监听）

-n 不解析服务名称（显示端口号而非服务名）

-p 显示使用套接字的进程

-s 显示套接字使用统计

-4 仅显示 IPv4 套接字

-6 仅显示 IPv6 套接字

-o 显示计时器信息

-e 显示详细的套接字信息

### 示例

#### 1、查看所有连接

ss -a

#### 2、查看所有 TCP 连接

ss -t -a

#### 3、查看所有UDP连接

ss -u -a

#### 4、查看所有监听端口

ss -tuln

### 总结

ss 命令是 Linux 网络诊断的强大工具，比 netstat 更高效，支持更复杂的过滤条件。掌握 ss 命令可以帮助你快速诊断网络连接问题，监控网络活动。

## ls命令

### 基本概念

ls 命令用于列出当前目录下的文件和子目录。它可以根据不同的选项提供各种不同的输出格式和信息。

### 用法

ls [选项] [文件/目录]

### 常用选项

-a：显示所有文件，包括隐藏文件（以.开头的文件）

-r：逆序显示文件列表

-t：按照修改时间排序，最近修改的文件先显示。

-l：以长格式输出文件列表，包括文件类型和权限等详细信息。

-h：与 -l 选项配合使用，以人性化的方式显示文件大小。

-S ：按照文件大小排序，文件大小越大的文件先显示

## ll命令

### 基本概念

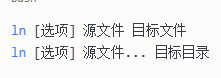
ll 命令是 ls -l 的简写，它也是用于列出指定目录下的文件和子目录，并以长格式输出文件列表，包括文件类型、权限、所有者、所属组、大小、修改时间等详细信息。

## ln命令

### 基本概念

ln 命令是 Linux 系统中常用的一个命令，它用于创建链接文件。Linux 中的链接文件可以分为两种类型：硬链接和符号链接(软链接)。

### 基本语法



#### 常用选项

-s : 创建软链接（符号链接）

-f : 强制创建，覆盖已存在的目标文件

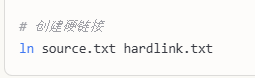
-i : 交互模式，覆盖前询问

-v : 详细模式，显示创建过程

-r : 创建相对于链接位置的符号链接

### 硬链接 vs 软链接

#### 硬链接



##### 特点

1、与原文件共享同一个inode

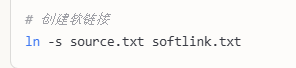
2、删除原文件不影响硬链接

3、不能跨文件系统

4、不能链接到目录

5、文件大小和原文件相同

#### 软链接



##### 特点

1、有独立的inode

2、删除原文件会导致软链接失效

3、可以跨文件系统

4、可以链接到目录

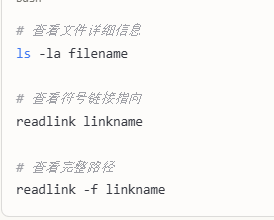
5、文件大小很小，只存储路径信息

### 示例

#### 1、基本软链接创建



#### 2、检查链接状态



# 查看符号链接指向

readlink linkname

# 查看完整路径

readlink -f linkname

#### 3、配置文件管理



### 总结

ln命令是Linux系统中非常重要的文件管理工具。硬链接适合在同一文件系统内创建文件的多个入口点，而软链接更加灵活，可以跨文件系统并链接目录。理解两者的区别和适用场景，能够帮助你更有效地管理文件系统。

## ifconfig命令

### 基本概念

ifconfig（接口配置）命令用于配置和管理网络接口。它提供了一种查看、设置和修改网络接口配置的方法。

### 作用

#### 1、查看网络接口信息

使用 ifconfig 命令不带参数，可以列出当前系统上所有的网络接口的配置信息。这些信息包括接口名称（如 eth0、wlan0）、接口的 MAC 地址、IPv4 和 IPv6 地址、网络子网掩码、广播地址等。

#### 2、启用或禁用网络接口

通过 ifconfig 命令，你可以启用或禁用指定的网络接口。例如，使用 ifconfig eth0 up 命令可以激活名为 eth0 的接口，而使用 ifconfig eth0 down 命令则可以停用它。

#### 3、配置IP地址

#### 4、配置网络子网掩码

#### 5、配置广播地址

#### 6、配置网络接口的 MTU

#### 7、显示统计信息

## dnf

### 基本概念

**DNF**（Dandified YUM）是现代Linux发行版的包管理器，在Rocky Linux 9中替代了YUM。它基于Python编写，提供了更好的性能、依赖解析和用户体验。

### 语法

dnf [options] command [arguments]

### 示例

#### 1、安装软件包



#### 2、卸载软件包



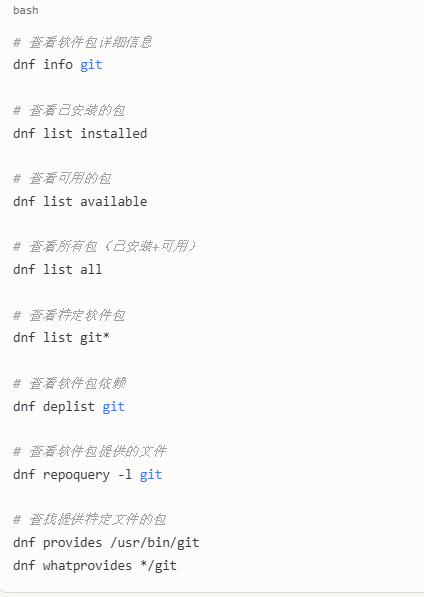
#### 3、升级软件包



#### 4、搜索软件包



#### 5、查询软件包信息



### 配置文件

主配置文件：/etc/dnf/dnf.conf



仓库配置目录：/etc/yum.repos.d/

### 总结

DNF是一个功能强大、现代化的包管理器，提供了：

1、智能依赖解析：自动处理复杂依赖关系

2、事务管理：支持回滚和重做操作

3、模块化支持：灵活的软件版本管理

4、丰富的查询功能：全面的包信息查询

5、插件扩展：通过插件扩展功能

## apt

### 基本概念

Ubuntu使用的包管理器是**apt**（Advanced Package Tool）

### 命令

#### apt update

更新软件包列表

#### apt install 包名

安装软件包

#### apt remove 包名

删除软件包

#### apt upgrade

升级已安装的软件包

## yum

### 基本概念

基于RPM的软件包管理器，**yum命令**是在Fedora和RedHat以及SUSE中基于rpm的软件包管理器，它可以使系统管理人员交互和自动化地更新与管理RPM软件包，能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软体包，无须繁琐地一次次下载、安装。yum提供了查找、安装、删除某一个、一组甚至全部软件包的命令，而且命令简洁而又好记。

### 语法

yum [options] [command] [package ...]

options：可选，选项包括

#### 参数

##### -h

（帮助）

##### -y

（当安装过程提示选择全部为 "yes"），

##### -q

（不显示安装的过程）等等。

##### command

要进行的操作

##### package

安装的包名

### 命令

#### install

安装rpm软件包

#### update

更新rpm软件包

#### check-update

检查是否有可用的更新rpm软件包

#### remove

删除指定的rpm软件包；

#### list

显示软件包的信息；

#### search

检查软件包的信息；

#### info

显示指定的rpm软件包的描述信息和概要信息；

#### clean

清理yum过期的缓存；

#### shell

进入yum的shell提示符；

#### resolvedep

显示rpm软件包的依赖关系；

#### localinstall

安装本地的rpm软件包；

#### localupdate

显示本地rpm软件包进行更新；

#### deplist

显示rpm软件包的所有依赖关系。

### 常用实例

#### 1、列出所有可更新的软件清单命令

yum check-update

#### 2、更新所有软件命令

yum update

#### 3、仅安装指定的软件命令

yum install <package\_name>

#### 4、仅更新指定的软件命令

yum update <package\_name>

#### 5、列出所有可安裝的软件清单命令

yum list

#### 6、删除软件包命令

yum remove <package\_name>

### 配置阿里云Centos镜像

https://developer.aliyun.com/mirror/centos

## rpm

### 基本概念

rpm命令来自于英文词组”RedHat Package Manager“的缩写，中文译为红帽软件包管理器，其功能是用于在Linux系统下对软件包进行安装、卸载、查询、验证、升级等工作。

RPM 是 Linux 中最早引入的软件包管理工具，它以 .rpm 后缀的软件包文件为单位进行操作。RPM 可以完成软件包的安装、升级、查询和卸载等操作。

1、独立性：RPM 软件包包含了应用程序及其依赖的所有文件和配置信息，可以在没有网络连接的情况下进行安装和管理。

2、包管理：RPM 可以管理软件包的安装、升级和卸载，通过验证软件包完整性与依赖关系，确保系统的稳定性。

3、命令行工具：RPM 提供了一系列的命令行工具，例如 rpm、rpm2cpio 和 rpmbuild 等，用于管理软件包。

### 适用的操作系统

#### Red Hat 系列

Red Hat Enterprise Linux (RHEL)

CentOS

Rocky Linux

AlmaLinux

Oracle Linux

#### SUSE 系列

openSUSE

SUSE Linux Enterprise Server (SLES)

#### Fedora 系列

Fedora（Red Hat 的社区版本）

#### 其他发行版

Mageia

PCLinuxOS

Mandriva Linux（已停止维护）

### **语法**

rpm [参数] 软件包

#### **常用参数**



##### -i

安装软件包

##### -e

卸载软件包

##### -v

显示命令执行过程

##### -h

安装软件包时列出标记

##### -q

查询软件包

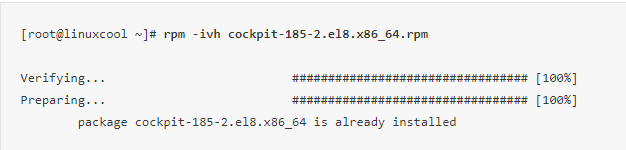
##### -a

查询所有的软件包

### 实例

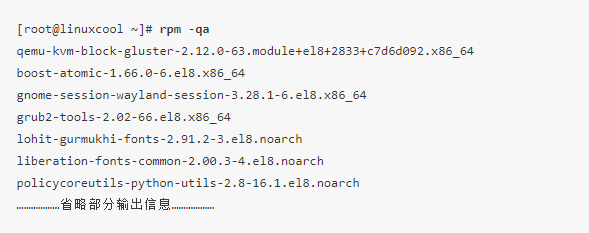
#### 1、正常安装软件包

rpm -ivh cockpit-185-2.el8.x86\_64.rpm



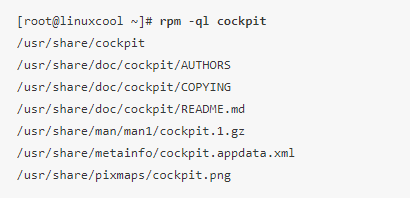
#### **2、显示系统已安装过的全部RPM软件包**

rpm -qa



#### 3、查询某个软件的安装路径

rpm -ql cockpit



## rpm和yum区别和联系

### 区别

#### 1、RPM

RPM是一种软件包管理系统，提供了对RPM格式软件包的管理功能。它通过执行命令来安装、查询、更新和卸载软件包。RPM直接操作软件包文件，无需与远程软件源进行交互。它可以将软件包的详细信息和安装位置记录到系统的数据库中。

#### 2、Yum

Yum是基于RPM的高级软件包管理器。它使用RPM格式软件包，但与RPM不同，Yum可以自动解决依赖关系，并从预配置的软件源（如官方镜像站点）下载和安装软件包。Yum还提供了更便捷的命令集合，使得软件包的管理更加简单。

### 联系

1、Yum基于RPM，使用RPM格式的软件包进行安装和管理。Yum利用RPM作为底层，实现了自动解决依赖关系、版本控制等功能。它使用RPM软件包来执行安装、查询、更新和卸载操作。

2、Yum可以与RPM一起使用，通过Yum可以方便地从预配置的软件源中获取RPM包，进行安装和管理。Yum还可以通过命令行界面或配置文件来管理软件包。

### 总结

总结来说，RPM是一种基本的软件包管理工具，用于处理RPM格式的软件包，提供了基本的安装、查询、更新和卸载功能。而Yum是基于RPM的高级软件包管理工具，提供了自动解决依赖关系、从预配置的软件源下载和安装软件包等功能，使得软件包的管理更加简单和方便。在实际使用中，可以根据需要选择使用RPM直接操作软件包，或者使用Yum来管理软件包。

## dnf、yum、rpm使用分析

### 使用统计

在现代Linux系统中，**YUM/DNF使用得更多**，RPM主要用于特殊情况。

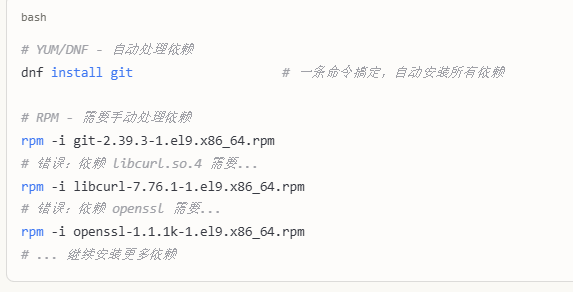
日常使用比例（大概估算）

**YUM/DNF**：约80-90%的日常软件管理

**RPM**：约10-20%的特殊情况使用

### 为什么YUM/DNF使用更多

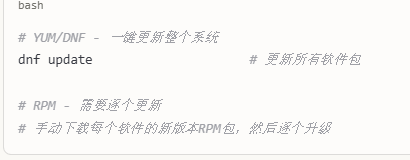
#### 1、自动依赖解决



#### 2、软件仓库集成



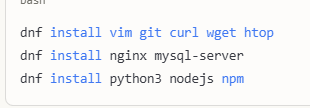
#### 3、系统更新便利性



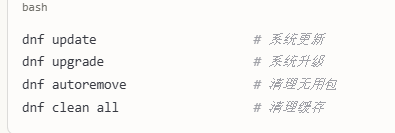
### 具体使用场景分析

#### YUM/DNF 主要使用场景（80-90%）

##### 1、日常软件安装



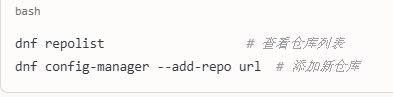
##### 2、系统维护



##### 3、软件搜索和信息查询

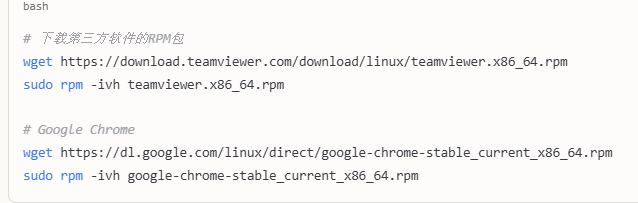


##### 4、仓库管理

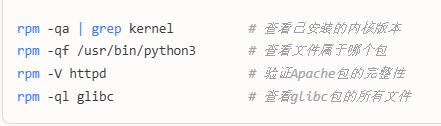


#### RPM 主要使用场景（10-20%）

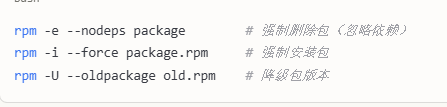
##### 1、安装第三方RPM包



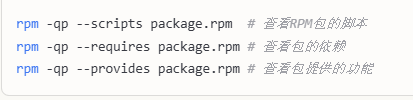
##### 2、系统诊断和查询



##### 3、低级别包管理



##### 4、包开发和测试



### 使用原则



### 命令对比列表



### 总结

YUM/DNF使用更频繁的原因：

1、用户友好：自动处理复杂的依赖关系

2、功能全面：集成了仓库管理、更新、搜索等功能

3、安全可靠：自动验证和完整性检查

4、现代化：符合现代Linux系统管理习惯

RPM仍然重要的原因：

1、底层基础：YUM/DNF底层仍使用RPM

2、特殊需求：某些高级操作只有RPM支持

3、系统诊断：RPM查询功能更直接

4、第三方软件：很多第三方软件仍提供RPM包

建议：日常使用DNF，需要时再使用RPM进行底层操作。

## wget

### 基本概念

Linux系统中的wget是一个下载文件的工具，它用在命令行下。对于Linux用户是必不可少的工具，我们经常要下载一些软件或从远程服务器恢复备份到本地服务器。wget支持HTTP，HTTPS和FTP协议，可以使用HTTP代理。所谓的自动下载是指，wget可以在用户退出系统的之后在后台执行。这意味这你可以登录系统，启动一个wget下载任务，然后退出系统，wget将在后台执行直到任务完成，相对于其它大部分浏览器在下载大量数据时需要用户一直的参与，这省去了极大的麻烦。

wget 可以跟踪HTML页面上的链接依次下载来创建远程服务器的本地版本，完全重建原始站点的目录结构。这又常被称作”递归下载”。在递归下载的时候，wget 遵循Robot Exclusion标准(/robots.txt). wget可以在下载的同时，将链接转换成指向本地文件，以方便离线浏览。

wget 非常稳定，它在带宽很窄的情况下和不稳定网络中有很强的适应性.如果是由于网络的原因下载失败，wget会不断的尝试，直到整个文件下载完毕。如果是服务器打断下载过程，它会再次联到服务器上从停止的地方继续下载。这对从那些限定了链接时间的服务器上下载大文件非常有用。

### 语法

wget [参数] [URL地址]

### 功能

用于从网络上下载资源，没有指定目录，下载资源回默认为当前目录。wget虽然功能强大，但是使用起来还是比较简单。

1、支持断点下传功能。

2、同时支持FTP和HTTP下载方式。

3、支持代理服务器。

4、设置方便简单

5、程序小，完全免费。

### 实例

#### 1、使用wget下载单个文件

wget http://www.minjieren.com/wordpress-3.1-zh\_CN.zip

说明：从网络下载一个文件并保存在当前目录，在下载的过程中会显示进度条，包含（下载完成百分比，已经下载的字节，当前下载速度，剩余下载时间）。

2、

## curl

### 基本概念

curl是一个功能强大的命令行工具，用于在Linux系统中与服务器进行数据交互。它支持多种协议，如HTTP、HTTPS、FTP等，并提供了各种选项和参数，可以进行文件下载、文件上传、发送请求、测试接口等一系列操作。

### 语法

curl [选项] [URL]

常用选项：

-X/--request：设置HTTP请求方法，如GET、POST、PUT等。

-H/--header：设置HTTP请求头信息。

-d/--data：指定要发送的数据，通常用于POST请求。

-i/--include：在输出中包含HTTP响应头信息。

-o/--output：将响应保存到指定文件。

-L/--location：跟随重定向。

-u/--user：设置用户名和密码。

-s/--silent：静默模式，不显示进度和错误信息。

### 实例

#### 1、发送HTTP请求

curl https://example.com

这个命令会发送一个GET请求到https://example.com，并输出服务器返回的内容。

#### 2、下载文件

curl -O https://example.com/file.zip

这个命令会下载https://example.com/file.zip文件，并保存到当前目录。

#### 3、发送POST请求

curl -X POST -d "name=John&age=30" https://example.com/api

这个命令会以POST方式发送数据name=John&age=30到https://example.com/api。

## telnet

### 基本概念

telnet命令通常用来远程登录。telnet程序是基于TELNET协议的远程登录客户端程序。Telnet协议是TCP/IP协议族中的一员，是Internet远程登陆服务的标准协议和主要方式。它为用户提供了在本地计算机上完成远程主机工作的 能力。在终端使用者的电脑上使用telnet程序，用它连接到服务器。终端使用者可以在telnet程序中输入命令，这些命令会在服务器上运行，就像直接在服务器的控制台上输入一样。可以在本地就能控制服务器。要开始一个 telnet会话，必须输入用户名和密码来登录服务器。Telnet是常用的远程控制Web服务器的方法。但是telnet因为采用明文传送报文，安全性不好，很多Linux服务器都不开放telnet服务，而改用更安全的ssh方式了。但仍然有很多别的系统可能采用了telnet方式来提供远程登录，因此弄清楚telnet客户端的使用方式仍是很有必要的。

telnet命令还可做别的用途，比如确定远程服务的状态、确定远程服务器的某个端口是否能访问。

### 语法

telnet[参数][主机]

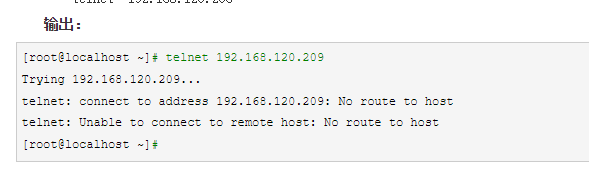
### 功能

执行telnet指令开启终端机阶段作业，并登入远端主机。

### 实例

#### 1、访问远程服务器

telnet 192.168.120.206



说明：

处理这种情况方法：

（1）确认ip地址是否正确。

（2）确认ip地址对应的主机是否已经开机。

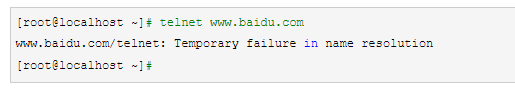
（3）如果主机已经启动，确认路由设置是否设置正确。（使用route命令查看）

（4）如果主机已经启动，确认主机上是否开启了telnet服务（使用netstat命令查看，TCP的23端口是否有LISTEN状态的行）

（5）如果主机已经启动telnet服务，确认防火墙是否放开了23端口的访问？（使用iptables-save查看）

#### 2、解析域名

telnet www.baidu.com



说明：

处理这种情况方法：

（1）确认域名是否正确

（2）确认本机的域名解析有关的设置是否正确（/etc/resolv.conf中nameserver的设置是否正确，如果没有，可以使用nameserver 8.8.8.8）

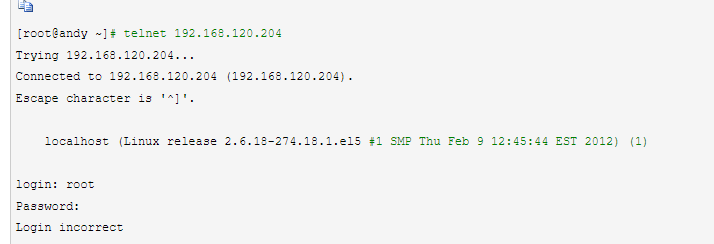
（3）确认防火墙是否放开了UDP53端口的访问（DNS使用UDP协议，端口53，使用iptables-save查看）

#### 3、启动telnet服务

service xinetd restart

#### 4、正常telnet

telnet 192.168.120.204

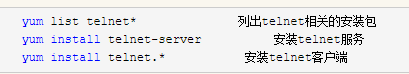


说明：一般情况下不允许root从远程登录，可以先用普通账号登录，然后再用su -切到root用户。

### 检测远程端口是否打开

### 安装telnet服务

centos、ubuntu安装telnet命令的方法。



执行下面两个命令即可。

yum install telnet-server 安装telnet服务

yum install telnet.\* 安装telnet客户端

## netstat

### 基本概念

netstat 是一个用于显示网络连接、路由表和网络接口等网络系统信息的命令行工具。在 Linux 系统中，netstat 命令非常常用，可以帮助我们查看当前系统的网络状态和连接信息。

### 基本语法

netstat [options]

#### 常用选项

-t: 显示 TCP 连接信息

-u: 显示 UDP 连接信息

-n: 显示 IP 地址和端口号，不进行域名解析

-a: 显示所有连接（包括监听和非监听状态）

-r: 显示路由表信息

-s: 显示网络统计信息

-p: 显示建立相关连接的程序名

-l: 仅显示监听状态的连接

-c: 持续输出，每隔一秒刷新一次

### 示例

#### 1、显示所有网络连接

netstat -a

#### 2、显示网络接口统计信息

netstat -s

#### 3、显示路由表信息

netstat -r

#### 4、查找特定端口的网络连接信息

netstat -apn | grep 端口号

## whereis

### 基本概念

whereis 是一个用于查找可执行文件、源代码文件和帮助文档的 Linux 命令。它的作用是定位特定程序或文件的位置。whereis 命令会搜索一组特定目录，这些目录通常包含系统命令和文件。

### 功能

#### 1、查找可执行文件

whereis 命令可以用于查找系统中安装的可执行文件的位置。这些可执行文件通常是系统命令，如 ls、cp、mv 等。通过执行 whereis 命令并指定要查找的命令名称，它会返回该命令的二进制文件路径。

#### 2、查找源代码文件

在一些情况下，你可能需要查找某个程序的源代码文件。whereis 命令可以帮助你查找已安装软件的源代码。它会在一些默认的源代码存储路径中进行搜索，并返回源代码文件的路径。

#### 3、查找帮助文档

大多数 Linux 系统提供了各种命令的帮助文档。使用 whereis 命令可以快速查找某个命令的帮助文档的位置。它会搜索存储帮助文档的默认目录，并返回相应的路径。需要注意的是，whereis 命令只搜索特定的系统目录，一般包括 /bin、/sbin、/usr/bin、/usr/sbin、/usr/local/bin、/usr/local/sbin 等。它不会搜索用户主目录下的文件或其他自定义目录。

### 示例

#### 1、查找命令的二进制文件

## whereis与which命令的区别

### 1、基本区别概述

#### which命令

用途：在PATH环境变量中查找可执行文件

返回：实际会执行的命令路径

搜索范围：仅搜索PATH中的目录

#### whereis命令

1、用途：在预定义目录中查找二进制文件、源码和手册页

2、返回：所有相关文件的位置

3、搜索范围：系统预定义的标准目录

## source命令

### source /etc/profile

用于在当前 shell 环境中加载并执行 /etc/profile文件的内容。/etc/profile 是一个系统级别的配置文件，用于设置全局的环境变量和默认的系统行为。在 Unix/Linux 系统中，这个文件通常包含一些系统范围的环境变量设置、路径配置、别名定义等。

当你运行 source /etc/profile 命令时，它会读取并执行 /etc/profile 文件中的所有命令，将其中的环境变量和其他设置应用到当前的 shell 会话中。这样做的目的是确保你在当前 shell 中能够访问和使用 /etc/profile 文件中定义的环境变量和其他配置。

## ps命令

### 基本概念

要对进程进行监测和控制,首先必须要了解当前进程的情况,也就是需要查看当前进程,而ps命令就是最基本同时也是非常强大的进程查看命令，使用该命令可以确定有哪些进程正在运行和运行的状态、进程是否结束、进程有没有僵尸、哪些进程占用了过多的资源等等。总之大部分信息都是可以通过执行该命令得到的，**ps**命令最常用的还是用于监控后台进程的工作情况，因为后台进程是不和屏幕键盘这些标准输入/输出设备进行通信的，所以如果需要检测其情况,便可以使用ps命令了。

### 说明

ps是**显示瞬间进程的状态，并不动态连续**，如果想对进程进行实时监控应该用top命令。

### 参数

#### -A

显示所有的进程，与 -e 具有同样的效用。

#### -e

显示系统中的所有进程。

#### -a

显示现行终端机下的所有进程，包括其他用户的进程。

#### -u

以用户为主的进程状态。

-au 显示较详细的进程信息

#### x

通常与 a 这个参数一起使用，可列出较完整信息。

### 输出格式规划

#### l

较长、较详细的将该 PID 的信息列出

#### j

工作的格式 (jobs format)

#### -f

做一个更为完整的输出

特别说明：由于ps能够支持的系统类型相当的多，所以他的参数多的离谱。

而且有没有加上 - 差很多，详细的用法应该要参考 man ps。

### 示例

#### 单独输出ps

ps

它会默认显示与当前终端相关的进程信息

下面是一些常见的ps参数和其含义：

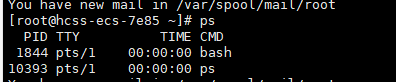
PID：进程的ID，唯一标识一个进程。

TTY：与进程关联的终端设备。

TIME：进程已经使用的CPU时间。

CMD：启动进程的命令行。

在单独输出ps指令时，默认情况下，它只会显示与当前终端关联的进程信息。这包括当前终端窗口中正在运行的进程以及可能由该终端窗口直接启动的其他进程。



#### 1、将目前属于您自己这次登入的 PID 与相关信息列出来

ps –l

#### 2、列出目前所有的正在内存当中的程序（进程）

ps aux

#### 3、以示例一的显示内容，显示出所有的程序

ps –lA

#### 4、显示jboss所在的进程（重点）

ps –aef|grep jboss

root 28311 27659 0 22:31 pts/0 00:00:00 grep --color=auto java

这是一个用于查找正在运行的Java进程的命令输出。

root：表示当前运行该命令的用户是root用户，即具有管理员权限的用户。

28311：是正在运行的Java进程的进程ID（PID）。

27659：是父进程的PID（通常是shell进程）。

0：表示该进程的CPU使用率。

22:31：表示进程启动的时间（格式为时:分）。

pts/0：表示该进程的终端设备，pts/0表示一个伪终端。

00:00:00：表示进程运行的持续时间，格式为时:分:秒。

grep --color=auto java：是用于查找包含关键字java进程的grep命令。

综合起来，这条命令的输出表示正在寻找并列出所有包含关键字java的进程。其中，root用户通过grep命令查找包含关键字java的进程，并且由于grep本身也包含了java关键字，所以在输出中也会列出grep进程本身。

## grep命令

### 基本概念

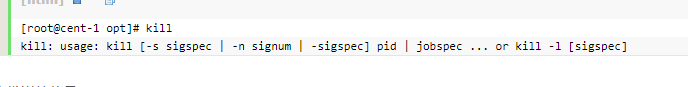
## kill命令

### 基本概念

kill命令是Linux下面的一个终止进程的命令。

### 语法

在Shell下面打kill可以看到kill的语法如下：



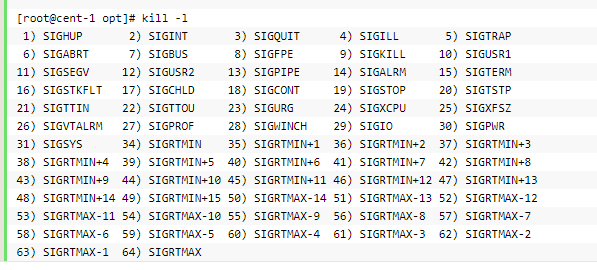
### 参数说明

-s 指定发送的信号

pid 终止的进程编号

sigspec 表示信号

通过kill –l可以看到所有的信号变量



kill pid和kill -s 15 pid含义一样，表示发送一个SIGTERM的信号给对应的程序，程序收到该信号后，将会发生以下事情：

1、程序立刻停止

2、程序释放相应资源后停止

3、程序可能仍然继续运行

大部分程序在接收到SIGTERM信号后，会先释放自己的资源，然后再停止。但也有一些程序在收到信号后，做一些其他事情，并且这些事情是可以配置的。也就是说，SIGTERM多半是会被阻塞，忽略的。

kill -9 pid

kill -9 pid等于kill -s 9 pid，表示强制尽快终止一个进程，多半admin会用这个命令。

说明：kill一个进程需要root权限去执行，一般用户无法执行这个命令。

## top命令

### 基本概念

top命令用于经常用来监控linux的系统状况，是常用的性能分析工具，能够实时显示系统中各个进程的资源占用情况。

### 语法

top [-] [d] [p] [q] [c] [C] [S] [s] [n]

### 参数说明

-d number: 指定每两次屏幕信息刷新之间的时间间隔。当然用户可以使用s交互命令来改变之。

-b：以批次的方式执行top

-p: 通过指定监控进程ID来仅仅监控某个进程的状态

top命令显示的页面还可以输入以下按键执行相应的功能（注意大小写区分的）

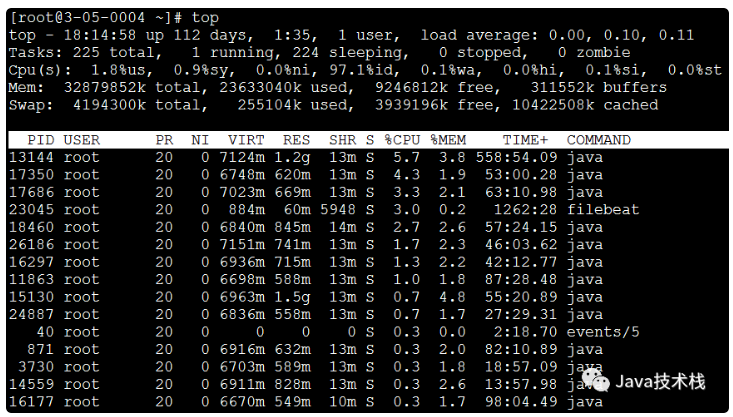


### 示例

#### 1、显示进程信息

top

相当于Windows任务管理器



可以看到，输出结果分两部分，前5行是总览，下面是具体的进程资源占用情况

下面逐行看一下

第1行

top - 18:14:58 up 112 days, 1:35, 1 user, load average: 0.00, 0.10, 0.11

依次表示：当前时间、系统已经运行的时间、当前登录的用户数、系统在过去的1分钟，5分钟，15分钟的负载

PS：从这一行我们可以知道以下信息：

当前时间是18:14:58

系统运行了112天1小时35分钟

当前有1个用户登录

在过去1分钟，5分钟，15分钟的负载分别是0.00, 0.10, 0.11

负载超过1，则表示超负荷

**第2行**

Tasks: 225 total,  1 running, 224 sleeping,  0 stopped,  0 zombie

[进程信息](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[total　　　　进程总数](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[running　　 运行中的进程数](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[sleeping　　睡眠中的进程数](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[stopped　　停止的进程数](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[zombie　　 僵尸进程数](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

PS：从这一行我们可以知道，当前总共225个进程

[**第3行**](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[Cpu(s): 1.8%us, 0.9%sy, 0.0%ni, 97.1%id, 0.1%wa, 0.0%hi, 0.1%si, 0.0%st](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[CPU使用情况](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[us ：用户进程占用CPU百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[sy ：内核进程占用CPU百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[ni ：改变过优先级的进程占用CPU百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

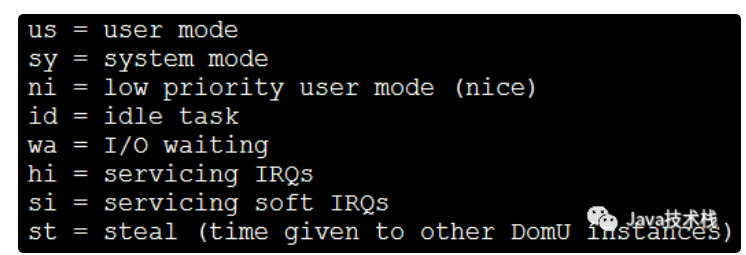
[id ：空闲CPU百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[wa ：IO等待的进程占用CPU百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[hi ：硬中断占用CPU的百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

[si ：软中断占用CPU的百分比](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247529043&idx=2&sn=ccbb24c037bbe434e86779c4206f0040&chksm=eb50f765dc277e7328851c2d884e557c362c8cbfdfea922fd49b2cbacffc4ab10034c784b82c&scene=21#wechat_redirect)

st ：



[**第4行**](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

[Mem: 32879852k total, 23633040k used, 9246812k free,  311552k buffers](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

[物理内存使用情况](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

* [total　　总的内存大小](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)
* [used　　已使用](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)
* [free　　未使用](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

[buffers　　内核缓冲区](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

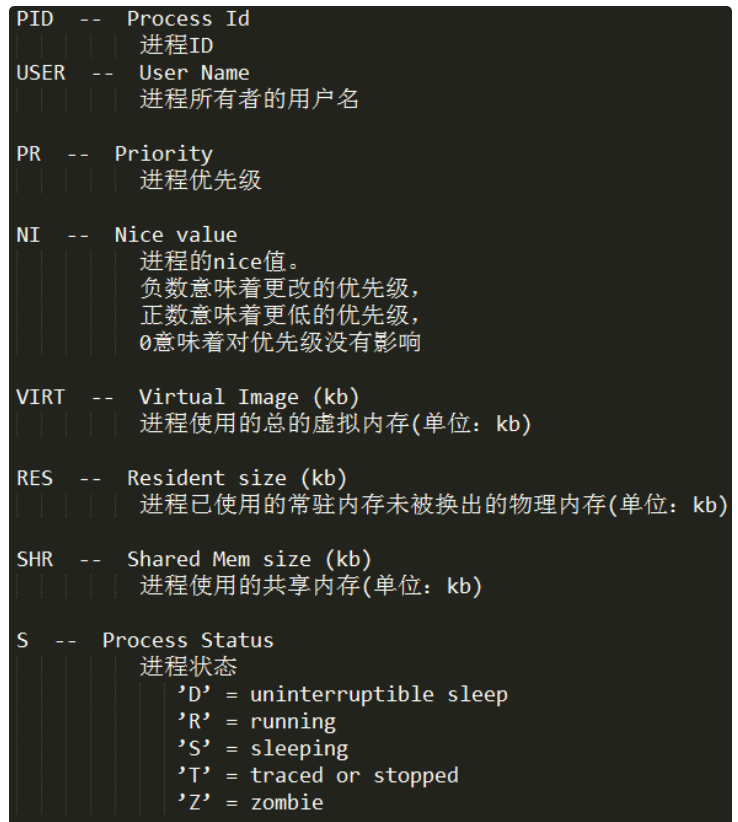
[**可用内存 = free + buffers + cached**](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

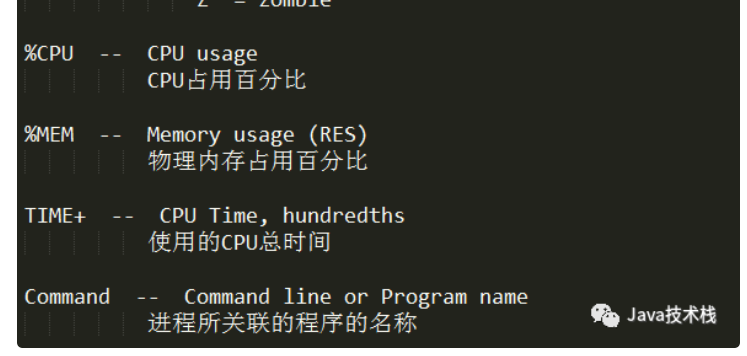
[**第5行**](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

[Swap: 4194300k total,  255104k used, 3939196k free, 10422508k cached](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

[虚拟内存使用情况](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)

[**其余行**](http://mp.weixin.qq.com/s?__biz=MzI3ODcxMzQzMw==&mid=2247531800&idx=2&sn=fbf496a2626d9d550c4ae95bb4f60c11&chksm=eb50f82edc2771386229a5bb81d865eee375f8ecbec665494417beb8c933260114d67b34c3a0&scene=21#wechat_redirect)





#### 2、显示完整命令

top –c

## free

### 基本概念

free命令用于显示系统中的内存使用情况。它会显示物理内存和交换空间的统计信息，包括可用内存、已用内存、缓存、交换空间等。

下面是free命令输出的各列详细解释：

total：总内存量，表示系统上总共可用的物理内存总量（单位为KiB）。

used：已用内存量，表示当前系统中正在使用的物理内存总量（单位为KiB）。

free：可用内存量，表示当前系统中未被使用的物理内存总量（单位为KiB）。

shared：共享内存量，表示多个进程共享使用的内存总量（单位为KiB）。

buffers：缓冲区内存量，表示用于存储文件系统元数据的缓冲区（单位为KiB）。

cached：缓存内存量，表示用于存储文件系统数据的缓存（单位为KiB）。

available：可用内存量，表示系统中可供进程使用的内存总量，包括未使用的内存、缓冲区和缓存（单位为KiB）。

此外，free命令还会显示交换空间的统计信息：

Swap：总交换空间量，表示系统上总共可用的交换空间总量（单位为KiB）。

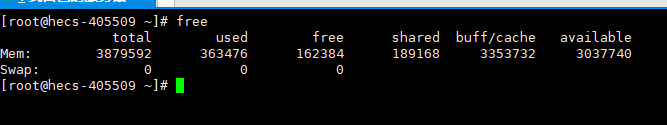
used：已用交换空间量，表示当前系统中正在使用的交换空间总量（单位为KiB）。

free：可用交换空间量，表示当前系统中未被使用的交换空间总量（单位为KiB）。

free命令的输出结果可以帮助我们了解系统当前内存的使用情况。特别是当系统内存紧张时，通过查看free命令的输出，可以快速判断是否存在内存压力，并采取相应的措施来优化系统性能，如释放不必要的内存或增加物理内存容量。

### 示例

free



## tar

### 基本概念

tar 命令是 Linux 系统中非常常用的文件打包和压缩命令，可以将多个文件或目录打包成一个文件，并可选地对其进行压缩。tar 命令用于创建 tar 归档文件，它可以将一个或多个文件或目录打包成一个单独的文件，并可以对其进行压缩操作，常见的压缩格式有 .tar、.gz、.bz2、.xz、.Z 等。

### 语法

tar [options] <archive.tar> <files/directories>

<archive.tar> 表示要创建的归档文件名，<files/directories> 则表示要添加到归档文件中的文件或目录。下面介绍一些 tar 命令的常用选项：

-c：创建新归档文件

-x：从归档文件中提取文件。

-v：显示归档文件中正在处理的文件的详细信息。

### 示例

#### 1、解压并解压缩归档文件

tar -zxvf archive.tar.gz

#### 2、创建归档文件

tar -cvf archive.tar file1 file2 dir1

将多个文件或目录打包成一个 .tar 归档文件。

#### 3、解压归档文件

tar -xvf archive.tar

从 .tar 归档文件中提取文件或目录。

## cp

### 基本概念

cp 是 Linux 中用于复制文件和目录的命令。

### 语法

cp [选项] 源文件 目标文件

以下是 cp 命令常用的选项及其功能：

-r 或 –-recursive：递归复制整个目录及其子目录。如果要复制的是目录，则必须使用该选项。

-i 或 --interactive：交互式操作，在复制前询问是否覆盖已经存在的文件。

-u 或 --update：只复制源文件中比目标文件新或者目标文件不存在的文件。

-v 或 --verbose：显示详细的复制过程。

-p 或 --preserve：保留文件的属性，包括所有者、组、权限等。

-f 或 --force：强制复制，即使目标文件已经存在也会覆盖。

--no-clobber：不覆盖已存在的目标文件。

--backup：在复制目标文件之前进行备份。

### 示例

#### 1、复制文件

cp file1.txt file2.txt # 将 file1.txt 复制为 file2.txt

#### 2、复制目录

cp -r dir1 dir2 # 递归复制 dir1 目录及其子目录到 dir2

## mv

### 基本概念

用于移动或重命名文件和目录的命令。

### 语法

mv [option] [source] [destination]

### 示例

#### 1、****移动文件****

mv file.txt /home/user/

要将文件file.txt从当前目录移动到目录/home/user/中。

## mount

### 基本概念

### 语法

### 实例

## iostat

## df

### 基本概念

查看文件系统磁盘空间使用情况统计。

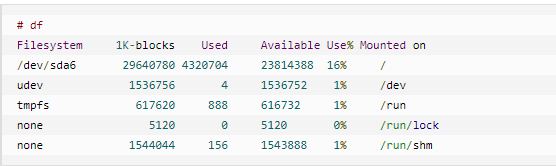
### 语法

df [选项]... [FILE]...

文件-h, --human-readable 使用人类可读的格式(预设值是不加这个选项的...)

### 实例

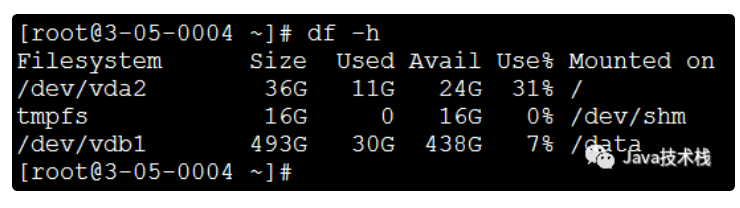
#### 1、显示文件系统的磁盘使用情况统计



第一列指定文件系统的名称，第二列指定一个特定的文件系统1K-块1K是1024字节为单位的总内存。用和可用列正在使用中，分别指定的内存量。

使用列指定使用的内存的百分比，而最后一栏"安装在"指定的文件系统的挂载点。

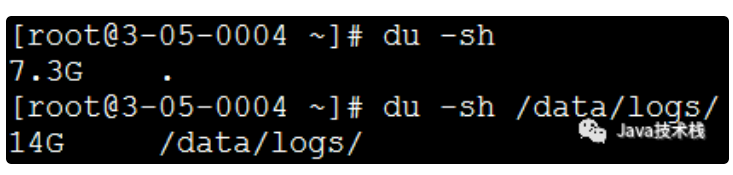
df –h



## du

### 基本概念

查看（计算）文件大小



还可以这样：du --max-depth=2 --block-size=M

或者：ll --block-size=M

## cat

### 基本概念

用于连接文件并打印到标准输出设备上。它可以读取和合并文件，并将其内容写入到标准输出。如果未指定文件名或者指定连字符 - 作为参数，则从标准输入读取内容，cat 命令复制标准输入到标准输出。cat 命令最常用于打印/查看一个或多个文本文件的内容。

### 语法

cat [OPTIONS] [FILE\_NAMES]

说明：

FILE\_NAMES 零个或多个文件名，可选参数。

### 实例

#### 1、查看文件内容

cat /etc/issue

cat 命令最基本和最常见的用法就是查看/打印文件的内容。读取文件内容写入到[标准输出](https://www.myfreax.com/stdout-stdin-and-redirection/)。例如命令 cat /etc/issue 查看 /etc/issue 文件内容，将会输出当前的 Linux 发行版的名称。

## head

### 基本概念

head 命令可用于查看文件的开头部分的内容，有一个常用的参数 -n 用于显示行数，默认为 10，即显示 10 行的内容。

### 语法

## tail

### 基本概念

tail 命令可用于查看文件的内容，有一个常用的参数 -f 常用于查阅正在改变的日志文件。tail -f filename 会把 filename 文件里的最尾部的内容显示在屏幕上，并且不断刷新，只要 filename 更新就可以看到最新的文件内容。

### 语法

tail [参数] [文件]

参数：

-f：循环读取

-q：不显示处理信息

-n<行数> 显示文件的尾部n行内容

### 示例

#### 1、显示 notes.log 文件的最后10 行

tail notes.log

默认显示最后10行

#### 2、跟踪名为 notes.log文件的增长情况

tail -f notes.log

此命令显示 notes.log 文件的最后 10 行。当将某些行添加至 notes.log 文件时，tail 命令会继续显示这些行，显示一直继续，直到您按下（Ctrl-C）组合键停止显示。

#### 3、显示行数

tail -n 20 filename.txt

将显示文件 filename.txt 的最后 20 行内容。

#### 4、实时显示文件的最后10行内容

tail -f -n 10 filename.txt

## more

### 基本概念

Linux more 命令类似 cat ，不过会以一页一页的形式显示，更方便使用者逐页阅读，而最基本的指令就是按空白键（space）就往下一页显示，按 b 键就会往回（back）一页显示，而且还有搜寻字串的功能（与 vi 相似），使用中的说明文件，请按 h 。

### 语法

more [-dlfpcsu] [-num] [+/pattern] [+linenum] [fileNames..]

参数：

-num：一次显示的行数

+num：从第 num 行开始显示

fileNames：欲显示内容的文档，可为复数个数

### 示例

#### 1、逐页显示 testfile 文档内容，如有连续两行以上空白行则以一行空白行显示。

more -s testfile

#### 2、从第 20 行开始显示 testfile 之文档内容。

more +20 testfile

### 常用操作命令

#### Ctrl+F

向下滚动一屏

#### 空格键

向下滚动一屏

#### Ctrl+B

返回上一屏

#### =

输出当前行的行号

#### q

退出more

## less

### 基本概念

less 与 more 类似，less 可以随意浏览文件，支持翻页和搜索，支持向上翻页和向下翻页。

### 语法

less [选项] 文件名

#### 选项

-N：显示行号。

-i：忽略搜索时的大小写。

+N：从指定行号开始显示文件。

q：退出less。

### 特点

1、分页浏览：less 将文件内容分成一页一页进行展示，当内容超出当前屏幕时，可以通过按下空格键向下翻页，按下b键向上翻页，或使用方向键进行滚动。

### 示例

#### 1、查看文件内容

less file.txt

#### 2、显示行号并搜索关键字

less -N file.txt

然后输入 /keyword 进行向下搜索。

#### 3、直接定位到指定行号

less +10 file.txt

从第 10 行开始显示文件内容。

#### 4、显示文件最后一页

less -N +G 文件名

## find

### 基本概念

find命令用来在指定目录下查找文件。任何位于参数之前的字符串都将被视为欲查找的目录名。如果使用该命令时，不设置任何参数，则find命令将在当前目录下查找子目录与文件。并且将查找到的子目录和文件全部进行显示。

语法：find(选项)（参数）

参数：起始目录：查找文件的起始目录

示例：

列出当前目录及子目录下所有文件和文件夹

find .

在/home目录下查找以.txt结尾的文件名

find /home –name "\*.txt"

当前目录及子目录下查找所有以.txt和.pdf结尾的文件

find . \( -name "\*.txt" -o -name "\*.pdf" \)

或

find . -name "\*.txt" -o -name "\*.pdf"

## su

### 基本说明

### 语法

### 实例

## sudo

### 基本概念

简单的说sudo是一种权限管理机制，管理员可以授权于一些普通用户去执行一些 root 执行的操作，而不需要知道 root 的密码。严谨些说sudo 允许一个已授权用户以超级用户或者其它用户的角色运行一个命令。当然，能做什么不能做什么都是通过安全策略来指定的。sudo 支持插件架构的安全策略，并能把输入输出写入日志。第三方可以开发并发布自己的安全策略和输入输出日志插件，并让它们无缝的和 sudo 一起工作。默认的安全策略记录在 /etc/sudoers 文件中。而安全策略可能需要用户通过密码来验证他们自己。也就是在用户执行 sudo 命令时要求用户输入自己账号的密码。如果验证失败，sudo 命令将会退出。(注意本文介绍的 sudo 命令运行在 ubuntu 14.04中。)

### 作用

以其他用户（默认root）身份执行命令，实现临时权限提升。

### 语法

sudo [-bhHpV][-s ][-u <用户>][指令]

或者

sudo [-klv]

### 参数

#### -b

在后台执行指令

#### -h

显示帮助

#### -H

将HOME环境变量设为新身份的HOME环境变量

#### -k

结束密码的有效期限，也就是下次再执行sudo时便需要输入密码

#### -l

列出目前用户可执行与无法执行的指令

-p

改变询问密码的提示符号

 -s

执行指定的shell  
-u  <用户>

以指定的用户作为新的身份，若不加上此参数，则预设以root作为新的身份

-v

延长密码有效期限5分钟。  
-V

显示版本信息。

-S

从标准输入流替代终端来获取密码

### 基本配置

系统默认创建了一个名为 sudo 的组。只要把用户加入这个组，用户就具有了 sudo 的权限。至于如何把用户加入 sudo 组，您可以直接编辑 /etc/group文件，当然您得使用一个有 sudo 权限的用户来干这件事，使用如下命令：

sudo vim /etc/group

在sudo 组中加入新的用户，要使用逗号分隔多个用户或者您可以使用 usermod 命令把用户添加到一个组中，命令如下：

sudo usermod -a -G sudo jack

上面的设置中我们把用户jack添加到了sudo组中，所以当用户 jack 登录后就可以通过 sudo 命令以 root 权限执行命令了。

### 用法

#### 1、基本使用

sudo ls /root # 以root身份查看/root目录

sudo apt update # 以root身份更新包列表

#### 2、切换到root shell



### 配置文件

1、/etc/sudoers - 主配置文件

2、使用visudo命令安全编辑

## vim

### 基本概念

vi编辑器是所有Unix及Linux系统下标准的编辑器，它的强大不逊色于任何最新的文本编辑器。vim 具有程序编辑的能力，可以以字体颜色辨别语法的正确性，方便程序设计；因为程序简单，编辑速度相当快速。vim可以当作vi的升级版本，他可以用多种颜色的方式来显示一些特殊的信息。vim会依据文件扩展名或者是文件内的开头信息，判断该文件的内容而自动的执行该程序的语法判断式，再以颜色来显示程序代码与一般信息。vim里面加入了很多额外的功能，例如支持正则表达式的搜索、多文件编辑、块复制等等。 这对于我们在Linux上进行一些配置文件的修改工作时是很棒的功能。

### vi的基本概念

基本上vi可以分为三种状态，分别是命令模式（command mode）、插入模式（Insert mode）和底行模式（last line mode），各模式的功能区分如下：

#### （1）命令行模式

控制屏幕光标的移动，字符、字或行的删除，移动复制某区段及进入Insert mode下，或者到last line mode。

#### （2）插入模式

只有在Insert mode下，才可以做文字输入，按「ESC」键可回到命令行模式

#### （3）底行模式

将文件保存或退出vi，也可以设置编辑环境，如寻找字符串、列出行号等。不过一般我们在使用时把vi简化成两个模式，就是将底行模式（last line mode）也算入命令行模式command mode）。

### vi基本操作

#### （1）进入vi

在系统提示符号输入vi及文件名称后，就进入vi全屏幕编辑画面。不过有一点要特别注意，就是您进入vi之后，是处于「命令行模式（command mode）」，您要切换到「插入模式（Insert mode）」才能够输入文字。初次使用vi的人都会想先用上下左右键移动光标，结果电脑一直哔哔叫，把自己气个半死，所以进入vi后，先不要乱动，转换到插入模式。

#### （2）切换到插入模式编辑文件

在命令行模式（command mode）下**按一下字母「i」**就可以进入插入模式（Insert mode），这时候你就可以开始输入文字了。

#### （3） Insert 的切换

您目前处于「插入模式（Insert mode）」，您就只能一直输入文字，如果您发现输错了字，想用光标键往回移动，将该字删除，就要**先按一下「ESC」键**转到「命令行模式（command mode）」再删除文字。

#### （4）退出vi及保存文件

在命令行模式下，按一下：冒号键进入Last line mode，例如：

: w filename （输入 「w filename」将文章以指定的文件名filename保存）

: wq (输入wq，存盘并退出vi)

: q! (输入q!，不存盘强制退出vi)

### 命令行模式功能键

#### （1）插入模式

按「i」切换进入插入模式「insert mode」，按“i”进入插入模式后是从光标当前位置开始输入文件。

按「a」进入插入模式后，是从目前光标所在位置的下一个位置开始输入文字

按「o」进入插入模式后，是插入新的一行，从行首开始输入文字。

#### （2） 从插入模式切换为命令行模式

按「ESC」键

#### （3）移动光标

vi可以直接用键盘上的光标来上下左右移动，但正规的vi是用小写英文字母「h」、「j」、「k」、「l」，分别控制光标左、下、上、右移一格。

按「ctrl」+「b」：屏幕往“后”移动一页

按「ctrl」+「f」：屏幕往“前”移动一页

按「ctrl」+「u」：屏幕往“后”移动半页

按「ctrl」+「d」：屏幕往“前”移动半页

按数字0：移到行的开头

按「$」：移动到光标所在行的“行尾”

按「G」：移动到文件的最后一行

## jps

### 基本概念

jps 命令是 JDK 提供的一个简单实用的命令行工具，用于查看正在运行的 Java 进程。通过执行 jps 命令，可以获取到进程 ID 和主类名称等信息，对于监视和管理 Java 进程非常有用。此外还可以结合其他命令使用，进行更深入的诊断和分析操作。

### 语法

jps [options]

-l：显示完整的类名（包括包路径）

-m：同时显示传递给主类的参数。

-v：显示传递给 JVM 的参数。

-q：仅显示进程 ID，不显示类名或传递给 JVM 的参数。

### 示例

#### 1、列出正在运行的 Java 进程

jps

这会显示所有正在运行的 Java 进程的进程 ID（PID）和类名。

#### 2、列出正在运行的 Java 进程和完整的类名

jps -l

这会显示所有正在运行的 Java 进程的进程 ID、完整的类名（包括包路径）

#### 3、列出正在运行的 Java 进程和传递给 JVM 的参数

jps -v

这会显示所有正在运行的 Java 进程的进程 ID、类名和传递给 JVM 的参数。

需要注意的是，jps 命令需要在安装了 Java 开发工具包（JDK）的环境中才能运行。如果在运行 jps 命令时出现问题，可以检查是否正确安装了 JDK，并确保 JAVA\_HOME 环境变量已正确设置。

## scp

### 基本概念

scp 是 Linux（Unix）系统中的一个命令，用于在本地主机和远程主机之间进行文件传输。它的作用是通过安全的加密通道在本地和远程主机之间复制文件或目录。

### 语法

scp [选项] 源 目标

选项：

-r：如果源是目录，则递归地复制整个目录及其内容。

-P 端口号：指定远程主机的 SSH 端口号，默认为 22。

-p：保留原始文件的修改时间、访问时间和权限等信息。

-i 密钥文件：使用指定的密钥文件进行身份验证。

### 示例

#### 1、将本地文件 file.txt 复制到远程主机的 /home/user/ 目录下



#### 2、从远程主机复制文件 file.txt 到本地的 /path/to/local/ 目录下：



#### 3、递归地复制整个目录 dir 到远程主机的 /home/user/ 目录下



## nohup

### 基本概念

用于在后台运行命令，并使其在终端关闭后继续执行。即使用户注销或关闭终端也不会停止该命令的执行。这对于需要长时间运行的命令非常有用，例如备份、编译或文件传输。

### 语法

nohup command &

参数说明：

command：需要在后台运行的命令。

&：将命令置于后台运行。

使用 nohup 命令可以将一个进程放到后台运行，并且不受当前终端会话的影响。如果没有使用 nohup 命令，当用户注销或关闭终端时，该进程也会停止运行。nohup 命令的输出被重定向到当前目录下的 nohup.out 文件中。这意味着可以在以后查看该命令的输出。

### 示例

nohup java -jar -Xms1024m -Xmx1024m -XX:PermSize=256m -XX:MaxPermSize=512m -XX:MaxNewSize=512m -Dspring.cloud.nacos.discovery.namespace=pre -Dspring.cloud.nacos.config.namespace=pre /data/package/${PROJECT\_NAME} >/dev/null 2>&1 &

# 防火墙

## 基本概念

Linux防火墙是网络安全的第一道防线。

## 核心功能

### 1、访问控制

（1）控制进出系统的网络流量

（2）根据IP地址、端口、协议过滤数据包

（3）阻止未授权的网络连接

### 2、端口管理

（1）开放必要的服务端口（如SSH:22、HTTP:80）

（2）关闭不需要的端口，减少攻击面

（3）限制特定IP访问特定端口。

# 关闭防火墙

## 命令

systemctl stop firewalld

**systemctl**：systemd系统和服务管理器命令

**stop**：停止服务的操作

**firewalld**：动态防火墙守护进程服务名

# 提高工作效率的命令

## 1、使用tab键进行自动完成

## 2、切换回上一个工作目录

使用cd -

这会将您带回到上一个工作目录。您无需输入长目录路径，也无需复制粘贴

说明：你至少要执行一次cd命令，否则 OLDPWD 环境变量不会包含任何值。

## 3、返回主目录

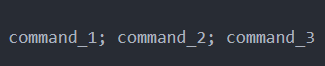
使用cd或者cd ~

## 4、列出目录的内容

使用ll而不是ls -l

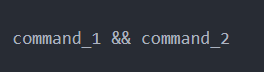
## 5、在一个命令中运行多个命令

您可以使用“;”分隔符。这样，您可以在一行中运行许多命令。无需等待先前的命令完成后再执行其他任务。



## 6、仅在上一个命令成功的情况下，才能在一个命令中运行多个命令

在这种情况下，可以使用&&分隔符。&&确保下一条命令仅在上一条命令成功执行时运行



## 7、轻松搜索您使用过的命令

只需使用ctrl + r键即可启动反向搜索并键入命令的某些部分。它将查询历史记录，并向您显示与搜索词匹配的命令。



说明：在某些Bash Shell中，还可以在搜索词中使用Page Up和Down键，它将自动完成命令。

## 8、解除Linux终端意外冻结的Ctrl + S

解冻终端是用Ctrl-Q

## 9、读取压缩日志而不解压缩

z命令提供了用于处理日志文件（例如less，cat，grep等）的常规命令的替代方法。

这样您就可以使用zless，zcat，zgrep等命令查看压缩包的内容，甚至不必显式提取压缩文件。



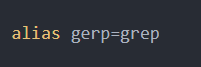
## 10、使用!$重新使用上一个命令中的最后一项

在许多情况下，使用上一个命令的参数很方便。假设您必须创建一个目录，然后进入新创建的目录。那么，您可以使用!$选项。



## 11、使用别名来修正错别字

例如，您可能经常将grep输入为gerp。如果您以这种方式在您的bashrc中放置一个别名



这样，您无需再次输入命令。

## 12、在Linux终端中复制粘贴

要善于用复制和粘贴的快捷键

## 13、终止正在运行的命令/进程

可以按Ctrl + C停止该正在运行的命令。

## 14、清空文件而不删除它

如果只想清空文本文件的内容而不删除文件本身，则可以使用类似于以下命令



## 15、查找是否有包含特定文本的文件



## 16、对任何命令都可使用帮助命令（help）

# 查看日志的命令

## 1、查询指定时间之后的日志信息

命令：

grep '时间' 日志文件名

**查询2021年10月20日，15点40分到41分的日志**

grep '2021-10-20 15:4[0-1]' logback.log

**查询2021年10月20日，15点41分的日志**

grep '2021-10-20 15:41' logback.log

**查询2021年10月20日，15点41分30秒的日志**

grep '2021-10-20 15:41:30' logback.log

2、查询指定的信息

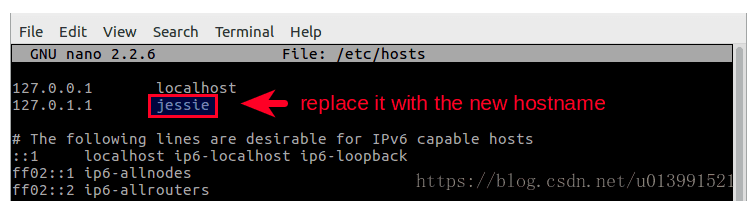
# 更改Linux系统的主机名

## 第一种方法：修改配置文件

主机名保存在/etc/hostname文件里，所以我们可以打开这个文件，手动编辑主机名。

sudo nano /etc/hosts

把旧的主机名删除，替换为新的主机名，保存文件就行了。要注意大小写



如果你不更新/etc/hosts文件，那么有的程序，如sudo，不知道如何解析新的主机名。如果你在更改Linux服务器的主机名，那么新的主机名应该要解析为Linux服务器的公网IP。如果更改个人电脑的主机名，那么新的主机名应该解析为127.0.0.1，或者127.0.1.1。

## 第二种方法：hostnamectl命令

命令语法：

sudo hostnamectl set-hostname <newhostname>

这条命令会删除/etc/hostname文件中的主机名，然后替换为新的主机名。和第一种方法一样，我们也需要更新/etc/hosts文件。这两种方法的本质都是一样的。

# linux系统如何清理缓存

## 1、缓存机制介绍

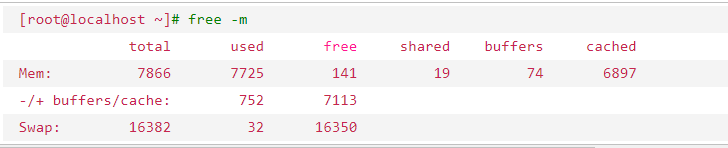
在Linux系统中，为了提高文件系统性能，内核利用一部分物理内存分配出缓冲区，用于缓存系统操作和数据文件，当内核收到读写的请求时，内核先去缓存区找是否有请求的数据，有就直接返回，如果没有则通过驱动程序直接操作磁盘。

缓存机制优点：减少系统调用次数，降低CPU上下文切换和磁盘访问频率。

CPU上下文切换：CPU给每个进程一定的服务时间，当时间片用完后，内核从正在运行的进程中收回处理器，同时把进程当前运行状态保存下来，然后加载下一个任务，这个过程叫做上下文切换。实质上就是被终止运行进程与待运行进程的进程切换。

## 2、查看缓存区及内存使用情况

使用命令：free –m



从上面的命令结果显示中可以看出：内存总共8G，已使用7725M，剩余141M，不少的人都是这么看的，但其实这样并不能作为实际的使用率。因为有了缓存机制，具体算法如下：

空闲内存=free（141）+buffers（74）+cached（6897）  
已用内存=total（7866）-空闲内存  
由此算出空闲内存是7112M，已用内存754M，这才是真正的使用率，也可参考-/+ buffers/cache这行信息也是内存正确使用率。

## 3、缓存区分buffers和cached区别

内核在保证系统能正常使用物理内存和数据量读写情况下来分配缓冲区大小，buffers用来缓存metadata及pages，可以理解为系统缓存，例如，vi打开一个文件。cached是用来给文件做缓存，可以理解为数据块缓存，例如，dd if=/dev/zero of=/tmp/test count=1 bs=1G 测试写入一个文件，就会被缓存到缓冲区中，当下一次再执行这个测试命令时，写入速度会明显很快。

## 4、Swap用途

Swap意思是交换分区，通常我们说的虚拟内存，是从硬盘中划分出的一个分区。当物理内存不够用的时候，内核就会释放缓存区（buffers/cache）里一些长时间不用的程序，然后将这些程序临时放到Swap中，也就是说如果物理内存和缓存区内存不够用的时候，才会用到Swap。

swap清理命令：

swapoff –a && swapon –a

说明：运行这个命令需要管理员的权限，这样清理有个前提条件，空闲的内存必须比已经使用的swap空间大。

## 5、释放缓存区内存的方法

清理pagecache（页面缓存），命令：

/proc/sys/vm/drop\_caches 或者sysctl -w vm.drop\_caches=1

清理dentries（目录缓存）和inodes，命令：

/proc/sys/vm/drop\_caches 或者sysctl -w vm.drop\_caches=2

清理pagecache、dentries和inodes，命令：

/proc/sys/vm/drop\_caches 或者sysctl -w vm.drop\_caches=3

上面三种方式都是临时释放缓存的方法，要想永久释放缓存，需要在/etc/sysctl.conf文件中配置：vm.drop\_caches=1/2/3，然后sysctl -p生效即可。说明：修改这个文件都需要管理员的权限。

另外，可以使用sync命令来清理文件系统缓存，还会清理僵尸(zombie)对象和它们占用的内存，例如：sync

说明：上面操作在大多数情况下都不会对系统造成伤害，只会有助于释放不用的内存。但是如果在执行这些操作时正在写数据，那么实际上在数据到达磁盘之前就将它从文件缓存中清除掉了，这可能会造成很不好的影响。那么如果避免这种事情发生呢？因此，这里不得不提一下/proc/sys/vm/vfs\_cache\_pressure这个文件，告诉内核，当清理inoe/dentry缓存时应该用什么样的优先级，如下图：



释放内存前先使用sync命令做同步，以确保文件系统的完整性，将所有未写的系统缓冲区写到磁盘中，包含已修改的 i-node、已延迟的块 I/O 和读写映射文件。否则在释放缓存的过程中，可能会丢失未保存的文件。

/proc是一个[虚拟文件系统](https://www.baidu.com/s?wd=%E8%99%9A%E6%8B%9F%E6%96%87%E4%BB%B6%E7%B3%BB%E7%BB%9F&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLmyRkrAnLmHTvryfYPhDd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPH6Yn1DzPWms)，可以通过对它的读写操作作为与kernel实体间进行通信的一种手段。也就是说可以通过修改/proc中的文件，来对当前kernel的行为做出调整。也就是说我们可以通过调整/proc/sys/vm/drop\_caches来释放内存。



# 问题

## 1、-bash: lsb\_release: command not found

### 解决方法

## 2、FirewallD is not running

它的意思是 firewalld 服务当前并未启动或没有在运行。Firewalld 是一个动态管理 Linux 防火墙的工具，通常用于 CentOS, RHEL, Fedora 等发行版上。它可以帮助管理员通过命令行或图形界面来配置防火墙规则，例如开放端口、配置区域、设置服务访问控制等。firewalld 的设计比传统的 iptables 更加灵活和易于管理。

### 原因

1、Firewalld 服务没有启动：如果你在运行 firewalld 相关的命令时收到此提示，意味着系统中的 firewalld 服务当前没有启动。

2、系统没有安装 firewalld：在某些 Linux 发行版中，firewalld 可能并未预装或已被卸载。这时你将无法使用它来管理防火墙。

3、服务被禁用：firewalld 服务可能已经安装但被配置为不开启，或者被管理员禁用。

### 解决方法

#### 1、检查 firewalld 服务的状态

使用 systemctl 命令来查看 firewalld 服务的状态

sudo systemctl status firewalld

如果输出类似于 firewalld.service - firewalld，并且状态显示为 inactive (dead)，说明服务当前没有运行。

如果输出显示 firewalld.service - firewalld，并且状态为 active (running)，说明服务正在运行。

#### 2、启动 firewalld 服务

如果 firewalld 服务未运行，可以使用以下命令来启动它

sudo systemctl start firewalld

这将立即启动防火墙服务。如果你想让 firewalld 在每次系统启动时自动启动，可以使用：

sudo systemctl enable firewalld

这会将 firewalld 服务设置为开机自启。

#### 3、安装 firewalld

如果 firewalld 未安装，可以使用包管理工具进行安装。以 **CentOS** 和 **RHEL** 为例，可以使用以下命令安装：

sudo yum install firewalld

对于 **Ubuntu** 或 **Debian** 系统，可以使用以下命令：

sudo apt install firewalld

安装完成后，使用 systemctl 启动并启用服务。

#### 4、检查防火墙状态

在 firewalld 启动后，你可以使用以下命令来检查当前的防火墙状态和规则：

sudo firewall-cmd --state

如果防火墙正在运行，命令会输出 running。如果它没有运行，你会看到类似 "not running" 的消息。

### 总结

当你看到 "FirewallD is not running" 输出时，说明 firewalld 防火墙服务当前没有运行。你可以通过查看服务状态、启动服务、安装或重新启用 firewalld 来解决这个问题。