## cd、pwd、ls命令

pwd : 显示当前操作目录

格式：cd [dirName] 跳转指定文件夹

cd - 返回上一次所处的目录

cd ~ 用户home目录

cd ~username 切换其他用户home目录

cd .. 返回上一级目录

cd ../.. 返回当前目录的上两层

ls 显示当前目录下的文件夹

ls –a 查看全部文件（包括.隐藏文件）

ls –l 显示文件大小等详情信息（相当于ll）

ls –ld 查看目录属性及权限信息

# Shell终端常用命令

Shell（也称为终端或壳）充当的是人与内核（硬件）

之间的翻译官，用户把一些命令“告诉”终端，它就会调用相应的程序服务去完成某些工作。

➢ 通过上下方向键来调取过往执行过的 Linux 命令；

➢ 命令或参数仅需输入前几位就可以用 Tab 键补全；

➢ 具有强大的批处理脚本；

➢ 具有实用的环境变量功能。

## man（查看帮助信息）

格式：man 命令名称

在 man 命令帮助信息的界面中，所包含的常用操作按键及其用途：

空格键 向下翻一页

PaGe down 向下翻一页

PaGe up 向上翻一页

home 直接前往首页

end 直接前往尾页

/ 从上至下搜索某个关键词，如“/linux”

? 从下至上搜索某个关键词，如“?linux”

n 定位到下一个搜索到的关键词

N 定位到上一个搜索到的关键词

q 退出帮助文档

man 命令帮助信息的结构以及意义

NAME 命令的名称

SYNOPSIS 参数的大致使用方法

DESCRIPTION 介绍说明

EXAMPLES 演示（附带简单说明）

OVERVIEW 概述

DEFAULTS 默认的功能

OPTIONS 具体的可用选项（带介绍）

ENVIRONMENT 环境变量

FILES 用到的文件

SEE ALSO 相关的资料

HISTORY 维护历史与联系方式

## echo（输出到终端）

echo 命令用于在终端输出字符串或变量提取后的值，格式为“echo [字符串 | $变量]”。

输出字符串：echo helloworld

输出变量值：echo $SHELL

## date（系统时间）

date 命令用于显示及设置系统的时间或日期，格式为“date [选项] [+指定的格式]”。

在date 命令中输入以“+”号开头的参数，即可按照指定格式来输出系统的时

间或日期。

date 命令中的参数以及作用

%Y 年（四位年信息）

%m 月（两位）

%d 日（两位）

%t 跳格[Tab 键]

%H 小时（00～23）

%I 小时（00～12）

%M 分钟（00～59）

%S 秒（00～59）

%j 今年中的第几天

例如：date “+%Y-%m-%d %H:%M:%S” 输出格式：年-月-日 时:分:秒

将系统的当前时间设置为 2017 年 9 月 1 日 8 点 30 分的 date 命令如下所示：

[root@linuxprobe ~]# date -s "20170901 8:30:00"

Fri Sep 1 08:30:00 CST 2017

date 命令中的参数%j 可用来查看今天是当年中的第几天。这个参数能够很好地区分备份时

间的新旧，即数字越大，越靠近当前时间。该参数的使用方式以及显示结果如下所示。

[root@linuxprobe ~]# date "+%j"

244

## reboot（重启）

## poweroff （关机）

## wget （下载网络文件）

wget 命令用于在终端中下载网络文件，格式为“wget [参数] 下载地址”。

wget 命令的参数以及作用

-b 后台下载模式

-P 下载到指定目录

-t 最大尝试次数

-c 断点续传

-p 下载页面内所有资源，包括图片、视频等

-r 递归下载

例如: wget <http://www.linuxprobe.com/docs/LinuxProbe.pdf>

## ps（查看系统进程状态）

ps命令用于查看系统中的进程状态，格式为“ps[参数]”。

ps 命令的参数以及作用

-a 显示所有进程（包括其他用户的进程）

-u 用户以及其他详细信息

-x 显示没有控制终端的进程

Linux 系统中，有 5 种常见的进程状态，分别为运行、中断、不可中断、僵死与停止，其各自

含义如下所示：

➢ R(运行) 进程正在运行或在运行队列中等待。

➢ S(中断) 进程处于休眠中，当某个条件形成后或者接收到信号时，则脱离该

状态。

➢ D(不可中断) 进程不响应系统异步信号，即便用 kill 命令也不能将其中断。

➢ Z(僵死) 进程已经终止，但进程描述符依然存在, 直到父进程调用 wait4()系统函数

后将进程释放。

➢ T(停止) 进程收到停止信号后停止运行。

当执行 ps aux 命令后通常会看到如表 2-7 所示的进程状态:

参数标题 中文注释 内容举例

USER 进程所有者 root

PID 进程ID号 1

%CPU 运算器占用率 0.0

$MEM 内存占用率 0.1

VSZ 虚拟内存使用量（单位:KB） 193792

RSS 占用的内存量（单位：KB） 6936

TTY 所在终端 ？

STAT 进程状态 Ss

START 被启动的时间 13:44

TIME 实际使用CPU的时间 0.04

COMMAND 命令名称与参数 /usr/lib/systemd/systemd --switched-root --system --deserialize 22

在linux系统中的命令参数有长短格式之分，长格式和长格式的两个参数不能合并，长格式

和短格式之间也不能合并，但短格式和短格式之间是可以合并的，合并后仅保留一个“-

（减号）即可，另外ps命令可允许参数不加减号（-）,因此可以写成ps aux的样子。

## top （任务管理器）

top 命令用于动态地监视进程活动与系统负载等信息，其格式为 top。

top 命令执行结果的前 5 行为系统整体的统计信息，其所代表的含义如下

第 1 行：系统时间、运行时间、登录终端数、系统负载（三个数值分别为 1 分钟、5

分钟、15 分钟内的平均值，数值越小意味着负载越低）。

➢ 第 2 行：进程总数、运行中的进程数、睡眠中的进程数、停止的进程数、僵死的进程

数。

➢ 第 3 行：用户占用资源百分比、系统内核占用资源百分比、改变过优先级的进程资源

百分比、空闲的资源百分比等。

➢ 第 4 行：物理内存总量、内存使用量、内存空闲量、作为内核缓存的内存量。

➢ 第 5 行：虚拟内存总量、虚拟内存使用量、虚拟内存空闲量、已被提前加载的内存量。

## pidof（查询某个进程的pid值）

pidof 命令用于查询某个指定服务进程的 PID 值，格式为“pidof [参数] [服务名称]”。

每个进程的进程号码值（PID）是唯一的，因此可以通过 PID 来区分不同的进程。例如，

可以使用如下命令来查询本机上 sshd 服务程序的 PID：

[root@linuxprobe ~]# pidof sshd

2156

## Kill（终止进程）

kill 命令用于终止某个指定 PID 的服务进程，格式为“kill [参数] [进程 PID]”。

接下来，我们使用 kill 命令把上面用 pidof 命令查询到的 PID 所代表的进程终止掉，其命

令如下所示。这种操作的效果等同于强制停止 sshd 服务。

[root@linuxprobe ~]# kill 2156

加入参数 -9 表示强制立即停止进程运行 kill -9 2256

## Killall（终止某服务全部进程）

killall 命令用于终止某个指定名称的服务所对应的全部进程，格式为：“killall [参数] [服

务名称]”。

[root@linuxprobe ~]# pidof httpd

13581 13580 13579 13578 13577 13576

[root@linuxprobe ~]# killall httpd

[root@linuxprobe ~]# pidof httpd

[root@linuxprobe ~]#

如果在系统终端中执行一个命令后想立即停止它，可以同时按下ctrl+c组合键（生产环境中常用的一个快捷键），这样将立即终止该命令的进程。或者，如果有些命令在执行时不断地在屏幕上输出信息，影响到后续命令的输入，则可以在执行命令时在末尾添加上&符号，表示命令将进入系统后台执行。

# 运维命令

## Ifconfig (查看网卡信息)

ifconfig命令用于获取网卡配置与网络状态等信息，格式为“ifconfig [网络设备] [参数]”。

使用 ifconfig 命令来查看本机当前的网卡配置与网络状态等信息时，其实主要查看的就 是网卡名称、inet 参数后面的 IP 地址、ether 参数后面的网卡物理地址（又称为 MAC 地址）， 以及 RX、TX 的接收数据包与发送数据包的个数及累计流量（即下面加粗的信息内容）

## uname (系统内核版本信息)

uname 命令用于查看系统内核与系统版本等信息，格式为“uname [-a]”。

在使用 uname 命令时，一般会固定搭配上-a 参数来完整地查看当前系统的内核名称、主 机名、内核发行版本、节点名、系统时间、硬件名称、硬件平台、处理器类型以及操作系统名 称等信息。

顺带一提，如果要查看当前系统版本的详细信息，则需要查看 redhat-release 文件，其命 令以及相应的结果如下：

[root@linuxprobe ~]# cat /etc/redhat-release

Red Hat Enterprise Linux Server release 7.0 (Maipo)

## uptime (系统负载信息)

uptime 用于查看系统的负载信息，格式为 uptime。

uptime 命令真的很棒，它可以显示当前系统时间、系统已运行时间、启用终端数量以 及平均负载值等信息。平均负载值指的是系统在最近 1 分钟、5 分钟、15 分钟内的压力情况（下面加粗的信息部分）；负载值越低越好，尽量不要长期超过 1，在生产环境中不要超5 。

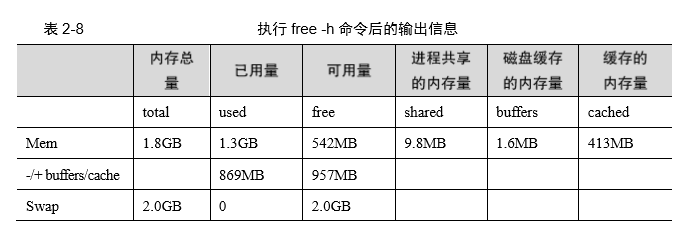
[root@linuxprobe ~]# uptime

22:49:55 up 10 min, 2 users, load average: 0.01, 0.19, 0.18

## free (查看内存使用量)

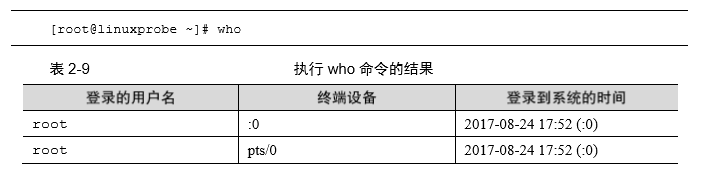
free 用于显示当前系统中内存的使用量信息，格式为“free [-h]”。 为了保证 Linux 系统不会因资源耗尽而突然宕机，运维人员需要时刻关注内存的使用量。 在使用 free 命令时，可以结合使用-h 参数以更人性化的方式输出当前内存的实时使用量信息。需要注意的是，输出信 息中的中文注释是作者自行添加的内容，实际输出时没有相应的参数解释。

[root@linuxprobe ~]# free -h



## who (查看当前用户)

who 用于查看当前登入主机的用户终端信息，格式为“who [参数]”。



## last(查看系统登录记录)

## history(查看历史操作命令)

格式：history [-c]

[root@localhost nodebook-20190602]# history

1 service mysql start

2 vim /etc/my.cnf

3 reboot

4 service mysql start

5 chown -R mysqlg:mysqlu /usr/local/mysql/data/

6 chown -R mysqlu:mysqlg /usr/local/mysql/data/

7 chmod -R 755 /usr/local/mysql/data

8 service mysql start

9 service mysql restart

10 cd /usr/local/mysql

加入-c参数清空历史记录

使用！编码数字 可以重复执行命令

改变 /etc/profile 文件中的

HISTSIZE 变量值 可以设置保存记录数量，默认保存1000条

## sosreport(搜集诊断信息并输出)

sosreport 命令用于收集系统配置及架构信息并输出诊断文档，格式为 sosreport。

当 Linux 系统出现故障需要联系技术支持人员时，大多数时候都要先使用这个命令来简

单收集系统的运行状态和服务配置信息，以便让技术支持人员能够远程解决一些小问题，亦

或让他们能提前了解某些复杂问题。在下面的输出信息中，加粗的部分是收集好的资料压缩

文件以及校验码，将其发送给技术支持人员即可：

[root@linuxprobe ~]# sosreport

sosreport (version 3.0)

This command will collect diagnostic and configuration information from

this Red Hat Enterprise Linux system and installed applications.

An archive containing the collected information will be generated in

/var/tmp and may be provided to a Red Hat support representative.

Any information provided to Red Hat will be treated in accordance with

the published support policies at:

https://access.redhat.com/support/

The generated archive may contain data considered sensitive and its

content should be reviewed by the originating organization before being

passed to any third party.

No changes will be made to system configuration.

Press ENTER to continue, or CTRL-C to quit.

Please enter your first initial and last name [linuxprobe.com]:

Please enter the case number that you are generating this report for:

Running plugins. Please wait ...

Running 70/70: yum...

Creating compressed archive...

Your sosreport has been generated and saved in:

/var/tmp/sosreport-linuxprobe.com-20170905230631.tar.xz

The checksum is: 79436cdf791327040efde48c452c6322

Please send this file to your support representative.

# 文本文件编辑命令

## cat（查看文本文件）适用内容较少

cat命令适合查看内容较少的文本文件

cat –n 显示行号

## more（查看内容较多的文本文件）

对于长篇的文本内容，推荐使用 more 命令来查看。more 命

令会在最下面使用百分比的形式来提示您已经阅读了多少内容。您还可以使用空格键或回车

键向下翻页

## head、tail、tailf

分别是查看文本文件的前n行、后n行、动态查看后n行

#head –n 5 <filename> 查看前五行，默认十行

#tail –n 5 <filename> 查看后五行，默认十行

#tailf <filename> 动态查看文件后面的内容，并且实时刷新（不退出的情况下）

## tr（文本内容批量替换）

tr 命令用于替换文本文件中的字符，格式为“tr [原始字符] [目标字符]”。

在很多时候，我们想要快速地替换文本中的一些词汇，又或者把整个文本内容都进行替

换，如果进行手工替换，难免工作量太大，尤其是需要处理大批量的内容时，进行手工替换更

是不现实。这时，就可以先使用 cat 命令读取待处理的文本，然后通过管道符（详见第 3 章）

把这些文本内容传递给 tr 命令进行替换操作即可。例如，把某个文本内容中的英文全部替换

为大写：[root@linuxprobe ~]# cat anaconda-ks.cfg | tr [a-z] [A-Z]

# 附录

VI与VIM区别

它们都是多模式编辑器，不同的是vim 是vi的升级版本，它不仅兼容vi的所有指令，而且还有一些新的特性在里面。

vim的这些优势主要体现在以下几个方面：

① 多级撤消

我们知道在vi里，按 u只能撤消上次命令，而在vim里可以无限制的撤消。

② 易用性

vi只能运行于unix中，而vim不仅可以运行于unix,windows ,mac等多操作平台。

③ 语法加亮

vim可以用不同的颜色来加亮你的代码。

④ 可视化操作

就是说vim不仅可以在终端运行，也可以运行于x window、 mac os、 windows。

⑤ 对vi的完全兼容

某些情况下，你可以把vim当成vi来使用。

vi和vim都是Linux中的编辑器，不同的是vim比较高级，可以视为vi的升级版本。vi使用于文本编辑，但是vim更适用于coding。

\*\*打开文件、保存、关闭文件(vim命令模式下使用) \*\*

命令的使用请先按ESC进入命令模式！

vim filename //打开filename文件

:w //保存文件

:w vpser.net //保存至vpser.net文件

:q //退出编辑器，如果文件已修改请使用下面的命令

:q! //退出编辑器，且不保存

:wq //退出编辑器，且保存文件

:x //退出编辑器，且保存文件

\*\*插入文本或行(vim命令模式下使用，执行下面命令后将进入插入模式，按ESC键可退出插入模式) \*\*

a //在当前光标位置的右边添加文本

i //在当前光标位置的左边添加文本

A //在当前行的末尾位置添加文本

I //在当前行的开始处添加文本(非空字符的行首)

O //在当前行的上面新建一行

o //在当前行的下面新建一行

R //替换(覆盖)当前光标位置及后面的若干文本

J //合并光标所在行及下一行为一行(依然在命令模式)