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如表2-2所示。

man man

表2-2                                            man命令中常用按键以及用途

|  |  |
| --- | --- |
| 按键 | 用处 |
| 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 |

一般来讲，使用man命令查看到的帮助内容信息都会很长很多，如果读者不了解帮助文档信息的目录结构和操作方法，乍一看到这么多信息可能会感到相当困惑。man命令的帮助信息的结构如表2-3所示。

表2-3                                         man命令帮助信息的结构以及意义

|  |  |
| --- | --- |
| 结构名称 | 代表意义 |
| 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 |

##### ****2.3 常用系统工作命令****

您现在阅读的这本书是刘遄老师在经历了十多年的运维学习以及数十期的培训授课后总结而成的，您可能无法在本节中找到某些之前见过的命令。但不用担心，之所以这样安排，原因是刘遄老师努力地将Linux命令与实战相结合，真正让读者在实操中理解技术，而不是单纯地把命令堆砌到书中让读者去硬背。

刘遄老师用了近一年的时间把最常用的Linux命令进行汇总、归纳、整理、分类后，把这些常用的命令合理安排到了后续章节中，然后采用以练代学的方式来加深读者的理解和掌握。从数年的培训成果反馈来看，这种方式相当有效，因此也相信这种方式肯定适合您的学习。

****1．echo命令****

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]”。

例如，把指定字符串“Linuxprobe.com”输出到终端屏幕的命令为：

[root@linuxprobe ~]# echo Linuxprobe.Com

该命令会在终端屏幕上显示如下信息：

Linuxprobe.Com

下面，我们使用$变量的方式提取变量SHELL的值，并将其输出到屏幕上：

[root@linuxprobe ~]# echo $SHELL

/bin/bash

****2．date命令****

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]”。

只需在强大的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，这样在日常工作时便可以把备份数据的命令与指定格式输出的时间信息结合到一起。例如，把打包后的文件自动按照“年-月-日”的格式打包成“backup-2017-9-1.tar.gz”，用户只需要看一眼文件名称就能大概了解到每个文件的备份时间了。date命令中常见的参数格式及作用如表2-4所示。

表2-4                                              date命令中的参数以及作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| %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 |

按照默认格式查看当前系统时间的date命令如下所示：

[root@linuxprobe ~]# date

Mon Aug 24 16:11:23 CST 2017

按照“年-月-日 小时:分钟:秒”的格式查看当前系统时间的date命令如下所示：

[root@linuxprobe ~]# date "+%Y-%m-%d %H:%M:%S"

2017-08-24 16:29:12

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

再次使用date命令并按照默认的格式查看当前的系统时间，如下所示：

[root@linuxprobe ~]# date

Fri Sep 1 08:30:01 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

****3．reboot命令****

reboot命令用于重启系统，其格式为reboot。

由于重启计算机这种操作会涉及硬件资源的管理权限，因此默认只能使用root管理员来重启，其命令如下：

[root@linuxprobe ~]# reboot

****4．poweroff命令****

poweroff命令用于关闭系统，其格式为poweroff。

该命令与reboot命令相同，都会涉及硬件资源的管理权限，因此默认只有root管理员才可以关闭电脑，其命令如下：

[root@linuxprobe ~]# poweroff

****5．wget命令****

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”。

如果您没有Linux系统的管理经验，当前只需了解一下wget命令的参数以及作用，然后看一下下面的演示实验即可，切记不要急于求成。后面章节将逐步讲解Linux系统的配置管理方法，可以等您掌握了网卡的配置方法后再来进行这个实验操作。表2-5所示为wget命令的参数以及参数的作用。

表2-5                                                wget命令的参数以及作用

|  |  |
| --- | --- |
| 参数 | 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 |

尝试使用wget命令从本书的配套站点中下载本书的最新pdf格式电子文档，这个文件的完整路径为<https://www.linuxprobe.com/docs/LinuxProbe.pdf>，执行该命令后的下载效果如下：

[root@linuxprobe ~]# wget https://www.linuxprobe.com/docs/LinuxProbe.pdf

--2017-08-24 19:30:12 -- https://www.linuxprobe.com/docs/LinuxProbe.pdf

Resolving www.linuxprobe.com (www.linuxprobe.com)... 220.181.105.185

Connecting to www.linuxprobe.com (www.linuxprobe.com)|220.181.105.185|:80...connected.

HTTP request sent, awaiting response... 200 OK

Length: 45948568 (44M) [application/pdf]

Saving to: ‘LinuxProbe.pdf’

100%[===========================================>] 45,948,568 32.9MB/s in 1.3s

2017-08-24 19:30:14 (32.9 MB/s) - ‘LinuxProbe.pdf’ saved [45948568/45948568]

接下来，我们使用wget命令递归下载[www.linuxprobe.com](https://www.linuxprobe.com/)网站内的所有页面数据以及文件，下载完后会自动保存到当前路径下一个名为www.linuxprobe.com的目录中。执行该操作的命令为wget -r -p [https://www.linuxprobe.com](https://www.linuxprobe.com/)，该命令的执行结果如下。

[root@linuxprobe ~]# wget -r -p https://www.linuxprobe.com

--2017-08-24 19:31:41-- https://www.linuxprobe.com/

Resolving www.linuxprobe.com... 106.185.25.197

Connecting to www.linuxprobe.com|106.185.25.197|:80... connected.

HTTP request sent, awaiting response... 200 OK

Length: unspecified [text/html]

Saving to: 'www.linuxprobe.com/index.html'

………………省略下载过程………………

****6．ps命令****

ps命令用于查看系统中的进程状态，格式为“ps [参数]”。

估计读者在第一次执行这个命令时都要惊呆一下—怎么会有这么多输出值，这可怎么看得过来？其实，刘遄老师通常会将ps命令与第3章的管道符技术搭配使用，用来抓取与某个指定服务进程相对应的PID号码。ps命令的常见参数以及作用如表2-6所示。

表2-6                                                  ps命令的参数以及作用

|  |  |
| --- | --- |
| 参数 | 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系统中时刻运行着许多进程，如果能够合理地管理它们，则可以优化系统的性能。在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，表2-7中只是列举了部分输出值，而且正常的输出值中不包括中文注释。

表2-7                                                             进程状态

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| USER | PID | %CPU | %MEM | VSZ | RSS | TTY | STAT | START | TIME | COMMAND |
| 进程的所有者 | 进程ID号 | 运算器占用率 | 内存占用率 | 虚拟内存使用量(单位是KB) | 占用的固定内存量(单位是KB) | 所在终端 | 进程状态 | 被启动的时间 | 实际使用CPU的时间 | 命令名称与参数 |
| root | 1 | 0.0 | 0.4 | 53684 | 7628 | ? | Ss | 07:22 | 0:02 | /usr/lib/systemd/systemd |
| root | 2 | 0.0 | 0.0 | 0 | 0 | ? | S | 07:22 | 0:00 | [kthreadd] |
| root | 3 | 0.0 | 0.0 | 0 | 0 | ? | S | 07:22 | 0:00 | [ksoftirqd/0] |
| root | 5 | 0.0 | 0.0 | 0 | 0 | ? | S< | 07:22 | 0:00 | [kworker/0:0H] |
| root | 7 | 0.0 | 0.0 | 0 | 0 | ? | S | 07:22 | 0:00 | [migration/0] |
|  | | | | ……………… | 省略部分输出信息 | ……………… | | | | |

如前面所提到的，在Linux系统中的命令参数有长短格式之分，长格式和长格式之间不能合并，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（-），因此可直接写成ps aux的样子。

****7．top命令****

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命令相当强大，能够动态地查看系统运维状态，完全将它看作Linux中的“强化版的Windows任务管理器”。top命令的运行界面如图2-6所示。

图2-6  top命令的运行界面

在图2-6中，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。其中数据均为CPU数据并以百分比格式显示，例如“97.1 id”意味着有97.1%的CPU处理器资源处于空闲。

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。

****8．pidof命令****

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

****9．kill命令****

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

****10．killall命令****

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]”。

通常来讲，复杂软件的服务程序会有多个进程协同为用户提供服务，如果逐个去结束这些进程会比较麻烦，此时可以使用killall命令来批量结束某个服务程序带有的全部进程。下面以httpd服务程序为例，来结束其全部进程。由于RHEL7系统默认没有安装httpd服务程序，因此大家此时只需看操作过程和输出结果即可，等学习了相关内容之后再来实践。

[root@linuxprobe ~]# pidof httpd

13581 13580 13579 13578 13577 13576

[root@linuxprobe ~]# killall httpd

[root@linuxprobe ~]# pidof httpd

[root@linuxprobe ~]#

如果我们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（生产环境中比较常用的一个快捷键），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一个&符号，这样命令将进入系统后台来执行。

刘遄老师主讲的Linux技术培训课程介绍：<https://www.linuxprobe.com/training>

##### ****2.4 系统状态检测命令****

作为一名合格的运维人员，想要更快、更好地了解Linux服务器，必须具备快速查看Linux系统运行状态的能力，因此接下来会逐个讲解与网卡网络、系统内核、系统负载、内存使用情况、当前启用终端数量、历史登录记录、命令执行记录以及救援诊断等相关命令的使用方法。这些命令都超级实用，还请读者用心学习，加以掌握。

****1．ifconfig命令****

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）：

[root@linuxprobe ~]# ifconfig****eno16777728****: flags=4163  mtu 1500

****inet 192.168.10.10****  netmask 255.255.255.0  broadcast 192.168.10.255

        inet6 fe80::20c:29ff:fec4:a409  prefixlen 64  scopeid 0x20

****ether 00:0c:29:c4:a4:09****  txqueuelen 1000  (Ethernet)

****RX packets 36  bytes 3176 (3.1 KiB)****

        RX errors 0  dropped 0  overruns 0  frame 0

****TX packets 38  bytes 4757 (4.6 KiB)****

        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73  mtu 65536

        inet 127.0.0.1  netmask 255.0.0.0

        inet6 ::1  prefixlen 128  scopeid 0x10

        loop  txqueuelen 0  (Local Loopback)

        RX packets 386  bytes 32780 (32.0 KiB)

        RX errors 0  dropped 0  overruns 0  frame 0

        TX packets 386  bytes 32780 (32.0 KiB)

        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

****2．uname命令****

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。

[root@linuxprobe ~]# uname -a

Linux linuxprobe.com 3.10.0-123.el7.x86\_64 #1 SMP Mon May 5 11:16:57 EDT 2017 x86\_64 x86\_64 x86\_64 GNU/Linux

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)

****3．uptime命令****

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过5。

[root@linuxprobe ~]# uptime

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

****4．free命令****

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。表2-8所示为在刘遄老师的电脑上执行free -h命令之后的输出信息。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

表2-8                                            执行free -h命令后的输出信息

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 内存总量 | 已用量 | 可用量 | 进程共享的内存量 | 磁盘缓存的内存量 | 缓存的内存量 |
|  | total | used | free | shared | buffers | cached |
| Mem: | 1.8G | 1.3G | 542M | 9.8M | 1.6M | 413M |
| -/+ buffers/cache: |  | 869M | 957M |  |  |  |
| Swap: | 2.0G | 0B | 2.0G |  |  |  |

****5．who命令****

who用于查看当前登入主机的用户终端信息，格式为“who [参数]”。

这三个简单的字母可以快速显示出所有正在登录本机的用户的名称以及他们正在开启的终端信息。表2-9所示为执行who命令后的结果。

[root@linuxprobe ~]# who

表2-9                                                   执行who命令的结果

|  |  |  |
| --- | --- | --- |
| 登陆的用户名 | 终端设备 | 登陆到系统的时间 |
| root | :0 | 2017-08-24 17:52 (:0) |
| root | pts/0 | 2017-08-24 17:52 (:0) |

****6．last命令****

last命令用于查看所有系统的登录记录，格式为“last [参数]”。

使用last命令可以查看本机的登录记录。但是，由于这些信息都是以日志文件的形式保存在系统中，因此黑客可以很容易地对内容进行篡改。千万不要单纯以该命令的输出信息而判断系统有无被恶意入侵！

[root@linuxprobe ~]# last

root       pts/0       :0               Mon Aug  24 17:52   still  logged in

root       :0          :0               Mon Aug  24 17:52   still  logged in

(unknown   :0          :0               Mon Aug  24 17:50 - 17:52   (00:02)

reboot     system boot 3.10.0-123.el7.x Tue Aug  25 01:49 - 18:17   (-7:-32)

root       pts/0       :0               Mon Aug  24 15:40 - 08:54   (7+17:14)

root       pts/0       :0               Fri Jul  10 10:49 - 15:37   (45+04:47)

………………省略部分登录信息………………

****7．history命令****

history命令用于显示历史执行过的命令，格式为“history [-c]”。

history命令应该是作者最喜欢的命令。执行history命令能显示出当前用户在本地计算机中执行过的最近1000条命令记录。如果觉得1000不够用，还可以自定义/etc/profile文件中的HISTSIZE变量值。在使用history命令时，如果使用-c参数则会清空所有的命令历史记录。还可以使用“!编码数字”的方式来重复执行某一次的命令。总之，history命令有很多有趣的玩法等待您去开发。

[root@linuxprobe ~]# history

1 tar xzvf VMwareTools-9.9.0-2304977.tar.gz

2 cd vmware-tools-distrib/

3 ls

4 ./vmware-install.pl -d

5 reboot

6 df -h

7 cd /run/media/

8 ls

9 cd root/

10 ls

11 cd VMware\ Tools/

12 ls

13 cp VMwareTools-9.9.0-2304977.tar.gz /home

14 cd /home****15 ls****

16 tar xzvf VMwareTools-9.9.0-2304977.tar.gz

17 cd vmware-tools-distrib/

18 ls

19 ./vmware-install.pl -d

20 reboot

21 history

[root@linuxprobe ~]# !15

anaconda-ks.cfg  Documents  initial-setup-ks.cfg  Pictures  Templates

Desktop          Downloads  Music                 Public    Videos

历史命令会被保存到用户家目录中的.bash\_history文件中。Linux系统中以点（.）开头的文件均代表隐藏文件，这些文件大多数为系统服务文件，可以用cat命令查看其文件内容。

[root@linuxprobe ~]# cat ~/.bash\_history

要清空当前用户在本机上执行的Linux命令历史记录信息，可执行如下命令：

[root@linuxprobe ~]# history -c

****8．sosreport命令****

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. ****此处敲击回车来确认收集信息****

Please enter your first initial and last name [linuxprobe.com]:****此处敲击回车来确认主机编号****

Please enter the case number that you are generating this report for:****此处敲击回车来确认主机编号****

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.xzThe checksum is: 79436cdf791327040efde48c452c6322****

Please send this file to your support representative.

##### ****2.5 工作目录切换命令****

工作目录指的是用户当前在系统中所处的位置。由于工作目录会牵涉系统存储结构相关的知识，因此第6章将详细讲解这部分内容。读者只需简单了解一下这里的操作实验即可，如果不能完全掌握也没有关系，毕竟Linux系统的知识体系太过庞大，每一位初学人员都需要经历这么一段时期。

****1．pwd命令****

pwd命令用于显示用户当前所处的工作目录，格式为“pwd [选项]”。

[root@linuxprobe etc]# pwd

/etc

****2．cd命令****

cd命令用于切换工作路径，格式为“cd [目录名称]”。

这个命令应该是最常用的一个Linux命令了。可以通过cd命令迅速、灵活地切换到不同的工作目录。除了常见的切换目录方式，还可以使用“cd -”命令返回到上一次所处的目录，使用“cd..”命令进入上级目录，以及使用“cd ~”命令切换到当前用户的家目录，亦或使用“cd ~username”切换到其他用户的家目录。例如，可以使用“cd 路径”的方式切换进/etc目录中：

[root@linuxprobe ~]# cd /etc

同样的道理，可使用下述命令切换到/bin目录中：

[root@linuxprobe etc]# cd /bin

此时，要返回到上一次的目录（即/etc目录），可执行如下命令：

[root@linuxprobe bin]# cd -

/etc

[root@linuxprobe etc]#

还可以通过下面的命令快速切换到用户的家目录：

[root@linuxprobe etc]# cd ~

[root@linuxprobe ~]#

****3．ls命令****

ls命令用于显示目录中的文件信息，格式为“ls [选项] [文件] ”。

所处的工作目录不同，当前工作目录下的文件肯定也不同。使用ls命令的“-a”参数看到全部文件（包括隐藏文件），使用“-l”参数可以查看文件的属性、大小等详细信息。将这两个参数整合之后，再执行ls命令即可查看当前目录中的所有文件并输出这些文件的属性信息：

[root@linuxprobe ~]# ls -al

total 60

dr-xr-x---. 14 root root 4096 May  4 07:56 .

drwxr-xr-x. 17 root root 4096 May  4 15:55 ..

-rw-------.  1 root root 1213 May  4 15:44 anaconda-ks.cfg

-rw-------.  1 root root  957 May  4 07:54 .bash\_history

-rw-r--r--.  1 root root   18 Dec 28  2013 .bash\_logout

-rw-r--r--.  1 root root  176 Dec 28  2013 .bash\_profile

-rw-r--r--.  1 root root  176 Dec 28  2013 .bashrc

drwx------. 10 root root 4096 May  4 07:56 .cache

drwx------. 15 root root 4096 May  4 07:49 .config

-rw-r--r--.  1 root root  100 Dec 28  2013 .cshrc

drwx------.  3 root root   24 May  4 07:46 .dbus

drwxr-xr-x.  2 root root    6 May  4 07:49 Desktop

drwxr-xr-x.  2 root root    6 May  4 07:49 Documents

drwxr-xr-x.  2 root root    6 May  4 07:49 Downloads

-rw-------.  1 root root   16 May  4 07:49 .esd\_auth

-rw-------.  1 root root  628 May  4 07:56 .ICEauthority

-rw-r--r--.  1 root root 1264 May  4 07:48 initial-setup-ks.cfg

drwx------.  3 root root   18 May  4 07:49 .local

drwxr-xr-x.  2 root root    6 May  4 07:49 Music

drwxr-xr-x.  2 root root    6 May  4 07:49 Pictures

drwxr-xr-x.  2 root root    6 May  4 07:49 Public

-rw-r--r--.  1 root root  129 Dec 28  2013 .tcshrc

drwxr-xr-x.  2 root root    6 May  4 07:49 Templates

drwxr-xr-x.  2 root root    6 May  4 07:49 Videos

-rw-------.  1 root root 1962 May  4 07:54 .viminfo

如果想要查看目录属性信息，则需要额外添加一个-d参数。例如，可使用如下命令查看/etc目录的权限与属性信息：

[root@linuxprobe ~]# ls -ld /etc

drwxr-xr-x. 132 root root 8192 Jul 10 10:48 /etc

##### ****2.6 文本文件编辑命令****

通过前面几个小节的学习，读者应该基本掌握了切换工作目录及对文件的管理方法。Linux系统中“一切都是文件”，而对服务程序进行配置自然也就是编辑程序的配置文件。如果不能熟练地查阅系统或服务的配置文件，那以后工作时可就真的要尴尬了。本节将讲解几条用于查看文本文件内容的命令。至于编辑器使用起来比较复杂，因此将放到第4章与Shell脚本内容一起讲解。

****1．cat命令****

cat命令用于查看纯文本文件（内容较少的），格式为“cat [选项] [文件]”。

Linux系统中有多个用于查看文本内容的命令，每个命令都有自己的特点，比如这个cat命令就是用于查看内容较少的纯文本文件的。cat这个命令也很好记，因为cat在英语中是“猫”的意思，小猫咪是不是给您一种娇小、可爱的感觉呢？

如果在查看文本内容时还想顺便显示行号的话，不妨在cat命令后面追加一个-n参数：

[root@linuxprobe ~]# cat -n initial-setup-ks.cfg

     1    #version=RHEL7

     2    # X Window System configuration information

     3    xconfig  --startxonboot

     4

     5    # License agreement

     6    eula --agreed

     7    # System authorization information

     8    auth --enableshadow --passalgo=sha512

     9    # Use CDROM installation media

    10    cdrom

    11    # Run the Setup Agent on first boot

    12    firstboot --enable

    13    # Keyboard layouts

    14    keyboard --vckeymap=us --xlayouts='us'

    15    # System language

    16    lang en\_US.UTF-8

………………省略部分输出信息………………

cat命令详解：<https://www.linuxcool.com/cat>

****2．more命令****

more命令用于查看纯文本文件（内容较多的），格式为“more [选项]文件”。

因为一旦使用cat命令阅读长篇的文本内容，信息就会在屏幕上快速翻滚，导致自己还没有来得及看到，内容就已经翻篇了。因此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：

[root@linuxprobe ~]# more initial-setup-ks.cfg

#version=RHEL7

# X Window System configuration information

xconfig  --startxonboot

# License agreement

eula --agreed

# System authorization information

auth --enableshadow --passalgo=sha512

# Use CDROM installation media

cdrom

# Run the Setup Agent on first boot

firstboot --enable

# Keyboard layouts

keyboard --vckeymap=us --xlayouts='us'

# System language

lang en\_US.UTF-8

ignoredisk --only-use=sda

# Network information

network  --bootproto=dhcp --device=eno16777728 --onboot=off --ipv6=auto

network  --bootproto=dhcp --hostname=linuxprobe.com--More--(43%)

****3．head命令****

head命令用于查看纯文本文档的前N行，格式为“head [选项] [文件]”。

在阅读文本内容时，谁也难以保证会按照从头到尾的顺序往下看完整个文件。如果只想查看文本中前20行的内容，该怎么办呢？head命令可以派上用场了：

[root@linuxprobe ~]# head -n 20  initial-setup-ks.cfg

#version=RHEL7

# X Window System configuration information

xconfig  --startxonboot

# License agreement

eula --agreed

# System authorization information

auth --enableshadow --passalgo=sha512

# Use CDROM installation media

cdrom

# Run the Setup Agent on first boot

firstboot --enable

# Keyboard layouts

keyboard --vckeymap=us --xlayouts='us'

# System language

lang en\_US.UTF-8

ignoredisk --only-use=sda

# Network information

network  --bootproto=dhcp --device=eno16777728 --onboot=off --ipv6=auto

[root@linuxprobe ~]#

****4．tail命令****

tail命令用于查看纯文本文档的后N行或持续刷新内容，格式为“tail [选项] [文件]”。

我们可能还会遇到另外一种情况，比如需要查看文本内容的最后20行，这时就需要用到tail命令了。tail命令的操作方法与head命令非常相似，只需要执行“tail -n 20 文件名”命令就可以达到这样的效果。tail命令最强悍的功能是可以持续刷新一个文件的内容，当想要实时查看最新日志文件时，这特别有用，此时的命令格式为“tail -f 文件名”：

[root@linuxprobe ~]# tail -f /var/log/messages

May  4 07:56:38 localhost gnome-session: Window manager warning: Log level 16: STACK\_OP\_ADD: window 0x1e00001 already in stack

May  4 07:56:38 localhost gnome-session: Window manager warning: Log level 16: STACK\_OP\_ADD: window 0x1e00001 already in stack

May  4 07:56:38 localhost vmusr[12982]: [ warning] [Gtk] gtk\_disable\_setlocale() must be called before gtk\_init()

May  4 07:56:50 localhost systemd-logind: Removed session c1.

Aug  1 01:05:31 localhost systemd: Time has been changed

Aug  1 01:05:31 localhost systemd: Started LSB: Bring up/down networking.

Aug  1 01:08:56 localhost dbus-daemon: dbus[1124]: [system] Activating service name='com.redhat.SubscriptionManager' (using servicehelper)

Aug  1 01:08:56 localhost dbus[1124]: [system] Activating service name='com.redhat.SubscriptionManager' (using servicehelper)

Aug  1 01:08:57 localhost dbus-daemon: dbus[1124]: [system] Successfully activated service 'com.redhat.SubscriptionManager'

Aug  1 01:08:57 localhost dbus[1124]: [system] Successfully activated service 'com.redhat.SubscriptionManager'

****5．tr命令****

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]

#VERSION=RHEL7

# SYSTEM AUTHORIZATION INFORMATION

AUTH --ENABLESHADOW --PASSALGO=SHA512

# USE CDROM INSTALLATION MEDIA

CDROM

# RUN THE SETUP AGENT ON FIRST BOOT

%END

****6．wc命令****

wc命令用于统计指定文本的行数、字数、字节数，格式为“wc [参数] 文本”。

每次我在课堂上讲到这个命令时，总有同学会联想到一种公共设施，其实这两者毫无关联。Linux系统中的wc命令用于统计文本的行数、字数、字节数等。如果为了方便自己记住这个命令的作用，也可以联想到上厕所时好无聊，无聊到数完了手中的如厕读物上有多少行字。wc的参数以及相应的作用如表2-10所示。

表2-10                                                   wc的参数以及作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| -l | 只显示行数 |
| -w | 只显示单词数 |
| -c | 只显示字节数 |

在Linux系统中，passwd是用于保存系统账户信息的文件，要统计当前系统中有多少个用户，可以使用下面的命令来进行查询，是不是很神奇：

[root@linuxprobe ~]# wc -l /etc/passwd

38 /etc/passwd

****7．stat命令****

stat命令用于查看文件的具体存储信息和时间等信息，格式为“stat 文件名称”。

stat命令可以用于查看文件的存储信息和时间等信息，命令stat anaconda-ks.cfg会显示出文件的三种时间状态（已加粗）：Access、Modify、Change。这三种时间的区别将在下面的touch命令中详细详解：

[root@linuxprobe ~]# stat anaconda-ks.cfg

File: ‘anaconda-ks.cfg’

Size: 1213 Blocks: 8 IO Block: 4096 regular file

Device: fd00h/64768d Inode: 68912908 Links: 1

Access: (0600/-rw-------) Uid: ( 0/ root) Gid: ( 0/ root)

Context: system\_u:object\_r:admin\_home\_t:s0****Access****: 2017-07-14 01:46:18.721255659 -0400****Modify****: 2017-05-04 15:44:36.916027026 -0400****Change****: 2017-05-04 15:44:36.916027026 -0400

Birth: -

****8．cut命令****

cut命令用于按“列”提取文本字符，格式为“cut [参数] 文本”。

在Linux系统中，如何准确地提取出最想要的数据，这也是我们应该重点学习的内容。一般而言，按基于“行”的方式来提取数据是比较简单的，只需要设置好要搜索的关键词即可。但是如果按列搜索，不仅要使用-f参数来设置需要看的列数，还需要使用-d参数来设置间隔符号。passwd在保存用户数据信息时，用户信息的每一项值之间是采用冒号来间隔的，接下来我们使用下述命令尝试提取出passwd文件中的用户名信息，即提取以冒号（：）为间隔符号的第一列内容：

[root@linuxprobe ~]# head -n 2 /etc/passwd****root****:x:0:0:root:/root:/bin/bash****bin****:x:1:1:bin:/bin:/sbin/nologin

[root@linuxprobe ~]# cut -d: -f1 /etc/passwd

root

bin

daemon

adm

lp

sync

shutdown

halt

mail

operator

games

ftp

nobody

dbus

polkitd

unbound

colord

usbmuxd

avahi

avahi-autoipd

libstoragemgmt

saslauth

qemu

rpc

rpcuser

nfsnobody

rtkit

radvd

ntp

chrony

abrt

pulse

gdm

gnome-initial-setup

postfix

sshd

tcpdump

linuxprobe

****9．diff命令****

diff命令用于比较多个文本文件的差异，格式为“diff [参数] 文件”。

在使用diff命令时，不仅可以使用--brief参数来确认两个文件是否不同，还可以使用-c参数来详细比较出多个文件的差异之处，这绝对是判断文件是否被篡改的有力神器。例如，先使用cat命令分别查看diff\_A.txt和diff\_B.txt文件的内容，然后进行比较：

[root@linuxprobe ~]# cat diff\_A.txt

Welcome to linuxprobe.com

Red Hat certified

Free Linux Lessons

Professional guidance

Linux Course

[root@linuxprobe ~]# cat diff\_B.txt

Welcome tooo linuxprobe.com

Red Hat certified

Free Linux LeSSonS

////////.....////////

Professional guidance

Linux Course

接下来使用diff --brief命令显示比较后的结果，判断文件是否相同：

[root@linuxprobe ~]# diff --brief diff\_A.txt diff\_B.txt

Files diff\_A.txt and diff\_B.txt differ

最后使用带有-c参数的diff命令来描述文件内容具体的不同：

[root@linuxprobe ~]# diff -c diff\_A.txt diff\_B.txt

\*\*\* diff\_A.txt 2017-08-30 18:07:45.230864626 +0800

--- diff\_B.txt 2017-08-30 18:08:52.203860389 +0800

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* 1,5 \*\*\*\*

! Welcome to linuxprobe.com

Red Hat certified

! Free Linux Lessons

Professional guidance

Linux Course

--- 1,7 ----

! Welcome tooo linuxprobe.com

!

Red Hat certified

! Free Linux LeSSonS

! ////////.....////////

Professional guidance

Linux Course

##### ****2.7 文件目录管理命令****

目前为止，我们学习Linux命令就像是在夯实地基，虽然表面上暂时还看不到成果，但其实大家的内功已经相当雄厚了。在Linux系统的日常运维工作中，还需要掌握对文件的创建、修改、复制、剪切、更名与删除等操作。

****1．touch命令****

touch命令用于创建空白文件或设置文件的时间，格式为“touch [选项] [文件]”。

在创建空白的文本文件方面，这个touch命令相当简捷，简捷到没有必要铺开去讲。比如，touch linuxprobe命令可以创建出一个名为linuxprobe的空白文本文件。对touch命令来讲，有难度的操作主要是体现在设置文件内容的修改时间（mtime）、文件权限或属性的更改时间（ctime）与文件的读取时间（atime）上面。touch命令的参数及其作用如表2-11所示。

表2-11                                             touch命令的参数及其作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| -a | 仅修改“读取时间”（atime） |
| -m | 仅修改“修改时间”（mtime） |
| -d | 同时修改atime与mtime |

接下来，我们先使用ls命令查看一个文件的修改时间，然后修改这个文件，最后再通过touch命令把修改后的文件时间设置成修改之前的时间（很多黑客就是这样做的呢）：

[root@linuxprobe ~]# ls -l anaconda-ks.cfg

-rw-------. 1 root root 1213 ****May  4 15:44**** anaconda-ks.cfg

[root@linuxprobe ~]# echo "Visit the LinuxProbe.com to learn linux skills" >> anaconda-ks.cfg

[root@linuxprobe ~]# ls -l anaconda-ks.cfg

-rw-------. 1 root root 1260 ****Aug  2 01:26**** anaconda-ks.cfg

[root@linuxprobe ~]# touch -d "2017-05-04 15:44" anaconda-ks.cfg

[root@linuxprobe ~]# ls -l anaconda-ks.cfg

-rw-------. 1 root root 1260 ****May  4 15:44**** anaconda-ks.cfg

****2．mkdir命令****

mkdir命令用于创建空白的目录，格式为“mkdir [选项] 目录”。

在Linux系统中，文件夹是最常见的文件类型之一。除了能创建单个空白目录外，mkdir命令还可以结合-p参数来递归创建出具有嵌套叠层关系的文件目录。

[root@linuxprobe ~]# mkdir linuxprobe

[root@linuxprobe ~]# cd linuxprobe

[root@linuxprobe linuxprobe]# mkdir -p a/b/c/d/e

[root@linuxprobe linuxprobe]# cd a

[root@linuxprobe a]# cd b

[root@linuxprobe b]#

****3．cp命令****

cp命令用于复制文件或目录，格式为“cp [选项] 源文件 目标文件”。大家对文件复制操作应该不陌生，在Linux系统中，复制操作具体分为3种情况：

如果目标文件是目录，则会把源文件复制到该目录中；

如果目标文件也是普通文件，则会询问是否要覆盖它；

如果目标文件不存在，则执行正常的复制操作。

cp命令的参数及其作用如表2-12所示。

表2-12                                               cp命令的参数及其作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| -p | 保留原始文件的属性 |
| -d | 若对象为“链接文件”，则保留该“链接文件”的属性 |
| -r | 递归持续复制（用于目录） |
| -i | 若目标文件存在则询问是否覆盖 |
| -a | 相当于-pdr（p、d、r为上述参数） |

接下来，使用touch创建一个名为install.log的普通空白文件，然后将其复制为一份名为x.log的备份文件，最后再使用ls命令查看目录中的文件：

[root@linuxprobe ~]# touch install.log

[root@linuxprobe ~]# cp install.log x.log

[root@linuxprobe ~]# ls

install.log x.log

****4．mv命令****

mv命令用于剪切文件或将文件重命名，格式为“mv [选项] 源文件 [目标路径|目标文件名]”。

剪切操作不同于复制操作，因为它会默认把源文件删除掉，只保留剪切后的文件。如果在同一个目录中对一个文件进行剪切操作，其实也就是对其进行重命名：

[root@linuxprobe ~]# mv x.log linux.log

[root@linuxprobe ~]# ls

install.log linux.log

****5．rm命令****

rm命令用于删除文件或目录，格式为“rm [选项] 文件”。

在Linux系统中删除文件时，系统会默认向您询问是否要执行删除操作，如果不想总是看到这种反复的确认信息，可在rm命令后跟上-f参数来强制删除。另外，想要删除一个目录，需要在rm命令后面加一个-r参数才可以，否则删除不掉。我们来尝试删除前面创建的install.log和linux.log文件：

[root@linuxprobe ~]# rm install.log

rm: remove regular empty file ‘install.log’? y

[root@linuxprobe ~]# rm -f linux.log

[root@linuxprobe ~]# ls

[root@linuxprobe ~]#

****6．dd命令****

dd命令用于按照指定大小和个数的数据块来复制文件或转换文件，格式为“dd [参数]”。

dd命令是一个比较重要而且比较有特色的一个命令，它能够让用户按照指定大小和个数的数据块来复制文件的内容。当然如果愿意的话，还可以在复制过程中转换其中的数据。Linux系统中有一个名为/dev/zero的设备文件，每次在课堂上解释它时都充满哲学理论的色彩。因为这个文件不会占用系统存储空间，但却可以提供无穷无尽的数据，因此可以使用它作为dd命令的输入文件，来生成一个指定大小的文件。dd命令的参数及其作用如表2-13所示。

表2-13                                               dd命令的参数及其作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| if | 输入的文件名称 |
| of | 输出的文件名称 |
| bs | 设置每个“块”的大小 |
| count | 设置要复制“块”的个数 |

例如我们可以用dd命令从/dev/zero设备文件中取出一个大小为560MB的数据块，然后保存成名为560\_file的文件。在理解了这个命令后，以后就能随意创建任意大小的文件了：

[root@linuxprobe ~]# dd if=/dev/zero of=560\_file count=1 bs=560M

1+0 records in

1+0 records out

587202560 bytes (587 MB) copied, 27.1755 s, 21.6 MB/s

dd命令的功能也绝不仅限于复制文件这么简单。如果您想把光驱设备中的光盘制作成iso格式的镜像文件，在Windows系统中需要借助于第三方软件才能做到，但在Linux系统中可以直接使用dd命令来压制出光盘镜像文件，将它变成一个可立即使用的iso镜像：

[root@linuxprobe ~]# dd if=/dev/cdrom of=RHEL-server-7.0-x86\_64-LinuxProbe.Com.iso

7311360+0 records in

7311360+0 records out

3743416320 bytes (3.7 GB) copied, 370.758 s, 10.1 MB/s

考虑到有些读者会纠结bs块大小与count块个数的关系，下面举一个吃货的例子进行解释。假设小明的饭量（即需求）是一个固定的值，用来盛饭的勺子的大小即bs块大小，而用勺子盛饭的次数即count块个数。小明要想吃饱（满足需求），则需要在勺子大小（bs块大小）与用勺子盛饭的次数（count块个数）之间进行平衡。勺子越大，用勺子盛饭的次数就越少。由上可见，bs与count都是用来指定容量的大小，只要能满足需求，可随意组合搭配方式。

****7．file命令****

file命令用于查看文件的类型，格式为“file 文件名”。

在Linux系统中，由于文本、目录、设备等所有这些一切都统称为文件，而我们又不能单凭后缀就知道具体的文件类型，这时就需要使用file命令来查看文件类型了。

[root@linuxprobe ~]# file anaconda-ks.cfg

anaconda-ks.cfg: ASCII text

[root@linuxprobe ~]# file /dev/sda

/dev/sda: block special

##### ****2.8 打包压缩与搜索命令****

在网络上，人们越来越倾向于传输压缩格式的文件，原因是压缩文件体积小，在网速相同的情况下，传输时间短。下面将学习如何在Linux系统中对文件进行打包压缩与解压，以及让用户基于关键词在文本文件中搜索相匹配的信息、在整个文件系统中基于指定的名称或属性搜索特定文件。本节虽然只有3条命令，但是其功能都比较复杂而且参数很多，因此放到了本章最后讲解。

****1．tar命令****

tar命令用于对文件进行打包压缩或解压，格式为“tar [选项] [文件]”。

在Linux系统中，常见的文件格式比较多，其中主要使用的是.tar或.tar.gz或.tar.bz2格式，我们不用担心格式太多而记不住，其实这些格式大部分都是由tar命令来生成的。刘遄老师将讲解最重要的几个参数，以方便大家理解。tar命令的参数及其作用如表2-14所示。

表2-14                                               tar命令的参数及其作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| -c | 创建压缩文件 |
| -x | 解开压缩文件 |
| -t | 查看压缩包内有哪些文件 |
| -z | 用Gzip压缩或解压 |
| -j | 用bzip2压缩或解压 |
| -v | 显示压缩或解压的过程 |
| -f | 目标文件名 |
| -p | 保留原始的权限与属性 |
| -P | 使用绝对路径来压缩 |
| -C | 指定解压到的目录 |

首先，-c参数用于创建压缩文件，-x参数用于解压文件，因此这两个参数不能同时使用。其次，-z参数指定使用Gzip格式来压缩或解压文件，-j参数指定使用bzip2格式来压缩或解压文件。用户使用时则是根据文件的后缀来决定应使用何种格式参数进行解压。在执行某些压缩或解压操作时，可能需要花费数个小时，如果屏幕一直没有输出，您一方面不好判断打包的进度情况，另一方面也会怀疑电脑死机了，因此非常推荐使用-v参数向用户不断显示压缩或解压的过程。-C参数用于指定要解压到哪个指定的目录。-f参数特别重要，它必须放到参数的最后一位，代表要压缩或解压的软件包名称。刘遄老师一般使用“tar -czvf 压缩包名称.tar.gz 要打包的目录”命令把指定的文件进行打包压缩；相应的解压命令为“tar -xzvf 压缩包名称.tar.gz”。下面我们来逐个演示下打包压缩与解压的操作。先使用tar命令把/etc目录通过gzip格式进行打包压缩，并把文件命名为etc.tar.gz：

[root@linuxprobe ~]# tar czvf etc.tar.gz /etc

tar: Removing leading `/' from member names

/etc/

/etc/fstab

/etc/crypttab

/etc/mtab

/etc/fonts/

/etc/fonts/conf.d/

/etc/fonts/conf.d/65-0-madan.conf

/etc/fonts/conf.d/59-liberation-sans.conf

/etc/fonts/conf.d/90-ttf-arphic-uming-embolden.conf

/etc/fonts/conf.d/59-liberation-mono.conf

/etc/fonts/conf.d/66-sil-nuosu.conf

………………省略部分压缩过程信息………………

接下来将打包后的压缩包文件指定解压到/root/etc目录中（先使用mkdir命令来创建/root/etc目录）：

[root@linuxprobe ~]# mkdir /root/etc

[root@linuxprobe ~]# tar xzvf etc.tar.gz -C /root/etc

etc/

etc/fstab

etc/crypttab

etc/mtab

etc/fonts/

etc/fonts/conf.d/

etc/fonts/conf.d/20-unhint-small-dejavu-sans.conf

………………省略部分解压过程信息………………

****2．grep命令****

grep命令用于在文本中执行关键词搜索，并显示匹配的结果，格式为“grep [选项] [文件]”。grep命令的参数及其作用如表2-15所示。

表2-15                                              grep命令的参数及其作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| -b | 将可执行文件(binary)当作文本文件（text）来搜索 |
| -c | 仅显示找到的行数 |
| -i | 忽略大小写 |
| -n | 显示行号 |
| -v | 反向选择——仅列出没有“关键词”的行。 |

。我们在这里只讲两个最最常用的参数：-n参数用来显示搜索到信息的行号；-v参数用于反选信息（即没有包含关键词的所有信息行）。这两个参数几乎能完成您日后80%的工作需要，至于其他上百个参数，即使以后在工作期间遇到了，再使用man grep命令查询也来得及。

在Linux系统中，/etc/passwd文件是保存着所有的用户信息，而一旦用户的登录终端被设置成/sbin/nologin，则不再允许登录系统，因此可以使用grep命令来查找出当前系统中不允许登录系统的所有用户信息：

[root@linuxprobe ~]# grep /sbin/nologin /etc/passwd

bin:x:1:1:bin:/bin:/sbin/nologin

daemon:x:2:2:daemon:/sbin:/sbin/nologin

adm:x:3:4:adm:/var/adm:/sbin/nologin

lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin

mail:x:8:12:mail:/var/spool/mail:/sbin/nologin

operator:x:11:0:operator:/root:/sbin/nologin

………………省略部分输出过程信息………………

****3．find命令****

find命令用于按照指定条件来查找文件，格式为“find [查找路径] 寻找条件 操作”。

本书中曾经多次提到“Linux系统中的一切都是文件”，接下来就要见证这句话的分量了。在Linux系统中，搜索工作一般都是通过find命令来完成的，它可以使用不同的文件特性作为寻找条件（如文件名、大小、修改时间、权限等信息），一旦匹配成功则默认将信息显示到屏幕上。find命令的参数以及作用如表2-16所示。

表2-16                                             find命令中的参数以及作用

|  |  |
| --- | --- |
| 参数 | 作用 |
| -name | 匹配名称 |
| -perm | 匹配权限（mode为完全匹配，-mode为包含即可） |
| -user | 匹配所有者 |
| -group | 匹配所有组 |
| -mtime -n +n | 匹配修改内容的时间（-n指n天以内，+n指n天以前） |
| -atime -n +n | 匹配访问文件的时间（-n指n天以内，+n指n天以前） |
| -ctime -n +n | 匹配修改文件权限的时间（-n指n天以内，+n指n天以前） |
| -nouser | 匹配无所有者的文件 |
| -nogroup | 匹配无所有组的文件 |
| -newer f1 !f2 | 匹配比文件f1新但比f2旧的文件 |
| --type b/d/c/p/l/f | 匹配文件类型（后面的字幕字母依次表示块设备、目录、字符设备、管道、链接文件、文本文件） |
| -size | 匹配文件的大小（+50KB为查找超过50KB的文件，而-50KB为查找小于50KB的文件） |
| -prune | 忽略某个目录 |
| -exec …… {}\; | 后面可跟用于进一步处理搜索结果的命令（下文会有演示） |

这里需要重点讲解一下-exec参数重要的作用。这个参数用于把find命令搜索到的结果交由紧随其后的命令作进一步处理，它十分类似于第3章将要讲解的管道符技术，并且由于find命令对参数的特殊要求，因此虽然exec是长格式形式，但依然只需要一个减号（-）。

根据文件系统层次标准（Filesystem Hierarchy Standard）协议，Linux系统中的配置文件会保存到/etc目录中（详见第6章）。如果要想获取到该目录中所有以host开头的文件列表，可以执行如下命令：

[root@linuxprobe ~]# find /etc -name "host\*" -print

/etc/avahi/hosts

/etc/host.conf

/etc/hosts

/etc/hosts.allow

/etc/hosts.deny

/etc/selinux/targeted/modules/active/modules/hostname.pp

/etc/hostname

如果要在整个系统中搜索权限中包括SUID权限的所有文件（详见第5章），只需使用-4000即可：

[root@linuxprobe ~]# find / -perm -4000 -print

/usr/bin/fusermount

/usr/bin/su

/usr/bin/umount

/usr/bin/passwd

/usr/sbin/userhelper

/usr/sbin/usernetctl

………………省略部分输出信息………………