

linux 常用命令学习总结

黄建春

(渝石网络 <http://www.fishsting.cn> 中国重庆)

作为一个程序员，我们在进行项目部署和运维时，经常会用到一些 linux 命令，可是这些命令老是忘记，每次用到的时候都要去谷歌百度，很是麻烦，因此，为了自己使用方便，对常见的 linux 命令进行了系统的总结，以便在用到时能够快速找到相关命令，同时有需要的朋友也可以参考收藏该篇文章，文章对讲到的每个命令都有详细的参数解释，并且给出一些常用例子，因此也非常适合用来学习。

全文命令分为以下 6 个模块：

- 1、常用系统工作命令
- 2、系统状态检测命令
- 3、工作目录切换命令
- 4、文本文件编辑命令
- 5、文件目录管理命令
- 6、打包压缩与搜索命令

在说具体命令之前，读者应该清楚 linux 命令的基本格式，如：命令名称 [命令参数] [命令对象]

注意：命令名称、命令参数、命令对象之间用空格键分隔。

以下内容为具体的命令详解：

一、常用系统工作命令

1、echo 命令

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
```

<https://blog.csdn.net/wanghuiwei888>

2、date 命令

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

<https://blog.csdn.net/wanghuiwei888>

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
```

<https://blog.csdn.net/wanghuiwei888>

3、reboot 命令（相当于 windows 的重启）

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（相当于 windows 的关机）

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”。

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

<https://blog.csdn.net/wanghuiwei888>

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]”。

常用用法：ps aux 或者 ps lax

注：

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。

<https://blog.csdn.net/wanghuiwei888>

7、top 命令（相当于 window 中的任务管理器）

top 命令用于动态地监视进程活动与系统负载等信息，默认每 10 秒刷新一次。

```

root@linuxprobe:~/Desktop
File Edit View Search Terminal Help
top - 19:19:51 up 11 min, 2 users, load average: 0.01, 0.17, 0.19
Tasks: 519 total, 2 running, 517 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.1 us, 0.6 sy, 0.0 ni, 97.1 id, 1.1 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 3869044 total, 900604 used, 2968440 free, 924 buffers
KiB Swap: 2097148 total, 0 used, 2097148 free. 278556 cached Mem

  PID USER      PR  NI   VIRT   RES   SHR S  %CPU  %MEM     TIME+ COMMAND
 3331 root        20   0 1755192 234352 37700 S 138.4   6.1   1:15.85 gnome-she+
 3782 root        20   0 123920   1800  1068 R   6.0   0.0   0:00.03 top
    1 root        20   0  53684   7612  2528 S   0.0   0.2   0:03.02 systemd
    2 root        20   0     0     0     0 S   0.0   0.0   0:00.04 kthreadd
    3 root        20   0     0     0     0 S   0.0   0.0   0:00.08 ksoftirqd+
    5 root         0 -20     0     0     0 S   0.0   0.0   0:00.00 kworker/0+
    7 root        rt    0     0     0     0 S   0.0   0.0   0:02.05 migration+
    8 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcu_bh
    9 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/0
   10 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/1
   11 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/2
   12 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/3
   13 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/4
   14 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/5
   15 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/6
   16 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/7
   17 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/8

```

- 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。

注：

第 3 行中的数据均为 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]”。

```
[root@linuxprobe ~]# kill 2156
```

10、killall 命令

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。

```
[root@linuxprobe ~]# pidof httpd
13581 13580 13579 13578 13577 13576

[root@linuxprobe ~]# killall httpd

[root@linuxprobe ~]# pidof httpd

[root@linuxprobe ~]#
```

注：

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。

二、系统状态检测命令

1、ifconfig 命令

此命令被用来获取网络接口配置信息并对此进行修改。由于此命令叫复杂，可以参考此篇文章进行学习 <https://xp9802.iteye.com/blog/2120351>。

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
```


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 user, 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	1.8GB	1.3GB	542MB	9.8MB	1.6MB	413MB
-/+ buffers/cache		869MB	957MB			
Swap	2.0GB	0	2.0GB			

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。

```
[root@linuxprobe ~]# who
```

表 2-9

执行 who 命令的结果

登录的用户名	终端设备	登录到系统的时间
root	:0	2017-08-24 17:52 (:0)
root	pts/0	2017-08-24 17:52 (:0) #wanghuiwei888

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.....
```

<https://blog.csdn.net/wanghuiwei888>

7、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。

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
```

三、工作目录切换命令

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
```

四、文本文件编辑命令

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。

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。

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 --enablesshadow --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.....

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`”。

如果需要阅读长篇小说或者非常长的配置文件，那么“小猫咪”可就真的不适合了。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 windows system configuration information
```

```
xconfig --startxonboot

# License agreement
eula --agreed
# System authorization information
auth --enablesshadow --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=enol6777728 --onboot=off --ipv6=auto
network --bootproto=dhcp --hostname=linuxprobe.com
--More-- (43%)
```

<https://blog.csdn.net/wanghuiwei888>

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 --enablesshadow --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=enol6777728 --onboot=off --ipv6=auto
[root@linuxprobe ~]#
```

<https://blog.csdn.net/wanghuiwei888>

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'
```

<https://blog.csdn.net/wanghuiwei888>

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
FIRSTBOOT --ENABLE
IGNOREDISK --ONLY-USE=SDA
# KEYBOARD LAYOUTS
KEYBOARD --VCKEYMAP=US --XLAYOUTS='US'
# SYSTEM LANGUAGE
LANG EN_US.UTF-8

# NETWORK INFORMATION
NETWORK --BOOTPROTO=DHCP --DEVICE=ENO1677728 --ONBOOT=OFF --IPV6=AUTO
NETWORK --HOSTNAME=LOCALHOST.LOCALDOMAIN
# ROOT PASSWORD
```

<https://blog.csdn.net/wanghuiwei888>

```
ROOTPW --ISCRYPTED $6$PDJF42G8C6PL069$II.PX/YFAQPO0ENW2PA7MOMKJLYOAE2ZJMZ2UZJ7
BH3U04OWTR1.WK/HXZ3XIGMZGJPCS/MGPYSSOI8HPCT8B/
# SYSTEM TIMEZONE
TIMEZONE AMERICA/NEW_YORK --ISUTC
USER --NAME=LINUXPROBE --PASSWORD=$6$A9V3INSTNBWEIR7D$JEGFYWBCDOOKJ9SODECCDO.
ZLF4OSH2AZ2SS2R05B6LZ2A0V2K.RJWSBALL2FEKQVGF640OA/TOK6J.7GUTO/ --ISCRYPTED --
GECOS="LINUXPROBE"
# X WINDOW SYSTEM CONFIGURATION INFORMATION
XCONFIG --STARTXONBOOT
# SYSTEM BOOTLOADER CONFIGURATION
BOOTLOADER --LOCATION=MBR --BOOT-DRIVE=SDA
AUTOPART --TYPE=LVM
# PARTITION CLEARING INFORMATION
CLEARPART --NONE --INITLABEL

%PACKAGES
@BASE
@CORE
@DESKTOP-DEBUGGING
@DIAL-UP
@FONTS
@GNOME-DESKTOP
@GUEST-AGENTS
@GUEST-DESKTOP-AGENTS
@INPUT-METHODS
@INTERNET-BROWSER
@MULTIMEDIA
@PRINT-CLIENT
@X11
```

<https://blog.csdn.net/wanghuiwei888>

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”。

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
<code>-l</code>	只显示行数
<code>-w</code>	只显示单词数
<code>-c</code>	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: -
```

<https://blog.csdn.net/wanghuiwei888>

8、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@localhost ~]# cat aaa
```

```
same
```

```
aaa
```

```
[root@localhost ~]# cat bbb
```

```
same
```

```
bbb
```

```
[root@localhost ~]# _
```

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@localhost ~]# diff --brief aaa bbb
```

```
File aaa and bbb differ
```

```
[root@localhost ~]# _
```

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：

```

FILEs aaa and bbb differ
[root@localhost ~]# diff -c aaa bbb
*** aaa 2019-05-25 11:18:57.389522794 +0800
--- bbb 2019-05-25 11:19:21.320793080 +0800
*****
*** 1,2 ****
    same
! aaa
--- 1,2 ----
    same
! bbb
[root@localhost ~]# _
    
```

五、文件目录管理命令

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：

```
[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：

- 1、如果目标文件是目录，则会把源文件复制到该目录中；
- 2、如果目标文件也是普通文件，则会询问是否要覆盖它；
- 3、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）

<https://blog.csdn.net/wanghuiwei888>

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。dd 命令的参数及其作用如下表。

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

<https://blog.csdn.net/wanghuiwei888>

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 this 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
```

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
```

六、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。`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

<https://blog.csdn.net/wangniwei888>

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。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.....
```

<https://blog.csdn.net/wanghuiwei888>

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/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
etc/fonts/conf.d/65-1-vlgothic-gothic.conf
etc/fonts/conf.d/65-0-lohit-bengali.conf
etc/fonts/conf.d/20-unhint-small-dejavu-sans.conf
.....省略部分解压过程信息.....
```

<https://blog.csdn.net/wanghuiwei888>

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下表。

-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

<https://blog.csdn.net/wanghuiwei888>

grep 命令是用途最广泛的文本搜索匹配工具，虽然有很多参数，但是大多数基本上都用不到。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.....
```

<https://blog.csdn.net/wanghuiwei888>

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”。

“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（后面的字母参数依次表示块设备、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）

<https://blog.csdn.net/wanghuiwei888>

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。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
```

<https://blog.csdn.net/wanghuiwei888>

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.....
```

<https://blog.csdn.net/wanghuiwei888>

注：

进阶实验：在整个文件系统中找出所有归属于 linuxprobe 用户的文件并复制到 /root/findresults 目录。

该实验的重点是“-exec {} \;”参数，其中的 {} 表示 find 命令搜索出的每一个文件，并且命令的结尾必须是“\;”。完成该实验的具体命令如下：

```
[root@linuxprobe ~]# find / -user linuxprobe -exec cp -a {} /root/findresults/\;88
```