

Linux 命令行

转载

weixin_30305735

于 2019-02-21 22:35:00 发布

312

收藏 1

版权

文章标签：

awk

运维

shell

Linux 命令笔记

一、目录/文件

1.1 目录文件日常操作

1	.	->	当前目录
2	..	->	上一级目录
3	.file/.dir	->	隐藏文件/文件夹

登录后复制

【ls】查看指定目录文件
ls命令是最常用的linux命令，要配合着选项使用。

1	常用命令
2	
3	ls -lh # 方便查看文件大小
4	ls -AsSh

【pwd】:查看当前目录

1	pwd等价于/bin/pwd -L，显示当前绝对路径，如果是链接，则显示链接路径
2	pwd -P等价于/bin/pwd -P，显示实际路径，而非链接路径
3	
4	如果文件夹被临时删除，pwd还是显示未删除文件夹的那个路径，这时候可以使用/bin/pwd，就会提示当前路径不存在

【cd】：切换目录

1	cd - # 上一目录
2	cd ~ # 家目录
3	cd # 家目录
4	cd .. # 上一目录
5	cd !\$ # 将上命令的参数做為cd 参数

【touch】：新建文件

1	1. 创建新文件同时可以指定一些时间参数
2	\$ touch newfile
3	
4	2. 一次性创建多个文件
5	\$ touch {1..10}.txt

可以对已有文件修改时间戳（ll显示的时间）

1	[1]
2	touch -d [[CC]YY]MMDD text
3	touch -t [[CC]YY]MMDDhhmm[.SS] text
4	CC为年数中的前两位，即“世纪数”；YY为年数的后两位，即某世纪中的年数。如果不给出CC的值
5	touch -d 20171004 text
6	touch -t 201710041330.30 text
7	
8	[2]
9	-r: 以另一文件为基准更新时间戳
10	以file1的时间戳为基准，将file2的改成一样的
11	touch -r file1 file2
12	
13	[3] 其他不常用参数
14	-a 或--time=atime或--time=access或--time=use 只更改存取时间。
15	-c 或--no-clobber 不覆盖任何文件。

▼

【mkdir】：新建文件夹

1	# 创建一个或多个的同级目录
2	mkdir adir bdir
3	
4	# 递归创建目录
5	# 当前目录下并没有cdir这个文件夹，如果使用单端mkdir cdir/ddir会报错。
6	# 应该使用
7	mkdir -p cdir/ddir
8	
9	# 指定权限
10	mkdir -m 777 dir_name
11	
12	# 显示创建信息（成功或失败）
13	mkdir -v dir_name
14	
15	# 创建时如遇到已存在的文件会报错

▼

【rm | rmdir】

1	# 只能删除空文件夹
2	rmdir dirname
3	
4	# 删除当前文件夹下所有文件并不提示
5	rm -f *
6	
7	# 递归删除dirname下所有文件
8	rm -r dirname
9	
10	# 上面二者结合，递归删除dirname下所有文件并不提示
11	rm -rf dirname

【cp】:复制文件或目录

1	# 递归复制old_dir目录下所有文件和文件夹到new_dir文件夹下
2	cp -r old_dir new_dir

注意：默认cp拷贝文件后会使用预设权限，即其他人没有更改的权限，需要使用-p或者-a文件所有的所有特性都一起复制过来（拉取源站数据缓存，不更改文件最后修改时间）

【mv】：移动或重命名

1	# 移动/tmp/test/sample.txt文件到当前目录下
2	mv /tmp/test/sample.txt ./

【cat】查看文件

作用：文本文件查看和连接工具，用于查看文本文件的内容。

命令格式：cat file 经常和more、head、tail、less以及管道命令结合使用，如：cat file | more、cat file | head等。

三个功能

1	1.一次显示整个文件:cat filename
2	2.从键盘创建一个文件:cat > filename 创建文件，输入内容，Ctrl+d 结束
3	3.将几个文件合并为一个文件:cat file1 file2 > file

命令参数

1	-A, --show-all 等价于 -vET
2	-b, --number-nonblank 对非空输出行编号
3	-e 等价于 -vE
4	-E, --show-ends 在每行结束处显示 \$
5	-n, --number 对输出的所有行编号,由1开始对所有输出的行数编号
6	-s, --squeeze-blank 有连续两行以上的空白行，就代换为一行的空白行
7	-t 与 -vT 等价
8	-T, --show-tabs 将跳格字符显示为 ^I
9	-u (被忽略)
10	-V, --show-nonprinting 使用 ^ 和 M- 引用，除了 LFD 和 TAB 之外

【tac】反向查看

cat是第一行到最后一行，tac是最后一行到第一行

【head | tail】

1	# 显示前10行
2	head xs.txt 等价于 cat xs.txt head 等价于 cat xs.txt head -n 10
3	
4	
5	# 显示最后10行
6	tail xs.txt 等价于 tail -n 10 xs.txt
7	
8	
9	# 显示除末尾10行外
10	head -n -10 xs.txt

【more | less】

更具体命令可以参考：Linux中more和less命令用法

1	【more】
2	# 回车就往下一行显示，空白键就往下一页显示
3	# 按 b 键就会往回一页显示，按 q 就会退出
4	more xs.txt
5	
6	# 查找比较鸡肋，并从该处前两行开始显示输出
7	more +/查找内容 xs.txt
8	
9	【less】
10	# 和more几乎一样的功能，但是less更灵活（可以使用任何vim的移动命令，还有标记功能很使用），比如查找命令，直接像vim一样输入/即可
11	less xs.txt
12	
13	ma：使用 a 标记文本的当前位置
14	'o：导航则标记 a 处
15	
16	F：实现和tail -f的功能，实时输出内容，tail +F xs.txt

小练习

选取xs.txt的10-20行

```
1 | head -n 20 xs.txt|tail
2
3 # 更快捷的方法
4 sed -n '10,20p' xs.txt
```

1.2 文件处理

wc

统计文件信息

```
1 # 统计所有信息（行数、单词数、字符数）
2 $ wc /etc/passwd
3
4 # 行数
5 $ wc -l /etc/passwd
6
7 # 单词数    # 中文无法统计
8 $ wc -w /etc/passwd
9
10 # 字节数
11 $ wc -c /etc/passwd
12
13 # 字符数
14 $ wc -m /etc/passwd
15
16 # 最长行字节数
17 $ wc -L /etc/passwd
```

sort

```
1 $ cat /etc/passwd | sort
2 $ cat /etc/passwd | sort -r
3
4 # 以:为分隔符，对第三列排序，所得的结果，再通过cut以:为分隔符取第三列
5 $ cat /etc/passwd | sort -t ':' -k 3 |cut -d ':' -f 3
```

uniq

```
1 # 显示每行重复频率
2 uniq -c 文件名
3
4 # 只显示有重复的行
5 uniq -d 文件名
```

挑战题目

```
1 wget http://labfile.oss.aliyuncs.com/courses/1/data1
2 # 里面是包含一些命令使用的列表
3
4 # 要求：从里面找出出现频率次数前3的命令并保存在/home/result
5
6 $ cat data1 | cut -c 8-|cut -d " " -f 1|sort | uniq -dc | sort -r -n -k1 | head -n 3 > /home/result
```

1.3 文字处理

tr

tr可以删除或者去重某文本信息中的某些文字。还可以进行替换操作

```
1 # 删除hello，里面所有l,o字符
2 $ echo 'hello' | tr -d 'lo'
3
4 # 去重hello里的l
5 $ echo 'hello' | tr -s 'l'
6
7 # hello里，l换成a，e换成b
8 $ echo 'hello' | tr 'le' 'ab'
```

col

将tab转换为等数量的空格，或者反转

```
1 -x    tab转空格
2 -h    空格转tab(默认)
```

```
1 # 查看 /etc/protocols 中的不可见字符，可以看到很多 ^I，这其实就是 Tab 转义成可见字符的符号
2 $ cat -A /etc/protocols
3
4 # 使用 col -x 将 /etc/protocols 中的 Tab 转换为空格，然后再使用 cat 查看，你发现 ^I 不见了
5 $ cat /etc/protocols | col -x | cat -A
```

sed

文件处理工具。

使得不需要打开文件就可以对文件进行操作（删除，替换，选取，新增）。以行为单位进行处理。

常用选项

```
1 -n:经过处理后的结果显示出来，不影响真实文件。
2 -e: 直接在指令列模式上进行 sed 的动作编辑；（没明白）
3 -f:直接将 sed 的动作写在一个档案内， -f filename 则可以执行 filename 内的sed 动作；
4 -i:直接修改读取的档案内容，而不是由屏幕输出。
```

常用命令：

```
1 a:新增， a 的后面可以接字符串，而这些字符串会在新的一行出现（目前的下一行）~
2 c:取代， c 的后面可以接字符串，这些字符串可以取代 n1,n2 之间的行！
3 d:删除，因为删除删除，所以 d 后面通常不接任何条件；
4 i:插入， i 的后面可以接字符串，而这些字符串会在新的一行出现（目前的上一行）；
5 p:列印，亦即将某个选择的资料印出。通常 p 会与参数 sed -n
6 s:取代，可以直接进行取代的工作哩！通常这个 s 的动作可以搭配正规表示法！例如 1,20s/old/new/g 就是啦！
```

删除某行

```
1 sed '1d' somefile      #删除第一行
2 sed '5d' somefile      #删除最后一行
3 sed '1,2d' somefile    #删除第一行到第二行
4 sed '2,5d' somefile    #删除第二行到最后一行
5
6 # 以上，只是输出，源文件并不会替换，若要对源文件进行修改，需加 -i
7 sed -i '1,2d' somefile
```

显示某行

```
1 sed -n '1p' somefile   #显示第一行
2 sed -n '$p' somefile   #显示最后一行
3 sed -n '1,2p' somefile #显示第一行到第二行
4 sed -n '2,5p' somefile #显示第二行到最后一行
```

使用模式进行查询

```
1 sed -n '/ruby/p' somefile #查询包括关键字ruby所在所有行
2 sed -n '/\$/p' somefile  #查询包括关键字$所在所有行，使用反斜线\屏蔽特殊含义
```

插入行

```
1 sed '1a drink tea' somefile      #第一行后增加字符串"drink tea"
2 sed '1,3a drink tea' somefile    #第一行到第三行后增加字符串"drink tea"
3 sed '1a drink tea\nor coffee' somefile #第一行后增加多行，使用换行符\n
```

替换行

```
1 sed '1c Hi' somefile      #第一行代替为Hi
2 sed '1,2c Hi' somefile    #第一行到第二行代替为Hi
```

替换行中部分数据

```
1 sed 's/ruby/bird/g somefile' #替换ruby为bird，记住这个并不会更改源文件，只是输出
2 sed 's/ruby//g' somefile     #删除ruby
```

【注意】：以上对源文件都不做修改，若要修改，要加上-i

1.4 文件重构

cut

【显示每行的某位置的内容】

```
1 # 前五个（包含第五个）
2 $ cut /etc/passwd -c -5
3 # 前五个之后的（包含第五个）
4 $ cut /etc/passwd -c 5-
5 # 第五个
6 $ cut /etc/passwd -c 5
7 # 2到5之间的（包含第五个）
8 $ cut /etc/passwd -c 2-5
```

【以指定的分隔符分隔，并返回某些列】

```
1 # 返回第一列和第六列
2 $ cut /etc/passwd -d ":" -f 1,6
```

awk

awk是一个强大的文本分析工具。

简单来说awk就是把文件逐行的读入，以空格为默认分隔符将每行切片（相当于Excel的分列），切开的部分再进行各种分析处理。

```
1 # 打印空白行，文件有几个空行，就输出几个空行
2 awk '/$/ {print "This is a blank line"}' awk.txt
3
4 # 打印全部列，引用变量
5 awk '{print $0}' awk.txt
6
7 # 打印前三列，引用变量
8 awk '{print $1,$2,$3}' awk.txt
9
10 # 指定间隔符为空格，获取第四列
```



```
11 | awk -F " " '{print $4}' awk.txt |
12 |
13 | # 重组表格
14 | awk -F ':' ' '{print $1"\t"$7}' awk.txt
15 |
16 | # 表头和结尾，会先输出name,shell
17 | awk -F ':' ' BEGIN {print "name,shell"} {print $1","$7} END {print "blue,/bin/nosh"}' awk.txt
18 |
19 | # 在awk中引用变量变量，在原理是拼接。
20 | limit=30
21 | df -Th| grep "/dev/vd" | sed 's/%/g' |awk '{ if($6>"$limit") print $6}'
```

1.5 其他命令

nl

-b：指定行号指定的方式，主要有两种：

```
1 | -b a：表示不论是否为空行，也同样列出行号(类似 cat -n)；
2 | -b t：如果有空行，空的那一行不要列出行号(默认值)；
```

-n：列出行号表示的方法，主要有三种：

```
1 | -n ln：行号在螢幕的最左方显示；
2 | -n rn：行号在自己栏位的最右方显示，且不加 0；
3 | -n rz：行号在自己栏位的最右方显示，且加 0；
```

```
1 | -w：行号栏位的占用的位数。
2 | nl -b a -n rz -w 3 text
```

1.6 文件查找

which: 查询软件

在PATH变量指定的路径中，搜索某个系统命令（可执行文件）的位置，并且返回第一个搜索结果。

参数选项（基本不用）

```
1 | -n 指定文件名长度，指定的长度必须大于或等于所有文件中最长的文件名。
2 | -p 与 -n参数相同，但此处的包括了文件的路径。
3 | -w 指定输出时栏位的宽度。
4 | -V 显示版本信息
```

grep: 搜索神器

搜索并筛选显示结果。

该命令经常配合管道命令来控制输出。

以下 是常用的选项：



【非常好用：不打开文件，直接搜索指定目录下文件内的内容】

```
1 | $ vim a # wongbingming
2 | $ vim b # wangbingming
3 | $ grep -rni "bingming" . # 当然这里也可以使用正则表达式
4 | ./a:1:wongbingming
5 | ./b:1:wangbingming
6 |
7 | 参数解释
8 | -r 递归遍历各个文件夹下的所有文件
9 | -n 显示在文件中的第几行查询到
10 | -I 忽略二进制文件
```

wheris: 简单快速

定位可执行文件、源代码文件、帮助文件在文件系统中的位置。

个搜索很快，因为它并没有从硬盘中依次查找，而是直接从数据库中查询。

```
$ whereis who
```

参数选项

```
1 | -b 定位可执行文件。
2 | -m 定位帮助文件。
3 | -s 定位源代码文件。
4 | -u 搜索默认路径下除可执行文件、源代码文件、帮助文件以外的其它文件。
5 | -B 指定搜索可执行文件的路径。
6 | -M 指定搜索帮助文件的路径。
7 | -S 指定搜索源代码文件的路径。
```

locate: 快而全

通过/var/lib/locate/locate.db数据库查找，不过这个数据库也不是实时更新的，系统会使用定时任务每天自动执行updatedb命令更新一次，所以有时候你刚添加的文件，它可能会找不到，需要手动执行一次 updatedb命令（在我们的环境中必须先执行一次该命令）。

```
1 | $ updatedb
2 | $ locate /etc/sh
3 |
4 | # 查找/etc/目录下所有以sh开头的文件
```

find: 小而细

鸟哥的Linux私房菜·find

每天一个linux命令（19）：find 命令概览

每天一个linux命令（20）：find命令之exec

1.7 文件传输/下载

scp

实现不同机器之间传输数据（加密）

[scp详解](#)

curl

Curl是一个命令行方式下传输数据的开源传输工具，支持多种协议包括：FTP,HTTP,HTTPS,IMAP,POP3,TELNET等。同样支持HTTP POST方法，PUT方法，FTP上传，cookie，用户名/密码认证，下载文件断点续传等，功能十分强大。

常用的，用于模拟浏览器请求。

[curl详解](#)

wget

测试速率

```
wget -S http://115.231.74.93:80/lvs.lxdns.net/test.rar && rm -rf test.rar*
```

1.8 文件压缩

zcat

查看压缩的文件内容

```
zcat file.gz
```

Linux上的压缩格式比Windows上多很多，在 Windows 上最常见的不外乎这三种 *.zip，*.rar，*.7z 后缀的压缩文件。而在 Linux 上面常见的格式除了以上三种外，还有 *.gz，*.xz，*.bz2，*.tar，*.tar.gz，*.tar.xz，*.tar.bz2，对于常见的压缩格式，tar已经可以解决，所以这里只介绍tar。tar并不能压缩和解压7z，zip等其他文件

tar

压缩示例

```
1 | # test是当前目录下一个文件夹
2 | $ tar -czvph test.tar.gz old_folder
```

参数解释

```
1 | -c 指明创建tar文件
2 | -z 指明生成gz文件
3 |
4 | -v 可视输出，不加就静默压缩
5 |
6 | -f 指明压缩后的文件名，必须-f后必须紧跟文件名，否则无效
7 | -p 当在其他主机还原时希望保留文件的属性
8 | -h 备份链接指向的源文件而不是链接本身
```

解包示例

```
1 | $ mkdir new_folder
2 | $ tar -xzvf test.tar.gz -C new_folder
```

参数解释

```
1 | -x 解压命令
2 | -z 指明源文件是gz文件
3 | -f 指明压缩后的文件名，必须-f后必须紧跟文件名，否则无效
4 |
5 | -v 可视化输出解包过程，一般不加，静默解包
6 |
7 | -C 指明解压到哪个目录
```

其他压缩格式

```
1 | *.tar.gz -xz
2 | *.tar.xz -J
3 | *tar.bz2 -j
```

gz

最简单的压缩格式

```
1 | # 压缩
2 | gzip somefile
3 |
4 | # 解压
5 | gzip -d somefile.gz
```

7zip

```
1 | yum install p7zip
2 |
```

```
3 # 解压 4 | 7za x file.7z
5
6 # 压缩
7 # 先将要压缩的文件都放到~/test/ 下
8 7za a file.7z ~/test/
```

zip

```
1 # 解压
2 unzip some.file
3
4 # 压缩
5 zip -r some.zip file1 file2 dir1 dir2...
```

1.9 rpm包管理

```
1 # 解压rpm包
2 rpm2cpio xxx.rpm |cpio -div
3
4 # 查看文件属于哪个rpm包
5 rpm -qf /path/filename
6
7 # 使用rpmrebuild重新生成rpm包
8 # 使用rpmrebuild需要依赖rpmbuild: yum install -y rpmbuild
9 rpmrebuild xxx
```

二、系统管理

2.0 环境变量

变量分为用户变量（env）和shell变量（set）

```
1 # 添加shell变量
2
3 [root@host ~]# myuser=wangbm
4 [root@host ~]# echo $myuser
5 wangbm
6 [root@host ~]# env|grep myuser
7 [root@host ~]# set|grep myuser
8 myuser=wangbm
9
10
11 # 添加用户变量，会发现用户变量会覆盖shell变量
12 [root@host ~]# export myuser=wangbingming
13 [root@host ~]# env|grep myuser
14 myuser=wangbingming
15 [root@host ~]# set|grep myuser
16 myuser=wangbingming
17
18
19 # 删除环境变量
20 [root@host ~]# unset myuser
21 [root@host ~]# env|grep myuser
22 [root@host ~]# set|grep myuser
```

一些系统的变量

```
1 # 这里生成的变量，对所有用户生效
2 /etc/profile
3
4 # 对特定user生效
5 /root/.bash_profile
6 /home/user/.bashrc
7
8 # 注意修改文件后，要手动source一下
```

2.1 相关查询

系统信息

```
1 # 查询开机时间
2 $ who -b
3
4 # 查询系统内核
5 $ uname -r
6
7 # 查询是否安装某个rpm包
8 $ rpm -qa|grep nova
9
10 # service 文件目录
11 $ /usr/lib/systemd/system # centos 7.x
12 $ /etc/init.d # centos 6.x
13
14 # 查询开机自启列表
15 $ systemctl list-unit-files
16 $ chkconfig --list
17
18 # 查看系统版本
19 $ cat /etc/redhat-release
20
21 # ----- 或者 -----
22 $ yum install redhat-lsb -y
23 $ lsb_release -a
24
25 # 进程树
26 $ pstree -p
27
28 # 查看系统运行多长时间
29 $ uptime/w
30
31 # 查看系统版本
32 $ lsb_release -a
33
34 # 查看CPU信息
35 $ cat /proc/cpuinfo
36 $ numactl -H
```

时间查询

输出操作系统的当前日期、时间和时区。

```
1 # -s参数用于修改当前的日期和时间
2 date -s 2007-10-17
3 date -s 18:05:00
4
5 用法: cd /CNLog/cache/qsLogBackSrcFull/bkDir/'date +%Y-%m-%d'
```

2.2 系统分区

分区介绍

Linux的分区，不同于Windows，一定要区别对待，不然会搞不明白。

Linux的分区的过程经历以下几个步骤

1. 设备分区：对硬盘存储空间的划分
2. 格式化：写入文件系统
3. 挂载：将分区挂载到目录上，才能访问数据

关于硬件对应的设备文件名，可以参考下图



其中以硬盘为例来说明

```
1 硬盘可以分为三种
2 hd : IDE硬盘接口（淘汰，接口最大传输100兆/s)
3 sd : SCSI硬盘接口（淘汰，接口最大传输200M），和SATA硬盘接口
4
5 现在都是SATA的硬盘接口
6
7 /dev/sda1 表示的是第一块（a）SATA硬盘的第一个分区（1）
8 /dev/sdb2 表示的是第二块（b）SATA硬盘的第二个分区（2）
```

分区类型

可以分为：主分区、扩展分区、逻辑分区

```
1 【主分区】：最多只能有4个（受硬盘结构限制，如果硬盘结构不变，将都被限制）
2
3 【扩展分区】：
4 1. 最多只能有一个
5 2. 主分区+扩展分区，一共只能有四个，可以少于
6 3. 扩展分区下面，不能存放数据，只能进行逻辑分区的划分
7
8 【逻辑分区】：挂载后就是一个目录下的空间，数量不受限制
```

格式化做了哪些事

```
1 【目的】
2 1. 不是为了清空数据
3 2. 主要是为了写入文件系统
4
5 【文件系统】
6 Windows:
7 FAT16（每个分区大小最大不能超过2G），FAT32（单个分区大小最大16G，单个文件大小不能超过4G），NTFS
8
9 Linux: ext2, ext3, ext4
10
11 ->>>>> 越往后越先进
12
13 【写入文件系统做了啥】
14 1. 分数据块
15 把空间分成若干个等大小的数据块（block），每个大小4kb
16 如果我们有一个文件10kb，那么会占用3个数据块，实际大小就会是12kb
17
18 你可以查看一个文件夹的大小，也都是4kb
19 ll -l
20
21 2. 建立数据表
22 一个文件被分成若干个的数据块，那么如果有用户访问的时候，就需要有一个表把这些数据块拼凑起来。
23 这个数据表就记录了这个文件由哪些数据块组成。
```

分区说明

```
1 必须分区
2 / 根目录，最高级目录，不分配的话，所有的文件都没存储，软件没法运行
3 /swap 交换分区，虚拟内存，4G以下，分2倍，4G以上，和真实内存一样即可
```



```
4 | 5 | 推荐分区
6 /boot    启动分区，防止/ 分区写满，导致系统无法启动，不需要很大，200M足矣
7
8
9 逻辑分区号：只能从5开始，即便3,4没有被使用
```

分区操作

关于分区操作可以参考这个: [分区操作](#)

```
1 ## 查看分区表信息
2 $ sudo fdisk -l
3 $ lsblk
4
5 # 删除分区前，先确认有没有挂载点，有的话需要先umount卸载
6 $ fdisk /dev/vdb  然后再接d,w
```

2.3 进程管理

ps、kill、killall

参考文档: [ps 命令的十个简单用法](#)

```
1 # 查看当前所有进程
2 ps -aux
3
4 # 终止pid为1095的进程
5 kill 1095
6
7 # 强制终止pid为1095的进程
8 # 9是信号强度，强制杀死
9 # 其他信号, -1 该信号让进程正常关闭，然后重新读取配置文件之后重启
10 # -15 正常结束进程的信号，kill命令默认信号
11 kill -9 1095
12
13 # 终止指定程序
14 killall 程序名
15
16 # pkill
17 pkill -9 httpd 强制终止进程
18 pkill -t -9 pts/1 强制杀死pts/1虚拟终端登入的进程
```

top

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状况，类似于Windows的任务管理器。

[linux的top命令参数详解](#)

```
1 top      实时得查看进程的状态，以及系统的一些信息（如 CPU、内存信息等），3s刷新一次
2 ps       来静态查看当前的进程信息
3 pstree   来查看当前活跃进程的树形结构。
```

pgrep

```
1 [root@wangbm web]# pgrep keepalived
2 21955
3 21956
4 21957
5 [root@wangbm web]# ps -ef|grep keepalived
6 root      21955      1  0 21:27 ?        00:00:00 /usr/sbin/keepalived -D
7 root      21956 21955   0 21:27 ?        00:00:00 /usr/sbin/keepalived -D
8 root      21957 21955   0 21:27 ?        00:00:00 /usr/sbin/keepalived -D
9 root      22035 12159   0 21:33 pts/0    00:00:00 grep --color=auto keepalived
```

2.4 设备信息管理

df

显示磁盘的相关信息

```
df -Th
```

free

输出内存的使用情况，m,g分别是指定单位，默认是kb

```
1 free -m
2 free -g
```

total 表示总内存大小；
used和free分别表示被使用和空闲内存大小；
share指可被多个进程共同享有的内存； buffers和cached用来保留最近访问的文件和数据，当其他进程需要更多的内存时，这些内容可以被缩减； Free命令还可以输出交换空间的相关信息。

ifconfig

显示或设置网络设备

last

列出目前与过去登入系统的用户相关信息。一般可用来查看系统重启记录

history

屏幕输出当前用户在命令行模式下执行的最后（1000个）命令

passwd

这个用于修改密码

非交互式修改密码

```
echo `root12#$`| passwd --stdin root
```

reboot/shutdown

重启/关机

```
1 reboot/shutdown -r now
2
3 # reboot [-n][-[w][d]([-l]) 重新启动计算机，使用权限是系统管理员
4 -n 重启前不将记录写回硬盘
5 -w 并不是真的重启，只是把记录写道/var/log/wtmp文件中
6 -d 不把记录写入/var/log/wtmp文件中
7 -i 重启前先把所有与网络相关的装备停止
```

rpm

```
1 # 安装rpm包
2 rpm -ivh monitor-system-1.2-1.i386.rpm
3
4 # 更新rpm包
5 rpm -U/Fvh （F只更新已存在的文件）
6
7 # 查询包中的文件
8 rpm -ql monitor-system
9
10 # 查询文件所属的包
11 rpm -qf /usr/local/squid/etc/squid.conf（绝对路径/到目录下查找）
12
13 # 查询所有包
14 rpm -qa | grep squid
15
16 # 卸载某个rpm包
17 rpm -e monitor-system
```

开启80端口

```
1 vi /etc/sysconfig/iptables
2
3 # 在22端口下面添加一行
4 -A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
5
6 # 重启防火墙
7 /bin/systemctl restart iptables.service
```

修改时区

```
1 # 由EDT（美国）改成CST（中国）
2
3 $ mv /etc/localtime /etc/localtime.bak
4 $ ln -s /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

查看机器连接的哪个交换机

```
1 $ service lldpd restart
2 $ lldpdcli show nei sum  # 查看JS_HuaiAnDXXZ4_166.3
3 .....
4 LLDP neighbors:
5 .....
6 Interface:   eth1, via: LLDP
7 ChassisID:   mac 00:6b:8e:01:1a:07
8 ChassisID:   mac 00:6b:8e:01:1a:07
9 SysName:     JS_HuaiAnDXXZ4_165.132_IPMI
10 Port:
11 PortID:      local 39
12 PortDescr:   Ethernet1/0/39
13 TTL:         120
14 .....
15 Interface:   eth0, via: LLDP
16 ChassisID:   mac 58:69:6c:62:7b:c2
17 ChassisID:   mac 58:69:6c:62:7b:c2
18 SysName:     JS_HuaiAnDXXZ4_166.3
19 Port:
20 PortID:      ifname TenGigabitEthernet 0/41
21 PortDescr:   TenGigabitEthernet 0/41
22 TTL:         121
23 .....
```

2.5 磁盘管理

df

显示指定磁盘文件的可用空间。

这里要理解挂载的概念。

所有的设备（磁盘/等其他介质）都需要挂载在一个目录，Linux才能访问数据。

具体可以参考这篇文章: [df 命令](#)

常用命令

```
1 $ df -Th
2 $ df -lh
3
4
5 $ df -h      以1024来换算
6 $ df -H      以1000来换算
7
8 $ df -t ext3  指定设备类型
9 $ df -T       文件系统的类型
10 $ df -li      查看inode的使用情况
```

关于inode可以查看：[inode的理解](#)

du

查看目录的容量

```
1 $ du -h -d 0 ~
2
3 # 参数解释
4 -h  以人能看得懂的方式显示
5 -d  扫描的目录层级，0表示只有一个指定目录，1表示次级目录
6
7 ~   是家目录，这里可以选你指定的目录
8
9 $ du -s
10 # 查看当前所处目录总容量大小，单位是k，不可跟-d
11
12 $ du -a
13 # 查看当前所处目录及所有子目录的所有文件，单位是k，建议不用
```

查看文件的大小

```
$ du -h file.txt
```

dd

dd命令用于转换和复制文件，不过它的复制不同于cp。之前提到过关于 Linux的很重要的一点，一切即文件，在 Linux上，硬件的设备驱动（如硬盘）和特殊设备文件（如/dev/zero和/dev/random）都像普通文件一样，只要在各自的驱动程序中实现了对应的功能，dd也可以读取自和/或写入到这些文件。这样，dd也可以用在备份硬件的引导扇区、获取一定数量的随机数据或者空数据等任务中。dd程序也可以在复制时处理数据，例如转换字节序、或在 ASCII 与 EBCDIC编码间互换。

这里有篇文章讲得很好：[Linux-dd命令详解](#)

补充一点

```
1 # 输出到文件
2 $ dd of=test bs=10 count=1 # 或者 dd if=/dev/stdin of=test bs=10 count=1
3
4 # 输出到标准输出
5 $ dd if=/dev/stdin of=/dev/stdout bs=10 count=1
6 # 注
7 在打完了这个命令后，继续在终端打字，作为你的输入
```

前面说到dd在拷贝的同时还可以实现数据转换，那下面就举一个简单的例子：将输出的英文字符转换为大写再写入文件

```
dd if=/dev/stdin of=test bs=10 count=1 conv=ucase
```

制作虚拟镜像并挂载

```
1 # 制作虚拟镜像
2 $ dd if=/dev/zero of=virtual.img bs=1M count=256
3 $ mkfs.ext4 virtual.img
4
5 # 使用一个循环设备 (/dev/loop) 连接虚拟镜像文件
6 $ sudo losetup /dev/loop0 virtual.img
7
8 # 挂载前查看下当前已挂载的设备
9 $ sudo mount
10
11 # 挂载
12 $ sudo mkdir /mnt/point
13 $ sudo mount /dev/loop0 /mnt/point
14
15 # 上面连接虚拟镜像文件和挂载设备两条命令可以合并成一条，挂载类型可以省略，会自动识别
16 $ sudo mount -o loop [-t ext4] /mnt/point
17
18 # 再次查看挂载的设备，就可以发现我们新挂载的
19 $ sudo mount
20
21
22 # 卸载：一定要加-fl。不然无法卸载
23 $ sudo umount -fl /mnt/point
24
25 # 查看所有与文件相关联的 loop 设备
26 $ losetup -a
27 # 卸载
28 $ losetup -d /dev/loop0
```

2.6 任务管理

一次性任务（at）

创建定时任务

```
1 $ sudo apt-get install at
2 $ at now+5 minutes
3 at>/bin/ls
4 at><EOL> # 就是Ctrl+d
5 job 2 at Sum Oct 15 14:58:00 2017 # 结束的时候，告知执行时间
```

管理定时任务

```
1 $ atq # 可以查看当前还有哪些定时任务，会显示任务号
2
3 # 查看具体任务代码
4 $ at -c <job_no>
5
6 # 删除任务
7 $ atrm <job_no>
```

例行性任务（crontab）

参考以下文字

- 每天一个linux命令（50）：[crontab命令](#)
- [鸟哥的私房菜](#)

做个示例

```
1 # 检查crond服务是否启动
2 $ ps aux|grep crond
3
4 # 创建任务
5 $ crontab -e
6
7 # 输入1 回车
8 # 任务是：每天凌晨3点备份日志到/home/temp/目录，文件名为日期
9 # 跳到文件尾部输入任务：* * * * * cp alternatives.lob /home/temp/${date \%Y-\%m-\%d}。保存退出
10 # 注意%需要加\来转义，不然会被当成执行使用
11
12 # 查看任务
13 $ crontab -l
14
15 # 删除任务（当前用户），当然也可以指定用户
16 $ crontab -r [-u user]
```

68168834.jpg

三、用户/权限管理

3.1 用户管理

创建用户

```
1 [useradd和adduser区别]
2 useradd: 只创建用户，创建完了用 passwd lilei去设置新用户的密码。更像一种命令。
3
4 adduser: 会创建用户，创建目录，创建密码（提示你设置），做这一系列的操作。更像是一个程序，执行确认一系列操作。
```

删除用户

```
sudo deluser lilei --remove-home
```

切换用户

```
1 su [user] # 切换到其他用户，环境变量不变，还是当前路径，如果不指定user，就切换到root
3 su - [user] # 切换到其他用户，环境变量也切换，路径变为user家目录，如果不指定username，就切换到root，相当于使用user进行登录
```

Ubuntu切换用户

```
1 # 切换到root
2 $ sudo su
3
4 # 切换到普通用户
5 $ su user
```

CentOS

```
1 $ su
2 $ su root
3 $ su -
4 $ su -root
5
6 $ su user
7 $ su -user
```

用户组

- 1
1. 每个用户都至少属于一个用户组，创建的时候如果不指定，就和当前用户的组一样（root用户组除外）。2
2. 一个用户可以属于多个用户组。

```
1 # 查看所属用户组
2 $ groups [user]
```

UID/GID/组ID

```
1 # 显示相关信息，如要查看root用户的信息
2 id root
3
4 # 所有文件都保存在/etc/passwd
5 vi /etc/passwd
```

管理用户：[查看用户的UID和GID](#)

管理密码：[Linux下/etc/shadow文件](#)

添加sudo用户组

```
1 【第一种方法】
2 在root下，visudo或者vi /etc/sudoers，找到root ALL=(ALL)的下一行添加一行，user ALL=(ALL)，user是对应的用户名
3
4 【第二种方法】
5 在root下，使用命令sudo usermod -G sudo user，将user加入sudo用户组
```

sudo免密

当有些操作只有root用户才能操作的时候，怎么办？

1.我们需要切换到root用户操作。

- 1
- 当前用户属于sudo组，可以使用sudo [command]
- 2
- 使用sudo输入一次密码免密使用5分钟。还是太麻烦，可以配置当前用户免密执行sudo。

如何免密配置

```
1 vi /etc/sudoers.d/<user>
2
3 # 添加内容
4 # 如果要指定特定的命令不需要密码的话，就把ALL替换成命令路径，如下
5 NOPASSWD: /sbin/mount, (root) NOPASSWD: /bin/umount
6 <user> ALL=(ALL) NOPASSWD:ALL
7 Defaults:shiyantou !requiretty
8
9 # 有时候，虽然用户设置免密了，但是还是需要输入密码，是group覆盖了，需要把group也设成免密。
```

参考资料：[sudo免密](#)

3.2 权限管理

更改文件所有者和所属组

```
1 [sudo] chown 用户组:用户 文件/文件夹
2
3 # 如下只更改所属用户
4 [sudo] chown 用户 文件/文件夹
```

修改文件权限

文件权限有读、写、执行三种

分别对应数字4,2,1，也就是2*2,2*1,2*0

👉logoblackfont

如何修改文件权限

```
1 【第一种方法】
2 chmod 777 文件
3
4 【第二种方法】·加減的方法
5 g、o 还有 u 分别表示 group、others 和 user，+ 和 - 分别表示增加和去掉相应的权限。
6
7 举个例子
8 比如一个文件权限是：-wr-wr-wr-
9 chmod go-wr 文件
10
11 然后文件权限就变成：-wr-----
```

禁止修改、删除、移动文件

chattr -i和chattr +i

```
1 +：在原有参数设定基础上，追加参数。
2 -：在原有参数设定基础上，移除参数。
3
4 命令：chattr [ -RV ] [ -v version ] [ mode ] files
5
6 A: 文件或目录的 atime (access time)不可被修改(modified)，可以有效预防如手提电脑磁盘I/O错误的发生。
7 S: 磁盘I/O同步选项，功能类似sync。
8 a: 即append，设定读参数后，只能向文件中添加数据，而不能删除，多用于服务器日志文 件安全，只有root才能设定这个属性。
9 c: 即comprese，设定文件是否经压缩后再存储。读取时需要经过自动解压操作。
10 d: 即no dump，设定文件不能成为dump程序的备份目标。
11 i: 设定文件不能被删除、改名、设定链接关系，同时不能写入或新增内容。i参数对于文件 系统的安全设置有很大帮助。
12 j: 即journal，设定此参数使得当通过 mount参数: data=ordered 或者 data=writeback 挂 载的文件系统，文件在写入时会先被记录(在journal中)。如果filesystem被设定参数为 data=journal，则该参数自动失效。
13 s: 保密性地删除文件或目录，即硬盘空间被全部收回。
14 u: 与s相反，当设定为u时，数据内容其实还存在于磁盘中，可以用于undeletion。
```

四、网络管理

iptables

命令格式

```
1 iptables [-t table] 命令 [chain] [rules] [-j target]
2
3 【参数解释】
4 table 表名: filter、nat、mangle、raw，后两者不常用
5 命令 对链的操作命令
6 -P或-policy 定义默认策略
7 -L或-list 查看iptables规则列表
8 -A或-append 在规则列表的最后增加1条规则
9 -I或-insert 在指定的位置插入1条规则
10 -D或-delete 从规则列表中删除1条规则
11 -R或-replace 替换规则列表中的某条规则
12 -F或-flush 删除表中所有规则
13 -Z或-zero 将表中数据包计数器和流量计数器归零
14 chain 链名: PREROUTING, INPUT, FORWARD, OUTPUT, POSTROUTIN
15
16 rules 规则，可以说是匹配规则。
17 分为
18 1. 【通用匹配】
19 -s: 指定作为源地址匹配，必须是IP，取反，就加一个！
20 -d: 表示匹配目标地址
21 -p: 用于匹配协议的（这里的协议通常有3种，TCP/UDP/ICMP）
22 -i eth0: 从这块网卡流入的数据，流入一般在INPUT和PREROUTING上
23 -o eth0: 从这块网卡流出的数据，流出一一般在OUTPUT和POSTROUTING上
24
25 2. 【扩展匹配】
26 -p tcp: TCP协议的扩展。一般有三种扩展
27 --dport: 指定目标端口，--dport 21或者 --dport 21-23（此时表示21,22,23），不能表示非连续端口
28 --sport: 指定源端口
29
30 --tcp-flags: TCP的标志位（SYN,ACK, FIN,PSH, RST,URG）
31 对于它，一般要跟两个参数：
32 1. 检查的标志位
33 2. 必须为1的标志位
34 --tcpflags syn,ack,fin,rst syn = --syn
35 表示检查这4个位，这4个位中syn必须为1，其他的必须为0。所以这个意思就是用于检测三次握手的第一次包的。对于这种专门匹配第一包的SYN为1的包，还有一种简写方式，叫做--syn
36
37 -p udp: UDP协议的扩展
38 --dport
39 --sport
40
41 -p icmp: icmp数据报文的扩展
42 --icmp-type:
43 echo-request(请求回显)，一般用8 来表示
44 echo-reply（响应的数据包） 一般用0来表示
```

参考文章

- 1
- [ubuntu配置iptables](#)
- 2
- [netfilter/iptables全攻略](#)
- 3
- [iptables详解](#)

开放端口

```
1 # 在filter表里添加规则
2 -A INPUT -m state --state NEW -m tcp -p tcp --dport 8500 -j ACCEPT
3
4 # 重启iptables
5 $ service iptables restart
6
7 # 监听端口
8 $ nc -l-p 8500 &
9
10 # 检测是否开启完成
11 $ netstat -tunl|grep 8500
```

端口转发

```
1 # 按开放端口配置好后才可转发。
2 # 假设我们现在要将36.250.x.x的8500端口转发到192.168.2.55的80端口上
3
4 # filter表
5 -A INPUT -m state --state NEW -m tcp -p tcp --dport 8500 -j ACCEPT
6
7 # nat表
8 -A PREROUTING -d 36.250.x.x -m tcp -p tcp --dport 8500 -j DNAT --to-destination 192.168.2.55:80
9
10 # 测试一下，在浏览器输入 36.250.x.x:8500
11 # 就可以看到我们的内容了。
```

ip 命令

```
1 # ubuntu 临时修改网络,需要刷新一下
2 ip addr flush dev ens4
```

六、Shell命令

6.1 定时任务

命令执行顺序控制

```
1 1、&&
2 方式: command1 && command2
3 如果command1执行成功, 则执行command2
4
5 2、||
6 方式: command1 || command2
7 如果command1执行失败, 则执行command2
```

管道通信

```
1 $ ps aux | grep mysqld
2
3 # 将ps aux得到的结果传给grep命令
```

nohup

一般在~终端或一个SSH连接运行一个软件或服务, 该软件或服务的生命周期受终端/SSH连接影响, 关闭后就自动也停止。

为了将程序放在后台运行, 可以使用nohup命令

```
$ nohup 程序路径/程序名 &
```

alias

给常用的长命令取别名, 变成短的, 提高效率

```
1 # 查看现有别名
2 $ alias
3
4 # 添加别名
5 $ alias catgra='cat /var/lib/mysql/grastate.dat'
6
7 # 取消别名状态
8 # 比如我们叫ll, 系统默认给我们加了别名, ll='ls -l --color=auto', 也就是加上颜色效果
9 # 如果我们不要颜色效果, 可以这样使用转义符, 使用原生的命令
10 $ \ll somedir
```

column

一个很好用的命令, 经常用于管道符后, 进行文本展示

```
1 $ mount
2 sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
3 proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
4 devtmpfs on /dev type devtmpfs (rw,nosuid,size=3996196k,nr_inodes=999049,mode=755)
5 securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
6
7 # 使用后, t是按表格的形式展示, -s "e" 指定分隔符
8 $ mount | column -t
9 sysfs                on /sys                type sysfs            (rw,nosuid,nodev,noexec,relatime)
10 proc                 on /proc               type proc             (rw,nosuid,nodev,noexec,relatime)
11 devtmpfs             on /dev                type devtmpfs         (rw,nosuid,size=3996196k,nr_inodes=999049,mode=755)
12 securityfs           on /sys/kernel/security type securityfs       (rw,nosuid,nodev,noexec,relatime)
```

重定向

```
>/dev/null 2>&1 &
```

分析下这个语句

```
command >/dev/null 2>&1 &
```

执行command后的标准输出不在屏幕显示, 而是直接丢入/dev/null 垃圾桶, 如果有错误输出, 则重定向到标准输出。最后&表示在后台运行。

七、业务相关

查看虚机的CPU是否支持虚拟化

```
1 egrep '(vmx|svm)' /proc/cpuinfo
2 # 如果有标红的vmx (Intel) 、svm (AMD) 说明就支持, virtualbox不支持 (坑)
3
4 # 或者直接查看
5 vi /proc/cpuinfo
```

查看模块是否加载

```
lsmod | grep kvm
```

开启服务

```
1 # CentOS 6
2 service xx start
3
4 # CentOS 7
5 systemctl start libvirt
```

开机自启动服务

```
1 CentOS 6
2 chkconfig acpid on
3
4 # CentOS 7
5 systemctl enable libvirt
```

开启shh登录, 修改端口等配置

```
1 vi /etc/ssh/sshd_config
2 .....
3 Port 57891
4 PasswordAuthentication yes
5 ClientAliveInterval 60
6 ClientAliveCountMax 10
```

关闭防火墙

```
1 setenforce 0
2 service firewalld stop
3 chkconfig firewalld off
```

换yum源

```
1 yum install wget
2
3 mv /etc/yum.repos.d/CentOS-Base.repo /root/
4
5 # 下载yum源, 这里是CentOS6的, 请下载对应版本
6 wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-6.repo
7
8 yum clean all
9 yum makecache
```

推荐阅读

- 命令的艺术

转载于:https://www.cnblogs.com/wongbingming/p/10415781.html

墨 文章知识点与官方知识档案匹配，可进一步学习相关知识

云原生入门技能树 > 首页 > 概览 13188 人正在系统学习中

Linux 命令行

ChatGPT提问

linux命令行 快乐linux命令行, 各种linux的终端命令。	01-29
Linux常用命令总结 Linux 常用命令 awk / arch / bc / cp / clock -w / chmod / chkconfig / cal / cut /chmod / cat / crotable / clean / date / diff / df / du -sh / echo / exit / find / fdisk -l / gzip / Grep / Help / his...	Tattoo 822
Linux-系统管理常用命令 收集了常用的Linux系统管理命令, 以待日后查询或使用。 adduser命令 Linux adduser命令用于新增使用者帐号或更新预设的使用者资料。 adduser 与 useradd 指令为同一指令 (經由符号连结 symbolic link) 。 使用权限: 系统管理员。 adduser是增加使...	songyi10的博客 1159
SQL笔记 目录 管理MySQL 事务 创建、删除、选择数据库 其他 创建表, 删除表, CREATE, DROP ALTER TABLE, 添加、删除或修改列, 修改约束 查询数据 SELECT 插入 INSERT INTO 更新 UPDATE 删除 DELETE TOP, 规定要返回的记录数目 操作符 连接 ...	Bingmouss的博客 352
Java中创建不可变集合、Stream流、异常体系、日志框架 Java中创建不可变集合、Stream流、异常体系、日志框架。	asnowdream的博客 476
Linux学习笔记 # 目录 1. 系统概述 1.1. 系统目录结构 2. 文件属性及操作 2.1. 文件基本属性 2.2. 文件操作 3. 用户和用户组管理 4. 系统管理 4.1. 设备管理 4.2. 进程管理 进程管理 4.3. 系统监视与日志 4.4. 系统初始化过程分析 5. 网络管理 5.1. 网络接口配置 5.2. 常用指令 5...	Bazinga的博客 212
Linux命令大全.epub Linux命令行大全	09-07
linux命令大全.pdf linux命令行大全.pdf	03-15
linux命令大全(分模块、很清晰全面).pdf linux命令行大全(分模块、很清晰全面)	12-15
Linux命令大全.pdf 学好Linux从命令行开始, 只需要静下心来学习几天就可以学好了。	04-21
linux 命令行 linux命令行, 快速入门必备手册。有助于学习Linux基础	04-06
快乐的linux命令行_linux_ linux命令行学习资料 (快乐的Linux命令行 适合小白)	09-30
Linux命令行技术大全 非常详细通俗易懂的一本linux命令行入门书籍, 很适合初学者!	06-29
Linux命令行大全 《Linux命令行大全》主要介绍linux命令行的使用, 循序渐进, 深入浅出, 引导读者全面掌握命令行的使用方法。 《linux命令行大全》分为四部分。第一部分开始了对命令行基本语言的学习之旅, 包括命令结构、文件系统的...	08-26
快乐的linux命令行中英对照 快乐的linux命令行, 是The Linux Command Line的中文版, 可以登录网址: http://billie66.github.io/TLCL/访问在线的版本。 这里是根据最新网页内容导出为PDF文件的中英对照版。该网址的内容会不断完善不断更新, 这里...	12-14
快乐的 Linux 命令行.pdf Linux系统下的命令实战	08-28

Linux命令行的艺术

这是在 Linux 上工作时发现的有用的**命令行**使用小窍门和笔记的精髓。有些小窍门是很基础的，而有些是相当地特别、复杂、或者晦涩难懂。这篇文章不长，但是如果你可以使用并记得这里的所有内容，那么你就懂得很多了。

快乐Linux命令行

新手学习linux 循序渐进的一个好书。 微博名人推荐。

Linux命令行安裝Oracle12c

(1) 查看Linux系统的内存大小 [agssvrYX@inmshgis04 ~]\$ grep MemTotal /proc/meminfo MemTotal: 49426728 kB (2) 显示Linux系统上的文件系统的磁盘使用情况 [agssvrYX@inmshgis04 ~]\$ df -h Filesystem Size Used ...

linux命令行大全 最新发布

Linux 命令行大全如下： 1. ls:

“相关推荐”对你有帮助么？

🤔❄️

有

关于我们 招贤纳士 商务合作 寻求报道 400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

京公网安备11010502030143 京ICP备19004656号 京网文(2020) 1865-165号 经营性网站备案信息 北京互联网违法和不良信息举报中心 国家网信办网络举报中心 中国互联网举报中心 网络中国举报中心 网络违法犯罪举报网站 网络侵权投诉 实名认证 版权申诉 出版物许可证 营业执照

©1999-2023北京创新乐知网络技术有限公司

weixin_30305735

关注

暂无认证

0	-	184万+	77万+		
原创	周排名	总排名	访问		等级
4035	104	162	0	1141	
积分	粉丝	获赞	评论		收藏

热门文章

[JavaWeb开发使用jsp还是html做前端页面](#)
@ 15267

[yarn报错There appears to be trouble with your network connection. Retrying... @ 10942](#)

[C语言中的警告overflow in implicit constant conversion @ 9439](#)

[关于element-ui表格table设置header-class-name样式不起作用的原因分析 @ 7018](#)

[关于奥威亚自动录屏系统的设置使用小笔记 @ 6955](#)

您愿意向朋友推荐“博客详情页”吗？

强烈不推荐
不推荐
一般般
推荐
强烈推荐

最新文章

[postgress根据下标，获取数组的值](#)

[postgress根据正则表达式切分，并获取切分后的某一个值](#)

[\[python\]泡菜存储（pickle）](#)

2019年 711篇	2018年 779篇
2017年 709篇	2016年 550篇
2015年 419篇	2014年 330篇
2013年 281篇	2012年 217篇
2011年 140篇	2010年 113篇
2009年 89篇	2008年 79篇
2007年 51篇	2006年 49篇
2005年 19篇	2004年 4篇

Same IELTS,
faster results!

目录

Linux 命令笔记

- 一、目录文件
 - 1.1 目录文件日常操作
 - 1.2 文件处理
 - 1.3 文字处理
 - 1.4 文件重构
 - 1.5 其他命令
 - 1.6 文件查找
 - 1.7 文件传输/下载
 - 1.8 文件压缩
 - 1.9 rpm包管理
- 二、系统管理
 - 2.0 环境变量
 - 2.1 相关查询