目录

[1、 rz工具 2](#_Toc518919278)

[2、 History命令 3](#_Toc518919279)

[2.1 History命令的变量说明 3](#_Toc518919280)

[2.2 参数说明 4](#_Toc518919281)

[3、 Head 5](#_Toc518919282)

[4、 Tail命令 6](#_Toc518919283)

[5、 Sort 7](#_Toc518919284)

[6、 Uniq 9](#_Toc518919285)

[7、 tr 11](#_Toc518919286)

[8、 grep 12](#_Toc518919287)

[9、 Find 15](#_Toc518919288)

[10、 awk 15](#_Toc518919289)

[11、 sed 15](#_Toc518919290)

[12、 sed、awk统合应用 15](#_Toc518919291)

## rz工具

rz，sz是Linux/Unix同Windows进行ZModem文件传输的命令行工具。

优点就是不用再开一个sftp工具登录上去上传下载文件。

sz：将选定的文件发送（send）到本地机器

rz：运行该命令会弹出一个文件选择窗口，从本地选择文件上传到Linux服务器

rz,sz安装方式如下：

[root@mysqldb ~]# yum install lrzsz

Loaded plugins: fastestmirror

Loading mirror speeds from cached hostfile

\* base: mirrors.sohu.com

\* extras: mirrors.sohu.com

\* updates: mirrors.sohu.com

Resolving Dependencies

--> Running transaction check

---> Package lrzsz.x86\_64 0:0.12.20-36.el7 will be installed

--> Finished Dependency Resolution

Dependencies Resolved

====================================================================================================================================

Package Arch Version Repository Size

====================================================================================================================================

Installing:

lrzsz x86\_64 0.12.20-36.el7 base 78 k

Transaction Summary

====================================================================================================================================

Install 1 Package

Total download size: 78 k

Installed size: 181 k

Is this ok [y/d/N]: y

Downloading packages:

lrzsz-0.12.20-36.el7.x86\_64.rpm | 78 kB 00:00:00

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Installing : lrzsz-0.12.20-36.el7.x86\_64 1/1

Verifying : lrzsz-0.12.20-36.el7.x86\_64 1/1

Installed:

lrzsz.x86\_64 0:0.12.20-36.el7

Complete!

[root@mysqldb ~]#

使用方法如下：

[root@mysqldb ~]# rz

rz waiting to receive.

正在开始 zmodem 传输。 按 Ctrl+C 取消。

正在传输 mysql-5.5.60.tar.gz...

100% 20532 KB 5133 KB/s 00:00:04 0 错误

## History命令

History命令用于查看历史运行的命令

### 2.1 History命令的变量说明

修改history变量可以通过修改/etc/bash.bashrc

1. 添加日期

vi /etc/bashrc

HISTTIMEFORMAT="%F %T"

export HISTTIMEFORMAT

. /etc/bashrc <<==等于source /etc/bashrc

History <<==结果如下

448 2017-01-11 11:42:01 vim /etc/bash.bashrc

449 2017-01-11 11:42:45 source /etc/bash.bashrc

450 2017-01-11 11:40:18 history

451 2017-01-11 11:41:11 ls

452 2017-01-11 11:41:12 pwd

453 2017-01-11 11:41:35 history

1. 调整记录的命令的长度

vi /etc/bashrc

HISTSIZE=10

export HISTTIMEFORMAT

. /etc/bashrc <<==等于source /etc/bashrc

1. 剔除连续重复的条目

vi /etc/bashrc

export HISTCONTROL=ignoredups:erasedups

. /etc/bashrc <<==等于source /etc/bashrc

1. 不记录某个命令

在变量HISTCONTROL中添加ignorespace，在使用命令时在前面加一个空格。

vi /etc/bashrc

HISTCONTROL=ignorespace

. /etc/bashrc <<==等于source /etc/bashrc

### 2.2 参数说明

History有如下参数：

-c：清空当前历史命令；

-a：将历史命令缓冲区中命令写入历史命令文件中；

-r：将历史命令文件中的命令读入当前历史命令缓冲区；

-w：将当前历史命令缓冲区命令写入历史命令文件中；

-d: 删除某一条记录；

!\*:执行某一个条历史命令；

!!：执行上一条历史命令；

1、-w命令，将命令保存到

[root@localhost ~]# history

258 2018-06-26 14:45:59echo $HISTTIMEFORMAT

259 2018-06-26 14:52:25history | wc -l

260 2018-06-26 14:56:38source /etc/bashrc

261 2018-06-26 14:56:50echo $HISTCONTROL

262 2018-06-26 15:05:08vi /etc/bashrc

263 2018-06-26 15:08:10. /etc/bashrc

264 2018-06-26 15:21:16ls

265 2018-06-26 15:23:12history

[root@localhost ~]# history -w history2

[root@localhost ~]# cat history2

#1529995559

echo $HISTTIMEFORMAT

#1529995945

history | wc -l

#1529996198

source /etc/bashrc

#1529996210

echo $HISTCONTROL

#1529996708

vi /etc/bashrc

#1529996890

. /etc/bashrc

#1529997676

ls

#1529997792

history

#1529997806

history -w history2

[root@localhost ~]#

2、显示最近几条命令

[root@localhost ~]# history 3

269 2018-06-26 15:28:57ll

270 2018-06-26 15:29:02ls -a

271 2018-06-26 15:30:53history 3

## Head

显示文件的前几行

[root@mysqldb ~]# head --help

Usage: head [OPTION]... [FILE]...

Print the first 10 lines of each FILE to standard output.

With more than one FILE, precede each with a header giving the file name.

With no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short options too.

-c, --bytes=[-]K print the first K bytes of each file; ##显示前多少个字符可与“-”“+”结合

with the leading '-', print all but the last

K bytes of each file

-n, --lines=[-]K print the first K lines instead of the first 10; ##显示前多少个行可与“-”“+”结合

with the leading '-', print all but the last

K lines of each file

-q, --quiet, --silent never print headers giving file names

-v, --verbose always print headers giving file names

--help display this help and exit

--version output version information and exit

K may have a multiplier suffix:

b 512, kB 1000, K 1024, MB 1000\*1000, M 1024\*1024,

GB 1000\*1000\*1000, G 1024\*1024\*1024, and so on for T, P, E, Z, Y.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>

For complete documentation, run: info coreutils 'head invocation'

显示倒数第40行以前

[root@mysqldb ~]# ll | sort -k 5 -nr | head -n -40

-rw-r--r--. 1 root root 52364199 Apr 17 12:34 google-chrome-stable\_current\_x86\_64.rpm

-rw-r--r--. 1 root root 51868499 Jun 20 14:21 mysql-5.7.22.tar.gz

-rw-r--r--. 1 root root 21025041 Jun 20 13:55 mysql-5.5.60.tar.gz

-rw-r--r--. 1 root root 20143759 Dec 7 2015 Python-3.5.1.tgz

显示前6行

[root@mysqldb ~]# ll | sort -k 5 -nr | head -n +6

-rw-r--r--. 1 root root 52364199 Apr 17 12:34 google-chrome-stable\_current\_x86\_64.rpm

-rw-r--r--. 1 root root 51868499 Jun 20 14:21 mysql-5.7.22.tar.gz

-rw-r--r--. 1 root root 21025041 Jun 20 13:55 mysql-5.5.60.tar.gz

-rw-r--r--. 1 root root 20143759 Dec 7 2015 Python-3.5.1.tgz

-rw-r--r--. 1 root root 4352897 Apr 9 15:32 tcl8.5.0-src.tar.gz

-rw-r--r--. 1 root root 1737022 Mar 27 00:04 redis-4.0.9.tar.gz

[root@mysqldb ~]# ll | sort -k 5 -nr | head -n 6

-rw-r--r--. 1 root root 52364199 Apr 17 12:34 google-chrome-stable\_current\_x86\_64.rpm

-rw-r--r--. 1 root root 51868499 Jun 20 14:21 mysql-5.7.22.tar.gz

-rw-r--r--. 1 root root 21025041 Jun 20 13:55 mysql-5.5.60.tar.gz

-rw-r--r--. 1 root root 20143759 Dec 7 2015 Python-3.5.1.tgz

-rw-r--r--. 1 root root 4352897 Apr 9 15:32 tcl8.5.0-src.tar.gz

-rw-r--r--. 1 root root 1737022 Mar 27 00:04 redis-4.0.9.tar.gz

## Tail命令

Tail命令用于显示内容的最近10行（默认）。

Tail –f与-s N 用于显示文件最新的变化；

[root@localhost ~]# tail -f test1

this is 1 for.

this is 2 for.

this is 3 for.

this is 4 for

Tail –n [-5/5/+5] 显示最后多少行

[root@localhost ~]# tail -n 5 /etc/passwd <<==显示最后5行，与tail –n -5一样

polkitd:x:999:998:User for polkitd:/:/sbin/nologin

avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin

avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin

postfix:x:89:89::/var/spool/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

[root@localhost ~]#

[root@localhost ~]#

[root@localhost ~]#

[root@localhost ~]# tail -n -5 /etc/passwd  <<==显示最后5行，与tail –n 5一样

polkitd:x:999:998:User for polkitd:/:/sbin/nologin

avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin

avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin

postfix:x:89:89::/var/spool/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

[root@localhost ~]#

[root@localhost ~]# tail -n +18 /etc/passwd <<==显示第18行到最后一行

postfix:x:89:89::/var/spool/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

Tail –c 显示最后多少字符

[root@localhost ~]# tail -c 90 /etc/passwd <<==显示最后90个字符，与tail –c -90一样

/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

[root@localhost ~]# tail -c -90 /etc/passwd <<==显示最后90个字符，与tail –c 90一样

/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

[root@localhost ~]#

[root@localhost ~]# tail -c +690 /etc/passwd <<==显示第690个字符以后的内容hi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin

postfix:x:89:89::/var/spool/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

[root@localhost ~]#

## Sort

[root@mysqldb ~]# sort --help

Usage: sort [OPTION]... [FILE]...

or: sort [OPTION]... --files0-from=F

Write sorted concatenation of all FILE(s) to standard output.

Mandatory arguments to long options are mandatory for short options too.

Ordering options:

-b, --ignore-leading-blanks ignore leading blanks <<=忽略空格

-d, --dictionary-order consider only blanks and alphanumeric characters <<=只考虑空格数字字符

-f, --ignore-case fold lower case to upper case characters <<=忽略大小写

-g, --general-numeric-sort compare according to general numerical value

-i, --ignore-nonprinting consider only printable characters

-M, --month-sort compare (unknown) < 'JAN' < ... < 'DEC' <<=比较月份

-h, --human-numeric-sort compare human readable numbers (e.g., 2K 1G) <<=智能比较数字

-n, --numeric-sort compare according to string numerical value <<=比较数值

-R, --random-sort sort by random hash of keys

--random-source=FILE get random bytes from FILE

-r, --reverse reverse the result of comparisons

--sort=WORD sort according to WORD:

general-numeric -g, human-numeric -h, month -M,

numeric -n, random -R, version -V

-V, --version-sort natural sort of (version) numbers within text

Other options:

--batch-size=NMERGE merge at most NMERGE inputs at once;

for more use temp files

-c, --check, --check=diagnose-first check for sorted input; do not sort <<=检查输入是否排序

-C, --check=quiet, --check=silent like -c, but do not report first bad line

--compress-program=PROG compress temporaries with PROG;

decompress them with PROG -d

--debug annotate the part of the line used to sort,

and warn about questionable usage to stderr

--files0-from=F read input from the files specified by

NUL-terminated names in file F;

If F is - then read names from standard input

-k, --key=KEYDEF sort via a key; KEYDEF gives location and type <<=指定排序字段

-m, --merge merge already sorted files; do not sort

-o, --output=FILE write result to FILE instead of standard output

-s, --stable stabilize sort by disabling last-resort comparison

-S, --buffer-size=SIZE use SIZE for main memory buffer

-t, --field-separator=SEP use SEP instead of non-blank to blank transition <<=指定分隔符

-T, --temporary-directory=DIR use DIR for temporaries, not $TMPDIR or /tmp;

multiple options specify multiple directories

--parallel=N change the number of sorts run concurrently to N

-u, --unique with -c, check for strict ordering; <<=忽略相同的行

without -c, output only the first of an equal run

-z, --zero-terminated end lines with 0 byte, not newline

--help display this help and exit

--version output version information and exit

使用-h进行智能排序，-r反序

[root@mysqldb ~]# du -sh \* | sort -h -r

1.4G mysql-5.5.60

416M mysql-5.7.22

184M Python-3.5.1

135M back

68M redis-4.0.9

68M redis

50M mysql-5.7.22.tar.gz

50M google-chrome-stable\_current\_x86\_64.rpm

29M tcl8.5.0

21M mysql-5.5.60.tar.gz

20M Python-3.5.1.tgz

19M nginx-1.13.12

19M nginx

使用-k指定对那一个字段进行排序，-n按数字进行排序

[root@mysqldb ~]# ll | sort -n -k 5 –r <<=对第五列按数字进行反向排序

-rw-r--r--. 1 root root 52364199 Apr 17 12:34 google-chrome-stable\_current\_x86\_64.rpm

-rw-r--r--. 1 root root 51868499 Jun 20 14:21 mysql-5.7.22.tar.gz

-rw-r--r--. 1 root root 21025041 Jun 20 13:55 mysql-5.5.60.tar.gz

-rw-r--r--. 1 root root 20143759 Dec 7 2015 Python-3.5.1.tgz

-rw-r--r--. 1 root root 4352897 Apr 9 15:32 tcl8.5.0-src.tar.gz

-rw-r--r--. 1 root root 1737022 Mar 27 00:04 redis-4.0.9.tar.gz

-rw-r--r--. 1 root root 1016311 Apr 10 22:19 nginx-1.13.12.tar.gz

-rw-r--r--. 1 root root 542422 Sep 16 2016 libappindicator-12.10.0-11.el7.src.rpm

-rw-r--r--. 1 root root 37376 Sep 16 2016 libappindicator-gtk3-12.10.0-11.el7.x86\_64.rpm

-rw-r--r--. 1 root root 9116 Apr 18 00:26 mysql57-community-release-el7-8.noarch.rpm

使用-M按月分排序

[root@mysqldb ~]# ll | sort -M -k 6

total 149632

drwxr-xr-x. 35 7161 31415 4096 Mar 4 21:49 mysql-5.7.22

-rw-r--r--. 1 root root 1737022 Mar 27 00:04 redis-4.0.9.tar.gz

drwxrwxr-x. 17 mysql mysql 4096 Apr 17 23:33 Python-3.5.1

drwxr-xr-x. 4 root root 34 Apr 23 18:12 rpmbuild

drwxr-xr-x. 9 1001 1001 186 Apr 17 23:49 nginx-1.13.12

-rw-------. 1 root root 1263 Apr 17 22:24 anaconda-ks.cfg

-rw-r--r--. 1 root root 1016311 Apr 10 22:19 nginx-1.13.12.tar.gz

-rw-r--r--. 1 root root 4352897 Apr 9 15:32 tcl8.5.0-src.tar.gz

使用-t指定分隔符

[root@mysqldb ~]# cat sort-test | sort -nrk 2 -t ":"

adf:456:93

aaa:123:33

ass:56:34

aab:45:22

bbb:34:22

avf:34:11

cic:32:66

## Uniq

[root@mysqldb ~]# uniq --help

Usage: uniq [OPTION]... [INPUT [OUTPUT]]

Filter adjacent matching lines from INPUT (or standard input),

writing to OUTPUT (or standard output).

With no options, matching lines are merged to the first occurrence.

Mandatory arguments to long options are mandatory for short options too.

-c, --count prefix lines by the number of occurrences 在每列旁边显示该行重复出现的次数

-d, --repeated only print duplicate lines, one for each group 仅显示重复出现的行列

-D, --all-repeated[=METHOD] print all duplicate lines

groups can be delimited with an empty line

METHOD={none(default),prepend,separate}

-f, --skip-fields=N avoid comparing the first N fields 忽略比较指定的栏位

--group[=METHOD] show all items, separating groups with an empty line

METHOD={separate(default),prepend,append,both}

-i, --ignore-case ignore differences in case when comparing 忽略大小写

-s, --skip-chars=N avoid comparing the first N characters 忽略比较指定的字符

-u, --unique only print unique lines 仅显示出一次的行列

-z, --zero-terminated end lines with 0 byte, not newline

-w, --check-chars=N compare no more than N characters in lines 指定要比较的字符

--help display this help and exit

--version output version information and exit

A field is a run of blanks (usually spaces and/or TABs), then non-blank

characters. Fields are skipped before chars.

Note: 'uniq' does not detect repeated lines unless they are adjacent.

You may want to sort the input first, or use 'sort -u' without 'uniq'.

Also, comparisons honor the rules specified by 'LC\_COLLATE'.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>

For complete documentation, run: info coreutils 'uniq invocation'

Uniq与sort结合使用，对文件进行统计和排序

[root@mysqldb ~]# cat uniq-test | gawk '{print $1}' | sort -r | uniq -c | sort -n

1 hang

1 tao

1 tt

2 hong

3 tang

[root@mysqldb ~]# cat uniq-test

tang 1987 2

hong 2012 3

tao 2012 2

tang 1990 3

tang 1990 2

hang 2011 3

hong 2010 2

tt 2010 3

## tr

从标准输入中转换、挤压和/或删除字符

[root@mysqldb ~]# tr --help

Usage: tr [OPTION]... SET1 [SET2]

Translate, squeeze, and/or delete characters from standard input,

writing to standard output.

-c, -C, --complement use the complement of SET1

-d, --delete delete characters in SET1, do not translate ##删除字符串

-s, --squeeze-repeats  replace each input sequence of a repeated character

that is listed in SET1 with a single occurrence

of that character

-t, --truncate-set1 first truncate SET1 to length of SET2 ##替换字符串

--help display this help and exit

--version output version information and exit

SETs are specified as strings of characters. Most represent themselves.

Interpreted sequences are:

\NNN character with octal value NNN (1 to 3 octal digits)

\\ backslash

\a audible BEL

\b backspace

\f form feed

\n new line

\r return

\t horizontal tab

\v vertical tab

CHAR1-CHAR2 all characters from CHAR1 to CHAR2 in ascending order

[CHAR\*] in SET2, copies of CHAR until length of SET1

[CHAR\*REPEAT] REPEAT copies of CHAR, REPEAT octal if starting with 0

[:alnum:] all letters and digits

[:alpha:] all letters

[:blank:] all horizontal whitespace

[:cntrl:] all control characters

[:digit:] all digits

[:graph:] all printable characters, not including space

[:lower:] all lower case letters

[:print:] all printable characters, including space

[:punct:] all punctuation characters

[:space:] all horizontal or vertical whitespace

[:upper:] all upper case letters

[:xdigit:] all hexadecimal digits

[=CHAR=] all characters which are equivalent to CHAR

Translation occurs if -d is not given and both SET1 and SET2 appear.

-t may be used only when translating. SET2 is extended to length of

SET1 by repeating its last character as necessary. Excess characters

of SET2 are ignored. Only [:lower:] and [:upper:] are guaranteed to

expand in ascending order; used in SET2 while translating, they may

only be used in pairs to specify case conversion. -s uses SET1 if not

translating nor deleting; else squeezing uses SET2 and occurs after

translation or deletion.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>

For complete documentation, run: info coreutils 'tr invocation'

-d删除指定字符

[root@mysqldb ~]# cat tr-test

tang is a hong.name is

[root@mysqldb ~]#

[root@mysqldb ~]#

[root@mysqldb ~]# tr -d is <tr-test

tang a hong.name

替换指定的字符

[root@mysqldb ~]# cat tr-test

tang is a hong.name is

[root@mysqldb ~]#

[root@mysqldb ~]#

[root@mysqldb ~]# tr -t " " "\n"<tr-test

tang

is

a

hong.name

is

## grep

搜索文件或输入的内容。常用参数有  
-v 显示没有匹配到的行

-i 忽略大小写

-n 显示行号

-s 忽略错误

-l 只显示文件名

-c 显示匹配的次数

-r 目录递归搜索

-F 不使用正则表达式，速度快

[root@mysqldb ~]# grep --help

Usage: grep [OPTION]... PATTERN [FILE]...

Search for PATTERN in each FILE or standard input.

PATTERN is, by default, a basic regular expression (BRE).

Example: grep -i 'hello world' menu.h main.c

Regexp selection and interpretation:

-E, --extended-regexp PATTERN is an extended regular expression (ERE) ##使用扩展正则表达式

-F, --fixed-strings PATTERN is a set of newline-separated fixed strings ##不使用正则表达式

-G, --basic-regexp PATTERN is a basic regular expression (BRE) ##使用基本正则表达式，默认

-P, --perl-regexp PATTERN is a Perl regular expression ####使用Perl正则表达式

-e, --regexp=PATTERN use PATTERN for matching

-f, --file=FILE obtain PATTERN from FILE

-i, --ignore-case ignore case distinctions ##忽略大小写

-w, --word-regexp force PATTERN to match only whole words ##强制进行全单词匹配

-x, --line-regexp force PATTERN to match only whole lines ##强制进行整行匹配

-z, --null-data a data line ends in 0 byte, not newline

Miscellaneous:

-s, --no-messages suppress error messages ##忽略错误信息

-v, --invert-match select non-matching lines ##选择没有匹配的行

-V, --version display version information and exit

--help display this help text and exit

Output control:

-m, --max-count=NUM stop after NUM matches

-b, --byte-offset print the byte offset with output lines ##输出匹配字节的偏移

-n, --line-number print line number with output lines ##输出行号

--line-buffered flush output on every line

-H, --with-filename print the file name for each match

-h, --no-filename suppress the file name prefix on output

--label=LABEL use LABEL as the standard input file name prefix

-o, --only-matching show only the part of a line matching PATTERN ##只显示匹配的内容

-q, --quiet, --silent suppress all normal output

--binary-files=TYPE assume that binary files are TYPE;

TYPE is 'binary', 'text', or 'without-match'

-a, --text equivalent to --binary-files=text

-I equivalent to --binary-files=without-match

-d, --directories=ACTION how to handle directories;

ACTION is 'read', 'recurse', or 'skip'

-D, --devices=ACTION how to handle devices, FIFOs and sockets;

ACTION is 'read' or 'skip'

-r, --recursive like --directories=recurse

-R, --dereference-recursive

likewise, but follow all symlinks

--include=FILE\_PATTERN

search only files that match FILE\_PATTERN

--exclude=FILE\_PATTERN

skip files and directories matching FILE\_PATTERN

--exclude-from=FILE skip files matching any file pattern from FILE

--exclude-dir=PATTERN directories that match PATTERN will be skipped.

-L, --files-without-match print only names of FILEs containing no match

-l, --files-with-matches print only names of FILEs containing matches ##只打印出文件名

-c, --count print only a count of matching lines per FILE ##只打印文件匹配的行数

-T, --initial-tab make tabs line up (if needed)

-Z, --null print 0 byte after FILE name

Context control:

-B, --before-context=NUM print NUM lines of leading context

-A, --after-context=NUM print NUM lines of trailing context

-C, --context=NUM print NUM lines of output context

-NUM same as --context=NUM

--group-separator=SEP use SEP as a group separator

--no-group-separator use empty string as a group separator

--color[=WHEN],

--colour[=WHEN] use markers to highlight the matching strings;

WHEN is 'always', 'never', or 'auto'

-U, --binary do not strip CR characters at EOL (MSDOS/Windows)

-u, --unix-byte-offsets report offsets as if CRs were not there

(MSDOS/Windows)

'egrep' means 'grep -E'. 'fgrep' means 'grep -F'.

Direct invocation as either 'egrep' or 'fgrep' is deprecated.

When FILE is -, read standard input. With no FILE, read . if a command-line

-r is given, - otherwise. If fewer than two FILEs are given, assume -h.

Exit status is 0 if any line is selected, 1 otherwise;

if any error occurs and -q is not given, the exit status is 2.

Report bugs to: bug-grep@gnu.org

GNU Grep home page: <http://www.gnu.org/software/grep/>

General help using GNU software: <http://www.gnu.org/gethelp/>

在目录中查找文件内容包含‘log-bin’的文件，并列出文件名 用-c参数可以显示每个文件匹配多少次

[root@mysqldb support-files]# grep -rsl "log-bin" \*

my-huge.cnf

my-huge.cnf.sh

my-innodb-heavy-4G.cnf

my-innodb-heavy-4G.cnf.sh

my-large.cnf

my-large.cnf.sh

my-medium.cnf

my-medium.cnf.sh

my-small.cnf

my-small.cnf.sh

[root@mysqldb support-files]# grep -rsc "log-bin" \* | sort -t ":" -nr -k 2 | head -n 5

my-medium.cnf.sh:2

my-medium.cnf:2

my-large.cnf.sh:2

my-large.cnf:2

my-huge.cnf.sh:2

正则表达试应用

匹配出现过两次a的后面是b的行

[root@mysqldb ~]# cat sort-test | grep 'a\{2\}b'

aab:45:22

[root@mysqldb ~]# cat sort-test

aaa:123:33

bbb:34:22

cic:32:66

aab:45:22

ass:56:34

adf:456:93

avf:34:11

匹配两个以上的a后面是b的行

[root@mysqldb ~]# cat sort-test | grep 'a\{2,\}b'

aab:45:22

aaab:45:22

aaaab:45:22

aaaaab:45:22

[root@mysqldb ~]#

匹配两个到三个的a后面是b，前面是c和行

[root@mysqldb ~]# cat sort-test | grep 'ca\{2,3\}b'

caab:45:22

caaab:45:22

使用-E启用扩展的正则表达式，与加\一样，或使用egrep，

[root@mysqldb ~]# cat sort-test | grep -E 'bb|ca{2}b'

bbb:34:22

caab:45:22

使用扩展的正则表达式进行逻辑运算，匹配bb或caab

[root@mysqldb ~]# cat sort-test | grep -E 'bb|caab'

bbb:34:22

caab:45:22

[root@mysqldb ~]#

## Find

[root@mysqldb ~]# find --help

Usage: find [-H] [-L] [-P] [-Olevel] [-D help|tree|search|stat|rates|opt|exec] [path...] [expression]

default path is the current directory; default expression is -print

expression may consist of: operators, options, tests, and actions:

operators (decreasing precedence; -and is implicit where no others are given):

( EXPR ) ! EXPR -not EXPR EXPR1 -a EXPR2 EXPR1 -and EXPR2

EXPR1 -o EXPR2 EXPR1 -or EXPR2 EXPR1 , EXPR2

positional options (always true): -daystart -follow -regextype

normal options (always true, specified before other expressions):

-depth --help -maxdepth LEVELS -mindepth LEVELS -mount -noleaf

--version -xautofs -xdev -ignore\_readdir\_race -noignore\_readdir\_race

tests (N can be +N or -N or N): -amin N -anewer FILE -atime N -cmin N

-cnewer FILE -ctime N -empty -false -fstype TYPE -gid N -group NAME

-ilname PATTERN -iname PATTERN -inum N -iwholename PATTERN -iregex PATTERN

-links N -lname PATTERN -mmin N -mtime N -name PATTERN -newer FILE

-nouser -nogroup -path PATTERN -perm [-/]MODE -regex PATTERN

-readable -writable -executable

-wholename PATTERN -size N[bcwkMG] -true -type [bcdpflsD] -uid N

-used N -user NAME -xtype [bcdpfls]

-context CONTEXT

actions: -delete -print0 -printf FORMAT -fprintf FILE FORMAT -print

-fprint0 FILE -fprint FILE -ls -fls FILE -prune -quit

-exec COMMAND ; -exec COMMAND {} + -ok COMMAND ;

-execdir COMMAND ; -execdir COMMAND {} + -okdir COMMAND ;

Report (and track progress on fixing) bugs via the findutils bug-reporting

page at http://savannah.gnu.org/ or, if you have no web access, by sending

email to <bug-findutils@gnu.org>.

查找文件使用-o表或， –name 按文件名查找 –iname可忽略大小写

[root@mysqldb ~]# find /root -name \*mysql\*.cnf -o -name \*mysql\*.sh

/root/mysql-5.5.60/mysql-test/include/default\_mysqld.cnf

/root/mysql-5.5.60/packaging/rpm-oel/mysql\_config.sh

/root/mysql-5.5.60/scripts/msql2mysql.sh

/root/mysql-5.5.60/scripts/mysql\_config.sh

/root/mysql-5.5.60/scripts/mysql\_convert\_table\_format.sh

/root/mysql-5.5.60/scripts/mysql\_find\_rows.sh

/root/mysql-5.5.60/scripts/mysql\_fix\_extensions.sh

/root/mysql-5.5.60/scripts/mysql\_install\_db.sh

/root/mysql-5.5.60/scripts/mysql\_secure\_installation.sh

/root/mysql-5.5.60/scripts/mysql\_setpermission.sh

/root/mysql-5.5.60/scripts/mysql\_zap.sh

/root/mysql-5.5.60/scripts/mysqlaccess.sh

按文件名或路径查找

[root@mysqldb ~]# find /root -path \*local\*

/root/Python-3.5.1/Doc/library/locale.rst

/root/Python-3.5.1/aclocal.m4

/root/Python-3.5.1/Objects/stringlib/localeutil.h

/root/Python-3.5.1/Modules/\_ctypes/libffi/aclocal.m4

/root/Python-3.5.1/Modules/\_localemodule.c

/root/Python-3.5.1/Modules/Setup.local

/root/Python-3.5.1/Modules/\_localemodule.o

/root/Python-3.5.1/Lib/locale.py

/root/Python-3.5.1/Lib/\_bootlocale.py

/root/Python-3.5.1/Lib/\_threading\_local.py

/root/Python-3.5.1/Lib/test/test\_urllib2\_localnet.py

/root/Python-3.5.1/Lib/test/test\_threading\_local.py

/root/Python-3.5.1/Lib/test/test\_locale.py

查找不是txt文件

find /home ! -name "\*.txt"

--type 查找文件类型 -maxdepth 查找深度

**f**普通文件

**l**符号连接

**d** 目录

**c**字符设备

**b**块设备

**s**套接字

**p**Fifo

[root@mysqldb ~]# find -type d -maxdepth 2

find: warning: you have specified the -maxdepth option after a non-option argument -type, but options are not positional (-maxdepth affects tests specified before it as well as those specified after it). Please specify options before other arguments.

.

./Python-3.5.1

./Python-3.5.1/Mac

./Python-3.5.1/Doc

./Python-3.5.1/Parser

./Python-3.5.1/Objects

./Python-3.5.1/Modules

./Python-3.5.1/Lib

./Python-3.5.1/PC

./Python-3.5.1/Programs

./Python-3.5.1/PCbuild

./Python-3.5.1/Tools

./Python-3.5.1/Misc

**按时间查找**

**访问时间**（-atime/天，-amin/分钟）：用户最近一次访问时间。

**修改时间**（-mtime/天，-mmin/分钟）：文件最后一次修改时间。

**变化时间**（-ctime/天，-cmin/分钟）：文件数据元（例如权限等）最后一次修改时间。

[root@mysqldb ~]# find -type f -mtime -7 ##查找最近7天修改过的文件

./.bash\_history

./uniq-test

./.lesshst

./.mysql\_history

./tr-test

./messages-20180605

./sort-test

[root@mysqldb ~]# find -type f -maxdepth 2 -mtime +200 ##查找超过200天没修改过的文件

find: warning: you have specified the -maxdepth option after a non-option argument -type, but options are not

./.bash\_logout

./.bash\_profile

./.bashrc

./.cshrc

./.tcshrc

./Python-3.5.1.tgz

./Python-3.5.1/install-sh

./Python-3.5.1/config.guess

./Python-3.5.1/configure.ac

./Python-3.5.1/pyconfig.h.in

./Python-3.5.1/aclocal.m4

./Python-3.5.1/setup.py

./Python-3.5.1/README

./Python-3.5.1/config.sub

./Python-3.5.1/configure

./Python-3.5.1/LICENSE

./Python-3.5.1/Makefile.pre.in

./libappindicator-12.10.0-11.el7.src.rpm

./libappindicator-gtk3-12.10.0-11.el7.x86\_64.rpm

find . -type f -size 文件大小单元

文件大小单元：

**b** —— 块（512字节）

**c** —— 字节

[**w**](http://man.linuxde.net/w) —— 字（2字节）

**k** —— 千字节

**M** —— 兆字节

**G** —— 吉字节

[root@mysqldb ~]# find -type f -size +100M 查找大于100M的文件

./mysql-5.5.60/libmysqld/libmysqld.a

[root@mysqldb ~]# find -type f -size -1k 查找1K的文件

对查找的文件执行操作

删除当前目录下所有.txt文件

find . -type f -name "\*.txt" -delete

-perm按权限查找

找出当前目录下权限不是644的php文件

find . -type f -name "\*.php" ! -perm 644

按用户查找

找出当前目录用户tom拥有的所有文件

find . -type f -user tom

按用户组查找

找出当前目录用户组sunk拥有的所有文件

find . -type f -group sunk

使用-exec 与 –ok一样

找出当前目录下所有root的文件，并把所有权更改为用户tom

find .-type f -user root -exec [chown](http://man.linuxde.net/chown) tom {} \;

上例中，**{}** 用于与**-exec**选项结合使用来匹配所有文件，然后会被替换为相应的文件名。

查找当前目录或者子目录下所有.txt文件，但是跳过子目录sk

-path "./sk" -prune

find . -path "./sk" -prune -o -name "\*.txt" -print

列出所有长度为零的文件

find . -empty

## awk

## sed

## sed、awk统合应用

对下面的内容按单词排序

The months of learning in Old Boy education are the few months that I think the time efficient is the most.I had also studied at other training institutions before, but I was hard to understand what the tutor said and hard to follow. It was just too much to learn with no outline.

[root@oldboy ~]# sed 's#[,\.]##g' <oldboy.log|tr " " "\n"|sort|uniq -c|sort -rn|head -5

4 the

3 to

2 was

2 months

2 I

[root@oldboy ~]# tr " ," "\n" <oldboy.log|awk '{S[$1]++}END{for(k in S) print S[k],k}'|sort -rn|head -5

4 the

3 to

2 was

2 months

2 I

[root@oldboy ~]# awk -F "[ ,.]+" '{for(i=1;i<NF;i++)S[$i]++}END{for(k in S) print S[k],k}' oldboy.log |sort -rn|head -5

4 the

3 to

3 I

2 was

2 months

[root@oldboy ~]# sed 's#[,. ]##g' oldboy.log|grep -o "."|sort|uniq -c|sort -rn|head -5

33 t

20 o

19 e

18 n

17 i

[root@oldboy ~]# sed 's#[,. ]##g' oldboy.log|grep -o "."|awk '{S[$1]++}END{for(k in S) print S[k],k}'|sort -rn|head -5

33 t

20 o

19 e

18 n

17 i

[root@oldboy ~]# sed 's#[,. ]##g' oldboy.log|awk -F "" '{for(i=1;i<NF;i++)S[$i]++}END{for(k in S) print S[k],k}'|sort -rn|head -5

33 t

20 o

18 n

18 e

17 i

## wget

## curl