

Linux 复杂命令

1. **find** *pectin[perm exec ctime type iname name]*果胶

-perm mode

File's permission bits are exactly mode (octal or symbolic).

-prune True; if the file is a directory, do not descend into it.

-exec command ;

Execute command; true if 0 status is returned.

-ctime n

File's status was last changed n*24 hours ago.

-type c

File is of type c:

b block (buffered) special

c character (unbuffered) special

d directory

p named pipe (FIFO)

f regular file

l symbolic link; this is never true if the -L option or the -follow option is in effect, unless the symbolic link is broken. If you want to search for symbolic links when -L is in effect, use -xtype.

s socket

D door (Solaris)

-name pattern

Base of file name (the path with the leading directories removed) matches shell pattern pattern.

-iname pattern

Like -name, but the match is case insensitive.

2. **grep** *wife BrAnch 女方*

Regex selection and interpretation:

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

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

-i, --ignore-case ignore case distinctions

-w, --word-regexp force PATTERN to match only whole words

Miscellaneous:

-v, --invert-matchselect non-matching lines

Output control:

-n, --line-number print line number with output lines

--line-bufferedflush output on every line

-H, --with-filenameprint the file name for each match

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

-c, --countprint only a count of matching lines per FILE

Context control:

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

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

3. **sed** *finer pig hand basic!*{5=\$+~}好的猪手,基本上有5个特征

OPTION:

-f script-file, --file=script-file

add the contents of script-file to the commands to be executed

-i[SUFFIX], --in-place[=SUFFIX]

edit files in place (makes backup if SUFFIX supplied)

-n, --quiet, --silent

suppress automatic printing of pattern space

-e script, --expression=script

add the script to the commands to be executed

-r, --regexp-extended

use extended regular expressions in the script.

COMMAND SYNOPSIS

b label

Branch to label; if label is omitted, branch to end of script.

**a **

text Append text, which has each embedded newline preceded by a backslash.

s/regexp/replacement/

Attempt to match regexp against the pattern space. If successful, replace that portion matched with replacement. The replacement

may contain the special character & to refer to that portion of the pattern space which matched, and the special escapes \1 through

\9 to refer to the corresponding matching sub-expressions in the regexp.

**i **

text Insert text, which has each embedded newline preceded by a backslash.

**c **

text Replace the selected lines with text, which has each embedded newline preceded by a backslash.

! After the address (or address-range), and before the command, a **!** may be inserted, which specifies that the command shall only be executed

if the address (or address-range) does not match.

= Print the current line number.

first~step

Match every step'th line starting with line first. For example, ``sed -n 1~2p'' will print all the odd-numbered lines in the input

stream, and the address 2~5 will match every fifth line, starting with the second. first can be zero; in this case, sed operates as

if it were equal to step. (This is an extension.)

\$ Match the last line.

addr1,+N

Will match addr1 and the N lines following addr1.

addr1,~N

Will match addr1 and the lines following addr1 until the next line whose input line number is a multiple of N.

first~step

Match every step'th line starting with line first. For example, ``sed -n 1~2p'' will print all the odd-numbered lines in the input

stream, and the address 2~5 will match every fifth line, starting with the second. first can be zero; in this case, sed operates as

if it were equal to step. (This is an extension.)

p Print the current pattern space.

P Print up to the first embedded newline of the current pattern space.

**i **

text Insert text, which has each embedded newline preceded by a backslash.

g G Copy/append hold space to pattern space.

h H Copy/append pattern space to hold space.

**a **

text Append text, which has each embedded newline preceded by a backslash.

n N Read/append the next line of input into the pattern space.

d Delete pattern space. Start next cycle.

D If pattern space contains no newline, start a normal new cycle as if the d command was issued. Otherwise, delete text in the pattern

space up to the first newline, and restart cycle with the resultant pattern space, without reading a new line of input.

4. **sort** *grind tomsk 对托木斯克小镇进行碾压*

- g, --general-numeric-sort
compare according to general numerical value
- r, --reverse
reverse the result of comparisons
- i, --ignore-nonprinting
consider only printable characters
- n, --numeric-sort
compare according to string numerical value
- d, --dictionary-order
consider only blanks and alphanumeric characters
- t, --field-separator=SEP
use SEP instead of non-blank to blank transition
- o, --output=FILE
write result to FILE instead of standard output
- m, --merge
merge already sorted files; do not sort
- S, --buffer-size=SIZE
use SIZE for main memory buffer
- k, --key=KEYDEF
sort via a key; KEYDEF gives location and type

5. **uniq** *cud 吐*

- c, --count
prefix lines by the number of occurrences
- u, --unique
only print unique lines
- d, --repeated
only print duplicate lines, one for each group

6. **awk** *print2f[ff] wife's[while if for else switch] bed[break exit default] case 打印老婆的床盒子*

-f program-file

-F fs

Control Statements

The control statements are as follows:

```
if (condition) statement [ else statement ]
while (condition) statement
do statement while (condition)
for (expr1; expr2; expr3) statement
for (var in array) statement
break
continue
delete array[index]
delete array
exit [ expression ]
{ statements }
switch (expression) {
case value|regex : statement
...
[ default: statement ]
}
```

I/O Statements

The input/output statements are as follows:

print Print the current record. The output record is terminated with the value of ORS.

7. **readelf** *IS heralds IS 的传令官*

SYNOPSIS

```
[-l|--histogram]
[-S|--section-headers|--sections]

[-h|--file-header]
[-e|--headers]
[-r|--relocs]
[-a|--all]
[-l|--program-headers|--segments]
[-d|--dynamic]
[-s|--syms|--symbols]
```

elffile...

8. **objdump** *hex drafts 十六进制草图*

SYNOPSIS

`[-h|--section-headers|--headers]`

`[-e|--debugging-tags]`

`[-x|--all-headers]`

`[-d|--disassemble]`

`[-r|--reloc]`

`[-a|--archive-headers]`

`[-f|--file-headers]`

`[-t|--syms]`

`[-s|--full-contents]`

`objfile...`

9. **gcc** *DWg cOmpoSE fILm 刀哥编电影*

`-c` Compile or assemble the source files, but do not link.

`-O`

`-O1` Optimize.

`-O2` Optimize even more.

`-O3` Optimize yet more.

These `-m` options are defined for the x86 family of computers.

`-march=cpu-type`

Generate instructions for the machine type `cpu-type`. In contrast to `-mtune=cpu-type`, which merely tunes the generated code for the

The choices for `cpu-type` are:

`i386`

Original Intel i386 CPU.

`i686`

When used with `-march`, the Pentium Pro instruction set is used, so the code runs on all i686 family chips. When used with `-mtune`, it has the same meaning as generic.

`-pg` Generate extra code to write profile information suitable for the analysis program `gprof`. You must use this option when compiling the

source files you want data about, and you must also use it when linking.

`-o file`

Place output in file `file`.

`-S` Stop after the stage of compilation proper; do not assemble. The output is in the form of an assembler code file for each non-assembler input file specified.

By default, the assembler file name for a source file is made by replacing the suffix `.c`, `.i`, etc., with `.s`.

Input files that don't require compilation are ignored.

`-D name`

Predefine `name` as a macro, with definition 1.

`-D name=definition`

The contents of definition are tokenized and processed as if they appeared during translation phase three in a `#define` directive. In

particular, the definition will be truncated by embedded newline characters.

`-E` Stop after the preprocessing stage; do not run the compiler proper. The output is in the form of preprocessed source code, which is sent to the standard output.

Input files that don't require preprocessing are ignored.

`-f` Many options have long names starting with `-f` or with `-W`

`-I dir`

Add the directory `dir` to the list of directories to be searched for header files.

`-Ldir`

Add directory `dir` to the list of directories to be searched for `-l`.

`-m machine options`

`-g` Produce debugging information in the operating system's native format (`stabs`, `COFF`, `XCOFF`, or `DWARF 2`). GDB can work with this debugging information.

`-W warning`