

Wildcard → pattern matching

*	0 ~ ... characters	*.txt	all files
?	1 characters	? .txt a? .txt a? :	all files start with a ↑ start with a $a^*.txt = a^*t$
[]	[] ← any between ← exactly one	[aeiou] ca[nt]j*	car cat candy catch
[!]	Not included between - 1 character	[!aeiou]* baseball! cricket!	

[a-g]* hyphen
create a range , one character between start with

[1-8]*

[L = alpha :]] characters upper and lower cases

[L = alnum :]] characters + digits

[L = digit :]] 0-9

[L = lower :]]

[L = space :]]

[L = upper :]]

\

escape character → match on wildcard character

* \ ?

file ends with ? mark.

Example: ls ?? ab, ab-

ls ?: - one character in length
- a, b, c,

a?.txt - start with a
- followed by one character

a*.txt

ab.txt

ab.txt
ac.txt

|s c [aeiou]t
— one character

|s [a-d]*
— range, but still one

|s (*)[c=digit=d]* → jazz.mp3

Input Output and Redirection

Stdin 0 }
Stdout 1 } File descriptor
Stderr 2 }

- > - redirects standard output to a file
 - overwrite (truncating) existing contents
- >> - red Stdout
 - append
- < - red stdIn

& Used with redirection to signal that a file descriptor
is being used .

2 > & 1

2 > file

> /dev/null redirect output to nowhere

ls -l > file.txt 把所有打印出来的文件信息都放进
ls >> file.txt 保存在file.txt里面

sort < file.txt 把 file.txt 通过 sort 命令中 .

ls -l 1> files.txt 2 same
ls -l > files.txt

sort < files.txt > sorted_files.txt

ls files.txt not-here 1> out 2> out_err

out = files.txt

out_err = ~~out~~

ls files.txt not-here > out_both 2> & 1
out put to out_both

error to output

single using
redirection.

Comparing files (content)

diff f1 f2

sdiff f1 f2 side-by-side comparison

vimdiff f1 f2 highlight

(3) diff f1 f2 → 3C3
line Nr f2

line Nr f1 action = a Add
c Change
d Delete

(4) sdiff f1 f2

| differing lines
< line from file 1
> line from file 2

vim diff → 分屏显示 2 个不同的窗口上。

Ctrl - w w go to next window
= q quite current
: qa quite all
: qa! force quit all (no changing)

cat (-n) secret
逐行显示

echo message >> secret

把 secret 文件的最后 - 1 行添加
到 content = message

Searching in Files and using Pipes

grep

lines matching a pattern

grep pattern line

-i search ignoring case

grep user secret

-c count (个数)

file secret 文件中找 user

-v invert match, get 不匹配的行

grep o secret
↓只找 O 字母

-n Precede output with line number

grep -v o secret
—— 找不含 O 的行

grep -i User secret

↓对大小写不敏感

grep -ci User secret

↓ 数量

strings

|
pipe

file Sales.data
用它来寻词 Sales.data 文件的内容。

Searching for Text in Binary Files strings

Strings Display printable content of a file

| : Pipe symbol

command - output | command - input

cat file | grep pattern

cat 打印 file 里面的内容，传入给 grep， grep 寻找指定内容。

Cut



Cut out selected portions of files,
if file is omitted → use standard input

Cut option

-d Use delimiter as the field separator

-f N Display Nth field

strings a.mps | grep -i hello.

↑
从文本文件 a.mps 读取，作为输入给 grep

以 human readable

grep 寻找里面的 hello，
不区分大小写。

hello world

Strings | a.mp3 | grep -i hello | head -1 | cut -d' ' -f2

Word

字符串分析

进行排序

通过空格

grep bob /a/b | cut -d: -f1 | sort | tr ":" "\n" | column -t

输出文本

transform 把=替换

column -t
转成表格

more

cat a.txt | less

less

Transferring and Copying Files over the Network

SCP

secure copy

SFTP

SSH file transfer protocol

SCP clients

SCP
SFTP

FTP

lpwd

连接到 server

SFTPS

put a.txt

SCP source destination

SFTP host Start a secure file transfer session

SFEP @ host with host

SFTP Linuxserve 通过 SSH 建立连接到 Linuxserve 上

SFTP admin@linuxserve

不同端口

! 同时建立本地连接。

把文件放到目标目录上。

Customizing the shell prompt 修改 prompt 的显示方式

\$PS1 Bash, ksh, sh

\$prompt Csh tcsh zsh

\d data

\h host name to the first period

\H Hostname

\n newline

(时间) \t 12小时 @ + am(pm)

\T 24小时 . \A + am/pm

Bash shell page

\u	Username
\w	Current working directory
\W	Base name of the current directory
\\$	User

如何将你空闲时
在 Linux 系统里的
Prompt - 定义为
/.bashrc profile 文件里
通过变量
然后 export。

echo 'export PS1="[\u@\h \w]\\$ "' > ~ / .bashrc

删除 PS1 去空格。

PS1 = " [\u@\h \\$ "

固定模式，
→ Jason@linux\$
如果修改了，
就重新修改。

PS1 = "<t \u@\h \w>\\$ "

→ 09:00:00 Jason@linux\$ ~

可以在终端里
.bashrc profile 定义
参数。

export PS1="[\u@\h \w \\$]"

且在 home directory

Aliases

shortcuts

use for long commands

alias [name [=value]] 格式

↓
可以用的地方

① fix typo error.

alias grep='grep'

② Make linux behave
like another OS

alias cls='clear'

创建 .alias
修改 alias

unalias name
unalias -a
删除所有 -

login/logout alias会消失。
可以→修改 .bash_profile
里面。

Environment variables

打印
printenv

printenv Home

var 的
值是

echo

echo \$_↓ HOME
要加美元符
大小写要注意

打印所有 var
name = "name"
HOSTNAME = web0.1

PATH = ... / ..

-般都是大寫

创造

export

Syntax

export VAR = "Noe"

↓ 不再有空格。

export TZ = "US|Pacific"
(time zone)

移除

unset

unset VAR
unset TZ

Top 40

EVR Text
Supplement

Processes and Job Control

PS	Display process status
PS -e	everything, All process
-f	full format listing
-u username	指定的某个人的 PS -uf user
-P pid	指定 id 下信息 PS -P 12345

ps -e

ps -ef | less

ps -ef

ps -eH

ps -e --forest } display process tree

ps -u username

pstree in a tree format

top — interactive process viewer

htop > Q 取消, F1-F6 是 CPU/TB/S/MEM

相同,
出来的结
构不同

Background and foreground Process

d

>+ctrl + c

kill ctrl + z

bg [%num]

fg [%num]

kill + ID

jobs [%num]

↳ jobs %% 当前所有的 job

jobs %+ % -

start command in background

kill the foreground process

suspend the foreground process
停止后台运行

Background a suspended process

Foreground a background p ↗ 前 ↙ F2

kill + pid

↓ fg % 2 把2调入前面

(is + jobs

当前所有的 job

杀掉 2 kill

kill [-sig] id

kill -l

→ . / prg &

= [1] 123 ← Pid of the prg
job number Pid

~ PS - P

= 1234

kill 123
kill -15 123

Send a signal to a process

kill -g 123 S 是 kill signal

display a list of signal → 会列出很多

Signal 的表
41 SIGTERM
42)

fg 反

反

ctrl + c 停止

jobs 用这个可以列出很多的任务

⋮

[-] 先前工作

[+] 一当前工作

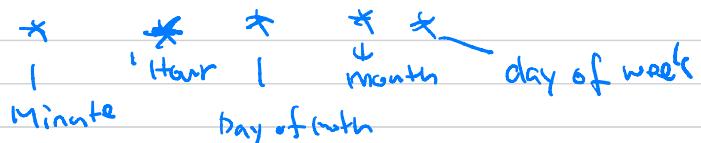
Scheduling Repeated job with cron

cron

a time based job scheduling ← automate tasks
schedule tasks

crontab

to create, read, update, delete job schedule



0 15 30 45 * * * /opt/bin 原理
每15分钟
*/15 * * * *

0-4 * * * * 每5分钟每分钟

- | | | |
|------------|-----------|-------|
| ④ yearly | 0 0 1 1 * | |
| ⑤ annually | 0 0 1 1 * | |
| ⑥ monthly | 0 0 1 * * | |
| ⑦ weekly | 0 0 * * 0 | · · · |

crontab file

install a new crontab from file

crontab my-cron

安装 my-cron

-l

list cron jobs

-e

edit

-r

remove all of your

! echo \$EDITOR 看常用的文本编辑器.

→ EDITOR=vi

→ crontab -e 现在用 vi 编辑 crontab -

Switching users and Running commands as others

SU [username]

change user ID

→ Superuser

→ root account

SU = su /

- 判断你有没有登陆，提供一个账号，User 是 logged, 然后切换到这个
也一样是通过

-c command

specify a command to be executed

-c "xx xx"
user 相关

Whoami

Whoami

→ Jade switch user

→ su Tom → Tom

创造 ENV 在 Jack下

su Tom ↴

↓

su - Tom

env 显示

env 不显示

当处在 Tom 用户下

{ su -c 'echo \$HOME' Jack → 不显示
| - - < Jack → 显示 Jack 的 Home 路径

Sudo

→ Superuser do

Sudo -l

显示所有的命令

sudo common

做为 root 去执行 command

sudo -u root command

① 相等

Sudo -u user command

Sudo su

→ to superuser

sudo su -

→ to superuser + /root

sudo su - username

→ to username

sudo -S

Start shell as root

S for Shell

sudo -u root -S

sudo -u user -S

Change sudo configuration

visudo

Edit the /etc/sudoers file 需先进入 su

admhunger ALL=(ALL) NOPASSWD:ALL

jason lnxsvr = (root) /etc/init.d/oracle

su 用户权限不同即可

Shell history and Tab completion

shell history

Store $\sim/.bash_history.$ $\sim/.history$ $\sim/.histfile$

history/

保存 250 命令 shell

HISTSIZE

命令 shell

export HISTSIZE=100

设置历史

! N

repeat by number

!!

previous

! String

repeat most recent command starting with "String"

! : N

<Event> <Separator> <Word>

: N or Word on the command line

: 0 → command

: 1 → first argument -

!! 显示所有

vi !:2 显示上面四面的内容

!^

!^ = !: /

! \$

命令回溯的最后一个 argument.

ctrl - r

reverse shell history search

ctrl - g

取消 command

!!

list commands

Installing Software on Ubuntu

APT	Advanced Packaging Tool
apt+	apt
apt-cache	search a package apt - cache search String
apt - get	Install apt - get install -y packages = to all automated
	remove apt - get remove packages configuration 不删除
	apt - get purge package configuration 全删除

	Info of Package	apt - cache	Show packages
	update the local list	apt - get update -y	
	of remote packages		
apt代替		apt search install purge update upgrade	
apt - cache			
apt - get			
dpkg	To dependency version list		
	dpkg - l	List all installed package	
	dpkg - S /path/to/file	List all package file	

dependencies

dpkg -L package list all file belongs to package

dpkg -i package.deb

Install

dpkg -r package

remove 这方法，configuration intact.

apt autoremove

uname -m

显示你的系统使用了什么设备

curl

下载网络文件

curl -O http://...
下载到本地。

RPM

dnf / yum

rpm

DEB

apt

dpkg