

# LINUX假期整理

---

## linux基础

---

### 1. 软件安装

---

源

### 2.相应操作

---

#### 软件更新

`sudo apt-get update`

#### 软件升级

`sudo apt-get upgrade`

#### 管理员权限

`apt-get update`

无管理员权限 + `sudo`!! 临时赋予管理员权限

#### 软件安装

`sudo apt-get install`

`apt-get install` ×

#### 查找可用软件包

`apt-cache search`

### 3.闭源系统 Unix MAC APPStore

---

解决方式 1.Linux 2.VMware ubuntu 3.aliyun Tencent 4.Pi (root)

### 4.命令

---

#### ls

当前目录下所有的文件

#### ls -a

查找隐藏文件。隐藏一个文件在该文件名字前加.

## **ls -l**

查找文件类型

## **ls -a /etc**

ls命令 -a选项 /etc执行对象

## **cp**

拷贝 cp a b c d e test

在cp命令最后要加一个目录

cp a b

将a 拷贝到b中

## **mkdir 184**

创建目录 184

## **touch a b c d e**

如果文件不存在 创建空白文件

## **echo a**

输出a

## **echo a > a**

将a读入到a中

## **echo \$HOME**

打印家目录的路径

## **cat**

强制读出

## **cd**

打开目录

## **cd -**

返回上次工作的目录

## **cd ..**

返回上一个目录

## pwd

打印工作目录

输出路径

## 返回家目录 (\$HOME)

cd /etc 或 cd ~

## rm

移除

## rm -f -r

## mv

将a移动到b

mv a b c d e test

将a b c d e 移动到test

## HOME=/

变量赋值必须连着写

## 5.LINUX 手册

---

### man 手册

---

man + 命令

查找 scp

### tldr + 命令

基于网络的手册

远程用户名缺省时远程用户名和本机用户名相同

**sudo apt-cache search pip | grep "python3" | grep "pip"**

| 管道

grep限制条件

**scp id\_rsa.pub g@192.168.2.194:./ssh/authorized\_keys**

将派中的authorized\_keys 拷贝到本机上

**ssh-copy-id -i /home/G/.ssh/id\_rsa.pub G@ 192.168.2.42**

将本机中的authorized\_keys 拷贝到派上

## ifconfig

查询自己的地址

## ssh-keygen

## touch

创建文件

## git add + 文件名

创建文件

## git rm + 文件名

删除文件

## git status

查找是否有未提交的文件

## git commit -m "作用"

提交

## git pull

更新最新储存库

## git log

查找回溯码

## git reset

进行回溯

## git fetch origin

查看仓库改动

## git config --global user.name "name"

## git config --global user.email "email"

# SHELL

---

## 1.变量定义

---

a=12

a=helloworld

a= `pwd`

a=\$a:a

局部变量: local a=12

## 2.位置变量

---

\$0:获取当前执行shell脚本的文件名,包括路径。

\$n:获取当前脚本的第n个参数, n=1...9, 如果n大于9, 则需要将n使用大括号括起来。

\* : 获取当前*shell*的所有参数, 将所有命令行参数视为单个字符串, 相当于“**1,2,3**”

#:得到执行当前脚本的参数个数

@: 获取这个程序所有参数, 并保留参数之间的任何空白, 相当于“1"2 "3”

## 3.特殊变量

---

?:判断上一指令是否成功执行

\$\$:取当前进程的

!:上一个指令的

## 4.输入

---

### read

-a array 把输入赋值到数组array中, 从索引号零开始

-d delimiter 用字符串delimiter中的第一个字符指示输入结束, 而不是一个换行符

-e 使用Readline来处理输入。这使得与命令行相同的方式编辑输入

-n num 读取num个输入字符而不是整行

-p prompt 未输入显示提示信息, 使用字符串prompt

-r Raw mode 不把反斜杠字符解释为转义字符

-s Silent mode

-t seconds 超时

-u fd 使用文件描述符fd中的输入, 而不是标准输入

## 5.输出

---

## echo

echo -e "Hello\n" 开启转义

echo "hellow"

## 6.函数

---

```
function _printf_ {  
    echo $1  
    return  
}  
  
_printf_() {  
    echo $1  
    return  
}  
function _printf_() {  
    echo $1  
    return  
}
```

## 7.流程控制

---

### -if

```
if [[ condition ]]; then  
    #statements  
fi  
  
if [[ condition ]]; then  
    #statements  
else  
    #statements  
fi  
  
if [[ condition ]]; then  
    #statements  
elif [[ condition ]]; then  
    #statements  
fi
```

### -while

```
while [[ condition ]]; do  
    #statements  
done
```

## -until 用法与while相同

## -case

```
case word in
    pattern )
        ;;
esac
```

## 8.数组

declare -a a

name[subscript]=value

name=(value1 value2..)

### 数组操作.

输入数组内容

`${array[*]}`

`${array[@]}`

确定数组元素个数

`${#array[@]}`

找到数组的下标

`${!array[@]}`

数组追加

`array+= {a b c}`

数组排序

sort

删除数组与元素

unset

## SHELL元字符

`=`: 变量名=值, 为变量赋值。注意左右变量值与`=`之间不要有空格

`` ``: 取命令的执行结果, 与`$`有相似之处

`$`: 变量值的替换, `$`变量名替换为shell变量的值; 为避免在文本连接时混淆, 请使用`${变量名}`; `$0...$9`代表shell文件的参数。`$( )`同`` ``; `${}`限定变量名的范围; `$[ ]`整数计算

>: 将标准输出重定向到文件

>>: 将标准输出追加到文件

<: 从文件中获取标准输入

<<

|: 管道命令例: p1 | p2 将p1的标准输出作为p2的标准输入

&: 后台运行命令, 最大好处是无需等待命令执行结束, 可以在同一命令行下继续输入命令

(): 在子shell中执行命令, 在子进程中执行

{ }: 在当前shell中执行命令, 或用在变量替换的界定范围

[]: 字符通配, 匹配括号内之一

;; 命令结束符。例如p1;p2表示先执行p1, 再执行p2

&&: 前一个命令执行成功后, 才继续执行下一个命令

||: 前一个命令执行失败后, 才继续执行下一个命令

!: 执行历史记录中的命令!731; 匹配最近的一次命令!echo; 取非ls/dev/sda[!1]; 结果取反!echo ok; echo \$?

%: \$1相当于fg 1

^: 取非, 和!雷同^string1^string2^将上一命令的string1替换为string2

~: home目录

#: 注释

\*: 通配符, 任意字符

?: 通配符, 任一字符

.: 当前目录: source

-: 减号; 上次工作目录

:: 真值; 空命令

\\: 转义

/: 目录分隔符

## Linux系统信息

---



## uptime

---

打印系统运行时长和平均负荷

语法：

uptime[phsV]

## who

---

显示当前登录系统的用户信息

语法：

who [option]

选项：

-H:显示各栏的标题信息列

-u:显示空置时间

-q:只显示登入系统的用户名称和总人数

## last

---

显示用户最近登录信息

语法：

last [fnx] <username...> <tty...>

选项：

-f:"file":制定记录文件

-n:"num":制定输出记录数

-x:显示系统关机，重新关机等信息

## date

---

显示或设置系统时间与日期

语法：

date [dsu] <参数>

选项：

-d:"string":显示字符串所指的时期

-s:"string":设置时间

-u:显示GMT（如果有CST的情况下）

参数：

<+日期格式>:显示使用的日期格式

## w

---

当前登录用户列表及正在执行的任务

语法:

w [husfV] <...>

options:

h:不打印头文件

s:使用短输出格式

f:切换显示FROM项，默认显示

wall "string"

广播给所有人string

## whoami

---

打印当前有效的用户名称

语法:

whoami有效用户

who am i 真实用户

## uname

---

打印当前系统信息

语法:

uname [options]

参数:

-a:全部信息

-m:电脑类型

-n:网络上主机的名称

-v:os版本

-p:处理器类型

-i:硬件平台

-o:操作系统名称

## cal

---

显示日历

cal [选项] <参数>

选项

-1, -3: 显示最近一个月和三个月的日历

-j:显示一年的第几天

-y显示当前年

参数

月：指定月份

年：指定年份

## 文件与目录

---

### 目录：

---

#### cd

切换当前工作目录

cd /etc :直接切换到/etc目录

cd .. :切换到上层目录

cd . :切换到当前目录

cd :回到自己的家目录

cd ~ :回到自己的家目录

cd - :回到上次工作目录

#### mkdir

创建目录

mkdir [pm]

-p:自动创建父目录

-m:设置权限

#### pwd

打印当前工作目录

L:显示逻辑工作目录

P:显示物理工作目录

#### rmdir

删除目录

`rmdir [p]`

p:删除祖先

## 文件目录管理：

---

### ls

显示文件及目录信息

### rm

删除

`rm [irf]`

-i:互动模式

-r:递归删除

-f:force

### basename

取文件名

### cp

拷贝

`cp []`

-i:若文件存在，询问用户

-r:递归复制

-a:pdr的集合

-p:连同文件属性一起拷贝

-d:若源文件为链接文件的属性，则复制链接文件的属性

-s:拷贝为软链接

-l:拷贝为硬链接

-u:源文件比目的文件新才会拷贝

### mv

移动

`mv []`

-i:互动模式

-f:force

-u:源文件更新才会移动

## dirname

取目录名

## 文件内容查阅：

---

### cat:

正向连续读

-A:相当于-vET

-v:列出看不出的字符

-E:显示断行符为\$

-T:显示TAB为^I

-b:列出行号

-n:列出行号，连同空行也编号

### nl

-b:行号指定的方式

-b a:相当于cat -a

-b t:相当于cat -b

-n:列出行号的表示方法

-n ln:行号在屏幕最左边显示

-n rn:行号在自己字段的最右边显示

-n rz:行号在自己字段的最右边显示，前面自动补全0

-w : 行号所占位数

### less

less file

/string 向下查找 n:继续向下查找

/?string反向查找 N:继续反向查询

### tail

tail [-n num]

-n num:显示文件后num行

-n +num:除了前num行，其他的都显示

## **tac**

与cat相反，从最后一行开始打印

## **more**

more file

/string:向下查找string关键字

:f 显示文件名称和当前显示的行数

q 离开

## **head**

head [-n num]

-n num:显示前num行

-n -num:除了后num行外，其他的都显示

man ls | nl -b a -n rz -w 8 | head -n 120 | tail -n 200

输出120到200

## **od**

-t:

a:默认字符输出

c:ASCII字符输出

d[size]:十进制输出

f[size]:浮点数输出

o[size]:八进制输出

x[size]:十六进制输出

## **修改文件时间**

---

**mtime:**内容数据改动时才更新这个时间

**ctime:**文件的权限，属性改动时更新这个时间

**atime:**文件的内容被取用access时，更新这个时间

**ls -l:**

权限 内存 时间 文件名

## touch []

-a:仅修改访问时间

-c:仅修改文件的时间，若文件不纯在，不新建

-d:修改文件日期

-m:仅修改mtime

-t:修改文件时间[yyymmddhhmm]

## 文件隐藏属性

---

### chattr:

chattr [+=[权限]

A:不修改atime

S:同步写入

a:只能增加数据

c:自动压缩，解压

d:不会被dump程序备份

i:不能删除，修改，建立连接

s:文件删除时，直接从磁盘删除

u:文件删除时，数据内容存在磁盘中

### lsattr:

lsattr []

-a:打印隐藏文件的隐藏属性

-d:如果是目录，仅打印目录的信息

-R:递归

## 文件特殊权限

---

set\_uid: s 二进制程序文件，非脚本 用户在执行改程序时获取程序所有者权限

set\_gid: s 目录和二进制程序文件 用户在该目录里，有效组变为目录所属组

sticky bit: t 目录 在该目录下，用户只能删除自己创建的内容

## linux文件属性

---

### ls -l

---

在你以root的身份登录linux之后执行ls -l

```
-rwxrwxr-x  1    root    root    8600    7月 14 14:45    .
```

权限	连接	所有者	用户组	文件容量	修改时间	文件名
----	----	-----	-----	------	------	-----

第一个字符代表这个文件是目录，文件或链接文件

①[d]目录

②[-]文件

③[l]连接文件

④[b]表示设备文件里面的可供储存的接口设备

⑤[c]表示设备文件里面的串行端口设备

```
-rwx rwx ---
```

文件所有者的权限    文件所属用户组的权限    其他人对此文件的权限

r可读w可写x可执行    ---无权限

第一组:文件所有者的权限

第二组:文件所属用户组的权限

第三组:其他人对此文件的权限

第二列表示有多少文件名连接到此节点

第五列为这个文件的容量大小,默认单位为B

第六列为这个文件的创建文件日期或者是最近的修改日期

## 改变文件属性与权限

---

chgrp:改变文件所属用户组

```
chgrp [-R] dirname/filename...
```

chown:改变文件所有者

```
chown [-R] 账号名称    文件或目录
```

```
chown [-R] 账号名称:组名    文件或目录
```

chmod:改变文件的权限

```
r:4 w:2 x:1
```

```
chmod [-R] xyz 文件或目录
```

## 进程管理

---

### free

---

打印系统情况和内存情况



- b|k|m|g:以字节,KB,M,G显示
- o:忽略缓冲区调节列
- t seconds:每隔seconds执行一次
- h:以可读形式显示

## dfstat

---

实现监控磁盘，C P U，网络等

## ps tree

---

以树状显示进程派生关系

- a:显示每个程序的完整指令
- n:使用 P I D 排序
- p:显示 P I D
- u:显示用户名
- l:使用长列格式显示树状

## kill

---

删除执行中的程序和工作

- a:处理当前进程时，不限制命令名和进程号的对应关系
- l 信号ID:不加信号 I D，则列出全部信号
- p pid:给 p i d 的进程只打印相关进程号，而不发送任何信号
- s 信号ID|信号 name: 指定要发出的信号
- u: 指定用户

## top

---

显示当前系统进程情况，内存，C P U等信息

- b:以批处理模式操作
- c:显示完整的命令
- d seconds: 屏幕刷新闻隔时间
- s:以安全模式运行
- S:累计模式
- u uname:指定username
- p pid:指定PID

-n nums:循环显示次数

-q:root时，给尽可能高的优先级

## ps

---

报告当前进程状态

ps -aux

ps -ef

## pgrep

---

查找进程 I D

-o:起始进程号

-n:结束进程号

-l:显示进程名称

-p pid:指定父进程

-g gid:指定进程组

-t tty:指定开启的进程终端

-u uid:指定uid

## pkill

---

批量按照进程名杀死进程

-o:起始pid

-n:结束pid

-p pid:指定父进程发送信号

-g:指定进程组

-t tty:指定终端

# 用户管理

---

## /etc/passwd

---

用户名 密码位 用户编号 归属组编号 姓名 \$HOME Shell

## /etc/shadow

---

用户名 以加密密码 密码改动信息 密码策略

## /etc/group

---

群组名 密码位 群组编号 组内用户

## /etc/gshadow

---

群组密码相关文件

## /etc/sudoers

---

用户名 权限定义 权限

## 用户管理相关命令

---

### su

切换用户

-|-l:重新登录

-m|-p:不更改环境变量

-c comand:切换后执行命令，并退出

### passwd

设定用户密码

-d:清除密码

-l:锁定账户

-e:使密码过期

:-S显示密码认证信息

-x days:密码过期后最大使用天数

-n days:密码冻结后最小使用时间

-s:更改登录shell

-f:更改用户信息

### chsh

更改用户 s h e l l

chsh -s Shell

### useradd

新建用户

-d:指定\$HOME

-m:自动建立\$ HOME  
-M:不自动建立\$HOME  
-s shell:设置用户登录shell  
-u uid:设置用户编号  
-g groupname:设定用户归属群组  
-G groupname:设置用户归属附加群组  
-n:不建立以用户名称为群组名称的群组  
-e days:缓冲时间，days天后关闭账号  
-c string:设置用户备注  
-D[表达式]:更改预设值

## id

显示用户信息

-g:下属所属组ID  
-G:显示附加组ID  
-n:显示用户，所属组或附加群组的名称  
-u:显示用户ID  
-r:显示实际ID`

## sudo

临时切换为 r o o t 用户

-s:切换为root shell  
-i:切换到root shell,并初始化  
-u username|uid:执行命令的身份  
-l:显示自己的权限

## gpasswd

设定群组密码

-a username:将用户加入数组  
-d username:将用户从群组中删除  
-r:删除密码  
-A username: 将用户设置为群组管理员  
-M username1,username2....:设置群组成员

## usermod

修改用户账号

- c string:修改备注信息
- d dir:修改\$HOME
- e days:密码期限
- f days:密码过期后宽限的日期
- g groupname:修改用户所属群组
- G groupname:修改用户所属附加群组
- l username:修改用户账号名称
- L:锁定用户密码，使密码无效
- s shell:修改用户登录后所使用的shell
- u uid:修改用户ID
- U:接触密码锁定

## userdel

删除用户

- r:删除用户相关文件和目录

# 数据提取

---

## cut

---

切分

- d c:以c字符分割
- f num:显示num字段的内容
- b num:字节
- c num:字符

## sort

---

排序

- f:忽略大小写
- b:忽略最前面的空格符
- M:以月份名称排序
- n:以纯数字方式排序

-r:反向排序

-u:uniq

-t:分隔符，默认【T A B】

-k:以那个分区排序

## uniq

---

去重

-i:忽略大小写字符的不同

-c:进行计数

## split

---

文件切分

-b SIZE:切分为SIZE大小的文件

-l num:以num行为大小切分

## tr

---

替换，压缩和删除

c:取代所有不属于第一字符集的字符

d:删除所有属于第一字符集的字符

s:将连续重复的字符以单独一个字符表示

t:先删除第一字符集较第二字符集多出的字符

## grep

---

检索

-a:将二进制文件以普通文件的形式搜寻数据

-c:统计搜寻到的次数

-i:忽略大小写

-n:顺序输出行号

-v:反向输出（输出没找到的）

## WC

---

统计字符，数字，行数

-l:仅列出行号

-w:仅列出多少字

-m:仅列出多少字符

## tee

---

双向重导项

## xargs

---

参数代换

-0:将特殊字符还原为普通字符、

-eEOF:当xargs读到EOF时停止

-p:执行指令前询问

-n num:每次执行command时需要的参数个数