|  |  |
| --- | --- |
| **命令及描述** | **例子** |
| **locale**  查看本地语言配置 |  |
| **ls**  显示文件 | ls –l 以详细信息的形式显示文件  ls –a 显示所有文件包括隐藏 |
| **cal**  显示日历 | cal 15 10 2018 显示2018年10月15日的日历 |
| **date**  **显示日期** |  |
| **[command] –help** | 显示命令帮助 |
| **man [command]** | 显示命令手册 |
| **who**  **查看谁在线** |  |
| **exit**  **退出** | 快捷键ctrl+d |
| **sync**  **将内存的数据同步到硬盘** | 一般在关机前执行 |
| **shutdown**  **关机** | shutdown –r 服务停掉后重启  shutdown –h 服务停掉后一分钟关机  shutdown –h now 服务停掉后马上关机  shutdown –h +10 10分钟后关机 |
| **halt**  **关机** |  |
| **poweroff**  **关机** |  |
| **reboot**  **重启** |  |
| **su**  **切换到root身份** |  |
| **nano**  **文本编辑器** |  |
| **权限说明** | 第1位  d：目录；-：文件；l：快捷方式；b：可储存设备；c：只读设备 |
| **chgrp**  **修改文件所属群组** | chgrp [groupName] [fileName]  -R 目录之下所有的目录跟文件的权限都会修改 |
| **chown**  **修改文件拥有者** | chown [username] [fileName] 修改文件拥有者  chown [username]:[groupname] [fileName] 同时修改文件的拥有者及群组  -R 目录之下所有的目录跟文件的权限都会修改 |
| **chmod**  **修改文件权限** | **数值修改法**  权限对应数值r:4 w:2 x:1  chmod 764 fileName => -rwxrw-r--  **字符修改法**  a=all u=user g=groups o=others  chmod u=rwx,go=rx fileName=> -rwxr-xr-x  chmod u+r fileName 在不修改文件其它权限下增加个拥有者读取权限 |
| **权限解释** | **文件**  r：读取文件的内容  w：修改文件的内容，不包括删除文件  x：执行文件  **目录**  r：读取目录下的文件名，仅看得到文件名  w：修改或删除目录及目录下文件的名称，移动目录下文件  x：进入该目录的权限或用命令执行处理该目录以下文件 |
| **目录解释** | **/usr 软件位置，相当于window的program files**  **/var 系统运行后占用的目录，经常变动的的**  目錄樹架構示意圖 |
| **目录路径** | **. 当前目录**  **.. 上一层目录**   * **回到前一层目录**   **~ 当前账号 家目录**  **~[user] 代表user这个账号的家目录** |
| **cd**  **切换目录** |  |
| **pwd**  **显示当前目录路径** | **pwd –P 显示快捷方式连接的路径** |
| **mkdir**  **创建一个新目录** | **mkdir –p test1/test2 会同时把上层目录test1也创建**  **mkdir –m 777 test1 创建目录的同时把权限也加上** |
| **rmdir**  **删除一个空目录** | **rmdir –p test1/test2 会连同上层的空目录也删了** |
| **cp**  **复制文件或目录** | **-a 相当于 –dr –preserve=all**  **-d 如果文件是快捷方式，那复制结果也是快捷方式，默认是复制真实文件；**  **-p 文件的权限、用户、时间都复制**  **-f 强制复制**  **-i 覆盖文件时进行提醒**  **-r 复制目录，递归复制**  **-l 進行硬式連結(hard link)的連結檔建立，而非複製檔案本身；**  **-s 复制成快捷方式**  **-u 有更新才复制** |
| **rm**  **删除文件或目录** | **-f 强制，忽略不存在的文件，不会出现警告**  **-i 删除前会进行提示**  **-r 递归删除，用在目录删除上，很危险的操作！** |
| **mv**  **移动文件或目录** | **-f 强制，如果目标存在，直接覆盖**  **-I 目录存在时，进行提示**  **-u 若目标文件已存在，且复制源比较新，才进行覆盖**  **mv file1 file2 修改文件或目录名称** |
| **basename**  **获取文件名** |  |
| **dirname**  **获取目录路径** |  |
| **cat**  **查看文件内容** | **-n 列出行号 连空白也有行号**  **-b 列出行号，空白没有**  **-T 将tab以^I显示出来**  **-v 显示一些看不出来的特殊字符**  **-E 将行结尾的$显示出来**  **-A 相当于 –vET** |
| **tac**  **与cat显示相反** |  |
| **nl**  **以行号查看与cat –b一样** |  |
| **more**  **查看内容，内容多用翻页** | **空格键：下一页**  **Enter：下一行**  **/字符串：查找某个字符串**  **b：往回翻**  **q：离开**  **:f 显示当前文件名及当前行号** |
| **less**  **查看文件内容** | **空格：向下翻页**  **pagedown 下翻页**  **pageup 上翻页**  **/字符串：查找 n：查找下一个 N：查找上一个**  **q：离开** |
| **head**  **查看文件前几行内容** | **head –n 20 fileName 查看文件前20行内容，默认是10行** |
| **tail**  **查看的最后几行内容** | **tail –n 20 fileName 查看文件最后20行内容，默认是10行**  **tail –n +100 fileName 文件100行以后的内容都显示出来** |
| **od**  **查看二进制文件** | **echo password | od –t** |
| **touch**  **修改文件相关时间**  **创建一个空文件** | **-a 仅修改文件的access time**  **-c 仅修改文件时间**  **-m 仅修改mtime**  **-t 修改时间但不用当前时间，格式为YYYYMMDDhhmm**  **-d 修改时间但不用当前时间** |
| **umask**  **权限预置** | **创建文件时最大权限值为 –rw-rw-rw**  **创建目录时最大权限值为 –rwxrwxrwx**  **umask 查看预置权限分**  **umask –S 以字符查看预置权限**   * **建立檔案時：(-rw-rw-rw-) - (-----w--w-) ==> -rw-r--r--** * **建立目錄時：(drwxrwxrwx) - (d----w--w-) ==> drwxr-xr-x** |
| **chattr**  **修改文件隐藏属性** | **選項與參數：**  **+ ：增加某一個特殊參數，其他原本存在參數則不動。**  **- ：移除某一個特殊參數，其他原本存在參數則不動。**  **= ：設定一定，且僅有後面接的參數**  **A ：當設定了 A 這個屬性時，若你有存取此檔案(或目錄)時，他的存取時間 atime 將不會被修改，可避免 I/O 較慢的機器過度的存取磁碟。(目前建議使用檔案系統掛載參數處理這個項目)**  **S ：一般檔案是非同步寫入磁碟的(原理請參考**[**前一章sync**](http://linux.vbird.org/linux_basic/0160startlinux.php#sync)**的說明)，如果加上 S 這個屬性時， 當你進行任何檔案的修改，該更動會『同步』寫入磁碟中。**  **a ：當設定 a 之後，這個檔案將只能增加資料，而不能刪除也不能修改資料，只有root 才能設定這屬性**  **c ：這個屬性設定之後，將會自動的將此檔案『壓縮』，在讀取的時候將會自動解壓縮， 但是在儲存的時候，將會先進行壓縮後再儲存(看來對於大檔案似乎蠻有用的！)**  **d ：當 dump 程序被執行的時候，設定 d 屬性將可使該檔案(或目錄)不會被 dump 備份**  **i ：這個 i 可就很厲害了！他可以讓一個檔案『不能被刪除、改名、設定連結也無法寫入或新增資料！』對於系統安全性有相當大的助益！只有 root 能設定此屬性**  **s ：當檔案設定了 s 屬性時，如果這個檔案被刪除，他將會被完全的移除出這個硬碟空間，所以如果誤刪了，完全無法救回來了喔！**  **u ：與 s 相反的，當使用 u 來設定檔案時，如果該檔案被刪除了，則資料內容其實還存在磁碟中， 可以使用來救援該檔案喔！**  **注意1：屬性設定常見的是 a 與 i 的設定值，而且很多設定值必須要身為 root 才能設定**  **注意2：xfs 檔案系統僅支援 AadiS 而已** |
| **lsattr**  **把隐藏属性显示出来** | **-d 用于目录**  **-R 连同子目录也显示**  **-a 把隐藏档的属性也秀出来** |
| **SUID**  **特殊权限** | **-rwsrwxrwx 在user上有s的代表SUID权限** |
| **SGID**  **特殊权限** | **-rwxrwsrwx 在group 上有s的代表SGID 权限** |
| **SBIT**  **特殊权限** | **仅对目录有效**  **-rwxrwxrwt**  **目录下创建的文件或目录只有自己或root能删除** |
| **设置特殊权限** | **4：SUID**  **2：SGID**  **1：SBIT**  **chmod 4777 file ; 具有SUID权限**  **chmod 6777 file; 具有SUID和SGID权限**  **chmod 7666 file ； 空的SUID、SGID、SBIT权限**  **chmod u+s file ; chmod g+s file; chmod o+t file;** |
| **file**  **查看文件类型** |  |
| **which**  **查找命令完整名称** |  |
| **whereis** |  |
| **locate**  **查找** |  |
| **updatedb** |  |
| **find**  **查找** | **-mtime n：n为数字，n天之前的24小时内改过的文件**  **-mtime +n：n天之前修改过的文件不包括n天**  **-mtime –n：n天之内修改过的**  **-newer file：file是一个文件，代表比file还要新的文件**  find 相關的時間參數意義  **-uid**  **-gid**  **-user name 查找文件拥有者为name的文件**  **-group name 查找文件群组为name的文件**  **-nouser 查找文件拥有者不为name的文件**  **-nogroup 查找文件群组不为name的文件**  **-name filename 查找名称为filename的文件**  **-size +-n 查找比n大或小的文件**  **-type 查找某种类型的文件（f）文件 （bc）设备 （d）目录 （l）快捷 （s）socket （p）FIFO**  **-perm 777 查找权限等于777的文件**  **-perm -700 查找权限包括700的文件**  **-perm /777 查找权限小于或等于777的文件**  **-exec 将查找到的结果进行其它命令操作** |
| **df**  **显示出文件系统分区的容量** | **-a 显示所有**  **-k 以kb为单位**  **-m 以mb为单位**  **-h 自动显示单位**  **-H 以m=1000k取代 m=1024k**  **-T 显示文件系统类型**  **-i 不显示容易，以inode数量显示** |
| **du**  **显示出目录所占容量** | **-a 默认只显示目录，加上这个参数，连文件也显示**  **-h 自动加上大小单位**  **-s 只显示目录总容量**  **-S 不包括子目录的容量**  **-k 以kb为大小单位**  **-m 以mb为大小单位** |
| **ln**  **创建连接** | **-s 不加这个参数的话，就是hard link，加s就是s link快捷方式**  **-f 目标存在的话，就先移除再创建** |
| **lsblk**  **显示所有硬盘列表** | **-d 只显示硬盘，不显示分区情况**  **-f 同时显示文件格式**  **-p 显示完整路径**  **-i 使用ASCII**  **-m 显示权限**  **-t 硬盘更详细信息** |
| **blkid**  **列出设备的uuid** |  |
| **parted**  **显示硬盘的分区类型及信息** | **parted /dev/sda print** |
| **fdisk (mbr)**  **gdisk (gpt)**  **硬盘分区** |  |
| **partprobe**  **更新分区表信息** | **-s 显示更新信息**  **分完区得重启才会更新，或者用这个命令进行更新** |
| **mkfs.xfs**  **以xfs格式化分区** |  |
| **mkfs.ext4**  **以ext4格式化分区** |  |
| **xfs\_repair**  **修复xfs文件系统** | **-f 修改文件面不是硬盘**  **-n 只是检查**  **-d 针对根目录下进行检查跟修复，很危险的操作** |
| **fsck.ext4**  **修改ext4文件系统** |  |
| **mount**  **将分区挂载到目录** | **mount /dev/sda3 /testmount**  **mount –o loop /tmp/Centos7.iso /data/centos\_dvd 直接挂载iso** |
| **umount**  **卸载** | **-f 强制卸载**  **-l 立刻卸载，比-f 还**  **-n 不更新/etc/mtab情况下卸载**  **umount /home** |
| **mknod**  **不懂有什么用** | **b c p** |
| **xfs\_admin**  **修改xfs分区的标签名及uuid** | **-l 显示分区标签名**  **-u 显示分区uuid**  **-L 设置分区标签名**  **-U 设置分区uuid**  **xfs\_admin –L text /dev/sda2** |
| **tune2fs**  **修改ext4分区的标签名及uuid** |  |
| **mkswap**  **将一个分区格式为swap格式** |  |
| **swapon**  **将一个swap分区启用** | **-s 查看swap启动情况** |
| **swapoff** |  |
| **free**  **查看当前内存使用情况** |  |
| **parted**  **同时支持gpt及mbr分区的命令** | **parted /dev/sda print 显示分区情况**  **parted /dev/sda mklabel gpt 转换成gpt分割表**  **parted /dev/sda mkpart primary fat32 36.0GB 36.5GB** |
| **gzip**  **压缩** | **-1 最快速度； -9速度最慢，压缩最好； 默认-6**  **-v显示出原文件与压缩后文件的对比信息**  **-d 把压缩文件解压出来**  **你进行压缩后，原文件或目录都会不见了** |
| **zgrep**  **从压缩文件里查找关键字** |  |
| **zcat/zmore/zless**  **用来查看压缩文件** |  |
| **bzip2**  **比gzip更好的压缩** |  |
| **bzcat** |  |
| **xz**  **更好的压缩** | **-k 保留原文件**  **-l 查看压缩文件压缩前后对比** |
| **xcat** |  |
| **tar** | **-c 建立打包**  **-x 解压**  **-C 解压在某个目录，搭配x**  **-t 查看压缩包里有哪些文件名**  **-v 查看压缩包里目录及权限**  **-z 用的是gzip解压缩，后缀为\*.tar.gz**  **-j 用的是bzip2解压缩，后缀为\*.tar.bz2**  **-J用的是xz解压缩，后缀为\*.tar.xz**  **-p 保留权限及属性，常用于备份**  **-P 保留绝对路径，这个还不太懂**  **--exclude= 不备份某个目录或文件**  **tar –zcv –f file.tar.gz 压缩**  **tar –ztv –f file.tar.gz 查看压缩里面的文件及目录**  **tar –zx –f file.tar.gz 解压文件**  **tar –zx –f file.tar.gz etc/shadow 解压单一文件** |
| **xfsdump**  **xfs文件系统的备份** | **xfsdump –l 0 –f /tmp/boot.dump /boot**  **-l 默认为0，即完整备份，1为以0了基础的增异备份，2为以0跟1为基础**  **-I 大写的i，查看备份情况** |
| **xfsrestore** | **-I 大写的i，查看备份情况**  **xfsrestore –f /tmp/boot.dump /boot** |
| **mkisofs**  **创建iso文件** | **-o 后面跟要保存的iso文件名**  **mkisofs –o file.iso /home** |
| **wodim**  **刻录** | **wodim -v dev=/dev/sr0 speed=4 -dummy -eject /tmp/system.img** |
| **dd**  **主要用来备份** | **if 就是input file**  **of 就是output file**  **bs 就是block size**  **count 多少个bs**  **dd if=/dev/sr0 of=/tmp/system.iso 把光盘的内容备份下来**  **dd if=/dev/sda1 of=/tmp/sda1.img 把sda1的内容备份下来**  **sda1分区如果有2G，但实际才使用了1G，那备份文件还是2G** |
| **cpio**  **备份还原** | **cpio –ovcB > file 备份**  **cpio –ivcdu < file 还原**  **cpio –ivct < file 查看** |
| **vi编辑器** | **一般指令模式 command mode**  **编辑模式 insert mode**  **指令列命令模式 command-line mode**  vi三種模式的相互關係  **快捷键**  **ctrl+f 向下翻页**  **ctrl+b 向上翻页**  **ctrl+d 向下翻半页**  **ctrl+u 向上翻半页**  **0或home 移动到当前行的最前面**  **&或end 移动到当前行的最后面**  **H 移动到当前屏幕第一行第一个字符**  **M 移动到当前屏幕中间行第一个字符**  **L 移动到当前屏幕最后一行第一个字符**  **G 移动到文件的最后一行**  **nG 移动到文件的第n行**  **gg 移动到文件的第一行**  **n回车 n为数字，向下移动n行**  **搜索**  **/keyword 向下查找关键字**  **?word 向上查找关键字**  **n 查找下一个**  **N 往前查找**  **:n1,n2s/word1/word2/g 从n1行到n2行中查找word1，并替换成word2**  **:1,$s/word1/word2/g 从第1行到最后1行查找word1，并替换成word2**  **:1,$s/word1/word2/gc 从第1行到最后1行查找word1，并替换成word2，并且在替换前进行确认提示**  **指令模式**  **x 相当于del**  **X 相当于backspace**  **dd 删除当前行**  **ndd 删除当前及往下n行**  **d1G 删除当前行到第一行的内容**  **dG 删除当前行到最后一行的内容**  **d$ 删除当前游标所在位置到当前行最后一个字符**  **d0删除当前游标所在位置到当前行第一个字符**  **yy 复制当前行**  **nyy 复制当前行及往下n行内容**  **y1G 复制当前行到第一行的内容**  **yG 复制当前行到最后一行的内容**  **y0复制当前游标所在位置到当前行第一个字符**  **y$ 复制当前游标所在位置到当前行最后一个字符**  **p 粘贴到当前行下边**  **P 粘贴到当前行上边**  **J 将当前行与下一行的内容合并为一行**  **c 向下删除10行，10cj，向下删除10行，10ck**  **u 撤销上一个动作**  **ctrl+r 重做上一个动作**  **. 重复当前动作**  **指令模式切换到编辑模式**  **i 在当前位置**  **I 在当前行第一个字符位置**  **a 在当前位置的下一个字符**  **A 在当前行的最后一个**  **o 在当前行的下一行插入新行**  **O 在当前行的上一行插入新行**  **r 进入取代模式，只取代当前位置的字符**  **R 进入取代模式，一直取代，按esc才退出**  **指令行模式**  **:w 保存**  **:w! 强制保存**  **:q 离开**  **:q! 强制离开**  **:wq 保存并离开**  **ZZ 如果文件没有改动，就不保存离开，如果有改动了，就保存后离开**  **:w filename 另存为**  **:r filename 将filename文件插入到当前内容中**  **:n1,n2 w filename 将n1行到n2行另存为**  **:! command 例如 :! ls-l /home 暂时离开vi指令执行外面的命令**  **:set nu 显示行号**  **:set nonu 隐藏行号**  **:set 查看设置与预设值不一样的设置**  **区域选择**  **v 字符选择**  **V 行选择**  **ctrl+v 区块选择**  **y 复制**  **d 删除**  **p 粘贴**  **多文件编辑**  **vi /file1 /file2**  **:files 查看同时打开哪些文件**  **:n 下一个文件**  **:N 上一个文件**  **:多视图**  **:sp filename 如果是同一个文件，就不用加上filename**  **ctrl+w+方向 切换不同视图**  **:q 离开**  **关键字补充**  **ctrl+x ctrl+n 以当前编辑的这个词作为关键字**  **ctrl+x ctrl+f 以当前目录的文件名做为关键字**  **ctrl+x ctrl+o 以当前文件扩展名做为语法补充** |
| vim 常用指令示意圖 | |
|  | **注：window 与linux 的断字符号不同，所以在window下写的shell脚本，可能无法直接在linux运行，所以需要用dos2unix转换** |
| **iconv**  **编码转换** |  |
| **type**  **查看指令类型** |  |
| **\**  **命令换行符** | **ctrl+u 删除从游标位置到最前面**  **ctrl+k 删除从游标位置到最后面**  **ctrl+a 游标位置移到最前**  **ctrl+e 游标位置移到最后** |
| **变量** | **echo $PATH 输出变量**  **echo ${PATH}**  **myname=caijt 变量赋值**  **echo $myname**  **unset $myname 取消变量**  **version=$(uname -r) version=`uname -r`**  **export myname 将变量变成环境变量，子程序中才能够使用** |
| **env**  **显示环境变量** |  |
| **主要环境变量** | **PS1 那个[root@localhost]#的字符串定义**  **$ shell的进程id**  **? 上个指令的回传值，执行成功的指令会返回0，错误的才会返回数值** |
| **read**  **等待用户输入并创建一个变量** | **-p 提示文字**  **-t 等待时间**  **read –p ‘Please input your name’ –t 30 name** |
| **declare**  **声明变量的类型**  **默认为字符串** | **-a 将变量定义为数组array类型**  **-i 将变量定义为数字int类型**  **-x 将变量变成环境变量**  **+x 将变量定义为私有变量**  **-r 将变量定义为readonly类型，不能修改，也不能unset**  **-p 显示变量类型**  **declare –i sum=1+2+3** |
| **数组** | **var[1]=1**  **var[2]=2**  **var[3]=3**  **echo ${var[1]}** |
| **ulimit**  **限制用户操作的文件大小，同时打开文件数量** |  |
| **变量的内容** | **删除**  **# 从前面往后面删，仅删除最短的**  **## 从前面往后面删，删除最长的**  **% 从后面往前面删，仅删除最短的**  **%% 从后面往前面删，删除最长的**  **echo ${PATH#/\*local/bin:}**  **替换**  **/ 仅替换第一个**  **// 替换全部**  **echo ${PATH/local/hello}** |
| **变量检测赋值** | **username=${user-root}**  **如果user没有赋值的话，就赋值root给username**  **username=${user:-root}**  **如果user没有赋值或为空字符串的话，就赋值root给username**  **username=${user+root}**  **如果user没有赋值的话，username=’’，有赋值的话，username=root**  **username=${user:+root}**  **如果user没有赋值或为空字符串的话，username=’’，有赋值的话，username=root**  **username=${user=root}**  **如果user没有赋值的话，user =root，username=root，有赋值的话，username=$user**  **username=${user:=root}**  **如果user没有赋值或为空字符串的话，user =root，username=root，有赋值的话，username=$user**  **username=${user?’no this name!’}**  **如果user没有赋值的话，就提示no this name，有赋值的话，username=$user**  **username=${user:? ’no this name!’}**  **如果user没有赋值或为空字符串的话，就提示no this name，有赋值的话，username=$user** |
| **alias**  **指令别名** | **alias lm=’ls -al’** |
| **history**  **查看历史命令** | **-c 清除历史记录**  **!! 执行上一个指令**  **!999 执行历史中第999个指令** |
| **读取配置文件加载到当前shell环境** | **source filename**  **. filename** |
| **set**  **终端环境设置** |  |
| **资料重导向** | 1. **等于 > 将指令正确输出的信息重导向**   **ll / > ~/rootfile 将/目录下的列表存到rootfile文件上**  **ll / >> ~/rootfile将/目录下的列表追加到rootfile文件上**   1. **将指令正确输出的信息重导向**   **2> 将错误的信息导向**  **2>> 将错误的信息追加导向**  **2> /dev/null 忽略错误信息**  **> rootfile 2>&1 将正确信息跟错误信息都存到rootfile文件里**  **&> rootfile 同上**  **cat > catfile < ~/.bashrc**  **cat > catfile << ‘eof’** |
| **; && ||**  **指令表达式** | **cmd1;cmd2**  **cmd1执行完后执行cmd2**  **cmd1 && cmd2 执行cmd1成功的话，就执行cmd2，cmd1错误的话，cmd2不执行**  **cmd1 || cmd2 执行cmd1成功的话，cmd2不执行，cmd1错误的话，cmd2执行** |
| **|**  **管道** | **ll a /etc | less** |
| **cut**  **（管道命令）**  **将每行的内容进行剪切** | **-d 分隔字符，将字符串砍成几段 与-f一起用**  **-f 取出第几段**  **-d 只剪切某个位置的字符**  **echo ${PATH} | cut –d ‘:’ –f 5** |
| **grep**  **（查找）**  **查找关键字** | **-v 反向选择 没有关键字的那些行才显示出来**  **-i 忽略大小写**  **-n 显示行号**  **-c 计算匹配的次数**  **-a**  **-A 例如-A3就是把匹配的行，后面3行也显示出来**  **-B 例如-A3就是把匹配的行，前面3行也显示出来**  **grep ‘keyword’ filename** |
| **sort** | **-t 分隔号**  **-k 以分隔后第几个字段来排序**  **-n 以数字类型排序，默认是以字符串**  **-f 忽略大小写**  **-b 忽略前面的空白**  **-M 以月份名字排序 JAN、DEC这种**  **-r 反向排序**  **-u 相同的资料行，仅出现一行** |
| **uniq**  **唯一** | **-i 忽略大小写**  **-c 进行计数** |
| **wc**  **统计** | **-l 多少行**  **-w 多少个字**  **-m 多少个字元** |
| **tee**  **双向重导向** | **-a 累加形式** |
| **tr**  **删除信息中的文字** | **-d 删除关键词**  **-s** |
| **col** | **-x 将tab转换成空白键** |
| **join**  **将多个文件有相同部分的行结合到一起** |  |
| **paste**  **简单地将多个文件结合到一起** |  |
| **expand**  **将tab转换成空白键** |  |
| **split**  **分割文件** | **-b 后面接分割大小**  **-l 以行号来分割**  **split –b 300k /etc/services /tmp/services-** |
| **xargs**  **将参数传给后面的指令** |  |
| **-**  **在管道中的用处**  **可以理解为临时变量** |  |
| **正则表达式** | **^word**  **word$**  **.**  **\**  **\***  **[abc]**  **[1-9]**  **[^abc]**  **\{1,5\}**  **延伸，得用egrep**  **+ 一个或多个**  **? 零个或一个**  **| 或**  **() 群组**  **()+** |
| **sed** | **cat –n /etc/passwd | sed ‘2,5d’ 删除第2到5行**  **cat –n /etc/passwd | sed ‘2a hello world’ 在第2行后面加上hello world**  **cat –n /etc/passwd | sed ‘2i hello world’ 在第2行前面加上hello world**  **cat –n /etc/passwd | sed ‘2,5c hello world’ 把2行到5行替换成hello world**  **sed ‘s/word/newword/g** |
| **printf** | **printf '%10s %5i %5i %5i %8.2f \n' $(cat printf.txt)** |
| **awk** | **last –n 5 | awk ‘{print $1 “\t” $3}’** |
| **diff**  **文件对比** |  |
| **cmp** |  |
| **patch** |  |
| **pr**  **打印** |  |
| **shell script** | **#!/bin/bash 第一行必须这样写** |
| **test**  **测试某个文件或目录存不存在** | **-e 测试文件或目录存不存在**  **-f 测试文件存不存在**  **-d 测试目录存不存在**  **-b 测试一个block device 设备存不存在**  **-c**  **-S**  **-p**  **-L**  **-r 测试是否有可读权限**  **-w测试是否有可修改权限**  **-x测试是否有可执行权限**  **-u测试是否有SUID权限**  **-g测试是否有SGID权限**  **-k测试是否有Sticky bit权限**  **-s 测试是否是空白文件**  **-nt 判断file1是否比file2新 test file1 –nt file2**  **-ot 判断file1是否比file2旧**  **-ef 判断file1与file2是否为同一个文件**  **test n1 –eq n2**  **-eq 两数相等**  **-ne 两数不相等**  **-gt n1大于n2**  **-lt n1小于n2**  **-ge n1大于等于n2**  **-le n1小于等于n2**  **-z 判断一个字符串是否为0**  **-n 判断一个字符串是否不为0**  **str1=str2**  **str1!=str2**  **-a 两情况同时成立**  **-o 两情况其中一个成立**  **! 相反情况** |
| **[]**  **用法跟test一样** | **[ -z “${HOME}” ]** |
| **参数** | **hello.sh opt1 opt2 opt3 opt4**  **$# 4个参数**  **$0 hello.sh**  **$1 opt1**  **$@ opt1 opt2 opt3 opt4**  **$\***  **shift 2 从左边移除2个参数** |
| **if then** | **if [ 条件判断式 ]; then**  **elif [ ]; then**  **else**  **fi** |
| **case** | **case ${1} in**  **“hello”)**  **echo “hello,how are you ?”**  **;;**  **“hello2”)**  **echo “hello2”**  **;;**  **\*)**  **echo “no”**  **;;**  **esac** |
| **function** | **function test1(){**  **echo “Your choice is ${1}”**  **}**  **test1 A;** |
| **while** | **while [ ]**  **do**  **done**  **当条件成立时，进行循环** |
| **until** | **until [ ]**  **do**  **done**  **当条件成立时，跳出循环** |
|  | **for name in name1 name2 name3**  **do**  **done**  **循环** |
|  | **for(( i=1; i<10; i=i+1 ))**  **do**  **done** |
| **sh** | **-n 测试脚本有无问题**  **-x 将脚本执行过程全部列出来** |
| **newgrp users** | **新建一个bash子环境，更改当前用户的有效群组** |
| **/etc/shadow**  **密码文件** | 1. **账号名称** 2. **密码** 3. **最近修改密码日期** 4. **密码不可被修改的天数** 5. **密码需要修改的天数** 6. **密码修改期限前警告的天数** 7. **密码过期后的宽限时间** 8. **账号失效日期** 9. **保留**   **密码过期就是还能用密码登录，但登录后必须修改密码，密码失效就是无法用密码登录了** |
| **useradd**  **新增一个用户** | **-r 创建系统账号 系统账号的uid一般小于1000，且不会创建家目录**  **-u 指定uid**  **-g 指定初始的群组名称**  **-G 指定该用户还具有的其它群组**  **-M 不要创建家目录**  **-m 创建家目录**  **-c 备注 /etc/passwd第5个字段的内容**  **-d 指定哪个目录为家目录，用绝对路径**  **-s 指定一个shell**  **-e 指定账号失效日期**  **-f 指定密码过期后的宽限时间**  **-D 显示用户预设值** |
| **passwd**  **修改用户密码** | **-S 显示密码相关信息**  **-l 使密码失效**  **-u 拿密码有效** |
| **chage** | **-l 显示用户关于密码的详细信息** |
| **usermod**  **修改用户** | **-l 修改用户名**  **其它跟useradd差不多** |
| **userdel**  **删除用户** | **-r 家目录也删除** |
| **id**  **查看用户信息** |  |
| **chsh**  **用户自己修改shell字段** |  |
| **groupadd**  **创建用户组** | **-g 指定GID**  **-r 创建系统群组** |
| **groupmod**  **修改群组** | **-g 修改GID**  **-n 修改名称** |
| **groupdel**  **删除群组** |  |
| **gpasswd** | **没加参数就是设置群密码**  **-A 设置群组管理员**  **-M 设置群组所有的用户成员**  **-r 移除群组密码**  **-R 使群组密码失效**  **群管理员**  **-a 将用户增加到群组中**  **-d 将用户从群组中删除** |
| **ACL**  **setfacl** | **-m 增加后面权限**  **-x 删除后面的权限**  **-b 删除所有的权限**  **-k 移除预设的ACL参数**  **-R**  **-d**  **u:用户:权限**  **setfacl –m u:caijt:rx 设定单一用户的权限**  **g:群组:权限**  **setfacl –m g:group2:rx**  **m:权限 就是最大权限，如果说u有rwx，但mask只有r，那也只有r的权限**  **setfacl –m d:u:caijt:rx 预设权限设定目录未来的文件acl** |
| **getfacl** |  |
| **su**  **切换登录身份** | **su – 切换到root身份，这个-不能忘**  **-c 以root身份运行命令**  **-l 切换用户** |
| **sudo** |  |
| **visudo** | **用户或群组 允许来源主机=可切换的身份 可以使用的命令**  **user ALL=(ALL) ALL**  **%group** |
| **write**  **给同时在线的用户发送信息** | **write user 终端号** |
| **wall**  **广播信息** |  |
| **mail**  **发送邮件** | **mail –s ‘hello’ caijt 发送邮件**  **mail 查看邮件** |
| **pwck**  **检查passwd文件账号是否有误** |  |
| **pwconv**  **检查账号与密码文件** |  |
| **pwunconv**  **这个是将密码文件shadow里面的密码写回passwd** | **这个指令最好不要** |
| **chpasswd**  **将未加密的密码，加密后写入shadow** | **echo “caijt:123456” | chpasswd 修改caijt的密码为123456** |
| **xfs\_quota** | **xfs\_quota –x –c “print”**  **xfs\_quota –x –c “report -ubih” /home**  **xfs\_quota –x –c “state” /home**  **xfs\_quota –x –c “limit –u bsoft=250m bhard=300m caijt” /home 设置用户**  **xfs\_quota –x –c “limit –g bsoft=950m bhard=1G group1” /home**  **xfs\_quota –x –c “timer –ug –b 14days” /home**  **项目容量限制，不太明白什么意思**  **echo “11:/home/cai” >> /etc/projects**  **echo “caiproject:11” >> /etc/projid**  **xfs\_quota –x –c “project –s caiproject”**  **xfs\_quota –x –c “report -ph” /home**  **xfs\_quota –x –c “limit –p bsoft=450m bhard=500m caiproject” /home**  **xfs\_quota- x –c “disable -up” /home 暂时取消quota的限制**  **xfs\_quota –x –c “enable -up” /home 恢复quota限制**  **xfs\_quota –x –c “off –up” /home 完全关闭**  **xfs\_quota –x –c “remove -p” /home 移除之前的项目限制** |
| **mdadm**  **磁盘阵列** | **mdadm –create /dev/md0 –auto=yes –level=5 –chunk=256K –radi-devices=3 –spare-devices=1 /dev/sda{5,6,7,8} 创建raid 5**  **mdadm –detail /dev/md0 查看信息**  **mdadm –manage /dev/md0 –fail /dev/sda7 使sda7分区损坏**  **mdadm –mange /dev/md0 –remove /dev/sda7**  **mdadm –manage /dev/md0 –add /dev/sda7**  **mdadm –stop /dev/md0** |
| **网络** | **ip address 查看网络ip地址**  **/etc/sysconfig/network-scripts/ 这里可以查看网卡配置**  **ifcfg-xxx一般是本地网卡，centos7安装后ONBOOT为no，说明默认是不开启的，改成yes后，service network restart重启下服务** |