

进一步熟悉和使用Linux系统

孟宁



关注孟宁

进一步熟悉和使用Linux系统

- ls、cd、pwd、cat、mkdir
- PATH环境变量以及常用快捷键
- 用户与用户组，useradd、groupadd
- 文件属性与权限操作，chmod、chown
- find 与 grep：搜索神器find命令基本用法
- tar命令
- ssh和sshd
- 实验：ssh登录到你的Linux系统，使用find、grep、tar等常用命令解决具体问题

PATH环境变量以及常用快捷键

- `echo $PATH`查看当前环境变量
- `export PATH=$PATH:/home/ubuntu/` 临时设置环境变量，当前会话有效
- `~/.bashrc`文件最末添加命令设置环境变量，然后重启shell，永久有效
- 对于不常用的命令， 目录+命令来执行
- `tab`可以补全命令
- `whereis` 查找命令所在目录

用户与用户组

- Linux系统是一个多用户多任务的分时操作系统，任何一个要使用系统资源的用户，都必须首先向系统管理员申请一个账号，然后以这个账号的身份进入系统。
- Linux用户分三大类：
 - root用户：超级用户，可以用来登录，操作系统任何文件和命令，拥有最高权限
 - 虚拟用户：不具有登录能力，系统本身拥有，不是后来添加的，但是系统运行不可缺的
 - 普通用户：有登录能力，权限受到限制

用户与用户组

- 添加用户时候，系统会自动
- 1：在/etc/passwd文件添加一行用户记录
- 2：在/home目录下创建新用户目录
- `$ sudo useradd -d /home/mengning -m mengning`
- `$ sudo passwd mengning`
- `$ ls /home/`
- `$ sudo userdel -r mengning`
- `whoami/groups`

用户与用户组

- 每个用户都有一个用户组，系统可以对一个用户组中的所有用户进行集中管理。不同Linux系统对用户组的规定有所不同，如Linux下的用户属于与它同名的用户组，这个用户组在创建用户时同时创建。
- 用户组的管理涉及用户组的添加groupadd、删除groupdel和修改groupmod。组的增加、删除和修改实际上就是对/etc/group文件的更新。

用户与用户组

- 与用户和用户组相关的信息都存放在一些系统文件中，这些文件包括/etc/passwd, /etc/shadow, /etc/group等。
- 每个用户都在/etc/passwd文件中有一个对应的记录行——用户名:口令:用户标识号:组标识号:注释性描述:主目录:登录Shell
- 加密后的用户口令字存放到/etc/shadow文件
- “组标识号”字段记录的是用户所属的用户组，对应着/etc/group文件中的一条记录。

文件属性与权限操作

- `$ ls -l`
- `-rwxr-xr-x 1 ubuntu ubuntu 430540 Dec 23 18:27 example.file`
- 文件的类型和权限、硬链接个数、User、Group、文件大小、日期、文件名

修改所属的用户或组

- `chown username filename`
- `chgrp groupname filename`
- `chown username.groupname filename`
- `chown` 和 `chgrp` 都有一个 `-R` 选项，该选项可以用来告诉它们递归地将所属权和组改变应用到整个目录树中
- `chown -R ubuntu /home/ubuntu`

文件属性与权限操作

- Linux的权限只有rwx三种：
- r(Read, 读取)：对文件而言，具有读取文件内容的权限；对目录来说，具有浏览目录的权限。
- w(Write,写入)：对文件而言，具有新增,修改,删除文件内容的权限；对目录来说，具有新建，删除，修改，移动目录内文件的权限。
- x(eXecute, 执行)：对文件而言，具有执行文件的权限；对目录来说该用户具有进入目录的权限。
- 例如drwxrwxrwx，文件类型、User、Group、Others
- 文件类型除了“-”普通文件外，“d”目录、“l”符号链接、“c”字符专门设备文件、“b”块专门设备文件、“p”先进先出、“s”套接字

修改文件的权限

- `chmod [ugo][=+-][rwx] filename`
- 用三个八进制数表示文件权限
- `chmod 777 filename = chmod ugo=rwx filename`

Linux的权限rwx的数值表示

- rwx 7
- rw- 6
- r-x 5
- r-- 4
- -wx 3
- -w- 2
- --x 1
- --- 0

find 与 grep

- find命令是一个无处不在命令，是linux中最有用的命令之一。find命令用于：在一个目录（及子目录）中搜索文件，你可以指定一些匹配条件，如按文件名、文件类型、用户甚至是时间戳查找文件。
- grep (global search regular expression(RE) and print out the line,全面搜索正则表达式并把行打印出来)是一种强大的文本搜索工具，它能使用正则表达式搜索文本，并把匹配的行打印出来。

find用法举例

- `find path -option [-print] [-exec -ok command] { } \;`
- `$find ~ -name "*.txt" -print` #在\$HOME中查.txt文件并显示
- `$find . -name "*.txt" -print`
- `$find . -name "[A-Z]*" -print` #查以大写字母开头的文件
- `$find /etc -name "host*" -print` #查以host开头的文件
- `$find . -name "[a-z][a-z][0-9][0-9].txt" -print` #查以两个小写字母和两个数字开头的txt文件

find用法举例

- `$find . -perm 755 -print`
- `$find . -perm -007 -exec ls -l {} \;` #查所有用户都可读写执行的文件同-perm 777
- `$find . -type d -print`
- `$find . ! -type d -print`
- `$find . -type l -print`
- `$find . -size +1000000c -print` #查长度大于1Mb的文件
- `$find . -size 100c -print` # 查长度为100c的文件
- `$find . -size +10 -print` #查长度超过10块的文件（1块=512字节）

grep用法举例

- `grep 'test' d*` 显示所有以d开头的文件中包含test的行。
- `grep 'test' aa bb cc` 显示在aa, bb, cc文件中匹配test的行。
- `sudo grep -v "#" /etc/host.conf`
- `ls -l | grep '^a'` 通过管道过滤ls -l输出的内容, 只显示以a开头的行。

grep正则表达式元字符

- ^ 锚定行的开始 如: '^grep'匹配所有以grep开头的行。
- \$ 锚定行的结束 如: 'grep\$'匹配所有以grep结尾的行。
- . 匹配一个非换行符的字符 如: 'gr.p'匹配gr后接一个任意字符, 然后是p。
- * 匹配零个或多个先前字符 如: '*grep'匹配所有有一个或多个空格后紧跟grep的行。 .*一起用代表任意字符。
- [] 匹配一个指定范围内的字符, 如'[Gg]rep'匹配Grep和grep。
- [^] 匹配一个不在指定范围内的字符, 如: '[^A-FH-Z]rep'匹配不包含A-F和H-Z的一个字母开头, 紧跟rep的行。
-

grep的选项

- -b 在每一行前面加上其所在的块号，根据上下文定位磁盘块时可能会用到
- -c 显示匹配到的行的数目，而不是显示行的内容
- -h 不显示文件名
- -i 比较字符时忽略大小写的区别
- -l (小写的字母L) 只列出匹配行所在文件的文件名（每个文件名只列一次），文件名之间用换行符分隔
- -n 在每一行前面加上它在文件中的相对行号
- -r 对目录下递归查询所有子目录
- -v 反向查找，只显示不匹配的行
- -w 把表达式作为词来查找，就好像它被\<和\>夹着那样。只适用于grep（并非所有版本的grep都支持这一功能，譬如，SCO UNIX就不支持）
-

tar命令

- `tar [-cxtzjvfpPN] 文件 目录/文件`
- `-c` : 建立一个压缩文件的参数指令(create 的意思);
- `-x` : 解开一个压缩文件的参数指令!
- `-t` : 查看 tarfile 里面的文件!
- `-z` : 是否同时具有 gzip 的属性? 亦即是否需要用 gzip 压缩?
- `-v` : 压缩的过程中显示文件! 这个常用, 但不建议用在背景执行过程!
- `-f` : 使用档名, 请留意, 在 f 之后要立即接档名喔! 不要再加参数!
- `-N` : 比后面接的日期(yyyy/mm/dd)还要新的才会被打包进新建的文件中!
- `--exclude FILE`: 在压缩的过程中, 不要将 FILE 打包!

ssh和sshd

- `sudo apt-get install openssh-server`
- `sudo ps -e |grep ssh` # 有sshd,说明ssh服务已经启动
- `sudo service ssh start`
- `sudo gedit /etc/ssh/sshd_config`

实验

- ssh登录到你的Linux系统，通过find找出/etc目录中所有*.conf并通过tar打包；通过grep将*.conf文件中包含“ubuntu”的行输出到一个文本文件