

• : 基本命令

1.Linux的基本原则:

- 1、由目的单一的小程序组成; 组合小程序完成复杂任务;
- 2、一切皆文件;
- 3、尽量避免捕获用户接口; (尽量不和用户进行交互, 就是一个程序一但开始运行, 就不需要用户进行任何操作, 如ls命令, ifconfig命令)
- 4、配置文件保存为纯文本格式;

用户接口: 是一种独特的应用程序, 能够为用户提供启动其他应用程序的机制

CLI: 命令提示符, 用户输入命令即可

shell: sh,csh,ksh,bash,zsh,tcsh

GUI:图形界面用户接口, 一般通过点击来启动应用程序

GNOME: C语言

KDE: C++

XFCE:轻量级

终端类型: tty命令可以查看

虚拟终端 (tty1-6) /dev/tty# #号为数字号码

伪终端(ssh、Telnet连接称为伪终端或者在图形界面下打开的终端) /dev/pts/#

物理终端 (console) /dev/console

串行终端 (使用特定速率向外打印字符, 通过串行接口连接) /dev/ttyS#

图形终端: 可以使用startx命令启动 startx -- :2 可以启动第二个图形终端

2.命令形式

命令格式:

命令 选项 参数

选项:

短选项: - 输入快捷简单

多个选项可以组合: -a -b = -ab

长选项: -- 意思明了

参数: 命令的作用对象

命令分为内部命令和外部命令

内部命令是系统shell自带的命令

外部命令是安装其他的程序产生的命令。

可以通过type 查看命令是内部还是外部命令: type 命令关键字

帮助的方式:

- 1、内部命令可以通过:help 命令 例子: help cd

- 2、外部命令可以通过:命令 --help 例如: ls --help
- 3、可以通过man命令获取命令帮助: 例如: man cd whatis 查看都有哪些man

章节

- 4、可以通过info命令获取命令帮助: 例如: info cd
- 5、通过百度\google等搜索引擎 可以寻求帮助
- 6、可以通过下载一个linux的中文命令大全

3.命令举例: ls

ls -l: 长格式

文件类型: 可使用file命令查看

-: 普通文件 (f)

d: 目录文件

b: 块设备文件 (block)就是存储数据以供系统存取的接口设备, 简单而言就

是硬盘

c: 字符设备文件 (character)即串行端口的接口设备, 例如键盘、鼠标等等

l: 符号链接文件(symbolic link file)

p: 命令管道文件(pipe)是一种特殊的文件类型, 它主要的目的是, 解决多个程序同时存取一个文件所造成的错误

s: 套接字文件(socket)这类文件通常用在网络数据连接。

我们可以启动一个程序来监听客户端的要求, 客户端就可以通过套接字来进行数据通信。

最常在 /var/run目录中看到这种文件类型。

文件权限: 9位, 每3位一组, 每一组: rwx(读, 写, 执行), r--

文件硬链接的次数

文件的属主(owner)

文件的属组(group)

文件大小(size), 单位是字节

时间戳(timestamp): 最近一次被修改的时间(mtime),可以使用stat命令可以看

到

访问:access atime 是在读取文件或者执行文件时更改的

修改:modify, mtime 是在写入文件时随文件内容的更改而更改的

改变:change, ctime metadata, 元数据 状态改变 是在写入文件、更改所有者、权限或链接设置时随inode 的内容更改而更改的。

-h: 做单位转换

-a: 显示以.开头的隐藏文件

. 表示当前目录

.. 表示父目录

-A

-d: 显示目录自身属性

-i: index node, inode

-r: 逆序显示

-R: 递归(recursive)显示

4.文件系统

FHS: filesystem hierarchy standard 文件系统层级标准

rootfs: 根文件系统 /

/boot: 系统启动相关的静态文件，如内核、initrd或者initramfs，以及grub(bootloader)

/dev: 所有设备的设备文件，设备文件也称为特殊文件（仅有元数据，没有数据，是设备的入口）

设备文件：

块设备：随机访问，数据块

字符设备：线性访问，按字符为单位

设备号：主设备号（major）和次设备号（minor） 主设备号相同的设备是同类设备(使用同一个驱动程序)

通过 cat /proc/devices 命令查看当前已经加载的设备驱动程序的主设备号

例如：ls -l /dev/sr0

brw-rw---- 1 root cdrom 11, 0 4月 1 08:59 /dev/sr0 11为主设备

号，0为次设备号

/etc: 配置文件

/home: 普通用户的家目录，每一个用户的家目录通常默认为/home/USERNAME

/root: 管理员的家目录；

/lib: 基本的共享库和内核模块

库文件 本质上来说库是一种可执行代码的二进制形式，可以被操作系统载入内存执行。

静态库，.a 静态库的代码在编译过程中已经被载入可执行程序，因此体积较大

动态库，.dll, .so (shared object) 共享库的代码是在可执行程序运行时才载入内存的，在编译过程中仅简单的引用，因此代码体积较小。

/lib/modules: 内核模块文件

/media: 为了移动设备的挂载点目录，例如光盘、U盘

/mnt: 挂载点目录，额外的临时文件系统

/opt: 可选目录，第三方程序的安装目录

/proc: 内核及进程信息的虚拟文件系统, 伪文件系统, 内核映射文件, 只存在内存中, 它以文件系统的方式为访问系统内核数据的操作提供接口。

/sys: 伪文件系统, 硬件设备信息虚拟文件系统接口, 跟硬件设备相关的属性映射文件, 该目录下是全局设备结构体系, 包含所有被发现的注册在各种总线上的各种物理设备

/tmp: 临时文件, /var/tmp

/var: 可变化的文件

 /var/log 日志信息

/bin: 基础的用户命令二进制程序, 为所有用户使用

/sbin: 系统二进制程序, 一般为管理命令

 /bin和/sbin 系统自身启动和运行时可能会用到的核心二进制命令

/usr: shared, read-only (Unix System Resource,即Unix系统资源的缩写), 不适合放在/bin或/etc目录下的额外的工具

 /usr/bin 大多数用户使用到的命令

 /usr/sbin 非核心的系统命令

 /usr/bin和/usr/sbin 系统运行时, 用户为完成某种操作可能用到的命令

 /usr/lib

/usr/local: 主要存放那些手动安装的软件

 /usr/local/bin

 /usr/local/sbin

 /usr/local/lib

/usr/share/man /usr/share/doc 帮助文件

应用程序的组成部分:

 二进制文件、库文件、配置文件、帮助文件

路径:

 绝对路径: 凡是以 "/" 开头的输入路径的方式都是绝对路径

 相对路径: 凡是以 "." 或者 ".." 开头的都是相对路径

 每个绝对路径都可以切割成2部分:

 基名: basename /path/to/somewhere 结果为somewhere

 目录名: dirname /path/to/somewhere 结果为/path/to

查看服务器基本信息:

 cat /proc/cpuinfo 查看cpu信息

 cat /proc/meminfo 查看内存信息

 free 查看内存使用情况

 uptime 监控CPU情况

uname -r 查看内核版本

cat /etc/redhat-release 查看发行版信息

hostname 查看主机名

hostnamectl set-hostname sunday-centos7 设置主机名为sunday-centos7, 会更改配置文件, 永久生效

ip addr 查看ip信息

ifconfig 查看ip信息

netstat 查看网络连接信息、路由表等, 常用来查看端口打开情况

ethtool eth0 查看网卡信息

fdisk -l 查看硬盘信息

df 查看设备挂载信息

- 5.其它常用命令

目录基本操作命令:

tree 树状图列出目录的内容

-C 在文件和目录清单加上色彩, 便于区分各种类型

ls 查看目录内容

cd 切换目录

pwd 查看当前目录位置

mkdir 创建目录、

-p 创建多级目录

-Z 设置安全上下文, 设置SELinux时有效

rmdir 删除空目录

rm 可以删除一个目录中的一个或多个文件或目录, 也可以将某个目录及其下属的所有文件及其子目录均删除掉。对于链接文件, 只是删除整个链接文件, 而原有文件保持不变。

-f 强制删除

-r或-R: 递归处理, 将指定目录下的所有文件与子目录一并处理;

mv 用来对文件或目录重新命名, 或者将文件从一个目录移到另一个目录中。

cp 用来将一个或多个源文件或者目录复制到指定的目的文件或目录。

-a: 此参数的效果和同时指定"-dpR"参数相同;

-d: 当复制符号连接时, 把目标文件或目录也建立为符号连接, 并指向与源文件或目录连接的原始文件或目录;

-f: 强行复制文件或目录, 不论目标文件或目录是否已存在;

-i: 覆盖既有文件之前先询问用户;

- p: 保留源文件或目录的属性;
- R/r: 递归处理, 将指定目录下的所有文件与子目录一并处理;

文件查看和处理命令:

touch 创建文件

touch 改变文件的atime和mtime (ctime会随之改变)

-a 改变atime

-m 改变mtime

stat 查看文件的属性及元数据

cat 用来显示文件的内容

cat m1 m2 > file (将文件m1和m2合并后放入文件file中)

-n 显示行号

通过标准输入生成文件testfile

```
[root@localhost ~]# cat > testfile << EOF
```

```
> Hello World!
```

```
> wellcome to china
```

```
> EOF
```

tac 倒序查看文件内容

more 显示文件内容, 每次显示一屏

按Space键: 显示文本的下一屏内容。

按Enter键: 只显示文本的下一行内容。

按h键: 显示帮助屏, 该屏上有相关的帮助信息。

按b键: 显示上一屏内容。

按q键: 退出命令。

less 分屏上下翻页浏览文件内容,用PageUp键向上翻页, 用PageDown键向下翻页

head 显示文件开头

tail 显示文件结尾

-f 监控某个文件变化

日期时间:

date、clock、hwclock、

cal 打开日历

bc 打开计算器

wc 统计文件的行数\单词数\字符数

- l 行数
- w 单词数
- c 字符数

du 统计文件和目录的大小

- s 汇总显示大小
- h 易读形式显示

alias 设置别名 alias a='mount /dev/cdrom /media'

vim : 三种模式: 命令模式、插入模式、末行模式

1. i l a A o O 进入到插入模式

2. : 末行模式 w 保存 q 退出 wq保存并且退出 ! 强制 三者可以结合使用

3.命令行模式: dd 删除[剪切] yy 复制 [n]yy 多行复制 p 粘贴

4.使vim显示行号, 在/root 下边创建一个 .vimrc 的文件, 使用vim打开, 添加set nu
, wq保存退出

命令模式下的操作:

- 1、翻页 Page Down或Ctrl+F 向下翻动一整页内容
Page Up或Ctrl+B 向上翻动一整页内容
- 2、行内快速跳转 Home键或 “^” 、数字 “0” 跳转至行首
End键或 “\$” 键 跳转到行尾
- 3、行间快速跳转 1G或者gg 跳转到文件的首行
G 跳转到文件的末尾行
#G 跳转到文件中的第#行
- 4、行号显示 :set nu 在编辑器中显示行号
:set nonu 取消编辑器中的行号显示
- 5、删除 x或Del 删除光标处的单个字符
dd 删除当前光标所在行
#dd 删除从光标处开始的#行内容
d^ 删除当前光标之前到行首的所有字符
d\$ 删除当前光标处到行尾的所有字符
- 6、复制 yy 复制当前行整行的内容到剪贴板
#yy 复制从光标处开始的#行内容
- 7、粘贴 p 小写 将缓冲区中的内容粘贴到光标位置处之后

P	大写	粘贴到光标位置处之前
8、查找	/word	从上而下在文件中查找字符串 “word”
	?word	从下而上在文件中查找字符串 “word”
	n	定位下一个匹配的被查找字符串
	N	定位上一个匹配的被查找字符串
9、取消操作	u	按一次取消最近的一次操作
		多次重复按u键，恢复已进行的多步操作
	U	用于取消对当前行所做的所有编辑
10、保存退出	ZZ	保存当前的文件内容并退出vi编辑器
11、可视化块操作界面	ctrl+v	进入块操作界面
	上、下、左、右箭头	可以进行选择块
	y	复制选择的快
	p	粘贴选择的快

末行模式的操作：

- 保存文件 :w 保存修改的内容
:w /root/newfile 另存为其它文件
- 退出vi :q 未修改退出
:q! 放弃对文件内容的修改，并退出vi
- 保存文件退出vi :wq 保存修改的内容并退出
- 打开新文件 :e ~/install.log 打开新的文件进行编辑
- 读入其它文件内容 :r /etc/filesystems 在当前文件中读入其他文件内容
- 替换 :s /old/new 将当前行中查找到的第一个字符 “old” 串替换为 “new”
:s /old/new/g 将当前行中查找到的所有字符串 “old” 替换为 “new”
:#,# s/old/new/g 在行号 “#,#” 范围内替换所有的字符串 “old” 为 “new”
:% s/old/new/g 在整个文件范围内替换所有的字符串 “old” 为 “new”
:s /old/new/c 在替换命令末尾加入c命令，将对每个替换动作提示用户进行确认