

## 7、linux命令使用细节和技巧小总结

到这里，我们已经学习了一些基础命令了，可以在前面学习的命令基础上，我们可以总结一下：

分析一下前面我们讲的命令的执行过程是：

用户输入命令——>传递到shell（命令解析器），shell把命令解析成二进制格式，这里用户的命令又分为**内部命令和外部命令**——>二进制命令进入系统内核，在内核中调用相关功能



**内部命令和外部命令的区别：**

内部命令：系统启动的时候这些命令就会被调入内存，常住在内存中，所以执行效率很高

外部命令：是系统软件的功能，在用户需要运行这些软件的时候，才从硬盘上把程序文件调入内存中，执行

**type 命令：**可以检测那些命令是外部命令那些是内部命令

**shell的提示符：**

完整的提示符：用户名---@---主机名---当前所在目录(~表示当前用户的家目录)---  
(# root/\$普通用户)

shell提示符有两个：“#”与“\$” 他们的区别

#表示是root用户登录，管理员账号登陆

\$表示普通用户登录

**shell（壳，命令解析器），可以有很多种**

查看所有shell类型

```
[root@centos7~]# cat /etc/shells
```

```
/bin/sh
```

```
/bin/bash
```

```
/sbin/nologin
```

```
/usr/bin/sh
```

```
/usr/bin/bash
```

`/usr/sbin/nologin`

`/bin/tcsh`

不同的用户使用不同的shell，这个取决于用户的配置，该配置在文件：`/etc/passwd`  
查看得知：`root`用户是使用**bash**这个shell！

linux不同文件类型对应的颜色：

- 白色：表示普通文件
- 蓝色：表示目录
- 绿色：表示可执行文件
- 红色：表示压缩文件
- 浅蓝色：链接文件
- 红色闪烁：表示软链接的文件有问题
- 黄色：表示设备文件
- 灰色：表示其它文件

用**alias**命令给复杂的命令做一个别名：

添加临时别名：

```
# alias viens33='vi /etc/sysconfig/network-scripts/ifcfg-ens33'
```

```
# viens33      //这时候，执行viens33命令就等同于执行vi
```

```
/etc/sysconfig/network-scripts/ifcfg-ens33
```

删除临时别名：`# unalias viens33`

上面的做法只是生成一个临时的别名，重启计算机，就失效了，这时如果我们想创建一个别名，重启计算机都不会失效，这时我们就必须要修改某个配置文件了，因为前面我们就已经讲过，Linux一切皆是文件，永远有效的配置一定是保存在某个文件中的，所以这个别名配置对应的文件：`# vim /root/.bashrc`

把这个命令：`alias viens33='vi /etc/sysconfig/network-scripts/ifcfg-ens33'` 写入文件就OK了！

注意1：这个文件目录`/root/.bashrc`，是`root`下的文件，所以这个配置只会对`root`用户生效，换了个用户，又会失效！如果要让这个别名对所有用户都有效，显然我们前面讲过，配置文件都是放在`/etc`下的，所以对应的全局别名配置文件就应该是`# vim /etc/bashrc`，配置内容和上面一致！

注意2：修改了**bashrc**文件后，要用**source /root/.bashrc** 命令重载入一些这个配置文件，然后这个别名才会生效！

**source**命令使Shell读入指定的Shell程序文件并依次执行文件中的所有语句

### 通过历史命令快速输入再执行一次：

方法1： 上下箭头键

方法2： 快捷键ctrl+r —>输入某条命令的关键字 —>找出来需要的命令，用右光标键点击，重执行的命令

方法3： 执行history命令——>查看历史命令的编号，然后！ 编号，也等于重新执行命令

### Linux快捷键总结：

结束运行中的程序： ctrl+c

退出终端： ctrl+d

清空屏幕： ctrl+L

打开历史命令收索： ctrl+r

命令补全： tab键

vi编辑文件的时候，快速定位光标到最后一行： G