# Shell 命令集锦

----- 来源于日常使用收集

# Linux 介绍

## 1.1 何为Shell

Shell 在计算机科学中称为课（区别于内核kernel ）,为使用者提供操作界面的程序，类似于Window下的cmd.exe 或者 command.exe 程序。同时他还是一门程序设计语言，交互式的执行或者解释用户输入的命令，或者自动的解释或者执行用户预先定义好的一连串命令（shell 脚本），定义了变量和参数，并且提供了高级语言中才具有的控制结构，例如 if,while,for 等。 【[百度百科](https://baike.baidu.com/item/shell/99702?fr=aladdin)】

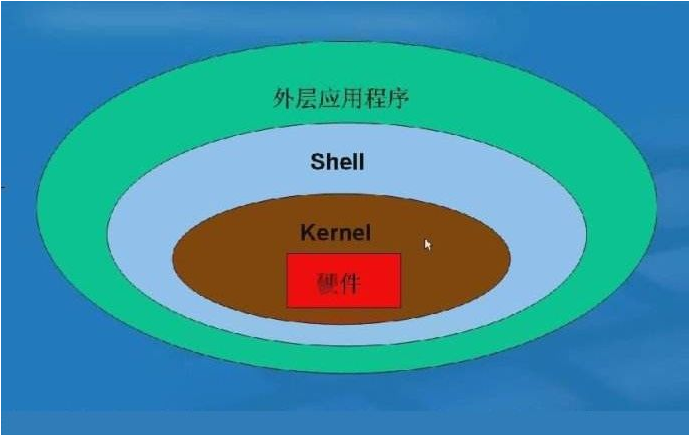
## 1.2 Linux 下Shell的类型

命令行式shell（Command Line Interface shell ，即CLI shell）

1. /bin/sh
2. /bin/bash
3. /bin/ksh
4. /bin/csh



Shell 解释器架构：



## 1.3 文件系统

### 1.3.1 物理设备的命名规则

在Linux系统中一切都是文件，硬件设备也不例外。既然是文件，就必须有文件名称。系统内核中的udev设备管理器会自动把硬件名称规范起来，目的是让用户通过设备文件的名字可以猜出设备大致的属性以及分区信息等；这对于陌生的设备来说特别方便。另外，udev设备管理器的服务会一直以守护进程的形式运行并侦听内核发出的信号来管理/dev目录下的设备文件。Linux系统中常见的硬件设备的文件名称如下图所示。

常见的硬件设备及其文件名称

硬件设备 文件名称

IDE设备 /dev/hd[a-d]

SCSI/SATA/U盘 /dev/sd[a-p]

软驱 /dev/fd[0-1]

打印机 /dev/lp[0-15]

光驱 /dev/cdrom

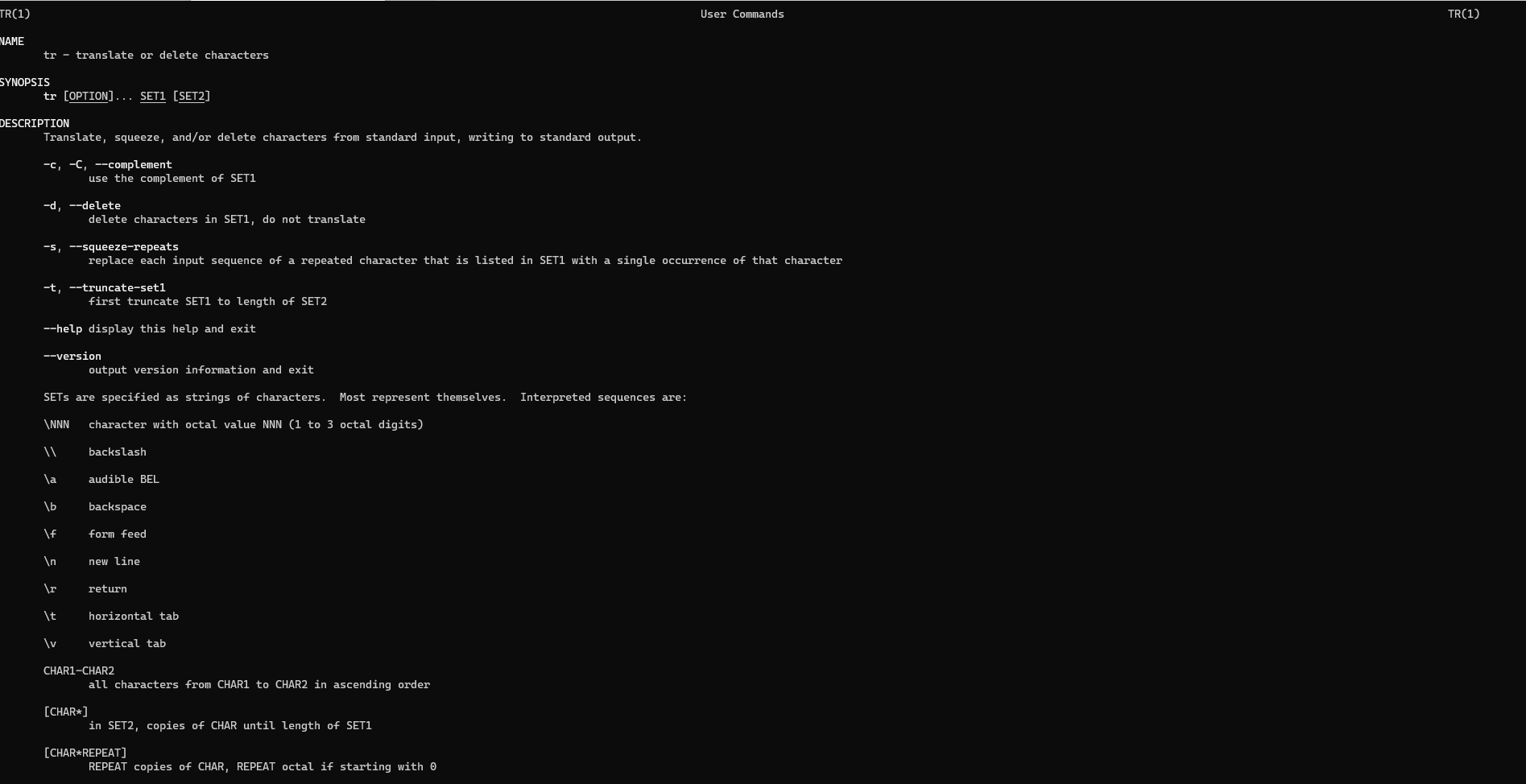
鼠标 /dev/mouse

磁带机 /dev/st0或/dev/ht0

# 常用命令

## 2.1 文字处理命令

### 2.1.1 tr命令 – 转换或者删除字符串





-sh-root>> tr –help

返回结果：



例子： 小写转换为大写

-sh-root#] echo “this is a lower word!” | tr [:lower:] [:upper:]

结果： THIS IS A LOWER WORD

例子：大写转换为小写

-sh-root#] echo “WORD” | tr [:upper:] [:lower:]

结果：word

例子：删除字符 – 删除数字

sh-root#] echo “hello 123 world 456” | tr –d ‘0-9’

结果: hello world

例子：将制表符换位空格

sh-root#] echo -e "this is \t name" | tr '\t' ' '

结果：this is name

例子：生成固定长度的随机密码

sh-root#] head /dev/urandom | tr -dc A-Za-z0-9 | head -c 20

### 2.2.2 sort 命令

sort 命令是将文件进行排序，并将排序结果标准输出。Sort 命令即可以从特定的文件，也可以从stdin 中获取输入。

【语法】

sort （选项）（参数）

【选项】

* -b : 忽略每行前面开始出的空格字符
* -c: 检查文件是否已经按照顺序排序
* -d: 排序时，将小写字母视为大写字母
* -i: 排序时，除了040至176之间的ASCII字符外，忽略其他的字符
* m: 将几个排序号的文件进行合并
* -n: 依照数值的大小进行排序
* -o: 输出文件，将排序后的结果存入制定的文件中
* -r: 以相反的顺序来排序
* -k : 指定要排序的栏位

【参数】

文件：制定待排序的文件列表

【示例】

sort –nrk 3 –t: sort.txt

-n 是按照数字进行排序，-r 是相反顺序，-k 是指定要排序的栏位（这里是3），-t 指定栏位分隔符号为冒号（：）

### 2.2.3 uniq 命令

Uniq 命令 用于报告或者忽略文件中的重复行，一般与sort命令 结合使用

【语法】

uniq （选项）（参数）

【选项】

## 2.2 网络管理

### 2.2.1 firewall-cmd命令

Firewall-cmd 是firewalld 的字符界面管理工具，firewalld 是CentOs7的一大特性，最大的好处有两个: 支持动态更新，不用重启服务。

Firewalld 跟iptables 比起来至少有两大好处:

（1）， firewalld 可以动态的修改单条规则，而不需要像iptables那样，在修改了规则后必须得全部刷新才能生效。

（2）， firewalld 使用方便

【语法】

firewalld （选项）（参数）

【选项】

-h, --help # 显示帮助信息

-V, --version # 显示版本信息

-q # 不打印状态消息

状态选项

--state # 显示firewalld 的状态

--reload # 不中断服务的重新加载

--complete-realod # 中断所有连接的重新加载

--runtime-to-permanent # 将当前防火墙的规则永久保存

**--check-config**  # 检查配置正确性

日志选项

--get-log-denied # 获取记录被拒绝的日志

端口转发

端口转发可以将指定地址访问制定的端口时，将流量转发值制定地址的制定端口，转发的目的如果不是制定IP的话就默认为本机，如果指定了IP却没指定端口，则默认使用来源端口，如果配置好端口转发之后不能用，可以检查下面两个问题。

（a）,比如我将3307 端口的流量转发至3306,3306是mysql数据库服务器使用的端口，检查3307端口是否开放。

（b）, 检查是否允许伪装IP，没有允许的话要开启伪装IP

Firewall-cmd –add-forward-port=port=3307:proto=tcp:toport=3306 # 将3307端口的流量转发至3306

Firewall-cmd --add-forward-port=port=3307:proto-tcp:toaddr=192.168.18.56 #将端口的流量转发至 192.168.18.56 的端口3307处

firewall-cmd --add-forward-port=port=3307:proto=tcp:toaddr=192.168.18.56:toport=3306 # 将3307端口的流量转发至192.168.18.56的3306端口

1. 当我们想把某个端口隐藏起来的时候，就可以在防火墙上阻止那个端口访问，然后再开一个不规则的端口，之后配置防火墙的端口转发，将流量转发过去。
2. 端口转发还可以做流量分发，一个防火墙拖着好多台运行着不同服务的机器，然后用防火墙将不同端口的流量转发至不同机器。

查看设置:

Firewall-cmd –state # 显示状态

Firewall-cmd –get-active-zone # 查看区域信息

Firewall-cmd –panic-on # 拒绝所有包

Firewall-cmd --panic-off # 取消拒绝状态

Firewall-cmd --reload # 更新防火墙规则

Firewall-cmd --complete-reload # 更新防火墙规则（断开连接）

Firewall-cmd –-zone=public –add-interface=eth3 –permanent

# 将接口eth3添加到public并永久生效

Firewall-cmd --set-default-zone=public

# 设置默认接口区域

Firewall-cmd –zone=public –list-ports

# 查看所有打开的端口

Firewall-cmd –zone=public –add-port=3306/tcp --permanent

#加入一个端口到区域public 并永久生效

Firewall-cmd –zone=public –add-service=smtp

# 打开一个服务，类似于将端口可视化，服务需要在配置文件中添加，/etc/firewalld 目录下有services文件夹，这个不详细说了，详情参考文档

Firewall-cmd –zone=public –remove-service=smtp

# 移除smtp服务

# 显示支持的区域列表

Firewall-cmd –get-zones

# 设置为家庭区域

Firewall-cmd --get-default-zone=home

# 查看当前域

Firewall-cmd –get-active-zones

# 设置当前域的接口

Firewall-cmd –-set-zone-of-interface=enp03s

# 临时修改网络接口（enp0s3）为内部区域（internal）

firewall-cmd --zone=internal --change-interface=enp03s

# 永久修改网络接口enp03s为内部区域（internal）

firewall-cmd --permanent --zone=internal --change-interface=enp03s

控制端口 / 服务

可以通过两种方式控制端口的开放，一种是指定端口号另一种是指定服务名。虽然开放 http 服务就是开放了 80 端口，但是还是不能通过端口号来关闭，也就是说通过指定服务名开放的就要通过指定服务名关闭；通过指定端口号开放的就要通过指定端口号关闭。还有一个要注意的就是指定端口的时候一定要指定是什么协议，tcp 还是 udp。知道这个之后以后就不用每次先关防火墙了，可以让防火墙真正的生效。

firewall-cmd --add-service=mysql # 开放mysql端口

firewall-cmd --remove-service=http # 阻止http端口

firewall-cmd --list-services # 查看开放的服务

firewall-cmd --add-port=3306/tcp # 开放通过tcp访问3306

firewall-cmd --remove-port=80tcp # 阻止通过tcp访问3306

firewall-cmd --add-port=233/udp # 开放通过udp访问233

firewall-cmd --list-ports # 查看开放的端口

伪装 IP

firewall-cmd --query-masquerade # 检查是否允许伪装IP

firewall-cmd --add-masquerade # 允许防火墙伪装IP

firewall-cmd --remove-masquerade# 禁止防火墙伪装IP

服务管理

# 显示服务列表

Amanda, ftp, Samba和tftp等最重要的服务已经被FirewallD提供相应的服务，可以使用如下命令查看：

firewall-cmd --get-services

# 允许ssh服务通过

firewall-cmd --enable service=ssh

# 禁止SSH服务通过

firewall-cmd --disable service=ssh

# 打开TCP的8080端口

firewall-cmd --enable ports=8080/tcp

# 临时允许Samba服务通过600秒

firewall-cmd --enable service=samba --timeout=600

# 显示当前服务

firewall-cmd --list-services

# 添加HTTP服务到内部区域（internal）

firewall-cmd --permanent --zone=internal --add-service=http

firewall-cmd --reload

# 在不改变状态的条件下重新加载防火墙