**案例1.写一个进度条脚本**

#!/bin/bash

jindu(){ #定义函数

while : #死循环

do

echo -ne '\033[42m \033[0m' #输出带颜色的进度条

#echo -n ‘#’

sleep 0.3 #调整输出节奏

done

}

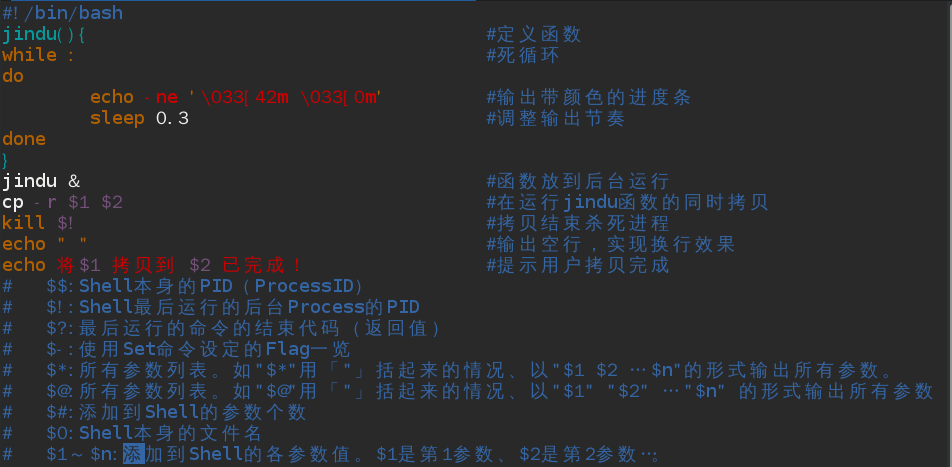
jindu & #函数放到后台运行

cp -r $1 $2 #在运行jindu函数的同时拷贝

kill $! #拷贝结束杀死进程

echo " " #输出空行，实现换行效果

echo 将$1 拷贝到 $2 已完成！ #提示用户拷贝完成



**案例2.一键部署nginx源码包**

#!/bin/bash

tar -xf nginx-1.8.0.tar.gz #tar解包

cd nginx-1.8.0 #cd到源码包目录下

yum -y install gcc pcre-devel openssl-devel #安装依赖软件

./configure #./configure 配置安装目录

make #make编译成可执行二进制程序

make install #make install安装，将编译好的文件复制到安装目录







**案例3.启动服务的脚本**

#!/bin/bash

case $1 in

start)

/usr/local/nginx/sbin/nginx #这个路径是启动nginx，没提示，成功

echo "开启服务";;

stop)

/usr/local/nginx/sbin/nginx -s stop

echo "关闭服务";;

restart)

/usr/local/nginx/sbin/nginx -s stop

/usr/local/nginx/sbin/nginx

echo "重启服务";;

status)

netstat -ntulp | grep nginx &> /dev/null

if [ $? -eq 0 ];then

echo "已经启动"

else

echo "已经关闭"

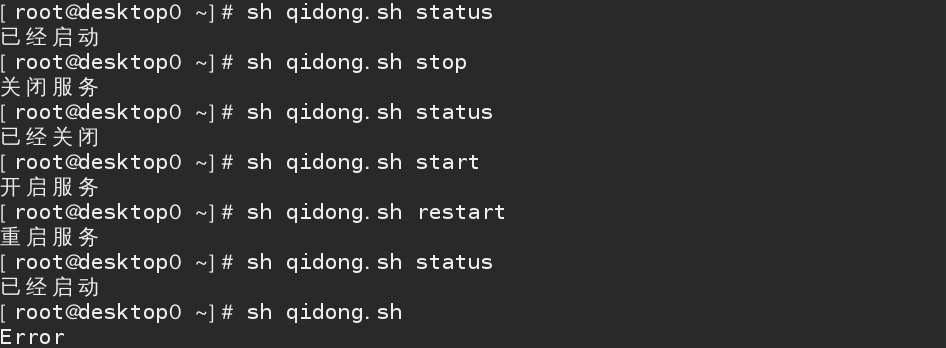
fi;;

\*)

echo Error;;

esac

#nginx不受systemctl管理



**案例4.监控脚本（awk过滤）**

#!/bin/bash

uptime | awk '{print "CPU负载：" $10}'

ifconfig eth0 | awk '/RX packets/{print "网卡接收数据流量：" $5}'

ifconfig eth0 | awk '/TX packets/{print "网卡发送数据流量：" $5}'

free | awk '/Mem:/{print"内存剩余：" $4}'

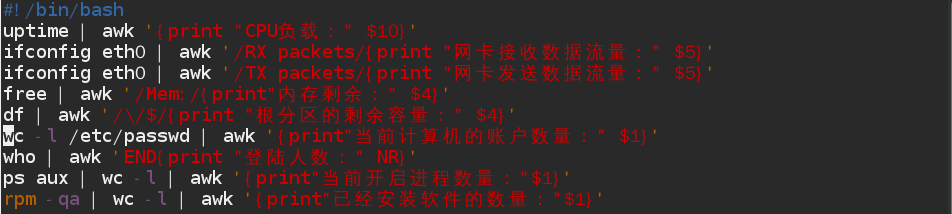
df | awk '/\/$/{print "根分区的剩余容量：" $4}'

wc -l /etc/passwd | awk '{print"当前计算机的账户数量：" $1}'

who | awk 'END{print "登陆人数：" NR}'

ps aux | wc -l | awk '{print"当前开启进程数量："$1}'

rpm -qa | wc -l | awk '{print"已经安装软件的数量："$1}'





**案例5.检查/var/log/secure，看看有没有人尝试破解密码，将其加入防火墙，block或者drop**

#!/bin/bash

awk '/Failed/{print $11}' /var/log/secure >> ip.txt

for i in `cat ip.txt`

do

firewall-cmd --zone=block --add-source=$i &> /dev/null

done

echo 已将尝试破解密码IP加入防火墙，设置为阻塞状态！



