shell day 01

================================================

准备linux虚拟机，开启192.168.4.7一台即可

bash优点， 支持tab键、快捷键、历史命令、别名、重定向、管道

交互式指令 passwd

非交互式指令 ls

脚本的编写规范：

1. 声明解释器

#!/bin/bash

1. 注释

#内容可以是脚本功能的描述，设计思路，脚本中各种参数的作用

1. 执行指令

ls

cd /

脚本的执行方式：

1. 添加x权限，使用绝对或者相对路径执行

chmod +x test01.sh

1. 使用解释器执行脚本，无需x权限，会开启解释器

子进程

bash test01.sh

1. 使用source命令执行脚本，无需x权限，不会开启

解释器的子进程

source test01.sh

测试脚本一： 了解用解释器或source执行脚本的不同

#!/bin/bash

echo nb

mkdir abc

cd abc

//使用bash执行看不到进入abc目录的效果，因为bash开启了子进程

执行完毕后退出了，父进程并没有进入abc目录，而使用source执行脚

本可以看到进入abc目录，因为没有开启子进程，而是父进程亲自进入

该目录

思考： 下列脚本用解释器或source执行有什么区别？

#!/bin/bash

echo nb

exit

**编写部署yum的脚本**

#!/bin/bash

#搭建yum仓库

mkdir /yum

mount /dev/cdrom /yum

rm -rf /etc/yum.repos.d/\*.repo

echo "[abc]

name=abcabc

baseurl=file:///yum

gpgcheck=0

enabled=1" > /etc/yum.repos.d/abc.repo

**编写脚本，部署httpd服务，定义默认页面是”web-test~~~”**

#!/bin/bash

yum -y install httpd

systemctl restart httpd

systemctl enable httpd

echo “web-test~~~” > /var/www/html/index.html

**测试时需要临时关闭防火墙**

systemctl stop firewalld

运行脚本后，可以利用真机的浏览器打开虚拟机的网页

http://192.168.4.7

**编写部署ftp服务的脚本，要求可以实现开机自启**

#!/bin/bash

yum -y install vsftpd &> /dev/null

systemctl restart vsftpd

systemctl enable vsftpd &> /dev/null

常量 固定不变的值

变量 以固定的名称存放肯能会变化的值，可以提高脚本对环境

变化的适应力

1. 自定义变量，变量的名称可以是数字、字母、下划线，不能以

数字开头，不能使用特殊符号

变量名称=变量的值 （创建变量，定义变量，给变量赋值）

abc=10 //定义变量

echo $abc //调用变量，显示变量的值，此时$a相当于10

abc=20 //如果多次赋值，最后一次生效

abc= //将变量的值定义为空，相当于取消变量的定义

unset abc //使用unset命令取消变量的定义

echo ${abc}RMB //为变量添加大括号避免与后续内容发生混淆

1. **环境变量，是由系统提前定义好，使用时直接调用**

USER当前用户名 、UID当前用户的id号

、HOME当前用户的家目录 、SHELL当前用户的解释器

、HOSTNAME服务器名字、PS1一级提示符 、PS2二级提示符

PWD显示当前所在位置、PATH存储执行指令的路径

1. 预定义变量与位置变量

$\* $# $$ $? $1 $2 $3 …..

#!/bin/bash

echo $1 执行脚本后跟的第一个位置参数

echo $2 执行脚本后跟的第二个位置参数

echo $3 执行脚本后跟的第三个位置参数

echo $\* 所有位置参数

echo $# 所有位置参数的个数

echo $$ 随机的进程号

echo $? 判断上一条指令是否执行成功，如果成功显示0，如果失败，则

显示非0

编写脚本，可以创建用户abc，并且配置密码123456

#!/bin/bash

useradd abc

echo 123456 | passwd --stdin abc

#!/bin/bash

useradd $1

echo $2 | passwd --stdin $1

env | grep HOST //env可以显示所有的环境变量，加

了grep可以过滤查询

abcd=8888 //定义测试变量

set | grep abcd= //使用set查询所有变量，可以配

合grep过滤查询

--------------------------------------------------

变量的扩展知识

1. **引号与反撇号**

**"" 双引号 界定范围**

touch "a b" //可以创建a空格b的文件

a=10

echo "$a" //可以调用变量

''  **单引号 界定范围 屏蔽特殊符号的功能**

touch 'a b' //也可以创建a空格b的文件

echo '$a' //无法调用变量，$是调用变量的特殊符号，被单引号屏蔽

``  **反撇号 调用命令的执行结果，或者使用$( )**

a=date //定义变量时如果使用命令则不会直接识别

a=`date` //使用反撇号可以将命令的执行结果赋值给变量

a=$(date) //效果同上

2，使用read指令，以交互式使用脚本

#!/bin/bash

read -p "请输入用户名" u

useradd $u

read -p "请输入密码" p

echo $p | passwd --stdin $u

stty -echo 屏蔽回显

stty echo 恢复回显

**改良版：输入密码时不显示**

#!/bin/bash

read -p "请输入用户名：" u

useradd $u

stty -echo

read -p "请输入密码:" p

stty echo

echo $p |passwd --stdin $u

下课休息 17:35回