牛犇第一阶段:admin(系统管理)

day01:云网络基础(TCP/IP,配置IP)

day02:Liunx系统简介与安装

day03:命令行基础,目录与文件管理

day04:权限和归属

day05:rpm,yum,磁盘分区格式化,NTP网络时间协议

day06:LVM逻辑卷

第一阶段:engineer(应用管理)

day01:shell基础,变量,判断,循环

day02:selinux概述,alias别名,firewalld防火墙策略

day03:iscsi技术应用,数据库概述及基础

day04:NFS共享,http基础

day05:动态网站,安全web

day06:邮件服务,parted分区,swap分区,配置ipv6,链路聚合

第一阶段:services(系统&服务管理进阶)

day01:ln,date,histroy,man,du,zip等命令,自定义yum仓库,vim技巧

day02:kvm的组成,virsh工具,镜像管理,写时复制技术(cow)

day03:DNS服务基础,缓存DNS(forward)

day04:Spit分离解析,RAID磁盘阵列,进程管理,日志管理,systemctl控制

day05:PXE装机

day06:rsync,inotify实时同步,Cobbler装机

##################################################################

一 教学环境

1.真机是Linux 账户:student 密码:tedu

2.虚拟化软件: 虚拟众多的硬件

虚拟机被损坏,与真机无关

########################################

典型服务模式

• C/S,Client/Server架构

– 由服务器提供资源或某种功能

– 客户机使用资源或功能

##############################################

TCP/IP协议简介

• TCP/IP是最广泛支持的通信协议集合

– 包括大量Internet应用中的标准协议

– 支持跨网络架构、跨操作系统平台的通信

• 主机与主机之间通信的三个要素

– IP地址(IP address)

– 子网掩码(subnet mask)

– IP路由(IP router):涉及路由器

#################################################

IP地址(IP address):唯一标识网络中一个节点地址

• 地址组成(点分十进制):

– 一共32个二进制数

　11110000.00001111.10101010.11111100

　　 11111111 = 255

例子： 　1.2.3.4 192.168.1.1 10.11.12.13

#################################################

为虚拟机win2008配置IP地址

利用管理员进行登录，密码为：tedu

配置IP地址：　　192.168.1.1　　子网掩码：255.255.255.0

– 右击桌面网络 --＞属性 --＞更改适配器设置

– 双击“本地连接” --＞ 属性

– 双击“Internet协议版本4(TCP/IPv4)”

– 配置完成后,单击“确定”完成

查看IP地址：

– 右击桌面网络 --＞属性 --＞更改适配器设置

– 双击“本地连接” --＞ 详细信息

采用ipconfig命令的方式，查看IP地址：

　windows键　＋　ｒ＝　开始＋运行

　　　输入cmd-->敲回车--＞输入　ipconfig-->敲回车

　　　--＞输入　exit　关闭命令行界面

################################################

• IP地址的分类

– A类:1 ~ 127 　网+主+主+主

– B类:128 ~ 191 　 网+网+主+主

– C类:192 ~ 223 　网+网+网+主

• 组播及科研专用

– D类:224 ~ 239 组播

– E类:240 ~ 254 科研

192.168.1.1=只看ip地址的第一个数字,进行判别分类

################################################

IP地址由网络位与主机位，两部来组成

网络位=座机号码的区号，标识一个网络

主机位=编号

子网掩码：用来区分IP地址的网络位与主机位

　　　　　利用二进制的１标识网络位

　　　　　利用二进制的0标识主机位

　192.168.1.1＝11000000.10101000.00000001.00000001

11111111.11111111.11111111.00000000

255.255.255.0

　座机号码：区号－号码（编号）

　　北京：010－12345678＝北京区域，编号为12345678的座机

　　石家庄：0311－7354444＝石家庄区域,编号为7354444的座机

　区号：标识区域

　192.168.1.1　＝　192.168.1网络，编号为１的主机

　192.168.１.2　＝　192.168.１网络，编号为２的主机

　相同网络通信，网络位相同

　192.168.1.1 255.255.255.0 = 192.168.1.1/24

192.168.1.1/24 :该IP地址有24个网络位

　1.该IP地址是哪一类 　 2.判别网络位与主机位

################################################

克隆虚拟机win2008

1.关闭虚拟机win2008

2.进行虚拟机克隆

　双击＂虚拟系统管理器＂--＞右击＂win2008＂--＞克隆--＞克隆

3.修改win2008-clone桌面背景

双击＂控制面板＂--＞显示--＞更改桌面背景---＞保存修改

4.配置win2008-clone的IP地址：192.168.1.2 255.255.255.0

– 右击桌面网络 --＞属性 --＞更改适配器设置

– 双击“本地连接” --＞ 属性

– 双击“Internet协议版本4(TCP/IPv4)”

– 配置完成后,单击“确定”完成

采用ipconfig命令的方式，查看IP地址：

　windows键　＋　ｒ＝　开始＋运行

　　　输入cmd-->敲回车--＞输入　ipconfig-->敲回车

　　　--＞输入　exit　关闭命令行界面

################################################

采用命令方式，测试网络联通性：

　　ping　　命令

　　ping 　对方IP地址　　　ping命令是双向，有去有回

　　ping 　192.168.1.1

　　ping 　192.168.1.２

###############################################

　ftp://172.40.50.114

###############################################

网关地址:一个网络到另一个网络的关口　（解决不同网络通信）

DNS服务器：将网站的域名解析为,对应的IP地址

www.qq.com-------＞腾讯的服务器

www.baidu.com-------＞百度的服务器

###############################################

#################################################################

day02

Linux系统简介

什么是Linux?

Linux是一种操作系统：可以让计算机硬件正常工作

Unix/Linux发展史

• UNIX诞生，1970-1-1

Linux的诞生

• Linux之父,Linus Torwalds

– 1991年10月,发布0.02版(第一个公开版)内核

– 1994年03月,发布1.0版内核

– 标准读音:　哩呐科斯

版本号:主版本.次版本.修订号

Linux发行版本

• 发行版的名称/版本由发行方决定

– Red Hat Enterprise Linux（RHEL） 5/6/7

– Suse Linux Enterprise 12

– Debian Linux 7.8

– Ubuntu Linux 14.10/15.04

– ......

################################################

• CentOS,社区企业操作系统

– Community Enterprise Operating System

– http://www.centos.org/

• 基于Linux的企业服务器

• 嵌入式系统

• 高性能大型运算

###############################################

安装Linux系统

CentOS,社区企业操作系统,建议2G以上内存

Ctrl + Alt = 鼠标回到真机

#################################################

预备知识

Linux目录结构：树型结构

• 最顶层为根目录(/)

– Unix/Linux的基本哲学理念:一切皆文件

　　根目录(/)：所有的数据都在此目录下（Linux系统的起点）

路径：/abc/nsd/1.txt

/dev：存放设备(硬盘或键盘或鼠标或显示器.....)相关的数据

Linux管理员用户为:root 　　进行登录

Linux磁盘表示

　hd,表示IDE设备

　sd,表示SCSI设备

　vd,表示虚拟设备

　/dev/sda:表示SCSI设备，第一块

　/dev/sdb:表示SCSI设备，第二块

　/dev/sdc:表示SCSI设备，第三块

　/dev/sdd:表示SCSI设备，第四块

#################################################

Linux基本操作

命令行基本操作

获取命令行界面

• 虚拟控制台切换( Ctrl + Alt + Fn 组合键)

– tty1:图形桌面

– tty2~tty6:字符控制台

• 右键 "打开终端"

编辑--->配置文件首选项--->修改字体大小

• 命令行提示标识的含义

– [当前用户@主机名 工作目录]$

– 若当前用户是root,则最后一个字符为 #

[root@svr7 桌面]#

– 否则,最后一个字符为 $

[teacher@svr7 桌面]$

Ctrl shift 　+ =　终端字体变大

Ctrl － =　终端字体变小

##################################################

查看及切换目录

• pwd — Print Working Directory

– 用途:查看当前工作目录

• cd — Change Directory

– 用途:切换工作目录

– 格式:cd [目标文件夹位置]

• ls — List

– 格式:ls [选项]... [目录或文件名]...

[root@localhost ~]# pwd #显示当前所在位置

[root@localhost ~]# cd /abc　　　　＃切换到/abc目录

bash: cd: /abc: 没有那个文件或目录

[root@localhost ~]# cd /　　　　　＃切换到　/　目录

[root@localhost /]# pwd

[root@localhost /]# ls　　　　　＃显示当前目录内容

[root@localhost /]# cd /boot

[root@localhost /boot]# pwd

[root@localhost /boot]# ls

[root@localhost /]# cd /opt

[root@localhost /opt]# pwd

[root@localhost /opt]# ls

　颜色：

　　蓝色：目录

　　黑色：文件

[root@localhost /]# ls

[root@localhost /]# ls /root

[root@localhost /]# ls /home

[root@localhost /]# ls /opt

[root@localhost /]# ls /boot

[root@localhost /]# ls /root

]# ls /root/anaconda-ks.cfg

]# cat /root/anaconda-ks.cfg #查看文件内容

]# cat /etc/passwd　　　#查看文件内容

]# cat /etc/hosts　　　　#查看文件内容

]# cat /etc/fstab　　　　#查看文件内容

]# cat /etc/redhat-release　#查看当前系统版本

###############################################

以 / 开始的绝对路径

以当前为参照的相对路径

[root@localhost /]# cd /etc/pki/

[root@localhost pki]# pwd

[root@localhost pki]# ls

[root@localhost pki]# cd /etc/pki/CA #绝对路径

[root@localhost CA]# pwd

[root@localhost CA]# cd /etc/pki/

[root@localhost pki]# pwd

[root@localhost pki]# ls

[root@localhost pki]# cd CA　　　　＃相对路径

[root@localhost CA]# pwd

[root@localhost CA]# cd /

[root@localhost /]# ls root #相对路径

[root@localhost /]# ls /root #绝对路径

###############################################

　　.. :表示上一级目录

[root@localhost /]# cd .. 　　　　#后退

[root@localhost /]# cd /etc/pki/

[root@localhost pki]# pwd

/etc/pki

[root@localhost pki]# cd ..

[root@localhost etc]# pwd

/etc

[root@localhost etc]# cd ..

[root@localhost /]# pwd

/

[root@localhost /]# cd /etc/pki/

[root@localhost pki]# cd ../..　　　＃一次性后退两层

[root@localhost /]# pwd

#################################################

• 真机列出CPU处理器信息

[root@svr7 桌面]# lscpu

.......

CPU(s): 4

.......

型号名称： Intel(R) Core(TM) i5-4430S CPU @ 2.70GHz

.......

• 真机检查内存大小、空闲情况

[root@svr7 桌面]# cat /proc/meminfo

MemTotal: 16330728 kB

.......

查看主机名和IP信息

• 列出当前系统的主机名称

[root@localhost /]# hostname　　#查看当前主机名

localhost.localdomain

[root@localhost /]# hostname　 A.tedu.cn #临时设置

[root@localhost /]# hostname　　#查看当前主机名

[root@localhost /]# exit 　　 #关闭当前终端

开启一个新的终端进行验证：

[root@A /]#　hostname

• 列出已激活的网卡连接信息

[root@A ~]# ifconfig　　　　　　　　　　　#查看IP地址信息

[root@A ~]# ifconfig eth0 192.168.1.1　#临时设置IP地址

[root@A ~]# ifconfig eth0

[root@A ~]# ping 192.168.1.1

按Ctrl + c 结束正在运行的命令

lo: 本地回环接口，专用于测试

inet 127.0.0.1 netmask 255.0.0.0

　　127.0.0.1：永久代表本机

[root@A ~]# ping 127.0.0.1

按Ctrl + c 结束正在运行的命令

##########################################

可以永久设置主机名与IP地址与子网掩码与网关地址与DNS服务器地址

修改主机名：设置系统主机名-->输入新的名字-->确定-->确定

[root@A ~]# nmtui

[root@nsd1903 ~]# hostname

nsd1903.tedu.cn

[root@nsd1903 ~]#

修改IP地址与子网掩码

[root@A ~]#　nmtui

　编辑连接 -->eth0-->编辑ipv4-->显示-->将自动修改手动

　-->按空格　勾选　自动连接 　 #每次开机自动生效

　-->按空格　勾选　对所有用户有效

激活配置

[root@A ~]#　nmtui

　启用连接 -->eth0-->敲回车　禁用-->敲回车　激活

　最终eth0前面有一个＊代表激活状态

###############################################

创建文档

• mkdir — Make Directory

– 格式:mkdir [/路径/]目录名...

[root@nsd1903 ~]# mkdir nsd01 　　 #当前创建目录

[root@nsd1903 ~]# pwd

/root

[root@nsd1903 ~]# ls

[root@nsd1903 ~]# mkdir /opt/nsd02　＃在指定路径下创建

[root@nsd1903 ~]# ls /opt/

• touch命令

– 用途:新建空文件　格式:touch 文件名...

[root@nsd1903 ~]# touch 1.txt　　　　#当前创建文件

[root@nsd1903 ~]# ls

[root@nsd1903 ~]# touch /opt/2.txt　＃在指定路径下创建

[root@nsd1903 ~]# ls /opt/

文本内容操作

• less分屏阅读工具

• 格式:less [选项] 文件名...

– 优势:支持前后翻页

• 基本交互操作

– 按 / 键向后查找(n、N切换)

[root@nsd1903 ~]# less /etc/passwd

　　按键盘　上　下　可以进行滚动

　　/a 　　#全文查找ａ　(n、N切换)

　　按ｑ退出

• head、tail 命令

– 格式:　head -n 数字 文件名

　　　　tail -n 数字 文件名

[root@nsd1903 ~]# head -2 /etc/passwd #显示头两行内容

[root@nsd1903 ~]# head -3 /etc/passwd

[root@nsd1903 ~]# tail -2 /etc/passwd　＃显示尾两行内容

[root@nsd1903 ~]# tail -3 /etc/passwd

• grep工具

– 用途:输出包含指定字符串的行

– 格式:grep 　'查找条件' 　　目标文件

[root@nsd1903 ~]# grep　 root 　/etc/passwd

[root@nsd1903 ~]# grep 　a　 /etc/passwd

#############################################

关机及重启操作

• 关机:poweroff

[root@svr7 ~]# poweroff

• 重启:reboot

[root@svr7 ~]# reboot

#############################################

编辑命令或路径的技巧：　Tab可以进行补全

[root@nsd1903 ~]# if(tab)(tab)

if ifdown ifup

ifcfg ifenslave

ifconfig ifstat

[root@nsd1903 ~]# ifco(tab)

[root@nsd1903 ~]# cat /et(tab)/red(tab)

CentOS Linux release 7.5.1804 (Core)

]# ls /etc/sysconfig/network-scripts/ifcfg-eth0

]# ls /et(tab)/sysco(tab)/netw(tab)-(tab)/ifc(tab)-e(tab)

day 03

如何编写命令行

• Linux命令

– 用来实现某一类功能的指令或程序

　　在Linux中执行大多数命令，都要找到相应的程序，将程序运行

– 命令的执行依赖于解释器(默认的解释器：/bin/bash)

　绿色：可以执行的程序

　内核：可以调配计算机硬件资源

　用户的指令------＞解释器------＞内核------＞调配计算机硬件

################################################

命令行的一般格式

• 基本用法

– 命令字 [选项]... [参数1] [参数2]...

– 短选项:-l、-A、-c、-d、.. ..

– 多个短选项-->复合选项:-lh、-lA、-ld、.. ..

– 长选项:--help、.. ..

查看命令的帮助信息

　　格式： 命令　 --help

[root@nsd1903 ~]# cat --help

[root@nsd1903 ~]# cat -n /etc/passwd #添加行号

[root@nsd1903 ~]# cat -n /etc/redhat-release

[root@nsd1903 ~]# head -2 /etc/passwd

[root@nsd1903 ~]# tail -3 /etc/passwd

#################################################

快速编辑技巧

• Tab键自动补全

– 可补齐命令字、选项、参数、文件路径、软件名、服务名

• 快捷键

– Ctrl + l:清空整个屏幕

– Ctrl + c:废弃当前编辑的命令行（结束正在运行的命令）

– Esc + . 或 Alt + .:粘贴上一个命令的参数

– Ctrl + u:清空至行首

– Ctrl + k:清空至行尾

– Ctrl + w:往回删除一个单词(以空格界定)

[root@nsd1903 ~]# ls /etc/redhat-release

/etc/redhat-release

[root@nsd1903 ~]# cat 　Alt + .

[root@nsd1903 ~]# cat -n 　Alt + .

[root@nsd1903 ~]# head -1　 Alt + .

　青色：快捷方式

#################################################

mount挂载操作:　让目录成为设备的访问点

安装软件，前提具备软件的安装包

１.互联网下载

２.从光盘中获得软件安装包

#############################################

Windows读取光盘的内容

光盘-----＞光驱设备-----＞CD驱动器图标(访问点)

如何让Linux读取光盘的内容

光盘-----＞光驱设备-----＞访问点(自己创建目录)　　/dvd

/dev/hdc

/dev/sr0

访问点就是一个目录

1.图形利用鼠标，将光盘放入，虚拟光驱设备

2.命令行查看光驱设备

[root@nsd1903 ~]# ls /dev/sr0

/dev/sr0

[root@nsd1903 ~]# ls /dev/cdrom

/dev/cdrom

[root@nsd1903 ~]# ls -l /dev/cdrom #显示详细信息

lrwxrwxrwx. 1 root root 3 4月 3 10:58 /dev/cdrom -> sr0

３.创建访问点

[root@nsd1903 ~]# mkdir /dvd

[root@nsd1903 ~]# ls /

[root@nsd1903 ~]# ls /dvd

• 使用mount命令

– 格式:mount 　设备路径　　　 挂载点目录

[root@nsd1903 ~]# mount　 /dev/cdrom 　/dvd/

[root@nsd1903 ~]# ls /dvd

[root@nsd1903 ~]# ls /dvd/Packages/

　　煤矿------＞洞口

• 使用umount命令

– 格式:umount 挂载点目录

[root@nsd1903 ~]#　umount /dvd

[root@nsd1903 ~]# ls /dvd

[root@nsd1903 ~]#　mkdir /abc

[root@nsd1903 ~]#　mount /dev/cdrom /abc

[root@nsd1903 ~]# ls /abc

注意：

　１.当前路径不要在挂载点目录内

[root@nsd1903 dvd]# umount /dvd/

umount: /dvd：目标忙。

(有些情况下通过 lsof(8) 或 fuser(1) 可以

找到有关使用该设备的进程的有用信息)

2.允许一个设备，具有多个不同的挂载点

3.一个挂载点，不允许挂载多个设备

４.挂载点的选择，自己创建的目录

#################################################

目录和文件管理

• pwd — Print Working Directory

– 用途:查看当前工作目录

• cd — Change Directory

– 用途:切换工作目录

– 格式:cd [目标文件夹位置]

　　/root：Linux中管理员的家目录

　　/home:存放所有普通用户家目录

~:用户家目录

~user表示用户user的家目录

[root@nsd1903 /]# useradd tom #创建用户tom

[root@nsd1903 /]# ls /home/

[root@nsd1903 /]# useradd dc #创建用户dc

[root@nsd1903 /]# ls /home/

[root@nsd1903 /]# cd ~dc #切换到普通用户dc的家目录

[root@nsd1903 dc]# pwd

[root@nsd1903 dc]# cd ~tom

[root@nsd1903 tom]# pwd

. 表示当前目录,经常与复制命令连用

.. 表示父目录

################################################

ls 列出文档及属性

• ls — List

– 格式:ls [选项]... [目录或文件名]

• 常用命令选项

– -l:以长格式显示,显示详细属性

– -A:包括名称以 . 开头的隐藏文档

– -d:显示目录本身(而不是内容)的属性

– -h:提供易读的容量单位(K、M等)

]# ls -l /etc/redhat-release

]# ls -l /etc/passwd

]# ls -l /etc/fstab

]# ls -l /etc/sysconfig/network-scripts/ifcfg-eth0

]# ls -l /root/　　　　＃显示目录内容的详细属性

]# ls -l /

]# ls -l /etc/passwd

]# ls -l -h /etc/passwd

]# ls -lh /etc/passwd　　　#显示详细属性时，加上大小单位

]# ls -lh /var/log/messages

]# ls -ld /root #显示目录本身的详细属性

]# ls -ld /

]# ls -A /root　　　　　＃显示目录的全部内容，包括隐藏数据

]# touch /opt/.nsd01.txt

]# ls /opt/

]# ls -A /opt/

#################################################

使用通配符

• 针对不确定的文档名称,以特殊字符表示

– \*:任意多个任意字符

– ?:单个字符

[root@nsd1903 ~]# ls /root/a\* #显示/root下以a开头

/root/anaconda-ks.cfg

[root@nsd1903 ~]# ls /boot/vm\*

[root@nsd1903 ~]# ls /dev/tty\*　　#显示/dev下以tty开头

[root@nsd1903 ~]# ls /etc/\*tab 　#显示/etc下以tab结尾

[root@nsd1903 ~]# ls /etc/\*.conf

显示以r开头，以.conf结尾

[root@nsd1903 ~]# ls /etc/r\*.conf

[root@nsd1903 ~]# ls /dev/tty\*

[root@nsd1903 ~]# ls /dev/tty?

[root@nsd1903 ~]# ls /dev/tty??

[root@nsd1903 ~]# ls /dev/tty???

• 针对不确定的文档名称,以特殊字符表示

– [a-z]:多个字符或连续范围中的一个,若无则忽略

– {a,min,xy}:多组不同的字符串,全匹配

[root@nsd1903 ~]# ls /dev/tty[1-8]

[root@nsd1903 ~]# ls /dev/tty[3-6]

[root@nsd1903 ~]# ls /dev/tty{1,3,5,7,9,21}

[root@nsd1903 ~]# ls /dev/tty{38,S0}

##############################################

别名的定义:简化复杂的命令

• 查看已设置的别名

– alias [别名名称]

• 定义新的别名

– alias 别名名称= '实际执行的命令行'

• 取消已设置的别名

– unalias [别名名称]

[root@nsd1903 ~]# hostname

nsd1903.tedu.cn

[root@nsd1903 ~]# hn

bash: hn: 未找到命令...

[root@nsd1903 ~]# alias hn='hostname'　　＃定义别名

[root@nsd1903 ~]# hn

nsd1903.tedu.cn

[root@nsd1903 ~]# alias 　　 #显示所有有效别名

[root@nsd1903 ~]# unalias hn　　　＃删除hn别名

[root@nsd1903 ~]# hn

bash: hn: 未找到命令...

[root@nsd1903 ~]#

###############################################

mkdir 创建目录

• mkdir — Make Directory

– 格式:mkdir [-p] [/路径/]目录名...

-p:创建父目录

[root@server0 ~]# mkdir -p /vod/movie/cartoon

[root@server0 ~]#　ls /vod

[root@server0 ~]#　ls /vod/movie

[root@nsd1903 ~]# mkdir -p /opt/aa/bb/cc/dd

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# ls /opt/aa

[root@nsd1903 ~]# ls /opt/aa/bb/

[root@nsd1903 ~]# ls /opt/aa/bb/cc/

rm 删除

• rm — Remove

– 格式:rm [选项]... 文件或目录...

• 常用命令选项

– -r、-f:递归删除(含目录)、强制删除

-r:递归,目录本身以及目录下所有

[root@nsd1903 ~]# mkdir /opt/nsd01

[root@nsd1903 ~]# mkdir /opt/nsd02

[root@nsd1903 ~]# touch /opt/1.txt

[root@nsd1903 ~]# ls /opt/

1.txt nsd01 nsd02

[root@nsd1903 ~]# rm -rf /opt/1.txt

[root@nsd1903 ~]# ls /opt/

nsd01 nsd02

[root@nsd1903 ~]# rm -rf /opt/\*

[root@nsd1903 ~]# ls /opt/

################################################

mv 移动/改名

• mv — Move

– 格式:mv 原文件 目标路径

[root@nsd1903 ~]# rm -rf /opt/\*

[root@nsd1903 ~]# mkdir /opt/nsd

[root@nsd1903 ~]# touch /opt/1.txt

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# mv /opt/1.txt /opt/nsd

[root@nsd1903 ~]# ls /opt/

nsd

[root@nsd1903 ~]# ls /opt/nsd

重命名：路径不变的移动

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# mv /opt/nsd/ /opt/abc

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# mv /opt/abc/ /opt/student

[root@nsd1903 ~]# ls /opt/

#################################################

cp 复制

• cp — Copy

– 格式:cp [选项]... 原文件 目标路径

• 常用命令选项

– -r:递归,复制目录时必须有此选项

[root@nsd1903 ~]# rm -rf /opt/\*

[root@nsd1903 ~]# cp /etc/passwd /opt/

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# cp /etc/fstab /opt/

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# cp -r /home/ /opt/

[root@nsd1903 ~]# ls /opt/

[root@nsd1903 ~]# cp -r /boot/ /opt/

[root@nsd1903 ~]# ls /opt/

复制时出现同名进行覆盖

在本次命令执行，临时取消别名的操作

[root@nsd1903 ~]# cp -r /boot/ /opt/

cp：是否覆盖"/opt/boot/grub2/device.map"？

按 Ctrl + c　取消操作

[root@nsd1903 ~]# \cp -r /boot/ /opt/

[root@nsd1903 ~]# \cp -r /boot/ /opt/

复制可以支持两个以上的参数，永远会把最后一个参数作为目标，其它的所有参数都会作为源

]# cp -r /root /etc/shadow /etc/hosts /etc/login.defs /opt/

]# ls /opt/

复制可以与　．　连用：将源数据，复制到当前路径下

]# cd /opt/

]# pwd

]# cp /etc/resolv.conf .

]# ls

]# cd /etc/sysconfig/network-scripts/

]# pwd

]# cp /etc/passwd .

]# ls

复制时，可以重新命名目标路径下的名称

[root@nsd1903 /]# rm -rf /opt/\*

[root@nsd1903 /]# cp /etc/passwd /opt/

[root@nsd1903 /]# ls /opt/

[root@nsd1903 /]# cp /etc/passwd /opt/nsd

[root@nsd1903 /]# ls /opt/

[root@nsd1903 /]# cp -r /boot/ /opt/abc

[root@nsd1903 /]# ls /opt/

请描述下列，两次复制的不同：

[root@nsd1903 /]# rm -rf /opt/\*

[root@nsd1903 /]# cp -r /home/ 　/opt/test

[root@nsd1903 /]# cp -r /home/　 /opt/test

　第一次复制，将/home复制到/opt目录下改名为test

　第二次复制，将/home复制到/opt/test目录下

##############################################

Linux文件的最后必须没有 /

##############################################

day04

使用vim创建/修改文件内容

三模式：命令模式　　　插入模式（输入模式）　　末行模式

[root@nsd1903 ~]# vim /opt/xiha.txt

命--------i 键---＞插入模式(Esc 键回到命令模式)

令

模

式--------: 键---＞ 末行模式 (Esc 键回到命令模式)

末行模式 :wq 　保存并退出

　　　　　　:q! 　强制不保存并退出

查找文本内容

• 根据字符串模式提取文本行

– grep [选项] '匹配模式' 文本文件...

• 常用命令选项

– -v,取反匹配

– -i,忽略大小写

[root@nsd1903 ~]# grep root /etc/passwd

[root@nsd1903 ~]# grep Root /etc/passwd

[root@nsd1903 ~]# grep -i Root /etc/passwd

]# grep -v root /etc/passwd　　#不包含root的行

• 常用的匹配模式

– word　 包含字符串word

– ^word 　以字符串word开头

– word$ 　以字符串word结尾

[root@nsd1903 ~]# grep root /etc/passwd

[root@nsd1903 ~]# grep ^root /etc/passwd

[root@nsd1903 ~]# grep root$ /etc/passwd

[root@nsd1903 ~]# grep bash$ /etc/passwd

##############################################

重定向输出操作：将前面命令的输出，写入到后面文本文件中

　　>：覆盖重定向

　　＞>：追加重定向

[root@nsd1903 ~]# ls --help

[root@nsd1903 ~]# ls --help > /opt/ls.txt

[root@nsd1903 ~]# less /opt/ls.txt

[root@nsd1903 ~]# hostname

[root@nsd1903 ~]# hostname > /opt/ls.txt

[root@nsd1903 ~]# cat /opt/ls.txt

[root@nsd1903 ~]# hostname >> /opt/ls.txt

[root@nsd1903 ~]# cat /opt/ls.txt

]# cat /etc/redhat-release >> /opt/ls.txt

]# cat /opt/ls.txt

[root@nsd1903 ~]# echo 123456

[root@nsd1903 ~]# echo 123456 > /opt/ls.txt

[root@nsd1903 ~]# cat /opt/ls.txt

[root@nsd1903 ~]# echo NSD1903 >> /opt/ls.txt

[root@nsd1903 ~]# cat /opt/ls.txt

###############################################

管道操作:连接或桥梁

| :将前面命令的输出，传递给后面命令，当做后面命令的参数

　/etc/passwd文件的８到12行内容

　]# head -12 /etc/passwd |　 tail -5

　]# head -12 /etc/passwd | tail -5 | cat -n

　]# cat -n /etc/passwd　 |　 head -12　 | 　tail -5

　]# ifconfig | head -2

　]# ifconfig | less

################################################

管理用户和组

用户： 1.登录操作系统　　２.方便对不同用户进行访问控制

　　组:方便管理用户

　　用户唯一标识： UID

　　组唯一标识： GID

　Linux管理员root的UID永远为0

Linux一个用户至少属于一个组

　　基本组：与用户同名，有系统创建加入完成

　　附加组（从属组）：管理员创建加入完成

添加用户

用户基本信息存放在 /etc/passwd 文件（系统级配置文件）

　root:x:0:0:root:/root:/bin/bash

用户名:密码占位符:UID:基本组的GID:用户描述信息:家目录:解释器

• 使用 useradd 命令

– useradd [选项]... 用户名

• 常用命令选项

– -u 用户id、-d 家目录路径、-s 登录解释器、-G 附加组

[root@nsd1903 ~]# useradd nsd01

[root@nsd1903 ~]# ls /home/

[root@nsd1903 ~]# id nsd01 #查看用户基本信息

[root@nsd1903 ~]# id haxi

id: haxi: no such user

[root@nsd1903 ~]# grep nsd01 /etc/passwd #查看用户信息

[root@nsd1903 ~]# useradd nsd02

[root@nsd1903 ~]# grep nsd /etc/passwd

[root@nsd1903 ~]# useradd -u 1600 nsd03 ＃指定UID创建

[root@nsd1903 ~]# grep nsd /etc/passwd

[root@nsd1903 ~]# useradd nsd04

[root@nsd1903 ~]# grep nsd /etc/passwd

]# useradd -d /mnt/abc nsd06　　＃指定用户的家目录创建

]# grep nsd06 /etc/passwd

-s 登录解释器

　/sbin/nologin：禁止用户登录系统的解释器

[root@nsd1903 ~]# useradd -s /sbin/nologin nsd07

[root@nsd1903 ~]# grep nsd07 /etc/passwd

-G 附加组

[root@nsd1903 ~]# groupadd tarena　　　　＃创建组

[root@nsd1903 ~]# useradd -G tarena nsd08

[root@nsd1903 ~]# id nsd08

uid=1605(nsd08) gid=1606(nsd08) 组=1606(nsd08),1605(tarena)

#############################################

设置登录密码

• 使用 passwd 命令

– passwd [用户名]

[root@nsd1903 ~]# useradd nsd10

[root@nsd1903 ~]# passwd nsd10

更改用户 nsd10 的密码 。

新的 密码：　　　　　　　　　　＃设置新的密码

无效的密码： 密码少于 8 个字符

重新输入新的 密码：　　　　　　　＃重新输入新的密码

passwd：所有的身份验证令牌已经成功更新。

[root@nsd1903 ~]# su - nsd10 #命令行临时切换身份

[nsd10@nsd1903 ~]$ passwd

更改用户 nsd10 的密码 。

为 nsd10 更改 STRESS 密码。

（当前）UNIX 密码：　　　　　　　　＃输入旧密码

新的 密码：　　　　　　　　　　　　＃设置新的密码

重新输入新的 密码：　　　　　　　　＃重新输入新的密码

passwd：所有的身份验证令牌已经成功更新。

[nsd10@nsd1903 ~]$ exit

非交互式设置密码

– echo '密码'　　 | 　　passwd 　--stdin　 用户名

[root@nsd1903 ~]# echo 123 | passwd --stdin nsd10

更改用户 nsd10 的密码 。

passwd：所有的身份验证令牌已经成功更新。

[root@nsd1903 ~]# echo redhat | passwd --stdin nsd10

更改用户 nsd10 的密码 。

passwd：所有的身份验证令牌已经成功更新。

###############################################

用户密码信息存放在 /etc/shadow 文件

[root@nsd1903 ~]# tail -1 /etc/shadow

nsd10:$6$pOWNdlhE$z9/irH93UkWigQr0FQgPC.3NmZGfVMtr6oym4

jWMWFCKRGeq4qYOysgXCR.1q9ztNfMJ91HJxGG34

V3Jq3b0n/:17990:０:99999:7:::

用户名:密码加密字符串:上一次修改密码的时间

#################################################

修改用户属性

• 使用 usermod 命令

– usermod [选项]... 用户名

• 常用命令选项

– -u 用户id、-d 家目录路径、-s 登录解释器

– -G 附加组

[root@nsd1903 ~]# useradd nsd11

[root@nsd1903 ~]# grep nsd11 /etc/passwd

[root@nsd1903 ~]# id nsd11

]# usermod -u 1700 -d /opt/nsd11 -G tarena -s /sbin/nologin nsd11

]# id nsd11

]# grep nsd11 /etc/passwd

#################################################

删除用户

• 使用 userdel 命令

– userdel [-r] 用户名

-r:连同家目录一并删除

[root@server0 ~]# userdel -r alex

[root@server0 ~]# id alex

id: alex: no such user

###############################################

管理组账号

添加组

组基本信息存放在 /etc/group 文件

[root@nsd1903 ~]# grep stugrp /etc/group

stugrp:x:1610:

组名:组的密码占位符:组的GID编号:组的成员列表

• 使用 groupadd 命令

– groupadd [-g 组ID] 组名

[root@nsd1903 ~]# groupadd stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

stugrp:x:1610:

[root@nsd1903 ~]# useradd kenji

[root@nsd1903 ~]# useradd jack

[root@nsd1903 ~]# useradd natasha

[root@nsd1903 ~]# useradd kaka

管理组成员

组成员信息存放在 /etc/gshadow 文件

• 使用 gpasswd 命令

– gpasswd -a 用户名 　组名

– gpasswd -d 用户名　 组名

[root@nsd1903 ~]# gpasswd -a kenji stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

[root@nsd1903 ~]# gpasswd -a natasha stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

[root@nsd1903 ~]# gpasswd -a kaka stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

[root@nsd1903 ~]# gpasswd -d kaka stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

[root@nsd1903 ~]# gpasswd -a jack stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

##################################################

删除组

• 使用 groupdel 命令

– groupdel 组名

[root@nsd1903 ~]# groupdel stugrp

[root@nsd1903 ~]# grep stugrp /etc/group

#################################################

tar备份与恢复

Linux压缩格式：

-z gzip ----＞　.gz

　　　-j bzip2 ----＞ .bz2

-J xz ----＞ .xz

归档和压缩

• 归档的含义

– 将许多零散的文件整理为一个文件

– 文件总的大小基本不变

• 压缩的含义

– 按某种算法减小文件所占用空间的大小

– 恢复时按对应的逆向算法解压

• tar 集成备份工具

– -c:创建tar包

– -x:释放tar包

– -f:指定tar包文件名称

– -z、-j、-J:调用 .gz、.bz2、.xz 格式的工具进行处理

– -t:显示tar包的文件清单

– -C:指定释放路径

tar制作压缩包（tar包）

　格式: tar 　选项　 /路径/压缩包的名字 /路径/源文件 ....

tar 　-zcf　 /路径/压缩包的名字 /路径/源文件 ....

tar 　-jcf　 /路径/压缩包的名字 /路径/源文件 ....

tar 　-Jcf　 /路径/压缩包的名字 /路径/源文件 ....

　　 -z gzip .gz -j bzip2 .bz2 -J xz .xz

红色：压缩

]# rm -rf /opt/\*

]# tar -zcf /opt/file.tar.gz /home/ /etc/passwd

]# ls /opt/

]# tar -tf /opt/file.tar.gz 　 #查看包里面内容

]# tar -tf /opt/file.tar.gz ｜　less

]# tar -jcf /opt/nsd.tar.bz2 /home/ /etc/passwd

]# tar -tf /opt/nsd.tar.bz2

]# ls /opt/

]# tar -Jcf /opt/abc.tar.xz /home/ /etc/passwd

]# tar -tf /opt/abc.tar.xz

]# ls /opt/

]# mkdir /nsd1903

]# tar -xf　 /opt/abc.tar.xz -C 　/nsd1903/

]# ls 　/nsd1903/

]# ls 　/nsd1903/home/

]# ls 　/nsd1903/etc/

#################################################

案例2:创建一个备份包

使用 tar 工具完成以下备份任务:

– 创建一个名为 /root/backup.tar.bz2 的归档文件

– 其中包含 /usr/local 目录中的内容

– tar 归档必须使用 bzip2 进行压缩

]# tar -jcf /root/backup.tar.bz2 /usr/local/

]# ls /root/

]# tar -tf /root/backup.tar.bz2 #查看包的内容

###############################################

day05

教学环境介绍

• 每个学员机上有三台预先配置好的虚拟机

– server —— 作为练习用服务器

– desktop —— 作为练习用客户机

– classroom —— 提供网关/DNS/软件素材等资源

　开机时，优先开启classroom虚拟机

使用rht-vmctl辅助工具

• 控制教学用虚拟机

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

]# rht-vmctl reset classroom　　//先重置资源服务器

]# rht-vmctl reset server

]# rht-vmctl reset desktop　　 //再重置答题虚拟机

虚拟机server

1.利用root用户进入系统，密码为redhat

2.查看系统版本:RHEL 7.0

3.查看主机名:server0.example.com

4.查看虚拟机eth0的IP地址:172.25.0.11

虚拟机desktop

1.利用root用户进入系统，密码为redhat

　2.查看系统版本:RHEL 7.0

　3.查看主机名:desktop0.example.com

　4.查看虚拟机eth0的IP地址:172.25.0.10

##################################################

利用真机，远程管理

１.真机测试通信

[student@room9pc01 ~]$ ping 172.25.0.11

[student@room9pc01 ~]$ ping 172.25.0.10

2.真机进行远程管理

　　ssh 用户名@对方的IP地址

-X:在远程管理时，开启对方的图形程序

[student@room9pc01 ~]$ ssh root@172.25.0.11

Last login: Mon Apr 8 10:37:39 2019 from 172.25.0.250

[root@server0 ~]# hostname

[root@server0 ~]# ifconfig | head -2

[root@server0 ~]# exit 　　#退出远程管理

补充快捷键: Ctrl + shift + t　开启一个新的终端

################################################

[root@server0 ~]# exit　　　　　＃退出远程管理

登出

Connection to 172.25.0.11 closed.

[student@room9pc01 ~]$ ssh -X root@172.25.0.11

Last login: Mon Apr 8 10:46:47 2019

/usr/bin/xauth: file /root/.Xauthority does not exist

[root@server0 ~]# firefox

[root@server0 ~]#

##################################################

权限和归属

基本权限

基本权限的类别

• 访问方式(权限)

– 读取:允许查看内容-read r

– 写入:允许修改内容-write w

– 可执行:允许运行和切换-execute x

文本文件:

r: cat less head tail

w: vim 修改并且保存 重定向　> >>

x: 写Shell脚本

• 权限适用对象(归属)

– 所有者:拥有此文件/目录的用户-user u

– 所属组:拥有此文件/目录的组-group g

– 其他用户:除所有者、所属组以外的用户-other o

　　１.txt lisi lisi

################################################

查看权限

• 使用 ls -l 命令

– ls -ld 文件或目录...

以－开头　代表为文本文件

以ｄ开头　代表为目录

以 l开头　代表为快捷方式

[root@server0 ~]# ls -ld /etc/

[root@server0 ~]# ls -l /etc/passwd

[root@server0 ~]# ls -ld /home/student

[root@server0 ~]# ls -l /etc/shadow

[root@server0 ~]# ls -ld /tmp

##########################################

设置基本权限

• 使用 chmod 命令

– chmod 归属关系+-=权限类别 　文档...

[-R]:递归，目录本身及目录下所有

[root@server0 ~]# mkdir /nsd01

[root@server0 ~]# ls -ld /nsd01

[root@server0 ~]# chmod u-w /nsd01

[root@server0 ~]# ls -ld /nsd01

[root@server0 ~]# chmod g+w /nsd01

[root@server0 ~]# ls -ld /nsd01

[root@server0 ~]# chmod u=rwx /nsd01

[root@server0 ~]# ls -ld /nsd01

[root@server0 ~]# chmod o+w /nsd01

[root@server0 ~]# ls -ld /nsd01

[root@server0 ~]# chmod u=rwx,g=rx,o=--- /nsd01

[root@server0 ~]# ls -ld /nsd01

[root@server0 ~]# mkdir -p /opt/aa/bb/cc

[root@server0 ~]# ls /opt/

[root@server0 ~]# chmod -R o=--- /opt/aa

[root@server0 ~]# ls -ld /opt/aa

[root@server0 ~]# ls -ld /opt/aa/bb/

[root@server0 ~]# ls -ld /opt/aa/bb/cc/

###############################################

Linux如何判断一个用户的具备的权限：　　　匹配及停止

　１.查看用户对于该文档，所处的角色 所有者>所属组>其他人

　２.查看相应角色位置的权限

###############################################

Permission denied：权限不足

目录的 r 权限:能够 ls 浏览此目录内容

目录的 w 权限:能够执行 rm/mv/cp/mkdir/touch/等更改目录内容的操作

目录的 x 权限:能够 cd 切换到此目录

#################################################

以root用户新建/nsddir目录，在此目录下新建readme.txt文件

并进一步完成下列操作

1）使用户lisi能够在此目录下创建子目录 切换用户 su - lisi

chmod o+w /nsddir/

2）使用户lisi不能够在此目录下创建子目录

chmod o-w /nsddir/

3）使用户lisi能够修改readme.txt文件内容

chmod o+w /nsddir/readme.txt

4）调整此目录的权限，使所有用户都不能cd进入此目录

chmod u-x,g-x,o-x /nsddir/

5）为此目录及其下所有文档设置权限 rwxr-x---

chmod -R u=rwx,g=rx,o=--- /nsddir/

　　　-R：递归设置权限，目录下及目录下所有

###############################################

设置文档归属

• 使用 chown 命令

– chown [-R] 属主 文档...

– chown [-R] :属组 文档...

– chown [-R] 属主:属组 文档...

[root@server0 ~]# mkdir /nsd03

[root@server0 ~]# ls -ld /nsd03

[root@server0 ~]# groupadd tarena

[root@server0 ~]# chown lisi:tarena /nsd03

[root@server0 ~]# ls -ld /nsd03

[root@server0 ~]# chown student /nsd03

[root@server0 ~]# ls -ld /nsd03

[root@server0 ~]# chown :root /nsd03

[root@server0 ~]# ls -ld /nsd03

################################################

利用root用户新建/nsd06目录，并进一步完成下列操作

1）将属主设为gelin01，属组设为tarena组

[root@server0 /]# useradd gelin01

[root@server0 /]# useradd gelin02

[root@server0 /]# groupadd tarena

[root@server0 /]# chown gelin01:tarena /nsd06

2）使用户gelin01对此目录具有rwx权限

除属主与属组之外的人，对此目录无任何权限

[root@server0 /]# chmod o=--- /nsd06

3）使用户gelin02能进入、查看此目录内容

[root@server0 /]# gpasswd -a gelin02 tarena

4）将gelin01加入tarena组, 将nsd06目录的权限设为rw-r-x---

再测试gelin01用户能否进入此目录

[root@server0 /]# gpasswd -a gelin01 tarena

[root@server0 /]# chmod u=rw,g=rx /nsd06

################################################

实现lisi用户可以读取/etc/shadow文件内容，您有几种办法?

1.修改其他人权限

chmod o+r /etc/shadow

2.利用所属组

chown :lisi /etc/shadow

chmod g+r /etc/shadow

3.利用所有者

chown lisi /etc/shadow

chmod u+r /etc/shadow

4.利用ACL策略

　　　　　setfacl -m u:lisi:r /etc/shadow

###############################################

附加权限（特殊权限）

Set GID

• 附加在属组的 x 位上

– 属组的权限标识会变为 s

– 适用于目录,Set GID可以使目录下新增的文档自动设

置与父目录相同的属组

– 继承所属组身份

[root@server0 ~]# mkdir /nsd09

[root@server0 ~]# ls -ld /nsd09

[root@server0 ~]# chown :tarena /nsd09

[root@server0 ~]# ls -ld /nsd09

[root@server0 ~]# mkdir /nsd09/test01

[root@server0 ~]# ls -ld /nsd09/test01

[root@server0 ~]# chmod g+s /nsd09/

[root@server0 ~]# ls -ld /nsd09/

[root@server0 ~]# mkdir /nsd09/test02

[root@server0 ~]# ls -ld /nsd09/test02

[root@server0 ~]# mkdir /nsd09/test02/abc

[root@server0 ~]# ls -ld /nsd09/test02/abc

################################################

Set UID

• 附加在属主的 x 位上

– 属主的权限标识会变为 s

– 适用于可执行文件,Set UID可以让使用者具有文件属

主的身份及部分权限

[root@server0 ~]# cp /usr/bin/mkdir /usr/bin/hahadir

[root@server0 ~]# chmod u+s /usr/bin/hahadir

[root@server0 ~]# ls -l /usr/bin/hahadir

[root@server0 ~]# ls -l /usr/bin/mkdir

[root@server0 ~]# su - dc

[dc@server0 ~]$ /usr/bin/mkdir dc01

[dc@server0 ~]$ /usr/bin/hahadir dc02

[dc@server0 ~]$ exit

###############################################

Sticky Bit

• 附加在其他人的 x 位上

– 其他人的权限标识会变为 t

– 适用于开放 w 权限的目录,可以阻止用户滥用 w 写入

权限(禁止操作别人的文档)

]# mkdir /home/public/

]# chmod ugo=rwx /home/public/

]# chmod o+t /home/public/

#############################################

acl访问控制列表(ACL策略)

acl策略的作用

• 文档归属的局限性

– 任何人只属于三种角色:属主、属组、其他人

– 无法实现更精细的控制

• acl访问策略

– 能够对个别用户、个别组设置独立的权限

– 大多数挂载的EXT3/4、XFS文件系统默认已支持

设置acl访问控制策略

• 使用 getfacl、setfacl 命令

– getfacl 文档...

– setfacl -m u:用户名:权限类别 文档...

– setfacl -m g:组名:权限类别 文档...

– setfacl -b 文档... #删除所有ACL策略

– setfacl -m u:用户名 文档... #删除单条指定ACL策略

　　　[-R]：递归设置ACL策略

day06

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

虚拟机server

1.利用root用户进入系统，密码为redhat

2.查看系统版本:RHEL 7.0

3.查看主机名:server0.example.com

4.查看虚拟机eth0的IP地址:172.25.0.11

虚拟机desktop

1.利用root用户进入系统，密码为redhat

　2.查看系统版本:RHEL 7.0

　3.查看主机名:desktop0.example.com

　4.查看虚拟机eth0的IP地址:172.25.0.10

##################################################

软件包管理

零散软件管理

1.首先需要具备软件的安装包

2.构建Web服务或FTP服务，共享光盘所有内容

　　　Web服务:提供网页内容的服务

3.将虚拟机classroom已经构建完成Web服务，已经共享光盘所有内容4.真机浏览器，访问测试

　　　 classroom.example.com

content/rhel7.0/x86\_64/dvd/Packages/

5.下载软件包

• 使用wget下载工具

– wget 软件包的URL网址　　＃默认下载到当前目录下

– wget 软件包的URL网址 　-O(大写) 　/目录路径/新文件名

提供FTP功能的软件包(vsftpd)

当前页面查找:vsftpd

[root@server0 ~]# wget http://classroom.example.com/content/rhel7.0/x86\_64/dvd/Packages/vsftpd-3.0.2-9.el7.x86\_64.rpm

[root@server0 ~]# ls

• RPM Package Manager,RPM包管理器

– rpm -q 软件名 　　　 #查询该软件是否已经安装

– rpm -ivh 软件名-版本信息.rpm...　　#安装软件包

– rpm -e 软件名...

[root@server0 ~]# rpm -q firefox

firefox-24.5.0-1.el7.x86\_64

[root@server0 ~]# rpm -q haha

未安装软件包 haha

[root@server0 ~]# rpm -q bash

bash-4.2.45-5.el7.x86\_64

[root@server0 ~]# rpm -q hostname

[root@server0 ~]# rpm -q zip

[root@server0 ~]# rpm -q httpd

]# rpm -ivh /root/vsftpd-3.0.2-9.el7.x86\_64.rpm

]# rpm -q vsftpd　　　#查询是否安装成功

]# rpm -e vsftpd　　　　#卸载软件

]# rpm -q vsftpd　　　#查询是否卸载成功

#################################################

了解:导入新的红帽签名

]# wget http://classroom.example.com/content/rhel7.0/x86\_64/dvd/RPM-GPG-KEY-redhat-release

]# rpm --import /root/RPM-GPG-KEY-redhat-release

]# rpm -q vsftpd

]# rpm -ivh /root/vsftpd-3.0.2-9.el7.x86\_64.rpm

########################################

升级Linux内核

1.下载新的内核软件包

]# wget http://classroom.example.com/content/rhel7.0/x86\_64/errata/Packages/kernel-3.10.0-123.1.2.el7.x86\_64.rpm

]# ls

2.安装新的内核软件包

[root@server0 qw~]# uname -r #查看内核版本

]# rpm -ivh /root/kernel-3.10.0-123.1.2.el7.x86\_64.rpm

[root@server0 ~]# uname -r

[root@server0 ~]# reboot

[student@room9pc01 ~]$ ssh -X root@172.25.0.11

[root@server0 ~]# uname -r

3.10.0-123.1.2.el7.x86\_64

[root@server0 ~]#

################################################

安装软件包常见提示:

1.软件包已经安装

]# rpm -ivh /root/vsftpd-3.0.2-9.el7.x86\_64.rpm

准备中... ################################# [100%]

软件包 vsftpd-3.0.2-9.el7.x86\_64 已经安装

[root@server0 ~]# LANG=en　　　　＃讲当前系统语言修改英文

]# rpm -ivh /root/vsftpd-3.0.2-9.el7.x86\_64.rpm

Preparing... ################################# [100%]

package vsftpd-3.0.2-9.el7.x86\_64 is already installed

[root@server0 ~]#

2.软件包的依赖关系

]# rpm -ivh /root/bind-chroot-9.9.4-14.el7.x86\_64.rpm

错误：依赖检测失败：

bind = 32:9.9.4-14.el7 被 bind-chroot-32:9.9.4-14.el7.x86\_64 需要

################################################

Yum软件包管理

　　自动解决软件包的依赖关系

Yum配置及使用(Yum软件包仓库)

服务：为客户端自动解决依赖关系安装软件包

服务端：虚拟机classroom

　　　1.众多的软件包　　2.仓库数据文件　　３.构建web或ftp服务

　光盘中既具备软件包，又具备仓库数据文件

　　　　　　　软件包 Packages

　　　　　　　仓库数据文件　repodata

　　　总结：构建web或ftp服务，共享光盘所有内容

客户端：指定服务端位置

　　客户端配置文件:/etc/yum.repos.d/\*.repo

　错误的客户端配置文件，会影响正确的客户端配置文件

　　　[] name baseurl enabled gpgcheck

[root@server0 ~]# rm -rf /etc/yum.repos.d/\*

[root@server0 ~]# vim /etc/yum.repos.d/dvd.repo

[nsd1903]　　　　　　　＃仓库标识

name=rhel7　　　　　　＃仓库描述信息

baseurl=http://classroom.example.com/content/rhel7.0/x86\_64/dvd　　　　　　　＃指定服务端在哪里

enabled=1　　　　　　　＃启用本文件

gpgcheck=0　　　　　　　＃不检测软件包的签名

[root@server0 ~]# yum repolist #列出仓库信息

源标识 源名称 状态

nsd1903 rhel7 4,305

repolist: 4,305

[root@server0 ~]#

Yum的使用:

[root@server0 ~]# yum -y install httpd

[root@server0 ~]# rpm -q httpd

httpd-2.4.6-17.el7.x86\_64

]# yum -y install sssd

]# yum -y install gcc

]# yum -y install bind-chroot

]# yum -y install vsftpd

]# yum -y install system-config-kickstart

[root@server0 ~]# yum 　remove　 gcc　 #卸载

[root@server0 ~]# yum clean all　　＃清空缓存

################################################

分区规划及使用

扇区大小默认为512字节

硬盘（块设备）分区管理

一块硬盘的“艺术”之旅

• 识别硬盘 => 分区规划 => 格式化 => 挂载使用

　毛坯楼层 => 打隔断 => 装修 => 入驻

一 识别硬盘

[root@server0 ~]# lsblk 　　　　＃显示当前系统中硬盘

　NAME 　SIZE 　　TYPE

　vda 　10G 　　 disk

　└─vda1 10G 　part

　vdb 　 10G 　　disk

二 分区规划

分区模式：MBR 主引导记录分区模式

　　• MBR/msdos 分区模式

　　– 1~4个主分区　或者　3个主分区+1个扩展分区(n个逻辑分区)

　　– 最大支持容量为 2.2TB 的磁盘

　　– 扩展分区不能格式化

　　/dev/sda5：该设备标识的含义

　　　　　　　　第一块SCSI接口的设备，第五个分区

　　　　　　　　第一块SCSI接口的设备，第一个逻辑分区

[root@server0 ~]# fdisk /dev/vdb

n 创建新的分区----->回车----->回车---->回车----->在last结束时 +1G

p 查看分区表

n 创建新的分区----->回车----->回车---->回车----->在last结束时 +2G

d 删除分区

w 保存并退出

[root@server0 ~]# lsblk

[root@server0 ~]# ls /dev/vdb[1-2]

三　格式化:赋予空间文件系统

　　文件系统：数据在空间存储的规则

　　Windows: FAT NTFS

Linux:

RHEL6:ext4

RHEL7:xfs

• mkfs 工具集

– mkfs.ext3 分区设备路径

– mkfs.ext4 分区设备路径

– mkfs.xfs 分区设备路径

– mkfs.vfat -F 32 分区设备路径

[root@server0 ~]# mkfs.ext4 /dev/vdb1　＃格式化文件系统

[root@server0 ~]# blkid /dev/vdb1 #查看文件系统类型

[root@server0 ~]# mkfs.xfs /dev/vdb2　＃格式化文件系统

[root@server0 ~]# blkid /dev/vdb2　　#查看文件系统类型

四　挂载设备进行使用

[root@server0 ~]# mkdir /mypart1

[root@server0 ~]# mount /dev/vdb1 /mypart1

[root@server0 ~]# df -h　　　#显示正在挂载设备的使用情况

[root@server0 ~]# mount /dev/vdb2 /mypart2

mount: 挂载点 /mypart2 不存在

[root@server0 ~]# mkdir /mypart2

[root@server0 ~]# mount /dev/vdb2 /mypart2

[root@server0 ~]# df -h　　　#显示正在挂载设备的使用情况

###################################################

五　永久挂载（开机自动挂载）

实现开机自动挂载

• 配置文件 /etc/fstab 的记录格式

设备路径 挂载点 类型 参数 备份标记 检测顺序

vim命令模式　按o另起新的一行 进入插入模式

[root@server0 ~]# vim /etc/fstab

/dev/vdb1 /mypart1 ext4 defaults 0 0

/dev/vdb2 /mypart2 xfs defaults 0 0

[root@server0 ~]# umount /mypart1 　#保证当前没有挂载

[root@server0 ~]# umount /mypart2　　#保证当前没有挂载

[root@server0 ~]# df -h

[root@server0 ~]# mount -a

检测/etc/fstab开机自动挂载配置文件,格式是否正确

检测/etc/fstab中,书写完成,但当前没有挂载的设备,进行挂载

[root@server0 ~]# df -h

###################################################

综合分区，/dev/vdb继续分区

最终有3个主分区，分别为1G、2G、2G

创建扩展分区 ---> 两个逻辑分区，分别为1G、1G

[root@server0 ~]# fdisk /dev/vdb

p 查看分区表

n 创建主分区----->回车----->回车---->回车----->在last结束时 +2G

p 查看分区表

n 创建扩展分区

----->回车---->起始回车----->结束回车 将所有空间给扩展分区

p 查看分区表

n 创建逻辑分区----->起始回车------>结束+1G

n 创建逻辑分区----->起始回车------>结束+1G

p 查看分区表

w 保存并退出

[root@server0 ~]# partprobe #刷新分区表

[root@server0 ~]# lsblk

#####################################################

总结：

1.识别硬盘　lsblk

2.分区规划 fdisk

3.刷新分区表　　partprobe

4.进行格式化　　mkfs.xfs mkfs.ext4 blkid

5.挂载使用　mount /etc/fstab开机自动挂载 mount -a df -h

####################################################

day07

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

虚拟机server

1.利用root用户进入系统，密码为redhat

2.查看系统版本:RHEL 7.0

3.查看主机名:server0.example.com

4.查看虚拟机eth0的IP地址:172.25.0.11

虚拟机desktop

1.利用root用户进入系统，密码为redhat

　2.查看系统版本:RHEL 7.0

　3.查看主机名:desktop0.example.com

　4.查看虚拟机eth0的IP地址:172.25.0.10

##################################################

一 添加硬盘

1.虚拟机server进行关机

[root@server0 ~]# poweroff

Connection to 172.25.0.11 closed by remote host.

Connection to 172.25.0.11 closed.

[student@room9pc01 ~]$

2.远程管理虚拟机server

[student@room9pc01 ~]$ ssh -X root@172.25.0.11

[root@server0 ~]# lsblk

二 分区规划

划分可以使用的5个分区,每个分区10G

[root@server0 ~]# fdisk /dev/vdc

p 查看分区表

n 创建主分区----->回车----->回车---->回车----->在last结束时 +10G

....... 连续划分3个主分区

p 查看分区表

n 创建扩展分区

----->回车---->起始回车----->结束回车 将所有空间给扩展分区

p 查看分区表

n 创建逻辑分区----->起始回车------>结束+10G

n 创建逻辑分区----->起始回车------>结束+10G

p 查看分区表

w 保存并退出

[root@server0 ~]# lsblk

LVM逻辑卷

作用: 整合分散的空间 大小可以动态扩大

1.零散空闲存储 (物理卷PV)

2.整合的虚拟磁盘 (卷组VG)

3.虚拟的分区 (逻辑卷LV)

将众多的物理卷(PV),组建成卷组(VG),再从卷组中划分逻辑卷(LV)

####################################################

逻辑卷的制作

successfully(成功)

1.创建卷组名为systemvg

格式: vgcreate 卷组名 设备路径........

[root@server0 ~]# vgcreate systemvg /dev/vdc[1-2]

[root@server0 ~]# pvs #查看物理卷信息

[root@server0 ~]# vgs #查看卷组信息

2.创建逻辑卷

格式: lvcreate -n 逻辑卷名字 -L 逻辑卷大小 卷组名

[root@server0 ~]# lvcreate -n vo -L 16G systemvg

Logical volume "vo" created

[root@server0 ~]# lvs #查看当前系统中,所有逻辑卷信息

[root@server0 ~]# vgs #查看当前系统中,所有卷组信息

###################################################

逻辑卷的使用

1.查看逻辑卷设备文件

[root@server0 ~]# ls /dev/systemvg/vo

[root@server0 ~]# ls -l /dev/systemvg/vo

[root@server0 ~]# ls /dev/dm-0

2.格式化文件系统,进行挂载

[root@server0 ~]# mkfs.xfs /dev/systemvg/vo

[root@server0 ~]# blkid /dev/systemvg/vo

[root@server0 ~]# vim /etc/fstab

/dev/systemvg/vo /mylv xfs defaults 0 0

[root@server0 ~]# mount -a

[root@server0 ~]# df -h

##################################################

逻辑卷的扩展

一 卷组有足够的剩余空间

1.扩展空间的大小

[root@server0 ~]# vgs

[root@server0 ~]# lvextend -L 18G /dev/systemvg/vo

[root@server0 ~]# lvs

2.扩展文件系统的大小

扩展ext4: resize2fs

扩展xfs: xfs\_growfs

[root@server0 ~]# xfs\_growfs /dev/systemvg/vo

[root@server0 ~]# df -h

二 卷组没有足够的剩余空间

1.扩展卷组的空间

[root@server0 ~]# vgextend systemvg /dev/vdc3

[root@server0 ~]# vgs

2.扩展逻辑卷空间的大小

[root@server0 ~]# vgs

[root@server0 ~]# lvextend -L 25G /dev/systemvg/vo

[root@server0 ~]# lvs

2.扩展逻辑卷文件系统的大小

[root@server0 ~]# xfs\_growfs /dev/systemvg/vo

[root@server0 ~]# df -h

###################################################

了解: 逻辑卷可以扩展,也可以缩小

###################################################

卷组划分空间的单位:PE

默认1PE为4M大小

[root@server0 ~]# vgdisplay systemvg #显示卷组详细信息

PE Size 4.00 MiB

请划分一个250M的逻辑卷,命名为lvtest01

[root@server0 ~]# vgchange -s 1M systemvg #修改PE的大小

[root@server0 ~]# vgdisplay systemvg

[root@server0 ~]# lvcreate -n lvtest01 -L 250M systemvg

[root@server0 ~]# lvs

• 创建卷组的时候设置PE大小

– vgcreate -s PE大小 卷组名 空闲分区...

• 创建逻辑卷的时候指定PE个数

– lvcreate -l PE个数 -n 逻辑卷名 卷组名

################################################

逻辑卷的删除

lvremove vgremove pvremove

优先删除逻辑卷,再去删除卷组,最后删除物理卷

[root@server0 ~]# lvremove /dev/systemvg/vo

Logical volume systemvg/vo contains a filesystem in use.

[root@server0 ~]# umount /mylv/

[root@server0 ~]# lvremove /dev/systemvg/vo

Do you really want to remove active logical volume vo? [y/n]: y

Logical volume "vo" successfully removed

[root@server0 ~]# lvs

删除卷组,需要首先删除,基于此卷组创建的所有逻辑卷

命令格式:vgremove 卷组名

####################################################

find高级使用

查找文件

• 根据预设的条件递归查找对应的文件

– find [目录] [条件1] [-a|-o] [条件2] ...

– 常用条件表示:

-type 类型(l快捷方式、d目录、f文本文件)

-name "文档名称"

-size +|-文件大小(k、M、G)

-user 用户名

-mtime 根据文件修改时间

############################################

-type 类型(l快捷方式、d目录、f文本文件)

[root@server0 ~]# find /boot/ -type l

[root@server0 ~]# ls /boot/grub/menu.lst

[root@server0 ~]# ls -l /boot/grub/menu.lst

[root@server0 ~]# find /boot/ -type d

[root@server0 ~]# find /boot/ -type f

[root@server0 ~]# find /root -type d

[root@server0 ~]# find /root -type f

[root@server0 ~]# man find #权威帮助信息

################################################

-name '文档名称'

[root@server0 ~]# find /etc/ -name 'passwd'

[root@server0 ~]# find /etc/ -name 'passwd\*'

[root@server0 ~]# find /etc/ -name '\*passwd\*'

[root@server0 ~]# mkdir /root/nsd01

[root@server0 ~]# mkdir /root/nsd1903

[root@server0 ~]# touch /root/nsd01.txt

[root@server0 ~]# find /root/ -name 'nsd\*'

[root@server0 ~]# find /root/ -name 'nsd\*' -type f

[root@server0 ~]# find /root/ -name 'nsd\*' -type d

####################################################

-size +|-文件大小(k、M、G) 百度 EB单位

[root@server0 ~]# find /boot/ -size +10M

[root@server0 ~]# find /boot/ -size +300k

-user 用户名

[root@server0 ~]# find / -user student

[root@server0 ~]# ls -ld /home/student

[root@server0 ~]# find / -user student -type f

###################################################

-mtime 根据文件修改时间

所有的时间都是过去时间

-mtime +10 #十天之前创建或修改的数据

-mtime -10 #最近十天之内创建或修改的数据

-mtime +90 #三个月之前创建或修改的数据

[root@server0 ~]# find /var/log/ -mtime +90

[root@server0 ~]# find /var/log/ -mtime +1000

[root@server0 ~]# ls -l /var/log/yum.log

[root@server0 ~]# date #查看当前系统时间

##################################################

find扩展使用

• 使用find命令的 -exec 操作

– find .. .. -exec 处理命令 {} \;

– 优势:以 {} 代替每一个find的查找结果,逐个处理,遇 \; 结束

]# find /boot/ -name 'vm\*'

]# find /boot/ -name 'vm\*' -exec cp {} /opt \;

]# ls /opt/

]# find /boot/ -size +10M

]# find /boot/ -size +10M -exec cp {} /opt \;

]# ls /opt/

###################################################

案例4:查找并处理文件

• 使用find命令完成以下任务

– 请创建目录/root/findfiles/

– 找出所有 用户 student 拥有的文件

– 把它们拷贝到 /root/findfiles/ 文件夹中

]# mkdir /root/findfiles

]# find / -user student -type f -exec cp {} /root/findfiles/ \;

]# ls -A /root/findfiles/

#################################################

查看当前系统时间:date

修改时间: date -s '年-月-日 时:分:秒'

[root@server0 ~]# date

[root@server0 ~]# date -s '2008-10-1 10:12:02'

[root@server0 ~]# date

[root@server0 ~]# date -s '2019-4-10 15:43'

[root@server0 ~]# date

################################################

NTP时间同步

NTP网络时间协议

• Network Time Protocol

– NTP服务器为客户机提供标准时间

– NTP客户机需要与NTP服务器保持沟通

NTP服务器:虚拟机classroom

NTP客户机:虚拟机server

1.安装chrony软件包

[root@server0 ~]# yum -y install chrony

软件包 chrony-1.29.1-1.el7.x86\_64 已安装并且是最新版本

无须任何处理

[root@server0 ~]# rpm -q chrony

chrony-1.29.1-1.el7.x86\_64

2.修改配置文件/etc/chrony.conf

[root@server0 ~]# vim /etc/chrony.conf

以#开头的行,为注释行

#server 0.rhel.pool.ntp.org iburst

#server 1.rhel.pool.ntp.org iburst

#server 2.rhel.pool.ntp.org iburst

server classroom.example.com iburst

3.重启服务(程序)

[root@server0 ~]# systemctl restart chronyd

[root@server0 ~]# systemctl enable chronyd #开机自启

daemon

英 [ˈdi:mən] 美 ['di:mən]

守护神;（希腊神话中）半人半神的精灵;[计]守护进程(程序)

4.验证

[root@server0 ~]# date -s '2008-10-1'

2008年 10月 01日 星期三 00:00:00 CST

[root@server0 ~]# date

[root@server0 ~]# systemctl restart chronyd

[root@server0 ~]# date

[root@server0 ~]# date

[root@server0 ~]# date

[root@server0 ~]# date

#################################################

cron计划任务

cron任务概述

• 用途:按照设置的时间间隔为用户反复执行某一项固

定的系统任务

• 软件包:cronie、crontabs

• 系统服务:crond

• 日志文件:/var/log/crond

如何编写crontab任务记录

• 配置格式可参考 /etc/crontab 文件

分 时 日 月 周 任务命令行(绝对路径)

\* \* \* \* \* #每分钟都执行一次

30 8 \* \* \* #每天早晨八点半 执行一次

0 21 \* \* 1-5 #周一至周五 晚上的九点正

0 21 \* \* 1,3,5,7 #周一 周三 周五 周日 晚上的九点正

1 8 1 \* 1 #周一与每月一号都会执行

0 \*/2 \* \* \* #每隔2个小时执行一次

\*:匹配范围内任意时间

,:分隔多个不连续的时间点

-:指定连续时间范围

/n:指定时间频率,每n ...

执行周期 配置说明

分钟 从0到59之间的整数

小时 从0到23之间的整数

日期 从1到31之间的整数

月份 从1到12之间的整数

星期 0~7之间的整数,0或7代表星期日

管理计划任务策略

• 使用 crontab 命令

– 编辑:crontab -e [-u 用户名]

– 查看:crontab -l [-u 用户名]

– 清除:crontab -r [-u 用户名]

每分钟记录当前系统的时间,写入到/opt/time.txt

[root@server0 ~]# date

[root@server0 ~]# date >> /opt/time.txt

[root@server0 ~]# cat /opt/time.txt

[root@server0 ~]# crontab -e -u root

\* \* \* \* \* date >> /opt/time.txt

[root@server0 ~]# crontab -l -u root

[root@server0 ~]# ls /var/spool/cron/

[root@server0 ~]# cat /var/spool/cron/root #生成的任务文件

[root@server0 ~]# cat /opt/time.txt

###############################################

小阶段2

day01

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

##################################################

Shell脚本基础

脚本:一个可以执行的文件,运行可以实现某种功能

创建用户zhangsan -------> useradd zhangsan

1. 编写一个面世问候 /root/hello.sh 脚本

– 显示出一段话 “Hello World”

[root@server0 ~]# vim /root/hello.sh

echo Hello World

[root@server0 ~]# chmod +x /root/hello.sh #赋予执行权限

[root@server0 ~]# /root/hello.sh #以绝对路径运行

规范Shell脚本的一般组成

• #! 环境声明,以下代码由那个程序进行运行

• # 注释文本

• 可执行代码

2. 编写一个能输出系统信息的 /root/sysinfo 脚本

1)输出当前红帽系统的版本信息

2)输出当前使用的内核版本

3)输出当前系统的主机名

[root@server0 ~]# vim /root/sysinfo

#!/bin/bash

cat /etc/redhat-release

uname -r

hostname

ifconfig | head -2

[root@server0 ~]# chmod +x /root/sysinfo

[root@server0 ~]# /root/sysinfo

################################################

重定向输出

>:只收集前面命令的正确输出

2>:只收集前面命令的错误输出

&>:收集前面命令的所有输出

[root@server0 ~]# echo 123 > /opt/1.txt

[root@server0 ~]# cat /opt/1.txt

[root@server0 ~]# cat /a

[root@server0 ~]# cat /opt/1.txt /a

[root@server0 ~]# cat /opt/1.txt /a > /opt/b.txt

[root@server0 ~]# cat /opt/b.txt

[root@server0 ~]# cat /opt/1.txt /a 2> /opt/b.txt

[root@server0 ~]# cat /opt/b.txt

[root@server0 ~]# cat /opt/1.txt /a &> /opt/b.txt

[root@server0 ~]# cat /opt/b.txt

##################################################

补充:

单引号 ' ' :屏蔽所有的特殊符号,原样输出

[root@server0 ~]# echo ' \* ? | . .. & '

$( ) 或 反撇号 ` `:将命令输出结果,直接参与下一次运行

[root@server0 opt]# cd /opt/

[root@server0 opt]# mkdir $(hostname)

[root@server0 opt]# date +%F

[root@server0 opt]# mkdir mydir-$(date +%F)

#################################################

案例:书写一个创建用户,设置密码的脚本user.sh

/dev/null:黑洞设备

[root@server0 /]# vim /root/user.sh

#!/bin/bash

useradd nsd04 &> /dev/null

echo 用户nsd04创建成功

echo 123 | passwd --stdin nsd04 &> /dev/null

echo 用户nsd04密码设置成功

[root@server0 /]# chmod +x /root/user.sh

[root@server0 /]# /root/user.sh

为了让脚本适应多变的环境与多种不同需求,所以使用变量

变量:会变化的量 容器=存放可以变化的量

以不变的名称,存放可以变化的值

变量名=存放的值

• 以不变的名称存放的可能会变化的值

– 变量名=变量值

– 方便以固定名称重复使用某个值

– 提高对任务需求、运行环境变化的适应能力

[root@server0 /]# cat /root/user.sh

#!/bin/bash

a=nsd07 #定义变量a,储存值nsd07

useradd $a &> /dev/null

echo 用户$a创建成功

echo 123 | passwd --stdin $a &> /dev/null

echo 用户$a密码设置成功

定义/赋值变量

• 设置变量时的注意事项

– 等号两边不要有空格

– 变量名只能由字母/数字/下划线组成,区分大小写

– 变量名不能以数字开头,不要使用关键字和特殊字符

– 若指定的变量名已存在,相当于为此变量重新赋值

查看/引用变量

– 引用变量值:$变量名

– 查看变量值:echo $变量名、echo ${变量名}

[root@server0 /]# a=redhat

[root@server0 /]# echo $a

[root@server0 /]# echo $a7

[root@server0 /]# echo ${a}7

变量的种类

环境变量:变量名为大写,变量的值有系统定义完成

USER=存储当前登录系统的用户

[root@server0 /]# echo $USER

root

[root@server0 /]# su - student

[student@server0 ~]$ echo $USER

student

[student@server0 ~]$ exit

logout

[root@server0 /]#

位置变量

• 在执行脚本时提供的命令行参数

– 表示为 $n,n为序号

– $1、$2、.. .. ${10}、${11}、.. ..

[root@server0 /]# vim /root/1.sh

#!/bin/bash

echo $1

echo $2

[root@server0 /]# chmod +x /root/1.sh

[root@server0 /]# /root/1.sh dc tc

[root@server0 /]# vim /root/user.sh

#!/bin/bash

#a=nsd07

useradd $1 &> /dev/null

echo 用户$1创建成功

echo 123 | passwd --stdin $1 &> /dev/null

echo 用户$1密码设置成功

[root@server0 /]# /root/user.sh dc

[root@server0 /]# vim /root/cat.sh

#!/bin/bash

cat -n $1 | head -$2

[root@server0 /]# chmod +x /root/cat.sh

[root@server0 /]# /root/cat.sh /etc/passwd 3

[root@server0 /]# /root/cat.sh /etc/shadow 2

降低脚本的使用难度,可以产生交互

read -p '屏幕提示信息'

1.直接产生交互

2.记录用户在键盘上的输入内容

3.将 用户在键盘上的输入内容,赋值给一个变量储存

[root@server0 /]# cat /root/user.sh

#!/bin/bash

read -p '请输入您要创建的用户名:' a

read -p '请输入您要设置的密码:' b

useradd $a &> /dev/null

echo 用户$a创建成功

echo $b | passwd --stdin $a &> /dev/null

echo 用户$a密码设置成功

[root@server0 /]#

预定义变量

• 用来保存脚本程序的执行信息

– 直接使用这些变量

– 不能直接为这些变量赋值

$# 已加载的位置变量的个数

$? 程序退出后的状态值,0表示正常,其他值异常

[root@server0 /]# vim /root/1.sh

#!/bin/bash

echo $1

echo $2

echo $#

[root@server0 /]# /root/1.sh haha xixi

[root@server0 /]# /root/1.sh haha xixi dc tc

[root@server0 /]# /root/1.sh

##################################################

条件测试

[ 测试表达式 ]

常用的测试选项

• 检查文件状态

-e:文档存在为真

-d:文档存在,且必须为目录才为真

-f:文档存在,且必须为文件才为真

[root@server0 /]# [ -e /etc/passwd ]

[root@server0 /]# echo $?

[root@server0 /]# [ -d /etc/passwd ]

[root@server0 /]# echo $?

[root@server0 /]# [ -d /etc/ ]

[root@server0 /]# echo $?

[root@server0 /]# [ -f /etc/fstab ]

[root@server0 /]# echo $?

• 比较整数大小

-gt:大于

-ge:大于等于

-eq:等于

-ne:不等于

-lt:小于

-le:小于等于

[root@server0 /]# [ 1 -eq 1 ]

[root@server0 /]# echo $?

0

[root@server0 /]# [ 1 -gt 1 ]

[root@server0 /]# echo $?

1

[root@server0 /]# [ 1 -ge 1 ]

[root@server0 /]# echo $?

0

• 字符串比对

==: 两个字符串一样为真

!=: 两个字符串不一样为真

[root@server0 /]# [ haha != student ]

[root@server0 /]# echo $?

0

[root@server0 /]# [ haha == haha ]

[root@server0 /]# echo $?

0

[root@server0 /]# [ $USER == root ]

[root@server0 /]# echo $?

0

##################################################

if选择结构

双分支

if [条件测试];then

命令序列xx

else

命令序列yy

fi

[root@server0 /]# vim /root/if01.sh

#!/bin/bash

if [ $1 -eq 1 ];then

echo hello

else

echo hi

fi

[root@server0 /]# /root/if01.sh 1

[root@server0 /]# /root/if01.sh 2

[root@server0 /]# vim /root/if02.sh

#!/bin/bash

if [ $# -eq 0 ];then

echo 您没有输入参数

else

echo 您一共输入了 $# 个参数

fi

[root@server0 /]# /root/if02.sh dc tc

您一共输入了2个参数

[root@server0 /]# /root/if02.sh

您没有输入参数

##################################################

if多分支处理

if [条件测试1];then

命令序列xx

elif [条件测试2];then

命令序列yy

else

命令序列zz

fi

案例:

利用read 读入用户输入的成绩

如果用户输入的成绩大于等于90,则输出 优秀

如果用户输入的成绩大于等于80,则输出 良好

如果用户输入的成绩大于等于70,则输出 一般

如果用户输入的成绩大于等于60,则输出 合格

如果以上条件均不满足,则输出 '再牛的肖邦,也弹不出哥的悲伤'

[root@server0 /]# vim /root/if03.sh

#!/bin/bash

read -p '请输入您的成绩:' num

if [ $num -ge 90 ];then

echo 优秀

elif [ $num -ge 80 ];then

echo 良好

elif [ $num -ge 70 ];then

echo 一般

elif [ $num -ge 60 ];then

echo 合格

else

echo '再牛的肖邦,也弹不出哥的悲伤'

fi

[root@server0 /]#

###################################################

案例:书写一个判断用户的脚本

利用位置变量$1,进行判断:

如果用户存在,输出用户基本信息(id 用户)

如果用户不存在,则创建该用户,输出用户创建成功

[root@server0 /]# vim /root/if04.sh

#!/bin/bash

id $1 &> /dev/null

if [ $? -eq 0 ];then

echo 用户已存在

id $1

else

useradd $1

echo 用户创建成功

fi

案例:

利用read 读入用户输入的IP地址

如果本机能够ping通该IP地址,则输出 可以通信

如果本机不能够ping通该IP地址,则输出 不可以通信

ping -c选项:指定ping几个包结束

[root@server0 /]# vim /root/if05.sh

#!/bin/bash

read -p '请输入您要测试的IP地址:' ip

ping -c 2 $ip &> /dev/null

if [ $? -eq 0 ];then

echo $ip可以通信

else

echo $ip不可以通信

fi

##################################################

for循环结构:解决反复执行重复性的操作

for 变量名 in 值列表

do

命令序列

done

[root@server0 /]# vim /root/for01.sh

#!/bin/bash

for a in lisi zhangsan wangwu dc tc

do

useradd $a &> /dev/null

echo $a创建成功

done

[root@server0 /]#

值的列表可以不参与循环体

[root@server0 /]# cat /root/for01.sh

#!/bin/bash

for a in {1..10}

do

echo I LOVE DC

done

造数工具: {起始..结束} {3..27} = 制造连续的数字 3到27

[root@server0 /]# cat /root/for02.sh

#!/bin/bash

for a in {1..10}

do

useradd nsd$a &> /dev/null

echo nsd$a创建成功

done

####################################################

案例4:编写一个判断脚本

在 server0 上创建 /root/foo.sh 脚本

1)当运行/root/foo.sh redhat,输出为fedora

2)当运行/root/foo.sh fedora,输出为redhat

3)当没有任何参数或者参数不是 redhat 或者fedora时,

其错误输出产生以下信息: /root/foo.sh redhat|fedora

####################################################

案例5:编写一个批量添加用户脚本

在 server0 上创建 /root/batchusers 脚本

1)此脚本要求提供用户名列表文件作为参数

2)如果没有提供参数,此脚本应该给出提示

Usage: /root/batchusers,退出并返回相应值

3)如果提供一个不存在的文件,此脚本应该给出提

示 Input file not found,退出并返回相应值

4)新用户的登录Shell为 /bin/false,无需设置密码

5)用户列表测试文件:

http://classroom/pub/materials/userlist

###################################################

小阶段2

day02

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

####################################################

案例4:编写一个判断脚本

在 server0 上创建 /root/foo.sh 脚本

1)当运行/root/foo.sh redhat,输出为fedora

2)当运行/root/foo.sh fedora,输出为redhat

3)当没有任何参数或者参数不是 redhat 或者fedora时,

其错误输出产生以下信息: /root/foo.sh redhat|fedora

[root@server0 ~]# vim /root/foo.sh

#!/bin/bash

if [ $# -eq 0 ];then

echo '/root/foo.sh redhat|fedora' >&2 #输出变成错误输出

exit 38 #程序退出返回值

elif [ $1 == redhat ];then

echo fedora

elif [ $1 == fedora ];then

echo redhat

else

echo '/root/foo.sh redhat|fedora' >&2 #输出变成错误输出

exit 39 #程序退出返回值

fi

[root@server0 ~]# /root/foo.sh tc

[root@server0 ~]# echo $?

####################################################

案例5:编写一个批量添加用户脚本

在 server0 上创建 /root/batchusers 脚本

1)此脚本要求提供用户名列表文件作为参数

2)如果没有提供参数,此脚本应该给出提示

Usage: /root/batchusers,退出并返回相应值

3)如果提供一个不存在的文件,此脚本应该给出提

示 Input file not found,退出并返回相应值

4)新用户的登录Shell为 /bin/false,无需设置密码

5)用户列表测试文件:

[root@server0 ~]# vim /root/userlist

duanwu

zhongqiu

zhsan

[root@server0 ~]# vim /root/batchusers

#!/bin/bash

if [ $# -eq 0 ];then #判断用户是否输入参数

echo 'Usage: /root/batchusers' >&2

exit 2

elif [ -f $1 ];then #判断用户输入参数,是否存在且为文件

for a in $(cat $1)

do

useradd -s /bin/false $a &> /dev/null

echo $a创建成功

done

else

echo 'Input file not found' >&2

exit 3

fi

[root@server0 ~]# /root/batchusers /root/userlist

###################################################

系统安全保护,SELinux安全机制

• Security-Enhanced Linux

– 美国NSA国家安全局主导开发,一套增强Linux系统安

全的强制访问控制体系

– 集成到Linux内核(2.6及以上)中运行

– RHEL7基于SELinux体系针对用户、进程、目录和文件

提供了预设的保护策略,以及管理工具

• SELinux的运行模式

– enforcing(强制)、permissive(宽松)

– disabled(彻底禁用)

任何模式变成disabled(彻底禁用)模式,必须修改配置文件重启系统

• 切换运行模式

– 临时切换:setenforce 1|0

– 固定配置:/etc/selinux/config 文件

虚拟机server

[root@server0 ~]# getenforce #查看当前SELinux状态

[root@server0 ~]# setenforce 0 #修改当前SELinux状态

[root@server0 ~]# getenforce

[root@server0 ~]# vim /etc/selinux/config #永久配置文件

命令模式按大写的C,可以删除光标之后进入插入模式

SELINUX=permissive

虚拟机desktop

[root@desktop0 ~]# getenforce

[root@desktop0 ~]# setenforce 0

[root@desktop0 ~]# getenforce

[root@desktop0 ~]# vim /etc/selinux/config

SELINUX=permissive

###################################################

配置用户环境

• 影响指定用户的 bash 解释环境

– ~/.bashrc,每次开启 bash 终端时生效

• 影响所有用户的 bash 解释环境

– /etc/bashrc,每次开启 bash 终端时生效

[root@server0 ~]# vim /root/.bashrc

alias hello='echo hello'

[root@server0 ~]# vim /home/student/.bashrc

alias hi='echo hi'

[root@server0 ~]# vim /etc/bashrc

alias tcc='echo dcc'

开启一个新的终端,再去测试(退出远程管理,在从新进行远程管理)

[root@server0 ~]# hello

[root@server0 ~]# hi #执行失败

[root@server0 ~]# tcc

[root@server0 ~]# su - student

[student@server0 ~]$ hello #执行失败

[student@server0 ~]$ hi

[student@server0 ~]$ tcc

[student@server0 ~]$ exit

真机上的操作:定义别名

[student@room9pc01 ~]$ vim /home/student/.bashrc

alias gos='ssh -X root@172.25.0.11'

alias god='ssh -X root@172.25.0.10'

真机上新开一个终端,进行验证

##################################################

防火墙策略管理

一 虚拟机server搭建基本Web服务

Web服务:提供一个页面内容的服务

1.安装httpd软件包

[root@server0 ~]# yum -y install httpd

[root@server0 ~]# rpm -q httpd

2.重启httpd服务

[root@server0 ~]# systemctl restart httpd

3.本机通过浏览器进行测试:

[root@server0 ~]# firefox 172.25.0.11

4.在虚拟机server书写一个页面文件

默认存放网页文件的路径: /var/www/html

默认页面文件名称:index.html

[root@server0 ~]# vim /var/www/html/index.html

<marquee><font color=red><h1>NSD1903阳光明媚

滚动 字体颜色 红色 标题字体

[root@server0 ~]# firefox 172.25.0.11

二 虚拟机server构建FTP服务器

FTP服务:文件传输服务

默认FTP共享路径:/var/ftp

1.安装vsftpd软件包

[root@server0 ~]# yum -y install vsftpd

2.重启vsftpd服务

[root@server0 ~]# systemctl restart vsftpd

3.验证FTP服务:

[root@server0 ~]# firefox ftp://172.25.0.11

#################################################

防火墙策略

作用:隔离

硬件防火墙

软件防火墙

firewalld服务基础

Linux的防火墙体系

• 系统服务:firewalld

• 管理工具:firewall-cmd、firewall-config

预设安全区域

• 根据所在的网络场所区分,预设保护规则集

– public:仅允许访问本机的sshd ping dhcpd服务

– trusted:允许任何访问

– block:阻塞任何来访请求(明确拒绝,客户端有接收回应)

– drop:丢弃任何来访的数据包(直接丢弃,客户端没有接收回应) 节省服务器资源

数据包: 目标IP地址 源IP地址 数据

防火墙判定的规则:匹配及停止

1.查看请求数据包中源IP地址,然后查询防火墙所有区域,那个区域有该IP地址的规则,则进入该区域

2.进入默认区域public(由root指定)

###############################################

一 防火墙默认区域的修改

虚拟机server:

]# firewall-cmd --get-default-zone #查看默认区域

虚拟机desktop:

]# ping -c 2 172.25.0.11 #可以通信

虚拟机server:

]# firewall-cmd --set-default-zone=block #修改默认区域

]# firewall-cmd --get-default-zone #查看默认区域

虚拟机desktop:

]# ping -c 2 172.25.0.11 #不可以通信,有回应

虚拟机server:

]# firewall-cmd --set-default-zone=drop #修改默认区域

]# firewall-cmd --get-default-zone #查看默认区域

虚拟机desktop:

]# ping -c 2 172.25.0.11 #不可以通信,没有回应

二 默认区域public 仅允许访问本机的sshd ping dhcpd服务

虚拟机server:

]# firewall-cmd --set-default-zone=public

]# firewall-cmd --get-default-zone

虚拟机desktop:

]# ping -c 2 172.25.0.11 #可以通信

在区域中添加允许访问的服务或协议

http:超文本传输协议

https:安全超文本传输协议

DNS:域名解析协议

FTP:文件传输协议

TFTP:简单文件传输协议

telnet:远程管理协议

pop3:邮件协议(用户收邮件协议)

SMTP:邮件协议(用户发邮件协议)

SNMP:简单的网络管理协议

虚拟机server:

# firewall-cmd --zone=public --list-all #查看区域规则

# firewall-cmd --zone=public --add-service=http #添加协议

# firewall-cmd --zone=public --list-all #查看区域规则

虚拟机desktop

# firefox 172.25.0.11 #可以访问

# firefox ftp://172.25.0.11 #不可以访问

虚拟机server:

]# firewall-cmd --zone=public --add-service=ftp #添加协议

]# firewall-cmd --zone=public --list-all #查看区域规则

虚拟机desktop

]# firefox ftp://172.25.0.11 #可以访问

####################################################

永久的防火墙规则 永久(permanent)

虚拟机server

]# firewall-cmd --reload #重新加载所有防火墙策略

]# firewall-cmd --zone=public --list-all #查看区域策略

]# firewall-cmd --permanent --zone=public --add-service=http #永久添加策略

]# firewall-cmd --zone=public --list-all #查看区域策略

]# firewall-cmd --reload #重新加载所有防火墙策略

]# firewall-cmd --zone=public --list-all #查看区域策略

####################################################

补充:

单独拒绝虚拟机desktop访问虚拟机server所有请求

虚拟机server

]# firewall-cmd --zone=block --add-source=172.25.0.10

]# firewall-cmd --zone=block --list-all

虚拟机desktop 访问测试

]# ping 172.25.0.11 #不可以访问

]# firefox 172.25.0.11 #不可以访问

真机 访问测试

]# ping 172.25.0.11 #可以访问

]# firefox 172.25.0.11 #可以访问

####################################################

####################################################

防火墙策略管理

一 虚拟机server搭建基本Web服务

1.安装httpd软件包

[root@server0 ~]# yum -y install httpd

2.重启httpd服务

[root@server0 ~]# systemctl restart httpd

3.在虚拟机server书写一个页面文件

默认存放网页文件的路径: /var/www/html

默认页面文件名称:index.html

]# echo '<h1>NSD1903' > /var/www/html/index.html

[root@server0 ~]# firefox 172.25.0.11

#################################################

防火墙策略

作用:隔离

硬件防火墙

软件防火墙

firewalld服务基础

Linux的防火墙体系

• 系统服务:firewalld

• 管理工具:firewall-cmd、firewall-config

预设安全区域

• 根据所在的网络场所区分,预设保护规则集

– public:仅允许访问本机的sshd ping dhcpd服务

– trusted:允许任何访问

– block:阻塞任何来访请求(明确拒绝,客户端有接收回应)

– drop:丢弃任何来访的数据包(直接丢弃,客户端没有接收回应) 节省服务器资源

数据包: 目标IP地址 源IP地址 数据

防火墙判定的规则:匹配及停止

1.查看请求数据包中源IP地址,然后查询防火墙所有区域,那个区域有该IP地址的规则,则进入该区域

2.进入默认区域public(由root指定)

###############################################

在区域中添加允许访问的服务或协议

http:超文本传输协议 默认端口:80

https:安全超文本传输协议 默认端口:443

DNS:域名解析协议 默认端口:53

FTP:文件传输协议 默认端口:21

TFTP:简单文件传输协议 默认端口:69

telnet:远程管理协议 默认端口:23

pop3:邮件协议(用户收邮件协议) 默认端口:110

SMTP:邮件协议(用户发邮件协议) 默认端口:25

SNMP:简单的网络管理协议 默认端口:161

虚拟机server:

# firewall-cmd --zone=public --list-all #查看区域规则

# firewall-cmd --permanent --zone=public --add-service=http #添加协议

# firewall-cmd --zone=public --list-all

# firewall-cmd --reload #重新加载防火墙所有策略

# firewall-cmd --zone=public --list-all #查看区域规则

虚拟机desktop

# firefox 172.25.0.11 #可以访问

####################################################

补充:

单独拒绝虚拟机desktop访问虚拟机server所有请求

虚拟机server

]# firewall-cmd --zone=block --add-source=172.25.0.10

]# firewall-cmd --zone=block --list-all

虚拟机desktop 访问测试

]# ping 172.25.0.11 #不可以访问

]# firefox 172.25.0.11 #不可以访问

真机 访问测试

]# ping 172.25.0.11 #可以访问

]# firefox 172.25.0.11 #可以访问

####################################################

端口: 编号 标识服务或协议或程序

可以利用root用户修改 服务或协议或程序的端口

实现本机的端口映射

• 本地应用的端口重定向(端口1 --> 端口2)

– 从客户机访问 端口1 的请求,自动映射到本机 端口2

– 比如,访问以下两个地址可以看到相同的页面:

客户端访问是:172.25.0.11 5423 ----->防火墙----->172.25.0.11 80

虚拟机server

]# firewall-cmd --permanent --zone=public

--add-forward-port=port=5423:proto=tcp:toport=80

]# firewall-cmd --reload //重载配置

]# firewall-cmd --zone=public --list-all

虚拟机desktop

]# firefox 172.25.0.11:5423 #访问时加上端口

###################################################

#####################################################

触发挂载

1.关闭虚拟机server,添加光驱设备,将光盘放入

2.安装autofs可以实现触发挂载软件

[root@server0 ~]# yum -y install autofs

3.重启autofs服务

[root@server0 ~]# systemctl restart autofs

[root@server0 ~]# systemctl enable autofs

4.进行触发挂载

[root@server0 ~]# ls /misc

[root@server0 ~]# ls /misc/cd

#################################################

]# ls /etc/auto.master #触发挂载的主配置文件

]# ls /etc/auto.misc #挂载配置文件

触发挂载必须为二层目录结构

/nsd/第一层称之为监控点,必须有管理员亲自创建

监控点由/etc/auto.master设置

[root@server0 /]# vim /etc/auto.master

/misc /etc/auto.misc

/nsd /etc/test

监控点 指定挂在配置文件的路径

翻译:

设置监控点为/misc,要想知道挂载什么设备,请看/etc/auto.misc文件

[root@server0 /]# fdisk /dev/vdb

[root@server0 /]# mkfs.xfs /dev/vdb1

[root@server0 /]# vim /etc/test

abc -fstype=xfs,rw :/dev/vdb1

挂载点 文件系统类型及参数 挂载设备

[root@server0 /]# systemctl restart autofs

[root@server0 /]# cd /nsd/abc

###############################################

添加一个 swap分区(交换分区)

作用:在一定程度上,缓解内存不足

相当于虚拟内存

Linux一般使用分区空间,来充当交换空间

[root@server0 ~]# fdisk /dev/vdb #划分一个512M的分区

[root@server0 ~]# mkswap /dev/vdb1 #格式化交换文件系统

[root@server0 ~]# blkid /dev/vdb1 #查看文件系统类型

[root@server0 ~]# swapon -s #查看交换空间组成 成员信息

[root@server0 ~]# swapon /dev/vdb1 #启用

[root@server0 ~]# swapon -s #查看交换空间组成 成员信息

[root@server0 ~]# swapoff /dev/vdb1 #停用

[root@server0 ~]# vim /etc/fstab

/dev/vdb1 swap swap defaults 0 0

[root@serverX ~]# swapon -a //检测/etc/fstab交换分区，书写正确会进行启用

[root@serverX ~]# swapon -s //查看交换分区启用情况

###################################################

小阶段2

day03

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

####################################################

环境准备:

虚拟机server

]# firewall-cmd --set-default-zone=trusted #允许所有访问

虚拟机desktop

]# firewall-cmd --set-default-zone=trusted #允许所有访问

###################################################

在虚拟机server:划分一个3G的主分区

[root@server0 ~]# fdisk /dev/vdb

[root@server0 ~]# lsblk

[root@server0 ~]# ls /dev/vdb1

###################################################

iSCSI服务基础

iSCSI磁盘的工作模式(默认端口:3260)

• Internet SCSI,网际SCSI接口

– 一种基于C/S架构的虚拟磁盘技术

– 服务器提供磁盘空间,客户机连接并当成本地磁盘使用

• backstore,后端存储

– 对应到服务端提供实际存储空间的设备,需要起一个管理名称

• target,磁盘组(木质的箱子)

– 是客户端的访问目标,作为一个框架,由多个lun组成

• lun,逻辑单元(将后端存储放入磁盘组)

– 每一个lun需要关联到某一个后端存储设备,在客户端

会视为一块虚拟硬盘

/dev/vdb1 (backstore,后端存储)

大冰箱---->纸质的包装 木质的箱子

• ISCSI Qualified Name(iqn) 名称规范

– iqn.yyyy-mm.倒序域名:自定义标识

– 用来识别 target 磁盘组,也用来识别客户机身份

• 名称示例

– iqn.2016-02.com.example:server0

– iqn.2016-02.com.example:desktop0

服务端:虚拟机server 提供磁盘空间

一 所需软件包:targetcli(服务端软件)

[root@server0 ~]# yum -y install targetcli

二 运行targetcli进行配置

[root@server0 ~]# targetcli

/> ls

1)创建后端存储(backstore,后端存储)

/> backstores/block create name=nsd dev=/dev/vdb1

/> ls

2)创建target,磁盘组(木质的箱子)

[root@server0 ~]# targetcli

/> iscsi/ create iqn.2019-04.com.example:server0

/> ls

3)配置lun,关联(将后端存储放入磁盘组)

/> iscsi/iqn.2019-04.com.example:server0/tpg1/luns create /backstores/block/nsd

/> ls

4)配置ACL访问控制,设置客户端声称的名字

/> iscsi/iqn.2019-04.com.example:server0/tpg1/acls create iqn.2019-04.com.example:desktop0

/> ls

5)设置本机提供服务端的IP地址与端口

/> iscsi/iqn.2019-04.com.example:server0/tpg1/portals create 172.25.0.11

/> ls

/> exit #保存并退出

三 重启服务端服务

[root@server0 ~]# systemctl restart target

[root@server0 ~]# systemctl enable target #设置开机自启动

ln -s '/usr/lib/systemd/system/target.service' '/etc/systemd/system/multi-user.target.wants/target.service'

#提示:会制作一个快捷方式,不是报错

###################################################

客户端:虚拟机desktop 访问使用服务端提供磁盘空间

一 安装客户端软件

• 所需软件包:iscsi-initiator-utils

Yum仓库软件包的补全: 1.Yum必须要有缓存 2.当前系统没有安装

Yum必须要有缓存 :yum repolist 可以生成缓存

[root@desktop0 ~]# rpm -q iscsi-initiator-utils

iscsi-initiator-utils-6.2.0.873-21.el7.x86\_64

[root@desktop0 ~]# yum -y install iscs(tab)

二 修改配置文件,指定客户端生成的命令

[root@desktop0 ~]# vim /etc/iscsi/initiatorname.iscsi

InitiatorName=iqn.2019-04.com.example:desktop0

三 重启iscsid服务,用于识别客户端声称的名字

[root@desktop0 ~]# systemctl restart iscsid

Warning: Unit file of iscsid.service changed on

disk, 'systemctl daemon-reload' recommended.

[root@desktop0 ~]# systemctl daemon-reload

[root@desktop0 ~]# systemctl restart iscsid

四 运行iscsiadm命令,发现服务端共享存储

Linux查看命令帮助的方法 : man iscsiadm

全文查找/example 按n项下跳转匹配,看到EXAMPLES

Ctrl - ：减小字体

Ctrl Shift + ：变大字体

]# iscsiadm --mode discoverydb --type sendtargets --portal 172.25.0.11 --discover

屏幕显示信息:

172.25.0.11:3260,1 iqn.2019-04.com.example:server0

五 重启iscsi服务(客户端服务),加载服务端共享存储

[root@desktop0 ~]# systemctl restart iscsi

[root@desktop0 ~]# systemctl enable iscsi

[root@desktop0 ~]# lsblk

#################################################

数据库基本管理

什么是数据库:存放数据的仓库

数据库的结构: 在数据库里,有很多的库,在每一个库中有很多的表格

一 部署mariadb数据库服务器(默认的端口 3306)

mariadb-server:提供服务端有关的系统程序

[root@server0 ~]# yum -y install mariadb-server

[root@server0 ~]# systemctl restart mariadb

二 数据库的基本操作

[root@server0 ~]# mysql #进入数据库,默认无需密码

MariaDB [(none)]> show databases; #查看所有数据库

MariaDB [(none)]> create database nsd; #创建库

MariaDB [(none)]> show databases; #查看所有数据库

MariaDB [(none)]> drop database nsd; #删除nsd库

MariaDB [(none)]> show databases; #查看所有数据库

MariaDB [(none)]> create database nsd1903; #创建库

MariaDB [(none)]> show databases;

MariaDB [(none)]> exit #15:05

三 为数据库管理员设置密码

数据库管理员为root,数据库所有用户信息,由mysql库中user表

系统管理员为root,系统所有用户信息,由/etc/passwd

MariaDB [(none)]> exit

Bye

[root@server0 ~]#

• 为数据库账号修改密码

– mysqladmin [-u用户名] [-p[旧密码]] password '新密码'

[root@server0 ~]# mysqladmin -u root password '123'

[root@server0 ~]# mysql -u root -p #交互式

Enter password:

[root@server0 ~]# mysql -u root -p123 #非交互式

四 表格操作: 表记录 表字段

编号 姓名 性别 联系方式 地址

1 dc 女 123 东村

2 tc 女 456 西村

3 lisi 男 789 北庄

四大操作: 增 insert 删 delete 改 update 查 select

[root@server0 ~]# mysql -u root -p123

MariaDB [(none)]> show databases;

MariaDB [test]> use mysql; #进入mysql库

MariaDB [mysql]> show tables; #查看当前库中有那些表格

MariaDB [mysql]> desc user; #查看表结构

数据库的导入操作

• 导入/恢复到数据库

]# wget http://classroom.example.com/pub/materials/users.sql

]# ls

导入数据到指定数据库:

]# mysql -u root -p123 nsd1903 < /root/users.sql

]# mysql -u root -p123

> use nsd1903;

> show tables;

+-------------------+

| Tables\_in\_nsd1903 |

+-------------------+

| base |

| location |

+-------------------+

###################################################

表格的查询:

select 表字段 from 表名;

]# mysql -u root -p123

> use nsd1903;

> select \* from base;

> use mysql;

> select \* from nsd1903.base;

> use nsd1903;

> select \* from location;

> select name from base;

> select id,name from base;

有条件查询:

查找base中密码为123的人?

> use nsd1903;

> select \* from base;

> select \* from base where password='123';

> select \* from base where password='456';

> select \* from base where name='tom';

> select id,name from base where name='tom';

> select name,password from base where name='tom';

• 禁止网络进行连接,只服务于本机

[root@server0 ~]# vim /etc/my.cnf #主配置文件

[mysqld]

skip-networking //手写,跳过网络监听

.. ..

[root@server0 ~]# systemctl restart mariadb

###############################################

数据库授权

– 除了 root 用户,此nsd1903数据库,只能被用户lisi查询,

此用户的密码为123

不需要创建系统用户lisi

– GRANT 权限列表 ON 数据库名.表名 TO 用户名@客户机地址 IDENTIFIED BY '密码';

grant select on nsd1903.\* to lisi@localhost identified by '123';

当lisi从本地localhost进行登录,输入密码123,将会获得nsd1903库所有表的查询权限.

[root@server0 ~]# mysql -u root -p123

> grant select on nsd1903.\* to lisi@localhost identified by '123';

> select user,host,password from mysql.user;

[root@server0 ~]# mysql -u lisi -p123

MariaDB [(none)]> exit

Bye

#################################################

案例4:使用数据库查询

2. 在系统 server0 上使用数据库 nsd1903,并使用相

应的 SQL 查询以回答下列问题:

1)base表密码是 solicitous 的人的名字?

[root@server0 ~]# mysql -u root -p123

> use nsd1903;

> select \* from base;

> select \* from base where password='solicitous';

2)有多少人的姓名是 Barbara 同时居住在 Sunnyvale?

> select \* from base,location

where base.name='Barbara' and

location.city='Sunnyvale' and base.id=location.id;

> select count(\*) from base,location

where base.name='Barbara' and

location.city='Sunnyvale' and base.id=location.id;

> insert base values('6','Barbara','789'); #插入表记录

> select \* from base;

> insert location values('6','Sunnyvale'); #插入表记录

> select \* from location;

1. 禁止空密码root用户访问 mariadb数据库

> use mysql;

> select user,host,password from user;

> select user,host,password from user where password='';

> delete from user where password='';

> select user,host,password from user;

> flush privileges; #刷新user所有记录

[root@server0 ~]# mysql -u root -h server0.example.com

登录失败

####################################################

小阶段2

day04

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

reset(还原)

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

####################################################

环境准备:

虚拟机server

]# firewall-cmd --set-default-zone=trusted #允许所有访问

虚拟机desktop

]# firewall-cmd --set-default-zone=trusted #允许所有访问

####################################################

配置NFS共享

NFS共享概述

• Network File System,网络文件系统

– 用途:为客户机提供共享使用的文件夹

– 协议:NFS(TCP/UDP 2049)、RPC(TCP/UDP 111)

• 所需软件包:nfs-utils

• 系统服务:nfs-server

虚拟机server:

1.安装软件包nfs-utils

[root@serverq0 ~]# rpm -q nfs-utils

nfs-utils-1.3.0-0.el7.x86\_64

2.创建共享目录

[root@server0 ~]# mkdir /public

[root@server0 ~]# echo 123 > /public/1.txt

[root@server0 ~]# ls /public/

3.修改配置文件/etc/exports,指定共享的目录

[root@server0 ~]# vim /etc/exports

[root@server0 ~]# man exports

– 共享的文件夹路径 客户机地址(权限) 客户机地址(权限) .. ..

/public \*(ro)

[root@server0 ~]# systemctl restart nfs-server

[root@server0 ~]# systemctl enable nfs-server

#################################################

客户端虚拟机desktop

1.查看服务端共享路径

[root@desktop0 ~]# showmount -e 172.25.0.11

Export list for 172.25.0.11:

/public \*

[root@desktop0 ~]#

2.挂载访问

[root@desktop0 ~]# mkdir /mnt/nfs

[root@desktop0 ~]# mount 172.25.0.11:/public /mnt/nfs/

[root@desktop0 ~]# df -h

[root@desktop0 ~]# ls /mnt/nfs

3.开机自动挂载

\_netdev:声明为网络设备

开机启动Linux系统时,首先具备网络参数后,再进行挂载本设备

[root@desktop0 ~]# vim /etc/fstab

设备路径 挂载点 类型 参数 备份标记 检测顺序

172.25.0.11:/public /mnt/nfs nfs defaults,\_netdev 0 0

[root@desktop0 ~]# umount /mnt/nfs/

[root@desktop0 ~]# df -h

[root@desktop0 ~]# mount -a

[root@desktop0 ~]# df -h

#####################################################

虚拟机server:

1.利用NFS服务发布/nsd共享目录

[root@server0 ~]# mkdir /nsd

[root@server0 ~]# echo 123 > /nsd/a.txt

[root@server0 ~]# ls /nsd/

2.允许所有客户端进行只读访问

[root@server0 ~]# cat /etc/exports

/public \*(ro)

/nsd \*(ro)

虚拟机desktop:

1.完成开机自动挂载到本机/mnt/test目录

172.25.0.11:/nsd /mnt/test nfs defaults,\_netdev 0 0

[root@desktop0 ~]# df -ah #显示当前系统,所有正在挂载信息

####################################################

HTTP服务基础

一 构建基本Web主机

Web通信基本概念

• 基于 B/S(Browser/Server)架构的网页服务

– 服务端提供网页

– 浏览器下载并显示网页

• Hyper Text Markup Language(html),超文本标记语言

• Hyper Text Transfer Protocol(http),超文本传输协议 默认端口 80

虚拟机server:

]# yum -y install httpd

]# systemctl restart httpd

]# echo '<h1>NSD1903' > /var/www/html/index.html

虚拟机desktop

]# firefox 172.25.0.11

####################################################

web服务器的软件

httpd(Apache软件基金会)

• 提供的默认配置

– Listen:监听地址:端口(80)

– ServerName:本站点注册的DNS名称(空缺)

– DocumentRoot:网页根目录(/var/www/html)

DNS服务器:虚拟机classroom提供域名解析

server0.example.com

www0.example.com

webapp0.example.com

测试解析结果:

[root@server0 ~]# nslookup server0.example.com

[root@server0 ~]# nslookup www0.example.com

[root@server0 ~]# nslookup webapp0.example.com

##################################################

DocumentRoot:网页文件根目录(/var/www/html)

虚拟机server

]# mkdir /var/www/myweb

]# echo '<h1>wo shi myweb' > /var/www/myweb/index.html

]# ls /var/www/myweb/

]# vim /etc/httpd/conf/httpd.conf

DocumentRoot "/var/www/myweb"

]# systemctl restart httpd

虚拟机desktop

]# firefox 172.25.0.11 #测试访问结果

#####################################################

网络路径与实际路径

客户端访问:

firefox 172.25.0.11:80-->服务端 httpd-->DocumentRoot /var/www/myweb-->index.html

客户端 : firefox 172.25.0.11/abc

服务端实际路径: : /var/www/myweb/abc/index.html

客户端 : firefox 172.25.0.11/var/www/myweb/abc

服务端实际路径: /var/www/myweb/var/www/myweb/abc

虚拟机server

]# mkdir /var/www/myweb/abc

]# echo '<h1>wo shi abc' > /var/www/myweb/abc/index.html

虚拟机desktop

]# firefox 172.25.0.11 #测试访问结果

]# firefox 172.25.0.11/abc #测试访问结果

##################################################

客户端访问Web: 直接输入IP地址 = 服务端 DocumentRoot

###################################################

配置文件调用

• 配置文件路径

– /etc/httpd/conf/httpd.conf #主配置文件

– /etc/httpd/conf.d/\*.conf #调用配置文件

###################################################

虚拟Web主机

虚拟主机的含义及类型

• 虚拟Web主机

– 由同一台服务器提供多个不同的Web站点

• 区分方式

– 基于域名的虚拟主机

– 基于端口的虚拟主机

– 基于IP地址的虚拟主机

server0.example.com

www0.example.com

webapp0.example.com

<VirtualHost IP地址:端口>

ServerName 此站点的DNS名称

DocumentRoot 此站点的网页根目录

</VirtualHost>

]# vim /etc/httpd/conf.d/nsd01.conf

<VirtualHost \*:80> #本机所有IP地址的80端口启用虚拟Web主机

ServerName server0.example.com #设置网站访问的域名

DocumentRoot /var/www/qq #设置网页文件根目录

</VirtualHost>

<VirtualHost \*:80>

ServerName www0.example.com

DocumentRoot /var/www/baidu

</VirtualHost>

]# mkdir /var/www/qq /var/www/baidu

]# echo '<h1>QQ' > /var/www/qq/index.html

]# echo '<h1>百度' > /var/www/baidu/index.html

]# systemctl restart httpd

###############################################

DocumentRoot 路径尽量放在/var/www , SELinux策略

###############################################

虚拟Web主机优先级,由上到下优先级递减

一旦使用了虚拟Web主机,所有的页面都必须使用虚拟Web主机来呈现

<VirtualHost \*:80>

ServerName www0.example.com

DocumentRoot /var/www/baidu

</VirtualHost>

<VirtualHost \*:80>

ServerName server0.example.com

DocumentRoot /var/www/qq

</VirtualHost>

<VirtualHost \*:80>

ServerName webapp0.example.com

DocumentRoot /var/www/myweb

</VirtualHost>

####################################################

补充: Samba共享服务 (Windows平台与Linux平台 跨平台的共享)

协议:smb(沟通验证身份) cifs(存储或传输数据)

默认情况下必须通用户验证,才可以访问

专门用于Samba共享服务,验证的用户:Samba共享帐号

1.必须要有相应的本地用户对应

2.采用独立的密码设置

虚拟机server:

1.安装软件包samba

[root@server0 ~]# yum -y install samba

2.创建Samba共享帐号

]# useradd -s /sbin/nologin harry

]# pdbedit -a harry #将用户添加为Samba共享帐号

new password:

retype new password:

pdbedit -x 用户名 #删除Samba共享帐号

pdbedit -L #查看当前系统,所有的Samba共享帐号

3.修改配置文件

[root@server0 ~]# vim /etc/samba/smb.conf

在命令模式下,输入G(大写)到全文的最后一行

[nsd] #设置共享名

path = /common #设置实际共享的目录

[root@server0 ~]# mkdir /common

[root@server0 ~]# echo 123 > /common/123.txt

[root@server0 ~]# ls /common/

4.重启smb服务

[root@server0 ~]# systemctl restart smb

5.修改SELinux策略,布尔值策略(功能的开关)

samba\_export\_all\_ro : Samba服务只读共享功能

]# getenforce #查看SELinux状态

]# getsebool -a | grep samba #查看SELinux所有布尔值

]# setsebool samba\_export\_all\_ro on #修改SELinux布尔值

]# getsebool -a | grep samba

]# getenforce

虚拟机desktop:

1.安装软件包,支持Samba共享的挂载,让本机识别cifs协议

]# yum -y install cifs-utils

]# rpm -q cifs-utils

2.挂载访问Samba共享

]# mkdir /mnt/smb

]# mount -o user=harry,pass=123 //172.25.0.11/nsd /mnt/smb/

格式:

mount -o user=用户名,pass=密码 //服务端IP地址/共享名 挂载点

]# df -h

3.开机自动挂载,书写/etc/fstab

//172.25.0.11/nsd /mnt/smb cifs defaults,\_netdev,user=harry,pass=123 0 0

[root@desktop0 ~]# umount /mnt/smb/

[root@desktop0 ~]# mount -a

[root@desktop0 ~]# df -h

##############################################

小阶段2

day05

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

####################################################

环境准备:

虚拟机server

]# firewall-cmd --set-default-zone=trusted #允许所有访问

虚拟机desktop

]# firewall-cmd --set-default-zone=trusted #允许所有访问

####################################################

<VirtualHost \*:80>

ServerName www0.example.com

DocumentRoot /var/www/web01

</VirtualHost>

<VirtualHost \*:80>

ServerName server0.example.com

DocumentRoot /var/www/web02

</VirtualHost>

<VirtualHost \*:80>

ServerName webapp0.example.com

DocumentRoot /var/www/web03

</VirtualHost>

####################################################

- 实现客户端访问www0.example.com网页内容为 大圣又归来

- 实现客户端访问server0.example.com网页内容为 大圣归来

- 实现客户端访问webapp0.example.com网页内容为 大圣累了

####################################################

网页内容访问

客户机地址限制

• 使用 <Directory> 配置区段

– 每个文件夹自动继承其父目录的ACL访问权限

– 除非针对子目录有明确设置

<Directory 目录的绝对路径>

.. ..

Require all denied|granted

Require ip IP或网段地址 .. ..

</Directory>

<Directory /var/www/web01>

Require all denied #拒绝所有人访问

</Directory>

<Directory /var/www/web02>

Require all granted #允许所有人访问

</Directory>

<Directory /var/www/web03>

Require ip 172.25.0.11 #仅允许172.25.0.11访问

</Directory>

##############################################

<Directory /var/www/web01>

Require all denied #拒绝所有人访问

</Directory>

<Directory /var/www/web01/abc>

Require all granted #允许所有人访问

</Directory>

/var/www/web01/abc/index.html #允许

/var/www/web01/nsd/index.html #拒绝

############################################

案例1:配置网页内容访问

在 Web 网站 http://server0.example.com 的

DocumentRoot 目录下创建一个名为 private 的子目录

要求如下:

1.虚拟机server,查看DocumentRoot 目录路径

]# cat /etc/httpd/conf.d/nsd01.conf

]# mkdir /var/www/web02/private

]# echo '<h1>wo shi private' > /var/www/web02/private/index.html

2.虚拟机desktop访问测试

]# firefox server0.example.com/private

3.从虚拟机server0上,任何人都可以浏览 private的内容,

但是从其他系统不能访问这个目录的内容

[root@server0 ~]# cat /etc/httpd/conf.d/nsd02.conf

<Directory /var/www/web02/private>

Require ip 172.25.0.11 #仅允许172.25.0.11访问

</Directory>

[root@server0 ~]# systemctl restart httpd

在虚拟机desktop测试: firefox server0.example.com/private

Forbidden

You don't have permission to access /private/ on this server.

####################################################

案例2:使用自定Web根目录

调整 Web 站点 http://server0.example.com 的网页目录,要求如下:

1)新建目录 /webroot,作为此站点新的网页目录

]# mkdir /webroot

]# echo '<h1>wo shi webroot' > /webroot/index.html

]# vim /etc/httpd/conf.d/nsd01.conf

......

<VirtualHost \*:80>

ServerName server0.example.com

DocumentRoot /webroot

</Virtualhost>

......

]# vim /etc/httpd/conf.d/nsd02.conf

.......

<Directory /webroot>

Require all granted #允许所有人进行访问

</Directory>

[root@server0 ~]# systemctl restart httpd

2)修改SELinux策略,安全上下文(标签)

[root@server0 ~]# ls -Zd /var/www/ #查看目录SELinux标签值

[root@server0 ~]# ls -Zd /webroot/

• 方式1:参照标准目录,重设新目录的属性

– chcon [-R] --reference=模板目录 新目录

[root@server0 ~]# chcon -R --reference=/var/www /webroot

[root@server0 ~]# ls -Zd /webroot

[root@server0 ~]# firefox server0.example.com

#####################################################

部署动态网站

静态网站:

• 服务端的原始网页 = 浏览器访问到的网页

– 由Web服务软件处理所有请求

– 文本(txt/html)、图片(jpg/png)等静态资源

动态网站:

• 服务端的原始网页 ≠ 浏览器访问到的网页

– 由Web服务软件接受请求,动态程序转后端模块处理

– PHP网页、Python网页、JSP网页......

部署Python动态页面,为站点 webapp0.example.com

虚拟机Server:

1.下载Python页面文件

[root@server0 ~]# cat /etc/httpd/conf.d/nsd01.conf

[root@server0 ~]# cd /var/www/web03/

[root@server0 web03]# wget http://classroom.example.com/pub/materials/webinfo.wsgi

[root@server0 web03]# ls

2.实现客户端方便访问,设置页面跳转(页面别名 地址重写)

webapp0.example.com ---> webinfo.wsgi

Alias 网络路径 实际路径

网络路径为 / : 用户直接输入域名

当客户端访问 网络路径 ,将 实际路径 页面文件呈现

[root@server0 ~]# vim /etc/httpd/conf.d/nsd01.conf

<VirtualHost \*:80>

ServerName webapp0.example.com

DocumentRoot /var/www/web03

Alias / /var/www/web03/webinfo.wsgi

当用户直接访问网页文件根目录时,将webinfo.wsgi页面呈现

</Virtualhost>

[root@server0 ~]# systemctl restart httpd

3.虚拟机Server,安装mod\_wsgi软件进行Pyhton页面代码的翻译

[root@server0 ~]# yum -y install mod\_wsgi

4.修改配置文件,添加翻译配置

[root@server0 ~]# vim /etc/httpd/conf.d/nsd01.conf

<VirtualHost \*:80>

ServerName webapp0.example.com

DocumentRoot /var/www/web03

WsgiScriptAlias / /var/www/web03/webinfo.wsgi

</Virtualhost>

[root@server0 ~]# systemctl restart httpd

访问测试:firefox webapp0.example.com

页面内容显示 UNIX时间戳:

从1970-1-1 0:0:0到现在时间所有经历的秒数

5.此虚拟主机侦听在端口8909

[root@server0 ~]# vim /etc/httpd/conf.d/nsd01.conf

Listen 8909 #httpd监听8909端口

<VirtualHost \*:8909> #设置虚拟web主机的端口 15:15上课

ServerName webapp0.example.com

DocumentRoot /var/www/web03

WsgiScriptAlias / /var/www/web03/webinfo.wsgi

</Virtualhost>

6.修改SELinux策略,非默认端口开放的策略

]# semanage port -l | grep http

]# semanage port -a -t http\_port\_t -p tcp 8909

-a:添加 -t:类型 -p:协议

]# semanage port -l | grep http

]# systemctl restart httpd

]# firefox webapp0.example.com:8909

##################################################

端口优先级最高,然后虚拟Web主机由上到下进行匹配

#################################################

安全Web服务

PKI公钥基础设施

• Public Key Infrastructure,公钥基础设施

– 公钥:主要用来加密数据

– 私钥:主要用来解密数据(与相应的公钥匹配)

– 数字证书:证明拥有者的合法性/权威性(单位名称、

有效期、公钥、颁发机构及签名、......)

– Certificate Authority,数字证书授权中心:负责证书

的申请/审核/颁发/鉴定/撤销等管理工作

虚拟机server部署安全的Web服务器

1.部署网站证书(营业执照)

]# cd /etc/pki/tls/certs/

]# wget http://classroom.example.com/pub/tls/certs/server0.crt

]# ls

2.部署根证书(工商局的信息)

]# cd /etc/pki/tls/certs/

]# wget http://classroom.example.com/pub/example-ca.crt

]# ls

3.部署用于解密数据的秘钥

]# cd /etc/pki/tls/private/

]# wget http://classroom.example.com/pub/tls/private/server0.key

]# ls

4.安装mod\_ssl软件,实现加密Web传输

[root@server0 /]# yum -y install mod\_ssl

[root@server0 /]# ls /etc/httpd/conf.d/

5.修改配置文件/etc/httpd/conf.d/ssl.conf

[root@server0 /]# vim /etc/httpd/conf.d/ssl.conf

vim末行模式 :set nu 开启行号功能

59 DocumentRoot "/var/www/web01"

60 ServerName www0.example.com:443

#指定网站证书

100 SSLCertificateFile /etc/pki/tls/certs/server0.crt

#指定秘钥

107 SSLCertificateKeyFile /etc/pki/tls/private/server0.key

#指定根证书

122 SSLCACertificateFile /etc/pki/tls/certs/example-ca.crt

[root@server0 /]# systemctl restart httpd

最后测试:

[root@server0 /]# firefox https://www0.example.com

小阶段2

day06

教学环境介绍

– 格式:rht-vmctl 控制指令 虚拟机名

真机上操作：

　]# rht-vmctl reset classroom　 //先重置资源服务器

　]# rht-vmctl reset server

　]# rht-vmctl reset desktop　　 //再重置练习虚拟机

####################################################

环境准备:

虚拟机server

]# firewall-cmd --set-default-zone=trusted #允许所有访问

虚拟机desktop

]# firewall-cmd --set-default-zone=trusted #允许所有访问

###################################################

基础邮件服务

电子邮件通信

• 电子邮件服务器的基本功能

– 为用户提供电子邮箱存储空间(用户名@邮件域名)

– 处理用户发出的邮件 —— 传递给收件服务器

– 处理用户收到的邮件 —— 投递到邮箱

快速部署postfix邮件服务器

• 装包、配置、起服务

– 默认的标准配置即可为本机提供发/收邮件服务

– 若有必要,可扩大服务范围(邮件域)

DNS服务器:虚拟机classroom

yg@server0.example.com

xln@server0.example.com

虚拟机server:

1.安装软件postfix,提供邮件功能的软件

[root@server0 ~]# rpm -q postfixpostfix-2.10.1-6.el7.x86\_64

[root@server0 ~]#

2. 修改配置文件/etc/postfix/main.cf

vim末行模式 :set nu

99 myorigin = server0.example.com #默认补全邮箱帐号后缀

116 inet\_interfaces = all #本机所有网卡都提供邮件服务

164 mydestination = server0.example.com

#判断为本域邮件依据

收件人:xln@server0.example.com

3.重启postfix邮件服务

[root@server0 ~]# systemctl restart postfix

使用mail命令发信/收信

– mail -s '邮件标题' -r 发件人 收件人[@收件域]...

• mail 收信操作

– mail [-u 用户名]

[root@server0 ~]# useradd yg

[root@server0 ~]# useradd xln

[root@server0 ~]# mail -s 'test01' -r yg xln

haha xixi hehe lele

. #当只有一个点时,提交并发送邮件

EOT

[root@server0 ~]# echo dc | mail -s 'test02' -r yg xln

[root@server0 ~]# mail -u xln

Heirloom Mail version 12.5 7/5/10. Type ? for help.

"/var/mail/xln": 2 messages 2 new

>N 1 yg@server0.example.c Thu Apr 18 10:31 18/588

> 1 #输入邮件的编号1,查看邮件内容

####################################################

准备交换分区

交换空间(虚拟内存):缓解内存不足

mkswap swapon swapoff swapon -s swapon -a

parted分区工具(划分GPT分区模式)

GPT分区模式: 128个主分区,最大空间18EB

[root@server0 ~]# parted /dev/vdb

(parted) mktable gpt #初始化磁盘,不能重复执行

(parted) mkpart #划分新的分区

分区名字? #随便写

文件系统类型？[ext2]? xfs #随便写不起实际作用

起始点？ 0

结束点？ 2G

忽略/Ignore/放弃/Cancel? Ignore

(parted) print

(parted) unit GB #显示单位

(parted) print

(parted) quit

(parted) rm #删除分区

[root@server0 ~]# lsblk

###################################################

配置IPv6地址

• IPv4 地址表示

– 32个二进制位,点 分隔的十进制数

• IPv6 地址表示

– 128个二进制位, 利用冒号分隔 8 部分,最终每一部分由16进制数标识

– 每段内连续的前置 0 可省略、连续的多个 : 可简化为 ::

– 例如: 2003:ac18:0000:0000:0000:0000:0000:0305

2003:ac18::305

为虚拟机server:配置IPv6地址2003:ac18::305/64

1.利用nmcli命令查看网卡名称

[root@server0 ~]# nmcli connection show

名称

System eth0

2.修改地址

]# nmcli connection modify 'System eth0'

ipv6.method manual ipv6.addresses 2003:ac18::305/64

connection.autoconnect yes

]# nmcli connection 修改 '识别的网卡名称'

ipv6.方法 手工配置 ipv6.地址 2003:ac18::305/64

每次开机自动启用

3.激活配置

[root@server0 ~]# nmcli connection up 'S(tab)'

[root@server0 ~]# ifconfig | less

inet6 2003:ac18::305 prefixlen 64

[root@server0 ~]# ping6 2003:ac18::305

##################################################

配置聚合连接(链路聚合)

• team,聚合连接(也称为链路聚合)

– 由多块网卡(team-slave)一起组建而成的虚拟网卡,即"组队"

– 作用1:热备份(activebackup)连接冗余

eth1 eth2

虚拟网卡 team0 192.168.1.1

虚拟机配置链路聚合:

1.创建虚拟网卡team0 ,参考 man teamd.conf 全文搜索/example

]# nmcli connection add type team

con-name team0 ifname team0 autoconnect yes

config '{"runner": {"name": "activebackup"}}'

]# nmcli connection 添加 类型 team(组队)

配置文件名 team0 网卡命名为 team0 每次开机自动启用

配置 链路聚合以activebackup(热备)工作的方式

]# ifconfig #查看team0网卡的生成

如果有敲错,请删除从新创建team0

]# nmcli connection delete team0

2.添加成员

]# nmcli connection add type team-slave

con-name team0-1 ifname eth1 master team0

]# nmcli connection add type team-slave

con-name team0-2 ifname eth2 master team0

]# nmcli connection 添加 类型 组队-成员

配置文件名 team0-1 网卡设备名 eth1 主设备为 team0

]# ls /etc/sysconfig/network-scripts/ #查看生成的配置文件

如果有敲错,请删除从新创建team0

]# nmcli connection delete team0-1 (生成配置文件名)

]# nmcli connection delete team0-2 (生成配置文件名)

3.配置IP地址,再激活

]# nmcli connection modify team0

ipv4.method manual ipv4.addresses 192.168.1.1/24

connection.autoconnect yes

]# nmcli connection up team0 #激活team0

]# ifconfig team0 | head -2 #查看team0的IP地址信息

]# teamdctl team0 state #查看team0信息

]# ifconfig eth2 down #将eth2网卡禁用

]# teamdctl team0 state #查看team0信息

]# ifconfig eth2 up #将eth2网卡启用

小阶段3

day01

新教学环境的搭建

一 构建两台虚拟机

真机上运行:

[student@room9pc01 ~]$ clone-vm7

Enter VM number: 8

Creating Virtual Machine disk image...... [OK]

Defining new virtual machine...... [OK]

[student@room9pc01 ~]$ clone-vm7

Enter VM number: 9

Creating Virtual Machine disk image...... [OK]

Defining new virtual machine...... [OK]

虚拟机 用户为root 密码为123456

###################################################

真机管理虚拟机

1.直接管理虚拟机,无需IP地址(将"虚拟系统管理软件"关闭)

[student@room9pc01 ~]$ virsh console A

退出: Ctrl + ]

[student@room9pc01 ~]$ virsh console B

退出: Ctrl + ]

2.配置IP地址:

虚拟机A

1.为eth0配置IP地址:192.168.4.7/24

2.配置主机名:svr7.tedu.cn

虚拟机B

1.为eth0配置IP地址:192.168.4.207/24

2.配置主机名:pc207.tedu.cn

[root@localhost /]# echo pc207.tedu.cn > /etc/hostname

[root@localhost /]# cat /etc/hostname

pc207.tedu.cn

[root@localhost /]# hostname

localhost.localdomain

[root@localhost /]# hostname pc207.tedu.cn

####################################################

[root@svr7 ~]# setip

Network name(eth0/eth1/eth2/eth3):eth0

Set IP(IP/24):192.168.4.7/24

Set Gateway(default none):

[root@svr7 ~]# ifconfig | head -2

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 192.168.4.7 netmask 255.255.255.0 broadcast 192.168.4.255

[root@svr7 ~]#

###################################################

真机设置别名

[student@room9pc01 ~]$ vim /home/student/.bashrc

alias goa='ssh -X root@192.168.4.7'

alias gob='ssh -X root@192.168.4.207'

[student@room9pc01 ~]$ exit

新开一个终端,验证goa与gob

##################################################

虚拟机查看文件内容

[root@svr7 ~]# cat /etc/motd

###################################################

为虚拟机A与虚拟机B构建Yum仓库

服务端:真机上构建ftp服务，共享光盘所有内容

1.安装vsftpd软件包

]$ rpm -q vsftpd

vsftpd-3.0.2-22.el7.x86\_64

]$ systemctl status vsftpd #查看服务状态

2.创建在/var/ftp/centos-1804

FTP服务默认共享:/var/ftp/

]$ ls /var/ftp/

]$ ls /var/ftp/centos-1804/

]$ firefox ftp://192.168.4.254/centos-1804/

3.开机自动挂载:

]$ vim /etc/fstab

/var/lib/libvirt/images/iso/CentOS7-1804.iso /var/ftp/centos-1804 iso9660 defaults 0 0

客户端:虚拟机A与虚拟机B /etc/yum.repos.d/\*.repo

[root@svr7 ~]# cat /etc/yum.repos.d/local.repo

[local\_repo]

name=CentOS-$releasever - Base

baseurl=ftp://192.168.4.254/centos-1804

enabled=1

gpgcheck=0

[root@svr7 ~]# yum -y install httpd

[root@pc207 ~]# cat /etc/yum.repos.d/local.repo

[local\_repo]

name=CentOS-$releasever - Base

baseurl=ftp://192.168.4.254/centos-1804

enabled=1

gpgcheck=0

[root@pc207 ~]# yum -y install httpd

###################################################

##################################################

补充:解决远程管理student用户虚拟机关闭问题

[student@room9pc01 ~]$ vim /home/student/.bash\_logout

里面的内容删除:/var/lib/libvirt/images/virt

####################################################

虚拟机C

1.为eth0配置IP地址:192.168.4.10/24

2.配置主机名:C.tedu.cn

3.使用真机FTP构建Yum仓库

虚拟机D

1.为eth0配置IP地址:192.168.4.20/24

2.配置主机名:D.tedu.cn

3.使用真机FTP构建Yum仓库

####################################################

扩展的几个应用

目录结构

• 认识Linux的目录层次:

– man hier

• 常见一级目录的用途

/boot 存放系统引导必需的文件,包括内核、启动配置

/bin、/sbin 存放各种命令程序

/dev 存放硬盘、键盘、鼠标、光驱等各种设备文件

/etc 存放Linux系统及各种程序的配置文件

/root、/home/用户名 分别是管理员root、普通用户的默认家目录

/var 存放日志文件、邮箱目录等经常变化的文件

/proc 存放内存中的映射数据,不占用磁盘

/tmp 存放系统运行过程中使用的一些临时文件

[root@svr7 ~]# ls -ld /tmp/

##################################################

权限的数值表示

• 权限的数值化

– 基本权限:r = 4,w = 2,x = 1

– 附加权限:SUID = 4,SGID = 2,Sticky Bit = 1

• 采用数值形式设置权限

– chmod [-R] nnn 文档...

– chmod [-R] xnnn 文档...

[root@svr7 ~]# mkdir /nsd01

[root@svr7 ~]# ls -ld /nsd01

[root@svr7 ~]# chmod 750 /nsd01

[root@svr7 ~]# ls -ld /nsd01

[root@svr7 ~]# chmod 700 /nsd01

[root@svr7 ~]# ls -ld /nsd01

[root@svr7 ~]# chmod 007 /nsd01

[root@svr7 ~]# ls -ld /nsd01

[root@svr7 ~]# chmod 077 /nsd01

[root@svr7 ~]# ls -ld /nsd01

###################################################

历史命令

• 管理/调用曾经执行过的命令

– history:查看历史命令列表

– history -c:清空历史命令

– !n:执行命令历史中的第n条命令

– !str:执行最近一次以str开头的历史命令

• 调整历史命令的数量

[root@svr7 ~]# vim /etc/profile

HISTSIZE=1000 //默认记录1000条

[root@svr7 ~]# history #查看历史命令

[root@svr7 ~]# history -c #清空历史命令

[root@svr7 ~]# history

1 history

[root@svr7 ~]#

[root@svr7 ~]# cat /etc/redhat-release

[root@svr7 ~]# ls /root/

[root@svr7 ~]# cat /etc/hostname

[root@svr7 ~]# history

[root@svr7 ~]# !cat #执行历史命令中,最近一条以cat开头

##################################################

• du,统计文件的占用空间

– du [选项]... [目录或文件]...

– -s:只统计每个参数所占用的总空间大小

– -h:提供易读容量单位(K、M等)

[root@svr7 ~]# du -sh /root

[root@svr7 ~]# du -sh /etc/

[root@svr7 ~]# du -sh /

[root@svr7 ~]# du -sh /boot

• date,查看/调整系统日期时间

– date +%F、date +%R

– date +"%Y-%m-%d %H:%M:%S"

– date -s "yyyy-mm-dd HH:MM:SS"

[root@svr7 ~]# date

[root@svr7 ~]# date +%Y #显示年

2019

[root@svr7 ~]# date +%m #显示月

04

[root@svr7 ~]# date +%d #显示日

19

[root@svr7 ~]# date +%H #显示 时

14

[root@svr7 ~]# date +%M #显示 分

08

[root@svr7 ~]# date +%S #显示 秒

[root@svr7 ~]# date +%F #显示 年-月-日

##################################################

制造快捷方式:

1.软连接

ln -s /路径/源文件 /路径/快捷方式名

若原始文件或目录被删除,连接文件将失效

软连接可存放在不同分区/文件系统

[root@svr7 ~]# cat /etc/redhat-release

[root@svr7 ~]# ln -s /etc/redhat-release /abc

[root@svr7 ~]# ls /

[root@svr7 ~]# ls -l /abc

[root@svr7 ~]# cat /abc

[root@svr7 ~]# ln -s /etc/redhat-release /

[root@svr7 ~]# ls /

2.硬连接

ln /路径/源文件 /路径/快捷方式名

若原始文件被删除,连接文件仍可用

硬连接与原始文件必须在同一分区/文件系统

[root@svr7 ~]# ls /opt/

[root@svr7 ~]# echo 123 > /opt/A.txt

[root@svr7 ~]# ln /opt/A.txt /opt/B

[root@svr7 ~]# ls /opt/

[root@svr7 ~]# cat /opt/A.txt

[root@svr7 ~]# cat /opt/B

[root@svr7 ~]# rm -rf /opt/A.txt

[root@svr7 ~]# ls /opt/

[root@svr7 ~]# cat /opt/B

###################################################

获取命令帮助

1.利用--help选项

[root@svr7 ~]# date --help

2.利用man命令

[root@svr7 ~]# man date

[root@svr7 ~]# man 5 passwd #5表示文件类型的帮助信息

#################################################

zip归档工具,跨平台的压缩归档格式

• 归档+压缩操作

– zip [-r] 备份文件.zip 被归档的文档...

[root@svr7 ~]# yum -y install zip

[root@svr7 ~]# zip -r /opt/test.zip /etc/passwd /home/

[root@svr7 ~]# ls /opt

[root@svr7 ~]# zip -r /opt/nb.zip /etc/fstab

[root@svr7 ~]# ls /opt

• 释放归档+解压操作

– unzip 备份文件.zip [-d 目标文件夹]

[root@svr7 ~]# yum -y install unzip

[root@svr7 ~]# mkdir /nsd02

[root@svr7 ~]# unzip /opt/test.zip -d /nsd02/

[root@svr7 ~]# ls /nsd02

[root@svr7 ~]# ls /nsd02/etc/

####################################################

将FTP服务器上,RHCE测试脚本

desktop0.sh rht-labcheck server0.sh

下载到本地/home/student/bin/赋予执行权限

[student@room9pc01 ~]$ chmod -R +x /home/student/bin/

[student@room9pc01 ~]$ ls /home/student/bin/

clone-vm7 desktop0.sh rht-labcheck server0.sh

[student@room9pc01 ~]$

##################################################

自定义Yum仓库

1.真机:具备从互联网下载的软件包

真机:访问FTP服务将 tools.tar.gz 下载真机的桌面上

2.将真机的tools.tar.gz包,上传到虚拟机A的/root目录下

scp依赖于ssh, scp=ssh + cp

scp /本地路径/源文件 root@对方的IP地址:/对方的路径

[student@room9pc01 ~]$ scp /home/student/桌面/tools.tar.gz root@192.168.4.7:/root/

[student@room9pc01 ~]$ goa

[root@svr7 ~]# ls /root/

Desktop tools.tar.gz

[root@svr7 ~]#

3.将tar进行解包

[root@svr7 ~]# tar -xf /root/tools.tar.gz -C /

[root@svr7 ~]# ls /

[root@svr7 ~]# ls /tools/

[root@svr7 ~]# ls /tools/other/

4.仓库数据文件

[root@svr7 ~]# yum -y install createrepo

[root@svr7 ~]# createrepo /tools/other/

[root@svr7 ~]# ls /tools/other/

[root@svr7 ~]# ls /tools/other/repodata/

5.修改配置文件

[root@svr7 ~]# cd /etc/yum.repos.d/

[root@svr7 yum.repos.d]# ls

[root@svr7 yum.repos.d]# vim local.repo

[local\_repo]

name=CentOS-$releasever - Base

baseurl=ftp://192.168.4.254/centos-1804

enabled=1

gpgcheck=0

[other]

name=my rpm

baseurl=file:///tools/other #file://表示本机为服务端

enabled=1

gpgcheck=0

[root@svr7 yum.repos.d]# yum repolist

软件的安装:

[root@svr7 /]# yum -y install cmatrix

[root@svr7 /]# rpm -ql cmatrix #显示包的安装清单

[root@svr7 /]# ls /usr/bin/cmatrix

/usr/bin/cmatrix

[root@svr7 /]# cmatrix

[root@svr7 /]# cmatrix -C red

[root@svr7 /]# yum -y install oneko

[root@svr7 /]# oneko

[root@svr7 /]# oneko & #将程序放入后台

[root@svr7 /]# yum -y install sl

[root@svr7 /]# sl

##################################################

vim编辑技巧

vi :Linux基本的文本编辑

命令模式操作

[root@svr7 ~]# cp /etc/passwd /opt/pass.txt

[root@svr7 ~]# vim /opt/pass.txt

光标跳转

上、下、左、右

Home 键 或 ^、数字 0 跳转到行首

End 键 或“$”键 跳转到行尾

PgUp 键、PgDn 键 向上翻页、向下翻页

1G 或 gg 跳转到全文的首行

G 跳转到全文的末尾行

复制/粘贴/删除

复制 yy、#yy 复制光标处的一行、#行

粘贴 p、P 粘贴到光标处之后、之前

x 或 Delete键 删除光标处的单个字符

dd、#dd 删除光标处的一行、#行

d^ 从光标处之前删除至行首

d$或D(大写) 从光标处删除到行尾

C(大写) 从光标处删除到行尾,进入插入模式

查找/撤销/保存

/root 向后查找字符串“root”

n、N 跳至后/前一个结果

u 撤销最近的一次操作

U 撤销对当前行的所有修改

Ctrl + r 取消前一次撤销操作(反撤销)

ZZ(大写) 保存修改并退出

####################################################

末行模式操作

读入其他文件内容

[root@svr7 ~]# echo 123 > /opt/1.txt

[root@svr7 ~]# echo abc > /opt/2.txt

[root@svr7 ~]# vim /opt/1.txt

:r /opt/2.txt #读入2.txt文件内容到当前文件

:r /etc/passwd #读入/etc/passwd文件内容到当前文件

字符串替换

:s /root/admin 替换光标所在的当前行第一个“root”

:s /root/admin/g 替换光标所在的当前行所有的“root”

:1,10 s/root/admin/g 替换第1-10行所有的“root”

:% s/root/admin/g 替换文件内所有的“root”

[root@svr7 ~]# vim /opt/pass.txt

vim /root/.vimrc

set number #自带行号

set tabstop=4 #修改tab默认字节

set autoindent #自动缩进

set ignorecase #忽略大小写

set smartcase

开关参数的控制

:set nu或nonu 显示/不显示行号

:set ai或noai 启用/关闭自动缩进

###################################################

小阶段3

day02

新教学环境的搭建

vim /boot/grub2/grub.cfg

console=tty0 console=ttyS1,115200n8 #102行添加后可用virsh console连接

dmidecode 查看硬件信息可判断是否是虚拟机

一 构建两台虚拟机

真机上运行:

虚拟机A

1.为eth0配置IP地址:192.168.4.7/24

2.配置主机名:svr7.tedu.cn

虚拟机B

1.为eth0配置IP地址:192.168.4.207/24

2.配置主机名:pc207.tedu.cn

##########################################################

真机:

root密码: tedu.com

##########################################################

U盘读取:

文件系统:FAT 一般可以直接识别

扩展FAT文件系统

文件系统:NTFS

[root@room9pc01 ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sda 8:0 0 465.8G 0 disk

├─sda1 8:1 0 297.7G 0 part /var/lib/libvirt/ima

└─sda2 8:2 0 100G 0 part /

sdb 8:16 0 931.5G 0 disk

└─sdb1 8:17 0 931.5G 0 part

loop0 7:0 0 8.8G 0 loop /var/ftp/centos-1804

[root@room9pc01 ~]#

[root@room9pc01 ~]# mount.ntfs-3g /dev/sdb1 挂载点

########################################################

真机远程管理虚拟机A

KVM构建及管理

虚拟化概述

• virtualization 资源管理

– x个物理资源 --> y个逻辑资源

– 实现程度:完全、部分、硬件辅助(CPU)

• 虚拟化主要厂商及产品

VMware VMware Workstation、vSphere

Microsoft VirtualPC、Hyper-V

RedHat KVM、RHEV

Citrix Xen

Oracle Oracle VM VirtualBox

安装虚拟化服务器平台

在虚拟机A:

• 主要软件包

– qemu-kvm :为 kvm 提供底层仿真支持

– libvirt-daemon :libvirtd 守护进程,管理虚拟机

– libvirt-client :用户端软件,提供客户端管理命令

– libvirt-daemon-driver-qemu :libvirtd 连接 qemu 的驱动

– virt-manager :图形管理工具

#######################################################

虚拟化服务:libvirtd

[root@svr7 ~]# systemctl restart libvirtd

[root@svr7 ~]# systemctl enable libvirtd

[root@svr7 ~]# virt-manager #运行图形管理虚拟化工具

[root@svr7 ~]# LANG=en virt-manager #以英文开启该工具

########################################################

使用虚拟化功能,需要CPU支持虚拟化功能

CPU支持虚拟化功能,有可能关闭

开启----->BIOS中进行开启

########################################################

连接本地KVM

• 使用 virt-manager 客户端工具

– 应用程序 --> 系统工具 --> 虚拟系统管理器

– 默认连本机的 libvirtd 服务

真机上操作:

[student@room9pc01 ~]$ virt-manager

virsh控制工具:

]$ virsh console 虚拟机名字 #真机直接管理相应虚拟机

]$ virsh

欢迎使用 virsh，虚拟化的交互式终端。

输入：'help' 来获得命令的帮助信息

'quit' 退出

virsh # console A

virsh nodeinfo :查看KVM节点(服务器)信息

virsh list [--all] :列出虚拟机

virsh net-list [--all] :列出虚拟网络

virsh dominfo 虚拟机名称 :查看指定虚拟机的信息

]$ virsh dominfo win2008

]$ virsh autostart win2008 #将虚拟机设置为开机自启

域 win2008标记为自动开始

]$ virsh dominfo win2008

]$ virsh autostart --disable win2008 #将虚拟机设置为开机不自启

域 win2008取消标记为自动开始

]$ virsh dominfo win2008

• 运行|重启|关闭指定的虚拟机

– virsh start|reboot|shutdown 虚拟机名称

• 强制关闭指定的虚拟机

– virsh destroy 虚拟机名称

[student@room9pc01 ~]$ virsh list --all

[student@room9pc01 ~]$ virsh start B

[student@room9pc01 ~]$ virsh list --all

[student@room9pc01 ~]$ virsh shutdown B

域 B 被关闭

[student@room9pc01 ~]$ virsh list --all

[student@room9pc01 ~]$ virsh start B

[student@room9pc01 ~]$ virsh list

[student@room9pc01 ~]$ virsh destroy B

域 B 被删除

[student@room9pc01 ~]$ virsh list --all

##########################################################

• 一台KVM虚拟机的组成

– xml配置文件(虚拟机配置文件):定义虚拟机的名称、UUID、CPU、内

存、虚拟磁盘、网卡等各种参数设置

[student@room9pc01 ~]$ ls /etc/libvirt/qemu/

– 磁盘镜像文件:保存虚拟机的操作系统及文档数据,

镜像路径取决于xml配置文件中的定义

[student@room9pc01 ~]$ ls /var/lib/libvirt/images/

导出xml配置文件

– 查看:virsh dumpxml 虚拟机名

– 备份:virsh dumpxml 虚拟机名 > 虚拟机名.xml

虚拟机的名称、UUID、虚拟磁盘文件路径、网卡唯一编号(MAC地址)

#########################################################

一 新建一台全新的虚拟机,虚拟机名字nsd01

虚拟机 :

内存为 2048M 硬盘 9G 虚拟机名字nsd01 网络类型:private1

软件包选择: 最小化安装 分区选择:自动分区 root密码设置为:1

二 利用,虚拟机nsd01作为模板,手动快速生成一台全新的虚拟机nsd02

1.将虚拟机nsd01的磁盘文件复制一份,修改磁盘文件的名称

[student@room9pc01 /]$ su - root

密码： teacher.com

[root@room9pc01 ~]#

]# virsh destroy nsd01 #强制关闭虚拟机nsd01

]# virsh list --all

]# cd /var/lib/libvirt/images/

]# du -sh nsd01.qcow2 #计算磁盘文件大小

]# cd /var/lib/libvirt/images/

]# cp nsd01.qcow2 nsd02.qcow2 #复制磁盘文件

]# du -sh nsd02.qcow2

2.将虚拟机nsd01的xml文件复制一份,修改内容

]# mkdir /xmlfile

]# virsh dumpxml nsd01 > /xmlfile/haha.xml #导出虚拟机xml文件

]# ls /xmlfile/

haha.xml

]# vim /xmlfile/haha.xml

修改虚拟机的名称: <name>nsd02</name>

UUID: <uuid></uuid> #删除整行内容,按u撤销

虚拟磁盘文件路径: file='/var/lib/libvirt/images/nsd02.qcow2'

网卡唯一编号(MAC地址) :

<mac address='52:54:00:32:a3:aa'/> #删除整行内容,按u撤销

3.导入虚拟机信息

]# virsh define /xmlfile/haha.xml #导入虚拟机信息

定义域 nsd02（从 /xmlfile/haha.xml）

]# virsh list --all

]# virt-manager

4.删除的命令

]# virsh list --all

]# virsh destroy nsd02 #关机

]# virsh undefine nsd02 #删除虚拟机nsd02,仅会删除/etc下xml文件

]# virsh list --all

]# ls /etc/libvirt/qemu #被删除

]# ls /xmlfile/ #存在

]# ls /var/lib/libvirt/images/nsd02.qcow2 #默认保留磁盘文件

###########################################################

以nsd01虚拟机为模板,制作虚拟机nsd03

1.复制nsd01虚拟机的磁盘文件

[root@room9pc01 /]# cd /var/lib/libvirt/images/

[root@room9pc01 images]# cp nsd01.qcow2 nsd03.qcow2

[root@room9pc01 images]# ls

2.修改nsd01的xml文件

[root@room9pc01 /]# virsh dumpxml nsd01 > /xmlfile/nsd03.xml

[root@room9pc01 /]# ls /xmlfile/

[root@room9pc01 /]# vim /xmlfile/nsd03.xml

修改虚拟机的名称: <name>nsd03</name>

UUID: <uuid></uuid> #删除整行内容,按u撤销

虚拟磁盘文件路径: file='/var/lib/libvirt/images/nsd03.qcow2'

网卡唯一编号(MAC地址) :

<mac address='52:54:00:32:a3:aa'/> #删除整行内容,按u撤销

[root@room9pc01 /]# virsh define /xmlfile/nsd03.xml

[root@room9pc01 /]# virsh list --all

########################################################

编辑虚拟机设置

• 对虚拟机的配置进行调整

– 编辑:virsh edit 虚拟机名

以nsd01虚拟机为模板,制作虚拟机nsd04

1.复制nsd01虚拟机的磁盘文件

[root@room9pc01 /]# cd /var/lib/libvirt/images/

[root@room9pc01 images]# cp nsd01.qcow2 nsd04.qcow2

[root@room9pc01 images]# ls

[root@room9pc01 /]# virsh list --all

[root@room9pc01 /]# virsh edit nsd01

修改虚拟机的名称: <name>nsd04</name>

UUID: <uuid></uuid> #删除整行内容,按u撤销

虚拟磁盘文件路径: file='/var/lib/libvirt/images/nsd04.qcow2'

网卡唯一编号(MAC地址) :

<mac address='52:54:00:32:a3:aa'/> #删除整行内容,按u撤销

[root@room9pc01 /]# virsh list --all

##########################################################

COW技术原理

• Copy On Write,写时复制

– 前端盘直接映射原始盘的数据内容

– 当前端盘的旧数据有修改时,在修改之前自动将旧数据存入前端盘

– 对前端盘的修改不回写到原始

• qemu-img 通过 -b 选项复用指定后端盘

– qemu-img create -f qcow2 -b 后端盘 前端盘

– qemu-img info 磁盘文件 #查看磁盘文件的信息

– qemu-img create 磁盘文件 #快速产生一个任意大小的磁盘文件,内容为空

1.查看nsd01.qcow2磁盘文件的信息

]# qemu-img info /var/lib/libvirt/images/nsd01.qcow2

#######################################################

通过写时复制技术,生成虚拟机abc01

– qemu-img create -f qcow2 -b 原始盘 前端盘 前端盘大小

1.以nsd01虚拟机磁盘文件为原始盘,产生abc01虚拟机的前端盘

]# qemu-img create -f qcow2 -b /var/lib/libvirt/images/nsd01.qcow2 /var/lib/libvirt/images/abc01.qcow2 10G

]# qemu-img info /var/lib/libvirt/images/abc01.qcow2

前端盘大小最好大于原始盘

一旦基于原始盘制作出前端盘,原始盘不能做任何改变

2.以前端盘abc01.qcow2产生虚拟机abc01

[root@room9pc01 /]# virsh edit nsd01

修改虚拟机的名称: <name>abc01</name>

UUID: <uuid></uuid> #删除整行内容,按u撤销

虚拟磁盘文件路径: file='/var/lib/libvirt/images/abc01.qcow2'

网卡唯一编号(MAC地址) :

<mac address='52:54:00:32:a3:aa'/> #删除整行内容,按u撤销

[root@room9pc01 /]#

#########################################################

1.以nsd01虚拟机磁盘文件为原始盘,产生nb虚拟机的前端盘

]# qemu-img create -f qcow2 -b /var/lib/libvirt/images/nsd01.qcow2 /opt/nb.qcow2 10G

]# qemu-img info /opt/nb.qcow2

2.以前端盘nb.qcow2产生虚拟机nb

[root@room9pc01 /]# virsh dumpxml nsd01 > /xmlfile/nb.xml

[root@room9pc01 /]# ls /xmlfile/

[root@room9pc01 /]# vim /xmlfile/nb.xml

修改虚拟机的名称: <name>nb</name>

UUID: <uuid></uuid> #删除整行内容, 删错可以按u撤销

虚拟磁盘文件路径: file='/opt/nb.qcow2'

网卡唯一编号(MAC地址) :

<mac address='52:54:00:32:a3:aa'/> #删除整行内容

[root@room9pc01 /]# virsh define /xmlfile/nb.xml

[root@room9pc01 /]# virsh list --all

#########################################################

离线访问虚拟机的磁盘文件内容

• 基本用法

guestmount -a 虚拟机磁盘路径 -i /挂载点

[root@room9pc01 /]# virsh destroy nb

[root@room9pc01 /]# mkdir /mnt/disk

[root@room9pc01 /]# ls /mnt/disk/

[root@room9pc01 /]# guestmount -a /opt/nb.qcow2 -i /mnt/disk/

[root@room9pc01 /]# ls /mnt/disk/

###########################################################

教学环境

虚拟机A:

]# LANG=en growpart /dev/vda 1 #扩展/dev/vda磁盘的第一个分区空间

]# lsblk

]# blkid /dev/vda1

]# xfs\_growfs /dev/vda1 #扩展/dev/vda1文件系统

]# df -h

#########################################################

小阶段3

day03

新教学环境的搭建

一 构建两台虚拟机

真机上运行:

虚拟机A

1.为eth0配置IP地址:192.168.4.7/24

2.配置主机名:svr7.tedu.cn

虚拟机B

1.为eth0配置IP地址:192.168.4.207/24

2.配置主机名:pc207.tedu.cn

##########################################################

DNS服务基础

• DNS服务器的功能

– 正向解析:根据注册的域名查找其对应的IP地址

– 反向解析:根据IP地址查找对应的注册域名,不常用

一级DNS服务器 二级DNS服务器 三级DNS服务器

所有有的域名都必须以点结尾

www.qq.com. www.baidu.com.

根域: . 点

一级域名: .cn .us .tw .hk .kr .jp

二级域名: .com.cn .net.cn .org.cn .......

三级域名: haxi.com.cn nb.com.cn dc.com.cn .......

• Full Qualified Domain Name(FQDN),完全合格主机名

主机头部,正确申请的域名结尾

vip.nb.com.cn www.nb.com.cn tts.nb.com.cn

####################################################

搭建Web服务器,此网站的FQDN为www.sina.com

####################################################

• BIND服务器端程序

– 主要执行程序:/usr/sbin/named

– 系统服务:named

– 默认端口:TCP/UDP 53

– 运行时的虚拟根环境:/var/named/chroot/

• 主配置文件:/etc/named.conf #设置本机负责解析的域名 tedu.cn

• 地址库文件:/var/named/ #完全合格的域名与IP地址的对应关系

##################################################

虚拟机A构建基本DNS服务

1.安装软件包 bind bind-chroot

bind-9.9.4-29.el7.x86\_64 //域名服务包

bind-chroot-9.9.4-29.el7.x86\_64 //提供虚拟根支持(牢笼政策)

2.修改主配置文件

[root@svr7 ~]# cp /etc/named.conf /etc/named.bak

[root@svr7 ~]# vim /etc/named.conf

options {

directory "/var/named"; #地址库文件存放位置

};

zone "tedu.cn" IN { #指定负责解析域名

type master; #指定为权威主服务器

file "tedu.cn.zone"; #指定地址库文件名称

};

3.建立相应的地址库文件

[root@svr7 ~]# cd /var/named

[root@svr7 named]# cp -p named.localhost tedu.cn.zone

-p:保持源文件权限不变进行复制

[root@svr7 named]# ls -l tedu.cn.zone

[root@svr7 named]# vim tedu.cn.zone

所有的域名都必须以点结尾,都要写最后的点

没有以点结尾,默认补全本地址库文件负责的域名

tedu.cn. NS svr7 #声明负责解析tedu.cn域名为svr7主机

svr7 A 192.168.4.7 #声明svr7主机的IP地址为192.168.4.7

www A 192.168.4.1 #正常解析记录

ftp A 192.168.4.2 #正常解析记录

4.重启named服务

[root@svr7 ~]# systemctl restart named

虚拟机B:

1.指定DNS服务器位置

/etc/resolv.conf #指定DNS服务器地址配置文件

]# echo nameserver 192.168.4.7 > /etc/resolv.conf

]# cat /etc/resolv.conf

2.测试DNS解析命令

]# nslookup www.tedu.cn

#########################################################

虚拟机A负责qq.com域名解析

1.修改主配置文件/etc/named.conf

......

zone "qq.com" IN {

type master;

file "qq.com.zone";

};

2.建立地址库文件

[root@svr7 /]# cd /var/named/

[root@svr7 named]# cp -p tedu.cn.zone qq.com.zone

[root@svr7 named]# vim qq.com.zone

qq.com. NS svr7

svr7 A 192.168.4.7

www A 1.1.1.1

ftp A 2.2.2.2

[root@svr7 /]# !sys

systemctl restart named

[root@svr7 /]#

#######################################################

特殊的解析记录

1.基于域名的负载均衡

[root@svr7 /]# vim /var/named/tedu.cn.zone

tedu.cn. NS svr7

svr7 A 192.168.4.7

www A 192.168.4.11

www A 192.168.4.12

www A 192.168.4.13

ftp A 192.168.4.2

[root@svr7 /]# systemctl restart named

2.泛域名解析

[root@svr7 /]# vim /var/named/tedu.cn.zone

tedu.cn. NS svr7

svr7 A 192.168.4.7

www A 192.168.4.11

ftp A 192.168.4.2

tedu.cn. A 10.20.30.40

\* A 1.2.3.4

[root@svr7 /]# !sys

systemctl restart named

[root@svr7 /]#

虚拟机B客户端测试:

nslookup hahahaha.tedu.cn

nslookup tedu.cn

3.有规律的泛域名解析

pc1.tedu.cn ------> 192.168.10.1

pc2.tedu.cn ------> 192.168.10.2

pc3.tedu.cn ------> 192.168.10.3

.......

pc50.tedu.cn ------> 192.168.10.50

内置函数: $GENERATE 起始-结束 #制造连续范围的数字

[root@svr7 ~]# vim /var/named/tedu.cn.zone

.......

$GENERATE 1-50 pc$ A 192.168.10.$

[root@svr7 ~]# !sys

systemctl restart named

[root@svr7 ~]#

4.解析记录的别名

[root@svr7 ~]# vim /var/named/tedu.cn.zone

.......

tts CNAME ftp

[root@svr7 ~]# !sys

systemctl restart named

虚拟机B客户端测试:

[root@pc207 ~]# nslookup tts.tedu.cn

Server: 192.168.4.7

Address: 192.168.4.7#53

tts.tedu.cn canonical name = ftp.tedu.cn.

Name: ftp.tedu.cn

Address: 192.168.4.2

[root@pc207 ~]#

#########################################################

DNS服务器的资源解析记录类型?

NS 声明DNS服务器记录

A 地址解析记录

CNAME 解析记录别名

##########################################################

/etc/resolv.conf #指定DNS服务器位置

客户端解析域名优先级

1./etc/hosts #直接提供解析结果,只为本机提供服务

2./etc/resolv.conf #寻找DNS服务器解析

[root@svr7 ~]# vim /etc/hosts

......

192.168.4.4 www.360.com

[root@svr7 ~]# ping www.360.com

#####################################################

DNS子域授权

父域:www.qq.com

子域:www.bj.qq.com

虚拟机A负责解析qq.com域名

虚拟机B负责解析bj.qq.com域名

虚拟机B:搭建为DNS服务器,负责解析的域名为bj.qq.com

[root@pc207 ~]# yum -y install bind bind-chroot

[root@pc207 ~]# cp /etc/named.conf /etc/named.bak

[root@pc207 ~]# vim /etc/named.conf

options {

directory "/var/named";

};

zone "bj.qq.com" IN {

type master;

file "bj.qq.com.zone";

};

[root@pc207 ~]# cd /var/named/

[root@pc207 named]# cp -p named.localhost bj.qq.com.zone

[root@pc207 named]# vim bj.qq.com.zone

bj.qq.com. NS pc207

pc207 A 192.168.4.207

www A 50.60.70.80

[root@pc207 /]# systemctl restart named

[root@pc207 /]# nslookup www.bj.qq.com 192.168.4.207

子域授权

虚拟机A负责解析qq.com域名

虚拟机B负责解析bj.qq.com域名

解析子域的域名,请求虚拟机A,可以解析

虚拟机A:

[root@svr7 ~]# vim /var/named/qq.com.zone

qq.com. NS svr7

bj.qq.com. NS pc207

svr7 A 192.168.4.7

pc207 A 192.168.4.207

www A 192.168.4.10

ftp A 2.2.2.2

[root@svr7 ~]# systemctl restart named

[root@pc207 /]# nslookup www.bj.qq.com 192.168.4.7

Server: 192.168.4.7

Address: 192.168.4.7#53

Non-authoritative answer: #非权威解答

Name: www.bj.qq.com

Address: 50.60.70.80

[root@pc207 /]#

递归解析:

接受客户端解析请求,本机与其他DNS服务器交流,最终将解析结果带回来的过程

[root@svr7 ~]# vim /etc/named.conf

options {

directory "/var/named";

recursion no;

};

迭代解析:本机与其他DNS服务器交流

#######################################################

验证迭代查询

• dig命令,更专业的DNS测试工具

– 由软件包 bind-utils 提供

– 格式: dig @DNS服务器 目标地址

[root@svr7 ~]# dig www.bj.qq.com 192.168.4.7

###################################################

缓存DNS,提高解析效率

虚拟机C

1.为eth0配置IP地址:192.168.4.10/24

2.配置主机名:C.tedu.cn

3.使用真机FTP构建Yum仓库

#####################################################

虚拟机A:真DNS服务器

虚拟机B上操作,搭建缓存DNS服务器

虚拟机C:客户端

#####################################################

虚拟机B上操作

[root@pc207 /]# vim /etc/named.conf

options {

directory "/var/named";

forwarders { 192.168.4.7; }; #转发给192.168.4.7

};

[root@pc207 /]# systemctl restart named

虚拟机C:做测试:

[root@C ~]# nslookup ftp.tedu.cn 192.168.4.207

Server: 192.168.4.207

Address: 192.168.4.207#53

Non-authoritative answer:

Name: ftp.tedu.cn

Address: 192.168.4.2

########################################################

小阶段3

day04

虚拟机A负责sina.com域名解析

1.修改主配置文件/etc/named.conf

......

zone "sina.com" IN {

type master;

file "sina.com.zone";

};

2.建立地址库文件

[root@svr7 /]# cd /var/named/

[root@svr7 named]# cp -p tedu.cn.zone sina.com.zone

[root@svr7 named]# vim sina.com.zone

sina.com. NS svr7

svr7 A 192.168.4.7

www A 1.1.1.1

[root@svr7 /]# !sys

systemctl restart named

##########################################################

Split分离解析(视图解析)

分离解析概述

什么是分离解析

• 当收到客户机的DNS查询请求的时候

– 能够区分客户机的来源地址

– 为不同类别的客户机提供不同的解析结果(IP地址)

– 为客户端提供最近服务器IP地址

BIND的view视图

• 根据源地址集合将客户机分类

– 不同客户机获得不同结果(待遇有差别)

view "联通" {

match-clients { 来源地址1; .. ..; };

zone "12306.cn" IN {

...... 地址库1;

}; };

view "铁通" {

match-clients { 来源地址2; .. .. ; };

zone "12306.cn" IN {

...... 地址库2;

}; };

###########################################################

注意事项:

1.由上到下进行,匹配及停止

2.分类要合理,每一个客户端都要找到自己的分类

3.所有的zone必须都在view中

view "nsd" {

match-clients { 192.168.4.1; }; #匹配客户端来源地址

zone "tedu.cn" IN {

......tedu.cn.zone;

};

};

view "haha" {

match-clients { any; }; #匹配客户端来源地址

zone "tedu.cn" IN {

...... tedu.cn.haha;

};

};

案例需求及要点

• 环境及需求

– 权威DNS:svr7.tedu.cn 192.168.4.7

– 负责区域:tedu.cn

– A记录分离解析 —— 以 www.tedu.cn 为例

客户机来自 解析结果

192.168.4.207------------> 192.168.4.100

其他地址------------> 1.2.3.4

虚拟机A操作:

[root@svr7 /]# vim /etc/named.conf

options {

directory "/var/named";

};

view "nsd" {

match-clients { 192.168.4.207; }; #匹配客户端来源地址

zone "tedu.cn" IN {

type master;

file "tedu.cn.zone"; ---->解析结果为 192.168.4.100

};

};

view "other" {

match-clients { any; }; #匹配客户端来源地址

zone "tedu.cn" IN {

type master;

file "tedu.cn.other"; ---->解析结果为 1.2.3.4

};

};

##########################################################

案例需求及要点

• 环境及需求

– 权威DNS:192.168.4.7

– 负责区域: tedu.cn sina.com

– A记录分离解析 —— 以 www.tedu.cn www.sina.com 为例

客户机来自 解析结果

192.168.4.207-----www.tedu.cn-------> 192.168.4.100

其他地址-----www.tedu.cn------->1.2.3.4

192.168.4.207-----www.sina.com-------> 192.168.4.200

其他地址-----www.sina.com------->10.20.30.40

注意事项:每一个view中zone的个数保持一致

view "nsd" {

match-clients { 192.168.4.207; };

zone "tedu.cn" IN {

type master;

file "tedu.cn.zone";---->解析结果为 192.168.4.100

};

zone "sina.com" IN {

type master;

file "sina.com.zone";---->解析结果为 192.168.4.200

};

};

view "other" {

match-clients { any; };

zone "tedu.cn" IN {

type master;

file "tedu.cn.other"; ---->解析结果为 1.2.3.4

};

zone "sina.com" IN {

type master;

file "sina.com.other";---->解析结果为 10.20.30.40

};

};

####################################################

补充分离解析 acl test { ; ; };

acl test { 192.168.4.207; 192.168.4.1; 192.168.4.2; };

view "nsd" {

match-clients { test; };

..........

##################################################

RAID阵列概述

• 廉价冗余磁盘阵列

– Redundant Arrays of Inexpensive Disks

– 通过硬件/软件技术,将多个较小/低速的磁盘整合成一个大磁盘

– 阵列的价值:提升I/O效率、硬件级别的数据冗余

– 不同RAID级别的功能、特性各不相同

• RAID 0,条带模式

– 同一个文档分散存放在不同磁盘

– 并行写入以提高效率

– 至少需要两块磁盘组成

• RAID 1,镜像模式

– 一个文档复制成多份,分别写入不同磁盘

– 多份拷贝提高可靠性,效率无提升

– 至少需要两块磁盘组成

• RAID5,高性价比模式

– 相当于RAID0和RAID1的折中方案

– 需要至少一块磁盘的容量来存放校验数据

– 至少需要三块磁盘组成

• RAID6,高性价比/可靠模式

– 相当于扩展的RAID5阵列,提供2份独立校验方案

– 需要至少两块磁盘的容量来存放校验数据

– 至少需要四块磁盘组成

• RAID 0+1/RAID 1+0

– 整合RAID 0、RAID 1的优势

– 并行存取提高效率、镜像写入提高可靠性

– 至少需要四块磁盘组成

###################################################

进程管理

程序: 静态的代码 占用磁盘空间

进程: 动态的代码 占用 CPU 内存

进程整体也是树型结果

父进程与子进程 特殊进程:僵尸进程 孤儿进程

进程唯一编号: PID

###################################################

查看进程树

• pstree — Processes Tree

– 格式:pstree [选项] [PID或用户名]

• 常用命令选项

– -a:显示完整的命令行

– -p:列出对应PID编号

systemd:所有进程的父进程,上帝进程

[root@svr7 ~]# useradd lisi

[root@svr7 ~]# pstree #查看整个进程树信息

[root@svr7 ~]# pstree lisi

bash───vim

[root@svr7 ~]# pstree -p lisi #-p:列出对应PID编号

bash(31421)───vim(31454)

[root@svr7 ~]# pstree -ap lisi #-a:显示完整的命令行

bash,31421

└─vim,31454 a.txt

[root@svr7 ~]#

查看进程快照

• ps — Processes Snapshot

– 格式:ps [选项]...

• 常用命令选项

– aux:显示当前终端所有进程(a)、当前用户在所有

终端下的进程(x), 以用户格式输出(u)

– -elf:显示系统内所有进程(-e)、以长格式输出(-l)

信息、包括最完整的进程信息(-f)

• ps aux 操作

– 列出正在运行的所有进程

• ps -elf 操作

– 列出正在运行的所有进程,PPID:父进程的PID号

统计命令: wc -l 文件 #统计文件内容有多少行

[root@svr7 ~]# wc -l /etc/passwd

统计正在运行的进程有多少个?

[root@svr7 ~]# ps aux | wc -l

[root@svr7 ~]# ps -elf | wc -l

[root@svr7 ~]# ps aux

[root@svr7 ~]# ps -elf

请统计/etc目录下有多少个以".conf"结尾的文档?

[root@svr7 ~]# find /etc/ -name '\*.conf' | wc -l

153

[root@svr7 ~]#

[root@svr7 ~]# find /etc/ -name '\*tab' | wc -l

9

[root@svr7 ~]#

################################################

进程动态排名

• top 交互式工具

– 格式:top [-d 刷新秒数] [-U 用户名]

[root@svr7 ~]# top -d 1

按键盘 P(大写) 按CPU占用排序

按键盘 M(大写) 按内存占用排序

检索进程

• pgrep — Process Grep

– 用途:pgrep [选项]... 查询条件

• 常用命令选项

– -l:输出进程名,而不仅仅是 PID

– -U:检索指定用户的进程

– -x:精确匹配完整的进程名

[root@svr7 ~]# pgrep -l a

[root@svr7 ~]# pgrep -l crond

[root@svr7 ~]# pgrep -U lisi

[root@svr7 ~]# pgrep -lU lisi

[root@svr7 ~]# pgrep -x named

[root@svr7 ~]# pgrep -xl named

#################################################

控制进程

进程的前后台调度

• 正在运行放入后台:

在命令行末尾添加“&”符号,不占用当前终端

• Ctrl + z 组合键

– 挂起当前进程(暂停并转入后台)

• jobs 命令

– 查看后台任务列表

• fg 命令

– 将后台任务恢复到前台运行

• bg 命令

– 激活后台被挂起的任务

[root@svr7 ~]# sleep 800 & 　#正在运行放入后台

[root@svr7 ~]# jobs 　　＃查看后台进程信息

[root@svr7 ~]# sleep 900

^Z 　　#按键盘上Ctrl+z暂停放入后台

[2]+ 已停止 sleep 900

[root@svr7 ~]# jobs

[root@svr7 ~]# bg 2　　　＃让后台编号为２的进程　继续运行

[root@svr7 ~]# jobs

[root@svr7 ~]# fg 1　　　＃让后台编号为１的进程　恢复到前台

sleep 800

^C　　　　#按键盘上Ctrl+ｃ暂停放入后台

[root@svr7 ~]# jobs

[root@svr7 ~]# fg 2　　　＃让后台编号为２的进程　恢复到前台

sleep 900

^C

[root@svr7 ~]# jobs

################################################

杀死进程

• 干掉进程的不同方法

– Ctrl+c 组合键,中断当前命令程序

– kill [-9] PID... 、kill [-9] %后台任务编号

– killall [-9] 进程名...

– pkill 查找条件

[root@svr7 ~]# sleep 1000 &

[root@svr7 ~]# sleep 2000 &

[root@svr7 ~]# sleep 3000 &

[root@svr7 ~]# sleep 4000 &

[root@svr7 ~]# jobs

[root@svr7 ~]# jobs -l

[root@svr7 ~]# kill 3083

[root@svr7 ~]# jobs -l

[root@svr7 ~]# kill -9 3086

[root@svr7 ~]# jobs -l

[root@svr7 ~]# killall sleep

[root@svr7 ~]# jobs -l

################################################

杀死一个用户开启的所有进程（强制踢出一个用户）

[root@svr7 ~]# killall -9 -u lisi

################################################

日志管理

日志的功能

• 系统和程序的“日记本”

– 记录系统、程序运行中发生的各种事件

– 通过查看日志,了解及排除故障

– 信息安全控制的“依据”

• 由系统服务rsyslog统一记录/管理

　时间、地点、人物、发生何事

常见的日志文件

日志文件

　/var/log/messages 记录内核消息、各种服务的公共消息

　/var/log/dmesg 记录系统启动过程的各种消息

　/var/log/cron 记录与cron计划任务相关的消息

　/var/log/maillog 记录邮件收发相关的消息

　/var/log/secure 记录与访问限制相关的安全消息

日志分析

• 通用分析工具

– tail、tailf、less、grep等文本浏览/检索命令

　tailf：跟踪日志消息

[root@svr7 ~]# echo 　123 　> 　/opt/1.txt

[root@svr7 ~]# cat 　/opt/1.txt

123

[root@svr7 ~]# tailf　 /opt/1.txt

123

#################################################s

用户登录分析

• users、who、w 命令

– 查看已登录的用户信息,详细度不同

• last、lastb 命令

– 查看最近登录成功/失败的用户信息

[root@svr7 ~]# users

[root@svr7 ~]# who

[root@svr7 ~]# w

[root@svr7 ~]# last -2 #最近登录成功

[root@svr7 ~]# lastb -2　　#最近登录失败

###############################################

日志消息的优先级

• Linux内核定义的事件紧急程度

– 分为 0~7 共8种优先级别

– 其数值越小,表示对应事件越紧急/重要

0 EMERG（紧急） 会导致主机系统不可用的情况

1 ALERT（警告） 必须马上采取措施解决的问题

2 CRIT（严重） 比较严重的情况

3 ERR（错误） 运行出现错误

4 WARNING（提醒） 可能会影响系统功能的事件

5 NOTICE（注意） 不会影响系统但值得注意

6 INFO（信息） 一般信息

7 DEBUG（调试） 程序或系统调试信息等

####################################################

使用journalctl工具

• 提取由 systemd-journal 服务搜集的日志

– 主要包括内核/系统日志、服务日志

• 常见用法

– journalctl | grep 关键词

– journalctl -u 服务名 [-p 优先级]

– journalctl -n 消息条数

– journalctl --since="yyyy-mm-dd HH:MM:SS" --

until="yyyy-mm-dd HH:MM:SS"

[root@svr7 ~]# journalctl -u httpd

[root@svr7 ~]# journalctl -xe #显示服务最近的报错信息

#####################################################

systemd介绍

• Linux系统和服务管理器

– 是内核引导之后加载的第一个初始化进程(PID=1)

– 负责掌控整个Linux的运行/服务资源组合

• 一个更高效的系统&服务管理器

– 开机服务并行启动,各系统服务间的精确依赖

– 配置目录:/etc/systemd/system/

– 服务目录:/lib/systemd/system/

– 主要管理工具:systemctl

对于服务的管理

systemctl restart 服务名 #重起服务

systemctl start 服务名 #开启服务

systemctl stop 服务名 #停止服务

systemctl status 服务名 #查看服务当前的状态

systemctl enable 服务名 #设置服务开机自启动

systemctl disable 服务名 #设置服务不开机自启动

RHEL6 运行级别

0：关机

1：单用户模式（基本功能的实现，破解Linux密码）

2：多用户字符界面（不支持网络）

3：多用户字符界面（支持网络）服务器默认的运行级别

4：未定义

5：图形界面

6：重起

切换运行级别：init 5

RHEL7 运行模式

字符模式：multi-user.target

图形模式：graphical.target

[root@svr7 /]# ls -l /lib/systemd/system/

当前直接切换到字符模式

[root@svr7 /]# systemctl isolate multi-user.target

当前直接切换到图形模式

[root@svr7 /]# systemctl isolate graphical.target

查看每次开机默认进入模式

[root@svr7 /]# systemctl get-default

graphical.target

设置永久策略，每次开机自动进入graphical.target

# systemctl set-default multi-user.target

# reboot

#################################################

小阶段3

day05

部署DHCP服务器

DHCP概述及原理

• Dynamic Host Configuration Protocol

– 动态主机配置协议,由 IETF(Internet 网络工程师任

务小组)组织制定,用来简化主机地址分配管理

• 主要分配以下入网参数

– IP地址/子网掩码/广播地址

– 默认网关地址、DNS服务器地址

DHCP原理：　广播进行，先到先得，一个网络里只能有一个DHCP服务器

• DHCP地址分配的四次会话

– DISCOVERY --> OFFER --> REQUEST -->ACK

开机启动项:匹配即停止

1.本机硬盘 2.光驱设备 3.U盘 4.网络

虚拟机Ａ：

1.装软件包 dhcp

[root@svr7 /]# yum -y install dhcp

2.配置文件 /etc/dhcp/dhcpd.conf

vim末行模式 :r /usr/share/doc/dhcp\*/dhcpd.conf.example

subnet 192.168.4.0 netmask 255.255.255.0 { #分配的网段

range 192.168.4.100 192.168.4.200; #分配的IP范围

option domain-name-servers 192.168.4.7; #分配DNS服务器

option routers 192.168.4.254; #分配网关地址

default-lease-time 600;

max-lease-time 7200;

}

[root@svr7 /]# systemctl restart dhcpd

####################################################

网络装机概述

• 规模化:同时装配多台主机

• 自动化:装系统、配置各种服务

• 远程实现:不需要光盘、U盘等物理安装介质

什么是PXE网络

• PXE,Pre-boot eXecution Environment

– 预启动执行环境,在操作系统之前运行

– 可用于远程安装

PXE组件及过程分析

• 需要哪些服务组件

– DHCP服务,分配IP地址、定位引导程序

– TFTP服务,提供引导程序下载

– HTTP服务,提供yum安装源

• 客户机应具备的条件

– 网卡芯片必须支持PXE协议

– 主板支持从网卡启动

##################################################

一 部署DHCP服务,指定下一个服务器地址

[root@svr7 /]# vim /etc/dhcp/dhcpd.conf

......

next-server 192.168.4.7; #指定下一个服务器地址

filename "pxelinux.0"; #指定引导文件名称

}

[root@svr7 /]# systemctl restart dhcpd

pxelinux.0:网卡引导文件(安装说明书) 二进制文件

安装一个软件,默认产生的网卡引导文件名为pxelinux.0

##################################################

二 搭建tftp服务

tftp:简单的文件传输协议 默认端口为69

默认共享位置:/var/lib/tftpboot

1.安装软件包tftp-server

[root@svr7 /]# yum -y install tftp-server

2.重启tftp服务

[root@svr7 /]# systemctl restart tftp

3.部署pxelinux.0文件

]# yum provides \*/pxelinux.0 #查询那个包产生,该文件

]# yum -y install syslinux

]# rpm -ql syslinux | grep pxelinux.0 #查询软件包安装清单

]# cp /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/

]# ls /var/lib/tftpboot/

pxelinux.0文件--->/var/lib/tftpboot/pxelinux.cfg/default

4.部署菜单文件

[root@svr7 /]# poweroff

Connection to 192.168.4.7 closed by remote host.

Connection to 192.168.4.7 closed.

[student@room9pc01 ~]$

图形为虚拟机A,添加一个光驱设备,放入光盘内容

[student@room9pc01 ~]$ goa

Last login: Thu Apr 25 08:55:41 2019 from 192.168.4.254

[root@svr7 ~]# mount /dev/cdrom /mnt/

mount: /dev/sr0 写保护，将以只读方式挂载

[root@svr7 ~]# ls /mnt/

[root@svr7 ~]# mkdir /var/lib/tftpboot/pxelinux.cfg

[root@svr7 ~]# cp /mnt/isolinux/isolinux.cfg /var/lib/tftpboot/pxelinux.cfg/default

[root@svr7 ~]# ls /var/lib/tftpboot/pxelinux.cfg/

[root@svr7 ~]# ls -l /var/lib/tftpboot/pxelinux.cfg/

5.部署 图形模块(vesamenu.c32) 与 背景图片(splash.png)

[root@svr7 ~]# cp /mnt/isolinux/vesamenu.c32 /mnt/isolinux/splash.png /var/lib/tftpboot/

[root@svr7 ~]# ls /var/lib/tftpboot/

6.部署 启动内核(vmlinuz) 与 驱动程序(initrd.img)

[root@svr7 ~]# cp /mnt/isolinux/vmlinuz /mnt/isolinux/initrd.img /var/lib/tftpboot/

[root@svr7 ~]# ls /var/lib/tftpboot/

initrd.img pxelinux.cfg vesamenu.c32

pxelinux.0 splash.png vmlinuz

7.修改菜单文件内容

# vim /var/lib/tftpboot/pxelinux.cfg/default

1 default vesamenu.c32 #默认加载图形模块

2 timeout 600 #读秒时间60秒 1/10秒

10 menu background splash.png #指定背景图片

11 menu title NSD1903 PXE Server #修改标题

61 label linux

62 menu label ^Install CentOS 7 #屏幕显示

63 menu default #读秒结束默认选择

64 kernel vmlinuz #调用内核

65 append initrd=initrd.img #解压驱动程序

##################################################

8.检查服务启动

[root@svr7 ~]# systemctl restart dhcpd

[root@svr7 ~]# systemctl enable dhcpd

[root@svr7 ~]# systemctl restart tftp

[root@svr7 ~]# systemctl enable tftp

#################################################

总结:

1.DHCP服务-->IP地址 next-server filename

2.tftp服务-->pxelinux.0

3.pxelinux.0-->读取默认菜单文件

4.default-->图形模块 背景图片 启动内核 驱动程序

验证:

新建一台虚拟机,安装方式选择 "网络引导安装(PXE)"

网络类型选择"private1"

####################################################

三 构建Web服务,共享光盘所有内容

1.安装软件包httpd

[root@svr7 ~]# yum -y install httpd

2.重启httpd服务

[root@svr7 ~]# systemctl restart httpd

[root@svr7 ~]# systemctl enable httpd

3.共享光盘所有内容

[root@svr7 ~]# mkdir /var/www/html/centos

[root@svr7 ~]# ls /var/www/html/centos

[root@svr7 ~]# mount /dev/cdrom /var/www/html/centos

mount: /dev/sr0 写保护，将以只读方式挂载

[root@svr7 ~]# ls /var/www/html/centos

[root@svr7 ~]# firefox 192.168.4.7/centos

#####################################################

四 部署无人值守安装,生成应答文件

1.安装图形的system-config-kickstart工具,生成应答文件

[root@svr7 ~]# yum -y install system-config-kickstart

2.运行system-config-kickstart工具,进行配置

[root@svr7 ~]# LANG=en system-config-kickstart

首先查看软件包选择(Package Select),是否可以使用

需要Yum仓库的支持,必须要求Yum光盘仓库标识为 [development]

[root@svr7 ~]# vim /etc/yum.repos.d/local.repo

[development]

........

[root@svr7 ~]# LANG=en system-config-kickstart

[root@svr7 ~]# ls /root/

[root@svr7 ~]# vim /root/ks.cfg

3.利用Web服务,将应答文件ks.cfg,共享给客户端

[root@svr7 ~]# cp /root/ks.cfg /var/www/html/

[root@svr7 ~]# ls /var/www/html/

4.修改菜单文件,指定应答文件ks.cfg

]# vim /var/lib/tftpboot/pxelinux.cfg/default

.........

label linux

menu label ^Install CentOS 7

menu default

kernel vmlinuz

append initrd=initrd.img ks=http://192.168.4.7/ks.cfg

#####################################################

总结:

1.DHCP服务-->IP地址 next-server filename

2.tftp服务-->pxelinux.0

3.pxelinux.0-->读取默认菜单文件

4.default-->图形模块 背景图片 启动内核 驱动程序 应答文件

5.ks.cfg应答文件--> 分区 语言 键盘类型 装包

# Use network installation

url --url="http://192.168.4.7/centos"

验证:

新建一台虚拟机,安装方式选择 "网络引导安装(PXE)"

网络类型选择"private1"

内存必须2G以上

##################################################

小阶段3

day06

rsync同步操作

rsync同步操作

• 命令用法

– rsync [选项...] 源目录 目标目录

• 同步与复制的差异

– 复制:完全拷贝源到目标

– 同步:增量拷贝,只传输变化过的数据

同步控制

• rsync操作选项

– -n:测试同步过程,不做实际修改

– --delete:删除目标文件夹内多余的文档

– -a:归档模式,相当于-rlptgoD

– -v:显示详细操作信息

– -z:传输过程中启用压缩/解压

[root@svr7 ~]# mkdir /nsd1903 /nsd04

[root@svr7 ~]# cp /etc/passwd /etc/fstab /nsd1903/

[root@svr7 ~]# ls /nsd1903/

[root@svr7 ~]# rsync -av /nsd1903 /nsd04 #同步本身

[root@svr7 ~]# ls /nsd04

[root@svr7 ~]# rsync -av /nsd1903/ /nsd04 #同步内容

[root@svr7 ~]# ls /nsd04

[root@svr7 ~]# touch /nsd1903/1.txt

[root@svr7 ~]# ls /nsd1903/

[root@svr7 ~]# ls /nsd04

[root@svr7 ~]# rsync -av /nsd1903/ /nsd04

[root@svr7 ~]# ls /nsd04

[root@svr7 ~]# rsync -av --delete /nsd1903/ /nsd04/

[root@svr7 ~]# ls /nsd1903

[root@svr7 ~]# ls /nsd04

[root@svr7 ~]# touch /nsd04/xixi.txt

[root@svr7 ~]# ls /nsd1903

[root@svr7 ~]# ls /nsd04

[root@svr7 ~]# rsync -av --delete /nsd1903/ /nsd04/

[root@svr7 ~]# ls /nsd1903

[root@svr7 ~]# ls /nsd04

####################################################

远程同步

虚拟机A目录/nsd1903与虚拟机B目录/opt进行同步

rsync+SSH同步= rsync + scp

• 与远程的 SSH目录保持同步

– 下行:rsync [...] user@host:远程目录 本地目录

– 上行:rsync [...] 本地目录 user@host:远程目录

虚拟机A:

]# rsync -av --delete /nsd1903/ root@192.168.4.207:/opt/

虚拟机B

]# ls /opt

]# touch /opt/haxi.txt

]# ls /opt

虚拟机A:

]# rsync -av --delete /nsd1903/ root@192.168.4.207:/opt/

虚拟机B

]# ls /opt

###################################################

完成实时同步准备:安装inotify-tools工具

源码包编译安装

源码编译安装的优势

• 主要优点

– 获得软件的最新版,及时修复bug

– 软件功能可按需选择/定制,有更多软件可供选择

– 源码包适用各种平台

rpm包: rpm -ivh yum install

源码包:----gcc make开发工具---->可以执行的程序--->运行安装

1.真机上传tools.tar.gz包,到虚拟机A的/root

[student@room9pc01 ~]$ scp /home/student/桌面/tools.tar.gz root@192.168.4.7:/root/

2.验证是否上传成功

[root@svr7 ~]# ls /root/

[root@svr7 ~]# tar -xf /root/tools.tar.gz -C /opt

[root@svr7 ~]# ls /opt/tools/

inotify-tools-3.13.tar.gz other

步骤1:安装gcc make开发工具(安装依赖关系的软件)

[root@svr7 ~]# yum -y install gcc make

[root@svr7 ~]# rpm -q gcc

[root@svr7 ~]# rpm -q make

步骤2:tar解包,释放源代码至指定目录

]# tar -xf /opt/tools/inotify-tools-3.13.tar.gz -C /

]# ls /inotify-tools-3.13/

步骤3: ./configure 配置,指定安装目录/功能模块等选项

--prefix=指定安装的位置

]# cd /inotify-tools-3.13/

]# ./configure

#检测系统是否安装gcc,检测系统是否可以支持编译安装

常见提示:gcc没有安装

checking for gcc... no

checking for cc... no

checking for cl.exe... no

configure: error: no acceptable C compiler found in $PATH

See `config.log' for more details.

步骤4:make 编译,生成可执行的二进制程序文件

]# cd /inotify-tools-3.13/

]# make

步骤5:make install 安装,将编译好的文件复制到安装目录

]# cd /inotify-tools-3.13/

]# make install

]# ls /usr/local/bin/inotifywait

###################################################

实时同步:虚拟机A目录/nsd1903与虚拟机B目录/opt进行同步

一 实现ssh的无密码验证,生成公钥 私钥

1.虚拟机A生成公钥 私钥

[root@svr7 ~]# ssh-keygen #一路回车

[root@svr7 ~]# ls /root/.ssh/

authorized\_keys id\_rsa id\_rsa.pub known\_hosts

known\_hosts:记录曾经远程管理过的主机

2.虚拟机A传递公钥到虚拟机B上

]# ssh-copy-id root@192.168.4.207

]# rsync -av --delete /nsd1903/ root@192.168.4.207:/opt/

虚拟机B

[root@pc207 ~]# ls /root/.ssh/

authorized\_keys

[root@pc207 ~]#

####################################################

二 监控目录内容的变化

inotifywait监控

• 基本用法

– inotifywait [选项] 目标文件夹

• 常用命令选项

– -m,持续监控(捕获一个事件后不退出)

– -r,递归监控、包括子目录及文件

– -q,减少屏幕输出信息

– -e,指定监视的 modify、move、create、delete、attrib等事件类别

inotifywait -mrq /nsd1903

rsync -a --delete /nsd1903/ root@192.168.4.207:/opt/

三 书写shell脚本完成,实时同步操作

循环解决重复性操作

for循环:适合写有次数的循环

for i in 值

do

重复性操作

done

死循环:无限次的循环

while循环:适合写无限次的循环

while [ 1 -eq 1 ]

do

重复执行的操作

done

[root@svr7 ~]# vim /root/rsync.sh

#!/bin/bash

while inotifywait -rqq /nsd1903

do

rsync -a --delete /nsd1903/ root@192.168.4.207:/opt/

done

[root@svr7 ~]# /root/rsync.sh

####################################################

小阶段3

cobbler装机

Cobbler装机平台,不同版本的多系统的安装

一 具备一个CentOS虚拟机

[root@svr7 ~]# cat /etc/redhat-release

CentOS Linux release 7.5.1804 (Core)

二、虚拟机设置

1.设置防火墙为trusted

]# firewall-cmd --set-default-zone=trusted

2.当前及永久设置SELinux状态为permissive

]# setenforce 0 #当前临时关闭

]# getenforce

]# vim /etc/selinux/config

SELINUX=

三、利用scp真机传递cobbler.zip包到虚拟机192.168.4.7中

# scp /home/student/桌面/cobbler.zip root@192.168.4.7:/root/

##################################################

四、搭建Cobbler装机平台

Cobbler概述软件，管理dhcp、tftp、Web服务

自由的导入镜像与ks应答文件

1.解压cobbler.zip包

[root@svr7 ~]# yum -y install unzip

[root@svr7 ~]# unzip /root/cobbler.zip -d /

[root@svr7 ~]# ls /cobbler/

cobbler\_boot.tar.gz cobbler\_web.png

cobbler\_rpm.zip

[root@svr7 ~]#

[root@svr7 ~]# unzip /cobbler/cobbler\_rpm.zip -d /opt/

[root@svr7 ~]# ls /opt/cobbler/

#####################################################

五、安装cobbler主程序、工具包等

]# yum -y install /opt/cobbler/\*.rpm httpd dhcp mod\_ssl

[root@svr7 ~]# rpm -q cobbler

cobbler-2.8.2-1.el7.x86\_64

[root@svr7 ~]#

################################################################

cobbler网络装机部署

1.安装软件 cobbler cobbler-web dhcp tftp-server pykickstart httpd tftp-server

cobbler #cobbler程序包

cobbler-web #cobbler的web服务包

pykickstart #cobbler检查kickstart语法错误

httpd #Apache web服务

dhcp #dhcp服务

tftp-server #tftp服务

2.配置cobbler

[root@svr7 /]# vim /etc/cobbler/settings

next\_server: 192.168.4.7 #设置下一个服务器还为本机

server: 192.168.4.7 #设置本机为cobbler服务器

manage\_dhcp: 1 #设置cobbler管理dhcp服务

pxe\_just\_once: 1 #防止客户端重复安装操作系统

开机启动项： 匹配及停止

1.硬盘启动 2.光驱设备 3.U盘 4.网络引导

3.配置cobbler的dhcp

[root@Cobbler /]# vim /etc/cobbler/dhcp.template

:%s /旧/新/g #全文替换

:%s /192.168.1/192.168.4/g #全文替换

5 次替换，共 4 行

4.绝对路径解压cobbler\_boot.tar.gz #众多的引导文件

# tar -tf /cobbler/cobbler\_boot.tar.gz #查看包里面内容

# tar -xPf /cobbler/cobbler\_boot.tar.gz #绝对路径释放

# ls /var/lib/cobbler/loaders/

5.启动相关服务

]# systemctl restart cobblerd

]# systemctl enable cobblerd

]# systemctl restart httpd

]# systemctl enable httpd

]# systemctl restart tftp

]# systemctl enable tftp

]# systemctl restart rsyncd #同步服务

]# systemctl enable rsyncd

6.同步刷新cobbler配置

[root@cobbler /]# cobbler sync #检测所有配置是否正确

......

\*\*\* TASK COMPLETE \*\*\*

]# firefox https://192.168.4.7/cobbler\_web

用户名：cobbler

密码：cobbler

##########################################################################################################

cobbler应用

cobbler import --path=挂载点 --name=导入系统命名（随意起）

导入安装镜像数据

# mount /dev/cdrom /dvd

# ls /dvd/

# cobbler import --path=/dvd --name=CentOS7

cobbler导入的镜像放在：/var/www/cobbler/ks\_mirror

[root@cobbler /]# cobbler list #查看有哪些系统

distros:

CentOS7-x86\_64 #安装客户端至少2G内存

profiles:

CentOS7-x86\_64

##############################################

删除

# cobbler list

# cobbler profile remove --name=CentOS7-x86\_64 #删除菜单信息

# cobbler distro remove --name=CentOS7-x86\_64 #删除镜像信息

###########################################

# umount /dvd/

# mount /dev/cdrom /dvd 将光盘换成RHEL7

# ls /dvd/

安装完成机器 默认root 密码 cobbler

############################################################################################################

自定义应答文件：开头注释行删除

[root@cobbler ~]# system-config-kickstart #生成ks文件

必须默认kickstart文件存放位置：/var/lib/cobbler/kickstarts/

[root@cobbler ~]# cobbler list

修改kickstart文件：

[root@cobbler ~]# cobbler profile edit --name=CentOS7 --kickstart=/var/lib/cobbler/kickstarts/自定义.cfg

[root@cobbler ~]# cobbler profile report

[root@cobbler ~]# cobbler sync #同步配置

###########################################################################################################