NFS网络文件系统共享(Linux与Linux平台之间) 所需软件包:nfs-utils 系统服务:nfs-server

搭建基本的NFS共享 服务端：1.安装软件包nfs-utils 2.创建目录及文件

3.修改主配置文件 /etc/exports（/abc ##文件夹路径 \*(ro) ##客户机地址(权限)）（exportfs -r可以重新加载）(no\_root\_squash)

4.重起服务nfs-server

客户端：修改配置文件：1.查看对方挂载：showmount -e serverX

2.vim /etc/fstab（172.25.0.11:/abc /mnt/nsd01 nfs defaults,\_netdev 0 0）

2.mount -a df -h

安全的NFS：服务端 1.装包nfs-utils 2.创建共享目录 2.下载密钥并改名（wget -O /etc/krb5.keytab）

4.配置：vim /etc/exports（/protected 172.25.X.0/24(rw,sec=krb5p)） 5.重起nfs-secure-server nfs-server

客户端 1.查看对方挂载：showmount -e serverX 2.创建挂载点 3.下载密钥并改名（wget -O /etc/krb5.keytab）

4.重起服务：nfs-secure

5.配置：vim /etc/fstab（serverX.example.com:/protected /mnt/nfssecure nfs sec=krb5p,\_netdev 0 0）

6.mount -a

[root@svr7 /]# vim /root/rsync.sh

#!/bin/bash

while inotifywait -rqq /opt

do

rsync -az --delete /opt/ root@192.168.4.207:/opt/

done

Cobbler装机平台,不同版本的多系统的安装-----防火墙 与selinux

搭建cobbler:1.解压 2.配置（cobbler） vim /etc/cobbler/settings

next\_server: 192.168.4.168 #设置下一个服务器还为本机

server: 192.168.4.168 #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

3.配置cobbler的dhcp 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 -xPf /Cobbler/cobbler\_boot.tar.gz #绝对路径释放

5.启动相关服务重起并开机自动重起（cobblerd，httpd，tftp，rsyncd）

6.同步刷新cobbler配置：cobbler sync #检测配置是否正确（COMPLETE）

7.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

cobbler导入的镜像放在：/var/www/cobbler/ks\_mirror

cobbler list 查询镜像系统

删除 ：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

自定义应答文件：开头注释行删除

system-config-kickstart #生成ks文件 存放位置：/var/lib/cobbler/kickstarts/

修改kickstart文件：cobbler profile edit --name=CentOS7.4-A --kickstart=/var/lib/cobbler/

kickstarts/自定义.cfg

cobbler profile report

cobbler sync #同步配置

网络自动装机PXE yum provides \*/pxelinux.0 #查询仓库中那个软件包产生

1.http----->将镜像挂载到\*/html/rh7dvd/ system-config-kickstart

2.dhcp----->PXE引导服务器位于 next-server 192.168.4.7 filename 引导文件 pxelinux.0

3.tftp-server(tftp)--->图形支持：vesamenu.c32（镜像） 背景图片：splash.png（镜像）

syslinux---->pxelinux.0(yum provides \*/pxelinux.0)--->tftpboot

4.tftp-server(tftp)--->rhel7-->内核vmlinuz、初始化文件initrd.img（镜像isolinux）

5.tftp-server(tftp)--->pxelinux.cfg---->default(<----镜像/isolinux.cfg)

一、部署 DHCP服务 装软件包 dhcp• 配置文件 /etc/dhcp/dhcpd.conf • 起服务 dhcpd

1.安装dhcp软件包 2.配置vim /etc/dhcp/dhcpd.conf(：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IP范围

option domain-name-servers 192.168.4.7; #DNS服务器

option routers 192.168.4.254; #网关地址

default-lease-time 600;

max-lease-time 7200;

next-server 192.168.4.7; #指定下一个服务器地址

filename "pxelinux.0"; } #指定网卡引导文件名称

3.重起

二、搭建tftp服务

1.安装软件包tftp-server 2.重起tftp服务（简单的文件传输协议69 默认共享路径：/var/lib/tftpboot/）

部署pxelinux.0文件 1.下载syslinux 2.cp /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/

部署菜单文件 1.挂载 2.cp /mnt/isolinux/isolinux.cfg /var/lib/tftpboot/pxelinux.cfg/default

部署图形的模块vesamenu.c32 及背景图片splash.png

部署启动内核vmlinuz 与驱动程序 initrd.img

三、修改菜单文件 vim /var/lib/tftpboot/pxelinux.cfg/default

开启行号

1 default vesamenu.c32 #默认加载图形的模块

2 timeout 600 #读秒时间 1/10

10 menu background splash.png #指定背景图片

11 menu title NSD1808 PXE Server ！ #菜单显示的标题

61 label linux

62 menu label Install RHEL7.4 #选项内容

63 menu default #读秒结束后默认选择

64 kernel vmlinuz #加载内核

65 append initrd=initrd.img #加载驱动 ks=http://192.168.4.7/ks.cfg #指定ks应答文件

四、搭建httpd服务（ mount /dev/cdrom /var/www/html/rhel7）

五、实现无人值守安装，生成应答文件 1.安装一个图形的工具(system-config-kickstart)

2.运行（必须要求Yum仓库标识 [development]）

六、共享ks应答文件 （cp /root/ks.cfg /var/www/html/）(YUM源，源ID设为 development)

DNS服务器的功能 将域名解析成ip地址 – 正向解析 反向解析 完全合格主机名（FQDN）ServerName www.nb.com

软件包：bind与bind-chroot – 运行时的虚拟根环境:/var/named/chroot/ #牢笼政策

– 系统服务:named – 默认端口: 53

分离解析概述：1.能够区分客户机的来源地址 2.不同的客户端解析同一个域名时，解析结果不同

3.为客户端提供最近的资源

检查配置语法：named-checkconf /etc/named.conf[配置文件]

named-checkzone tedu.cn tedu.cn.zone [区域名][配置文件]

• 主配置文件: /etc/named.conf #设置本服务器负责解析的域名

options {

directory "/var/named"; ##地址文件的默认位置 13

recursion no; ##关闭递归查询 29

};

acl test { 192.168.4.207; 192.168.7.0/24; 192.168.10.1; };

view "nsd" {

match-clients { test; };

zone "tedu.cn" IN { ---->51

type master;

file "tedu.cn.zone";

};

zone "qq.com" IN {

type master;

file "qq.com.zone";

};

};

view "other" {

match-clients { any; };

zone "tedu.cn" IN {

type master;

file "tedu.cn.other";

};

zone "qq.com" IN {

type master;

file "qq.com.other";

};

};

• 地址库文件: /var/named/tedu.cn.zone #主机名与IP地址的对应关系 复制的时候机的cp -p

$TTL 1D

@ IN SOA @ rname.invalid. (

0 ; serial

1D ; refresh

1H ; retry

1W ; expire

3H ) ; minimum

qq.com. NS sur7

bj.qq.com. NS pc207

sur7 A 192.168.4.7

pc207 A 192.168.4.207

www A 1.1.1.1

tts CNAME www

主机映射记录文件/etc/hosts(起到DNS解析的功能) 只为本机提供解析功能

搭建DNS服务器: 1.递归解析 （递归查询是默认开启的，也可以设置 recursion yes; 来明确启用） 2.迭代解析

服务端： 1.安装 2.修改主配置文件/etc/named.conf 3.建立地址库文件/var/named/ 4.重起named服务

（1.搭建多区域的DNS 2.特殊的解析记录（轮询） 3.泛域名解析（最后一条记录 \*） 4.解析记录的别名 CNAME

5.有规律的泛域名解析 $GENERATE 1-50 pc$ A 192.168.20.$ 6.DNS子域授权）

缓存DNS服务器,缓存加速 方式1:全局转发 2:根域迭代

1.有DNS服务器 2.修改DNS主配置文件，指定缓存DNS服务器配置 （/etc/named.conf）

options {

directory "/var/named";

forwarders { 192.168.1.1; };

};

客户端：1.vim /etc/resolv.conf (nameserver 192.168.1.1) 2.nslookup www.bj.qq.com 192.168.4.7

搭建基本Web服务：BS架构 默认端口号80 防火墙trusted 检查语法:httpd -t

服务端:1.安装httpd(Apache)软件包 2.创建默认目录与文件/index.html

3.创建设置调用配置文件：/etc/httpd/conf.d/\*.conf

<virtualhost \*:80> #本机所有IP启用虚拟Web

servername www.qq.com #此站点的DNS名称

documentroot /webapp #此站点的网页根目录

wsgiscriptalias / /var/www/nsd01/webinfo.wsgi ##网页跳转

</virtualhost>

listen 8909 ## 主机侦听在端口8909

<virtualhost \*:8909>

servername www.163.com

documentroot /var/www/nsd02

directoryindex index ##起始页/首页文件名

</virtualhost>

4.创建设置权限目录：/etc/httpd/conf.d/\*.conf

<Directory /var/www/myweb/private>

Require ip 172.25.0.11(not ip 10.252.46.165)（all granted）(all denied)

</Directory>

5.使用自定义根目录：chcon [-R] --reference=模板目录 新目录 6.重起httpd服务

部署动态网站：1.安装mod\_wsgi软件包,专用于翻译Python代码页面

2.设置动态网站（调用配置文件）：WsgiScriptAlias / /var/www/nsd02/webinfo.wsgi 3.重起

修改主机侦听端口8909：1.修改调用配置文件Listen 8909 <VirtualHost \*:8909>

2.selinux非默认端口的开放：查semanage port -l | grep http #查

semanage port -a -t http\_port\_t -p tcp 8909 #改 -a:添加 -t：类型 -p：协议

3.重起并验证 ：firefox webapp0.example.com:8909

安全的Web服务器

服务端：1.安装支持安全https的软件mod\_ssl

2.部署网站证书（营业执照）（1） cd /etc/pki/tls/certs （2）wget 路径/server0.crt

3.部署根证书（公安局信息）（1） cd /etc/pki/tls/certs （2）wget 路径/example-ca.crt

4.部署私钥（用于解密） （1）cd /etc/pki/tls/private/ （2）wget 路径/server0.key

5.修改配置文件：vim /etc/httpd/conf.d/ssl.conf 修改100行 107行 122行（59行及60行）

6.书写一个网站用于测试 DocumentRoot "/var/www/html" 7.重起httpd服务

客户端 ：firefox https://www0.example.com

搭建FTP服务:文件数据传输

服务端 1.安装软件包vsftpd 2.重起vsftpd服务(默认共享数据路径：/var/ftp)

sed -i 's/^#anon/anon/' /etc/vsftpd/vsftpd.conf

匿名上传 #anon\_upload\_enable=YES

#anon\_mkdir\_write\_enable=YES

chmod 777 /var/ftp/pub

客户端 本机测试 firefox ftp://172.25.0.11

Yum软件包仓库，自动解决依赖关系

1.众多的软件包 2.仓库清单文件（repodata）3.构建Web服务或FTP服务传递数据

yum服务端：将镜像文件挂载到/var/ftp下，本机测试，开机自动重起

客户端配置文件：/etc/yum.repos.d/\*.repo（[] name baseurl enabled gpgcheck）

(clean all repolist -y install -y reinstall remove)

1.baseurl=ftp://192.168.4.254/rhel7

2.baseurl=http://classroom.example.com/content/rhel7.0/x86\_64/dvd/

自定义yum仓库

1.具备互联网下载的包，真机传递tools.tar.gz到虚拟机A

scp /root/桌面/tools.tar.gz root@192.168.4.7:/root

2.虚拟机A解tar包

3.运行createrepo生成仓库数据文件# createrepo /tools/other/

4.书写客户端配置文件 baseurl=file:///tools/other（指定本机）

源码编译安装：（需要删除直接删除目录下）步骤1: 安装gcc与make 步骤2: tar解包,释放源代码至指定目录

步骤3: （1）cd /inotify-tools-3.13/（2）./configure --prefix=/opt/myrpm #指定安装位置

作用（1）配置,指定安装目录/功能模块等选项 （2） 作用：检测本机是否安装gcc

步骤4: make 编译,生成可执行的二进制程序文件 步骤5 make install 安装,将编译好的文件复制到安装目录

zip归档工具,跨平台的压缩工具（1）归档+压缩操作– zip [-r] 备份文件.zip 被归档的文档...

（2） 释放归档+解压操作– unzip 备份文件.zip [-d 目标文件夹]

硬盘：扇区的大小为512字节 1PB=1000TB 1EB=1000PB

磁盘分类：hd,表示IDE设备 sd,表示SCSI设备 vd,表示虚拟化设备

1.识别硬盘lsblk 2.分区规划(fdisk -l /dev/vdb ：查看分区模式MBR/msdos最大2.2TB 或者 GPT最大到18EB)

fdisk /dev/vdb d删除 n新增 p查找 q不保存退出 w保存退出 刷新分区表：partprobe

3.格式化：mkfs.xfs /dev/vdb3 查看文件系统（格式化）：blkid /dev/vdb3

4.开机自动挂载：vim /etc/fstab

设备路径 挂载点 类型 参数 备份标记 检测顺序（可手动挂载）

检查及自动挂载：mount -a 查看已挂载磁盘分区信息：df -Th

parted /dev/vdb "mktable gpt" "mkpart" "ext2" "ext2" "0G" "1G"

parted大容量分区，专用于gpt分区方案

parted /dev/vdb #编辑 mklabel(mktable) gpt #指定分区模式为GPT print #查看分区表

mkpart #划分分区 忽略/Ignore/放弃/Cancel? Ignore #输入i选择忽略 unit GB # 使用GB作为显示单位

交换空间：以空闲分区充当的交换空间，相当于虚拟内存,当物理内存不够用时,使用磁盘空间来模拟内存，在一定程度上缓解内存不足的问题

mkswap /dev/vdb1 #格式化交换文件系统 swapon（swapoff ） /dev/vdb1 #启用（停用）的命令

swapon -s #查看交换空间的成员

开机自启：vim /etc/fstab （/dev/vdb1 swap swap defaults 0 0）

swapon -a #检测书写是否正确 swapon -s

LVM逻辑卷的管理: 将众多的物理卷( PV),组成卷组(VG),再从卷组中划分出逻辑卷(LV)

作用：1.可以整合分散的空间 2.容量大小可以扩大------------------>查看：pvs vgs（vgdisplay） lvs(can)

1.直接创建卷组:vgcreate 卷组的名 设备路径1 设备路径2 .......

2.通过卷组划分逻辑卷:lvcreate -n 逻辑卷名 -L 逻辑卷大小(300M)) 卷组名

3.格式化：mkfs.ext4 设备路径（blkid）

4.开机自动挂载： vim /etc/fstab 设备路径 挂载点 类型 参数 备份标记 检测顺序

检查及自动挂载：mount -a 查看已挂载磁盘分区信息：df -h(挂载点)

逻辑卷的扩展:1.扩展卷组vgextend 卷组名 设备路径 2.扩展空间的大小：lvextend -L（l） 大小(300M) 设备路径

3.扩展文件系统的大小 resize2fs 设备路径（扩展ext4文件系统） xfs\_growfs:扩展xfs文件系统

卷组划分空间的单位：PE（默认4M） 1.查看卷组PE大小 vgdisplay systemvg

2.修改方式 卷组存在：vgchange -s PE大小 卷组名---创建卷组时，可以通过-s选项指定PE的大小

创建卷组的时候设置PE大小：vgcreate -s PE大小 卷组名 设备路径

逻辑卷的删除：先卸载 逻辑卷（lvremove）--->卷组(vgremove)-------> 物理卷(pvremove)

Samba服务基础，Linux与Windows跨平台的共享 （协议:SMB(TCP 139 用户验证)、CIFS(TCP 445 传输数据)

Samba用户： 专用来访问共享文件夹的用户 1.采用独立设置的密码 2.需要提前建立同名的系统用户(可以不设密码）

搭建Samba共享：服务端 1.安装软件包samba（系统服务:smb）

2.创建samba共享帐号及目录/文件（useradd pdbedit -（a增L查x删））（/sbin/nologin）

3.SELinux布尔值ro（服务功能的开关on 或 off）加-P选项才能实现永久设置（查getsebool -a 改setsebool）及本地目录

4.修改配置文件/etc/samba/smb.conf （[common] #共享名 path = /common #共享实际路径）(注意空格) 5.重起smb服务

客户端：1.安装软件cifs-utils（samba-client：smbclient -L serverX -U 用户名） 2.开机自动挂载 （/etc/fstab）

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

读写：服务端：1234

5.[devops] path = /devops write list = chihiro workgroup = STAFF hosts allow = 172.25.0.0/24(注意空格)

6 7.rw 8.修改本地目录权限

多用户Samba共享:SMB客户端的 multiuser 挂载技术

客户端：添加挂载参数 – multiuser – sec=ntlmssp

cifscreds add -u 共享用户名 服务器地址

ISCSI网络磁盘： 服务器提供磁盘空间,客户机连接并当成本地磁盘使用 默认端口：3260

backstore,后端存储 target,磁盘组 lun,逻辑单元

IQN名字规范：iqn.yyyy-mm.倒序域名:自定义标识 作用：1.用来识别 target磁盘组,2.用来识别客户机身份

服务端 一、划分新的分区

二、部署iSCSI共享存储 1.安装软件targetcli 2.运行targetcli可以再交互式界面

A:创建与命名后端存储 backstores/block create name=nsd dev=/dev/vdb1（以下皆可用ls查询）

B:创建target,磁盘组iscsi/ create iqn.2018-09.com.example:server0

C:lun,逻辑单元

iscsi/iqn.2018-09.com.example:server0/tpg1/ luns create /backstores/block/nsd

D:配置访问控制，设置访问服务时，客户端声称的名字iscsi/iqn.2018-09.com.example:server0/tpg1/ acls create iqn.2018-09.com.example:desktop0

E:客户端访问本机的IP地址及端口

iscsi/iqn.2018-09.com.example:server0/tpg1/ portals create 172.25.0.11 3260-->exit退出

三、重起服务target

客户端：1.安装所需软件包iscsi-initiator-utils 2.修改配置文件/etc/iscsi/initiatorname.iscsi

InitiatorName=iqn.2018-09.com.example:desktop0

3.需重启服务 iscsid（刷新iqn）--

自起vim /var/lib/iscsi/nodes/iqn.2016-02.com.example\:server0/\*/default--->node.conn[0].startup = automatic

4.发现服务端共享存储 参考# man iscsiadm 搜索全文 /example

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

5.本机识别服务端共享：重起服务iscsi

用户帐号作用： 1. 访问控制的设置，不同的用户具备不同权限 2. 提高安全可靠性

useradd usermod(u d s g G从属组) userdel(r)（管理员0 系统程序用户1-999 普通用户1000）

组帐号作用：方便管理用户帐号 所有用户至少属于一个组 基本组 附加组

groupadd groupmod(n g) grouddel gpasswd(a d)

修改密码：passwd – passwd [用户名]+++++++++++++++++

切换用户身份：su - 用户

非交互式设置用户密码 – echo '密码' | passwd --stdin 用户名

存放的是用户的基本信息 /etc/passwd

组基本信息存放在 /etc/group

用户密码信息存放在 /etc/shadow

权限：基本权限：访问方式(权限) 以 - 开头：代表文本文件 以 d 开头：代表目录 以 l 开头：快捷方式

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

权限适用对象：所有者-user:u 所属组-group:g 其他用户-other: o a=ugo

设置基本权限：chmod [-R] 归属关系（ugo）+-=权限类别（rwx-） 文档...

设置文档归属：chown [-R] 属主:属组 文档..

附加权限(特殊权限):Set GID(属组x位上标识为s)Set UID 属主的x位标识为s Sticky Bit 其他人的x位标识为t

acl权限： 能够对个别用户、个别组设置独立的权限 查看：getfacl 文档...

setfacl [-R]：递归设置ACL权限 -m u:用户名:权限类别 文档... -m g:组名:权限类别 文档...

– setfacl -x u:用户名 文档... #删除指定的ACL -b 文档... #删除所有的ACL

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（粘滞位）（tmp）

root创建目录默认755权限 umask 权限掩码0022 （改umask 077）

LDAP网络用户认证，达到用户的集中管理

客户端：软件包：sssd authconfig-tui (配置完重起) autofs（+nfs）:/dev/ ##本地挂载

1.下载autofs 2.vim /etc/auto.master /home/guests /etc/guests.rule

3.vim /etc/guests.rule \* -rw classroom.example.com:/home/guests/& 4.重起

NTP网络时间协议：1.NTP服务器为客户机提供标准时间 2.NTP客户机需要与NTP服务器保持沟通

客户端 1.安装chrony软件包

2.修改程序主配置文件，指定服务端位置（/etc/chrony.conf） 3.重起程序（chronyd）4.验证-->重起

cron周期性计划任务：1.软件包:cronie、crontabs 2.系统服务:crond 3.日志文件:/var/log/crond

\*: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

crontab （-e-r-l）（加减查） -u root （分时日月周 任务命令行）

防火墙： 隔离作用 系统服务:firewalld 管理工具:firewall-cmd、firewall-config（图形化）

根据所在的网络场所区分,预设保护规则集：1.public:仅允许访问本机的ssh、ping、dhcp服务

2.trusted:允许任何访问 3.block:阻塞任何来访请求，明确拒绝 4.drop:丢弃任何来访的数据包，直接丢弃

查看（修改）默认区域：firewall-cmd --get（set）-default-zone（=public）

重新加载防火墙配置：firewall-cmd --reload

列出 public区域中规则：firewall-cmd --zone=public --list-all

临时（永久）添加服务：firewall-cmd （--permanent） --zone=public --add-service=http

单独拒绝172.25.0.10访问：firewall-cmd --zone=block --3add-source=172.25.0.10

实现本机的端口映射：访问两个地址可以看到相同的页面：

firewall-cmd --permanent --zone=public --add-forward-port=port=5423:proto=tcp:toport=80