

一、下载安装篇	5
1. 1 DOWNLOAD	5
1. 2 BURN	6
1. 2 INSTALLTION	6
1. 3 UPDATE SYSTEM AND PATCH	10
1. 4 第三方 YUM 源	10
1. 5 CONFIGURE VIM	10
二、配置时间服务器	10
三、配置 SSH 服务器	11
3. 1 SSH 安全设置	11
3. 2 配置 SSH 信任关系	12
3. 3 USE PARALLEL SSH	12
四、OPENVPN 原理及安装配置详解	12
4. 1 PKI 架构原理技术	12
4. 2 PKI 架构图例解释	19
4. 3 PKI 总结	25
对称加密	25
非对称加密	25
单向加密	25
秘钥交换机制	25
4. 4 OPENVPN 下载地址	26
4. 5 OPENVPN 配置文件详解	27
4. 5. 1 服务器端配置文件详解	27
4. 5. 2 客户端配置文件详解	31
4. 6 运行原理	34
4. 7 WINDOWS 版安装配置教程	35
安装 OpenVPN	35
OpenVPN 创建证书和密钥	36
OpenVPN 配置	44
4. 8 LINUX 版安装配置教程	47
安装 OpenVPN 的前提条件	47
安装 Linux 版 OpenVPN	47
OpenVPN 配置	48
五、DNS 服务器安装配置	58
5. 1 WINDOWS DNS 服务器安装配置篇	59
5. 2 LINUX DNS 服务器配置	70
5. 3 POWERDNS 安装配置	73
5. 4 路由器配置 DNS 服务器	76
六、STORAGE SERVER	76
6. 1 NFS 网络文件系统	76

Linux 平台	76
Windows 平台	78
6.2 iSCSI 存储服务器	80
Linux 平台	80
Windows 平台	83
存储多路径	87
开源存储系统 FreeNAS	88
开源存储系统 Openfiler	88
企业级存储系统 Open-E	88
七、PXE SERVER	89
八、安装配置 ORACLE 12C	90
8.1 环境准备	90
8.1 ADD ORACLE NET LISTENER	97
8.2 CREATE DATABASE	101
8.3 MANAGER	107
8.4 ORACLE 数据库备份及还原	108
8.5 RMAN 备份还原	108
九、ORACLE RAC 集群架构详解	116
方案拓扑	117
软硬件环境	117
IP 地址和域名规划	118
组件环境安装【两个节点】	120
调整 TMPFS 文件系统的大小【双节点】	121
配置 DNS 解析【双节点】	121
关闭 NTP 服务【双节点】	123
创建用户和组【双节点】	123
创建目录并赋予权限【双节点】	123
配置读写文件权限【双节点】	124
配置环境变量（双节点）	124
配置系统内核参数【双节点】	125
配置 SSH 双节点信任【双节点】	126
配置存储多路径（双节点）	126
安装集群软件【双节点】	129
修改 IP 地址详细记录	150
十、VNC SERVER	150
十一、XRDП 远程桌面	151
十二、DRBD 分布式存储	153
十三、DRBD+HEARTBEAT+MYSQL 高可用	155
十四、备份服务器 BACULA	167
架构图	167

服务器端	169
客户端	171
恢复	172
使用图形界面 GUI	172
客户端 GUI 图像界面	173
十五、虚拟化	173
KVM	173
OVIRT	176
国产虚拟化 CECOSI	178
点击下载	213
点击下载	213
点击下载	216
点击下载	224
点击下载	230
点击下载	237
点击下载	244
操作指南	244
XEN	244
十六、分布式存储 GLUSTERFS	245
十七、弹性存储 LVM	254
十八、版本控制服务器	255
18.1 SVN 服务器搭建和使用（一）	255
18.2 SVN 服务器搭建和使用（二）	265
18.3 SVN 服务器搭建和使用（三）	272
18.4 LINUX 版本控制服务器配置	277
十九、FTP 服务器大全	278
19.1 VSFTPD	278
二十、日志服务器	281
二十一、DHCP 服务器（PXE-SERVER）	282
二十二、ACL	288
二十三、RSYNC 同步	288
二十四、RKHUNTER	289
二十五、杀毒	289
二十六、LINUX 加入 WINDOWS AD 域	289
主机入侵防御系统 HIDS	290
二十七、LINUX 配置 JAVA 环境	290
二十八、LINUX JAVA 容器 TOMCAT	291

二十九、LINUX 建立私有云存储	292
三十、COCKPIT 系统管理面板	293
四十、LINUX 防火墙 FIREWALLD 配置详解	293
四十一、LINUX 破解密码	295
四十二、登录次数限制、限制登录	296
四十三、磁盘配额 QUATA	296
四十四、WEBMAIN 管理面板	297
四十五、监控系列	300
MONITORIX	300
MUNIN	301
分布式监控 ZABBIX	302
CACTI+NAGIOS+MRTG	307
四十六、负载均衡	318
PEN	318
LVS	324
LVS+KEEPALIVED	324
四十七、OPENLDAP 域服务器	325

一、下载安装篇

1. 1 DownLoad

Centos 7 是完全兼容 RHEL7 的社区版本，该操作系统是免费开源的，号称完全兼容“RHEL7”，有一大批社区技术支持，版本功能强大、稳定，是企业服务器平台首选。

下载地址：http://mirror.centos.org/centos-7/7.2.1511/isos/x86_64/

版本介绍：

版本	版本介绍
CentOS-7-x86_64-DVD-1511.iso	This DVD image contains all the packages that can be installed using the installer. This is the recommended image for most users.
CentOS-7-x86_64-NetInstall-1511.iso	This is the network install and rescue image. The installer will ask from where it should fetch the packages to be installed. This image is most useful if you have a local mirror of CentOS packages.
CentOS-7-x86_64-Everything-1511.iso	This image contains the complete set of packages for CentOS 7. It can be used for installing or populating a local mirror. This image needs a dual layer DVD or an 8GB USB flash drive. These images are Live images of CentOS 7. Depending on the name they use the respective display manager. They are designed for testing purposes and exploring the CentOS 7 environment. They will not modify the content of your hard disk, unless you choose to install CentOS 7 from within the Live environment. Please be advised that you can not change the set of installed packages in this case. This needs to be done within the installed system using 'yum'.
CentOS-7-x86_64-LiveKDE-1511.iso	The aim of this image is to install a very basic CentOS 7 system, with the minimum of packages needed to have a functional system. Please burn this image onto a CD and boot your computer off it. A preselected set of packages will be installed on your system. Everything else needs to be installed using yum. The set of packages installed by this image is identical to
CentOS-7-x86_64-Minimal-1511.iso	

the one installed when choosing
the group named "Minimal" from the full DVD image.

1.2 Burn

下载好镜像后，将镜像刻录成DVD/CD，插入光驱中，设置光驱启动顺序。刻录程序可以选择系统自带的，也可以选择第三方的。

第三方烧录软件：<http://down3.3987.com:801/2010/ImgBurn.rar>

1.2 Installation

- 进入安装界面，三个选项安装、测试、排错测试



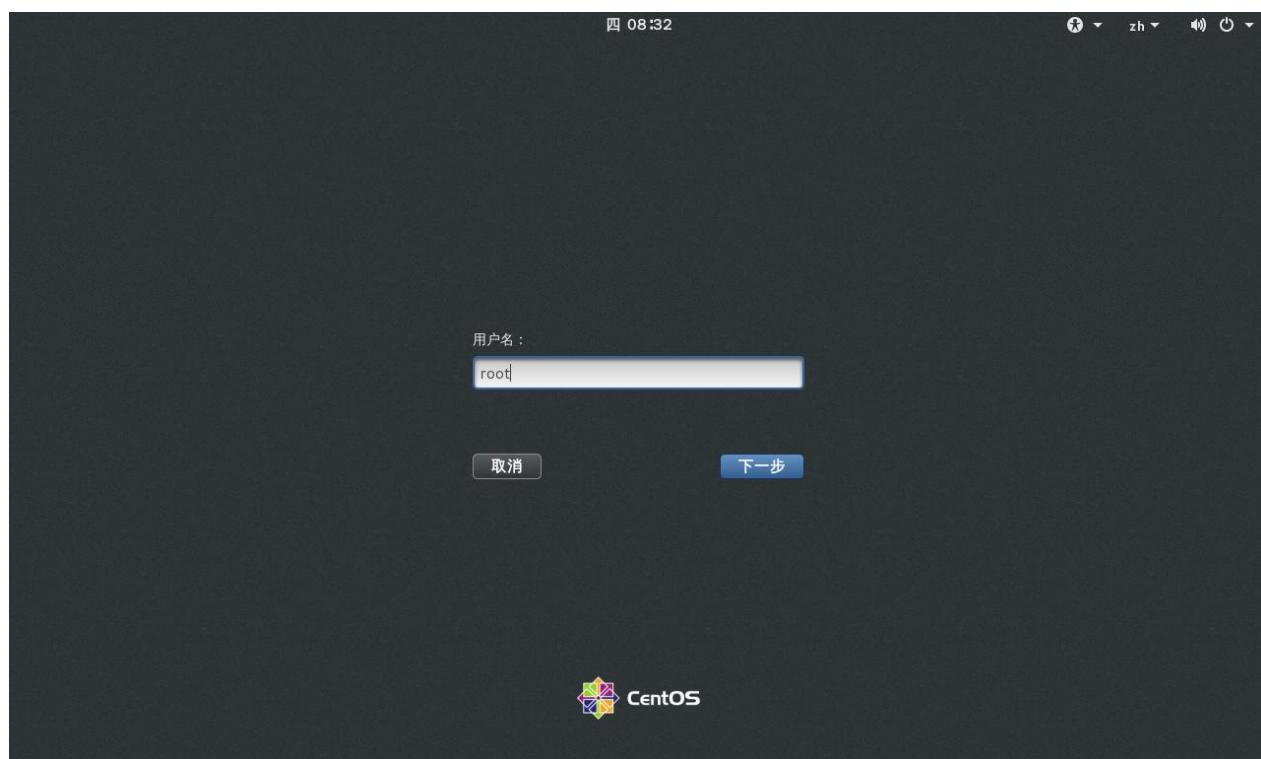
- 勾选相应的选项，然后开始安装。



■ 开始安装，进入安装进度。



■ 安装完成，开始基本登录设置



■ 更改网卡名为 Eth0

[root@rdh 桌面]# vim /etc/default/grub

```
GRUB_CMDLINE_LINUX="net.ifname=0 biosdevname=0 rhgb quiet"
```

```
[root@rdh 桌面]# grub2-mkconfig -o /boot/grub2/grub.cf
```

■ Add User

```
Useradd
```

```
-D: 默认值， 默认采用/etc/default/useradd
```

```
-b: 设置基本宿主目录， 类似/home
```

```
-c: 做基本描述， 做用户名的基本描述， 一般跟用户名的全称
```

```
-d: 跟用户的家目录
```

```
-e: 跟用户的过期时间， 一般的格式 YY-MM-DD
```

```
-f: 过期之后的彻底禁用时间， 0 表示立即禁止， -1 表示禁用此功能
```

```
-u: 表示 UID
```

```
-g: 表示 GID
```

```
-s: 表示登录的 shell， 一般常用的一个是/sbin/nologin
```

```
-G: 所表示的用户组 Group1、 Group2、 Group3
```

```
-m: --Create-home, 创建宿主目录
```

```
-K: 更改/etc/login.defs
```

```
PASS_MAX_DAYS 99999
```

```
PASS_MIN_DAYS 0
```

```
PASS_MIN_LEN 5
```

```
PASS_WARN_AGE 7
```

```
[root@rdh 桌面]# useradd wmm
```

```
[root@rdh 桌面]# echo "jstvps" |passwd --stdin wmm
```

更改用户 wmm 的密码。

passwd: 所有的身份验证令牌已经成功更新。

```
[root@rdh 桌面]# usermod -G wheel wmm
```

```
[root@rdh 桌面]# vim /etc/pam.d/su
```

```
auth required pam_wheel.so use_uid
```

```
[root@rdh 桌面]# vim /etc/aliases
```

```
[root@rdh 桌面]# newaliases
```

■ Stop Firewalld

如果你的防火墙是不必要的，比如说你的 LAN 里已经有防火墙，可以考虑关闭防火墙

```
[root@rdh 桌面]# systemctl stop firewalld
```

```
[root@rdh 桌面]# systemctl disable firewalld
```

```
[root@rdh 桌面]# systemctl is-enabled firewalld
```

Disabled

```
[root@rdh 桌面]# vim /etc/sysconfig/selinux
```

```
SELINUX=disabled
```

```
[root@rdh 桌面]# setenforce 0
```

```
[root@rdh 桌面]# getenforce
```

Permissive

■ 设置主机名

```
[root@rdh 桌面]# hostnamectl set-hostname RDH.COM
```

```
[root@rdh 桌面]# hostname
```

rdh.com

■ 列出服务

```
[root@rdh 桌面]# systemctl list-unit-files|grep httpd
```

1.3 Update System and patch

```
[root@rdh 桌面]# yum -y install update
```

1.4 第三方 YUM 源

```
[root@rdh ~]# vim /etc/default/grub
GRUB_CMDLINE_LINUX="net.ifnames=0 biosdevname=0 rhgb quiet"
[root@rdh ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
[root@rdh ~]# yum -y install yum-plugin-priorities
[root@rdh ~]# sed -i -e "s/\\]$\\]\\npriority=1/g"
/etc/yum.repos.d/CentOS-Base.repo
[root@rdh ~]# yum -y install epel-release
[root@rdh ~]# sed -i -e "s/\\]$\\]\\npriority=5/g" /etc/yum.repos.d/epel.repo
[root@rdh ~]# sed -i -e "s/enabled=1/enabled=0/g" /etc/yum.repos.d/epel.repo
[root@rdh ~]# yum --enablerepo=epel -y install mariadb*
[root@rdh ~]# yum -y install
http://pkgs.repolinux.org/rpmforge-release/rpmforge-release-0.5.3-1.el7.rf.x86\_64.rpm
```

【sed 's/要被取代的字串/新的字串/g'】

1.5 Configure VIM

二、配置时间服务器

```
[root@rdh ~]# yum -y install ntp
[root@rdh ~]# vim /etc/ntp.conf
restrict 0.0.0.0 mask 0.0.0.0 nomodify notrap
server 127.0.0.1
server 192.168.88.212
[root@rdh ~]# systemctl start ntpd
[root@rdh ~]# systemctl enable ntpd
[root@rdh ~]# ntpq -p
=====
[root@rdh ~]# yum -y install chrony
[root@rdh ~]# vim /etc/chrony.conf
server 127.0.0.1
server 192.168.88.212
allow 0.0.0.0/24
[root@rdh ~]# chronyc sources
[root@rdh ~]# ntpdate localhost【客户端设置】
```

三、配置 SSH 服务器

```
[root@rdh ~]# vim /etc/ssh/sshd_config
78 PermitEmptyPasswords no
79 PasswordAuthentication yes
=====Scp 远程复制=====
[root@rdh ~]# scp anaconda-ks.cfg root@192.168.88.212:/root
=====Sftp 安全 Ftp 服务器=====
sftp>
bye: 退出
cd: 切换目录
chdir: 更改目录
chgrp: 更改组
chmod: 更改权限
chown: 更改用户
df: 显示
dir: 显示目录
exit: 退出
get: 下载
help: 帮助
pwd: 显示当前目录
mget: 批量下载
mput: 批量上传
```

3.1 SSH 安全设置

```
#semanage port -a -t ssh_port_t -p tcp
#Port 22
更改端口，必须操作更改端口
#LoginGraceTime 2m
#PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
allowusers
allowgroups
denyusers
denygroups
=====Tcp Wrapper=====
规则：先允许在拒绝
sshd : ALL except 192.168.88.
sshd : .rdh.com , 192.168.88. , except wmm
ChrootDirectory /home
```

3.2 配置 SSH 信任关系

```
[root@rdh ~]# rm -rf ~/.ssh/  
[root@rdh ~]# mkdir ~/.ssh/  
[root@rdh ~]# chmod 700 ~/.ssh/  
[root@rdh ~]# cd ~/.ssh/  
[root@rdh .ssh]# ssh-keygen -t rsa  
[root@rdh .ssh]# ssh 192.168.88.212 cat  
/home/wmm/.ssh/id_rsa.pub>>authorized_keys  
[root@rdh .ssh]# scp authorized_keys 192.168.88.213:/home/wmm/.ssh/  
[root@rdh .ssh]# ssh 192.168.88.212 date
```

3.3 Use Parallel SSH

```
[root@rdh ~]# yum --enablerepo=epel -y install pssh
```

四、OpenVPN 原理及安装配置详解

4.1 PKI 架构原理技术

PKI：公开秘钥基础设施

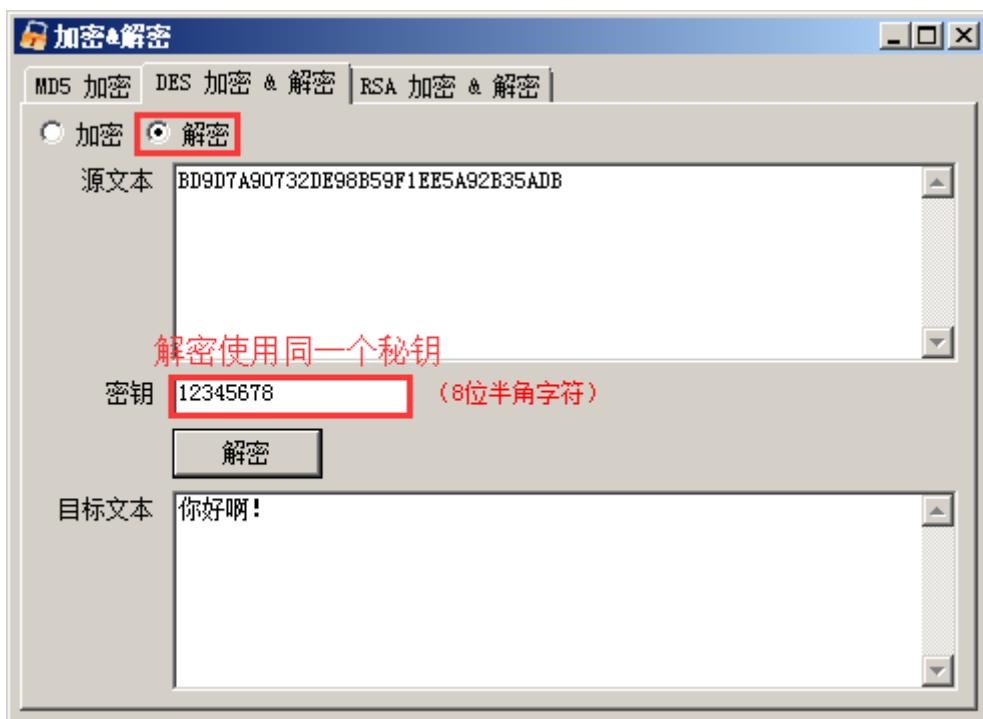
案例：甲想传送一份合同给乙，这个合同对甲乙双方都比较重要，要求信息真实、不被篡改、能稳定的传送到乙方。

正常的想法是甲必须对该合同进行加密才能保证不被其他人查看其内容，那么到底采用什么加密技术才能使合同传送既安全又快速呢？可以采用一些熟悉的对称加密算法：DES、3DES、RCA。对称加密算法的特点是就是快速安全且加密解密都使用同一个秘钥。我们使用秘钥对合同进行加密后发送给乙方，那么问题来了，乙方如何知道该秘钥呢？邮件发送、电话通知，这些都容易泄露信息。

===== 加密 =====



===== 解密 =====



问题 2：那么黑客截获了该文件，是否用同一算法就能解开此文件呢？

不可以，加密和解密都需要两个组件，加密算法和解密秘钥，只有解密秘钥才能解密，黑客不知道此秘钥。

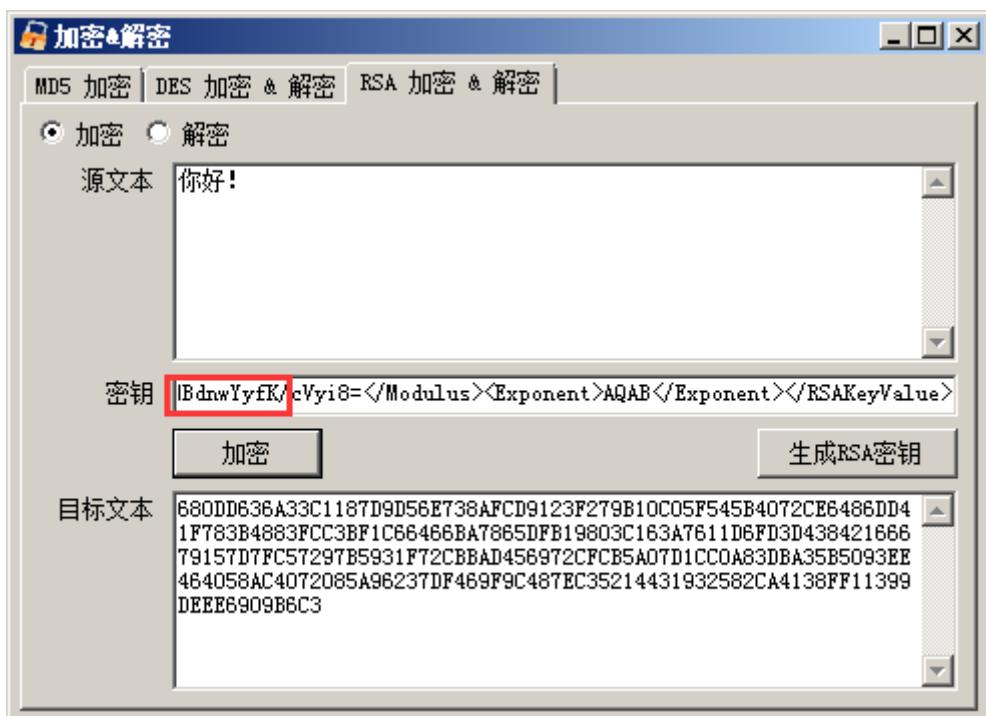
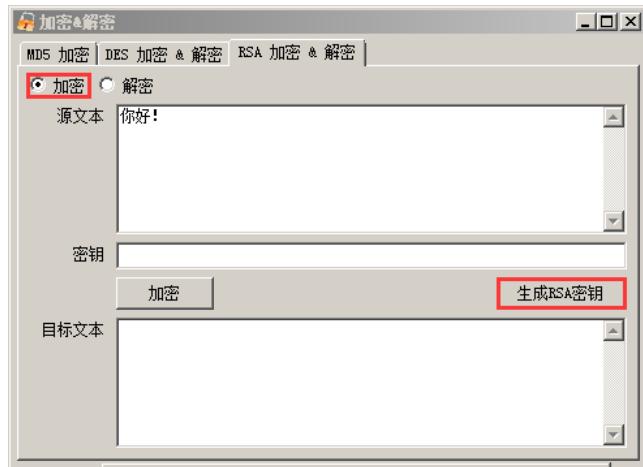
问题 3：既然黑客不知道此秘钥，那么该秘钥怎么才能安全的传送给乙方呢？电话通知，可能被窃取，Internet 传送，可能被窃取。怎么办？

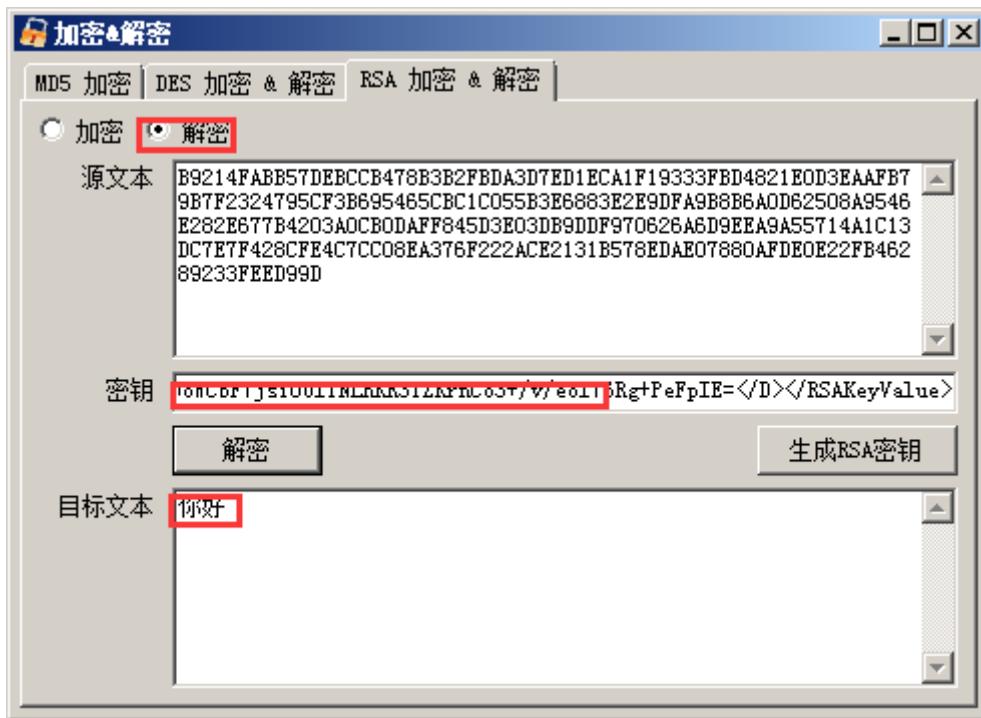
用非对称加密算法加密对称秘钥后进行传送。与对称加密算法不同，非对称加密算法有两个秘钥：公开秘钥（PublicKey）和私有秘钥（PrivateKey）。公钥和私钥是一对，如果用公钥进行加密，那么私钥就进行解密；如果私钥进行加密，那么公钥就进行解密；公钥可以在 Internet 上传送，私钥自己保存。即使黑客截获了公钥，他得不到私钥，也不能进行解密，也就解不开密文。

=====公钥加密、私钥解密=====

问题 4：既然乙的公钥可以加密非对称加密算法的对称秘钥，那么为什么不直接用乙的公钥直接加密文件呢？

不可以，因为非对称加密算法的加密速度非常慢，比对称加密算法慢 10-100 倍，且会使密文变长，所以一般用对称加密算法加密文件，然后用非对称加密算法加密对称加密算法对应的对称秘钥。



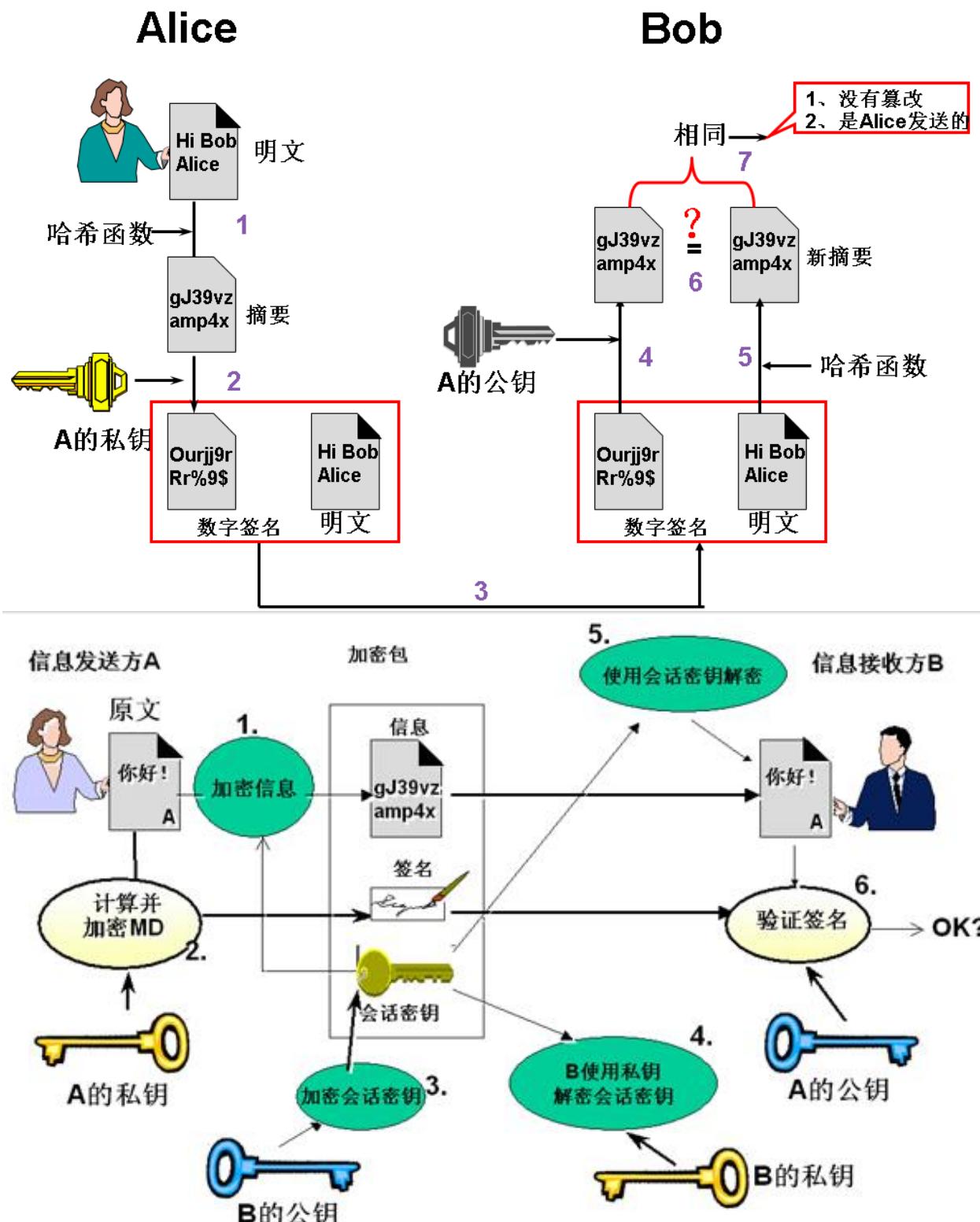


问题 5：如何黑客截获到公钥和加密文件，他也没法看到详细内容，因为他没有乙的私钥，但是他可以使用对称加密算法加密一份假文件，并用乙的公钥加密这份假文件，发送给乙方，乙方用自己的私钥解开用公钥加密的假文件，并愉快的阅读其内容。怎么解决？

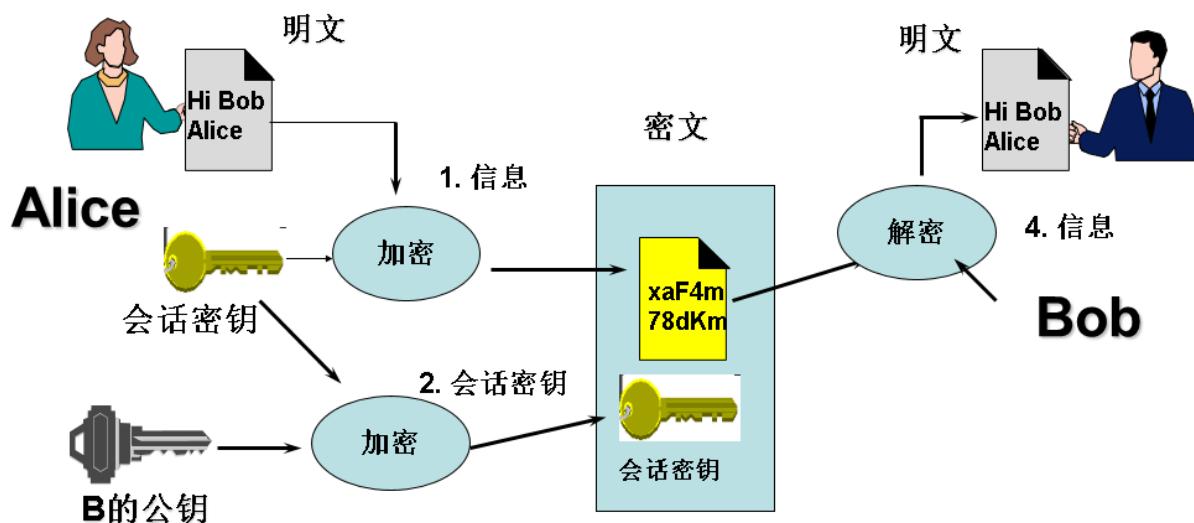
使用数字签名来证明其身份、使用 MD5 和 SHA 来检验数据完整性。数字签名指用散列算法 MD5、SHA1 等加密大数据块中提取的摘要信息，且是单向的，且不能从摘要信息恢复到任何一个原文，但如果原信息发生任何改动，摘要也会随之改动。这样甲用户可以用散列算法加密文件，形成摘要，然后用私钥加密摘要信息，这样即使黑客获取了也无用，因为黑客不能从摘要信息中获取任何信息，但乙不一样，它可以使用公钥解密，得到其摘要（如果乙能打开摘要，表示该该文件肯定是乙发的，因为只有乙的公钥才能解开乙的私钥，乙的私钥只有乙自己知道）【用私钥加密用散列函数加密的摘要信息叫“数字签名”】，并对解密后的合同文件同样进行散列算法，同样得到一个摘要，然后比较两个摘要是否一致，如果一致，表示该合同文件没有被篡改，如果不一致，表示该合同文件被篡改。

用数字签名能够校验数据的完整性和防抵赖。

将数字摘要和数字签名结合

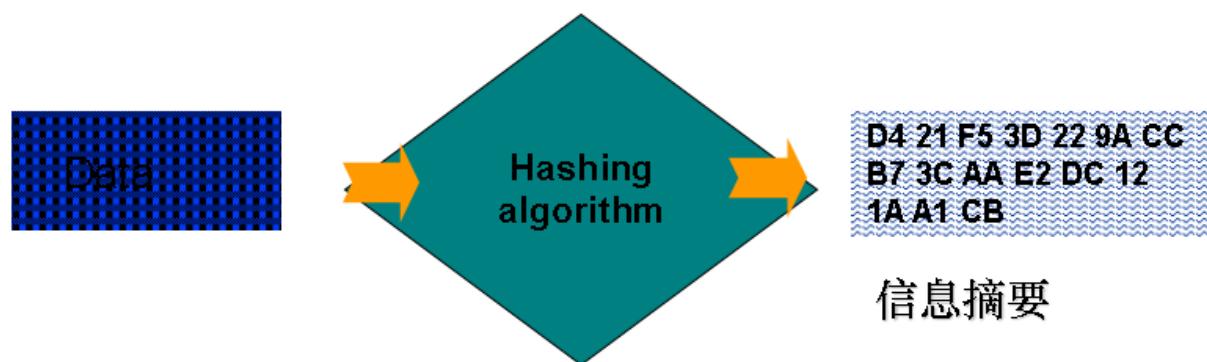


组合密码技术



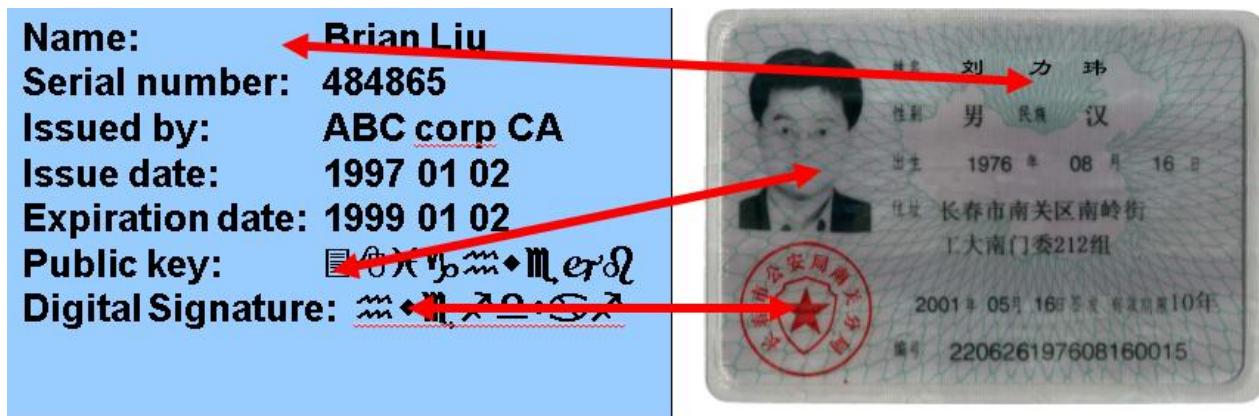
- 产生一个一次性，**对称密钥——会话密钥**
- **用会话密钥加密信息**
- 最后用接收者的**公钥加密会话密钥——因为它很短**

摘要算法 (Hash)



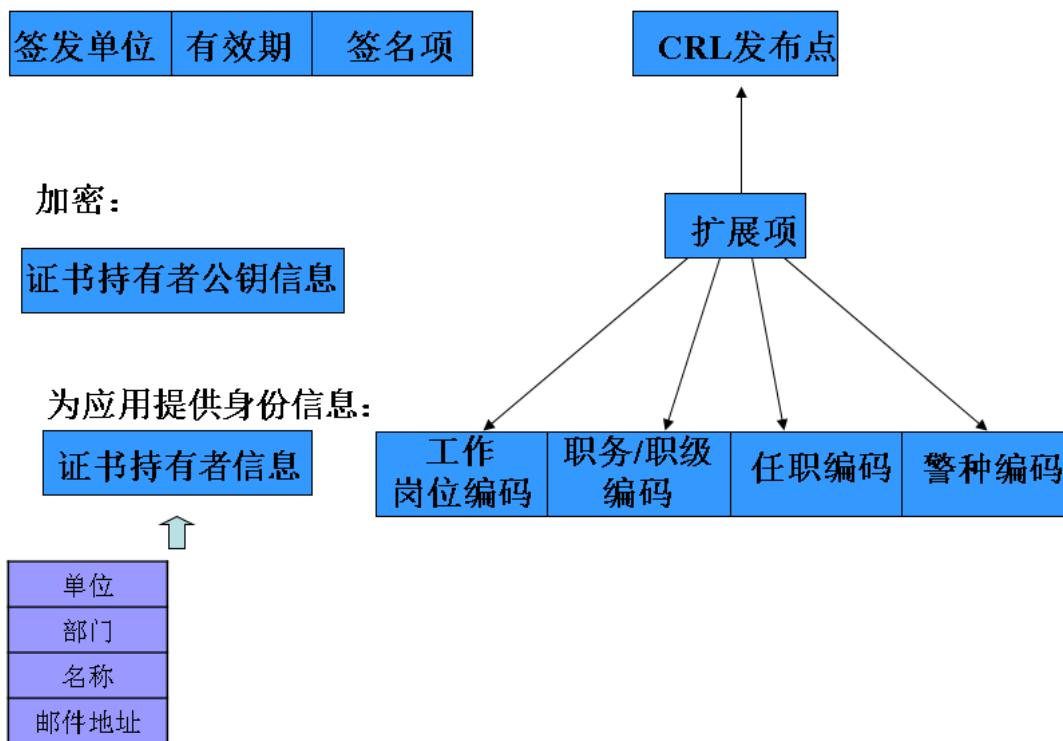
问题 6：用对称加密算法加密合同文件，用非对称加密算法加密对称秘钥，用数字签名保证数据的完整性和防抵赖【用私钥加密用散列函数加密的摘要信息叫“数字签名”】，那么这样就万无一失了吗？不能，关键是乙怎么保证自己收到的公钥一定是甲发送的呢？用 CA 数字证书，来绑定公钥和公钥附属信息。

数字证书是一个经过证书授权中心数字签名的包含公钥及公钥所有者信息的文件，是网络通信中标识通信双方身份信息的一系列数据，他提供了一种在 Internet 网上进行身份认证的方式，类似人们日常生活中的身份证件和护照。



证书格式

证书有效性校验:

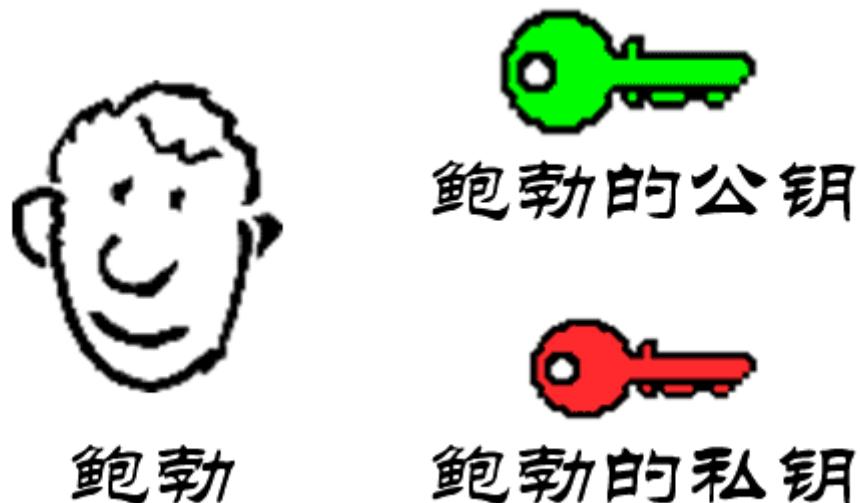


CA 授权数字证书中心，将对所有使用公钥的用户发送一个数字证书，该数字证书包含公钥和公钥附属的合法信息，而 CA 授权数字中心使用数字签名技术保证数字证书的合法性和不可篡改，因为数字证书是公开的，所以可以在 Internet 上传送，数字证书+密文+摘要信息一同发送给乙方，而乙方通过授权数字中心的公钥进行解密和校验，如果一切正常，那么可以详细该公钥属于甲。

问题 6: 用对称加密算法加密合同文件 (DES、AES、3DES)，用非对称加密算法 (RSA、RC6) 加密对称秘钥，形成“会话秘钥”，用数字签名技术(用私钥加密用散列函数加密的摘要信息，散列函数 (MD5、SHA-1)) 保证数据的完整性和防抵赖，用数字证书技术来确保公钥及公钥所有者信息的真实性，用第三方时间戳来保证否认某一段的时间抵赖。

4.2 PKI 架构图例解释

1.



注：鲍勃有两把钥匙，一把是公钥，另一把是私钥。

2.



注：鲍勃把公钥送给他的朋友们——帕蒂、道格、苏珊——每人一把。

3.



"Hey Bob,
how about
lunch at
Taco Bell. I
hear they
have free
refills!"

苏珊



公钥加密

HNFmsEm6Un
BejhhyCGKO
KJUxhiygSBC
EiC0QYIh/Hn
3xgiKBcyLK1
UcYiYlxz2lCF
HDC/A



HNFmsEm6Un
BejhhyCGKO
KJUxhiygSBC
EiC0QYIh/Hn
3xgiKBcyLK1
UcYiYlxz2lCF
HDC/A

鲍勃

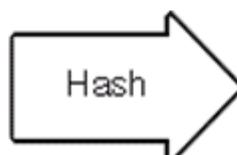


私钥解密

"Hey Bob,
how about
lunch at
Taco Bell. I
hear they
have free
refills!"

注：鲍勃收信后，用私钥解密，就看到了信件内容。这里要强调的是，只要鲍勃的私钥不泄露，这封信就是安全的，即使落在别人手里，也无法解密。

5、



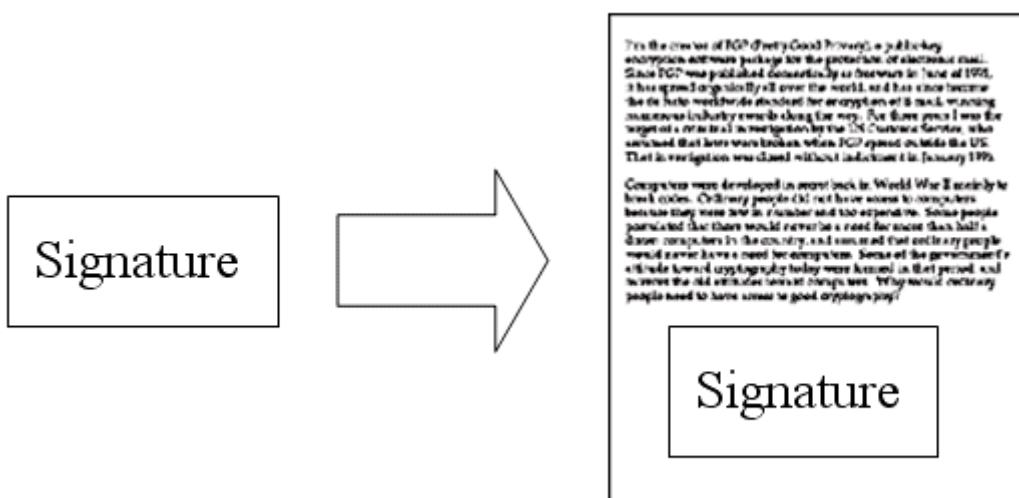
注：鲍勃给苏珊回信，决定采用“数字签名”。他写完后先用 Hash 函数，生成信件的摘要 (digest)。

6.



注：然后，鲍勃使用私钥，对这个摘要加密，生成“数字签名” (signature)。

7.



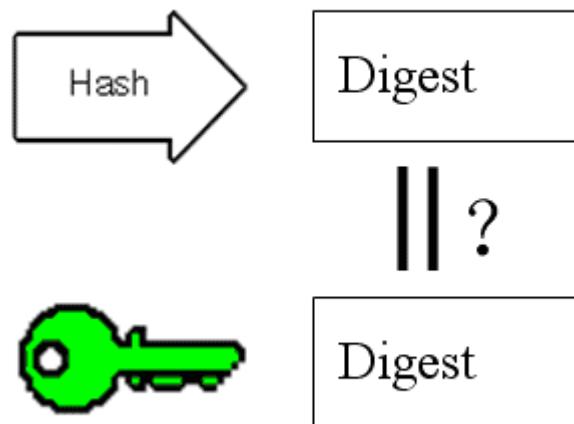
注：鲍勃将这个签名，附在信件下面，一起发给苏珊。

8.



注：苏珊收信后，取下数字签名，用鲍勃的公钥解密，得到信件的摘要。由此证明，这封信确实是鲍勃发出的。防抵赖技术。

9.



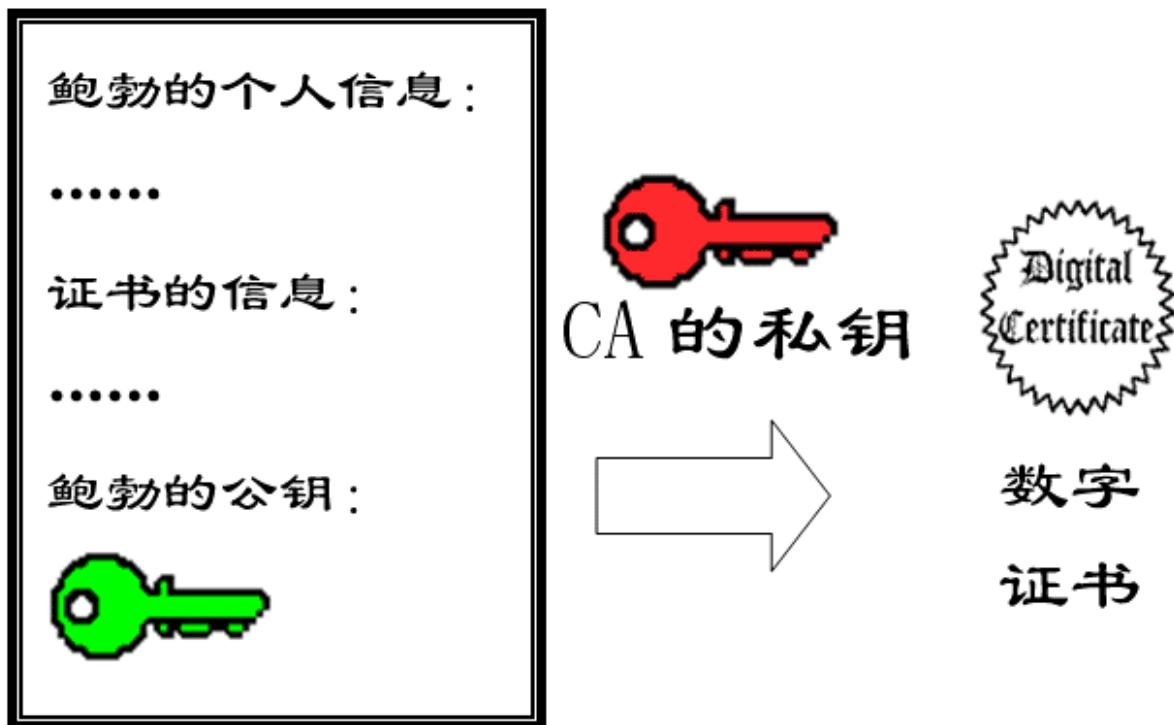
注：苏珊再对信件本身使用 Hash 函数，将得到的结果，与上一步得到的摘要进行对比。如果两者一致，就证明这封信未被修改过。

10.



注：复杂的情况出现了。道格想欺骗苏珊，他偷偷使用了苏珊的电脑，用自己的公钥换走了鲍勃的公钥。此时，苏珊实际拥有的是道格的公钥，但是还以为这是鲍勃的公钥。因此，道格就可以冒充鲍勃，用自己的私钥做成“数字签名”，写信给苏珊，让苏珊用假的鲍勃公钥进行解密。

11.



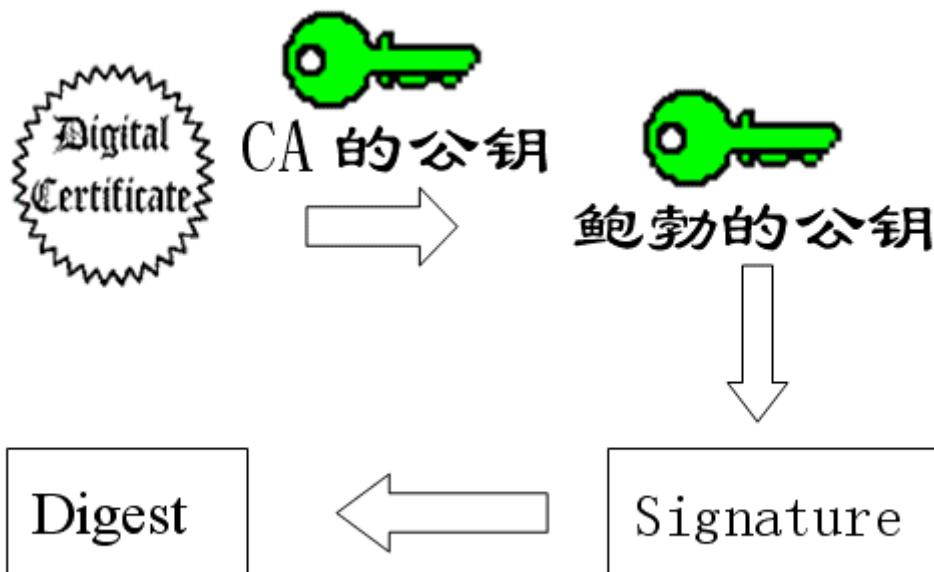
注：后来，苏珊感觉不对劲，发现自己无法确定公钥是否真的属于鲍勃。她想到了一个办法，要求鲍勃去找“证书中心”(certificate authority，简称 CA)，为公钥做认证。证书中心用自己的私钥，对鲍勃的公钥和一些相关信息一起加密，生成“数字证书”(Digital Certificate)。

12.



注：鲍勃拿到数字证书以后，就可以放心了。以后再给苏珊写信，只要在签名的同时，再附上数字证书就行了。

13.



注：苏珊收信后，用 CA 的公钥解开数字证书，就可以拿到鲍勃真实的公钥了，然后就能证明“数字签名”是否真的是鲍勃签的。

4.3 PKI 总结

互联网上中间人攻击通常用的三种方式：窃听、数据篡改、会话劫持。

数据加密的常用的三种方式有：对称加密、非对称加密、单向加密。

对称加密

- 1、加密方和解密方使用同一个密钥。
- 2、加密解密的速度比较快，适合数据比较长时的使用。
- 3、密钥传输的过程不安全，且容易被破解，密钥管理也比较麻烦。
- 4、加密算法：DES(Data Encryption Standard)、3DES、AES(Advanced Encryption Standard，支持 128、192、256、512 位密钥的加密)、Blowfish。
- 5、加密工具：openssl、gpg(pgp 工具)。

非对称加密

- 1、每个用户拥用一对密钥加密：公钥和私钥。
- 2、公钥加密，私钥解密；私钥加密，公钥解密。
- 3、公钥传输的过程不安全，易被窃取和替换。
- 4、由于公钥使用的密钥长度非常长，所以公钥加密速度非常慢，一般不使用其去加密。
- 5、某一个用户用其私钥加密，其他用户用其公钥解密，实现数字签名的作用。
- 6、公钥加密的另一个作用是实现密钥交换。
- 7、加密和签名算法：RSA、ELGamal。
- 8、公钥签名算法：DSA。
- 9、加密工具：gpg、openssl

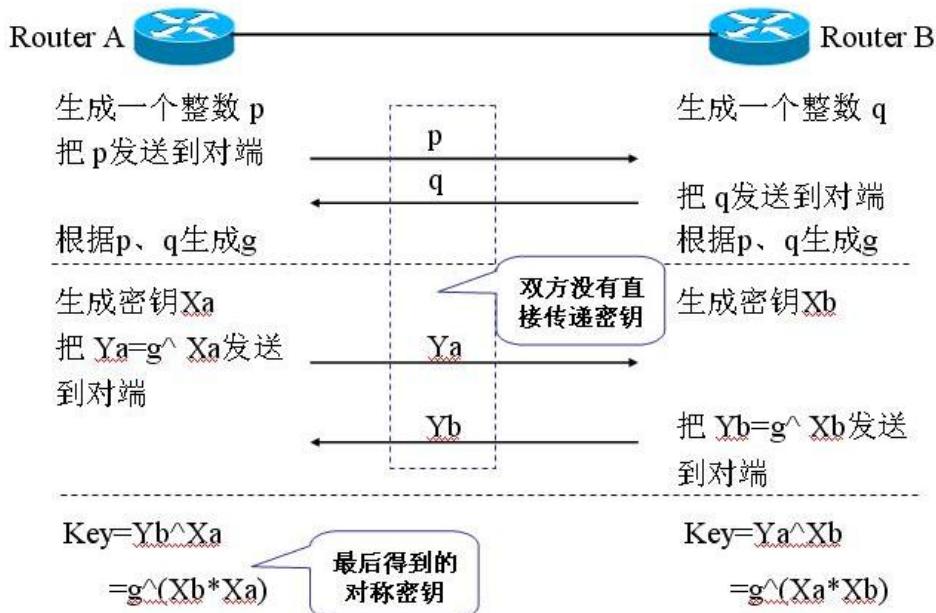
单向加密

- 1、特征：雪崩效应、定长输出和不可逆。
- 2、作用是：确保数据的完整性。
- 3、加密算法：md5（标准密钥长度 128 位）、sha1（标准密钥长度 160 位）、md4、CRC-32
- 4、加密工具：md5sum、sha1sum

秘钥交换机制

- 1、公钥加密实现：发送方用接收方的公钥加密自己的密钥，接收方用自己的私钥解密得到发送方的密钥，反过来亦然，从而实现密钥交换。
- 2、使用 DH 算法：前提发送方和接受方协商使用同一个大素数 P 和生成数 g，各自产生的随机数 X 和 Y。发送方将 g 的 X 次方 mod P 产生的数值发送给接收方，接收方将 g 的 Y 次方 mod P 产生的数值发送给发送方，发送方再对接收的结果做 X 次方运算，接收方对接收的结果做 Y 次方运算，最终密码形成，密钥交换完成。

DH 算法的基本原理



4.4 OpenVPN 下载地址

Linux 版本: <http://pan.baidu.com/s/1hq3iQJ6>

Windows 版本: <http://pan.baidu.com/s/1dDpHNjz>

OpenVPN 是一个功能齐全的 SSL VPN，它使用行业标准的 SSL/TLS 协议实现了 OSI 模型第 2 层（数据链路层）或第 3 层（网络层）的安全网络扩展。OpenVPN 支持基于证书、智能卡以及用户名/密码等多种形式的灵活的客户端认证方法，并可以通过应用于 VPN 虚拟接口的防火墙规则为指定用户或用户组设置访问控制策略。OpenVPN 不是一个 Web 应用程序代理，也不需要通过 Web 浏览器来进行操作。

- [OpenVPN 快速入门](#)
- [安装 OpenVPN](#)
- [选择使用基于路由还是基于桥接的 VPN？](#)
- [编号私有子网](#)
- [创建 CA，并生成服务器和客户端证书](#)
- [启动 VPN 并进行初步连通性测试](#)
- [随系统启动自动运行](#)
- [控制运行中的 OpenVPN 进程](#)
- [扩大 VPN 作用范围，包含服务器或客户端子网中的其他计算机](#)
- [推送 DHCP 选项到客户端](#)
- [配置指定客户端的规则和访问策略](#)
- [使用其他的身份验证方法](#)
- [如何使用客户端的智能卡为 OpenVPN 配置添加双重认证](#)
- [路由所有客户端流量（包括 Web 流量）通过 VPN](#)
- [在使用动态 IP 地址的计算机上运行 OpenVPN 服务器](#)
- [通过 HTTP 代理连接 OpenVPN 服务器](#)
- [通过 OpenVPN 连接 Samba 网络共享服务器](#)
- [实现具备负载均衡/故障转移功能的配置](#)

- [增强 OpenVPN 的安全性](#)
- [撤销证书](#)
- [附加的安全注意事项](#)

4.5 OpenVPN 配置文件详解

本文将介绍如何配置 OpenVPN 服务器端的配置文件。在 Windows 系统中，该配置文件一般叫做 `server.ovpn`；在 Linux/BSD 系统中，该配置文件一般叫做 `server.conf`。虽然配置文件名称不同，但其中的配置内容与配置方法却是相同的。

4.5.1 服务器端配置文件详解

```
#####
# 针对多客户端的 OpenVPN 2.0 的服务器端配置文件示例
#
# 本文件用于多客户端<->单服务器端的 OpenVPN 服务器端配置
#
# OpenVPN 也支持单机<->单机的配置(更多信息请查看网站上的示例页面)
#
# 该配置支持 Windows 或者 Linux/BSD 系统。此外，在 Windows 上，记得将路径加上双引号，
# 并且使用两个反斜杠，例如: "C:\\Program Files\\OpenVPN\\config\\foo.key"
#
# '# or ';'开头的均为注释内容
#####

#OpenVPN 应该监听本机的哪些 IP 地址？
#该命令是可选的，如果不设置，则默认监听本机的所有 IP 地址。
;local a.b.c.d

# OpenVPN 应该监听哪个 TCP/UDP 端口？
# 如果你想在同一台计算机上运行多个 OpenVPN 实例，你可以使用不同的端口号来区分它们。
# 此外，你需要在防火墙上开放这些端口。
port 1194

#OpenVPN 使用 TCP 还是 UDP 协议？
;proto tcp
proto udp

# 指定 OpenVPN 创建的通信隧道类型。
# "dev tun"将会创建一个路由 IP 隧道，
# "dev tap"将会创建一个以太网隧道。
#
# 如果你是以太网桥接模式，并且提前创建了一个名为"tap0"的与以太网接口进行桥接的虚拟接口，则你可以使用"dev tap0"
#
```

```
# 如果你想控制 VPN 的访问策略，你必须为 TUN/TAP 接口创建防火墙规则。  
#  
# 在非 Windows 系统中，你可以给出明确的单位编号 (unit number)，例如"tun0"。  
# 在 Windows 中，你也可以使用"dev-node"。  
# 在多数系统中，除非你部分禁用或者完全禁用了 TUN/TAP 接口的防火墙，否则 VPN 将不起作用。  
;dev tap  
dev tun  
  
# 如果你想配置多个隧道，你需要用到网络连接面板中 TAP-Win32 适配器的名称 (例如"MyTap")。  
# 在 XP SP2 或更高版本的系统中，你可能需要有选择地禁用掉针对 TAP 适配器的防火墙  
# 通常情况下，非 Windows 系统则不需要该指令。  
;dev-node MyTap  
  
# 设置 SSL/TLS 根证书 (ca)、证书 (cert) 和私钥 (key)。  
# 每个客户端和服务器端都需要它们各自的证书和私钥文件。  
# 服务器端和所有的客户端都将使用相同的 CA 证书文件。  
#  
# 通过 easy-rsa 目录下的一系列脚本可以生成所需的证书和私钥。  
# 记住，服务器端和每个客户端的证书必须使用唯一的 Common Name。  
#  
# 你也可以使用遵循 X509 标准的任何密钥管理系统来生成证书和私钥。  
# OpenVPN 也支持使用一个 PKCS #12 格式的密钥文件 (详情查看站点手册页面的"pkcs12"指令)  
ca ca.crt  
cert server.crt  
key server.key # 该文件应该保密  
  
# 指定迪菲·赫尔曼参数。  
# 你可以使用如下名称命令生成你的参数：  
# openssl dhparam -out dh1024.pem 1024  
# 如果你使用的是 2048 位密钥，使用 2048 替换其中的 1024。  
dh dh1024.pem  
  
# 设置服务器端模式，并提供一个 VPN 子网，以便于从中为客户端分配 IP 地址。  
# 在此处的示例中，服务器端自身将占用 10.8.0.1，其他的将提供客户端使用。  
# 如果你使用的是以太网桥接模式，请注释掉该行。更多信息请查看官方手册页面。  
server 10.8.0.0 255.255.255.0  
  
# 指定用于记录客户端和虚拟 IP 地址的关联关系的文件。  
# 当重启 OpenVPN 时，再次连接的客户端将分配到与上一次分配相同的虚拟 IP 地址  
ifconfig-pool-persist ipp.txt  
  
# 该指令仅针对以太网桥接模式。  
# 首先，你必须使用操作系统的桥接能力将以太网网卡接口和 TAP 接口进行桥接。
```

```
# 然后，你需要手动设置桥接接口的 IP 地址、子网掩码：  
# 在这里，我们假设为 10.8.0.4 和 255.255.255.0。  
# 最后，我们必须指定子网的一个 IP 范围(例如从 10.8.0.50 开始，到 10.8.0.100 结束)，以便于分配给连接的客  
户端。  
# 如果你不是以太网桥接模式，直接注释掉这行指令即可。  
;server-bridge 10.8.0.4 255.255.255.0 10.8.0.50 10.8.0.100  
  
# 该指令仅针对使用 DHCP 代理的以太网桥接模式，  
# 此时客户端将请求服务器端的 DHCP 服务器，从而获得分配给它的 IP 地址和 DNS 服务器地址。  
#  
# 在此之前，你也需要先将以太网网卡接口和 TAP 接口进行桥接。  
# 注意：该指令仅用于 OpenVPN 客户端，并且该客户端的 TAP 适配器需要绑定到一个 DHCP 客户端上。  
;server-bridge  
  
# 推送路由信息到客户端，以允许客户端能够连接到服务器背后的其他私有子网。  
# (简而言之，就是允许客户端访问 VPN 服务器自身所在的其他局域网)  
# 记住，这些私有子网也要将 OpenVPN 客户端的地址池 (10.8.0.0/255.255.255.0) 反馈回 OpenVPN 服务器。  
;push "route 192.168.10.0 255.255.255.0"  
;push "route 192.168.20.0 255.255.255.0"  
  
# 为指定的客户端分配指定的 IP 地址，或者客户端背后也有一个私有子网想要访问 VPN，  
# 那么你可以针对该客户端的配置文件使用 ccd 子目录。  
# (简而言之，就是允许客户端所在的局域网成员也能够访问 VPN)  
  
# 举个例子：假设有个 Common Name 为"Thelonious"的客户端背后也有一个小型子网想要连接到 VPN，该子网为  
192.168.40.128/255.255.255.248。  
# 首先，你需要去掉下面两行指令的注释：  
;client-config-dir ccd  
;route 192.168.40.128 255.255.255.248  
# 然后创建一个文件 ccd/Thelonious，该文件的内容为：  
#     iroute 192.168.40.128 255.255.255.248  
#这样客户端所在的局域网就可以访问 VPN 了。  
# 注意，这个指令只能在你是基于路由、而不是基于桥接的模式下才能生效。  
# 比如，你使用了"dev tun"和"server"指令。  
  
# 再举个例子：假设你想给 Thelonious 分配一个固定的 IP 地址 10.9.0.1。  
# 首先，你需要去掉下面两行指令的注释：  
;client-config-dir ccd  
;route 10.9.0.0 255.255.255.252  
# 然后在文件 ccd/Thelonious 中添加如下指令：  
#     ifconfig-push 10.9.0.1 10.9.0.2  
  
# 如果你想要为不同群组的客户端启用不同的防火墙访问策略，你可以使用如下两种方法：
```

```
# (1) 运行多个 OpenVPN 守护进程，每个进程对应一个群组，并为每个进程(群组)启用适当的防火墙规则。
# (2) (进阶) 创建一个脚本来动态地修改响应于来自不同客户的防火墙规则。
# 关于 learn-address 脚本的更多信息请参考官方手册页面。
;learn-address ./script

# 如果启用该指令，所有客户端的默认网关都将重定向到 VPN，这将导致诸如 web 浏览器、DNS 查询等所有客户端流量都经过 VPN。
# (为确保能正常工作，OpenVPN 服务器所在计算机可能需要在 TUN/TAP 接口与以太网之间使用 NAT 或桥接技术进行连接)
;push "redirect-gateway def1 bypass-dhcp"

# 某些具体的 windows 网络设置可以被推送到客户端，例如 DNS 或 WINS 服务器地址。
# 下列地址来自 opendns.com 提供的 Public DNS 服务器。
;push "dhcp-option DNS 208.67.222.222"
;push "dhcp-option DNS 208.67.220.220"

# 去掉该指令的注释将允许不同的客户端之间相互"可见"(允许客户端之间互相访问)。
# 默认情况下，客户端只能"看见"服务器。为了确保客户端只能看见服务器，你还可以在服务器端的 TUN/TAP 接口上设置适当的防火墙规则。
;client-to-client

# 如果多个客户端可能使用相同的证书/私钥文件或 Common Name 进行连接，那么你可以取消该指令的注释。
# 建议该指令仅用于测试目的。对于生产使用环境而言，每个客户端都应该拥有自己的证书和私钥。
# 如果你没有为每个客户端分别生成 Common Name 唯一的证书/私钥，你可以取消该行的注释(但不推荐这样做)。
;duplicate-cn

# keepalive 指令将导致类似于 ping 命令的消息被来回发送，以便于服务器端和客户端知道对方何时被关闭。
# 每 10 秒钟 ping 一次，如果 120 秒内都没有收到对方的回复，则表示远程连接已经关闭。
keepalive 10 120

# 出于 SSL/TLS 之外更多的安全考虑，创建一个" HMAC 防火墙"可以帮助抵御 DoS 攻击和 UDP 端口淹没攻击。
# 你可以使用以下命令来生成：
#   openvpn --genkey --secret ta.key
#
# 服务器和每个客户端都需要拥有该密钥的一个拷贝。
# 第二个参数在服务器端应该为'0'，在客户端应该为'1'。
;tls-auth ta.key 0 # 该文件应该保密

# 选择一个密码加密算法。
# 该配置项也必须复制到每个客户端配置文件中。
;cipher BF-CBC      # Blowfish (默认)
;cipher AES-128-CBC # AES
;cipher DES-EDE3-CBC # Triple-DES
```

```
# 在 VPN 连接上启用压缩。  
# 如果你在此处启用了该指令，那么也应该在每个客户端配置文件中启用它。  
comp-lzo  
  
# 允许并发连接的客户端的最大数量  
;max-clients 100  
  
# 在完成初始化工作之后，降低 OpenVPN 守护进程的权限是个不错的主意。  
# 该指令仅限于非 Windows 系统中使用。  
;user nobody  
;group nobody  
  
# 持久化选项可以尽量避免访问那些在重启之后由于用户权限降低而无法访问的某些资源。  
persist-key  
persist-tun  
  
# 输出一个简短的状态文件，用于显示当前的连接状态，该文件每分钟都会清空并重写一次。  
status openvpn-status.log  
  
# 默认情况下，日志消息将写入 syslog (在 Windows 系统中，如果以服务方式运行，日志消息将写入 OpenVPN 安装目录的 log 文件夹中)。  
# 你可以使用 log 或者 log-append 来改变这种默认情况。  
# "log" 方式在每次启动时都会清空之前的日志文件。  
# "log-append" 这是在之前的日志内容后进行追加。  
# 你可以使用两种方式之一 (但不要同时使用)。  
;log          openvpn.log  
;log-append  openvpn.log  
  
# 为日志文件设置适当的冗余级别 (0~9)。冗余级别越高，输出的信息越详细。  
#  
# 0 表示静默运行，只记录致命错误。  
# 4 表示合理的常规用法。  
# 5 和 6 可以帮助调试连接错误。  
# 9 表示极度冗余，输出非常详细的日志信息。  
verb 3  
  
# 重复信息的沉默度。  
# 相同类别的信息只有前 20 条会输出到日志文件中。  
;mute 20
```

4.5.2 客户端配置文件详解

本文将介绍如何配置 OpenVPN 客户端的配置文件。在 Windows 系统中，该配置文件一

般叫做 `client.ovpn`; 在 Linux/BSD 系统中, 该配置文件一般叫做 `client.conf`。虽然配置文件名称不同, 但其中的配置内容与配置方法却是相同的。

```
#####
# 针对多个客户端的 OpenVPN 2.0 的客户端配置文件示例
#
# 该配置文件可以被多个客户端使用, 当然每个客户端都应该有自己的证书和密钥文件
#
# 在 Windows 上此配置文件的后缀应该是 ".ovpn", 在 Linux/BSD 系统中则是 ".conf"
#####

# 指定这是一个客户端, 我们将从服务器获取某些配置文件指令
client

# 在大多数系统中, 除非你部分禁用或者完全禁用了 TUN/TAP 接口的防火墙, 否则 VPN 将不起作用。
;dev tap
dev tun

# 在 Windows 系统中, 如果你想配置多个隧道, 则需要该指令。
# 你需要用到网络连接面板中 TAP-Win32 适配器的名称(例如"MyTap")。
# 在 XP SP2 或更高版本的系统中, 你可能需要禁用掉针对 TAP 适配器的防火墙。
;dev-node MyTap

# 指定连接的服务器是采用 TCP 还是 UDP 协议。
# 这里需要使用与服务器端相同的设置。
;proto tcp
proto udp

# 指定服务器的主机名(或 IP)以及端口号。
# 如果有多个 VPN 服务器, 为了实现负载均衡, 你可以设置多个 remote 指令。
remote my-server-1 1194
;remote my-server-2 1194

# 如果指定了多个 remote 指令, 启用该指令将随机连接其中的一台服务器,
# 否则, 客户端将按照指定的先后顺序依次尝试连接服务器。
;remote-random

# 启用该指令, 与服务器连接中断后将自动重新连接, 这在网络不稳定的情况下(例如: 笔记本电脑无线网络)非常有用。
resolv-retry infinite

# 大多数客户端不需要绑定本机特定的端口号
nobind

# 在初始化完毕后, 降低 OpenVPN 的权限(该指令仅限于非 Windows 系统中使用)
```

```
;user nobody
;group nobody

# 持久化选项可以尽量避免访问在重启时由于用户权限降低而无法访问的某些资源。
persist-key
persist-tun

# 如果你是通过 HTTP 代理方式来连接到实际的 VPN 服务器，请在此处指定代理服务器的主机名(或 IP)和端口号。
# 如果你的代理服务器需要身份认证，请参考官方手册页面。
;http-proxy-retry # 连接失败时自动重试
;http-proxy [proxy server] [proxy port #]

# 无线网络通常会产生大量的重复数据包。设置此标识将忽略掉重复数据包的警告信息。
;mute-replay-warnings

# SSL/TLS 参数配置。
# 更多描述信息请参考服务器端配置文件。
# 最好为每个客户端单独分配.crt/.key 文件对。
# 单个 CA 证书可以供所有客户端使用。
ca ca.crt
cert client.crt
key client.key

# 指定通过检查证书的 nsCertType 字段是否为"server"来验证服务器端证书。
# 这是预防潜在攻击的一种重要措施。
#
# 为了使用该功能，你需要在生成服务器端证书时，将其中的 nsCertType 字段设为"server"
# easy-rsa 文件夹中的 build-key-server 脚本文件可以达到该目的。
ns-cert-type server

# 如果服务器端使用了 tls-auth 密钥，那么每个客户端也都应该有该密钥。
;tls-auth ta.key 1

# 指定密码的加密算法。
# 如果服务器端启用了 cipher 指令选项，那么你必须也在这里指定它。
;cipher x

# 在 VPN 连接中启用压缩。
# 该指令的启用/禁用应该与服务器端保持一致。
comp-lzo

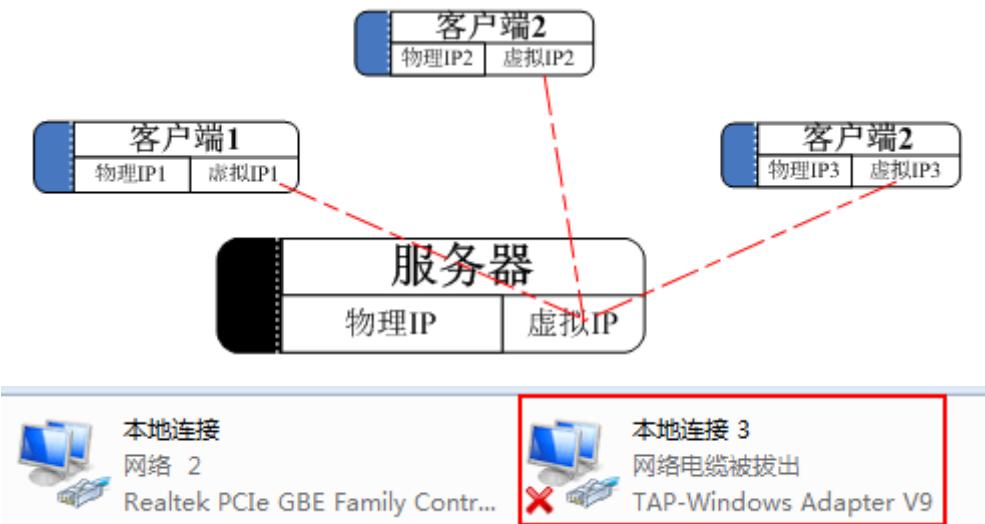
# 设置日志文件冗余级别(0~9)。
# 0 表示静默运行，只记录致命错误。
```

```
# 4 表示合理的常规用法。
# 5 和 6 可以帮助调试连接错误。
# 9 表示极度冗余，输出非常详细日志信息。
verb 3

# 忽略过多的重复信息。
# 相同类别的信息只有前 20 条会输出到日志文件中。
;mute 20
```

4.6 运行原理

OpenVPN 是一个用于创建虚拟专用网络 (Virtual Private Network) 加密通道的免费开源软件。使用 OpenVPN 可以方便地在家庭、办公场所、住宿酒店等不同网络访问场所之间搭建类似于局域网的专用网络通道。OpenVPN 使用方便，运行性能优秀，支持 Solaris、Linux 2.2+ (Linux 2.2+ 表示 Linux 2.2 及以上版本，下同)、OpenBSD 3.0+、FreeBSD、NetBSD、Mac OS X、Android 和 Windows 2000+ 的操作系统，并且采用了高强度的数据加密，再加上其开源免费的特性，使得 OpenVPN 成为中小型企业及个人的 VPN 首选产品。OpenVPN 的运行原理其实很简单，其核心机制就是在 OpenVPN 服务器和客户端所在的计算机上都安装一个虚拟网卡 (又称虚拟网络适配器)，并获得一个对应的虚拟 IP 地址。OpenVPN 的服务器和多个客户端就可以通过虚拟网卡，使用这些虚拟 IP 进行相互访问了。其中，OpenVPN 服务器起到一个路由和控制的作用 (相当于一个虚拟的路由器)。



当然，这个虚拟网卡毕竟是虚拟的，我们发送给虚拟网卡的数据在经过封装之后还是需要通过物理网卡才能发送出去。因此，OpenVPN 还要做的另外一件事就是，采用数据加密、身份验证等各种手段确保数据安全无误地到达目的地。

在 OpenVPN 中，最常用的数据加密手段，就是采用 SSL 协议。使用 SSL 协议进行传输就需要相应的证书和密钥，因此我们使用 OpenVPN 之前，还需要在服务器端生成相应的证书、密钥。

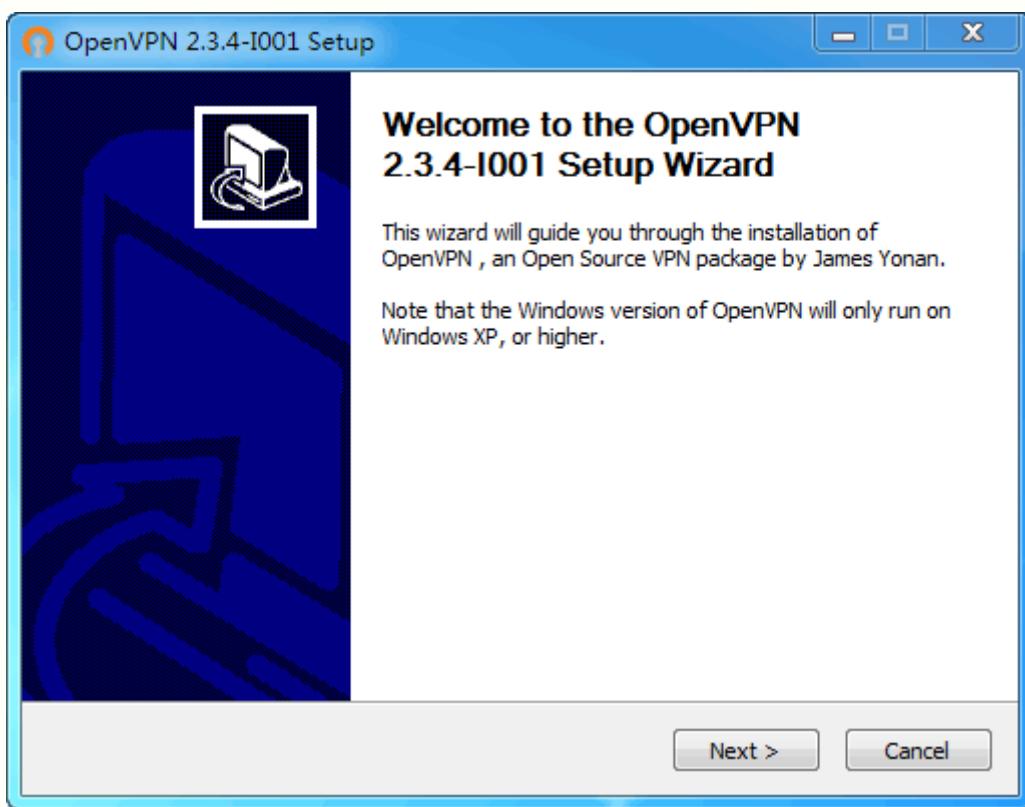
4.7 Windows 版安装配置教程

安装 OpenVPN

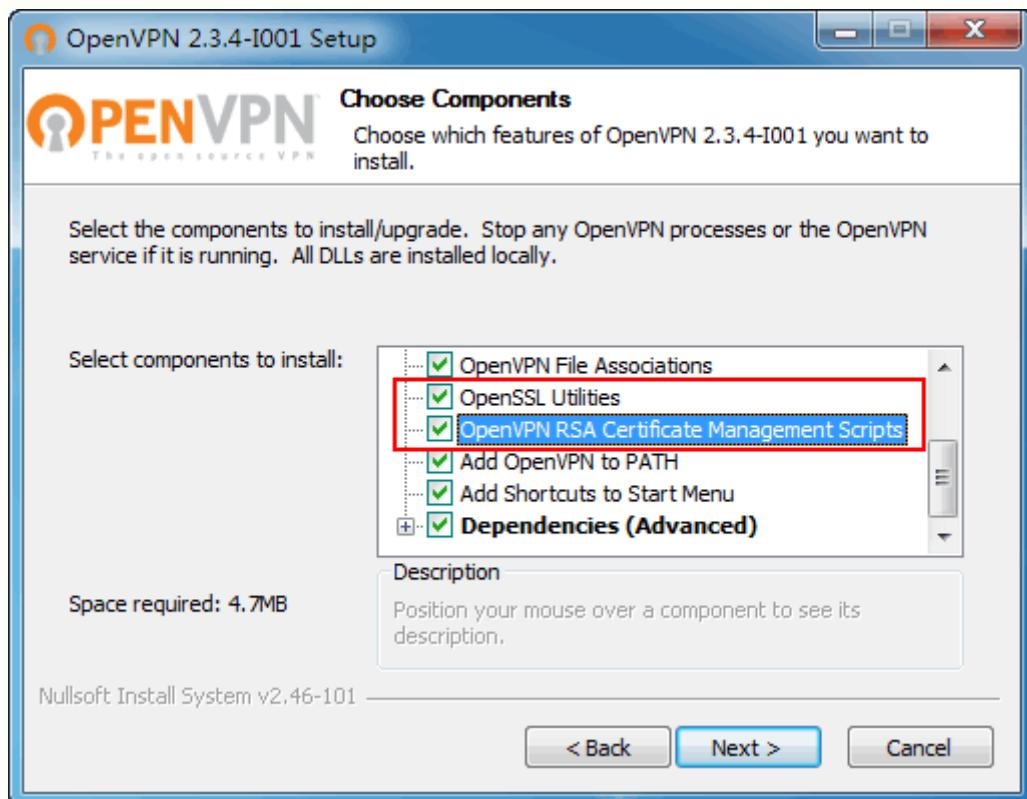
在 Windows 上安装 OpenVPN 是比较简单的，因为 OpenVPN 官方直接为我们提供了可执行的安装程序 (.exe)，不管是在 OpenVPN 服务器端还是在客户端上，安装步骤都没有什么差别。

在这里，我们以目前最新的 `openvpn-install-2.3.4-1001-i686.exe` (更新于 2014-05-02，[下载地址](#)) 来介绍如何在 Windows 上安装 OpenVPN。

首先，直接双击该安装文件。看到下面这个界面就应该不会感到陌生了——“还是熟悉的配方，还是熟悉的味道”，我们也就不再一步步赘述安装步骤了，只是对某些关键步骤进行简要说明。



如下图所示，在选择安装哪些 OpenVPN 组件的配置界面中，如果你的 OpenVPN 是 2.3 及以上版本，请确保勾选下图所示的两个组件选项（该组件在 OpenVPN 2.2 中已默认选中，主要用于生成加密证书、密钥等）。



此外，在安装 OpenVPN 的过程中，系统会提示类似如下信息。这里安装的其实就是 OpenVPN 将要使用到的虚拟网卡，请点击【安装】按钮以允许安装。



OpenVPN 创建证书和密钥

在 [OpenVPN 运行原理](#)一文中我们已经提到，OpenVPN 除了安装虚拟网卡来创建虚拟专用网络外，还要做的就是使用 SSL 协议以及相应的用户密码、证书密钥等手段进行数据加密、身份验证等。

因此，在安装完 OpenVPN 后，我们就需要生成一些证书给服务器以及客户度使用。当然，我们还需要给服务器和客户端创建一个配置文件，否则的话，我们无法对服务器和客户端进行细粒度的控制。更何况，使用 OpenVPN 的用户如此之多，你的客户端和服务器怎

么才能知道哪些是它的"同伙"的呢。

首先，我们需要在 OpenVPN 服务器端创建证书和密钥（服务器端和客户端使用的证书和密钥，都由服务器端负责创建）。

修改批处理文件模板

1、使用文本编辑器打开 OpenVPN 安装目录/easy-rsa/vars.bat.sample 文件（这实际上是一个批处理文件的模板，用于设置初始化的用户变量）。

如下图所示，我们只需要将红色矩形框中的 **HOME** 变量值改为文件夹 easy-rsa 的所在路径即可。至于其他变量，你可以不作修改，也可以根据个人需要，按照我们给出的蓝色文字提示进行修改。

建议更改图中 31~35 行内容处的变量值，因为后面每次生成证书都需要输入相关信息，设置默认值可以避免重复输入。

```
5 ↓  
6 set HOME=D:\OpenVPN\easy-rsa↓ 设置RSA根目录  
7 set KEY_CONFIG=openssl-1.0.0.cnf↓ OpenSSL配置  
8 ↓ 文件位置  
9 rem Edit this variable to point to↓  
10 rem your soon-to-be-created key↓  
11 rem directory.↓  
12 rem↓  
13 rem WARNING: clean-all will do↓  
14 rem a rm -rf on this directory↓  
15 rem so make sure you define↓  
16 rem it correctly!↓  
17 set KEY_DIR=keys↓ 生成的密钥文件存储目录  
18 ↓  
19 rem Increase this to 2048 if you↓  
20 rem are paranoid. This will slow↓  
21 rem down TLS negotiation performance↓  
22 rem as well as the one-time DH parms↓  
23 rem generation process.↓  
24 set KEY_SIZE=1024↓ 密钥长度  
25 ↓  
26 rem These are the default values for fields↓  
27 rem which will be placed in the certificate.↓  
28 rem Change these to reflect your site.↓  
29 rem Don't leave any of these parms blank.↓  
30 ↓  
31 set KEY_COUNTRY=US↓  
32 set KEY_PROVINCE=CA↓  
33 set KEY_CITY=SanFrancisco↓ 设置证书  
34 set KEY_ORG=OpenVPN↓ 或密钥的  
35 set KEY_EMAIL=mail@host.domain↓ 相关信息  
36 set KEY_CN=changeme↓  
37 set KEY_NAME=changeme↓  
38 set KEY_OU=changeme↓  
39 set PKCS11_MODULE_PATH=changeme↓  
40 set PKCS11_PIN=1234↓  
41 ←
```

初始化命令行

接着，我们打开命令提示符窗口，并转到 easy-rsa 目录，然后依次执行如下命令完成初始化工作：

```
init-config  
vars  
clean-all
```

```
D:\OpenUPN\easy-rsa>init-config  
D:\OpenUPN\easy-rsa>copy vars.bat.sample vars.bat  
已复制 1 个文件。  
D:\OpenUPN\easy-rsa>vars  
D:\OpenUPN\easy-rsa>clean-all  
系统找不到指定的文件。  
已复制 1 个文件。  
已复制 1 个文件。  
D:\OpenUPN\easy-rsa>
```

第一次执行 `clean-all` 时，提示"系统找不到指定的文件"是正常的，不用管它。该命令会删除掉之前生成的所有证书和密钥文件，以避免与之后全新生成的证书和密钥发生冲突。

创建 CA 证书

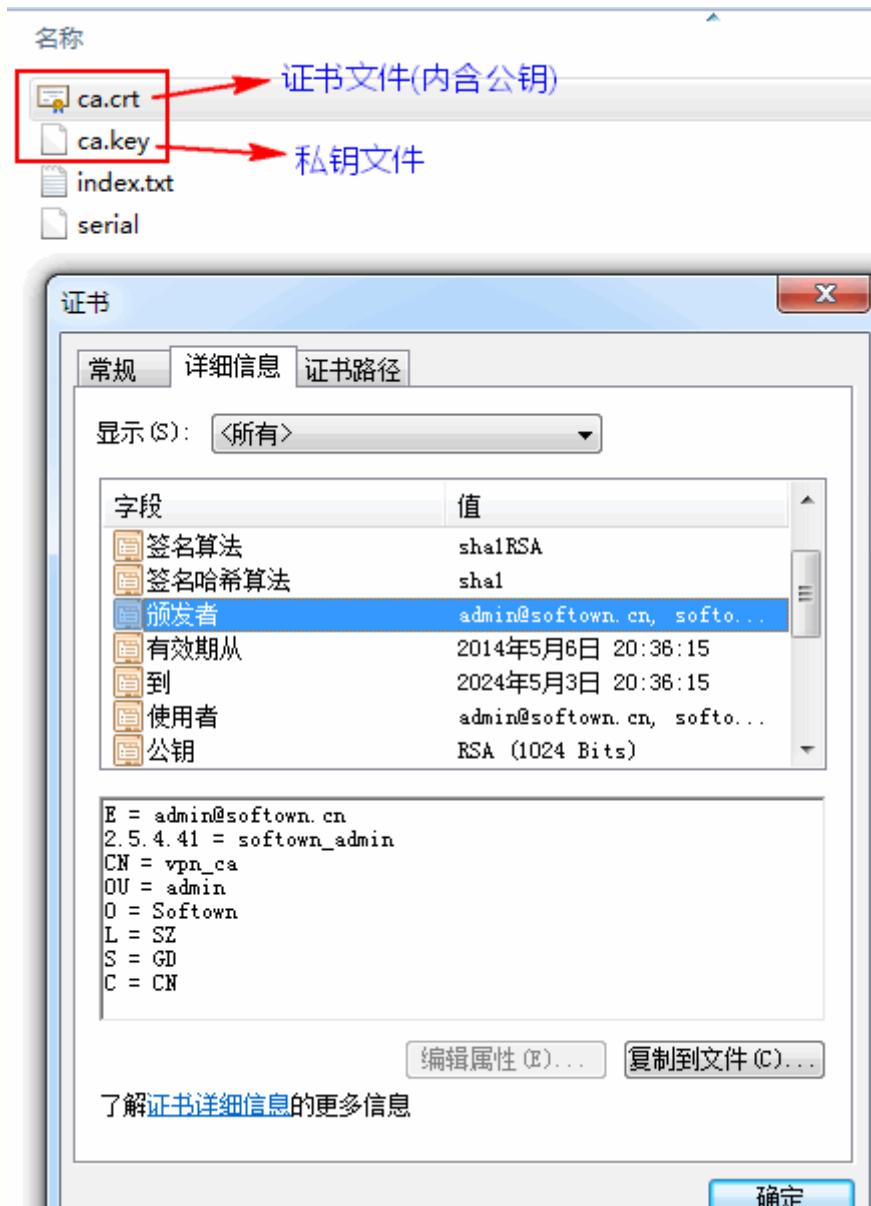
就在上述命令提示符窗口中，我们继续执行命令 `build-ca` 来生成 CA 根证书密钥对。

如下图所示，除了 `build-ca` 命令外，其他输入主要用于设置根证书的签名信息，包括国家、省、市、组织名称、单位名称、通用名、名称、邮箱地址等，请根据个人需要自行输入。输入框前面"[]"中的内容表示默认值，如果你不输入、直接按回车，则表示使用默认值；如果你输入"."则表示该字段信息留空。其中 Common Name(通用名称) 比较重要，相当于我们常说的"账号"（此处设为 `vpn_ca`）。

```
D:\OpenVPN\easy-rsa>build-ca
WARNING: can't open config file: /etc/ssl/openssl.cnf
Loading 'screen' into random state - done
Generating a 1024 bit RSA private key
-----+
-----+
writing new private key to 'keys\ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:CN
State or Province Name (full name) [CA]:GD
Locality Name (eg, city) [SanFrancisco]:SZ
Organization Name (eg, company) [OpenUPN]:Softown
Organizational Unit Name (eg, section) [changeme]:admin
Common Name (eg, your name or your server's hostname) [changeme]:vpn_ca
Name [changeme]:softown_admin
Email Address [mail@host.domain]:admin@softown.cn

D:\OpenVPN\easy-rsa>
```

此时，我们就可以在证书存放目录中看到生成的 CA 证书和私钥文件了。双击该证书，我们还可以看到刚才输入的相关信息。



创建服务器端证书

接着，我们使用命令 `build-key-server server` 来创建服务器端证书和私钥。整个创建过程与前面创建 CA 根证书的流程比较类似，我们只需要注意 Common Name 的参数值为 "server" 并随后设置相应的密码（这里设为 "server_pwd"，一般不会用到密码），最后输入两次 "y" 进行确认即可。

```
D:\OpenVPN\easy-rsa>build-key-server server
WARNING: can't open config file: /etc/ssl/openssl.cnf
Loading 'screen' into random state - done
Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to 'keys\server.key'

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [US]:CN
State or Province Name (full name) [CA]:GD
Locality Name (eg, city) [SanFrancisco]:SZ
Organization Name (eg, company) [OpenVPN]:Softown
Organizational Unit Name (eg, section) [changeme]:admin
Common Name (eg, your name or your server's hostname) [changeme]:server
Name [changeme]:server
Email Address [mail@host.domain]:admin@softown.cn

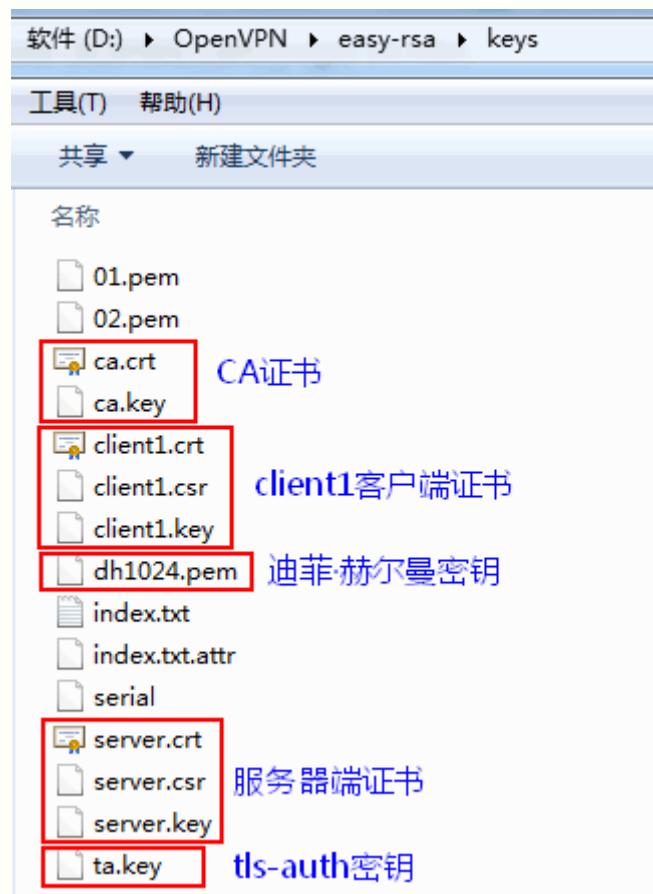
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:server_pwd
An optional company name []:softown
WARNING: can't open config file: /etc/ssl/openssl.cnf
Using configuration from openssl-1.0.0.cnf
Loading 'screen' into random state - done
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName          :PRINTABLE:'CN'
stateOrProvinceName  :PRINTABLE:'GD'
localityName         :PRINTABLE:'SZ'
organizationName     :PRINTABLE:'Softown'
organizationalUnitName:PRINTABLE:'admin'
commonName           :PRINTABLE:'server'
name                 :PRINTABLE:'server'
emailAddress         :IA5STRING:'admin@softown.cn'
Certificate is to be certified until May 4 11:10:08 2024 GMT <3650 days>
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

创建客户端证书

服务器端证书创建完毕后，我们就可以为所有的客户端分别创建证书了。创建命令为 `build-key clientName`，例如 `build-key client1`、`build-key client2`。

创建客户端证书的流程和创建服务器端证书的流程完全一致，只需要注意 Common Name 字段即可，在这里就不再赘述了。



OpenVPN 配置

下面，我们就开始进行 OpenVPN 的相关配置工作。现在，我们从 easy-rsa/keys 目录中拷贝出服务器端和客户端所需的文件，分别将其放在自己的 OpenVPN 安装路径/config 目录中。

其中，服务器端需要用到的文件有：

```
ca.crt  
ca.key  
dh1024.pem (如果最初的变量 KEY_SIZE 设为 2048，这里就是 dh2048.pem)  
server.crt  
server.key  
ta.key (如果不开启 tls-auth，则无需该文件)
```

客户端 client1 需要用到的文件有：

```
ca.crt  
client1.crt  
client1.key (名称 client1 根据个人设置可能有所不同)
```

ta.key(如果不开启 tls-auth, 则无需该文件)

当然, 我们还需要在 config 目录中各自放置一个配置文件, 服务器端的配置文件名为 server.ovpn, 客户端的配置文件为 client.ovpn。

这两个配置文件该如何编写呢? OpenVPN 已经在 sample-config 目录中为我们提供了相关的示例文件 server.ovpn 和 client.ovpn, 并且配置文件中的每个配置选项均有详细的英文说明(配置文件中 "#" 或 ";" 开头的均为注释内容)。



在这里, 我们先给出 server.ovpn 的详细配置, 并注明每项配置的作用。

```
local 192.168.1.101      #指定监听的本机 IP(因为有些计算机具备多个 IP 地址), 该命令是可选的, 默认监听所有 IP 地址。
port 1194                #指定监听的本机端口号
proto udp                 #指定采用的传输协议, 可以选择 tcp 或 udp
dev tun                   #指定创建的通信隧道类型, 可选 tun 或 tap
ca ca.crt                 #指定 CA 证书的文件路径
cert server.crt           #指定服务器端的证书文件路径
key server.key             #指定服务器端的私钥文件路径
dh dh1024.pem              #指定迪菲赫尔曼参数的文件路径
server 10.0.0.0 255.255.255.0    #指定虚拟局域网占用的 IP 地址段和子网掩码, 此处配置的服务器自身占用 10.0.0.1。
ifconfig-pool-persist ipp.txt    #服务器自动给客户端分配 IP 后, 客户端下次连接时, 仍然采用上次的 IP 地址(第一次分配的 IP 保存在 ipp.txt 中, 下一次分配其中保存的 IP)。
tls-auth ta.key 0            #开启 TLS-auth, 使用 ta.key 防御攻击。服务器端的第二个参数值为 0, 客户端的为 1。
keepalive 10 120            #每 10 秒 ping 一次, 连接超时时间设为 120 秒。
comp-lzo                  #开启 VPN 连接压缩, 如果服务器端开启, 客户端也必须开启
client-to-client            #允许客户端与客户端相连接, 默认情况下客户端只能与服务器相连接
persist-key
persist-tun                #持久化选项可以尽量避免访问在重启时由于用户权限降低而无法访问的某些资源。
```

```
status openvpn-status.log      #指定记录 OpenVPN 状态的日志文件路径  
verb 3                         #指定日志文件的记录详细级别, 可选 0-9, 等级越高日志内容越详细
```

接着是 `client.ovpn`。

```
client                  #指定当前 VPN 是客户端  
dev tun                 #必须与服务器端的保持一致  
proto udp                #必须与服务器端的保持一致  
remote 192.168.1.101 1194      #指定连接的远程服务器的实际 IP 地址和端口号  
resolv-retry infinite      #断线自动重新连接, 在网络不稳定的情况下(例如: 笔记本电脑无线网络)  
非常有用。  
nobind                  #不绑定特定的本地端口号  
persist-key  
persist-tun  
ca ca.crt                #指定 CA 证书的文件路径  
cert client1.crt          #指定当前客户端的证书文件路径  
key client1.key            #指定当前客户端的私钥文件路径  
ns-cert-type server        #指定采用服务器校验方式  
tls-auth ta.key 1          #如果服务器设置了防御 DoS 等攻击的 ta.key, 则必须每个客户端开启; 如果未  
设置, 则注释掉这一行;  
comp-lzo                  #与服务器保持一致  
verb 3                     #指定日志文件的记录详细级别, 可选 0-9, 等级越高日志内容越详细
```

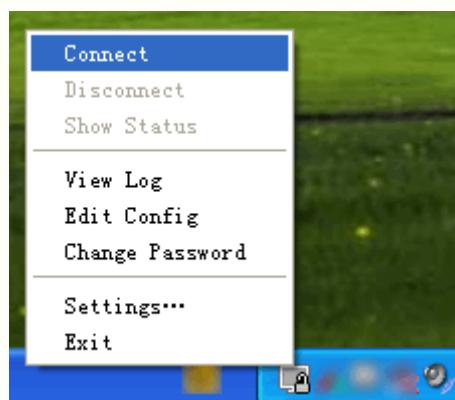
实际上, 将两个模板文件中与 IP 地址有关的配置修改一下, 就可以直接拿来使用。

关于 OpenVPN 配置文件的更多信息请参考 `server.ovpn` 配置详解和 `client.ovpn` 配置详解。

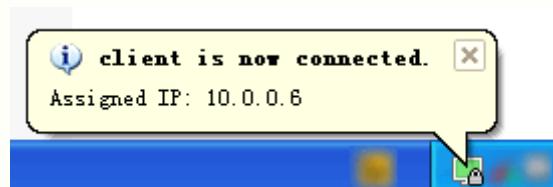
到这里, 我们的 OpenVPN 配置就完成了, 现在该是收获的时候了。

我们先进入服务器端所在计算机的 Windows 服务界面(【开始】->【运行】->【services.msc】), 然后启动 OpenVPN Service 服务, 从而启动服务器端的 OpenVPN。

然后, 我们切换到客户端计算机, 双击安装 OpenVPN 时在桌面上生成的 OpenVPN GUI 图标, 此时任务栏右下角会出现如下图所示的托盘图标, 右键该图标, 点击【connect】即可启动 OpenVPN 客户端, 并尝试连接服务器。

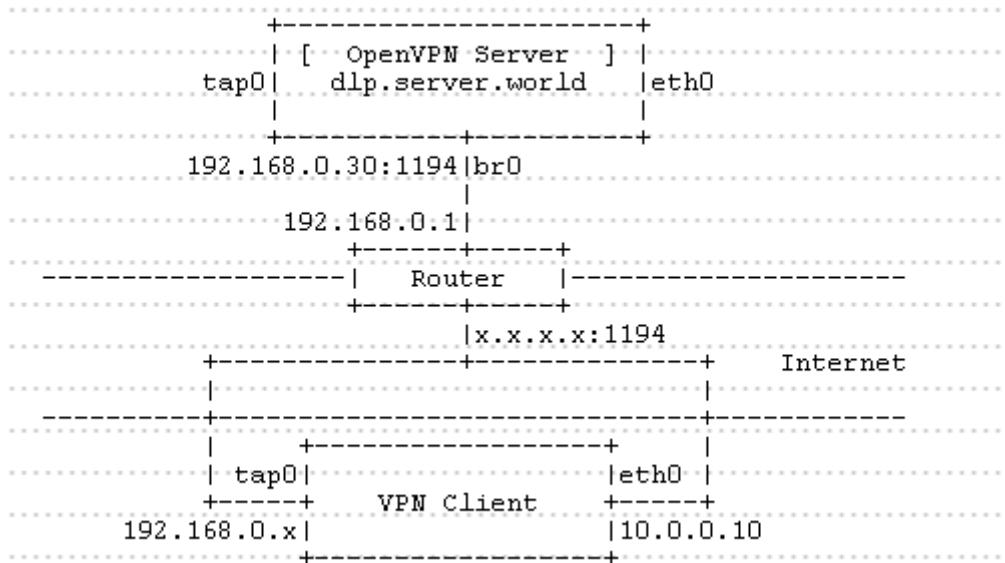


如果连接成功，该托盘图标将会变为绿色，如下图所示：



到这里，我们的 OpenVPN 配置都已经大功告成了~！现在，你就可以使用 10.0.0.* 等虚拟 IP 来访问包括服务器在内的其他 OpenVPN 成员了。

4.8 Linux 版安装配置教程



安装 OpenVPN 的前提条件

在安装 OpenVPN 之前你必须确保你已经安装了 C 编译器(例如 gcc)、OpenSSL、LZO(一种无损压缩算法)、PAM(一种可插入式的身份验证模块)。如果你安装了 yum，可以使用如下命令来安装这些工具(选择性地安装之前没有安装的软件包即可)：

```
#安装 gcc、openssl、lzo、pam  
yum install -y gcc openssl-devel lzo-devel pam-devel
```

Ubuntu、Debian 等系统使用 apt-get 进行安装，安装命令请[参考这里](#)。

安装 Linux 版 OpenVPN

首先，我们需要先下载 OpenVPN 安装程序，上面已经给出了 Linux 版 OpenVPN 的下载地址，在这里我们就不多说了。

在这里，我们将下载的安装文件移动到 /usr/local/ 文件夹中(你也可以自行移动到其他目录)。

```
[root@softown ~]# cd /usr/local/  
[root@softown local]# ls  
bin build etc games include lib lib64 libexec openvpn-2.3.4.tar.gz sbin share src
```

接着，我们使用 **tar** 命令，将该压缩文件解压到当前目录，完整命令为：**tar zxf openvpn-2.3.4.tar.gz**。

```
[root@softown local]# tar zxf openvpn-2.3.4.tar.gz  
[root@softown local]# ls  
bin build etc games include lib lib64 libexec openvpn-2.3.4 openvpn-2.3.4.tar.gz sbin share src
```

接着，我们依次执行如下命令：

```
#跳转到解压后的 openvpn 目录  
cd openvpn-2.3.4  
#调用 configure  
.configure  
#编译  
make  
#安装  
make install
```

OpenVPN 配置

安装 OpenVPN 完毕后，接下来就应该配置 OpenVPN，以便于其能正常工作。配置 OpenVPN 主要有两个步骤：一是生成服务器和客户端所需的各种证书，二是编写服务器和客户端所需的配置文件。

下载 easy-rsa

想要生成各种证书和密钥，我们还要用到 easy-rsa（只有服务器端需要 easy-rsa，客户端无需安装）。坑爹的是，OpenVPN 2.3.x 并没有自带这个东西，我们还需要去 [GitHub 下载 easy-rsa](#)。更坑爹的是，GitHub 上的 easy-rsa 已经升级到了 3.0 版本，该版本几乎重写了之前所有的程序代码，连使用方法都全变了，然而 OpenVPN 的官方文档并没有作相应更新，其中介绍的仍然是 easy-rsa 2.0 的操作方法。为了避免不必要的麻烦，我们推荐下载 2.0 版本的 easy-rsa，你也可以直接[点击这里](#)下载。

在这里我们将下载的 easy-rsa-release-2.x.zip 文件放在 /root 目录中。我们依次执行如下命令：

```
#转到 easy-rsa 安装文件所在目录  
cd /root/  
#解压该安装文件  
unzip -q easy-rsa-release-2.x.zip
```

执行结果如下图所示（其中的 ls 命令用于查看文件列表）：

```
[root@softown openvpn-2.3.4]# cd /root/
[root@softown ~]# ls
anaconda-ks.cfg  easy-rsa-release-2.x.zip  install.log  install.log.syslog 公共的 模
[root@softown ~]# unzip -q easy-rsa-release-2.x.zip
[root@softown ~]# ls
anaconda-ks.cfg  easy-rsa-release-2.x  easy-rsa-release-2.x.zip  install.log  install
[root@softown ~]# ls easy-rsa-release-2.x
configure.ac  COPYING  COPYRIGHT.GPL  distro  doc  easy-rsa  Makefile.am  README
```

接着，我们将上图中所标注的 easy-rsa-release-2.x/easy-rsa 文件夹复制到 OpenVPN 的解压目录中，命令如下：

```
# 复制解压后的 easy-rsa 目录到 OpenVPN 解压目录下
cp -r easy-rsa-release-2.x/easy-rsa /usr/local/openvpn-2.3.4
```

```
[root@softown ~]# cp -r easy-rsa-release-2.x/easy-rsa /usr/local/openvpn-2.3.4
[root@softown ~]# ls /usr/local/openvpn-2.3.4
aclocal.m4  compile  config-msvc.h  configure.ac  debug  include
AUTHORS  config.guess  config-msvc-version.h.in  config-version.h.in  depcomp  INSTALL
build  config.h  config.status  contrib  distro  install-sh
ChangeLog  config.h.in  config.sub  COPYING  doc  INSTALL-win32.t
compat.m4  config.log  configure  COPYRIGHT.GPL  easy-rsa  libtool
```

然后，我们执行命令 `cd /usr/local/openvpn-2.3.4/easy-rsa/2.0` 从而进入 OpenVPN 下的 easy-rsa/2.0 目录。

```
[root@softown ~]# cd /usr/local/openvpn-2.3.4/easy-rsa/2.0
[root@softown 2.0]# ls
build-ca      build-key-server  openssl-0.9.6.cnf  sign-server-req
build-dh      build-req        openssl-0.9.8.cnf  vars
build-inter   build-req-pass  openssl-1.0.0.cnf  whichopensslcnf
build-key     clean-all       pktool
build-key-pass inherit-inter  revoke-full
build-key-pkcs12 list-crl     sign-req
```

上面说了这么多，实际上就只是下载了 easy-rsa 2.0，并将解压后的 easy-rsa 子目录复制到了 OpenVPN 的主目录下。这里的文件夹 2.0 就是我们以后生成各种证书和密钥的数据地了。

使用 easy-rsa 生成 CA 证书

在生成证书之前，我们建议你对 2.0 目录中的 vars 文件稍作修改。vars 文件存储的是一些用户变量设置信息，每次生成证书都会使用到其中的某些变量。如下图所示，我们着重建议你关注红色矩形框内的变量，并选择性地对其进行修改（你可以不修改这些参数，但不要把这些参数留空）。

KEY_SIZE：表示密钥的长度，一般为 1024 或 2048（长度越长，性能耗费越多）。

#下面是一些用户相关信息配置

KEY_COUNTRY: 所在国家

KEY_PROVINCE: 所在省

KEY_CITY: 所在城市

KEY_ORG: 所在组织

KEY_EMAIL: 邮箱地址

KEY_OU: 机构单位或部门名称

```
# are paranoid. This will slow
# down TLS negotiation performance
# as well as the one-time DH parms
# generation process.
export KEY_SIZE=2048

# In how many days should the root CA key expire?
export CA_EXPIRE=3650

# In how many days should certificates expire?
export KEY_EXPIRE=3650

# These are the default values for fields
# which will be placed in the certificate.
# Don't leave any of these fields blank.
export KEY_COUNTRY="US"
export KEY_PROVINCE="CA"
export KEY_CITY="SanFrancisco"
export KEY_ORG="Fort-Funston"
export KEY_EMAIL="me@myhost.mydomain"
export KEY_OU="MyOrganizationalUnit"

# X509 Subject Field
# export KEY_NAME="EasyRSA"

# PKCS11 Smart Card
# export PKCS11_MODULE_PATH="/usr/lib/changeme.so"
# export PKCS11_PIN=1234
```

下面，我们就开始来生成证书了。保持当前目录为 OpenVPN 根目录/easy-rsa/2.0。然后依次执行下列命令：

```
#初始化命令，用于设置后续命令所需的相关变量信息
```

```
./vars
```

```
#清除之前创建的所有证书和密钥
```

```
./clean-all
```

```
#生成 CA 证书和密钥
```

```
./build-ca
```

注意：证书的用户信息可以根据需要自行输入。如果不输入、直接回车，则表示该字段使用 "[]" 中的默认值(也就是前面 vars 文件中设置的参数值)；如果输入". ."，则表示该字段留空。在这里需要注意 Common Name 字段，这相当于证书的"用户名"，请确保每个证书的 Common Name 字段是唯一的。

```
[root@softown 2.0]# ./vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on /usr/local/openvpn-2.3.4/easy-rsa/2.0/keys
[root@softown 2.0]# ./clean-all
[root@softown 2.0]# ./build-ca
Generating a 2048 bit RSA private key
.....+++
.....+++
writing new private key to 'ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:CN      输入该证书的用户信息
State or Province Name (full name) [CA]:GD
Locality Name (eg, city) [SanFrancisco]:SZ
Organization Name (eg, company) [Fort-Funston]:softown
Organizational Unit Name (eg, section) [MyOrganizationalUnit]:softown_admin
Common Name (eg, your name or your server's hostname) [Fort-Funston CA]:OpenVPN_CA
Name [EasyRSA]:
Email Address [me@myhost.mydomain]:admin@softown.cn
[root@softown 2.0]# ls keys
ca.crt  ca.key  index.txt  serial 生成的CA证书和密钥
```

到这里，我们的 CA 证书和密钥就已经生成成功了，生成的证书和密码默认均存放在当前目录的子文件夹 keys 中。

生成服务器端证书

接下来，我们为服务器和客户端生成各自所需的证书和密钥（所有的证书和密钥都必须由 OpenVPN 服务器上的 easy-rsa 生成）。

我们可以执行命令 `./build-key-server server` 来生成服务器端所需的证书和密钥。

如下图所示，与创建 CA 证书一样，我们先输入证书的相关信息，并在最后输入两次 "y" 确认生成即可。

```
[root@softown 2.0]# ./build-key-server server
Generating a 2048 bit RSA private key
.....+ ++
.....+ ++
writing new private key to 'server.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:CN
State or Province Name (full name) [CA]:GD
Locality Name (eg, city) [SanFrancisco]:SZ
Organization Name (eg, company) [Fort-Funston]:sofr^H^H^H^H^H^H^H
Organizational Unit Name (eg, section) [MyOrganizationalUnit]:softown_admin
Common Name (eg, your name or your server's hostname) [server]:server
Name [EasyRSA]:
Email Address [me@myhost.mydomain]:Common Name

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:server_pwd 密码
An optional company name []:softown
Using configuration from /usr/local/openvpn-2.3.4/easy-rsa/2.0/openssl-1.0.0.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName :PRINTABLE:'CN'
stateOrProvinceName :PRINTABLE:'GD'
localityName :PRINTABLE:'SZ'
organizationName :T61STRING:'sofr^H^H^H^H^H^H^H^H'
organizationalUnitName:T61STRING:'softown_admin'
commonName :PRINTABLE:'server'
name :PRINTABLE:'EasyRSA'
emailAddress :IA5STRING:'me@myhost.mydomain'
Certificate is to be certified until May 6 08:10:35 2024 GMT (3650 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

生成客户端证书

与创建服务器端证书类似，我们可以使用命令 `./build-key clientName` 来生成客户端证书和密钥，其中 `clientName` 为自定义的客户端名称（例如：`client1`、`client2`、`jim`、`tom`）。如果需要为多个客户端生成证书，只需要分别执行多次即可。

```
[root@softown 2.0]# ./build-key client1
Generating a 2048 bit RSA private key
.....+ ++
.....+
writing new private key to 'client1.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:
State or Province Name (full name) [CA]:
Locality Name (eg, city) [SanFrancisco]:
Organization Name (eg, company) [Fort-Funston]:
Organizational Unit Name (eg, section) [MyOrganizationalUnit]:
Common Name (eg, your name or your server's hostname) [client1]:
Name [EasyRSA]:
Email Address [me@myhost.mydomain]:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:client1_pwd
An optional company name []:softown
Using configuration from /usr/local/openvpn-2.3.4/easy-rsa/2.0/openssl-1.0.0.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName :PRINTABLE:'US'
stateOrProvinceName :PRINTABLE:'CA'
localityName :PRINTABLE:'SanFrancisco'
organizationName :PRINTABLE:'Fort-Funston'
organizationalUnitName:PRINTABLE:'MyOrganizationalUnit'
commonName :PRINTABLE:'client1'
name :PRINTABLE:'EasyRSA'
emailAddress :IA5STRING:'me@myhost.mydomain'
Certificate is to be certified until May 6 08:18:41 2024 GMT (3650 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

生成迪菲·赫尔曼交换密钥

此外，我们还需要为 OpenVPN 的服务器端创建迪菲·赫尔曼交换密钥，命令为 `./build-dh`（无需额外输入，耐心等待生成完毕即可）。迪菲·赫尔曼交换密钥是一种安全协议，用以对数据进行加密。

```
[root@softown 2.0]# ./build-dh
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
.....
+.....+
.....+
.....+
```

生成 TLS-auth 密钥

这一步骤是可选操作。OpenVPN 提供了 TLS-auth 功能，可以用来抵御 Dos、UDP 端口淹没攻击。出于安全考虑，你可以启用该功能；启用该功能，你需要执行命令 `openvpn --genkey --secret keys/ta.key` 来生成 TLS-auth 所需的密钥文件。

到这里，我们的证书生成就告一段落了。如果你以后想要生成新的客户端或执行其他操作，只需要先执行命令 `./vars`，然后执行相应的命令即可，例如 `./build-key client2`。

最后，我们来看看我们一共生成了哪些证书和密钥。

编号①：CA 证书和密钥

编号②：客户端 client1 的证书和密钥，

编号③：迪菲·赫尔曼交换密钥（如果你的 KEY_SIZE=1024，则该文件名称为 dh1024.pem）。

编号④：服务器端证书和密钥。

编号⑤：启用 tls-auth 所需的文件。

```
[root@softown 2.0]# ls keys ②
01.pem ① ca.crt client1.crt client1.key index.txt      index.txt.attr.old serial ④ server.crt
02.pem ca.key client1.csr dh2048.pem ③ index.txt.attr index.txt.old    serial.old server.csr
```

编写配置文件

众所周知，OpenVPN 虽然可以分为客户端和服务器，不过它们的安装程序是完全一样的，只是通过不同的证书和配置文件来进行区分。在这里，我们先在 OpenVPN 主目录下创建一个 config 目录，并将其所需的证书和密钥文件拷贝到该目录中。

其中，服务器端需要用到的文件有：

```
ca.crt
ca.key
dh2048.pem (如果最初的变量 KEY_SIZE 设为 1024，这里就是 dh1024.pem)
server.crt
server.key
ta.key (如果不开启 tls-auth，则无需该文件)
```

客户端 client1 需要用到的文件有：

```
ca.crt
client1.crt
client1.key (名称 client1 根据个人设置可能有所不同)
ta.key (如果不开启 tls-auth, 则无需该文件)
```

在这里, 我们以 OpenVPN 服务器端为例来演示上述操作流程。

```
#创建 config 目录
mkdir /usr/local/openvpn-2.3.4/config
#复制证书和密钥文件到 config 目录
cp keys/ca.crt keys/ca.key keys/server.crt keys/server.key keys/dh2048.pem keys/ta.key
/usr/local/openvpn-2.3.4/config
```

```
[root@softown 2.0]# mkdir /usr/local/openvpn-2.3.4/config
[root@softown 2.0]# cp keys/ca.crt keys/ca.key keys/server.crt keys/server.key keys/dh2048.pem keys/ta.key /usr/local/openvpn-2.3.4/config
[root@softown 2.0]# ls /usr/local/openvpn-2.3.4/config
ca.crt ca.key dh2048.pem server.crt server.key ta.key
```

此外, 我们还需要为服务器和每个客户端的 config 目录分别编写一个配置文件, 服务器端的配置文件为 `server.conf`, 客户端的配置文件为 `client.conf`。

这两个配置文件该如何编写呢? OpenVPN 已经在 `sample/sample-config-files` 子目录中为我们提供了相关的示例文件 `server.conf` 和 `client.conf`, 并且配置文件中的每个配置选项均有详细的英文说明(配置文件中 "#" 或 ";" 开头的均为注释内容)。

```
[root@softown 2.0]# ls /usr/local/openvpn-2.3.4/sample/sample-config-files/
client.conf      loopback-server      README          tls-home.conf
firewall.sh      office.up          server.conf      tls-office.conf
home.up          openvpn-shutdown.sh  static-home.conf xinetd-client-config
loopback-client  openvpn-startup.sh  static-office.conf xinetd-server-config
```

现在, 我们先将 `server.conf` 文件拷贝到 config 目录中, 然后再对其进行修改。

```
#转到 sample-config-files 目录
cd /usr/local/openvpn-2.3.4/sample/sample-config-files
#复制 server.conf 到 config 目录中
cp server.conf /usr/local/openvpn-2.3.4/config
```

```
[root@softown 2.0]# cd /usr/local/openvpn-2.3.4/sample/sample-config-files/
[root@softown sample-config-files]# ls
client.conf  home.up      loopback-server  openvpn-shutdown.sh  README      static-home.conf  tls-home.conf  xinetd-client-config
firewall.sh  loopback-client  office.up      openvpn-startup.sh  server.conf  static-office.conf  tls-office.conf  xinetd-server-config
[root@softown sample-config-files]# cp server.conf /usr/local/openvpn-2.3.4/config
```

在这里, 我们先给出 `server.conf` 的详细配置, 并注明每项配置的作用。

```
local 192.168.1.106      #指定监听的本机 IP(因为有些计算机具备多个 IP 地址), 该命令是可选的, 默认监听所有 IP 地址。
```

```

port 1194          #指定监听的本机端口号
proto udp          #指定采用的传输协议，可以选择 tcp 或 udp
dev tun            #指定创建的通信隧道类型，可选 tun 或 tap
ca ca.crt          #指定 CA 证书的文件路径
cert server.crt    #指定服务器端的证书文件路径
key server.key     #指定服务器端的私钥文件路径
dh dh2048.pem      #指定迪菲赫尔曼参数的文件路径
server 10.0.0.0 255.255.255.0  #指定虚拟局域网占用的 IP 地址段和子网掩码，此处配置的服务器自身占用
10.0.0.1。
ifconfig-pool-persist ipp.txt  #服务器自动给客户端分配 IP 后，客户端下次连接时，仍然采用上次的 IP
地址(第一次分配的 IP 保存在 ipp.txt 中，下一次分配其中保存的 IP)。
tls-auth ta.key 0      #开启 TLS-auth，使用 ta.key 防御攻击。服务器端的第二个参数值为 0，客户端的为 1。
keepalive 10 120       #每 10 秒 ping 一次，连接超时时间设为 120 秒。
comp-lzo              #开启 VPN 连接压缩，如果服务器端开启，客户端也必须开启
client-to-client       #允许客户端与客户端相连接，默认情况下客户端只能与服务器相连接
persist-key           #持久化选项可以尽量避免访问在重启时由于用户权限降低而无法访问的某些资源。
persist-tun            #持久化选项可以尽量避免访问在重启时由于用户权限降低而无法访问的某些资源。
status openvpn-status.log  #指定记录 OpenVPN 状态的日志文件路径
verb 3                #指定日志文件的记录详细级别，可选 0-9，等级越高日志内容越详细

```

接着是客户端配置文件 `client.conf`。

```

client          #指定当前 VPN 是客户端
dev tun         #必须与服务器端的保持一致
proto udp       #必须与服务器端的保持一致
remote 192.168.1.106 1194  #指定连接的远程服务器的实际 IP 地址和端口号
resolv-retry infinite  #断线自动重新连接，在网络不稳定的情况下(例如：笔记本电脑无线网络)非常有用。
nobind          #不绑定特定的本地端口号
persist-key
persist-tun
ca ca.crt        #指定 CA 证书的文件路径
cert client1.crt #指定当前客户端的证书文件路径
key client1.key  #指定当前客户端的私钥文件路径
ns-cert-type server #指定采用服务器校验方式
tls-auth ta.key 1  #如果服务器设置了防御 DoS 等攻击的 ta.key，则必须每个客户端开启；如果未设置，则
注释掉这一行；
comp-lzo          #与服务器保持一致
verb 3            #指定日志文件的记录详细级别，可选 0-9，等级越高日志内容越详细

```

实际上，将两个模板文件中与 IP 地址有关的配置修改一下，就可以直接拿来使用。
关于 OpenVPN 配置文件的更多信息请参考 [server.conf 配置详解](#) 和 [client.conf 配置详解](#)。
启动 OpenVPN

当我们把服务器和客户端所需的证书、密钥和配置文件都分配完毕之后，我们就可以

尝试启动 OpenVPN 来检查我们的工作成果了。

在 Linux 中，我们可以直接执行以下命令来启动 OpenVPN：

openvpn 配置文件路径

如果你是服务器端，就指定 `server.conf` 文件的路径，如果你是客户端就指定 `client.conf` 文件的路径。

请注意：

配置文件中的文件路径涉及到相对路径的，均以启动 OpenVPN 时的所在目录为准。由于我们在配置文件中设置的文件路径都是相对 `config` 目录的路径，因此我们也只能在 `config` 目录下才能正常启动 OpenVPN。如果你想在任何地方都能使用上述命令启动 OpenVPN，建议你将配置文件与文件路径相关的部分全部改为绝对路径。

OpenVPN 服务器所在计算机必须允许 OpenVPN 通过防火墙，你可以禁用掉防火墙，或者将 OpenVPN 设为可信程序，或者开放 1194 端口。

启动服务器和客户端都需要一定的权限，建议用 `root` 账户或 `sudo` 命令进行启动。

以下就是 OpenVPN 服务器的启动效果：

```
[root@softown config]# openvpn server.conf &
[1] 31975
[root@softown config]# Fri May  9 20:41:35 2014 OpenVPN 2.3.4 x86_64-unknown-linux-gnu [SSL: RSA
Fri May  9 20:41:35 2014 library versions: OpenSSL 1.0.1e-fips 11 Feb 2013, LZO 2.03
Fri May  9 20:41:35 2014 NOTE: your local LAN uses the extremely common subnet address 192
m public locations such as internet cafes that use the same subnet.
Fri May  9 20:41:35 2014 Diffie-Hellman initialized with 2048 bit key
Fri May  9 20:41:35 2014 Control Channel Authentication: using 'ta.key' as a OpenVPN static
Fri May  9 20:41:35 2014 Outgoing Control Channel Authentication: Using 160 bit message ha
Fri May  9 20:41:35 2014 Incoming Control Channel Authentication: Using 160 bit message ha
Fri May  9 20:41:35 2014 Socket Buffers: R=[124928->131072] S=[124928->131072]
Fri May  9 20:41:35 2014 ROUTE_GATEWAY 192.168.1.1/255.255.255.0 IFACE=eth0 HWADDR=00:0c:2
Fri May  9 20:41:35 2014 TUN/TAP device tun0 opened
Fri May  9 20:41:35 2014 TUN/TAP TX queue length set to 100
Fri May  9 20:41:35 2014 do_ifconfig, tt->ipv6=0, tt->did_ifconfig_ipv6_setup=0
Fri May  9 20:41:35 2014 /sbin/ifconfig tun0 10.0.0.1 pointopoint 10.0.0.2 mtu 1500
Fri May  9 20:41:35 2014 /sbin/route add -net 10.0.0.0 netmask 255.255.255.0 gw 10.0.0.2
Fri May  9 20:41:35 2014 UDPv4 link local (bound): [undef]
Fri May  9 20:41:35 2014 UDPv4 link remote: [undef]
Fri May  9 20:41:35 2014 MULTI: multi_init called, r=256 v=256
Fri May  9 20:41:35 2014 IFCONFIG POOL: base=10.0.0.4 size=62, ipv6=0
Fri May  9 20:41:35 2014 IFCONFIG POOL LIST
Fri May  9 20:41:35 2014 Initialization Sequence Completed
```

无论是服务器还是
客户端，启动时看
到这条信息一般就
表示启动成功了

客户端的启动效果如下：

```
softown@SoftownHost:~/openvpn-2.3.4/config$ sudo openvpn client.conf &
[1] 31359
softown@SoftownHost:~/openvpn-2.3.4/config$ Fri May  9 21:32:54 2014 OpenVPN 2.3.4 x86_64
Fri May  9 21:32:54 2014 library versions: OpenSSL 1.0.1f 6 Jan 2014, LZO 2.06
Fri May  9 21:32:54 2014 WARNING: file 'client1.key' is group or others accessible
Fri May  9 21:32:54 2014 WARNING: file 'ta.key' is group or others accessible
Fri May  9 21:32:54 2014 Control Channel Authentication: using 'ta.key' as a OpenVPN stat
Fri May  9 21:32:54 2014 Outgoing Control Channel Authentication: Using 160 bit message h
Fri May  9 21:32:54 2014 Incoming Control Channel Authentication: Using 160 bit message h
Fri May  9 21:32:54 2014 Socket Buffers: R=[212992->131072] S=[212992->131072]
Fri May  9 21:32:54 2014 UDPv4 link local: [undef]
Fri May  9 21:32:54 2014 UDPv4 link remote: [AF_INET]192.168.1.106:1194
Fri May  9 21:32:54 2014 TLS: Initial packet from [AF_INET]192.168.1.106:1194, sid=5c3f18
Fri May  9 21:32:54 2014 VERIFY OK: depth=1, C=CN, ST=GD, L=SZ, O=softown, OU=softown_adm
Fri May  9 21:32:54 2014 VERIFY OK: nsCertType=SERVER
Fri May  9 21:32:54 2014 VERIFY OK: depth=0, C=CN, ST=GD, L=SZ, O=soft\08\08\08\08\08\08\
Fri May  9 21:32:54 2014 Data Channel Encrypt: Cipher 'BF-CBC' initialized with 128 bit k
Fri May  9 21:32:54 2014 Data Channel Encrypt: Using 160 bit message hash 'SHA1' for HMAC
Fri May  9 21:32:54 2014 Data Channel Decrypt: Cipher 'BF-CBC' initialized with 128 bit k
Fri May  9 21:32:54 2014 Data Channel Decrypt: Using 160 bit message hash 'SHA1' for HMAC
Fri May  9 21:32:54 2014 Control Channel: TLSv1, cipher TLSv1/SSLv3 DHE-RSA-AES256-SHA, 2
Fri May  9 21:32:54 2014 [server] Peer Connection Initiated with [AF_INET]192.168.1.106:1
Fri May  9 21:32:56 2014 SENT CONTROL [server]: 'PUSH_REQUEST' (status=1)
Fri May  9 21:32:56 2014 PUSH: Received control message: 'PUSH_REPLY', route 10.0.0.0 255.2
Fri May  9 21:32:56 2014 OPTIONS IMPORT: timers and/or timeouts modified
Fri May  9 21:32:56 2014 OPTIONS IMPORT: --ifconfig/up options modified
Fri May  9 21:32:56 2014 OPTIONS IMPORT: route options modified
Fri May  9 21:32:56 2014 ROUTE_GATEWAY 192.168.1.1/255.255.255.0 IFACE=eth0 HWADDR=00:0c:
Fri May  9 21:32:56 2014 TUN/TAP device tun0 opened
Fri May  9 21:32:56 2014 TUN/TAP TX queue length set to 100
Fri May  9 21:32:56 2014 do_ifconfig, tt->ipv6=0, tt->did_ifconfig_ipv6_setup=0
Fri May  9 21:32:56 2014 /sbin/ifconfig tun0 10.0.0.6 pointopoint 10.0.0.5 mtu 1500
Fri May  9 21:32:56 2014 /sbin/route add -net 10.0.0.0 netmask 255.255.255.0 gw 10.0.0.5
Fri May  9 21:32:56 2014 Initialization Sequence Completed
```

我们在客户端尝试 **ping** 服务器的虚拟 IP 地址 10.0.0.1，顺利 **ping** 通。

```
softown@SoftownHost:~/openvpn-2.3.4/config$ ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.725 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.609 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.603 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.617 ms
```

=====第二版本=====

第二版本地址：http://www.server-world.info/en/note?os=CentOS_7&p=openvpn&f=1
可供参考

五、DNS 服务器安装配置

DNS：域名解析系统

A 记录：就是主机名对 IP 的映射。

CNAME 别名解析：就是 A 记录的别名。

MX 记录：邮件交换器，专门给邮件使用。

NS 名称服务器：就是标识 DNS 服务器的名称、标识 DNS 服务器的特有名称。

SOA：起始授权机构，就是该域的起始的权威 DNS 服务器，提供该域解析的初始 DNS 服务器。

序列号：就是记录当前 DNS 服务器的状态 ID，表示当前 DNS 的 ID。

主服务器：主服务器一定是以. 为结尾的，表示主服务器，表示记录 ns1.rdh.com. admin.rdh.com.

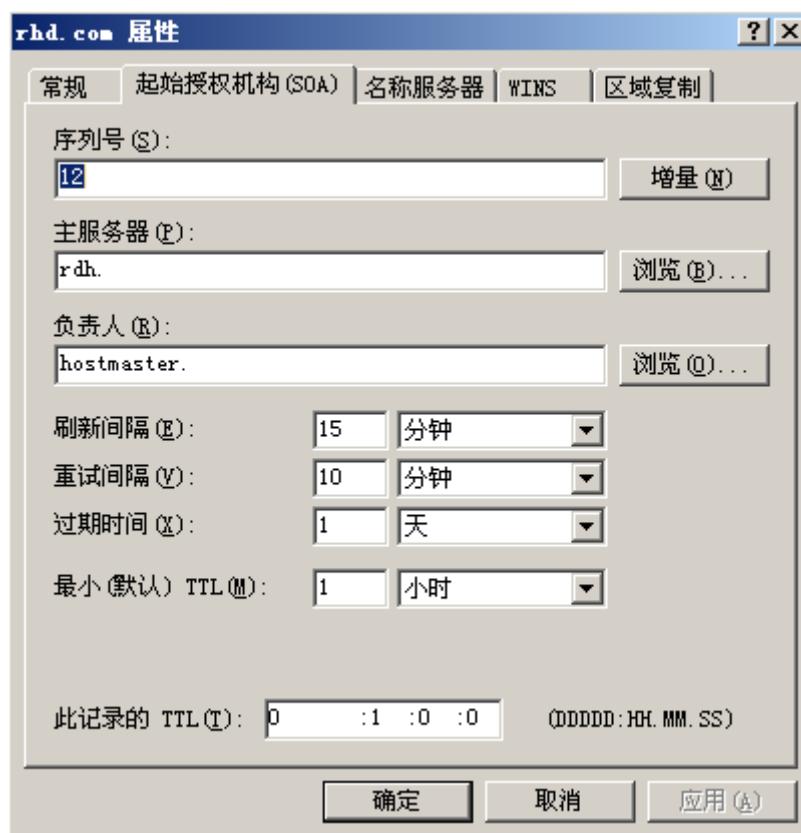
刷新时间：就是多长时间刷新这个状态表，当刷新后主服务器和辅助服务器的状态表进行对比，来进行区域传送。

重试间隔：就是进行区域传送失败后，就是重新进行区域传送时间。

过期时间：如果一天，还是没有完成区域传送，我们就认为是主服务器不可用。

TTL 生存时间：就是缓存时间。

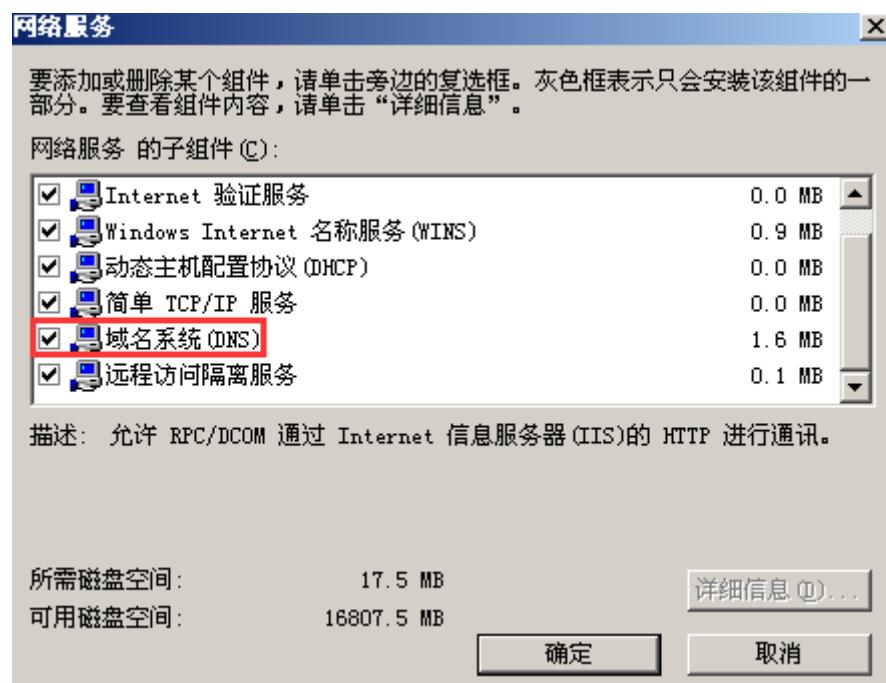
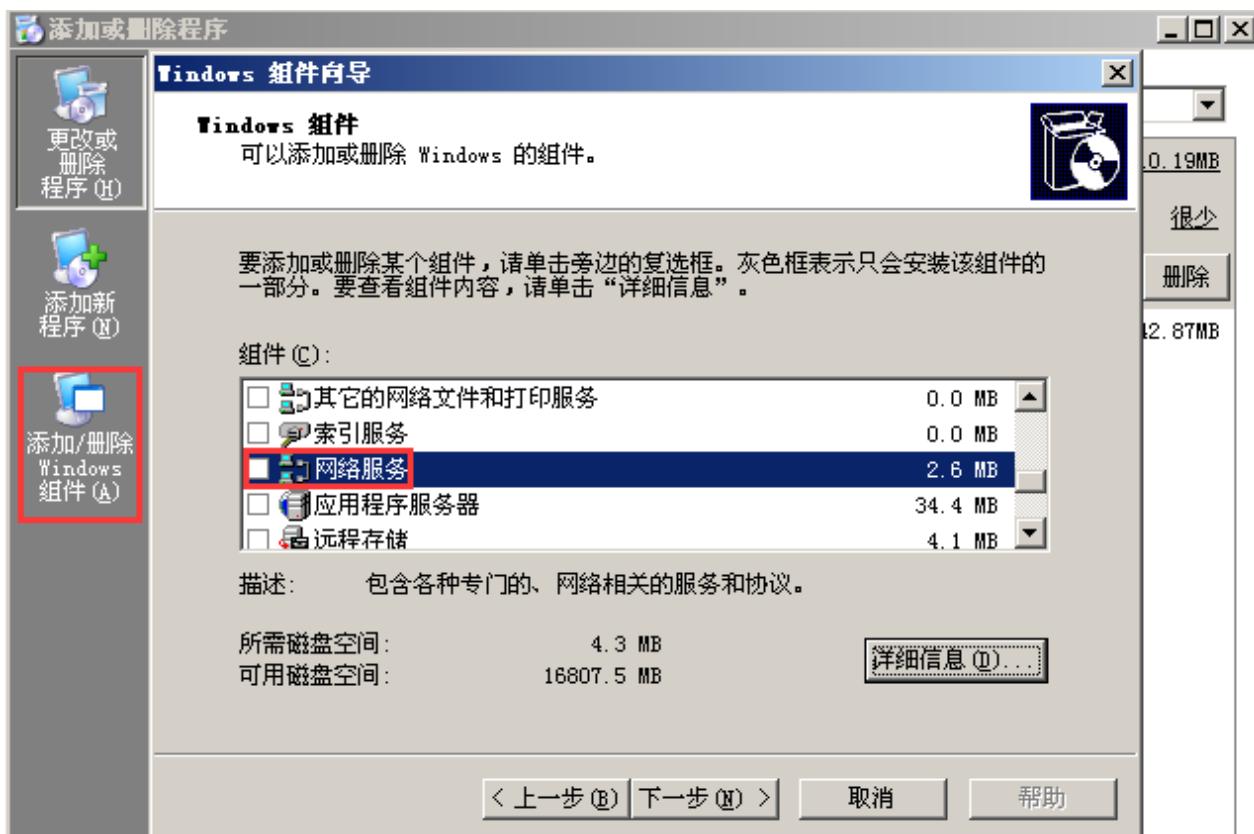
WINS 服务器：就是局域网计算机名称解析，NETBIOS 就是计算机名和 IP 的对应。

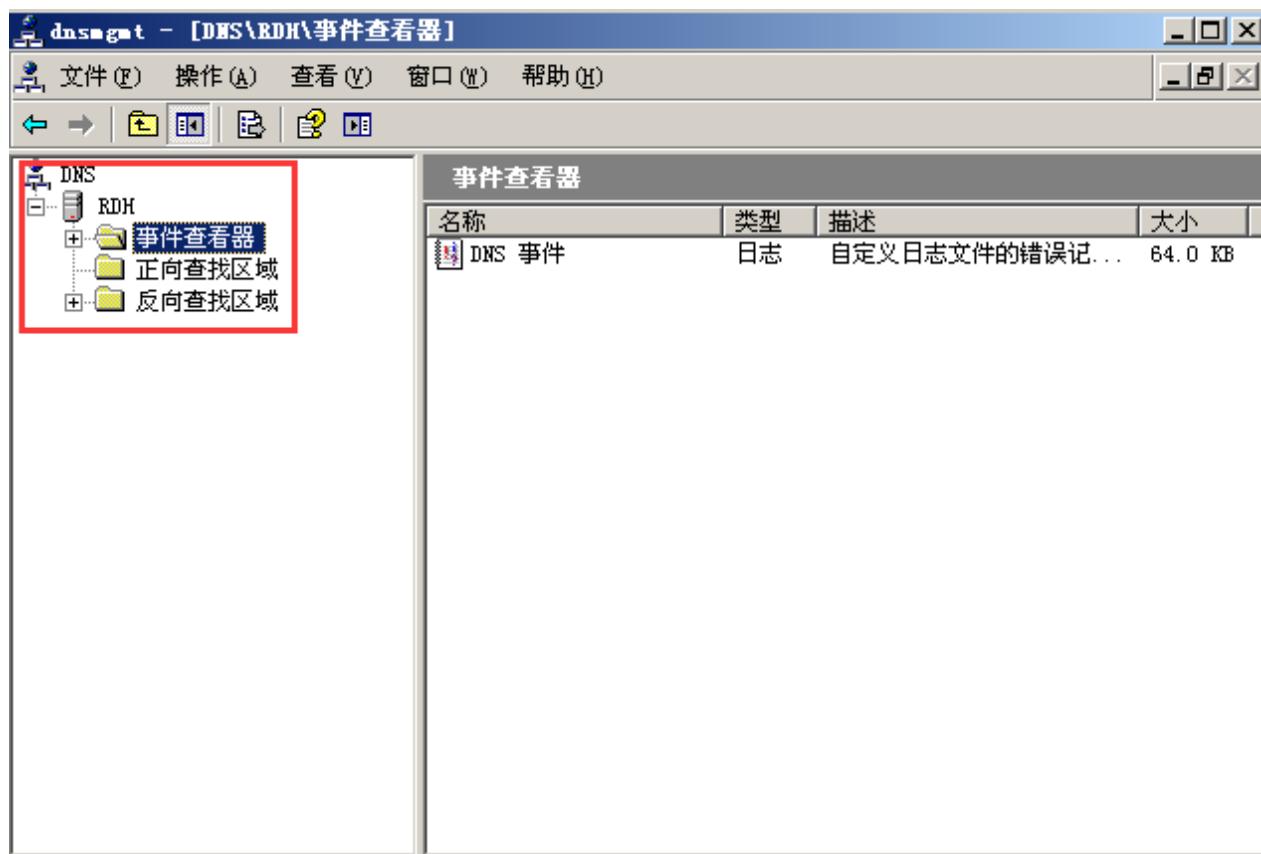


DNS 解析原理：一层一层递归解析，可以使用命令：dig +trace www.baidu.com

5.1 Windows DNS 服务器安装配置篇

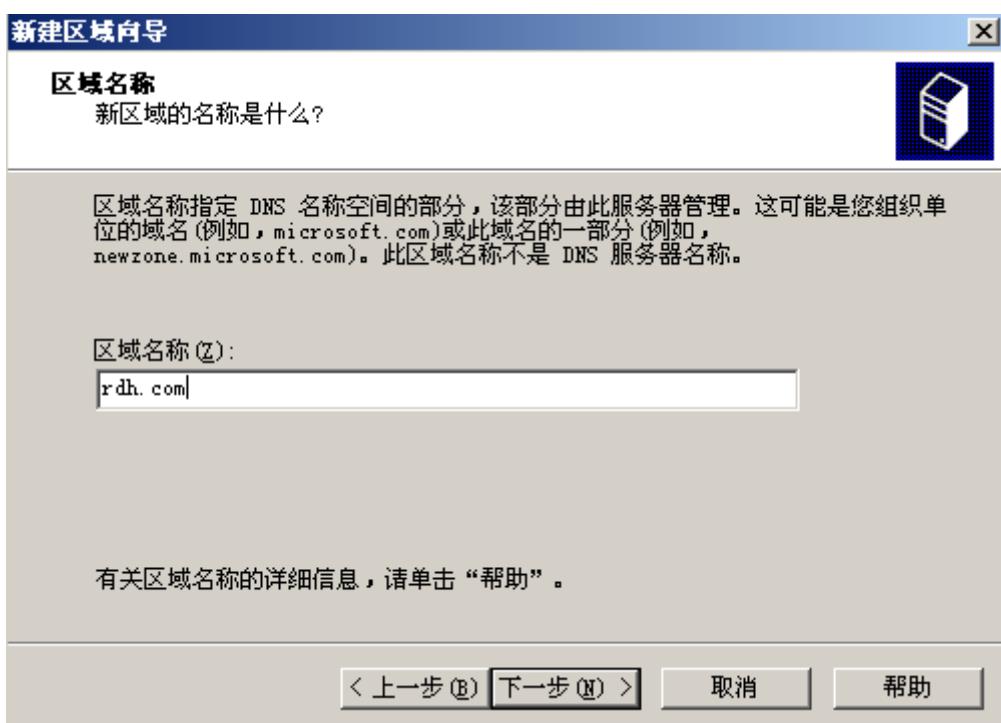
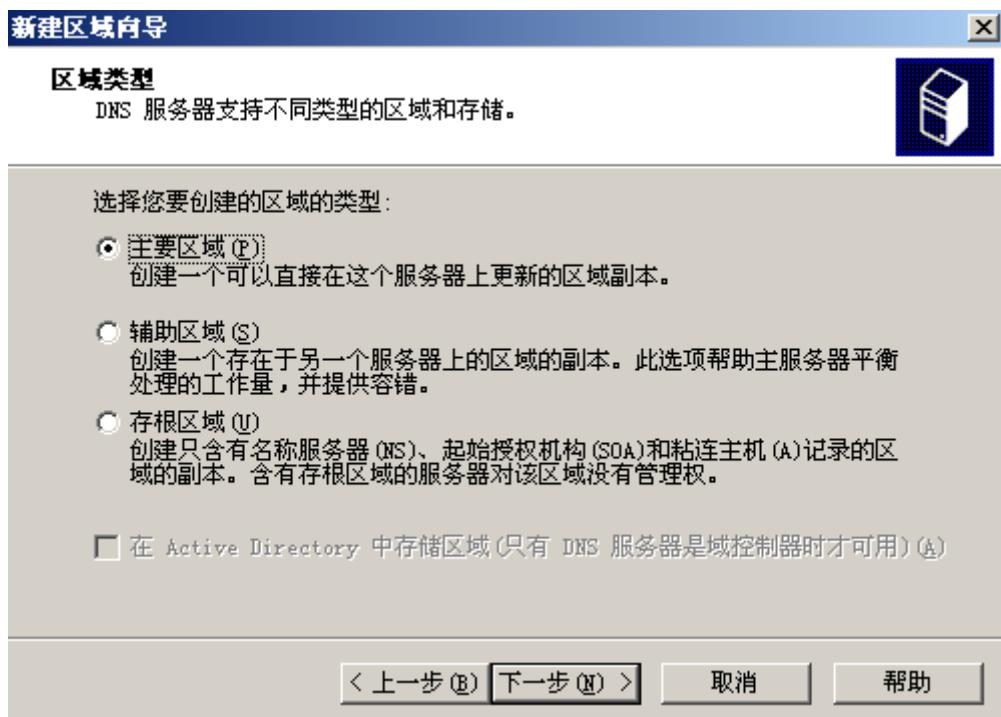
一、安装服务器组件、在控制面板、添加删除程序、添加删除组件、网络服务、添加 DNS 域名系统。

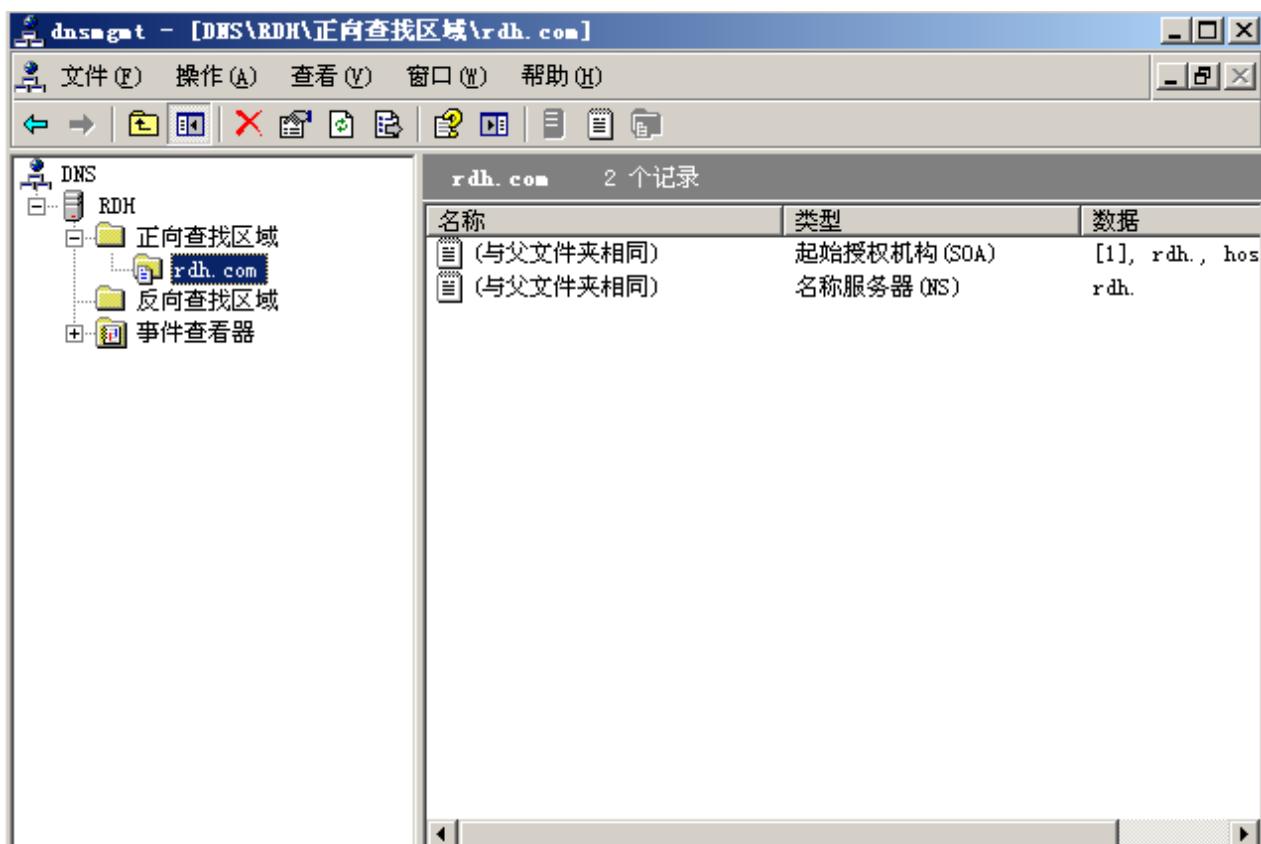
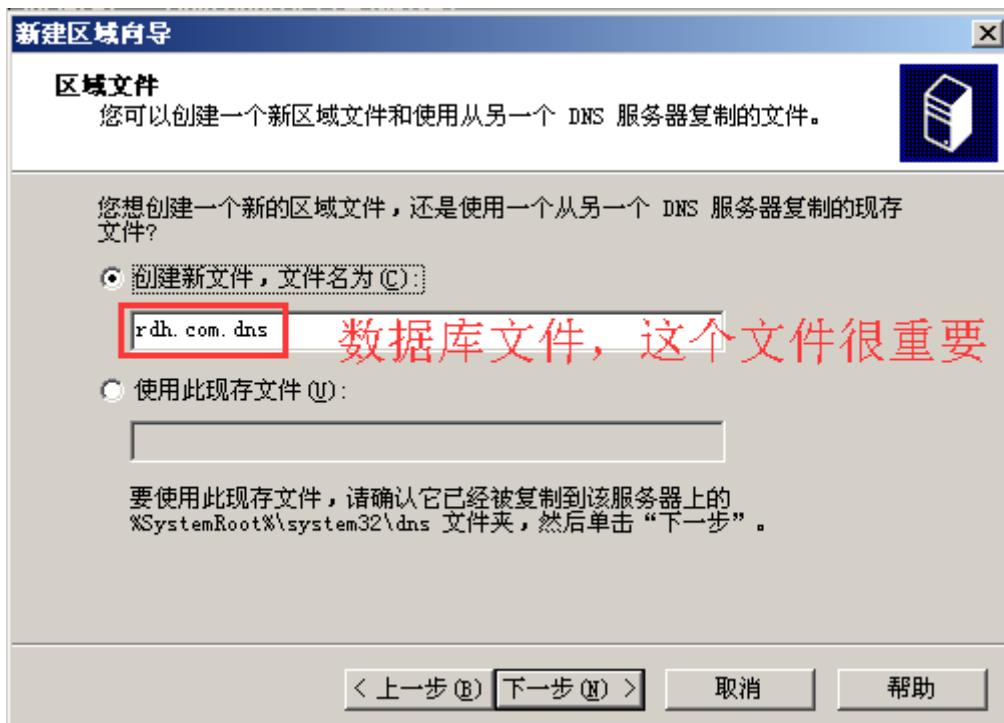




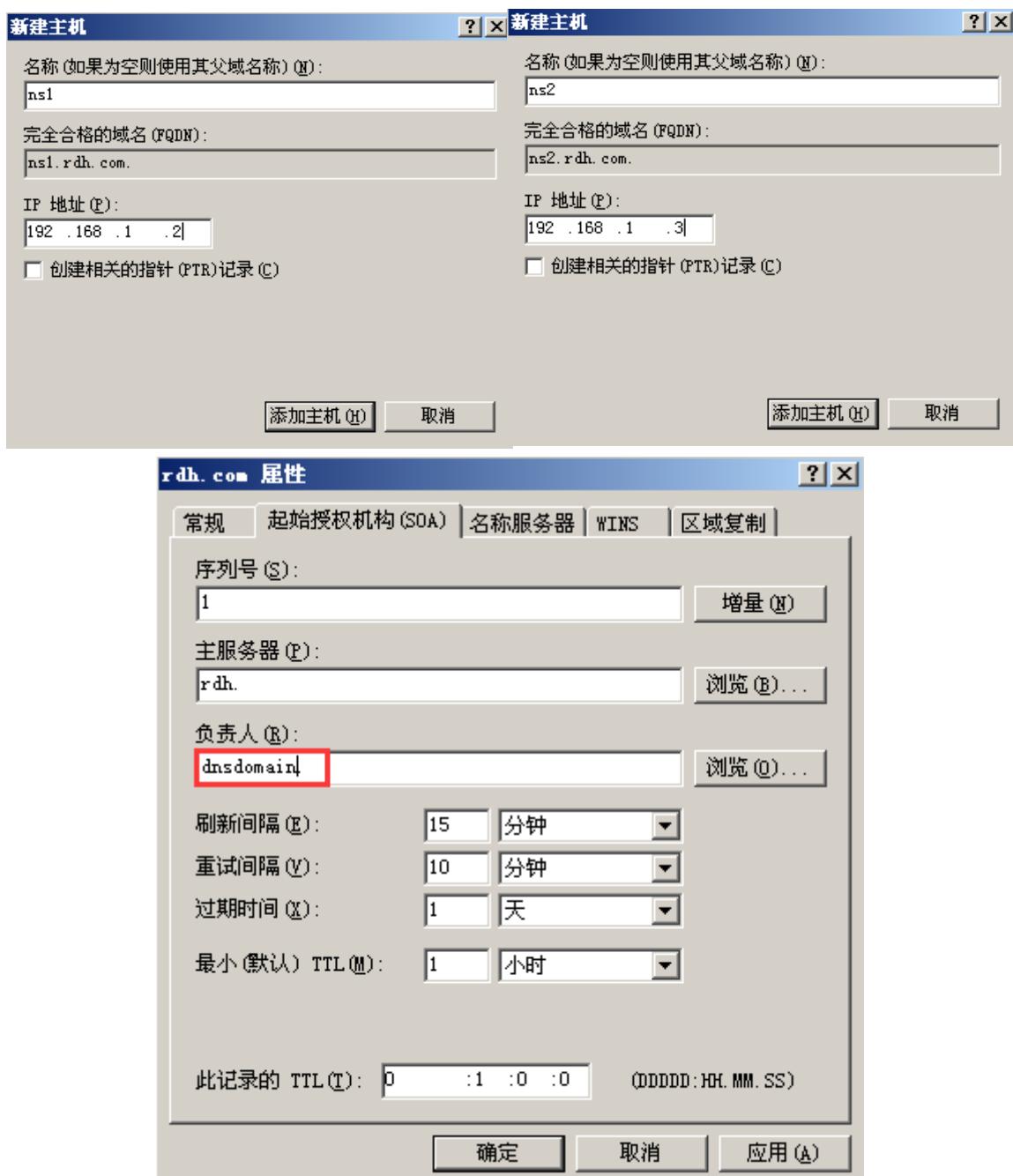
二、创建正向、反向查找区域

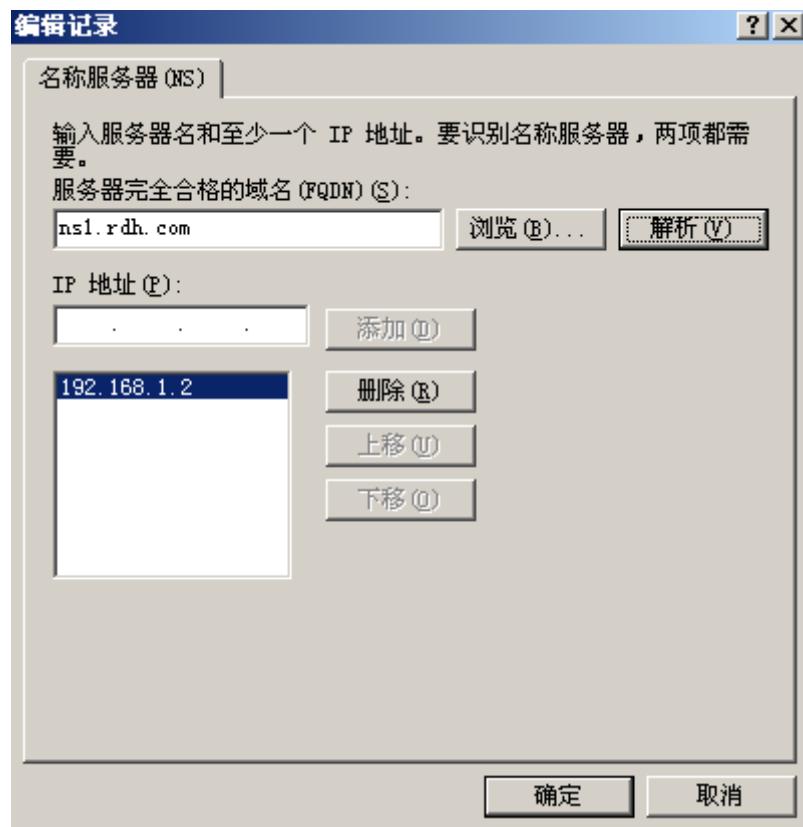


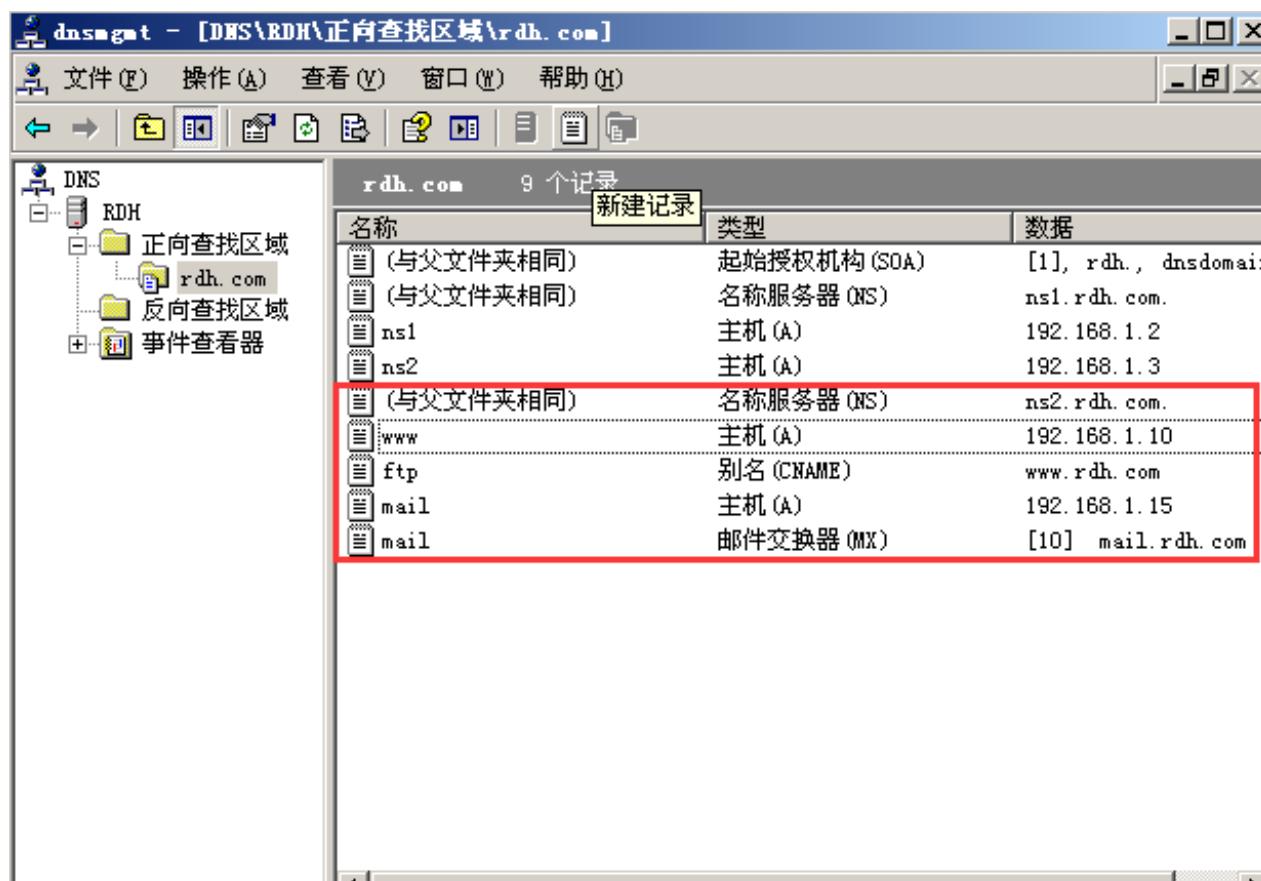
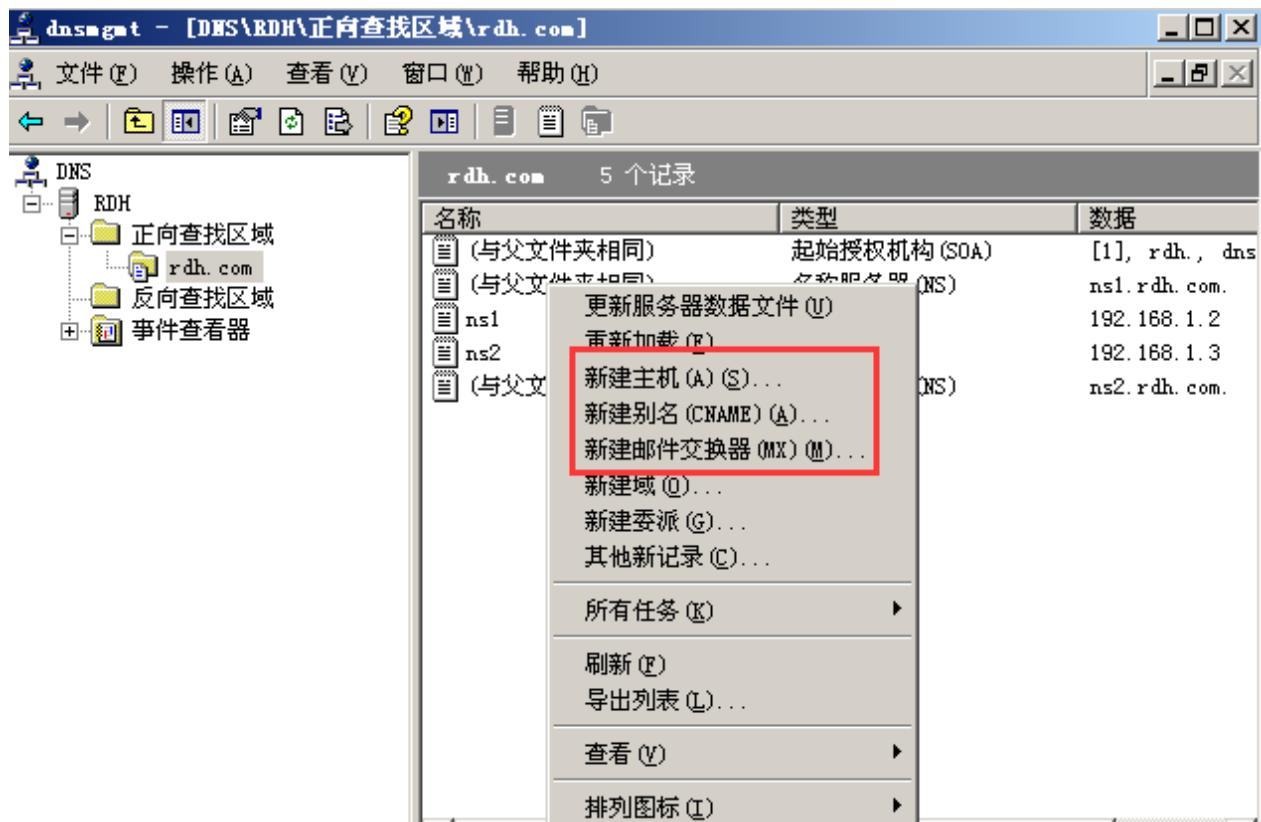




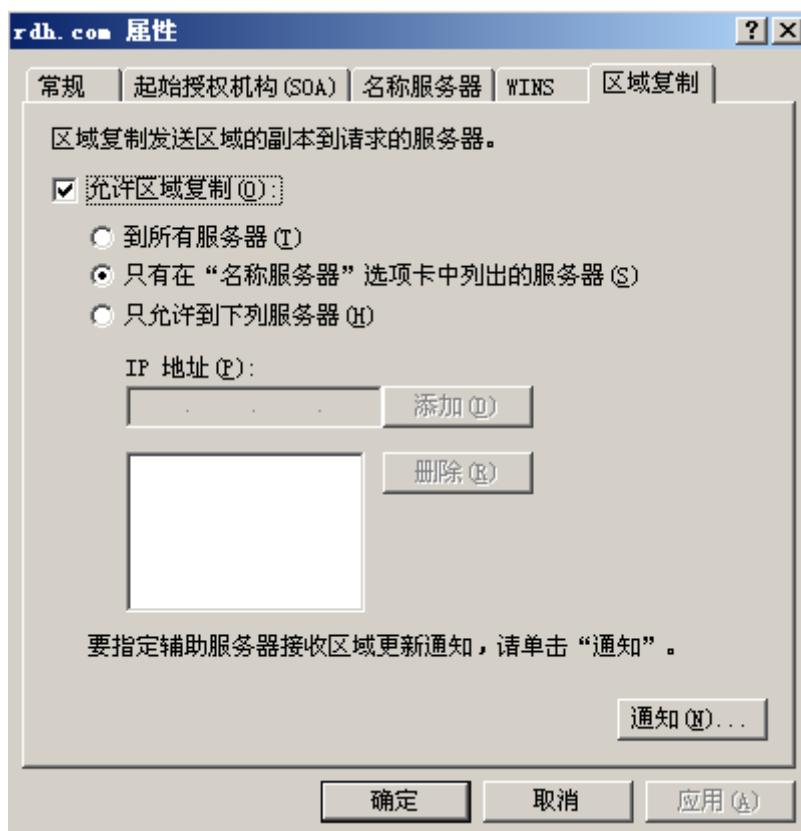
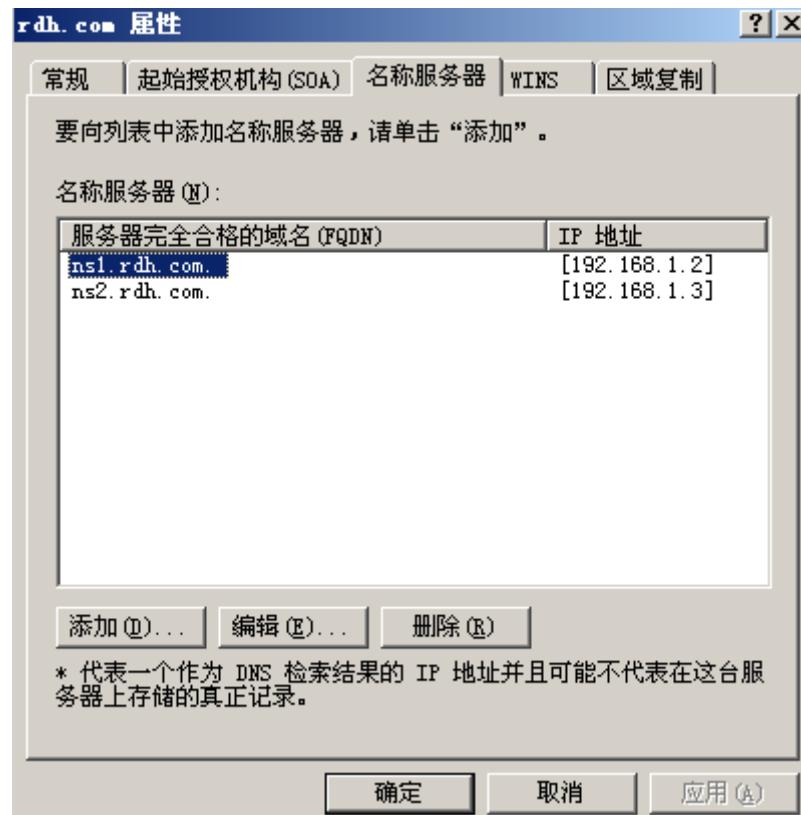
三、创建 SOA 记录、NS 记录、A 记录、MX 记录、CNAME 记录

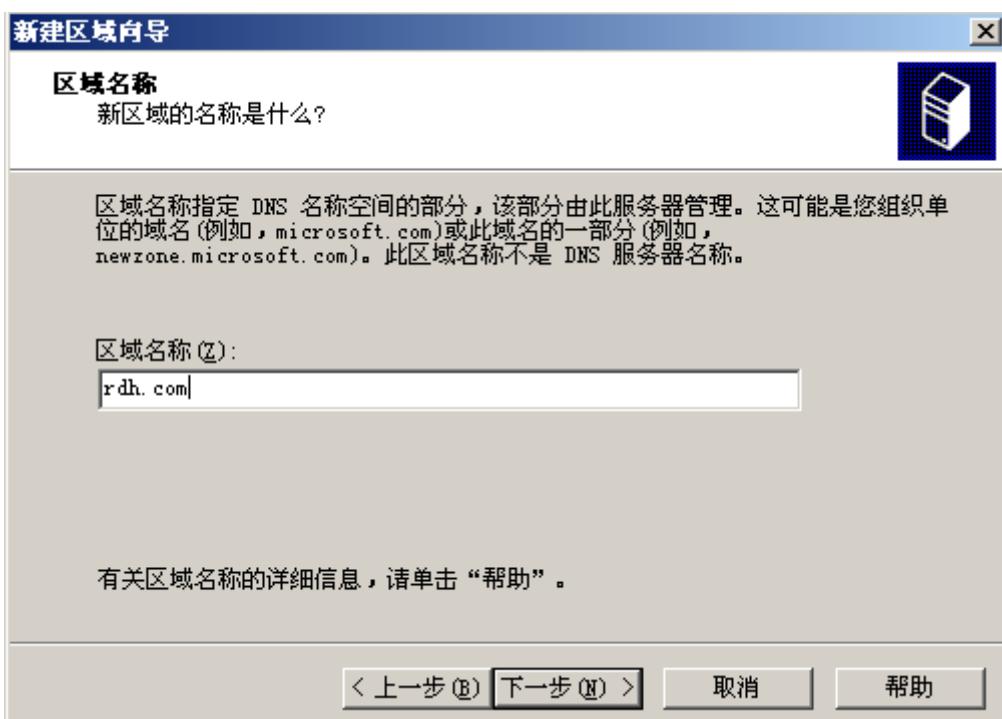
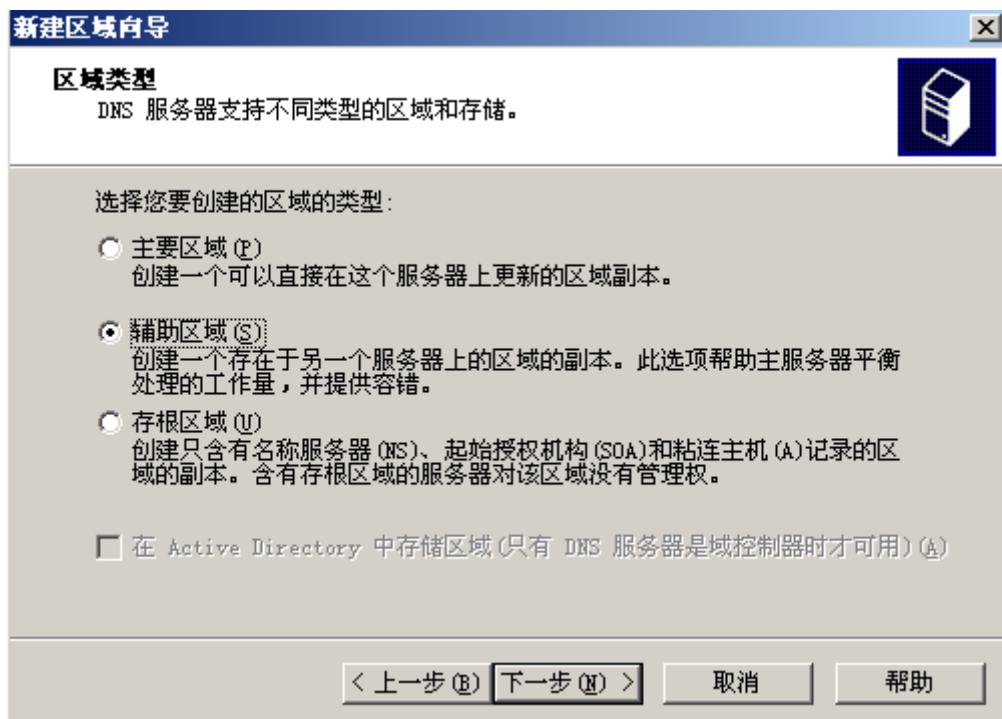


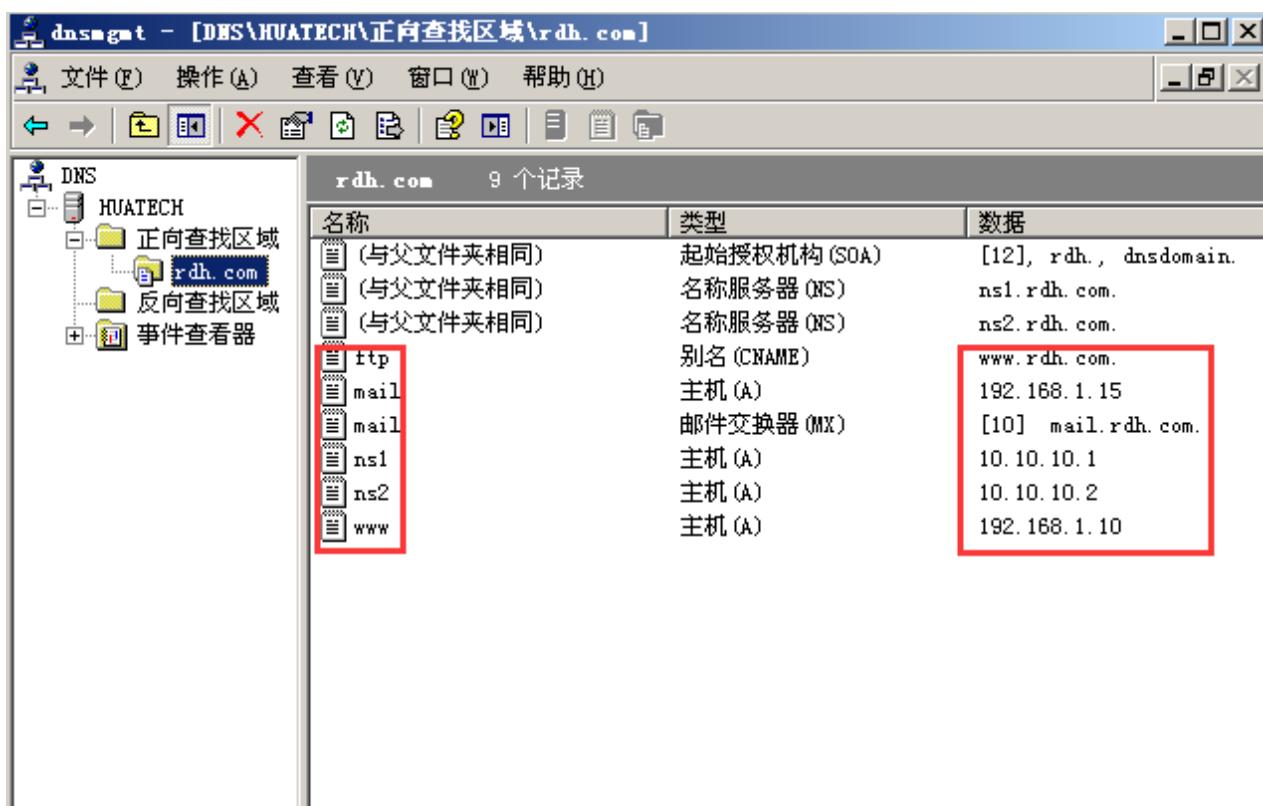
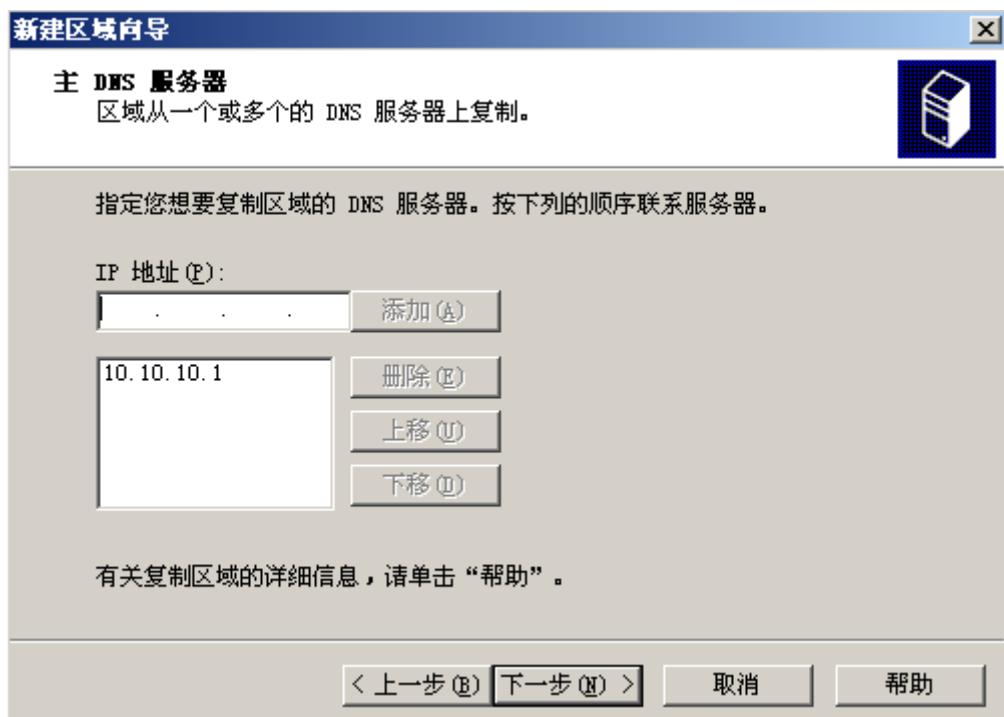




四、创建辅助区域







四、DNS 备份

- 1、停止 DNS 服务（在运行中执行命令：net stop dns）。
- 2、打开 regedit, 找到 HKEY_LOCAL_MACHINE\system\currentcontrolset\services\dns
- 3、将 DNS 这个分支导出。命名为 dns-1.reg。
- 4、找到 HKEY_LOCAL_MACHINE\software\microsoft\windowsNT\currentversion\DNSserver。
- 5、将 DNSserver 分支导出，命名为 dns-2.reg
- 6、打开%systemroot%\system32\dns\, 把其中的所有*.dns 文件复制出来，并和 dns-1.reg 及 dns-2.reg 保存在一起。

五、DNS 恢复

- 1、当区域里的 DNS 服务器发生故障，重新建立一台 win2003 服务器，并与所要替代的 DNS 服务器起相同的名字，设置相同的 DNS 后缀和 IP 地址。
- 2、在新系统中安装并启动 DNS 服务。
- 3、把前面备份出来的 *.dns 文件复制到新系统的%systemroot%\system32\dns\文件夹中。
- 4、停用 DNS 服务。
- 5、把备份的 dns-1.reg 和 dns-2.reg 导入到注册表中（如果机器名不同的时候，要替换一下注册表文件中的机器名。）。
- 6、重新启动 DNS 服务。

The screenshot shows a Windows Notepad window titled "rdh.com.dns - 记事本". The file contains a DNS zone configuration for the "rdh.com" zone. Several records are highlighted with red boxes:

- A red box surrounds the SOA record:

```
; Database file rdh.com.dns for rdh.com zone.  
; Zone version: 12  
  
@ IN SOA rdh. dnsdomain. (  
    12 ; serial number  
    900 ; refresh  
    600 ; retry  
    86400 ; expire  
    3600 ) ; default TTL
```
- A red box surrounds the NS records:

```
; Zone NS records  
  
@ NS ns1.rdh.com.  
@ NS ns2.rdh.com.
```
- A red box surrounds the A records for the zone records:

```
; Zone records  
  
ftp CNAME www.rdh.com.  
mail A 192.168.1.15  
mail MX 10 mail.rdh.com.  
ns1 A 192.168.1.1  
ns2 A 192.168.1.2  
www A 192.168.1.10
```

5.2 Linux DNS 服务器配置

```
[root@rdh ~]# yum -y install bind bind-utils
[root@rdh ~]# yum -y install php php-devel php-mysql php-httpd mysql mysql-server
mysql-devel
[root@rdh ~]# yum -y install php-*
[root@rdh ~]# service httpd start
[root@rdh ~]# service mysqld start
[root@rdh ~]# chkconfig httpd on
[root@rdh ~]# chkconfig mysqld on
[root@rdh ~]# mysqladmin -u root password jstvps
[root@rdh ~]# chkconfig named on
[root@rdh ~]# cd /etc/yum.repos.d/
```

```
[root@rdh yum.repos.d]# wget
http://repos.amberdms.com/config/centos/6/amberdms-c6-public.repo
[root@rdh ~]# yum -y install namedmanager-www namedmanager-bind
[root@rdh ~]# chkconfig --level 35 namedmanager_logpush on
[root@rdh ~]# chown named.root /etc/named.namedmanager.conf
[root@rdh ~]# cd /usr/share/namedmanager/resources/
[root@rdh resources]# ./autoinstall.pl
[root@rdh ~]# crontab -e
* * * * * php -q
/usr/share/namedmanager/bind/namedmanager_bind_configwriter.php>>/var/log/namedmanager_bind_configwriter
[root@rdh ~]# vim /etc/named.conf
include "/etc/named.namedmanager.conf";
[root@rdh ~]# vim /etc/namedmanager/config-bind.php
$config["api_url"] = "http://192.168.1.117/namedmanager";
// Application Install Location
$config["api_server_name"] = "ns1.example.com";
// Name of the DNS server (important: part of the authentication process)
$config["api_auth_key"] = "mykey";
https://192.168.1.117/namedmanager/
```

UserName: setup

Password: setup123

NamedManager

logged on as setup | settings | logout

Overview Changelog Domains/Zones Name Servers Configuration User Management

View Domain Add Domain Import Domain

Domain Details Domain Records Delete Domain

DOMAIN NAME RECORDS

Below is a list of all the records for your domain name, if you change any of them and click save, the changes will be applied and the name servers will reload shortly.

Domain Details

Domain rdh.com selected for adjustment

Nameserver Configuration

The following is a list of all the nameservers that this domain is managed by. These are auto-populated with the domains configured in the DB, however you can add your own records if you wish to sub-delegate the domain (for example, setting internal.example.com to be handled by another name server)

Type	TTL	Name/Origin	Content	
NS	86400	Domain Name (usually rdh.com)	FQDN of the name server	delete
NS	86400	Domain Name (usually rdh.com)	FQDN of the name server	delete

Mailserver Configuration

Configure all the mailservers for the system here, remember that all mail will be delivered to the server with the lowest priority by default.

Type	TTL	Priority	Name/Origin	Content	
MX	120		Origin (usually rdh.com)	FQDN or hostname of mail server	delete
MX	120		Origin (usually rdh.com)	FQDN or hostname of mail server	delete

=====配置文件=====

```
[root@rdh ~]# vim /etc/named.conf
listen-on port 53 { 127.0.0.1; }; 监听端口，也可写为 { 127.0.0.1; 192.168.139.46; }
listen-on-v6 port 53 { ::1; }; 对 IPV6 的支持
directory      "/var/named"; DNS 数据库存储目录
dump-file     "/var/named/data/cache_dump.db"; 缓存的备份目录
pid-file      "/var/run/named/named.pid"; 进程文件 PID
allow-query    { localhost; }; 允许查询的主机，一般 any
recursion yes; 是否允许递归查询
```

```
forwards { 202.102.224.67; }; 配置转发器，如果本地没有记录，交给转发器递归查询
allow-transfer { 辅助服务器地址; }; 辅助服务器地址，主辅区域传送
acl "acl1" {
    192.168.1.0/24;
    192.168.2.0/24;
} ; 定义一个 ACL，访问控制列表
zone "rdh.com" IN {
    type master;
    file "rdh.com.zone";
    allow-transfer { 192.168.1.2; };
};
zone "1.168.192.in-addr.arpa" IN {
    type master;
    file "1.168.192.zone";
    allow-transfer { 192.168.1.2; };
};
[root@rdh named]# cp -R named.localhost rdh.com.zone
[root@rdh named]# vim rdh.com.zone
```

```
$TTL 1D
@      IN SOA ns1.rdh.com. root.ns1.rdh.com. (
                           0          ; serial
                           1D         ; refresh
                           1H         ; retry
                           1W         ; expire
                           3H )       ; minimum
@      IN NS  rdh.com.
ns1   IN A   192.168.1.117
www   IN A   192.168.1.119
mail  IN A   192.168.1.220
        IN MX 10 mail.rdh.com.
```

ftp IN CNAME www.rdh.com.

SOA 起始授权机构：表示该正向域的授权合法的 DNS 服务器 ID 和名称。

Root 表示管理员

Serial 序列号：DNS 服务器状态的 ID，用来和辅助服务器同步信息。

```
$TTL 1D
@ IN SOA ns1.rdh.com. root.ns1.rdh.com. (
    0 ; serial
    1D ; refresh
    1H ; retry
    1W ; expire
    3H ) ; minimum
117 IN PTR ns1.rdh.com.
119 IN PTR www.rdh.com.
220 IN PTR mail.rdh.com.
    IN NS ns1.rdh.com.
```

创建主区域，创建辅助区域

```
zone "rdh.com" IN {
    type master;
    file "rdh.com.zone";
    allow-transfer { 192.168.1.2; };
};

zone "1.168.192.in-addr.arpa" IN {
    type master;
    file "1.168.192.zone";
    allow-transfer { 192.168.1.3; };
};

zone "rdh.com" IN {
    type slave;
    file "file/rdh.com.zone";
    masters { 192.168.1.1; };
};

zone "1.168.192.in-addr.arpa" IN {
    type slave;
    file "file/1.168.192.zone";
    masters { 192.168.1.1; };
};
```

开启服务

```
[root@rdh ~]# systemctl start named
[root@rdh ~]# systemctl enable named
DNS1=
[root@rdh ~]# dig +trace www.baidu.com
```

5.3 PowerDNS 安装配置

一、安装 YUM 库

```
[root@node-rac1 ~]# wget http://soft.laozuo.org/powerdns/epel-release-6-8.noarch.rpm
```

```
[root@node-rac1 ~]# rpm -ivh epel-release-6-8.noarch.rpm
```

二、安装支持数据库

```
[root@node-rac1 ~]# yum -y install mysql mysql-server
[root@node-rac1 ~]# mysql_secure_installation
[root@node-rac1 ~]# chkconfig mysqld on
[root@node-rac1 ~]# mysql -u root -p
mysql> CREATE DATABASE powerdns;
mysql> GRANT ALL ON powerdns.* TO 'powerdns'@'localhost' IDENTIFIED BY 'powerdns';
mysql> flush privileges;
mysql> USE powerdns;
CREATE TABLE domains (
    id INT auto_increment,
    name VARCHAR(255) NOT NULL,
    master VARCHAR(128) DEFAULT NULL,
    last_check INT DEFAULT NULL,
    type VARCHAR(6) NOT NULL,
    notified_serial INT DEFAULT NULL,
    account VARCHAR(40) DEFAULT NULL,
    primary key (id)
);
mysql> CREATE UNIQUE INDEX name_index ON domains(name);
CREATE TABLE records (
    id INT auto_increment,
    domain_id INT DEFAULT NULL,
    name VARCHAR(255) DEFAULT NULL,
    type VARCHAR(6) DEFAULT NULL,
    content VARCHAR(255) DEFAULT NULL,
    ttl INT DEFAULT NULL,
    prio INT DEFAULT NULL,
    change_date INT DEFAULT NULL,
    primary key(id)
);

mysql> CREATE INDEX rec_name_index ON records(name);
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE INDEX nametype_index ON records(name, type);
Query OK, 0 rows affected (0.32 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE INDEX domain_id ON records(domain_id);
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

```
CREATE TABLE supermasters (
    ip VARCHAR(25) NOT NULL,
    nameserver VARCHAR(255) NOT NULL,
    account VARCHAR(40) DEFAULT NULL
);
```

```
mysql> quit
```

三、安装 Apache 服务

```
[root@node-rac1 ~]# yum -y install httpd* php-http
[root@node-rac1 ~]# service httpd start
[root@node-rac1 ~]# chkconfig httpd on
```

四、安装 PHP 支持环境

```
yum -y install httpd php php-devel php-gd php-imap php-ldap php-mysql php-odbc
php-pecl php-xml php-xmlrpc php-mbstring php-mcrypt php-mhash gettext
[root@node-rac1 ~]# yum install php-pecl-DB php-pecl-MDB2-Driver-mysql
```

五、安装 PDNS

```
[root@node-rac1 ~]# yum -y install pdns pdns-backend-mysql
[root@node-rac1 ~]# vim /etc/pdns/pdns.conf
gmysql-host=localhost
gmysql-user=powerdns
gmysql-password=powerdns
gmysql-dbname=powerdns
[root@node-rac1 ~]# /etc/init.d/pdns start
[root@node-rac1 ~]# chkconfig pdns on
[root@node-rac1 ~]# cd /var/www/html/
wget http://downloads.sourceforge.net/project/poweradmin/poweradmin-2.1.7.tgz
mysql> GRANT SELECT, INSERT, UPDATE, DELETE
      -> ON powerdns.* 
      -> TO 'wmm'@'localhost'
      -> IDENTIFIED BY 'jstvps';
```

```
[root@node-rac1 inc]# vim config.inc.php 复制内容并修改权限为 777 删除 install
```

```
http://192.168.1.106/poweradmin-2.1.7
```

1. 安装 mysql + php 环境。

2. 安装 phpMyAdmin

```
yum -y install phpMyAdmin
```

安装完成后如下配置：

配置文件在/usr/share/phpMyAdmin 下，进入 libraries 目录

a. 修改/etc/phpMyAdmin/config.inc.php, , 修改前应先备份一下

```
['host']='localhost';      //除非数据库不在本机，此处不要更改
['port']='';默认为 3306
['auth_type']='config'
['user']='YOUR_USER_NAME'
['password']='YOUR_PASSWORD'      // 注意这里的 user 和 password 仅用于
auth_type=config 的情况下，密码不要为空
['blowfish_secret']='php' cookie 认证字符，可为任意，注意要不为空才行的
```

b. 修改/etc/httpd/conf.d/phpmyadmin.conf

本文件是 phpmyadmin 的访问控制文件，保证远程访问。如下修改即可：

```
<Directory /usr/share/phpMyAdmin/>
    Order Deny,Allow
    Deny from All
    Allow from 127.0.0.1
    Allow from ipaddress
</Directory>
```

测试：

<http://IP/phpMyAdmin/>

```
rm -rf /etc/udev/rules.d/70-persistent-net.rules
```

5.4 路由器配置 DNS 服务器

<http://www.laozuo.org/3924.html>

```
R1(config)#ip name-server 4.2.2.5
R1(config)#ip host alan 192.168.1.10
R1(config)#ip domain-name cisco.com
R1(config)#ip domain-lookup
```

六、Storage Server

DAS：直接存储模式，就是存储直接挂在到 PC 或 Server 的总线上。

NAS：TCP/IP 网络存储方式、类似 FTP、NFS、SAMBA。

SAN：FC-SAN、IP-SAN

6.1 NFS 网络文件系统

Linux 平台

```
[root@rdh ~]# vim /etc/exports
```

参数说明：

rw：读写权限

ro：只读权限

no_root_squash：登入 NFS 主机，使用该共享目录时相当于该目录的拥有者，如果是 root 的话，那么对于这个共享的目录来说，他就具有 root 的权限，这个参数『极不安全』，不建议使用。

root_squash 登入 NFS 主机，使用该共享目录时相当于该目录的拥有者。但是如果是以 root 身份使用这个共享目录的时候，那么这个使用者（root）的权限将被压缩成为匿名使用者，即通常他的 UID 与 GID 都会变成 nobody 那个身份。

all_squash：不论登入 NFS 的使用者身份为何，他的身份都会被压缩成为匿名使用者，通常也就是 nobody。

Anonuid：自行设定 UID 值，这个 UID 值必须存在/etc/passwd 当中。

Anongid：自行设定 GID 值，这个 GID 值必须存在/etc/passwd 当中。

Sync：将资料同步到硬盘和内存当中。

Async：将资料先同步到内存，后同步到硬盘。

部分实例：

```
/share *(rw)
/share *(rw, no_root_squash)
/share 192.168.1.0/24(rw, no_root_squash) *(ro)
/share 192.168.1.*(rw, no_root_squash) *(ro)
/share *.rdh.com(rw, all_squash, anonuid=40, anongid=40)
[root@rdh ~]# systemctl start rpcbind nfs-server
[root@rdh ~]# systemctl enable rpcbind nfs-server
[root@rdh ~]# showmount -e localhost
[root@rdh ~]# showmount -e localhost
```

Export list for localhost:

```
/share *
[root@rdh ~]# mount -t nfs rdh.com:/share /home
[root@rdh ~]# vim /etc/fstab
```

```
rdh.com:/share /home nfs defaults 0 0
```

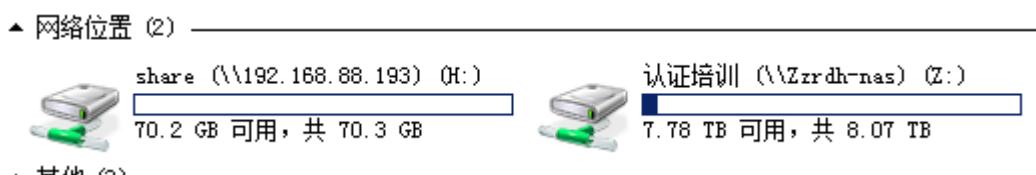
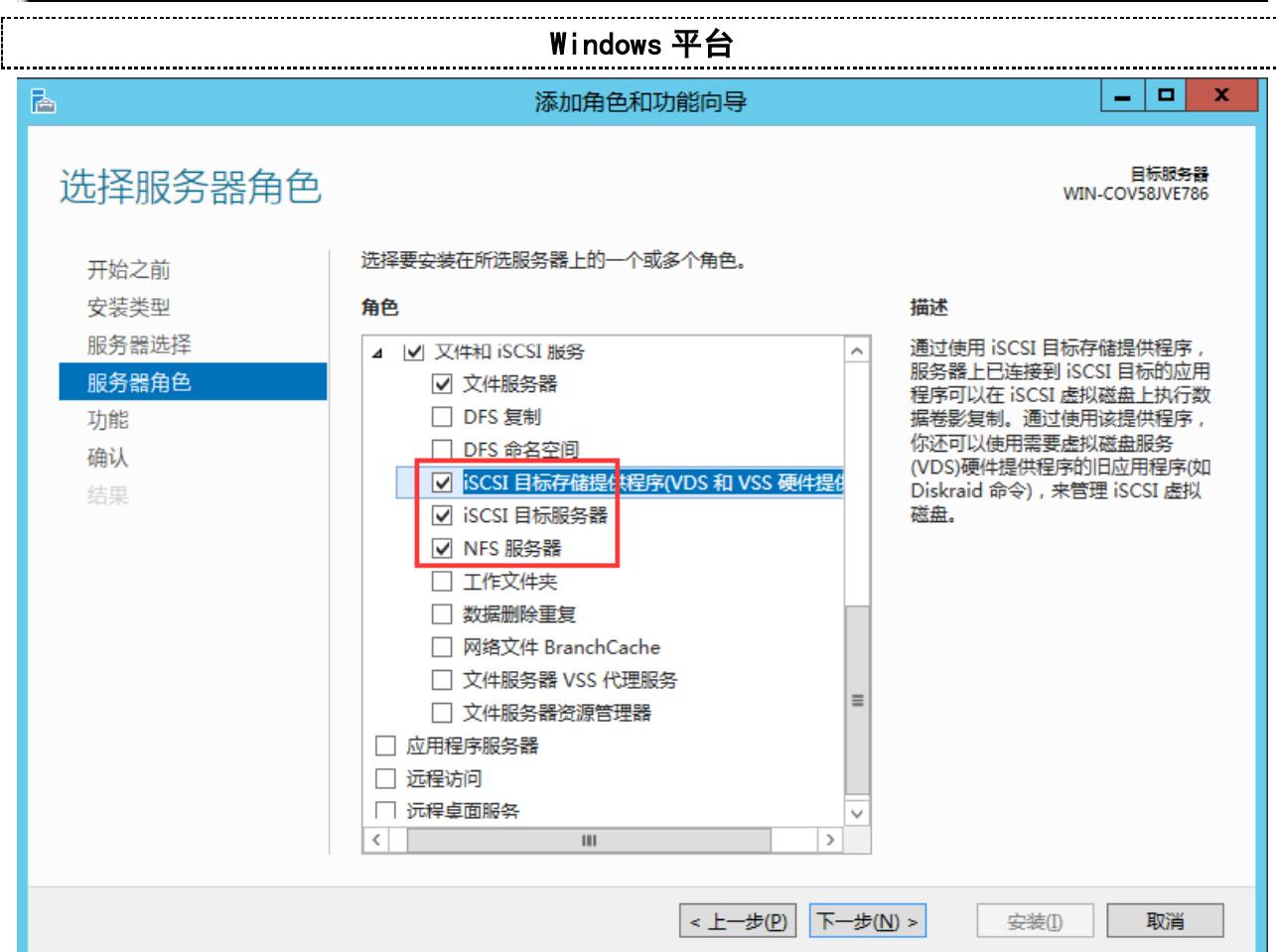
=====自动挂载=====

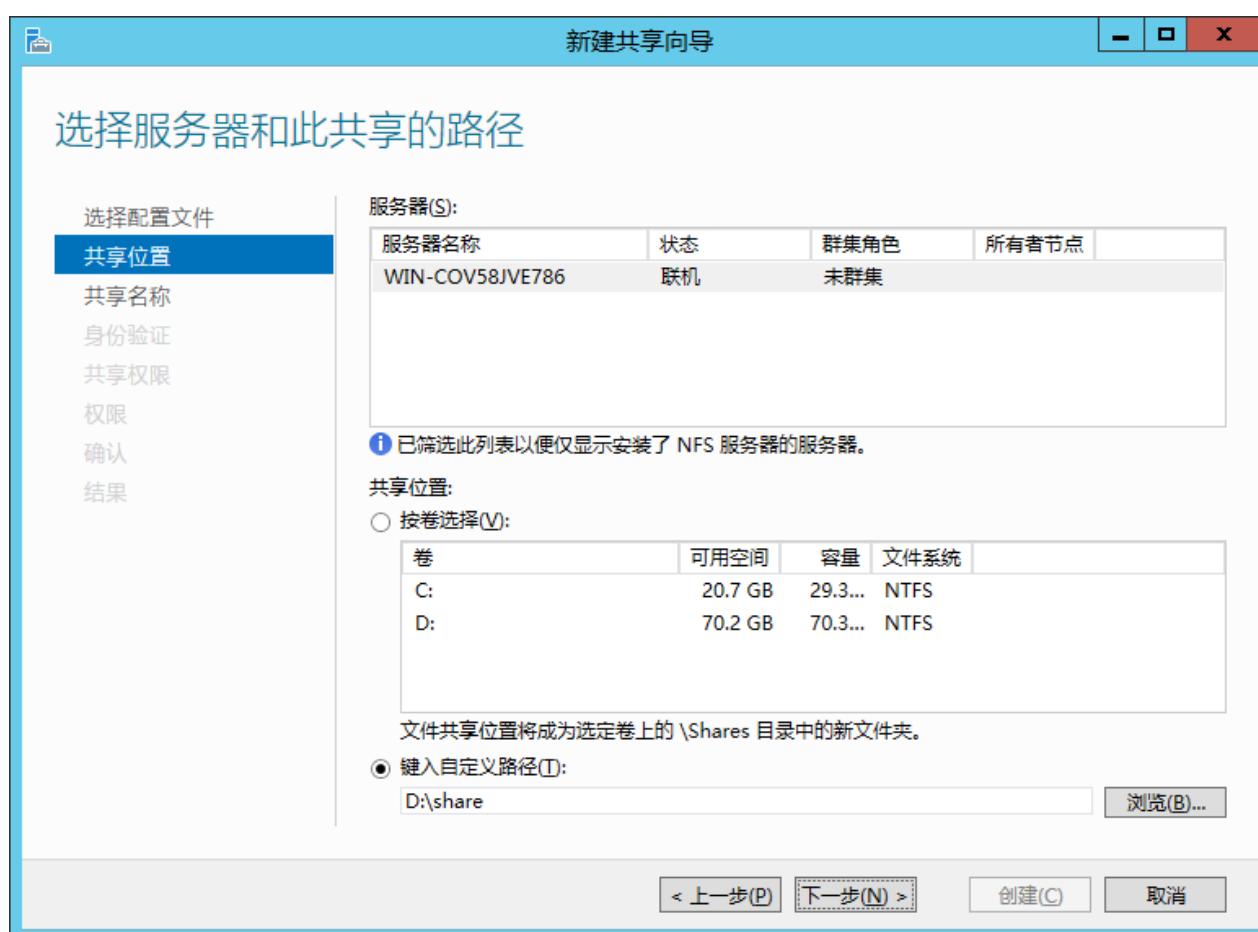
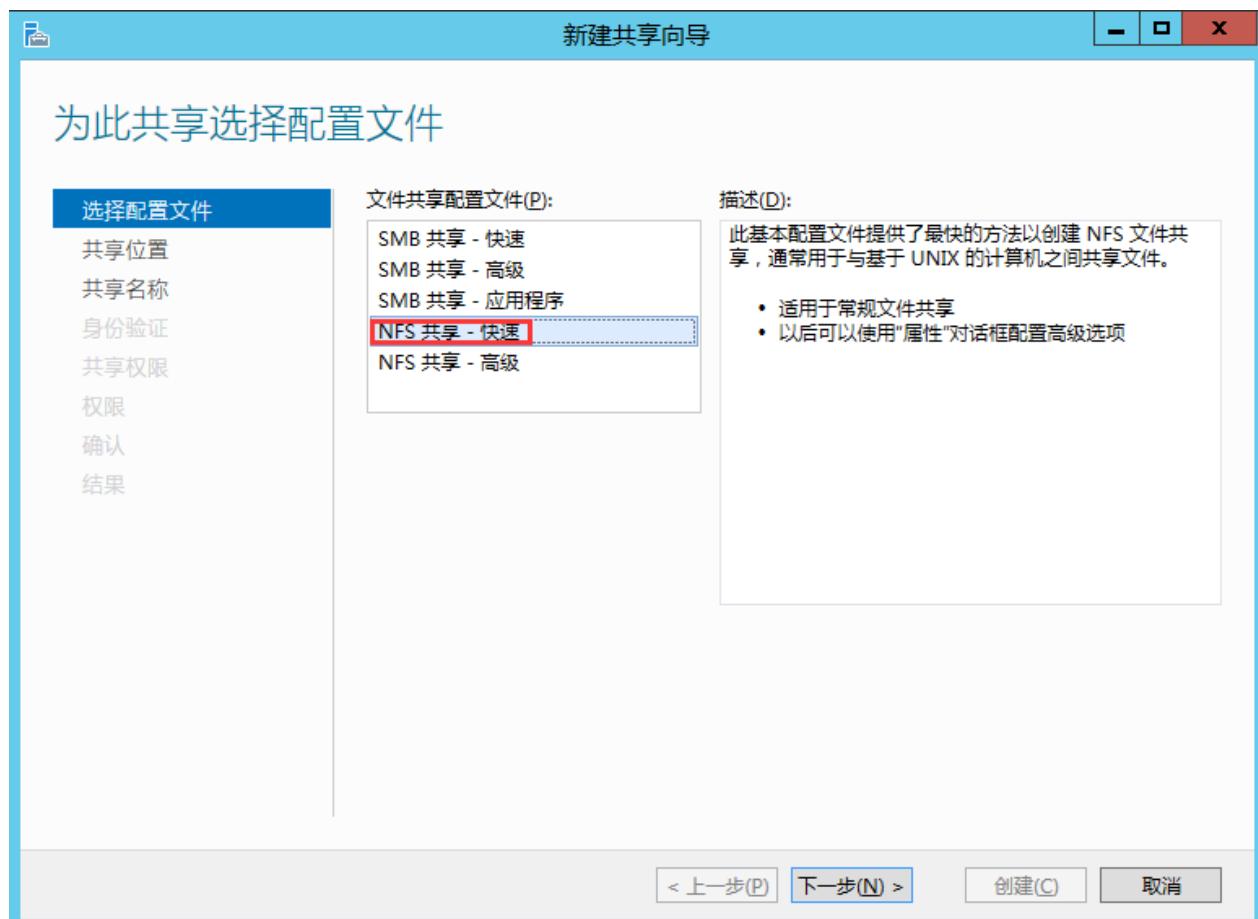
```
[root@rdh ~]# yum -y install autofs
[root@rdh ~]# vim /etc/auto.master
/- /etc/auto.mount
[root@rdh ~]# mkdir /mntdir
[root@rdh ~]# vim /etc/auto.mount
/mntdir -fstype=nfs, rw rdh.com:/home 【共享目录】
```

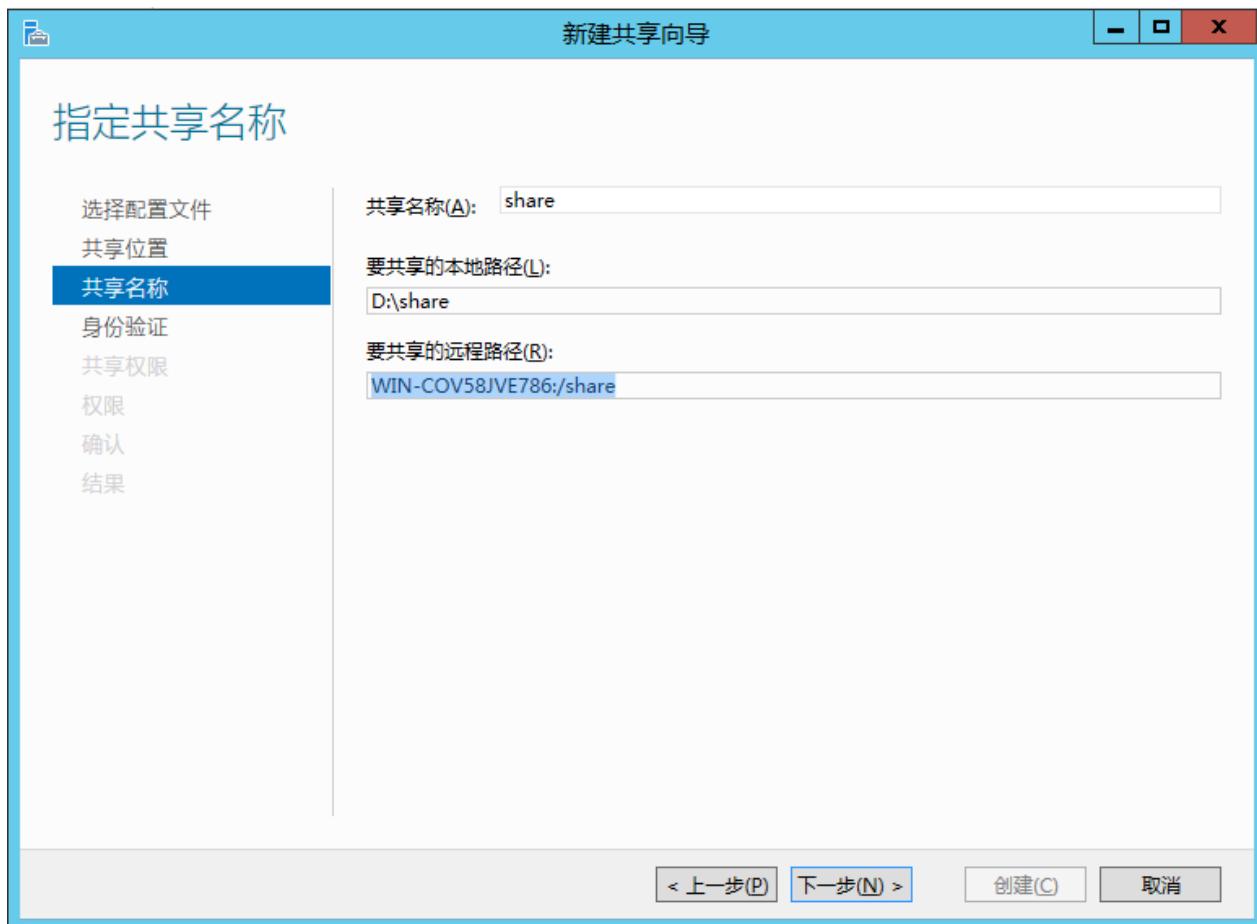
Windows 客户端挂载



```
C:\>Administrator>mount 192.168.88.216:/share H:\  
H: 现已成功连接到 192.168.88.216:/share
```





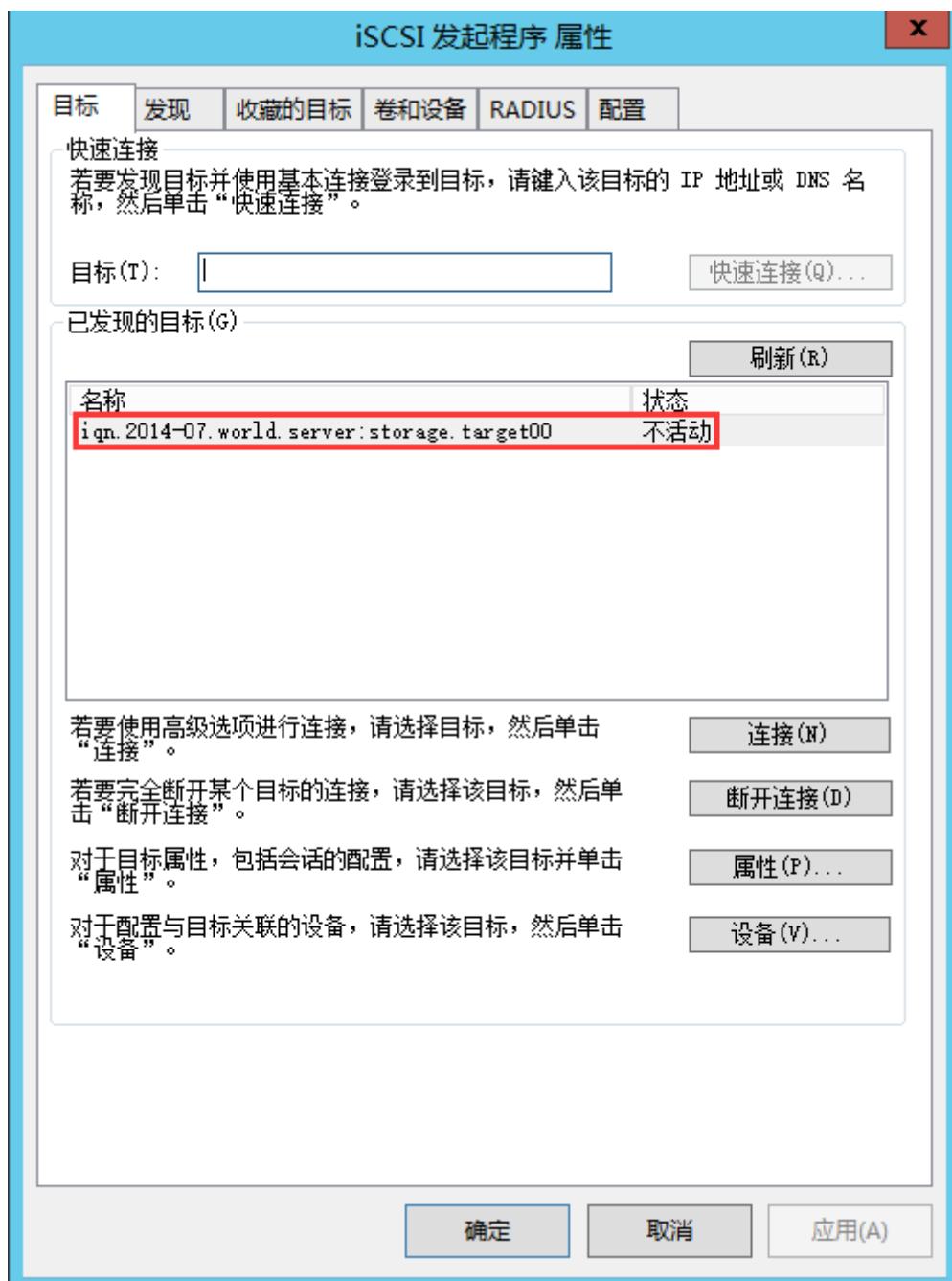


6.2 iSCSI 存储服务器

Linux 平台

```
[root@rdh ~]# yum -y install targetcli
[root@rdh ~]# mkdir /iscsi_disks
[root@rdh ~]# targetcli
/> cd backstores/fileio
/backstores/fileio> create disk01 /iscsi_disks/disk01.img 10G
/backstores/fileio> cd /iscsi
/iscsi> create iqn.2016-2-22.com.rdh:storage
/iscsi> create iqn.2014-07.world.server:storage.target00
/iscsi> cd iqn.2014-07.world.server:storage.target00/tpg1/portals/
/iscsi/inqn.20.../tpg1/portals> create 192.168.88.222
/iscsi/inqn.20.../tpg1/portals> cd ../luns
/iscsi/inqn.20.../t00/tpg1/luns> create /backstores/fileio/disk01
/iscsi/inqn.20.../t00/tpg1/luns> cd ../acls
/iscsi/inqn.20.../t00/tpg1/acls> create iqn.2014-07.world.server:www.server.world
/iscsi/inqn.20.../t00/tpg1/acls> cd iqn.2014-07.world.server:www.server.world
/iscsi/inqn.20.../server.world> set auth userid=wmm
Parameter userid is now 'wmm'.
/iscsi/inqn.20.../server.world> set auth password=jstvps
```

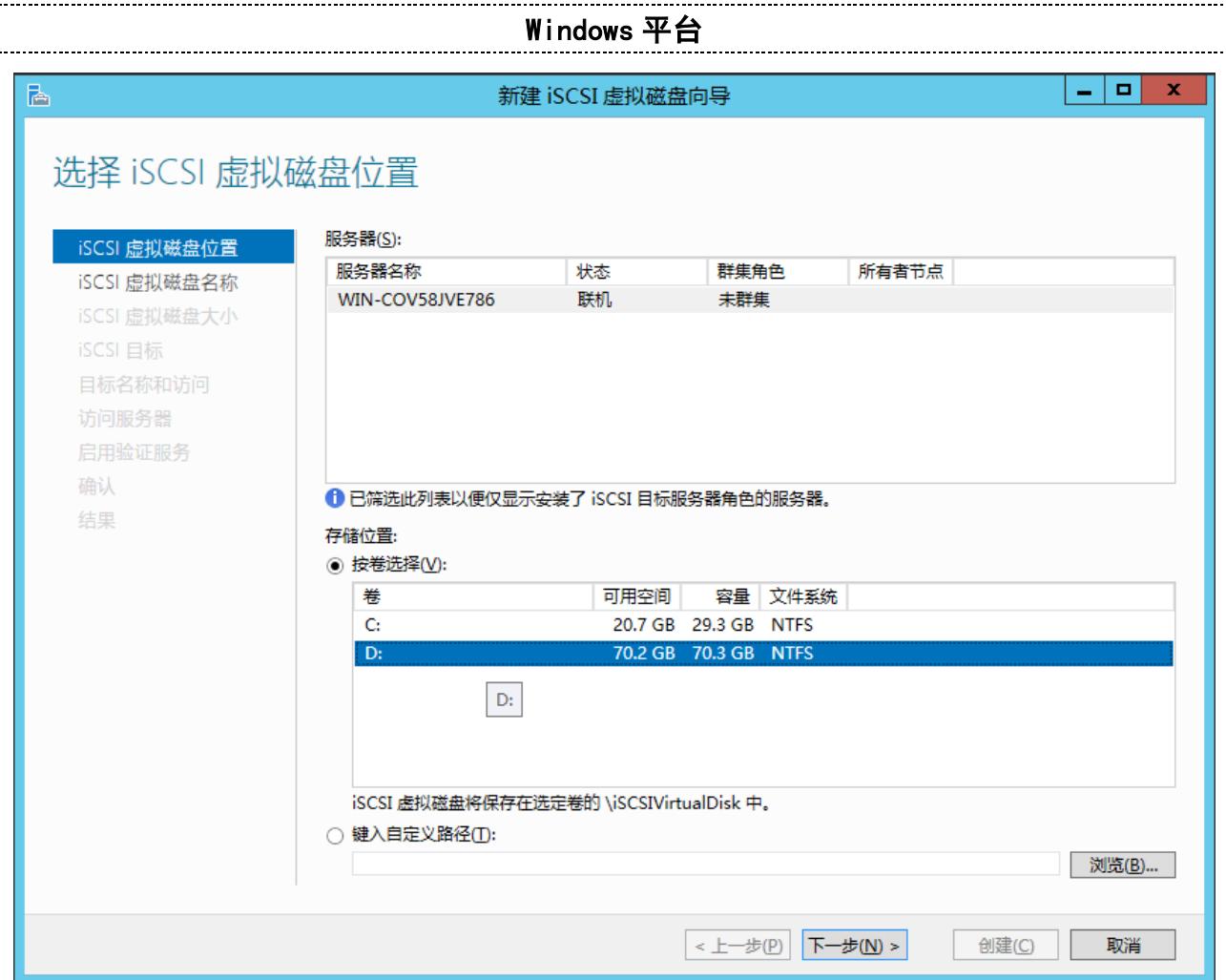
```
Parameter password is now 'jstvps'.
/iscsi/iqn.20....server.world> exit
[root@rdh ~]# systemctl enable target
[root@rdh ~]# netstat -tulnp|grep 3260
tcp          0      0 0.0.0.0:3260          0.0.0.0:*          LISTEN
-
```

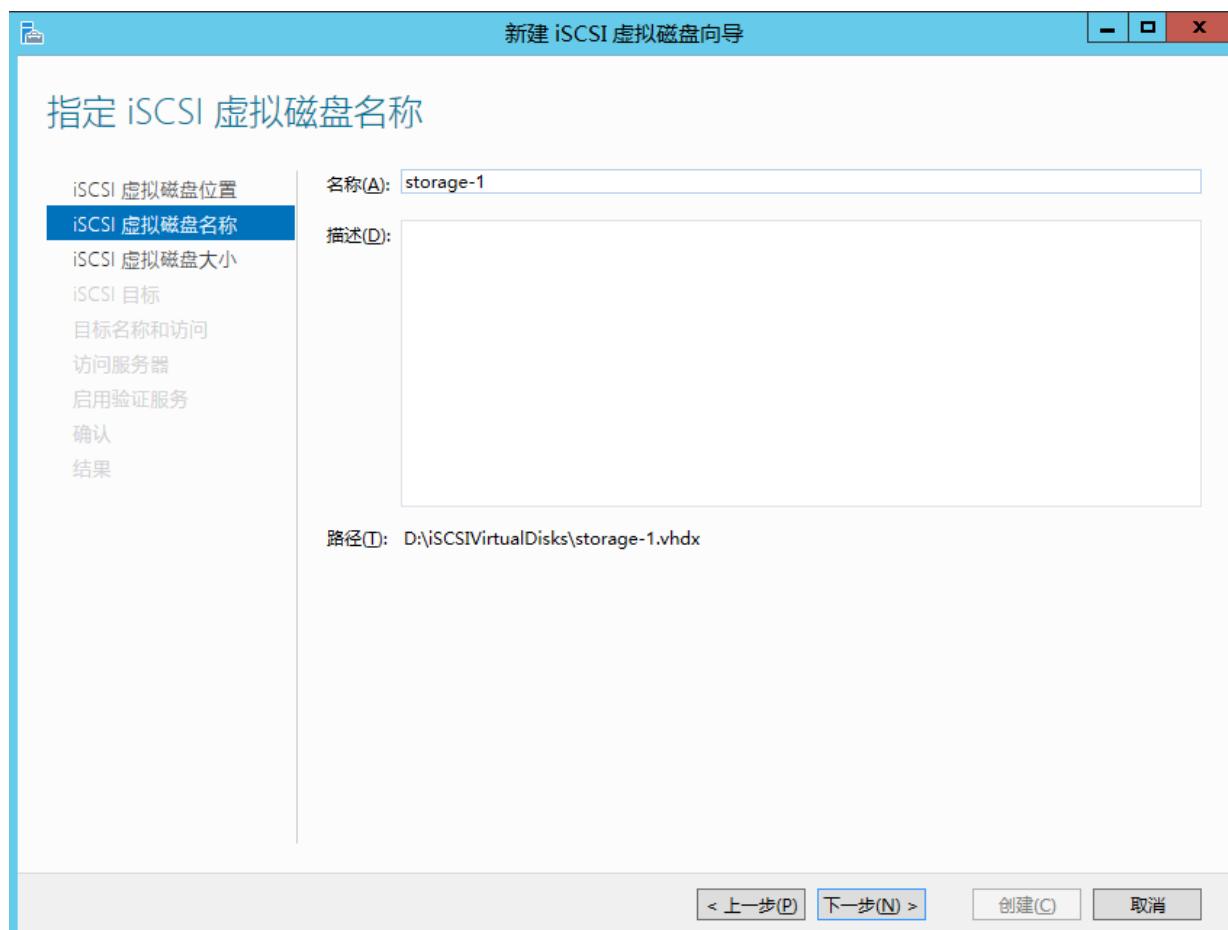


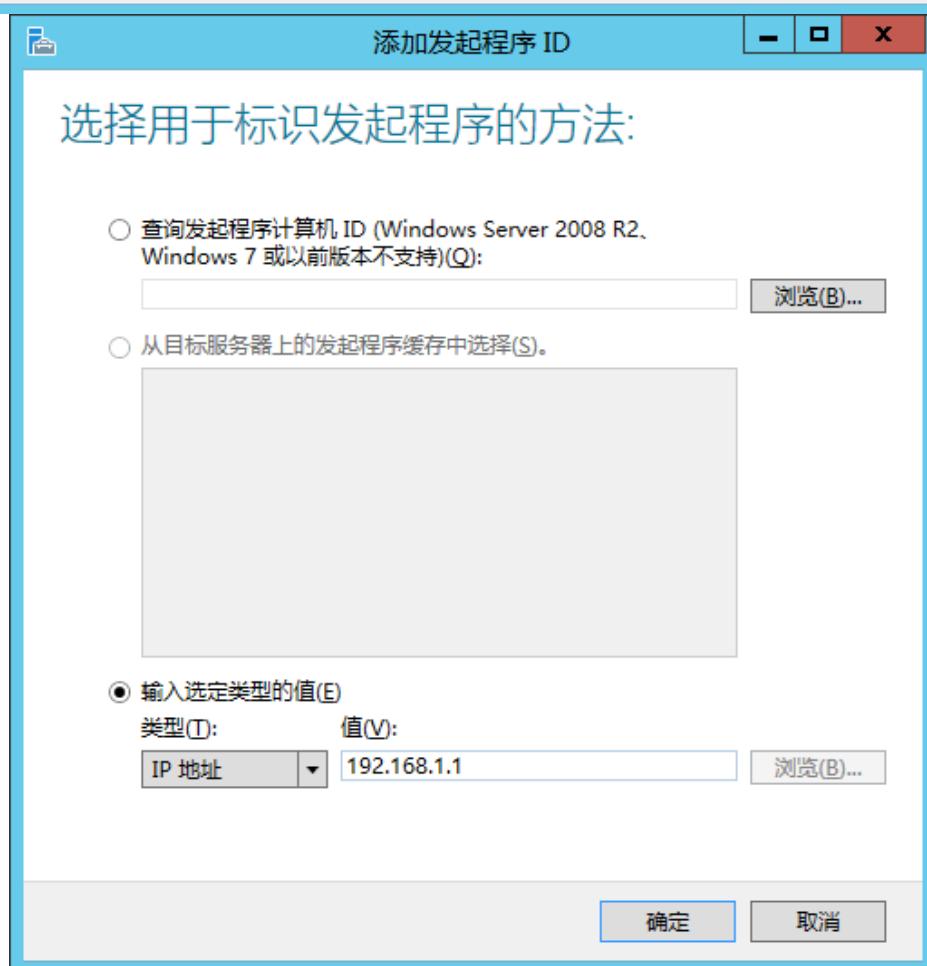
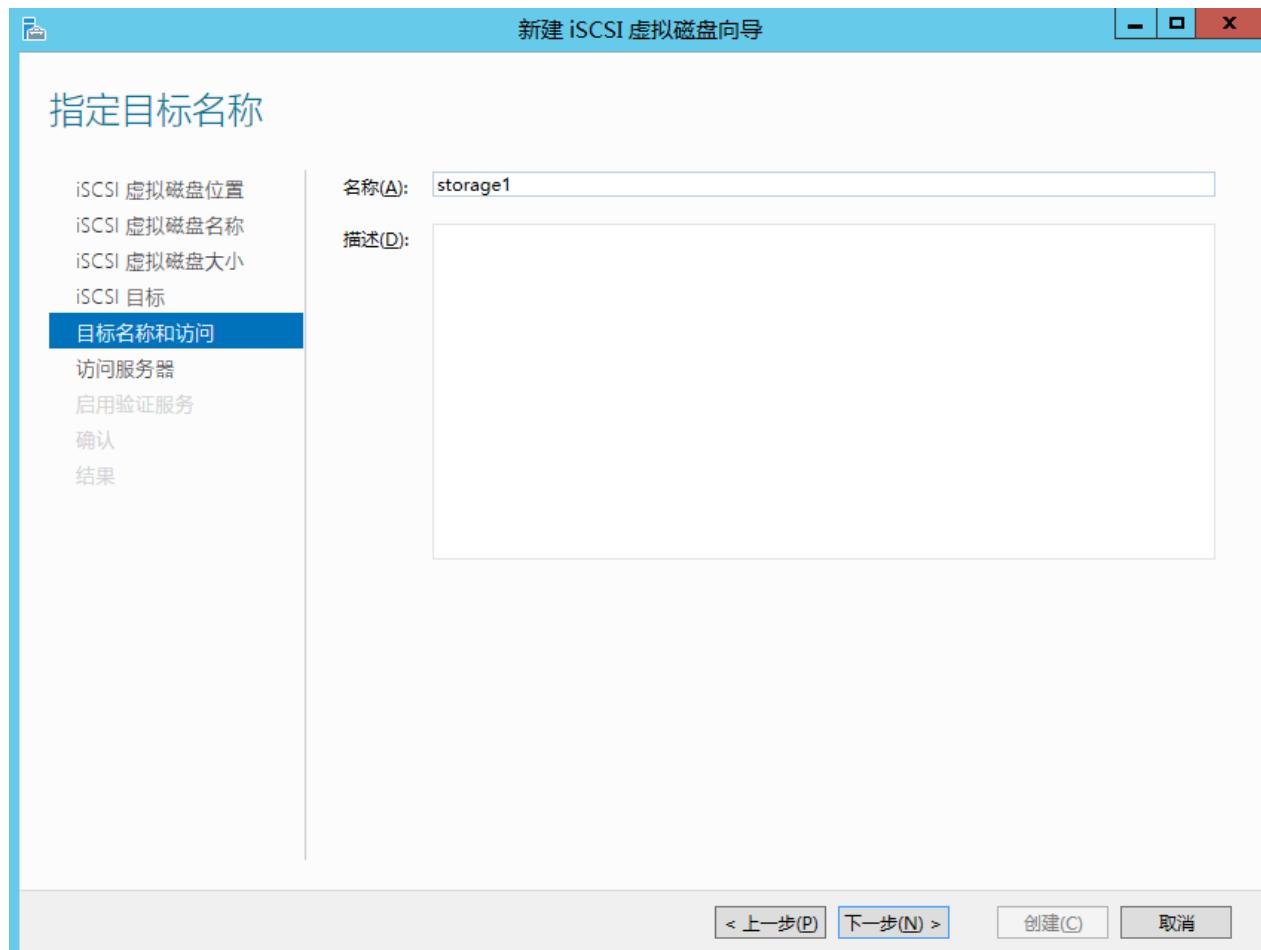


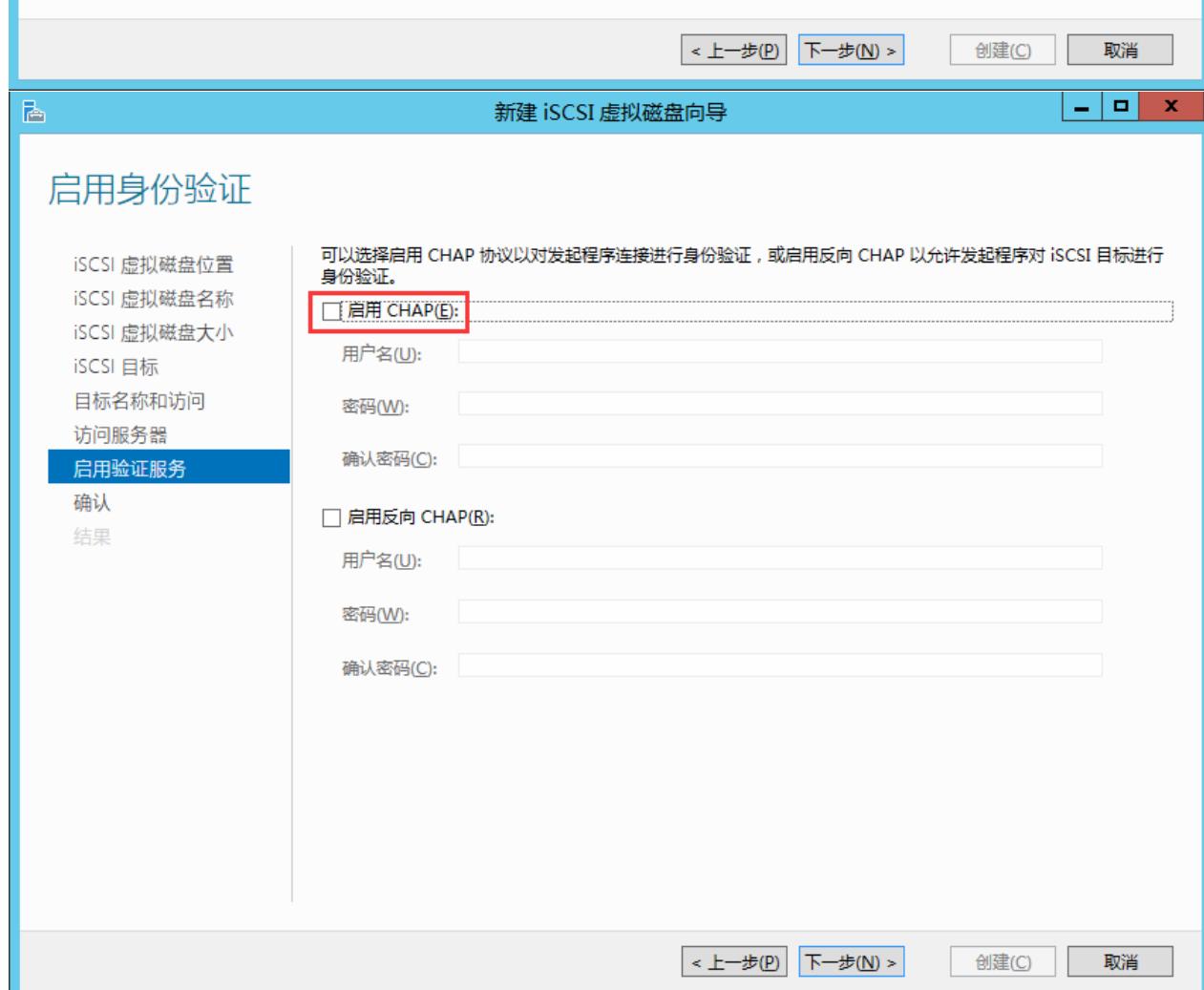
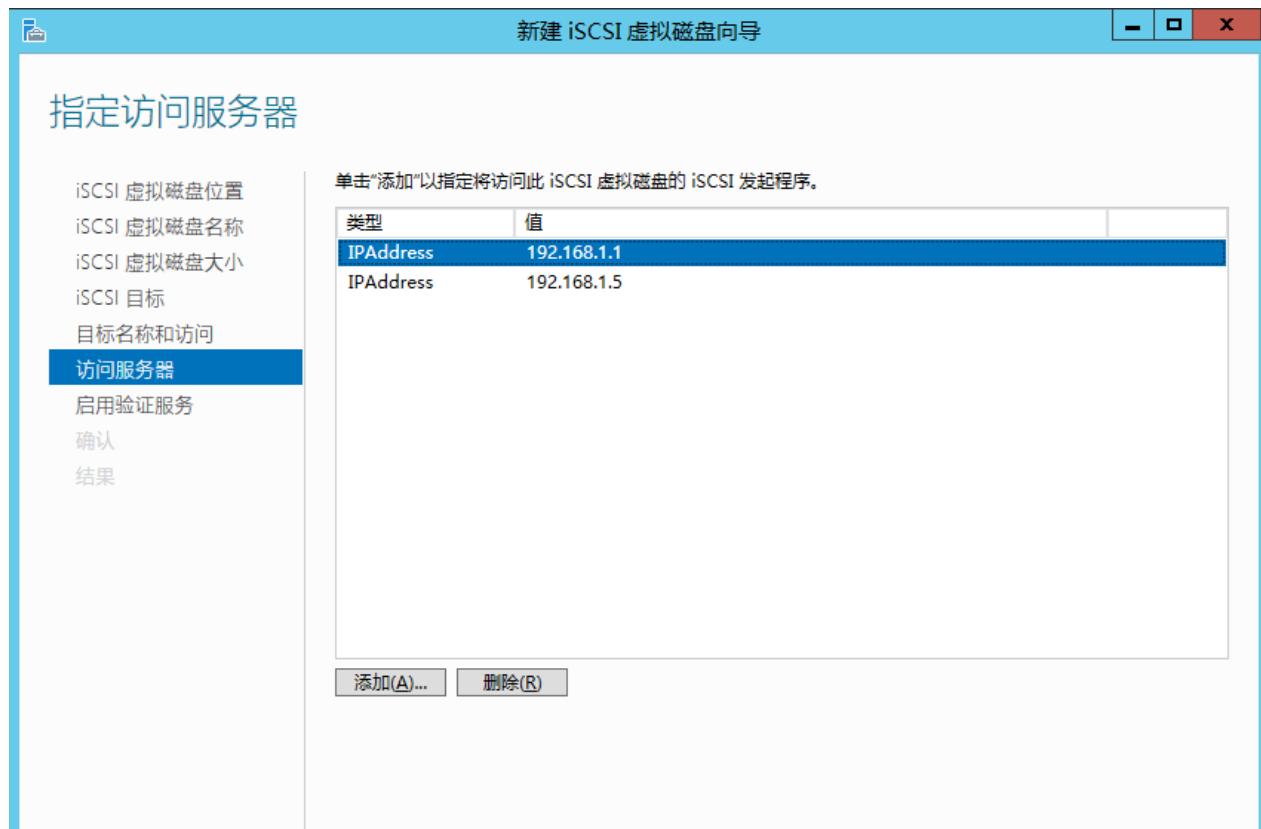
```
[root@rdh ~]# yum --enablerepo=epel -y install scsi-target-utils
[root@rdh ~]# mkdir /iscsi_disks
[root@rdh ~]# dd if=/dev/zero of=/iscsi_disks/disk0.img bs=1 seek=5G
[root@rdh iscsi_disks]# dd if=/dev/zero of=/iscsi_disks/disk0.img bs=1G count=1
[root@rdh ~]# vim /etc/tgt/targets.conf
```

```
<target iqn.2015-2.com.rdh:storage1>
backing-store /iscsi_disks/disk0.img
initiator-address 192.168.88.193
incoming-user wmm jstvps
</target>
[root@rdh ~]# systemctl start tgtd
[root@rdh ~]# systemctl enable tgtd
```









存储多路径

Linux 存储多路径配置：第一种方式

```
[root@node-rac1 Packages]# /etc/init.d/multipathd start
[root@node-rac1 Packages]# yum -y install device-mapper-multipath
[root@node-rac1 Packages]# modprobe dm-multipath
[root@node-rac1 Packages]# chkconfig multipathd on
[root@node-rac1 Packages]# mpathconf --enable --find_multipaths y
--with_multipathd y --with_chkconfig y
[root@node-rac1 Packages]# /etc/init.d/multipathd restart
```

Linux 存储多路径配置：第二种方式

```
# for i in `cat /proc/partitions | awk {'print $4'} |grep sd`; do echo "### $i:
`scsi_id --whitelist /dev/$i`"; done
1. # multipath.conf written by anaconda
2.
3. defaults {
4.   user_friendly_names yes
5. }
6. blacklist {
7.   devnode "^(ram|rawloop|fdmddm|srscdst) [0-9]*"
8.   devnode "^(hd[a-z])"
9.   devnode "^(dcssblk[0-9])*"
10. device {
11.   vendor "DGC"
12.   product "LUNZ"
13. }
14. device {
15.   vendor "IBM"
16.   product "S/390.*"
17. }
18. # don't count normal SATA devices as multipaths
19. device {
20.   vendor "ATA"
21. }
22. # don't count 3ware devices as multipaths
23. device {
24.   vendor "3ware"
25. }
26. device {
27.   vendor "AMCC"
28. }
29. # nor highpoint devices
30. device {
31.   vendor "HPT"
```

```
32. }
33. wwid "20080930-1"
34. wwid "20080930-1"
35. device {
36. vendor Cisco
37. product Virtual_CD_DVD
38. }
39. wwid "*" //其实可以注释这项，这样就不需要单独填写 blacklist_exceptions
40. }
41. blacklist_exceptions { //排除在黑名单之外的 wwid
42. wwid "360060160a2212f00a67e0b91f2dbe111"
43. wwid "360060160a2212f0044a0fc6ef5eae111"
44. }
45. multipaths {
46. multipath {
47. uid 0 //磁盘读所属用户 uid
48. gid 0 //磁盘所属组 gid
49. wwid "360060160a2212f00a67e0b91f2dbe111" //wwid 号
50. mode 0600 //磁盘读写权限
51. }
52. multipath {
53. wwid "360060160a2212f0044a0fc6ef5eae111"
54. alias data //别名
55. }
56. .... //还可以根据实际情况，配置其它磁盘的别名、uid、gid、mode etc...
57.
58. }
```

开源存储系统 FreeNAS

下 载 地 址：
<http://pan.baidu.com/s/1gd7oAqB#path=%252FFreeNAS%252FFreeNAS-9.3%252FSTABLE%252F201506292332>

使用指南：http://www.getnas.com/freenas/freenas_install.html

开源存储系统 Openfiler

下 载 地 址：
<https://sourceforge.net/projects/openfiler/files/openfiler-distribution-iso-2.99-x64/>

企业级存储系统 Open-E

下载地址：<http://www.open-e.com/download/serve/3vHZLASpSNpTKr3d9bro61/iso/>
使用指南：<http://www.open-e.com/download/open-e-data-storage-software-v7/iso/>

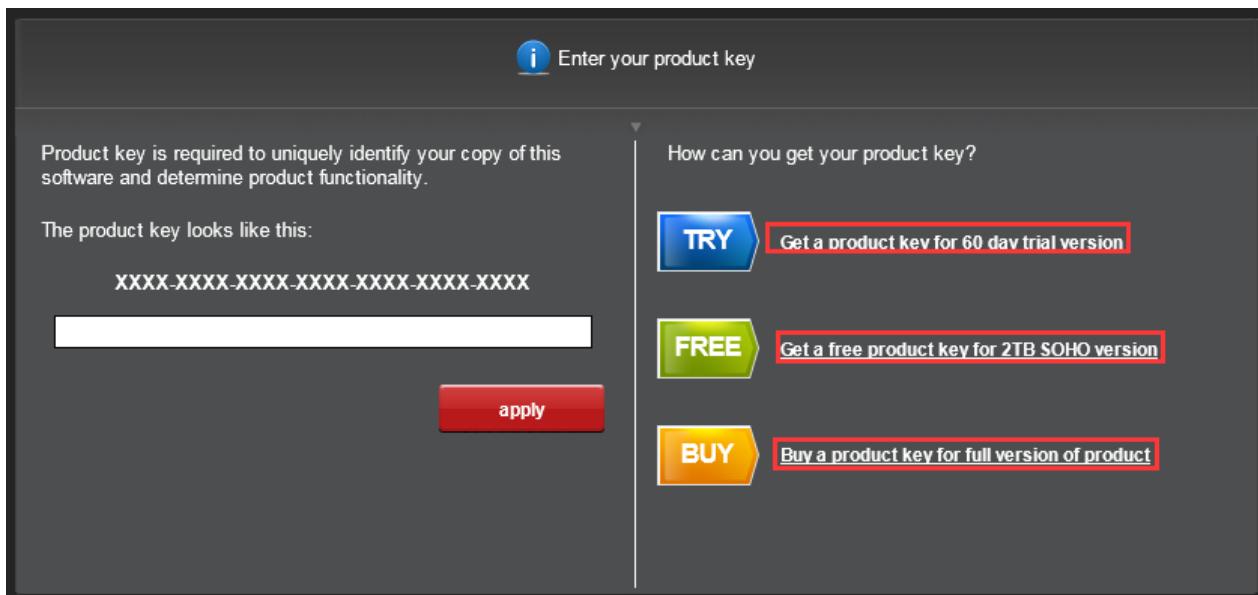
User Guides & Manuals



Open-E DSS V7
Quick Start Guide



Open-E DSS V7
User Manual



第一个 TRY 的意思是：只试用 60 天

第二个 FREE 的意思是：只能支持 2TB 的办公模式

第三个 BUY 的意思是：购买

教程地址：

http://wenku.baidu.com/link?url=3JY9Teh7CNWeIVo9F3DArz-XS-14qt16X5fAkMV-HqigDefniG58KGyNQN7cRZxQYaVq66kwARDxQzu_7MMi-uIJ2gDXQPVjmqWAD71vS8m

七、PXE SERVER

```
[root@rdh ~]#yum -y install syslinux xinetd tftp-server
[root@rdh ~]# vim /etc/xinetd.d/tftp
disable = no
[root@rdh ~]# systemctl start xinetd
[root@rdh ~]# systemctl enable xinetd
[root@rdh ~]# yum --enablerepo=epel -y install dhcp
[root@rdh ~]# vim /etc/dhcp/dhcpd.conf
[root@rdh ~]# mkdir /var/lib/tftpboot/pxelinux.cfg
[root@rdh ~]# cp /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/
filename      "pxelinux.0";
next-server    "192.168.88.222";
subnet 10.0.0.0 netmask 255.255.255.0 {
# specify the range of lease IP address
range dynamic-bootp 10.0.0.200 10.0.0.254;
# specify broadcast address
```

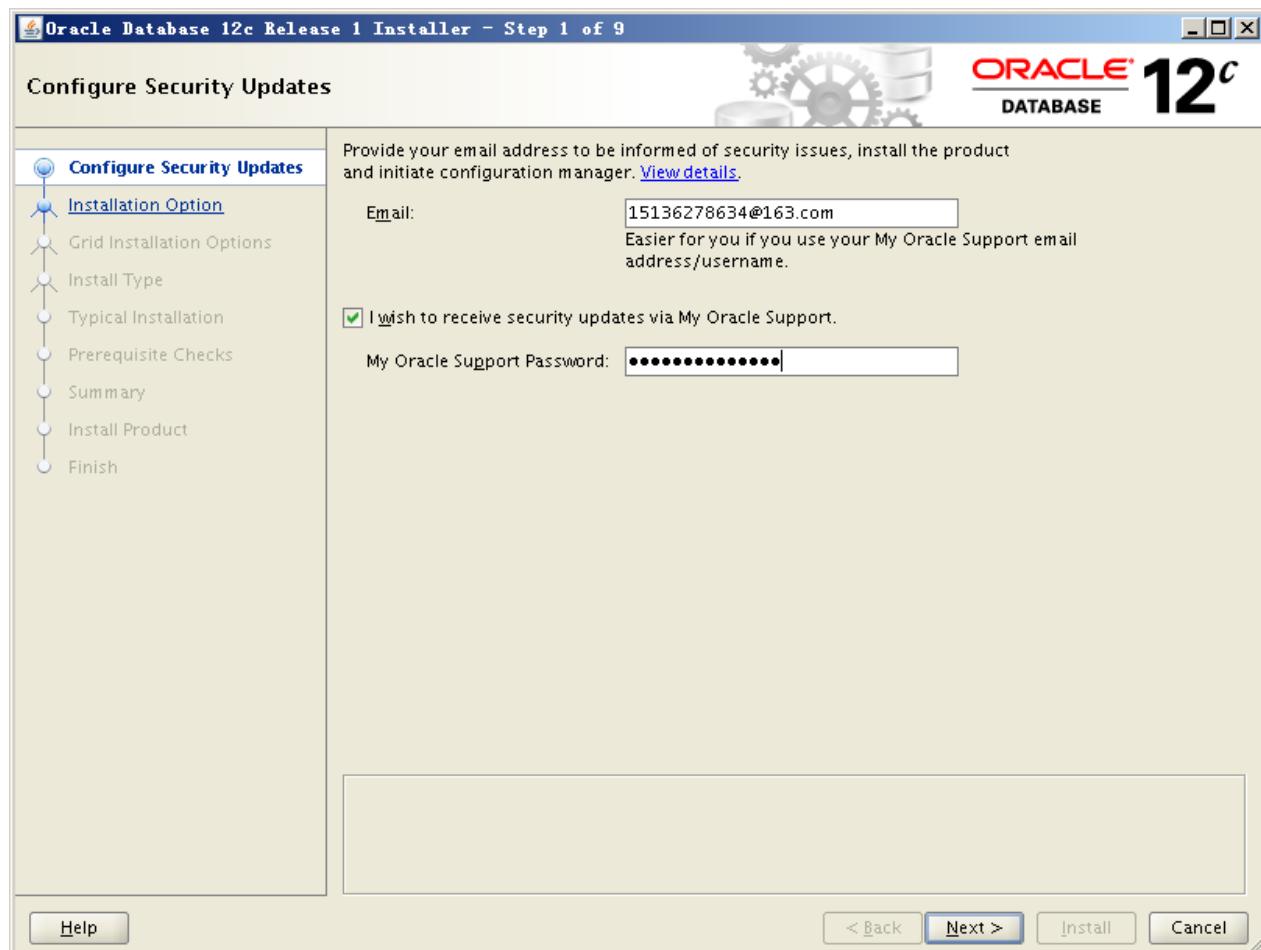
```
option broadcast-address 10.0.0.255;
# specify default gateway
option routers 10.0.0.1;
}
```

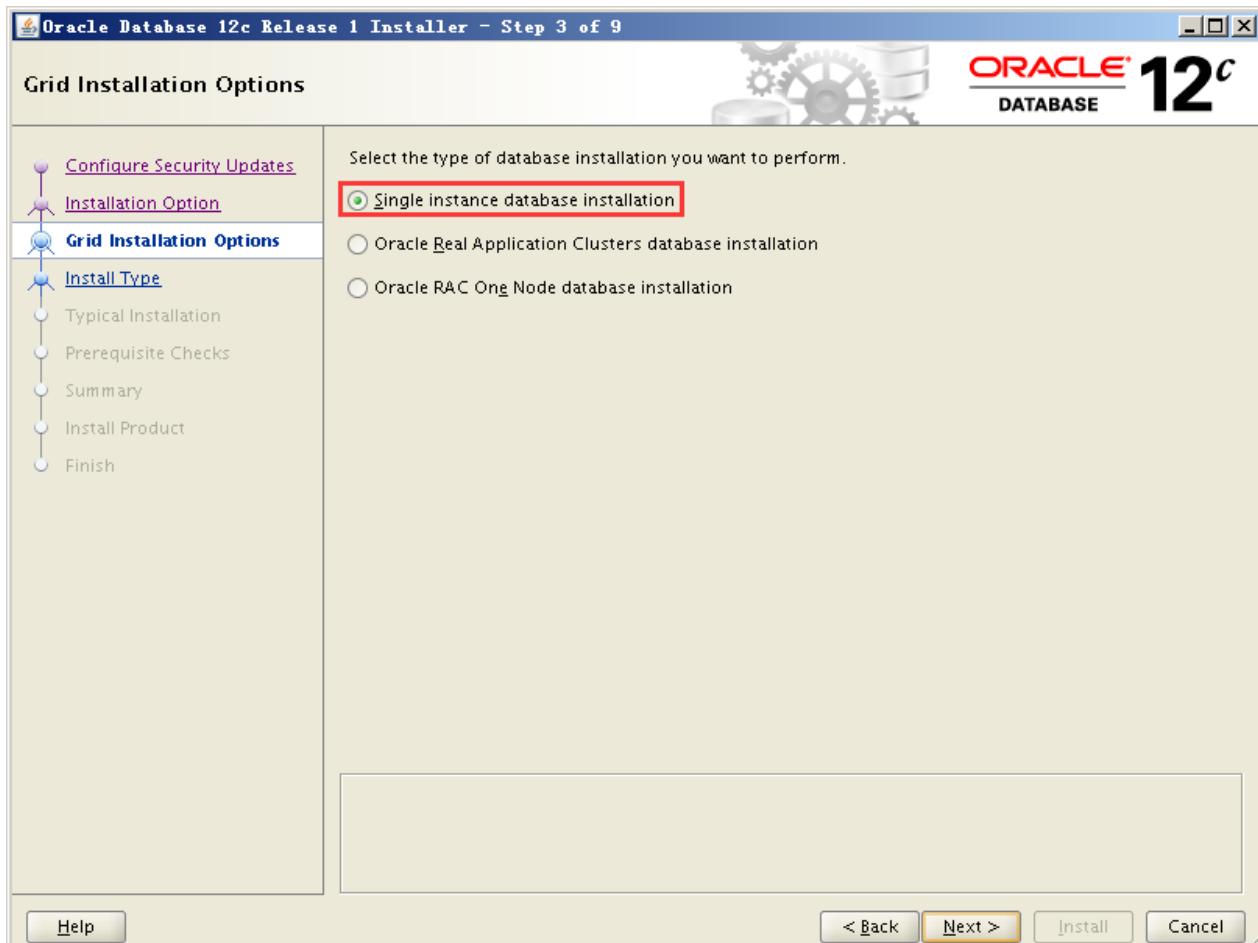
八、安装配置 Oracle 12C

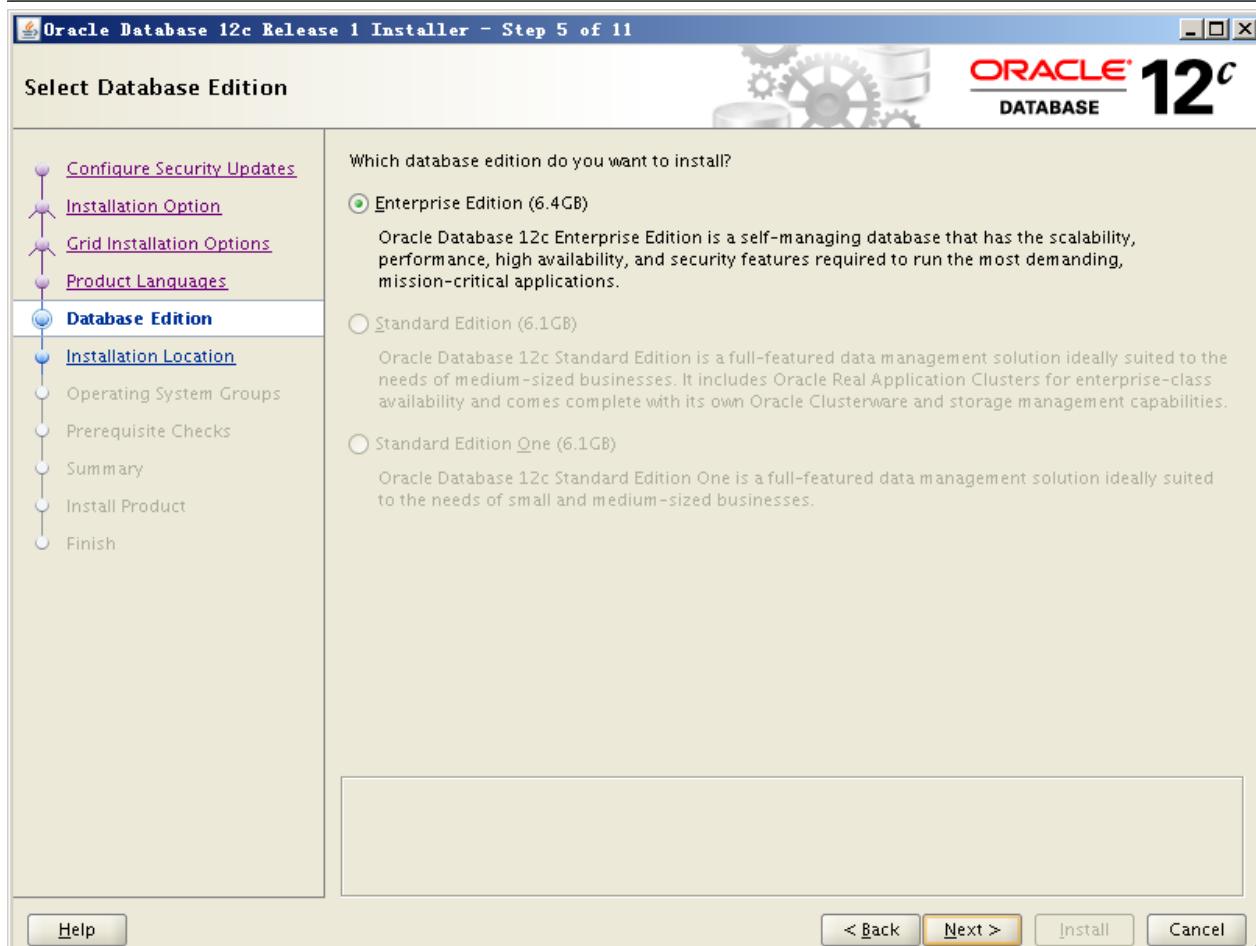
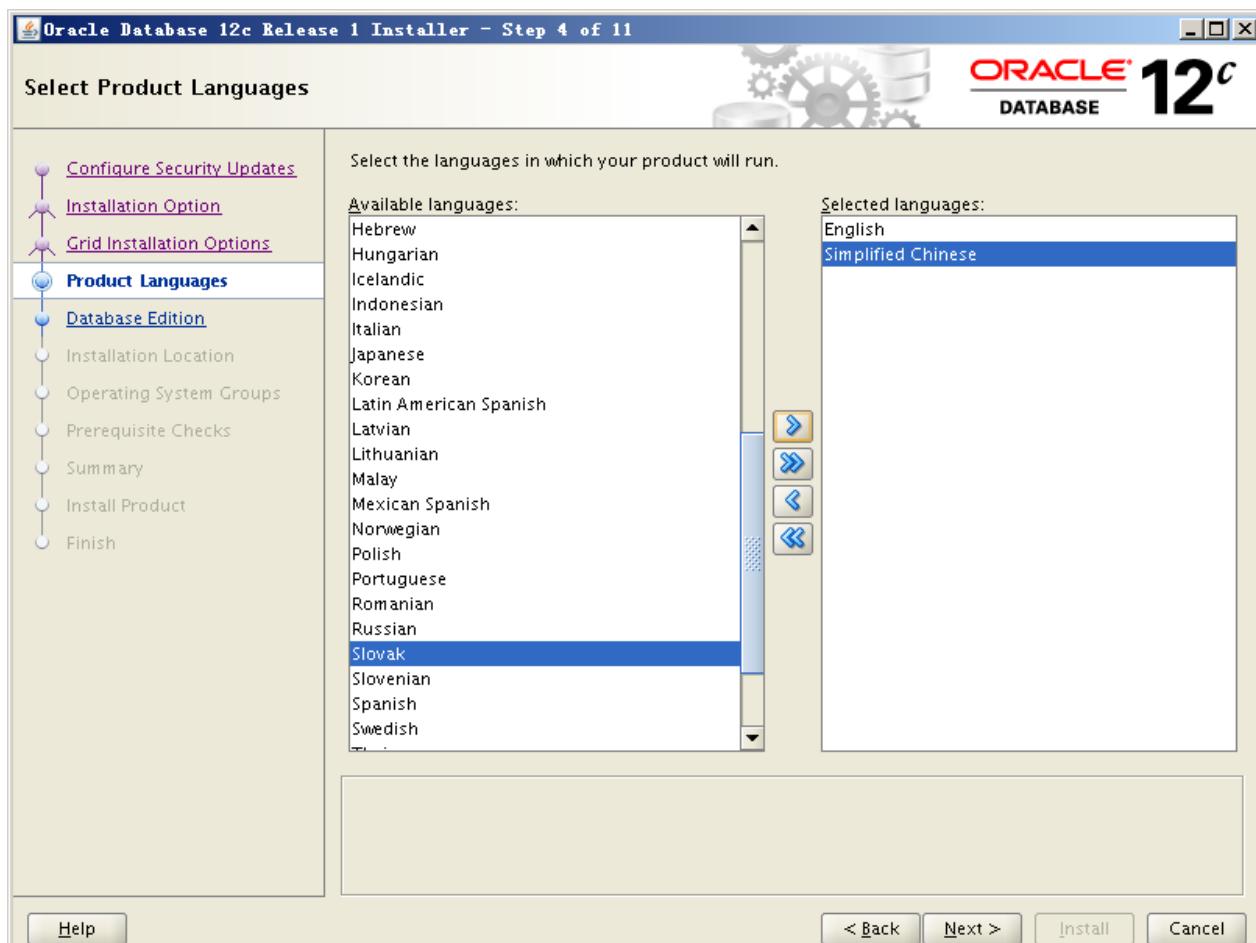
8.1 环境准备

```
[root@rdh ~]# yum -y install binutils compat-libcap1 gcc gcc-c++ glibc glibc.i686
glibc-devel glibc.i686 ksh libaio libaio.i686 libaio-devel libaio-devel.i686
libgcc libgcc.i686 libstdc++ libstdc++17.i686 libstdc++-devel
libstdc++-devel.i686 compat-libstdc++-33 compat-libstdc++-33.i686 libXi
libXi.i686 libXtst libXtst.i686 make sysstat
[root@dlp ~]# MEMTOTAL=$(free -b | sed -n '2p' | awk '{print $2}')
[root@rdh ~]# SHMMAX=$(expr $MEMTOTAL / 2)
[root@rdh ~]# SHMMNI=4096
[root@rdh ~]# PAGESIZE=$(getconf PAGE_SIZE)
[root@rdh ~]# cat >> /etc/sysctl.conf << EOF
fs.aio-max-nr = 1048576
fs.file-max = 6815744
kernel.shmmmax = $SHMMAX
kernel.shmall = `expr \$SHMMAX / \$PAGESIZE \` \* `expr \$SHMMNI / 16 \```
kernel.shmmni = $SHMMNI
kernel.sem = 250 32000 100 128
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
EOF
[root@rdh ~]# sysctl -p
[root@rdh ~]# i=54321; for group in oinstall dba backupdba oper dgdba kmdba; do
> groupadd -g $i $group; i=`expr $i + 1`
> done
[root@rdh ~]# useradd -u 1200 -g oinstall -G dba,oper,backupdba,dgdba,kmdba -d
/home/oracle oracle
[root@rdh ~]# passwd oracle
[root@rdh ~]# mkdir -p /u01/app/oracle
[root@rdh ~]# chown -R oracle:oinstall /u01/app
[root@rdh ~]# chmod -R 775 /u01
[root@rdh ~]# vi /etc/pam.d/login
session required pam_limits.so
[root@rdh ~]# vim /etc/security/limits.conf
```

```
oracle soft nproc 2047
oracle hard nproc 16384
oracle softnofile 1024
oracle hardnofile 65536
oracle soft stack 10240
oracle hard stack 32768
[root@rdh ~]# su oracle
[oracle@rdh ~]$ vim ~/.bash_profile
umask 022
export ORACLE_BASE=/u01/app/oracle
[oracle@rdh tmp]$ unzip linuxamd64_12102_database_1of2.zip
[oracle@rdh tmp]$ unzip linuxamd64_12102_database_2of2.zip
[oracle@rdh tmp]$ ./database/runInstaller
```







Oracle Database 12c Release 1 Installer - Step 6 of 11

Specify Installation Location

Configure Security Updates
Installation Option
Grid Installation Options
Product Languages
Database Edition
Installation Location
Operating System Groups
Prerequisite Checks
Summary
Install Product
Finish

Specify a path to place all Oracle software and configuration-related files installed by this installation owner. This location is the Oracle base directory for the installation owner.

Oracle base: /u01/app/oracle

Specify a location for storing Oracle database software files separate from database configuration files in the Oracle base directory. This software directory is the Oracle database home directory.

Software location: /u01/app/oracle/product/12.1.0/dbhome_1

Help < Back Next > Install Cancel

Oracle Database 12c Release 1 Installer - Step 7 of 11

Create Inventory

Configure Security Updates
Installation Option
Grid Installation Options
Product Languages
Database Edition
Installation Location
Create Inventory
Prerequisite Checks
Summary
Install Product
Finish

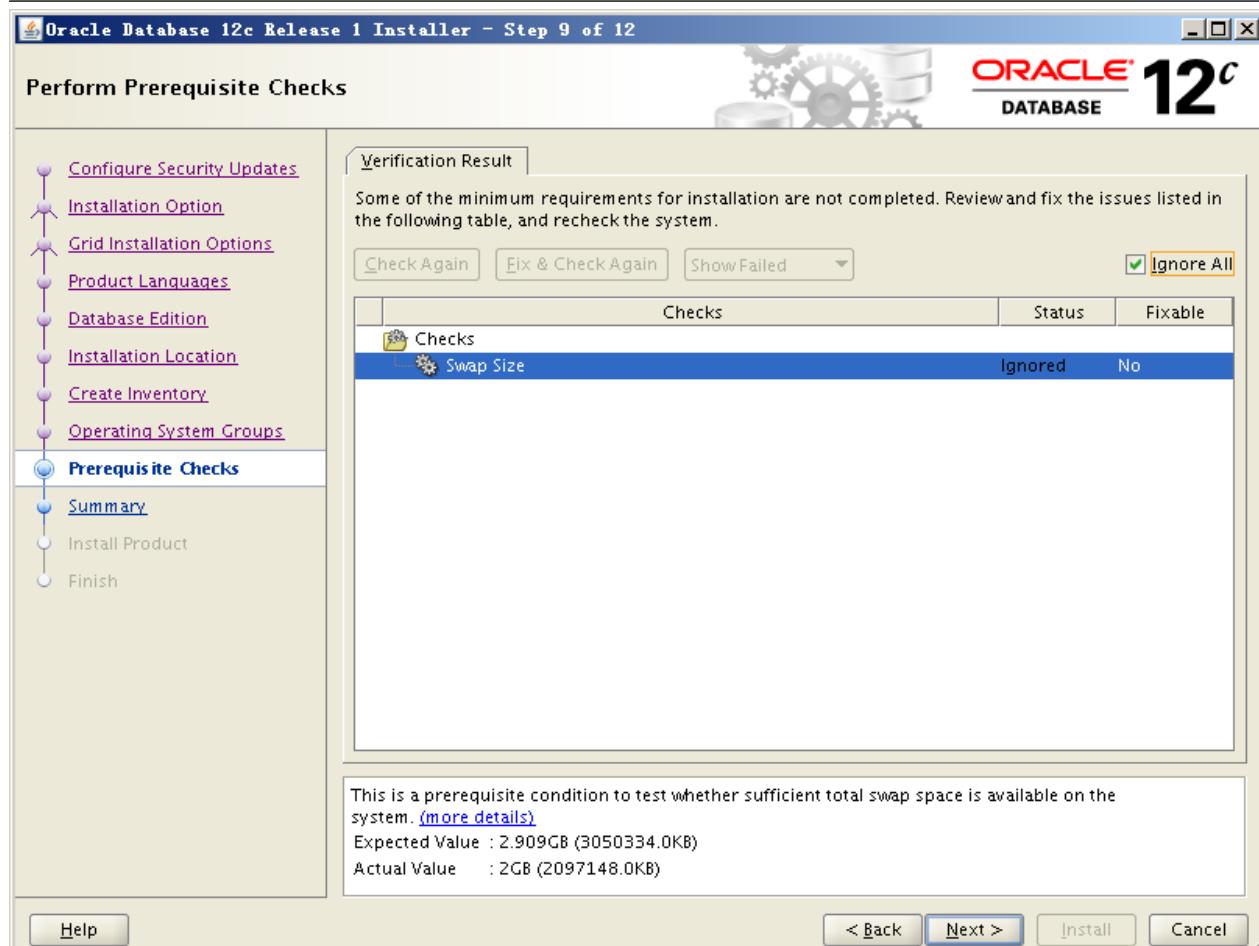
You are starting your first installation on this host. Specify a directory for installation metadata files (for example, install log files). This directory is called the "inventory directory". The installer automatically sets up subdirectories for each product to contain inventory data. The subdirectory for each product typically requires 150 kilobytes of disk space.

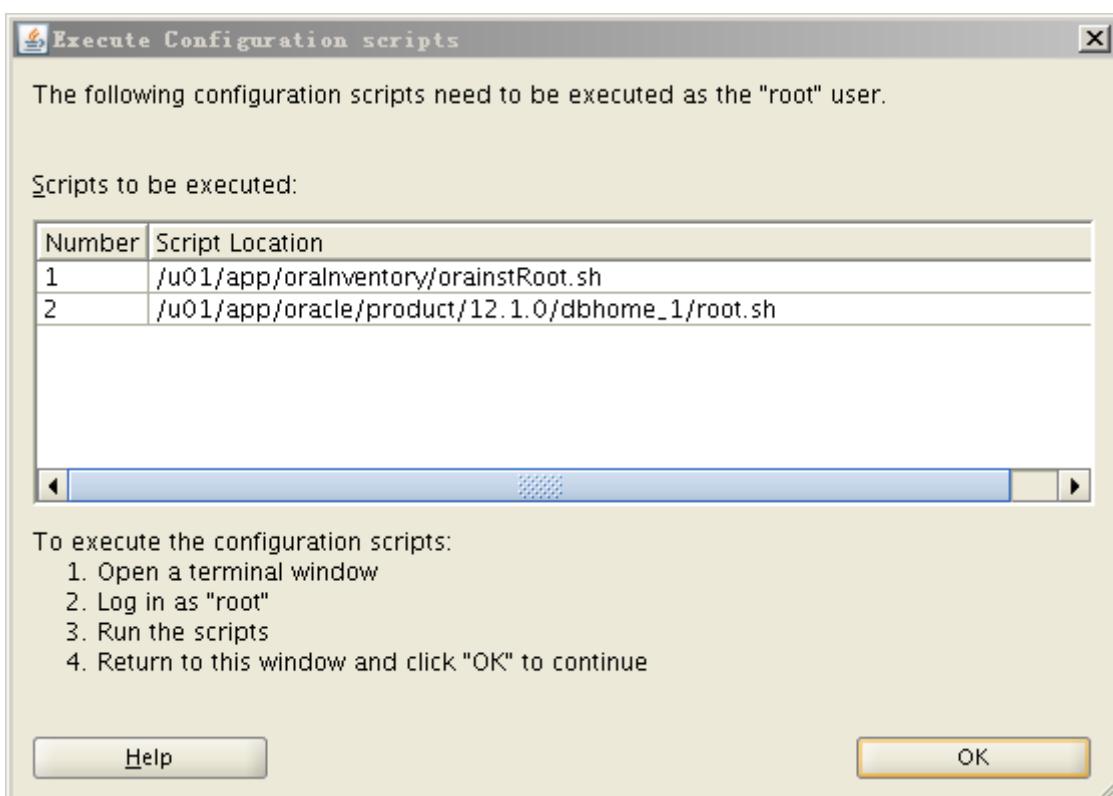
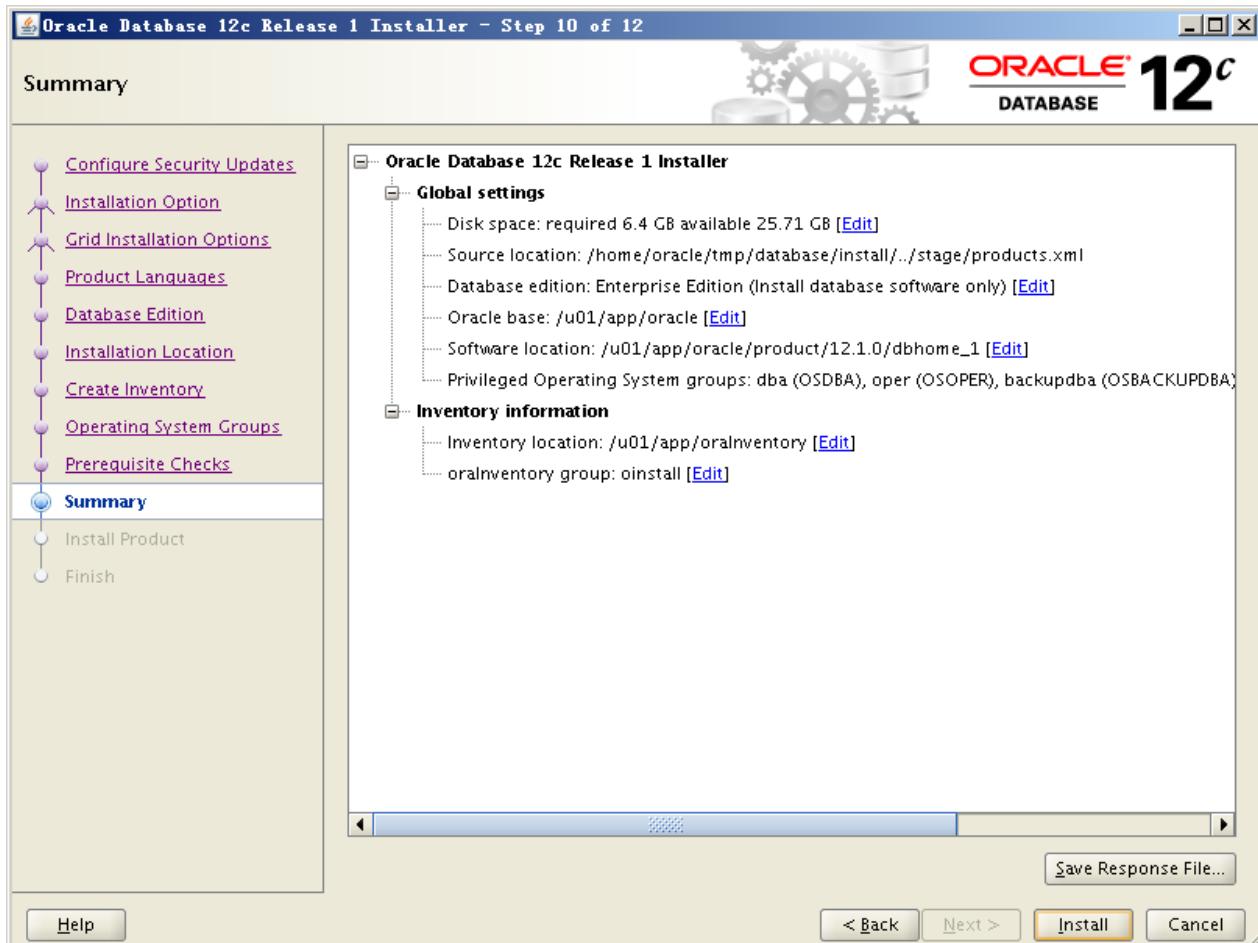
Inventory Directory: /u01/app/oralInventory

Specify an operating system group whose members have write permission to the inventory directory (oralInventory).

oralInventory Group Name: oinstall

Help < Back Next > Install Cancel



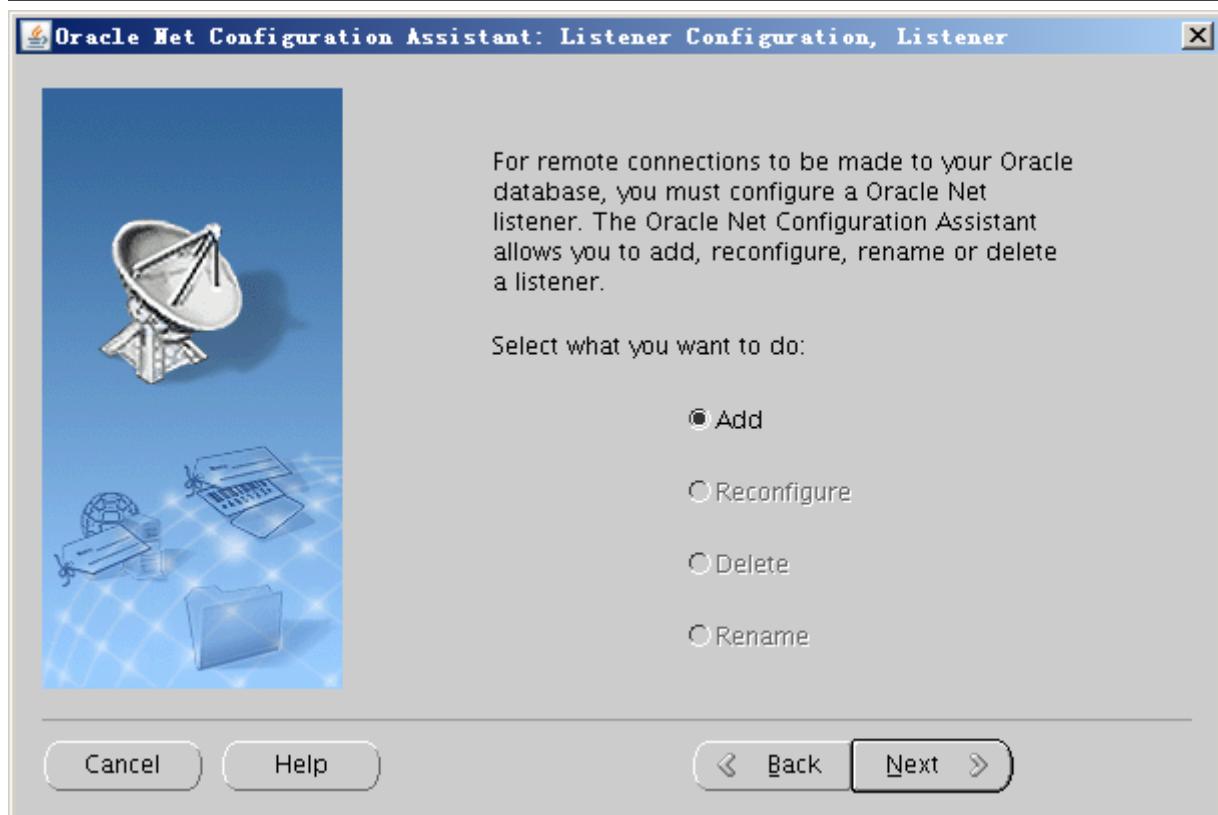
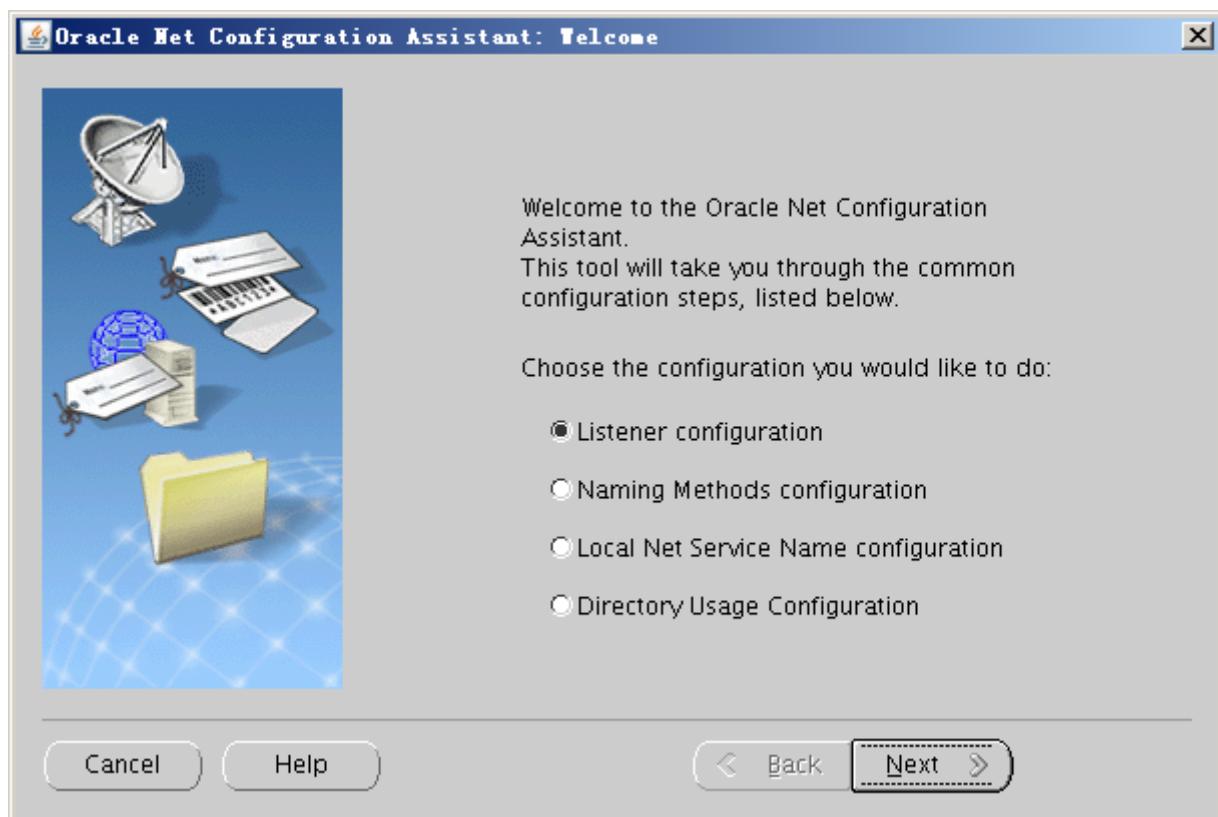


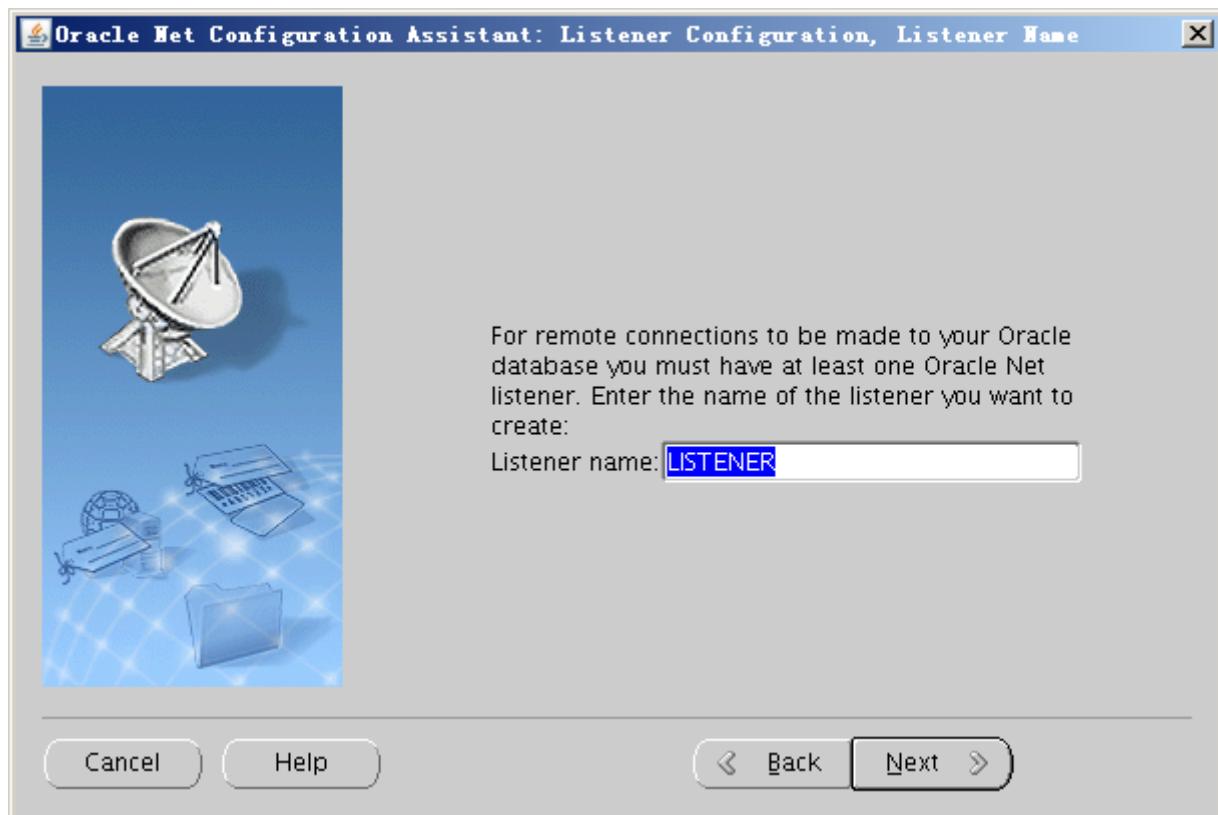
```
[root@rdh ~]# /u01/app/oralnventory/orainstRoot.sh
[root@rdh ~]# /u01/app/oracle/product/12.1.0/dbhome_1/root.sh
```

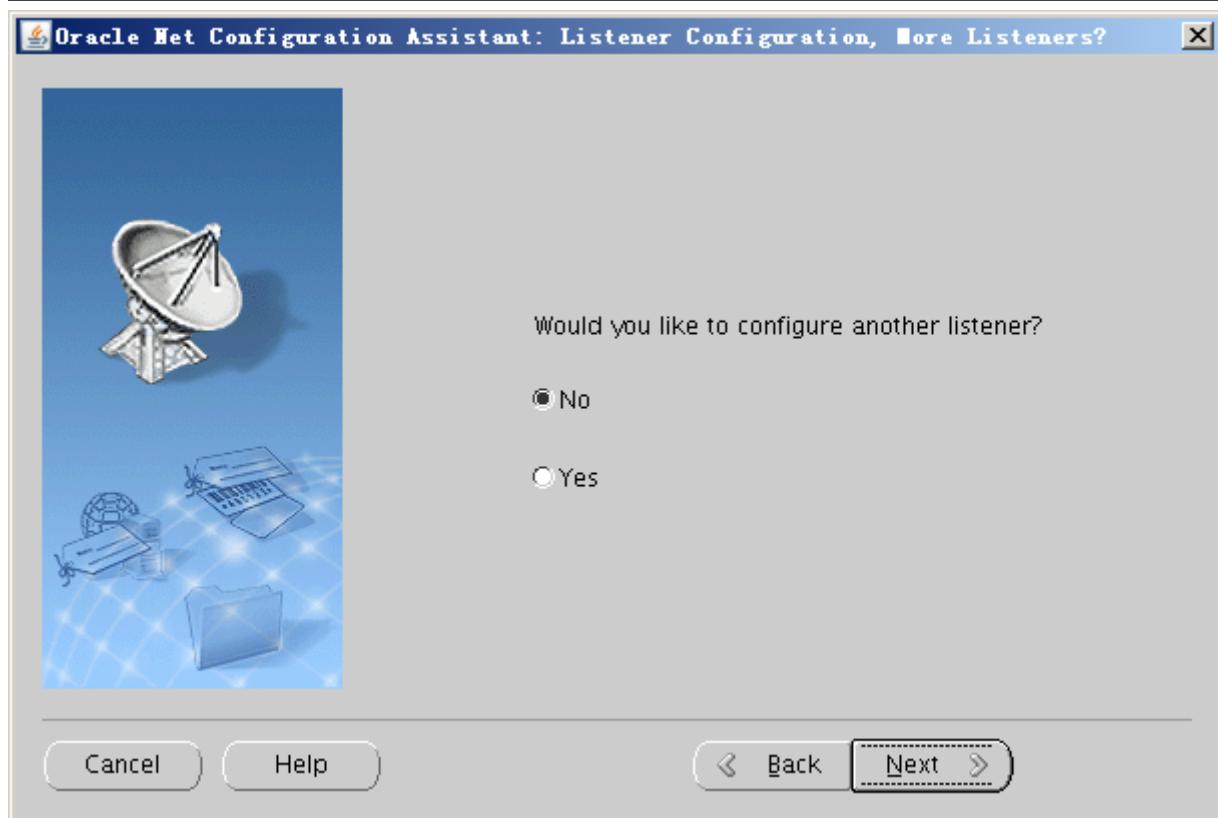
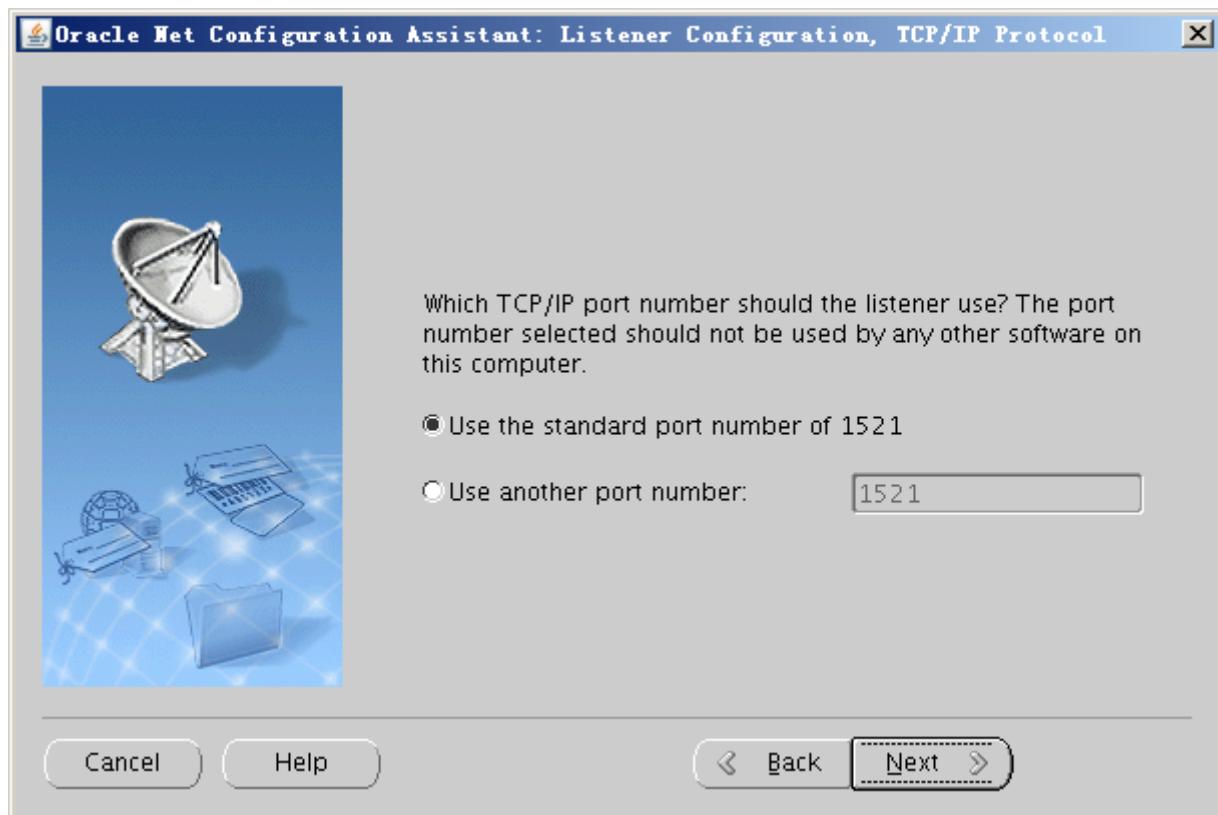


8.1 Add Oracle Net Listener

```
[oracle@rdh ~]$ vim ~/.bash_profile
export ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
export PATH=$PATH:$ORACLE_HOME/bin
[oracle@rdh ~]$ source ~/.bash_profile
```







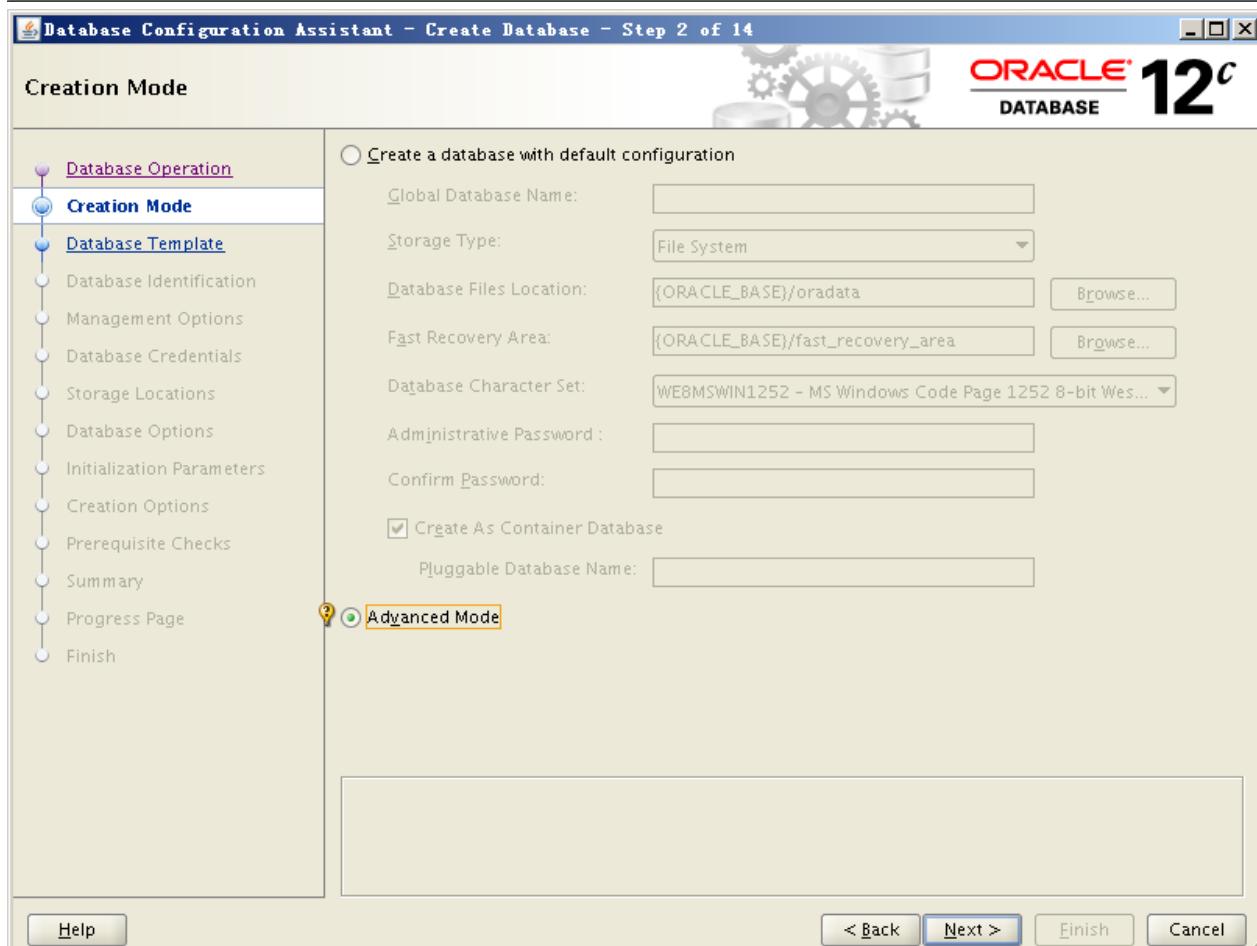
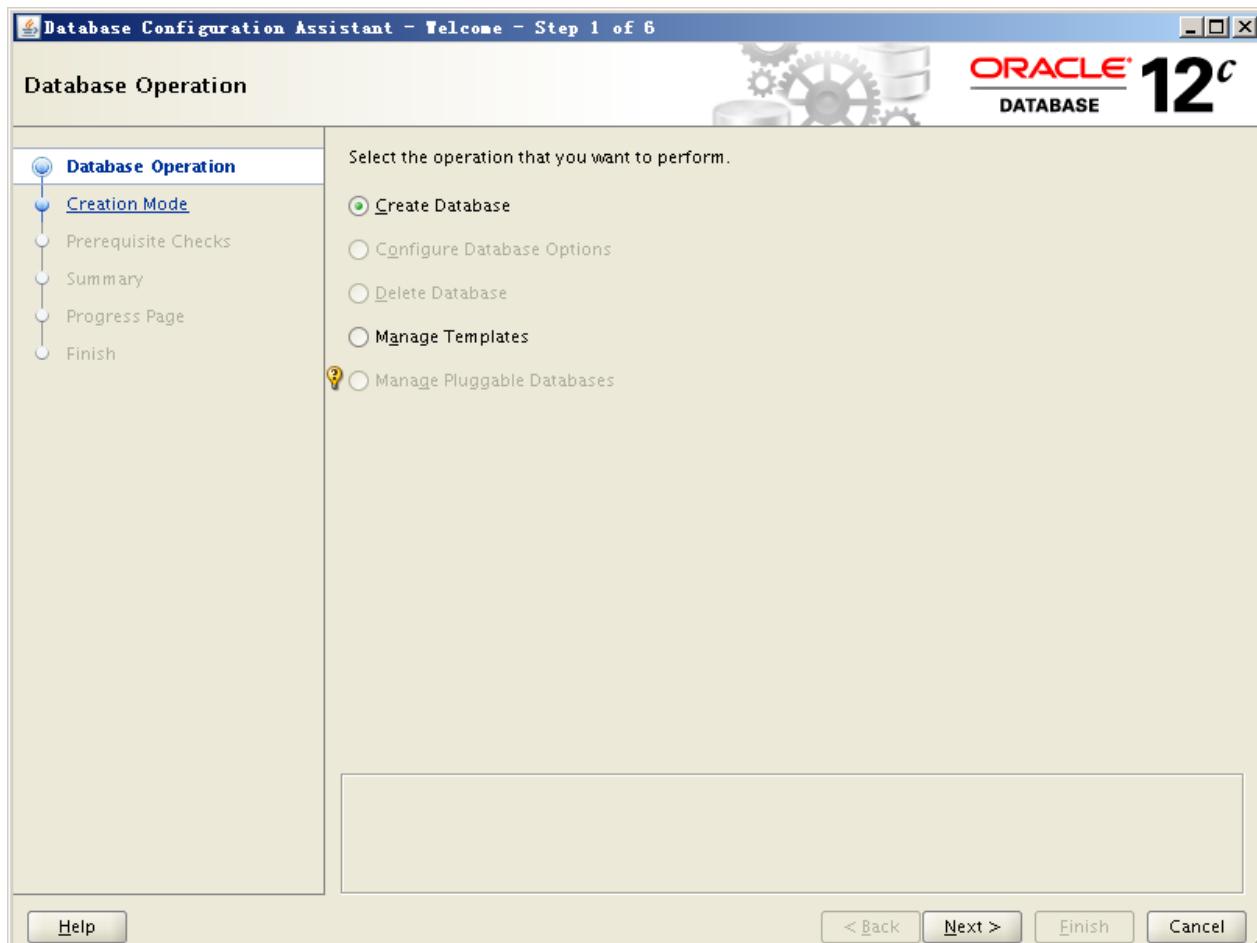


[oracle@rdh ~]\$ ss -napt

State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port
LISTEN	0	5	192.168.122.1:53	*:*
LISTEN	0	128		*:22
LISTEN	0	128	127.0.0.1:631	*:*
LISTEN	0	100	127.0.0.1:25	*:*
LISTEN	0	128	127.0.0.1:6010	*:*
LISTEN	0	128	127.0.0.1:6011	*:*
ESTAB	0	0	192.168.88.222:22	192.168.88.180:53048
ESTAB	0	52	192.168.88.222:22	192.168.88.180:52884
LISTEN	0	128	:22	:::*
LISTEN	0	128	::1:631	:::*
LISTEN	0	100	::1:25	:::*
LISTEN	0	128	::1:6010	:::*
LISTEN	0	128	::1:6011	:::*
TIME-WAIT	0	0	::1:18701	::1:6010

8.2 Create DataBase

[oracle@rdh ~]\$ dbca



Database Configuration Assistant - Create Database - Step 3 of 14

ORACLE[®] DATABASE 12c

Database Template

Templates that include datafiles contain pre-created databases. They allow you to create a new database in minutes, as opposed to an hour or more. Use templates without datafiles only when necessary, such as when you need to change attributes like block size, which cannot be altered after database creation.

Select a template for your database.

Select	Template	Includes Datafiles
<input checked="" type="radio"/>	General Purpose or Transaction Processing	Yes
<input type="radio"/>	Custom Database	No
<input type="radio"/>	Data Warehouse	Yes

Show Details...

Help < Back **Next >** Finish Cancel

Database Configuration Assistant - Create Database - Step 4 of 14

ORACLE[®] DATABASE 12c

Database Identification

Provide the identifier information required to access the database uniquely. An Oracle database is uniquely identified by a Global database name, typically of the form "name.domain". Additionally, a database is referenced by at least one Oracle instance which is uniquely identified from any other instance on this system by an Oracle system identifier (SID).

Global Database Name:

SID:

Create As Container Database

Creates a database container for consolidating multiple databases into a single database and enables database virtualization. A container database (CDB) can have zero or more pluggable databases (PDB).

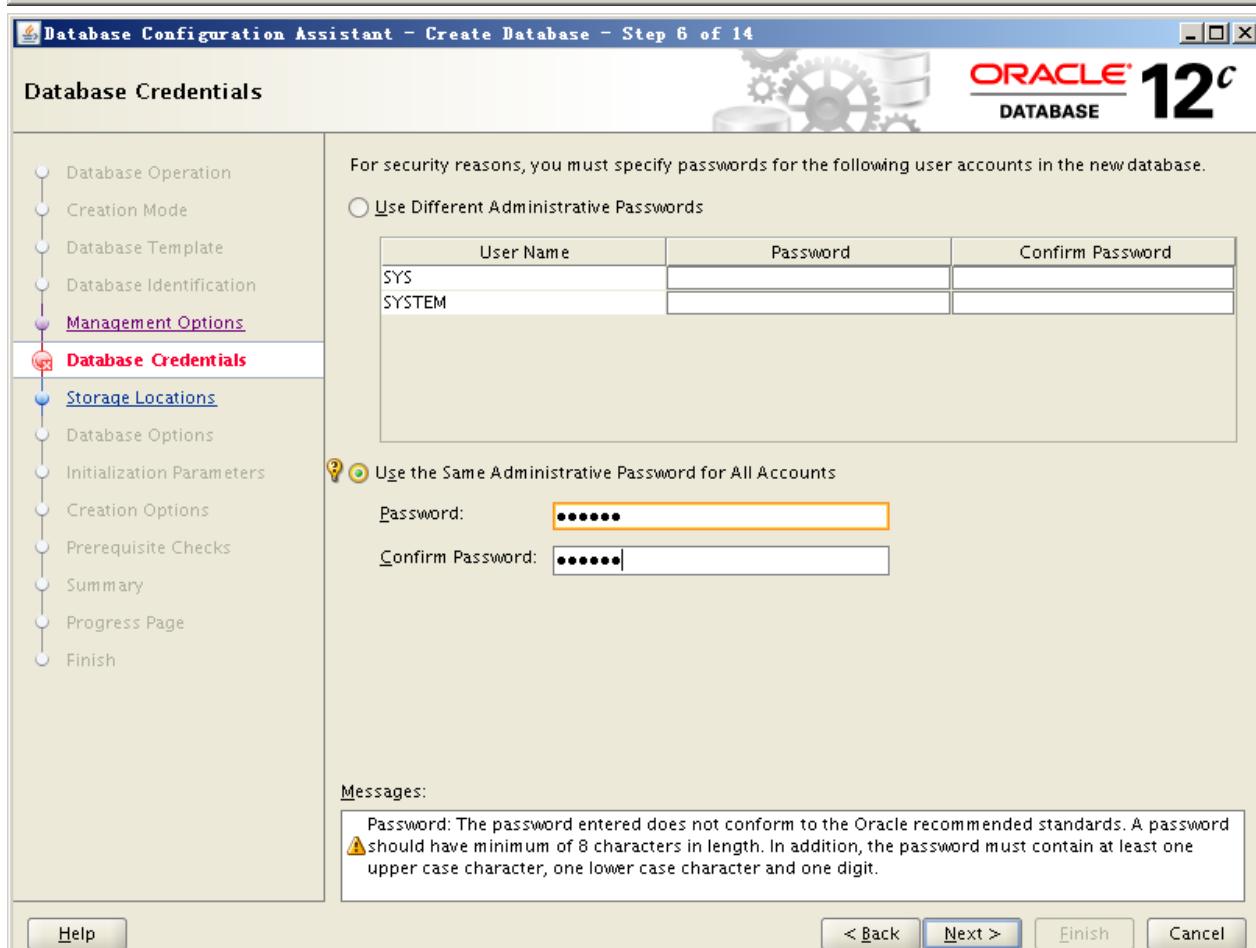
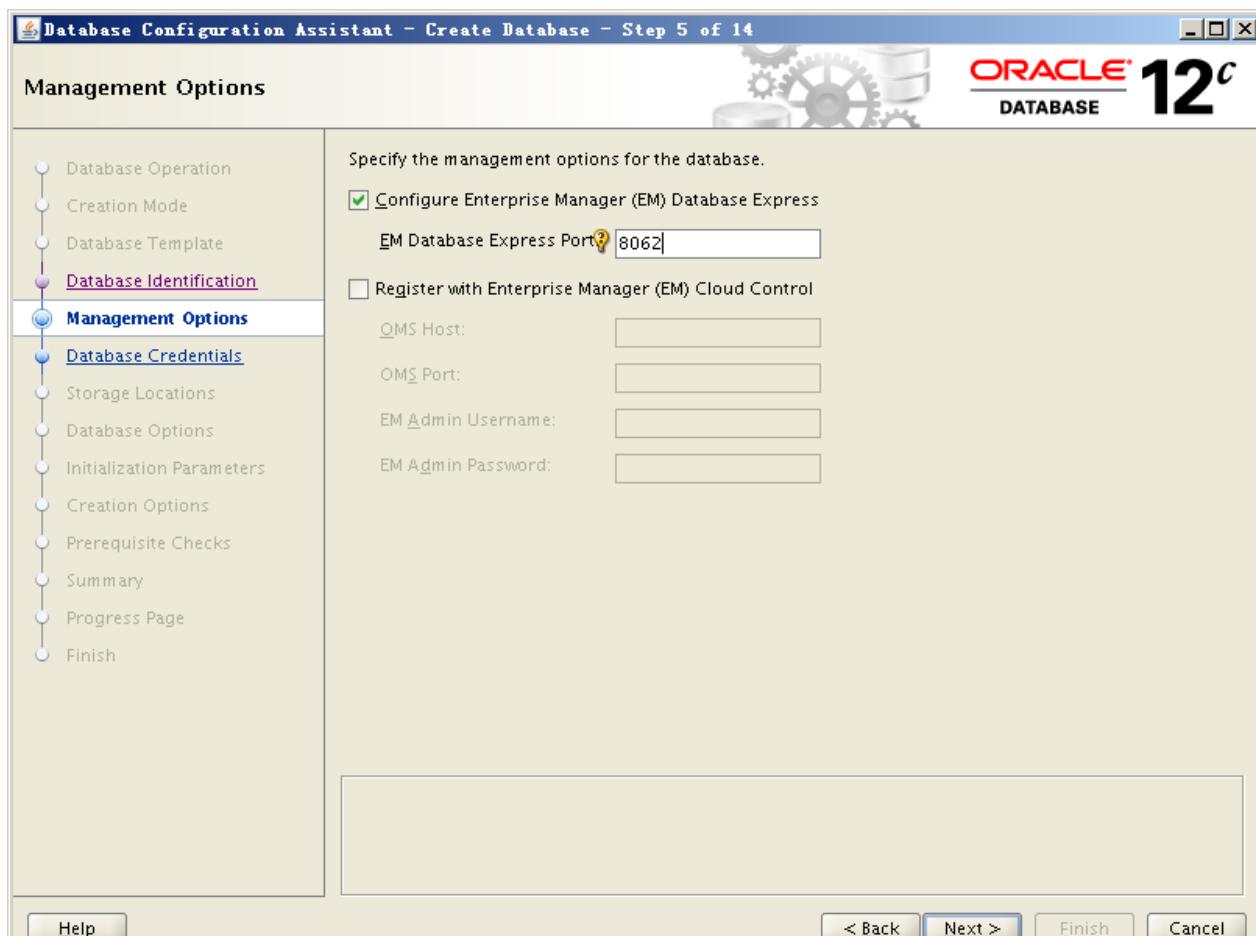
Create an Empty Container Database

Create a Container Database with one or more PDBs

Number of PDBs:

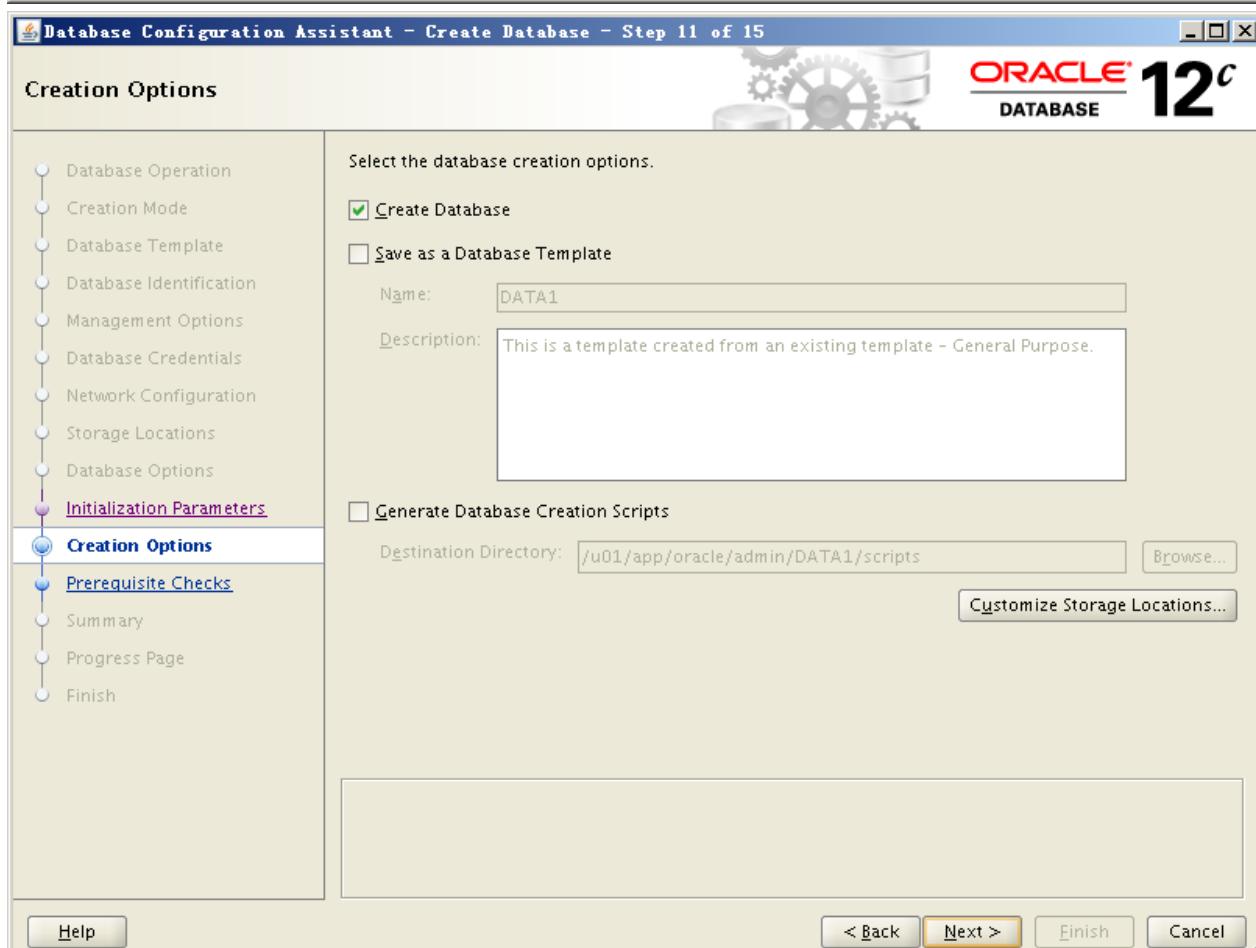
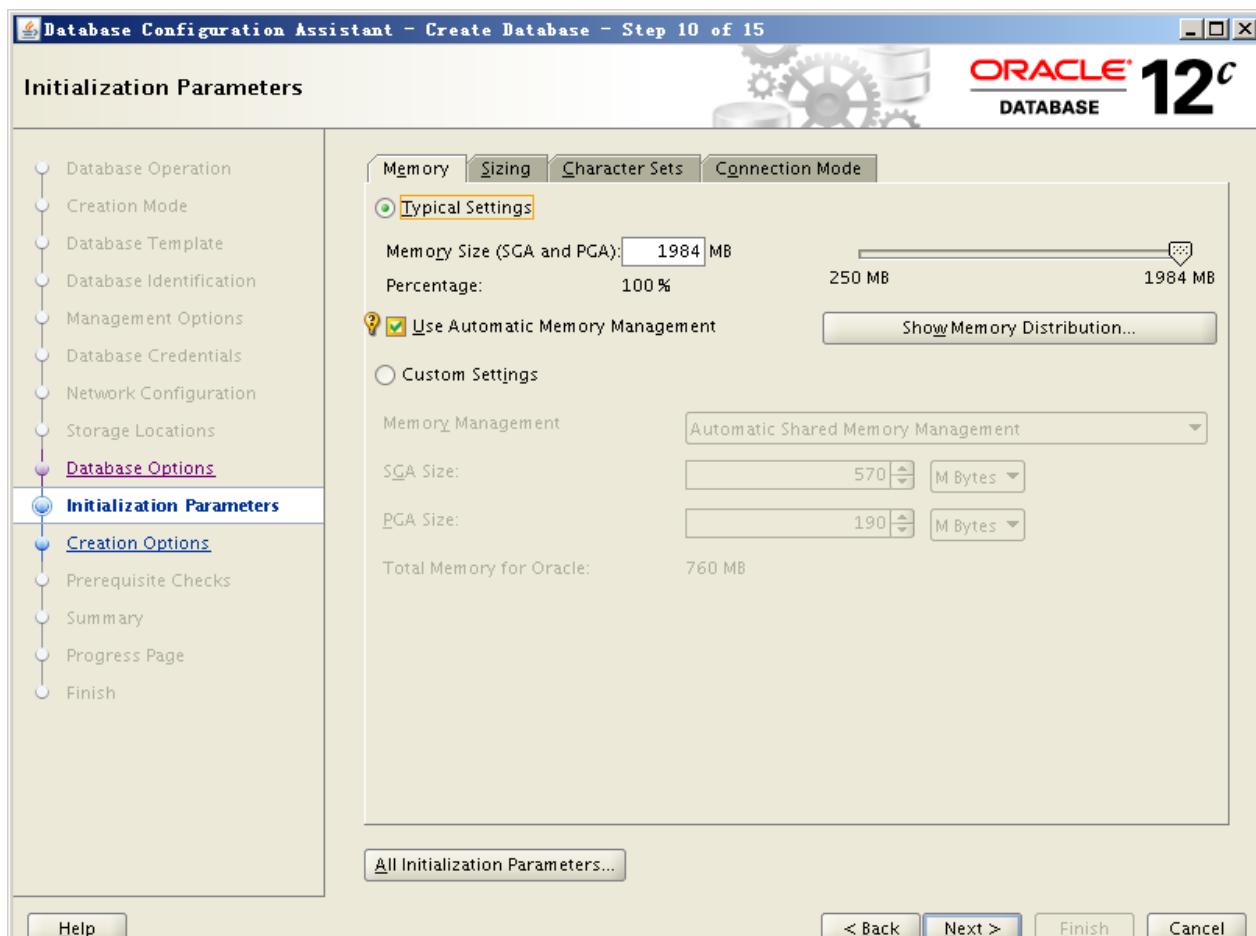
PDB Name:

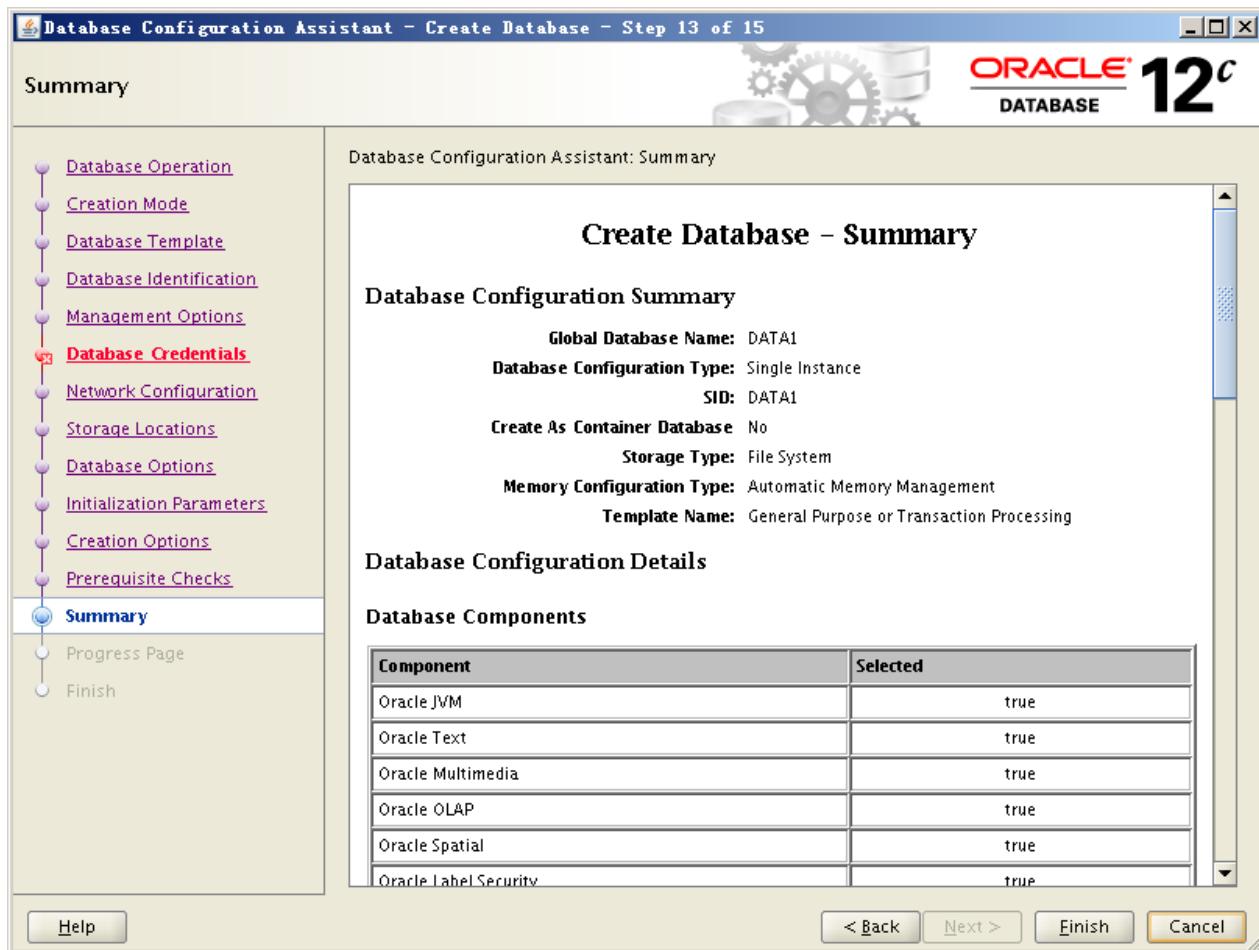
Help < Back **Next >** Finish Cancel



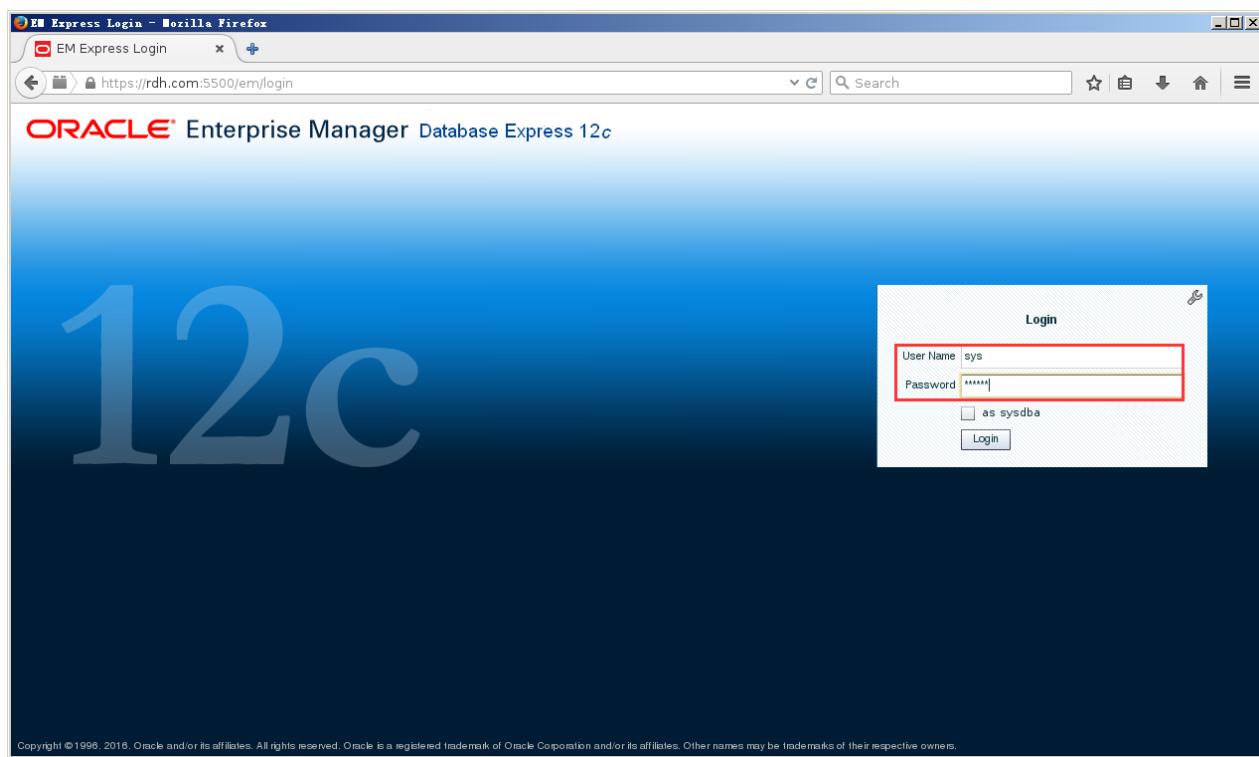
netca 提示 use another port number. 更换其他端口都是相同提示无法创建监听。这个问题是由/etc/hosts 表里 IP 或主机名问题导致。

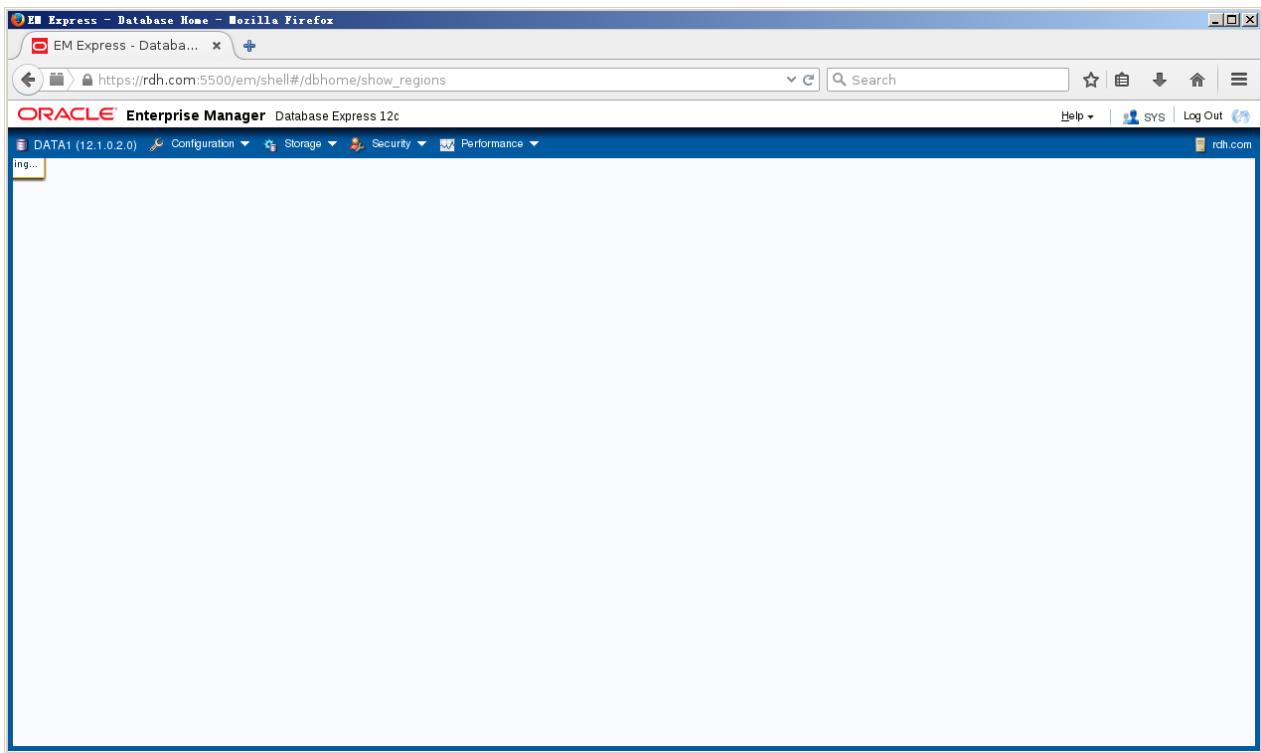






8.3 Manager





8.4 Oracle 数据库备份及还原

开启 Listener:

```
[oracle@rdh admin]$ lsnrctl

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 23-FEB-2016 09:48:40
Copyright (c) 1991, 2014, Oracle. All rights reserved.

Welcome to LSNRCTL, type "help" for information.

LSNRCTL> help
The following operations are available
An asterisk (*) denotes a modifier or extended command:

start          stop           status         services
version         reload          save_config    trace
spawn          quit            exit           set*
show*

LSNRCTL> 
```

8.5 RMAN 备份还原

```
[oracle@rdh bin]$ echo $ORACLE_HOME
[oracle@rdh bin]$ echo $ORACLE_SID
[oracle@rdh bin]$ export ORACLE_SID=DATA1
[oracle@rdh bin]$ rman target /
RMAN> startup
```

```
Oracle instance started  
database mounted  
database opened
```

```
Total System Global Area    3154116608 bytes
```

Fixed Size	2929352 bytes
Variable Size	771755320 bytes
Database Buffers	2365587456 bytes
Redo Buffers	13844480 bytes

RMAN 可以进行增量备份：数据库，表空间，数据文件

只有使用过的 block 可以被备份成 backup set

表空间与数据文件对应关系： dba_data_files / v\$datafile_header

在 noarchivelog 模式下，可以使用 RMAN 备份 read-only 和 offline 的表空间

ORACLE RMAN 停机备份：

备份

RMAN 连接上 ORACLE, WINDOWS 下在命令模式下

RMAN TARGET /

连接本地数据库用的是本地认证模式。RMAN 连接数据库必须在 `dedicate` 模式下。因此在 `share` 模式下的数据库应配置一个 `dedicate` 的连接用于 RMAN 连接。

如果要把控制文件、参数文件也一起备份

```
configure controlfile autobackup on; //打开 autobackup
```

```
configure controlfile autobackup off; //关闭 autobackup
```

关闭数据库

```
RMAN>shutdown immediate
```

mount 数据库 RMAN 的备份必须在 MOUNT 模式下，这也就是控制文件备份的重要。

```
RMAN>startup mount
```

最简单的备份

```
RMAN>backup database
```

就这一句就 OK 了

这样的备份，备份集在数据库默认位置。`%oracle_home%/ora92/database`

当然你也可以用 run 来灵活的定义你的备份。

```
RMAN>run {  
>allocate channel d1 type disk; //分配通道  
>backup full database      //全备份数据库  
>include current controlfile //包括当前的 controlfile  
>format 'e:\oracle\orders\db_%d_%s_%p_%t_%T'; //备份文件位置和文件名格式  
>release channel d1; //释放通道  
>}
```

恢复

1、数据文件损坏，而控制文件是好，或者已经恢复

RMAN 连上数据库

```
startup mount
```

```
restore database
```

```
recover database noredo;
alter database open resetlogs;
```

recover database noredo 该命令指示 RMAN 执行最后的恢复操作以准备打开这个数据库。因为是在 NOARCHIVELOG 模式下并且不应用任何归档的重做日志并且丢失了联机重做日志，所以要求在这条命令中使用 noredo 参数。

最后使用 alter database open resetlogs 打开数据库。由于已经还原了控制文件并且需要重新构建重做日志，所以必须用 resetlogs。

2、还原控制文件

```
startup nomount;
set dbid =
restore controlfile from autobackup ;
alter database mount;
restore database;
recover database noredo;
alter database open resetlogs;
alter database open;
```

在这个例子中有一个 DBID 这个可以

select * from v\$database 中查到。但是一个数据库在控制文件坏掉了不能 OPEN 如何能看到呢这就在平时把这个 DBID 记下来。

这是最简单的用 RMAN 备份与恢复的例子，但从中可以看到 RMAN 备份与恢复的梗概。

ORACLE RMAN 在线备份：

1. ORACLE RMAN 在线备份之前需要切换日志方式为归档日志；

a. 关闭数据库

```
SQL> shutdown immediate;
```

b. 启动数据库到 mount 状态

```
SQL> startup mount;
```

c. 启用归档模式

```
SQL> alter database archivelog;
```

d. 查看修改后的数据库备份和恢复策略及归档文件的位置

```
SQL> archive log list;
```

注意：修改成 archive 模式之后，之前所有的数据库备份均无效。

e. 修改相应的初始化参数

Oracle10g 之前，你还需要修改初始化参数使数据库处于自动归档模式。

可用 SQL> show parameter log_archive_start; 查看

NAME	TYPE	VALUE
------	------	-------

```
log_archive_start          boolean    FALSE
```

```
SQL> alter system set log_archive_start=true scope=spfile;
```

重启数据库此参数生效，此时数据库处于自动归档模式。

当然你也可以不做第 5 步，直接

```
SQL>archive log start
```

使数据库启用自动归档，但是重启后数据库仍然处于手工归档模式。

2. 运行：RMAN target /

3. RMAN 信息保存：默认保存在 control file 中，保存周期 7 天

调整: alter system set control_file_record_keep_time=天数;

4. 搭建独立数据库保存 RMAN 备份信息

由于只有一个数据库，就建在本身数据库上

- a. 创建表空间 RC: create tablespace rc datafile size 10M autoextend on next 1M
- b. 创建用户 RC:

```
CREATE USER rc IDENTIFIED BY rc TEMPORARY TABLESPACE temp DEFAULT TABLESPACE rc  
QUOTA UNLIMITED ON rc;
```

- c. 授权 RC: GRANT RECOVERY_CATALOG_OWNER TO rc;

d. 搭建:

```
rman catalog rc/rc@orcl
```

```
RMAN>create catalog;
```

```
RMAN>exit
```

```
rman target / catalog rc/rc@orcl
```

```
RMAN>register database;
```

e. 使用:

```
rman target / catalog rc/rc@orcl
```

这种连接方式后，数据就会在控制文件和 catalog 各存一份

5. 全局参数配置:

查看: show all;

修改: configure 参数名称 具体设置

例如: 修改是否自动保存 control file: configure controlfile autobackup on;

恢复默认值: configure 参数名称 clear;

关键参数:

a. 保存周期: retention policy

默认是 redundancy 1: 保留一个备份;

可用值: recovery window of 7 days: 保留可以满足 7 天恢复周期的备份

根据条件检查: report obsolete: RMAN 会根据保存周期参数来列出可以删除的备份

删除多于备份: delete obsolete

b. 优化备份: backup optimization: RMAN 会自动忽略已经备份过的内容(数据文件, 归档日志, 备份块)

前提: 备份指定同一个 channel

c. 默认备份渠道: default device type to disk: 默认备份到磁盘, 路径为 flash recovery area

渠道类型:

disk: 文件系统路径

flash recovery area: 默认路径

sbt: 磁带设备

修改到磁盘其他路径: configure channel device type disk format '路径/%U';

例如: configure channel device type disk format '/tmp/movedata/%U';

6. 备份结果

backupset: backup (as backupset) 备份内容, 里面分割成一个或多个 backup piece, 只有该类型备份可以进行压缩。

copy: backup as copy 备份内容

按类型查看:

backupset 查看: list backup summary (list backupset summary)

查看详情: list backupset BS

copy 查看: list copy

按内容查看:

整个数据库: list backup of database;

tablespace: list backup of tablespace users;

数据文件: list backup of datafile n;

控制文件: list backup of controlfile;

归档日志: list archivelog all;

按规则查看:

查看根据保存规则可删除文件: report obsolete;

查看根据保存规则需要备份内容: report

RMAN 和 OS 结合检查: corsscheck 内容;

7. 备份方式

full: 全备;

Incremental: 增量备份

可以增量备份的类型: 数据库, 数据文件, 表空间

a. 首先需要做 level 0 备份作为基础。例如: backup incremental level 0 备份内容; (备份内容: 所有使用过的 data block, 和 image copy 不同)

b. 增量类型:

累计增量: backup cumulative level 1 备份内容;

差异增量: backup incremental level 1 备份内容;

区别: 累计增量始终是基于 level 0 的备份;

第一次差异增量是基于 level 0 的备份; 从第二开始就是基于前一次增量备份

c. Image 备份增量方式:

第一次: 以 Image 全备为基础;

第二次: 基于全备, 做增量备份; 完成后合成一个 Image 全备

第三次: 基于第二次的 Image 全备, 做增量备份; 完成后合成一个 Image 全备
实现:

例如: 针对 tablespace example

RUN {

```
RECOVER COPY OF tablespace example WITH TAG 'incr_update';
BACKUP INCREMENTAL LEVEL 1 FOR RECOVER OF COPY WITH TAG 'incr_update'
    tablespace example;
}
```

d. 开启参数"block change tracking", 可以提高速度

查看状态: SELECT status FROM v\$block_change_tracking;

默认值: DISABLED

开启: ALTER DATABASE ENABLE BLOCK CHANGE TRACKING; (默认存放路径 OMF 中的 DB_CREATE_FILE_DEST)

设置文件路径:

```
ALTER DATABASE ENABLE BLOCK CHANGE TRACKING USING FILE
'/u01/oradata/MYSID/rman_change_track.f' REUSE;
```

关闭: ALTER DATABASE DISABLE BLOCK CHANGE TRACKING;

备份并检查: `bakcup check logical` 备份内容;

不备份只是检查文件: `backup validate` 备份内容;

如果检查有报错, 查看: `v$backup_corruption; v$copy_corruption`

8. 备份内容:

整个数据库: `RMAN>backup database;`

经典整库备份: `backup as compressed backupset database include current controlfile plus archivelog delete input;`

tablespace: `RMAN>backup tablespace 名字;`

数据文件: `RMAN>backup datafile n; (n: 具体的数据文件编号 select file_name, file_id, tablespace_name from dba_data_files;)`

控制文件: `RMAN>backup current controlfile;`

或者 `RMAN>backup database include current controlfile;`

日志文件: `RMAN>backup archivelog all;`

或者 `RMAN>backup database plus archivelog;`

参数文件: `RMAN>backup spfile;`

9. 还原

a. 完全恢复

方法一: 从最近的备份集恢复整个数据库, 数据库会自动运行 redo 和 archive 日志 (完全恢复):

`SQL>shutdown immediate`

`SQL>startup mount`

`RMAN>restore database;`

`RMAN>recover database;`

`RMAN>sql 'alter database open';`

方法二: 从 tag 恢复整个数据库, 数据库也会运行 redo 和 archive 日志 (完全恢复), 结果与上面的脚本一样:

1. 查看标签:

`RMAN> list backupset summary;`

Key	TY	LV	S	Device	Type	Completion Time	#Pieces	#Copies	Compressed	Tag
25	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T104634
28	B	O	A	DISK		25-JUL-11	1	1	NO	TAG20110725T104645
29	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T104711
30	B	F	A	DISK		25-JUL-11	1	1	NO	TAG20110725T104713
31	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T105333
32	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T105350
33	B	1	A	DISK		25-JUL-11	1	1	NO	TAG20110725T105353
34	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T105408
35	B	F	A	DISK		25-JUL-11	1	1	NO	TAG20110725T105411
36	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T111403
37	B	1	A	DISK		25-JUL-11	1	1	NO	TAG20110725T111405
38	B	A	A	DISK		25-JUL-11	1	1	NO	TAG20110725T111421
39	B	F	A	DISK		25-JUL-11	1	1	NO	TAG20110725T111423

```
SQL>shutdown immediate;
SQL>startup mount;
RMAN>restore database from tag TAG20110725T104645;
RMAN> recover database from tag TAG20110725T104645;
RMAN> alter database open;
```

b. 不完全恢复:

```
SQL>shutdown immediate;
SQL>startup mount;
RMAN>restore database from tag TAG20110725T104645;
RMAN>recover database until time "to_date('2011-08-04 15:37:25', 'yyyy/mm/dd
hh24:mi:ss')";
RMAN>alter database open resetlogs;
```

注意：使用后所有的备份集都无效了，确保安全需要重新对数据库进行全备（ORACLE10G之后，resetlog 之前的备份还是可以用的）

关键表空间恢复 (system / undotbs1 / sysaux):

```
SQL>shutdown abort
SQL>startup mount
RMAN>restore tablespace 名字;
RMAN>recover tablespace 名字;
RMAN>sql 'alter database open';
```

非关键表空间恢复 (example / users):

```
select * from v$datafile_header; 表空间与数据文件对应关系
SQL>alter database datafile 数字 offline;
RMAN>restore tablespace 名字;
RMAN>recover tablespace 名字;
SQL>alter database datafile 数字 online;
```

10. 删除备份

所有 backup 备份集: delete backup;
所有 copy 备份机: delete copy;
特定备份机: delete backupset 19;
删除根据保存规则可删除文件: delete obsolete;
删除过期的备份:
delete expired backupset;
delete expired copy;

11. RUN 块

例如:

```
RMAN> RUN {
    ALLOCATE CHANNEL c1 DEVICE TYPE sbt;
    ALLOCATE CHANNEL c2 DEVICE TYPE sbt;
    ALLOCATE CHANNEL c3 DEVICE TYPE sbt;
    BACKUP
    INCREMENTAL LEVEL = 0
    FORMAT '/disk1/backup/df_%d_%s_%p.bak'
    (DATAFILE 1, 4, 5 CHANNEL c1)
```

```
(DATAFILE 2, 3, 9 CHANNEL c2)
(DATAFILE 6, 7, 8 CHANNEL c3);
ALTER SYSTEM ARCHIVE LOG CURRENT;
}
```

12. 外部变量:

语言: set nls_lang=american

日期: set nls_date_format=yyyy-mm-dd....

13. RMAN script

前提条件: 有 catalog

写法: (global 表示可以由多个数据库调用)

create global script. 名

comment "备注说明"

{脚本内容}

例如:

```
create global script. abc
comment "test"
{backup current controlfile;}
```

调用: run {execute script. 名}

例如: run {execute script. abc;}

查看: print script. 名

改写:

```
replace global script. 名
comment "备注说明"
{脚本内容}
```

删除: delete script. 名;

14. 永久保留备份

条件是备份不能保留在 flash recovery area 中;

a. 创建备份:

```
RUN
{
ALLOCATE CHANNEL c1 DEVICE TYPE disk format '/tmp/autobackup/%U';
BACKUP tablespace example;
}
```

b. 查找该备份:

```
list backupset of tablespace example;
```

c. 修改属性为永久

```
change backupset 编号 keep forever nologs;
```

15. 建立多个固定大小的备份

例如: example 测试备份大小是大于 50M

run

{

```
allocate channel c1 device type disk maxpiecesize 10M format '/tmp/autobackup/%U';
backup tablespace example;
}
```

/tmp/autobackup 目录下有六个文件

16. 运行脚本: backup recovery area

备份内容:

- a. control file autobackup;
- b. incremental backup sets

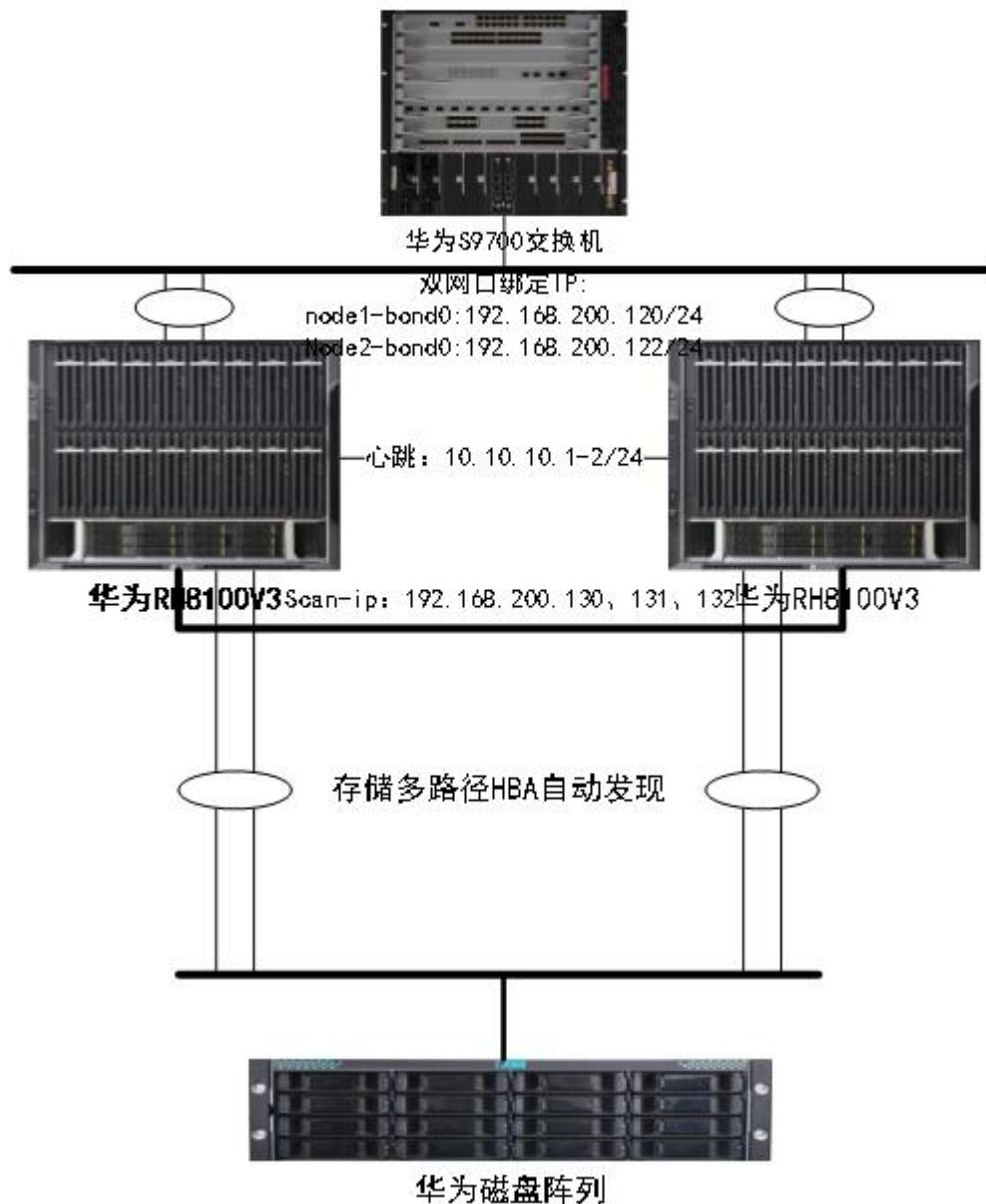
17. block change tracking

- a. 针对整个数据库;
- b. 默认存放路径: background_dump_dest

九、Oracle RAC 集群架构详解

方案拓扑

XMJ RAC实现拓扑



软硬件环境

操作系统: Centos 6.5。

安装软件: Oracle RAC 11G R2、Oracle GRID 11G R2。

硬件环境: 典型的 FC-SAN 架构, 华为高性能服务器 (512G 内存), HBA 卡, 光纤盘阵。

下载地址

<http://www.oracle.com/technetwork/cn/database/enterprise-edition/downloads/112010-linx8664soft-098700-zhs.html>

适用于 Linux x86-64 的 Oracle Database 11g 第 2 版 (11.2.0.1.0)

- ◆ [linux.x64_11gR2_database_1of2.zip](#) (1,239,269,270 字节) (cksum - 3152418844)
- ◆ [linux.x64_11gR2_database_2of2.zip](#) (1,111,416,131 字节) (cksum - 3669256139)

使用说明

1. 所有文件都是 .zip 格式。这里提供了一个解压缩工具以备不时之需。
2. 下载这两个文件，并将它们解压缩到同一个目录中。
3. 这里提供了安装指南和 Oracle Database 11g 通用文档。
4. 在[这里](#)查阅该产品的认证表。

适用于 Linux x86-64 的 Oracle Database 11g 第 2 版 Client (11.2.0.1.0)

- ◆ [linux.x64_11gR2_client.zip](#) (706,187,979 字节) (cksum - 3654981652)

包含适用于 Linux 的 Oracle Client 库。如果您只需要客户端库，则需要下载该文件。

适用于 Linux x86-64 的 Oracle Database 11g 第 2 版 Grid Infrastructure (11.2.0.1.0)

- ◆ [linux.x64_11gR2_grid.zip](#) (1,052,897,657 字节) (cksum - 3369676398)

包含 Grid Infrastructure 软件，其中包括 Oracle Clusterware、自动存储管理 (ASM) 和 ASM Cluster File System。在网格环境中安装 Oracle Real Application Clusters、Oracle Real Application Clusters One Node 或其他 Oracle 软件之前，需要先下载和安装该文件。

适用于 Linux x86-64 的 Oracle Fusion Middleware Web Tier Utilities 11g (11.1.1.2.0)

- ◆ [ofm_webtier_linux_11.1.1.2.0_64_disk1_1of1.zip](#) (1,324,028,787 字节) (cksum - 949127178)

包含 Oracle HTTP Server 及相关模块。如果您希望将 HTTP 设置为通过 Apache HTTP Server 访问数据库，则需要下载该文件。

适用于 Linux x86-64 的 Oracle Database Gateways 11g 第 2 版 (11.2.0.1.0)

- ◆ [linux.x64_11gR2_gateways.zip](#) (664,343,081 字节) (cksum - 3517244335)

包含连接非 Oracle 数据库的 Oracle 数据库网关。如果您希望设置异构数据集成环境，则需要下载该文件。

Oracle Database 11g 第 2 版 Examples

- ◆ [linux.x64_11gR2_examples.zip](#) (555,366,950 字节) (cksum - 1191529449)

包含有关如何使用 Oracle Database 的示例。如果您是新接触 Oracle 且希望尝试一些文档中提供的示例，则需要下载该文件。

适用于 Linux x86-64 的 Oracle De-install Utility (11.2.0.1.0)

- ◆ [linux.x64_11gR2_deinstall.zip](#) (119,761,381 字节) (cksum - 2764327803)

适用于 Linux x86-64 的 Oracle De-install Utility (11.2.0.2.0)

- ◆ [linux.x64_11202_deinstall.zip](#) (110,436,214 字节) (cksum - 1897765276)

IP 地址和域名规划

主机名	网卡	IP 地址	网关	备注
node-rac1	bond0 (eth0+eth1)	192.168.200.120/24	192.168.200.1	node-rac1
node-priv1	eth2	10.10.10.1/24		node-priv1
node-vip1		192.168.200.121/24		node-vip1
node-rac2	bond0 (eth0+eth1)	192.168.200.122/24	192.168.200.1	node-rac2
node-priv2	eth2	10.10.10.2/24		node-priv2
node-vip2		192.168.200.123		node-vip2
rac-scan 地址		192.168.200.125、126、127		

ISCSI 存储 IP	eth3、eth4	192.168.1.1-2/24		
initiator-01	eth3、eth4	192.168.1.3-4/24		node-rac1 用
initiator-02	eth3、eth4	192.168.1.5-6/24		node-rac2 用

集群及主机域名

集群名称	rac-cluster			
集群数据库	xmldb		数据库实例	
			xmldb1 xmldb2	
账号密码	RAC 账号	oracle/oracle		grid/grid
数据库 SYS 密码	xmjrac			
oracle 主目录	/u01/app/12.1.0			
主机 对应 的 ASM 序号	node-rac1	ASM1	node-rac2	ASM2

存储磁盘及空间

OCR	10G	/dev/mapper/MP1	集群配置信息状态信息
OCR 镜像	10G	/dev/mapper/MP2	OCR 的镜像盘
FRA	10G	/dev/mapper/MP3	闪回数据
DATA1	600G	/dev/mapper/MP4	存储数据
DATA2	600G	/dev/mapper/MP5	镜像存储数据
DATA3	600G	/dev/mapper/MP6	镜像存储分区

配置本地 YUM 源【node1、node2】

```
[root@node-rac1 ~]# cd /etc/yum.repos.d/
[root@node-rac1 yum.repos.d]# ls
CentOS-Base.repo  CentOS-Debuginfo.repo  CentOS-Media.repo  CentOS-Vault.repo
[root@node-rac1 yum.repos.d]# mv CentOS-* /root/yum
[root@node-rac1 yum.repos.d]# vim local.repo
[local]
name=local yum server
baseurl=file:///mnt/media/
enabled=1
gpgcheck=0
[root@node-rac1 yum.repos.d]# yum clean all
Loaded plugins: aliases, changelog, downloadonly, fastestmirror, kabi, presto,
               : refresh-packagekit, security, tmprepo, verify, versionlock
Loading support for CentOS kernel ABI
Cleaning repos: local
Cleaning up Everything
0 delta-package files removed, by presto
[root@node-rac1 yum.repos.d]# yum makecache
Loaded plugins: aliases, changelog, downloadonly, fastestmirror, kabi, presto,
               : refresh-packagekit, security, tmprepo, verify, versionlock
Loading support for CentOS kernel ABI
Determining fastest mirrors
```

```
local | 4.0 kB 00:00 ...
local/group_gz | 220 kB 00:00 ...
local/filelists_db | 5.8 MB
00:00 ...
local/primary_db | 4.4 MB 00:00 ...
local/other_db | 2.7 MB 00:00 ...
```

Metadata Cache Created

关闭安全策略【node1、node2】

```
[root@node-rac1 ~]# service iptables stop
[root@node-rac1 ~]# chkconfig iptables off
[root@node-rac1 ~]# iptables -F
[root@node-rac1 ~]# iptables -X
[root@node-rac1 ~]# iptables -Z
[root@node-rac1 ~]# vim /etc/sysconfig/selinux
SELINUX=disabled
[root@node-rac1 ~]# setenforce 0
[root@node-rac1 ~]# getenforce
Permissive
[root@node-rac1 ~]# service NetworkManager stop
Stopping NetworkManager daemon: [ OK ]
[root@node-rac1 ~]# chkconfig NetworkManager off
配置完成后重启两个节点
```

组件环境安装【两个节点】

```
[root@node-rac1 ~]# yum -y install binutils compat-libcap1 compat-libstdc++-33
compat-libstdc++-33.i686    gcc      gcc-c++    glibc     glibc.i686   glibc-devel
glibc-devel.i686 ksh libgcc libgcc.i686 libstdc++ libstdc++.i686 libstdc++-devel
libstdc++-devel.i686 libaio libaio.i686 libaio-devel libaio-devel.i686 libXext
libXext.i686 libXtst libXtst.i686 libX11 libX11.i686 libXau libXau.i686 libxcb
libxcb.i686 libXi libXi.i686 make sysstat unixODBC unixODBC-devel
```

双网口绑定【双节点】

```
[root@node-rac1 ~]# vim /etc/sysconfig/network-scripts/ifcfg-bond0
DEVICE=bond0
ONBOOT=yes
IPADDR=192.168.200.120
BOOTPROTO=none
NETMASK=255.255.255.0
TYPE=Ethernet
GATEWAY=192.168.200.1
USERCTL=no
[root@node-rac1 ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=none
```

```
TYPE=Ethernet
USERCTL=no
MASTER=bond0
SLAVE=yes
[root@node-rac1 ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=none
TYPE=Ethernet
USERCTL=no
MASTER=bond0
SLAVE=yes
[root@node-rac1 ~]# vim /etc/modprobe.d/bond0.conf
[root@node-rac1 ~]# service network restart
设定完成后，重启服务器
```

调整 TMPFS 文件系统的大小【双节点】

```
[root@node-rac1 ~]# vim /etc/fstab
tmpfs           /dev/shm          tmpfs   defaults, size=3G
0 0
[root@node-rac1 ~]# mount -o remount /dev/shm
[root@node-rac1 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2        39G  5.2G  32G  15% /
tmpfs            3.0G  224K  3.0G  1% /dev/shm
/dev/sda1       194M   35M  150M  19% /boot
/dev/sda5        4.9G  138M  4.5G  3% /home
/dev/sr0         4.2G   4.2G    0 100% /media/CentOS_6.5_Final
```

调整基本 BUG【双节点】

```
[root@node-rac1 ~]# vim /etc/sysconfig/network
NOZEROCONF=yes
[root@node-rac1 ~]# vim /etc/sysctl.conf
kernel.panic_on_oops = 1
[root@node-rac1 ~]# vim /etc/sysconfig/network
ORACLEASM_SCANORDER="dm"
ORACLEASM_SCANEXCLUDE="sd"
[root@node-rac1 ~]# /etc/init.d/oracleasm restart
```

配置 DNS 解析【双节点】

```
[root@node-rac1 ~]# vim /etc/hosts
=====Public=====
192.168.200.120 node-rac1.xmj.com node-rac1
192.168.200.122 node-rac2.xmj.com node-rac2
=====Private=====
10.10.10.1 node-priv1.xmj.con node-priv1
```

10.10.10.2 node-priv2.xmj.com node-priv2

=====Vip=====

192.168.200.121 node-vip1.xmj.com node-vip1

192.168.200.123 node-vip2.xmj.com node-vip2

=====Scan=====

192.168.200.130 rac-scan.xmj.com rac-scan

192.168.200.131 rac-scan.xmj.com rac-scan

192.168.200.132 rac-scan.xmj.com rac-scan

【master】

[root@node-rac1 ~]# yum -y install bind bind-chroot caching-nameserver

[root@node-rac1 ~]# vim /etc/named.conf

[root@node-rac1 ~]# cd /var/named/

[root@node-rac1 named]# cp -P named.localhost xmj.com.zone

[root@node-rac1 named]# vim xmj.com.zone

\$TTL 1D

@ IN SOA xmj.com root.xmj.com. (

0 ; serial

1D ; refresh

1H ; retry

1W ; expire

3H) ; minimum

IN NS dns.xmj.com.

rac-scan IN A 192.168.200.130

rac-scan IN A 192.168.200.131

rac-scan IN A 192.168.200.132

dns IN A 192.168.200.120

[root@node-rac1 named]# cp -P xmj.com.zone 192.168.200.local

[root@node-rac1 named]# vim 192.168.200.local

\$TTL 1D

@ IN SOA xmj.com. root.xmj.com. (

0 ; serial

1D ; refresh

1H ; retry

1W ; expire

3H) ; minimum

IN NS dns.xmj.com.

130 IN PTR scan-rac.xmj.com.

131 IN PTR scan-rac.xmj.com.

132 IN PTR scan-rac.xmj.com.

【slave】

[root@node-rac2 Desktop]# vim /etc/named.conf

zone "xmj.com" IN {

type slave;

file "slaves/xmj.com.zone";

```
masters { 192.168.200.120; };  
};  
  
zone "200.168.192.in-addr.arpa" IN {  
    type slave;  
    file "slaves/192.168.200.local";  
    masters { 192.168.200.120; };  
};  
[root@node-rac2 Desktop]# /etc/init.d/named start  
Generating /etc/rndc.key: [ OK ]  
Starting named: [ OK ]  
[root@node-rac2 Desktop]# chkconfig named on  
[root@node-rac1 named]# vim /etc/resolv.conf  
search xmj.com  
nameserver 192.168.200.120  
nameserver 192.168.200.122
```

关闭 NTP 服务【双节点】

```
[root@node-rac1 ~]# service ntpd stop  
[root@node-rac1 ~]# chkconfig ntpd off  
[root@node-rac1 ~]# mv /etc/ntp.conf /etc/ntp.conf.bak
```

创建用户和组【双节点】

```
groupadd -g 1100 oinstall  
groupadd -g 1200 dba  
groupadd -g 1300 oper  
groupadd -g 2100 asmadmin  
groupadd -g 2200 asmdba  
groupadd -g 2300 asmoper  
useradd -u 777 -g oinstall -G dba,oper,asmadmin,asmdba -d /home/oracle -s /bin/bash  
-c "Oracle Software Owner" oracle  
echo "oracle" | passwd --stdin oracle  
useradd -u 888 -g oinstall -G dba,asmadmin,asmdba,asmoper -d /home/grid -s  
/bin/bash -c "grid Infrastructure Owner" grid  
echo "grid" | passwd --stdin grid
```

创建目录并赋予权限【双节点】

```
mkdir -p /u01/app/grid  
mkdir -p /u01/app/12.1.0/grid  
mkdir -p /u01/app/oracle  
chown -R oracle:oinstall /u01  
chown -R grid:oinstall /u01/app/grid  
chown -R grid:oinstall /u01/app/12.1.0  
chmod -R 775 /u01
```

```
mkdir -p /u01/app/oracle  
chown oracle:oinstall /u01/app/oracle  
chmod -R 775 /u01
```

配置读写文件权限【双节点】

```
[root@node-rac1 ~]# vim /etc/security/limits.conf  
oracle      soft    nproc    2047  
oracle      hard    nproc    16384  
oracle      soft    nofile   1024  
oracle      hard    nofile   65536  
grid        soft    nproc    2047  
grid        hard    nproc    16384  
grid        soft    nofile   1024  
grid        hard    nofile   65536  
[root@node-rac1 ~]# vim /etc/security/limits.conf  
session required /lib/security/pam_limits.so  
session required pam_limits.so  
[root@node1 ~]# vim /etc/profile  
if [ $USER = "oracle" ]||[ $USER = "grid" ]; then  
    if [ $SHELL = "/bin/ksh" ]; then  
        ulimit -p 16384  
        ulimit -n 65536  
    else  
        ulimit -u 16384 -n 65536  
    fi  
fi
```

配置环境变量（双节点）

【oracle】

```
[root@node-rac1 ~]# su oracle  
[oracle@node-rac1 ~]$ vim .bash_profile  
export PATH  
export TMP=/tmp  
export TMPDIR=$TMP  
export ORACLE_HOSTNAME=node1.cty.com          (node2上改为 node2.cty.com)  
export ORACLE_SID=racdb1                      (node2上改为 racdb2)  
export ORACLE_BASE=/u01/app/oracle  
export ORACLE_HOME=$ORACLE_BASE/product/12.1.0/db_1  
export ORACLE_UNQNAME=racdb                  (node1&node2一定要保持一致)  
export TNS_ADMIN=$ORACLE_HOME/network/admin  
export ORACLE_TERM=xterm  
export PATH=/usr/sbin:$PATH  
export PATH=$ORACLE_HOME/bin:$PATH  
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib
```

```
export CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORACLE_HOME/rdbms/jlib
export EDITOR=vi
export LANG=en_US
export NLS_LANG=AMERICAN_AMERICA.AL32UTF8
export NLS_DATE_FORMAT='yyyy/mm/dd hh24:mi:ss'
umask 022
【grid】
export PATH
export TMP=/tmp
export TMPDIR=$TMP
export ORACLE_SID=+ASM1                                (node2 上改为 +ASM2)
export ORACLE_BASE=/u01/app/grid
export ORACLE_HOME=/u01/app/12.1.0/grid
export ORACLE_TERM=xterm
export NLS_DATE_FORMAT='yyyy/mm/dd hh24:mi:ss'
export TNS_ADMIN=$ORACLE_HOME/network/admin
export PATH=/usr/sbin:$PATH
export PATH=$ORACLE_HOME/bin:$PATH
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib
export CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORACLE_HOME/rdbms/jlib
export EDITOR=vi
export LANG=en_US
export NLS_LANG=AMERICAN_AMERICA.AL32UTF8
umask 022
```

配置系统内核参数【双节点】

```
net.ipv4.ip_forward = 0
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
kernel.sysrq = 0
kernel.core_uses_pid = 1
net.ipv4.tcp_syncookies = 1
kernel.msgmnb = 65536
kernel.msgmax = 65536
kernel.shmmmax = 4294967295
kernel.shmall = 268435456
kernel.shmmni = 4096
fs.aio-max-nr = 1048576
fs.file-max = 6815744
kernel.sem = 250 32000 100 128
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
```

```
net.core.wmem_max = 1048576
net.ipv4.tcp_wmem = 262144 262144 262144
net.ipv4.tcp_rmem = 4194304 4194304 4194304
```

配置 SSH 双节点信任【双节点】

【oracle】

```
[oracle@node-rac1 ~]$ mkdir ~/.ssh/
[oracle@node-rac1 ~]$ chmod 700 ~/.ssh/
[oracle@node-rac1 ~]$ cd ~/.ssh/
[oracle@node-rac1 .ssh]$ ssh-keygen -t rsa
[oracle@node-rac1 .ssh]$ ssh      node-rac2      cat
/home/oracle/.ssh/id_rsa.pub>>authorized_keys
[oracle@node-rac1 .ssh]$ ssh      node-rac1      cat
/home/oracle/.ssh/id_rsa.pub>>authorized_keys
[oracle@node-rac1 .ssh]$ chmod 644 authorized_keys
[oracle@node-rac1 .ssh]$ scp authorized_keys node-rac2:/home/oracle/.ssh/
```

【grid】

```
[grid@node-rac1 ~]$ mkdir ~/.ssh/
[grid@node-rac1 ~]$ chmod 700 ~/.ssh/
[grid@node-rac1 ~]$ cd ~/.ssh/
[grid@node-rac1 .ssh]$ ssh-keygen -t rsa
[oracle@node-rac1 .ssh]$ ssh      node-rac2      cat
/home/oracle/.ssh/id_rsa.pub>>authorized_keys
[oracle@node-rac1 .ssh]$ ssh      node-rac1      cat
/home/oracle/.ssh/id_rsa.pub>>authorized_keys
[oracle@node-rac1 .ssh]$ chmod 644 authorized_keys
[oracle@node-rac1 .ssh]$ scp authorized_keys node-rac2:/home/oracle/.ssh/
```

配置存储多路径 (双节点)

第一种方式:

```
[root@node-rac1 Packages]# /etc/init.d/multipathd start
[root@node-rac1 Packages]# yum -y install device-mapper-multipath
[root@node-rac1 Packages]# modprobe dm-multipath
[root@node-rac1 Packages]# chkconfig multipathd on
[root@node-rac1 Packages]# mpathconf --enable --find_multipaths y
--with_multipathd y --with_chkconfig y
[root@node-rac1 Packages]# /etc/init.d/multipathd restart
[root@node-rac1 ~]# oracleasm configure -i
Configuring the Oracle ASM library driver.
```

This will configure the on-boot properties of the Oracle ASM library driver. The following questions will determine whether the driver is loaded on boot and what permissions it will have. The current values will be shown in brackets ('[]'). Hitting <ENTER> without typing an

answer will keep that current value. Ctrl-C will abort.

```
Default user to own the driver interface []: grid
Default group to own the driver interface []: asmadmin
Start Oracle ASM library driver on boot (y/n) [n]: y
Scan for Oracle ASM disks on boot (y/n) [y]: y
Writing Oracle ASM library driver configuration: done
[root@node-rac1 ~]# oracleasm init
```

第二种方式：

```
# for i in `cat /proc/partitions | awk {'print $4'} |grep sd`; do echo "## $i:
`scsi_id --whitelist /dev/$i`"; done
59.# multipath.conf written by anaconda
60.
61.defaults {
62.user_friendly_names yes
63.}
64.blacklist {
65.devnode "^(\ram\raw\loop\fd\md\dm\-\sr\sc\dst)[0-9]*"
66.devnode "\hd[a-z]"
67.devnode "\dc\ss\bl\k[0-9]*"
68.device {
69.vendor "DGC"
70.product "LUNZ"
71.}
72.device {
73.vendor "IBM"
74.product "S/390.*"
75.}
76.# don't count normal SATA devices as multipaths
77.device {
78.vendor "ATA"
79.}
80.# don't count 3ware devices as multipaths
81.device {
82.vendor "3ware"
83.}
84.device {
85.vendor "AMCC"
86.}
87.# nor highpoint devices
88.device {
89.vendor "HPT"
90.}
91.wwid "20080930-1"
```

```
92.wwid "20080930-1"
93.device {
94.vendor Cisco
95.product Virtual_CD_DVD
96.}
97.wwid "*" //其实可以注释这项，这样就不需要单独填写 blacklist_exceptions
98.}
99.blacklist_exceptions { //排除在黑名单之外的 wwid
100.wwid "360060160a2212f00a67e0b91f2dbe111"
101.wwid "360060160a2212f0044a0fc6ef5eae111"
102.}
103.multipaths {
104.multipath {
105.uid 0 //磁盘读所属用户 uid
106.gid 0 //磁盘所属组 gid
107.wwid "360060160a2212f00a67e0b91f2dbe111" //wwid 号
108.mode 0600 //磁盘读写权限
109.}
110.multipath {
111.wwid "360060160a2212f0044a0fc6ef5eae111"
112.alias data //别名
113.}
114..... //还可以根据实际情况，配置其它磁盘的别名、uid、gid、mode etc...
115.
116.}
```

配置完了之后，重启 multipathd 服务，之后通过 multipath -ll 查看经过多路径软件绑定后的磁盘。注意，如果要对磁盘进行格式化，请采用 /dev/mapper/[alias] 这类设备名进行 fdisk。

	只	在	节	点
=====				=====
1=====				

```
[root@node-rac1 ~]# oracleasm createdisk OCR /dev/mapper/mpathap5
Writing disk header: done
Instantiating disk: done
[root@node-rac1 ~]# oracleasm createdisk OCR1 /dev/mapper/mpathap6
Writing disk header: done
Instantiating disk: done
[root@node-rac1 ~]# oracleasm createdisk FRA /dev/mapper/mpathap7
Writing disk header: done
Instantiating disk: done
[root@node-rac1 ~]# oracleasm createdisk FRA1 /dev/mapper/mpathap8
Writing disk header: done
Instantiating disk: done
[root@node-rac1 ~]# oracleasm createdisk DATA /dev/mapper/mpathap9
Writing disk header: done
```

```
Instantiating disk: done
[root@node-rac1 ~]# oracleasm createdisk DATA1 /dev/mapper/mpathap10
Writing disk header: done
Instantiating disk: done
[root@node-rac1 ~]# oracleasm createdisk DATA2 /dev/mapper/mpathap11
=====
只 在 节 点
2=====

[root@node-rac2 ~]# oracleasm scandisks
Cleaning any stale ASM disks...
Scanning system for ASM disks...
Instantiating disk "OCR"
Instantiating disk "OCR1"
Instantiating disk "FRA"
Instantiating disk "FRA1"
Instantiating disk "DATA"
Instantiating disk "DATA1"
Instantiating disk "DATA2"
=====
两 个 节 点
=====

[root@node-rac1 ~]# oracleasm listdisks
DATA
DATA1
DATA2
FRA
FRA1
OCR
OCR1
```

安装集群软件【双节点】

```
[root@node-rac1 ~]# mkdir /install/
[root@node-rac1 ~]# chown -R grid.oinstall /install/
[root@node-rac1 ~]# chmod 775 /install
[grid@node-rac1 grid]$ ./runcluvfy.sh stage -pre crsinst -no node-rac1,node-rac2
-fixup -verbose 【自检】
[root@node-rac1 Desktop]# sh /tmp/CVU_12.1.0.1.0_grid/runfixup.sh
Pre-check for cluster services setup was successful.
```

Oracle Cluster Registry (OCR) 用于解决集群中的健忘症的问题。

健忘是由于某个节点更新了 OCR 中的内容，而集群中的另外一些节点此时处于关闭，维护或重启阶段，OCR Master 进程来不及将其信息更新到这些异常节点缓存而导致的不一致。譬如，在 A 节点发出了添加 ocr 镜像的命令，在这个时候 B 节点处于重启阶段。重启后 A 已经更新完毕，而此时 B 并不知道已经为 ocr 增加了一个新的镜像磁盘，这时就会造成配置丢失，也就是所谓的“健忘症”。

VOTEDISK 是用于解决脑裂(Split Brain) 的问题。

在集群中，节点间通过某种机制（心跳）了解彼此的健康状态，以确保各节点协调工作。假设只有“心跳”出现问题，各个节点还在正常运行，这时，每个节点都认为其他的节点宕机了，自己是整个集群环境中的“唯一健在者”，自己应该获得整个集群的“控制权”。在集群环境中，存储设备都是共享的，这就意味着数据灾难，这种情况就是“脑裂”。

科普二：ASM 的三种冗余模式

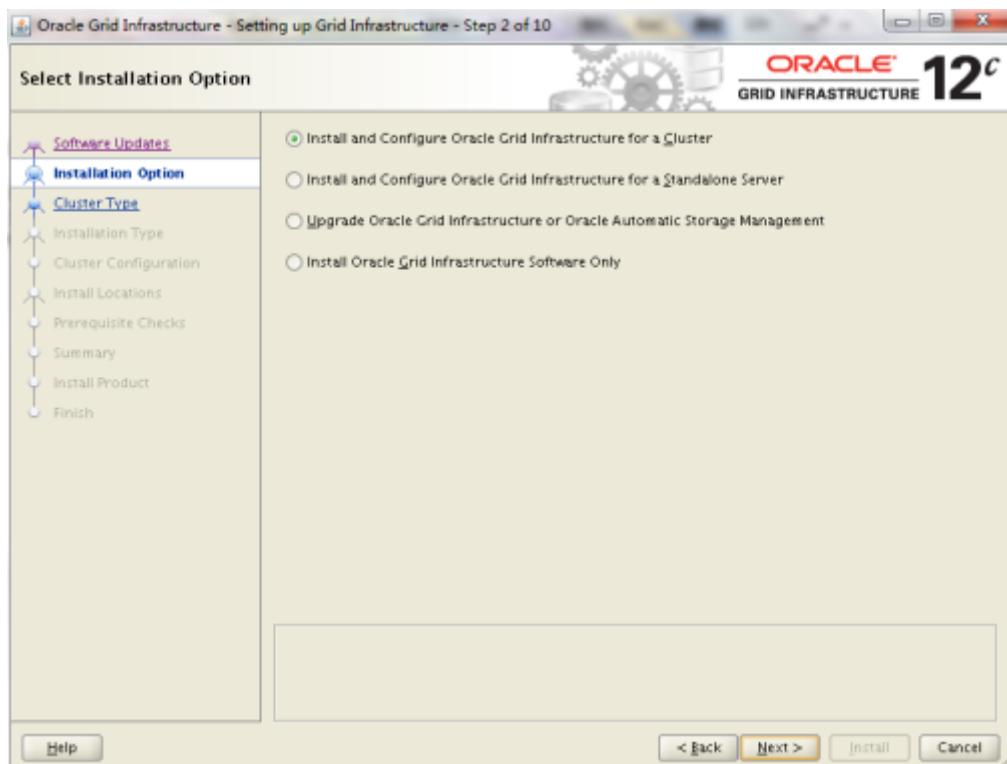
高冗余 (High): asm 使用 3 份镜像存储，以提高性能和数据的安全，最少需要三块磁盘（三个 failure group）；有效磁盘空间是所有磁盘设备 大小之和的 1/3，虽然冗余级别高了，但是硬件的代价也最高。

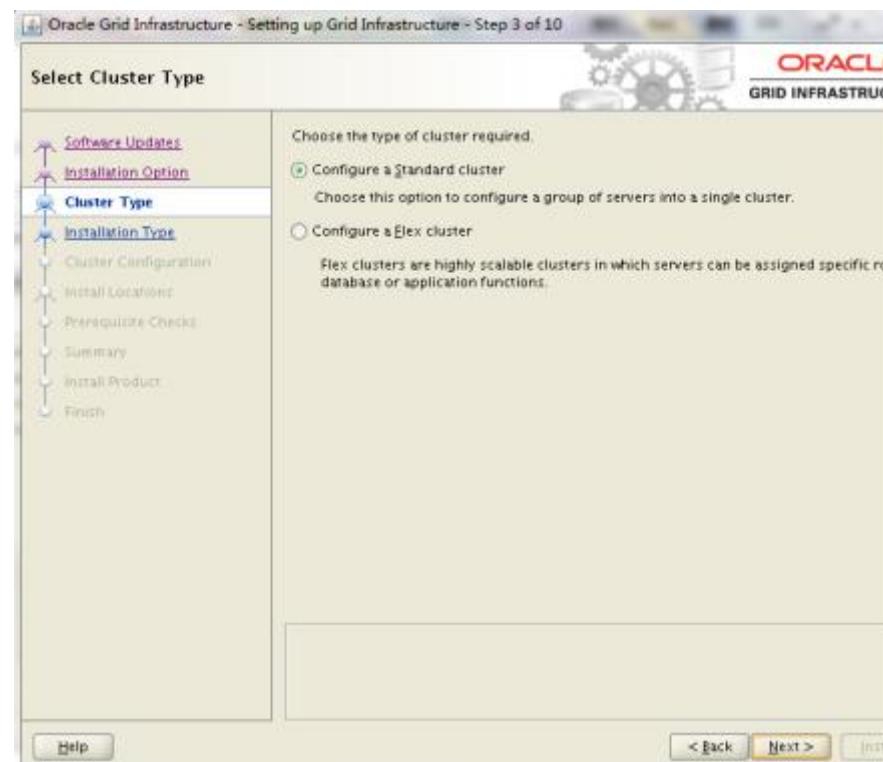
常规冗余 (Normal): asm 使用 2 份镜像存储，以提高性能和数据的安全，最少需要两块磁盘（两个 failure group）；有效磁盘空间是所有磁盘设备 大小之和的 1/2，一般用常规冗余就 ok。

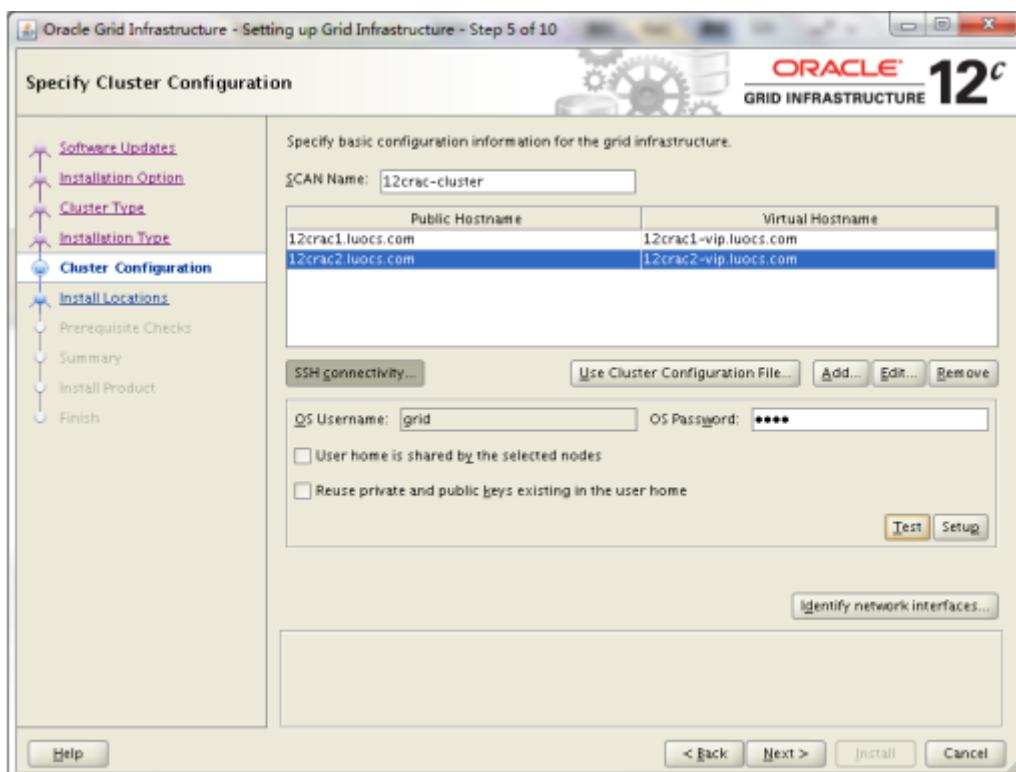
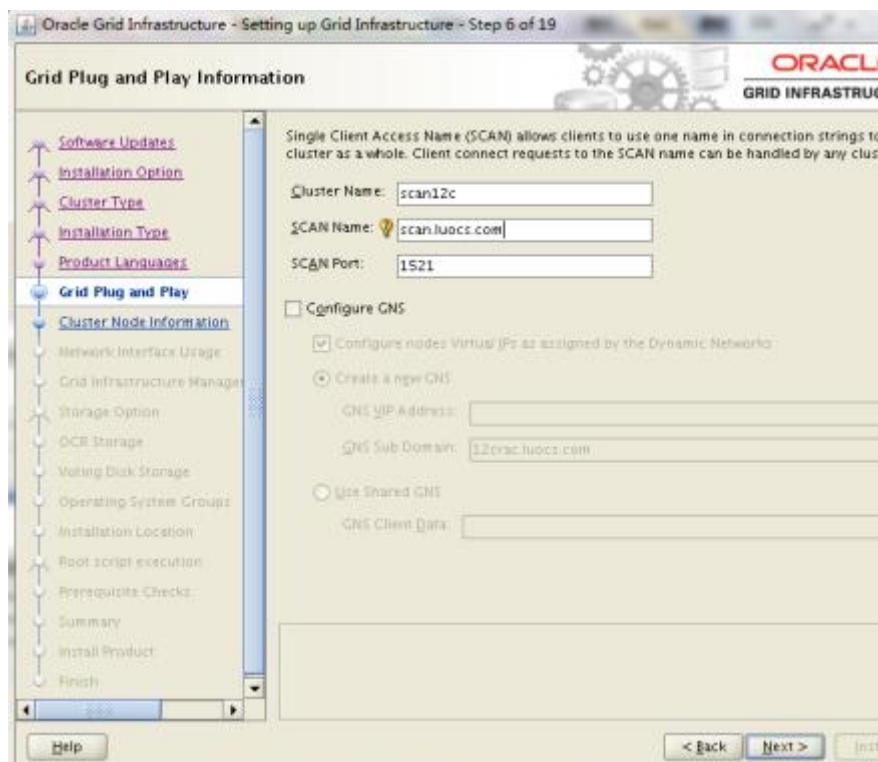
外部冗余 (External): asm 不对数据库文件镜像，可以通过 raid 磁盘镜像；所用磁盘最少，有效磁盘空间是所有磁盘设备的大小之和。

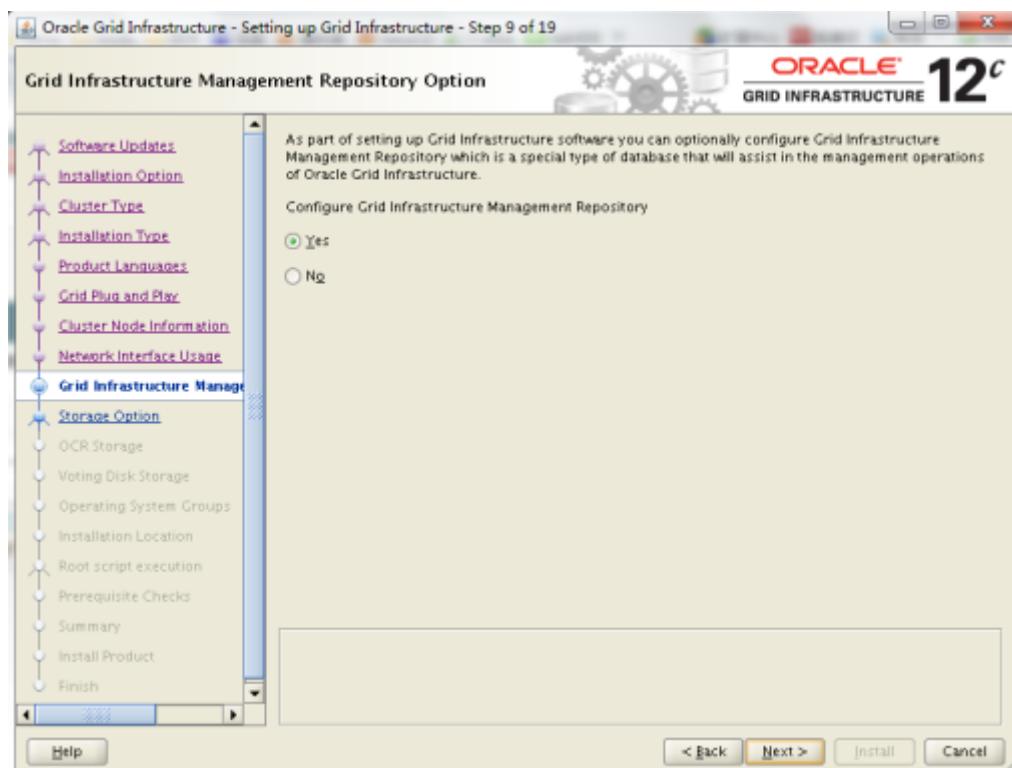
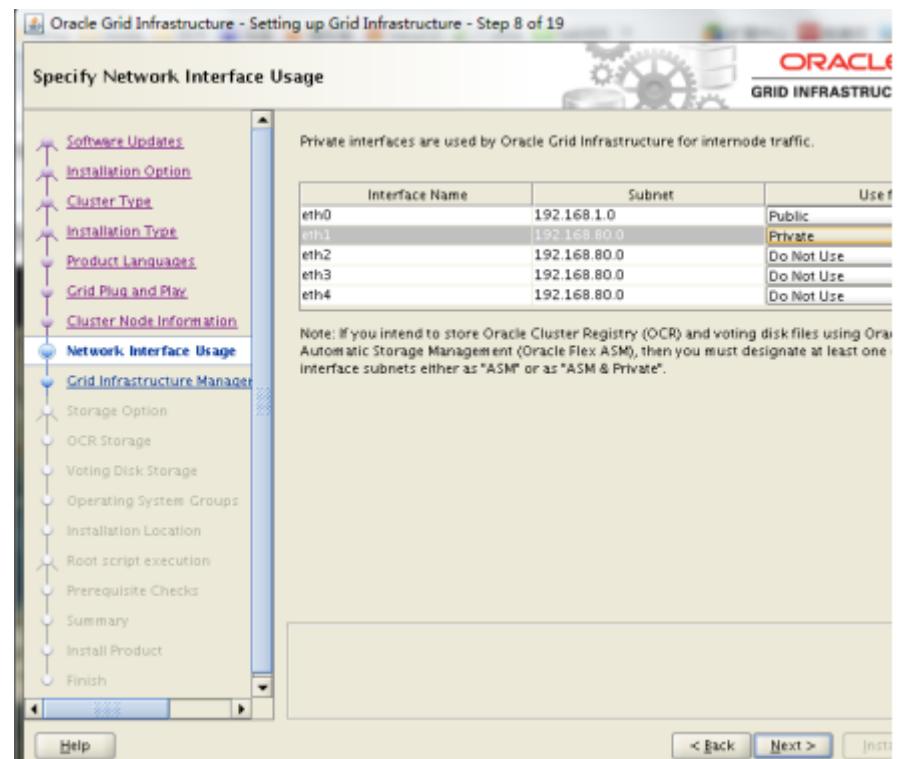
补充：1、如果是用来放置 ocr/voting disk 的 diskgroup，那么 external, normal, high 分别至少需要 1, 3, 5 个 disk。

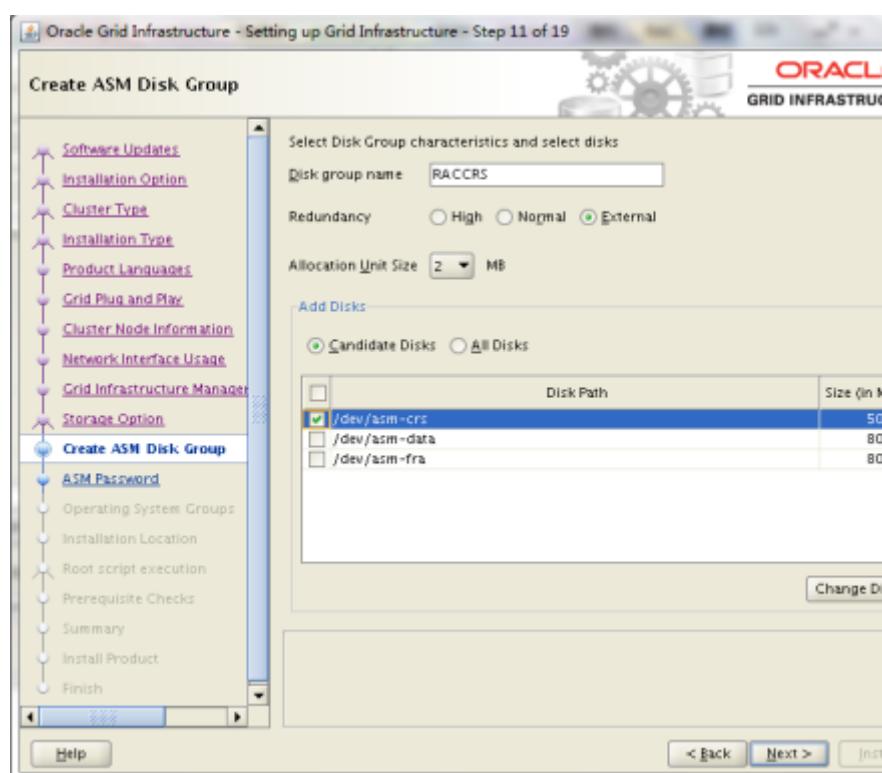
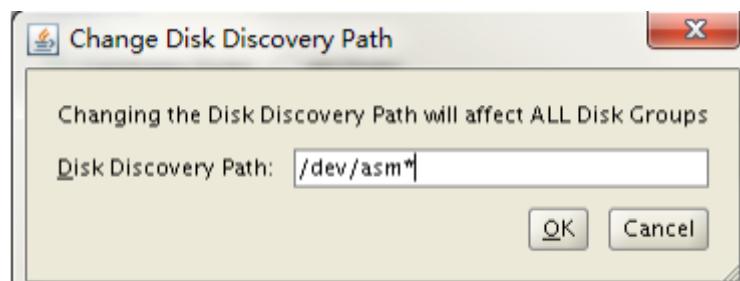
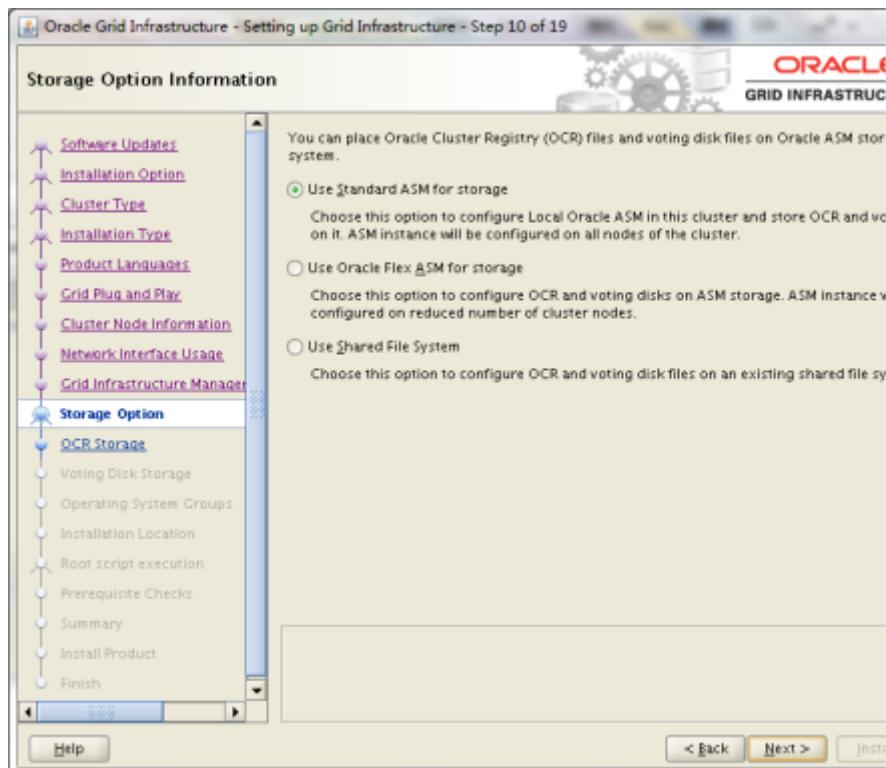
2、如果是用来放置 data file 的 diskgroup，那么 external, normal, high 分别至少需要 1, 2, 3 个 disk。

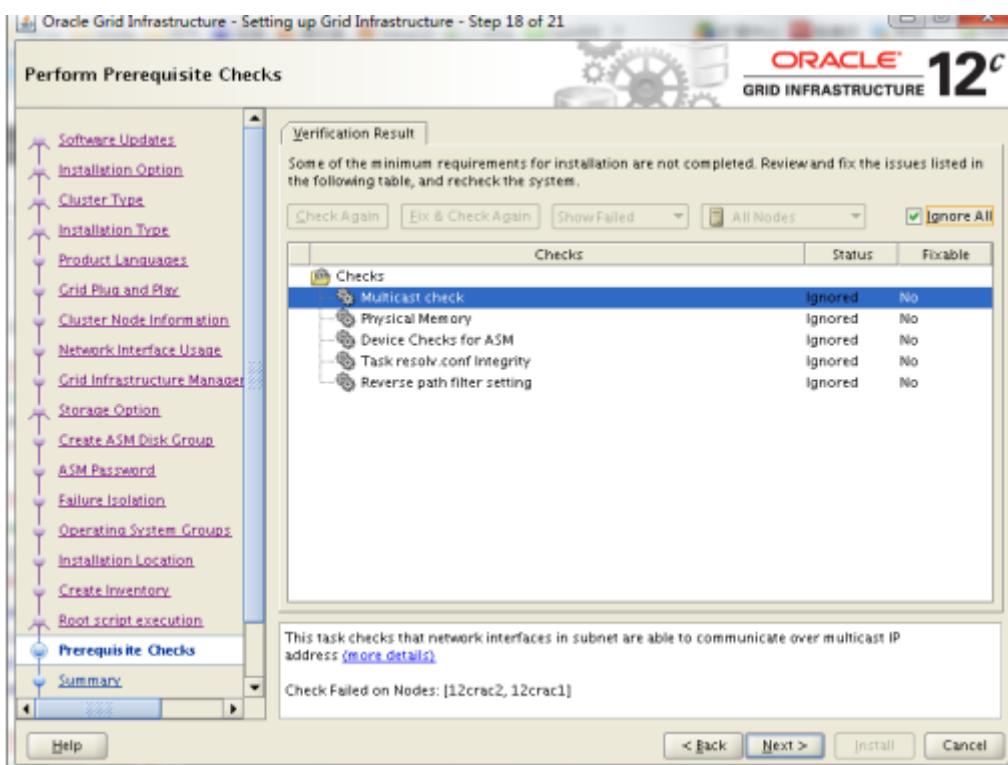
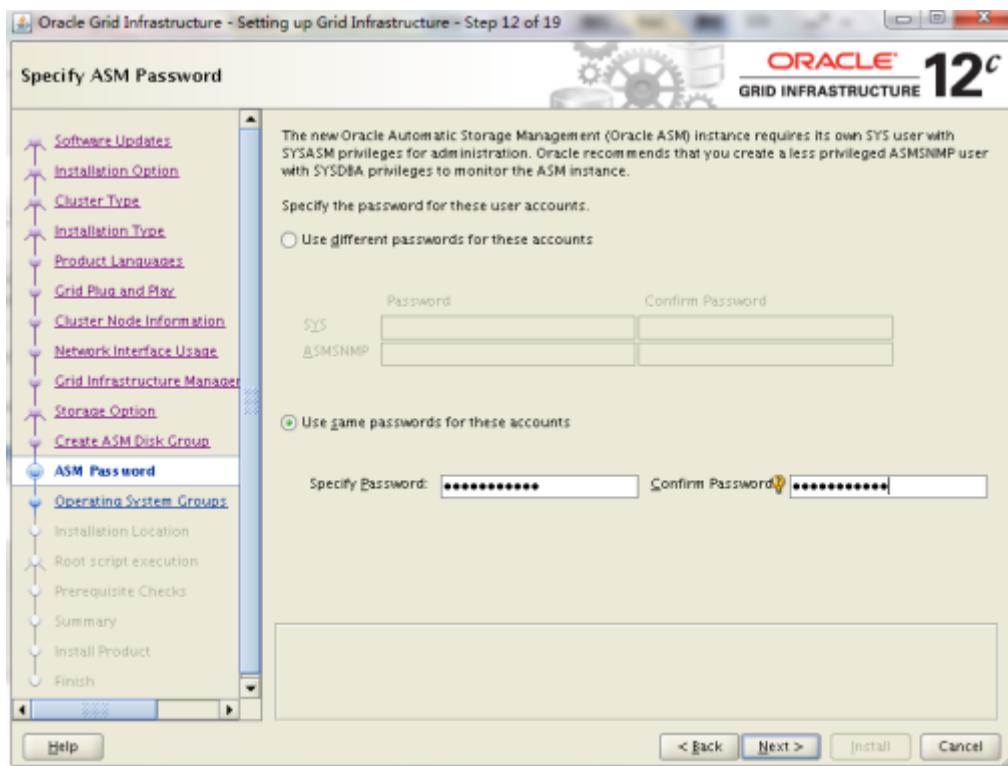


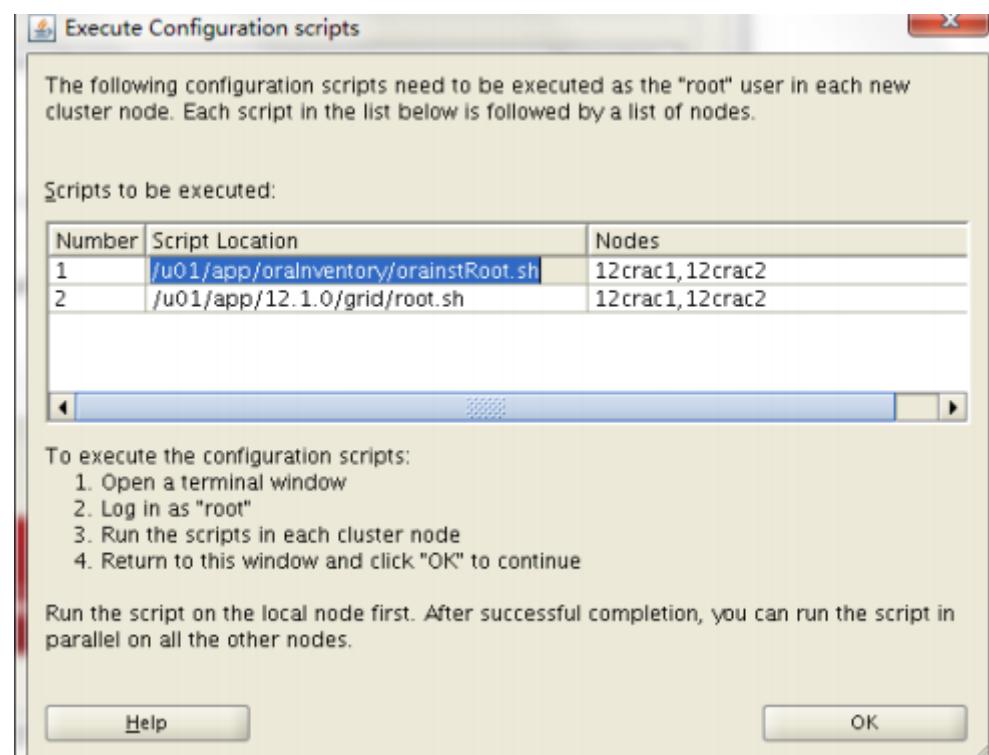










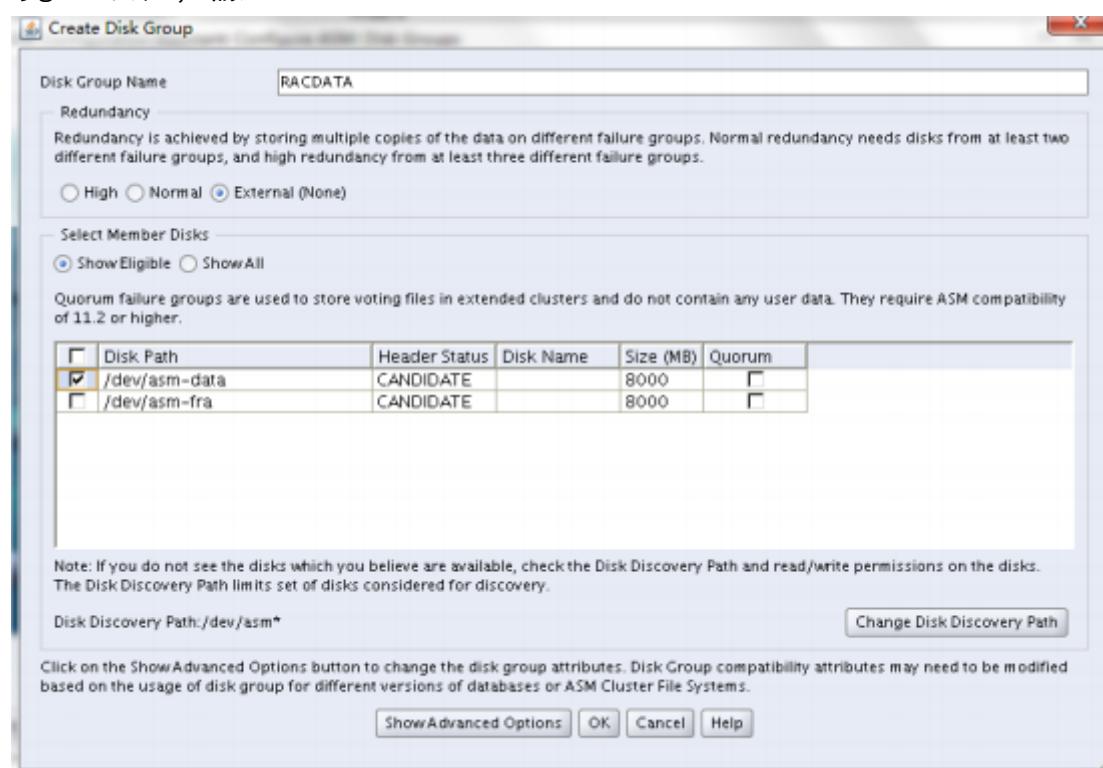


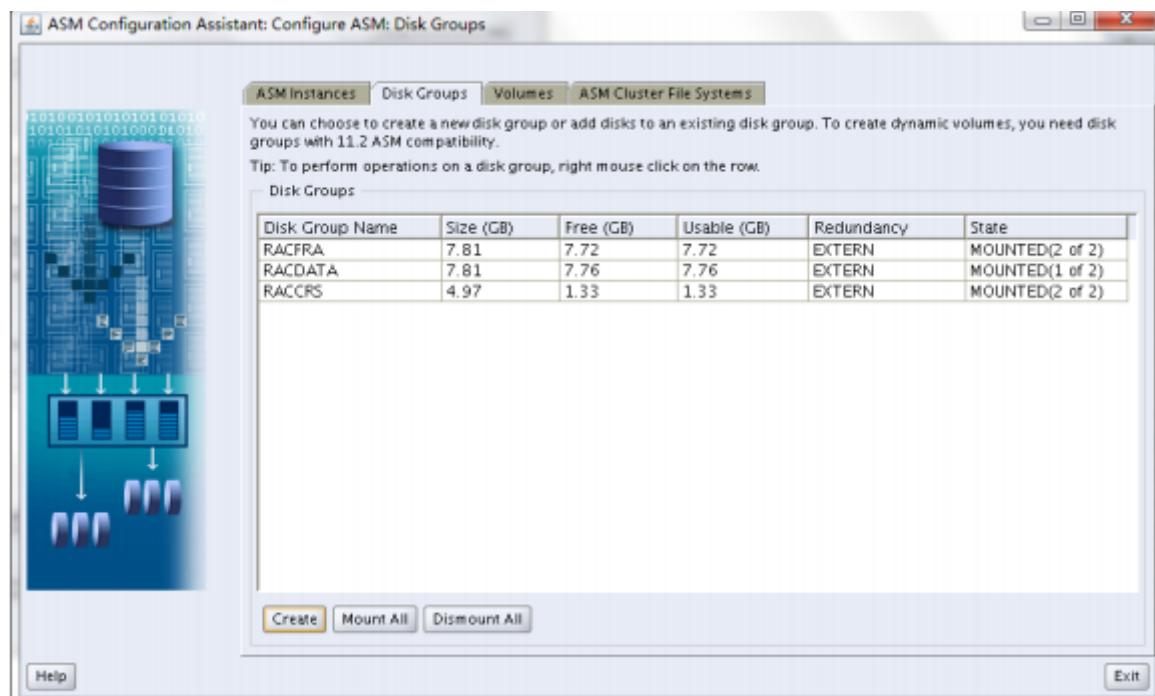
后分别执行 Root. sh 现在 node-rac1 上执行，后在 node-rac2 上执行

查看状态 crsctl statat res -t，查看状态 stable

创建 ASM 磁盘组

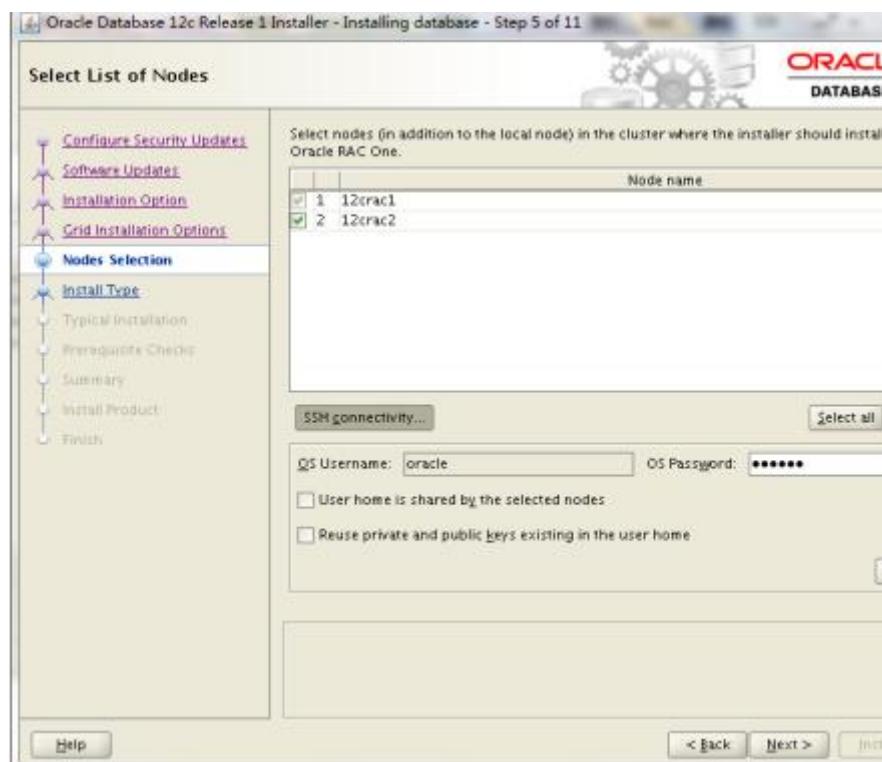
切换到 grid 用户，输入 asmca

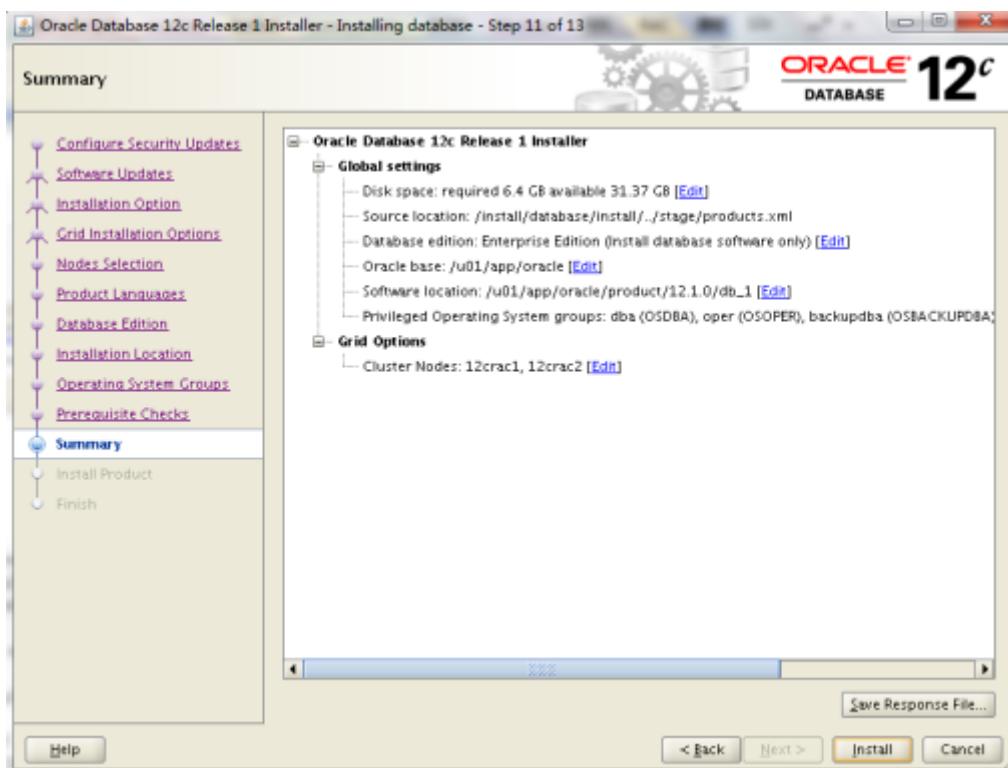


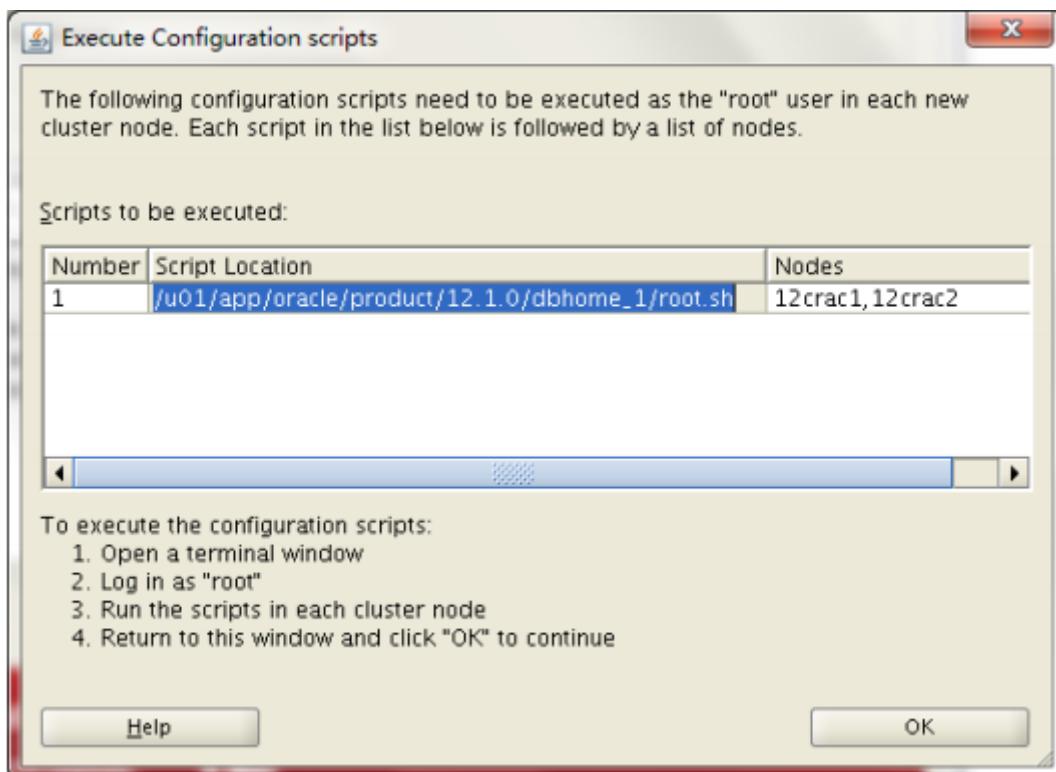


安装集群数据软件



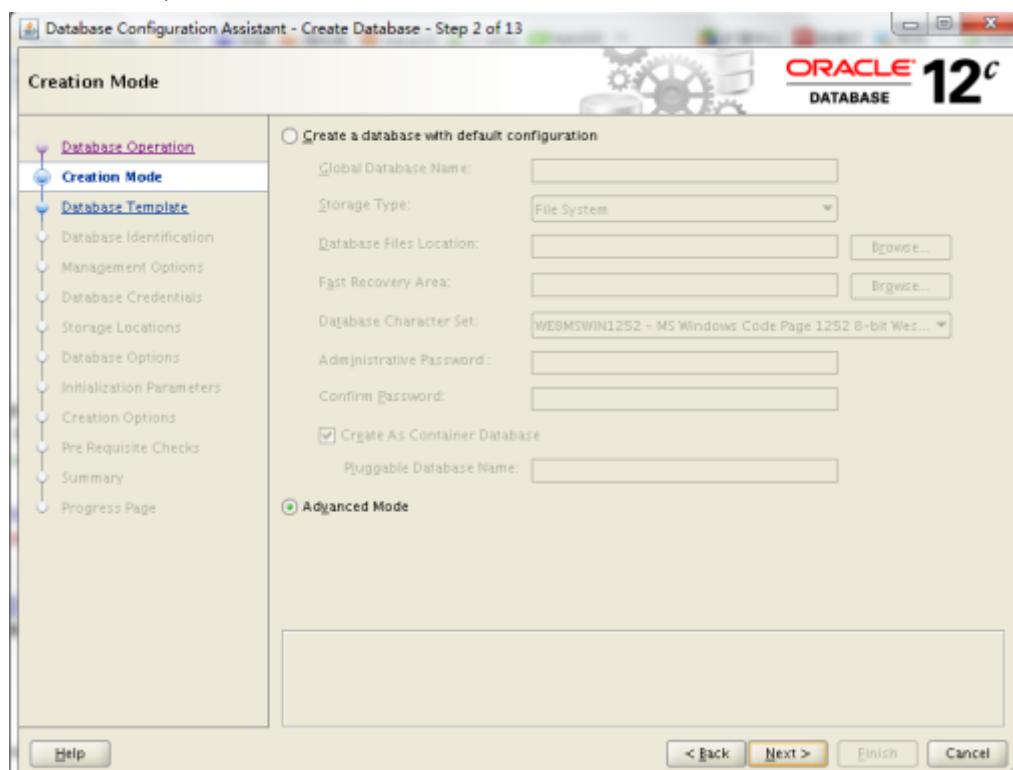


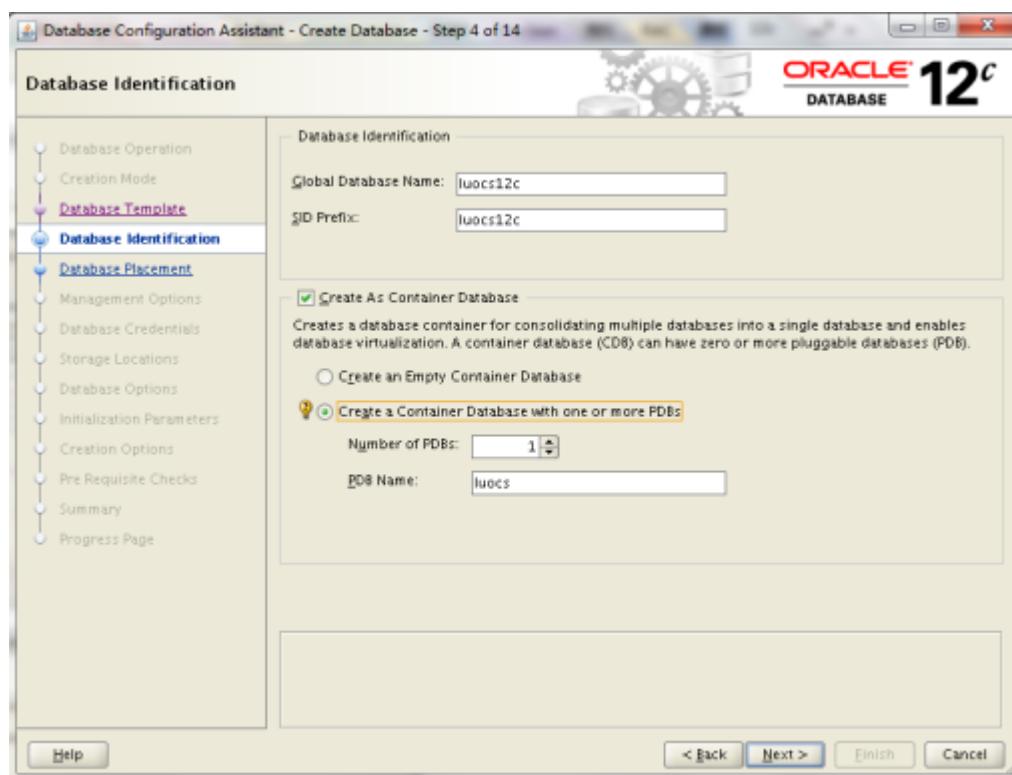
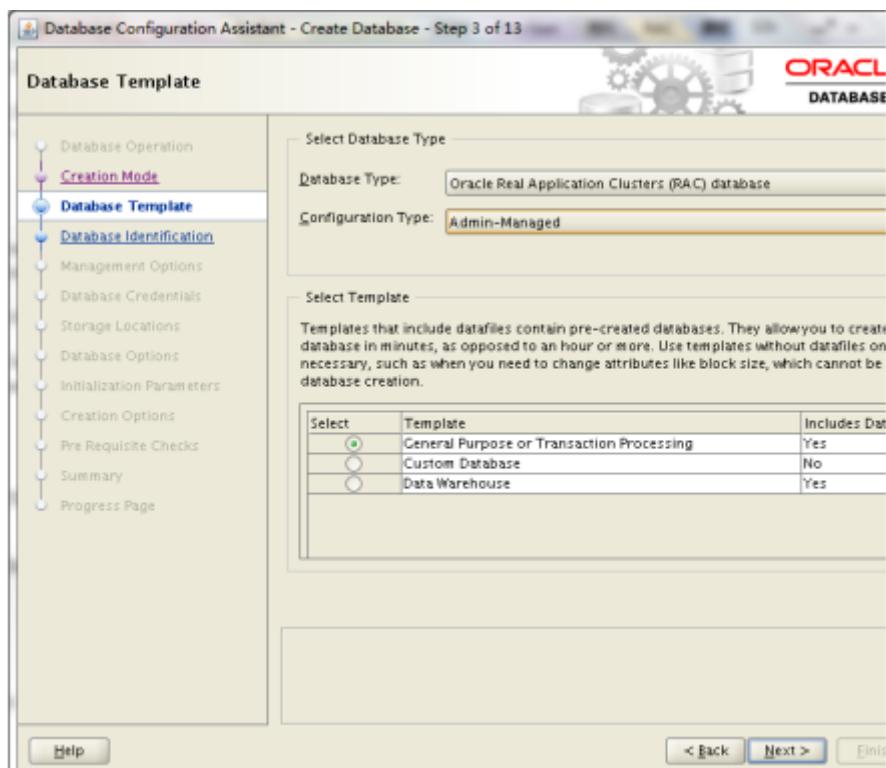


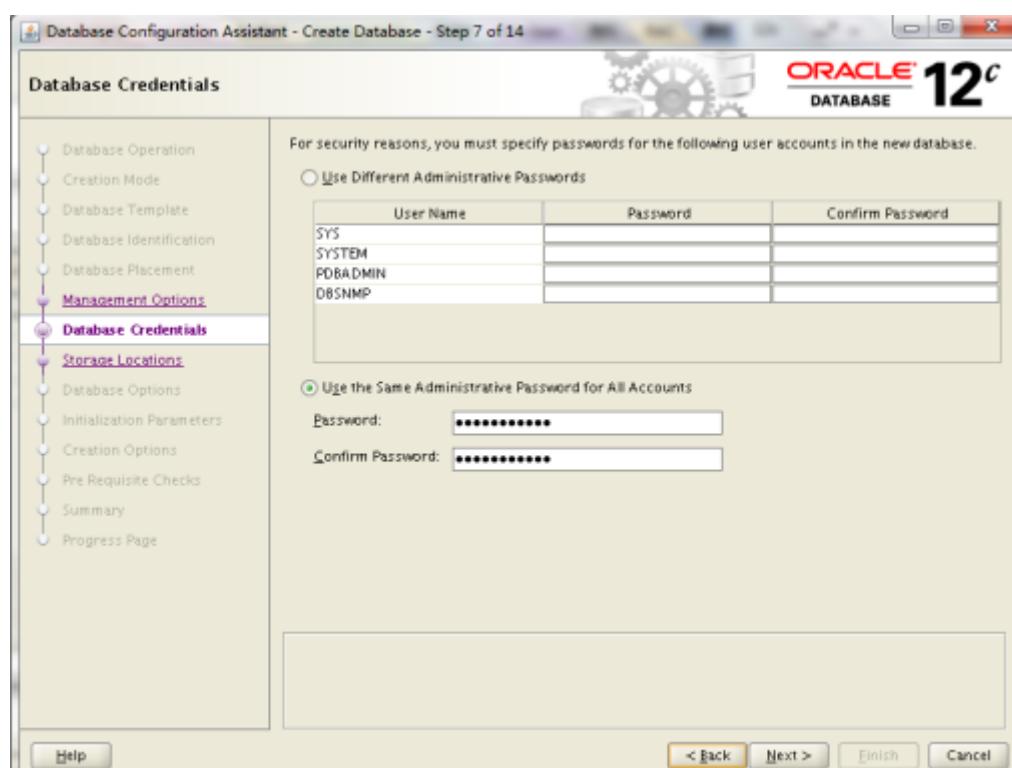
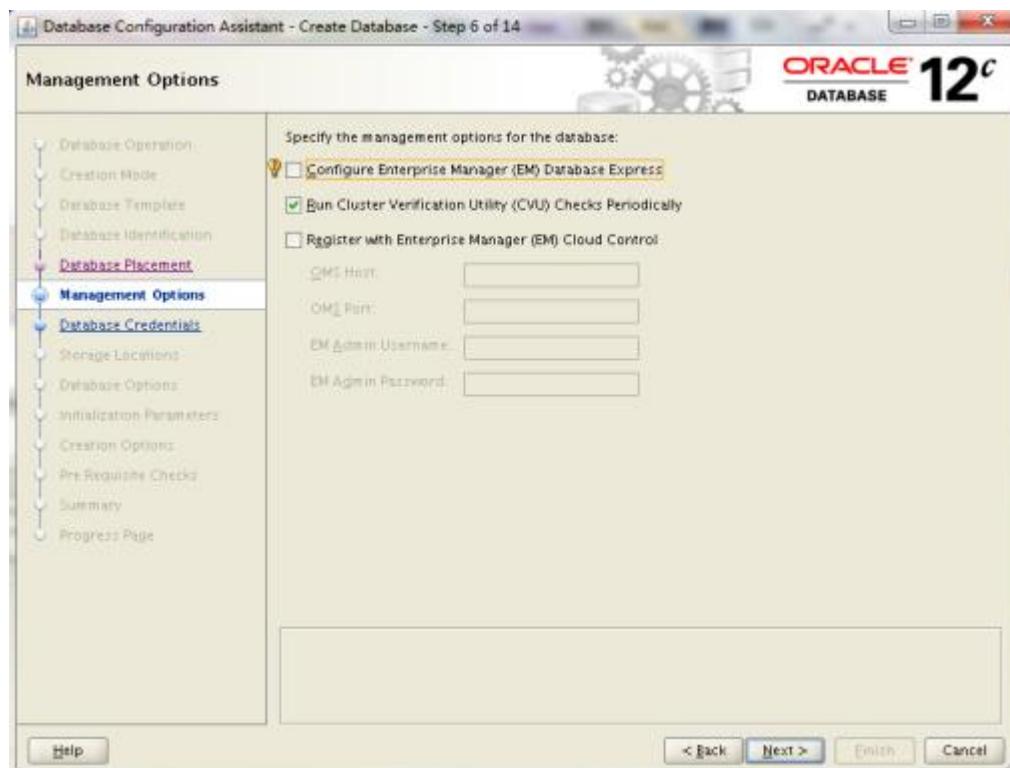


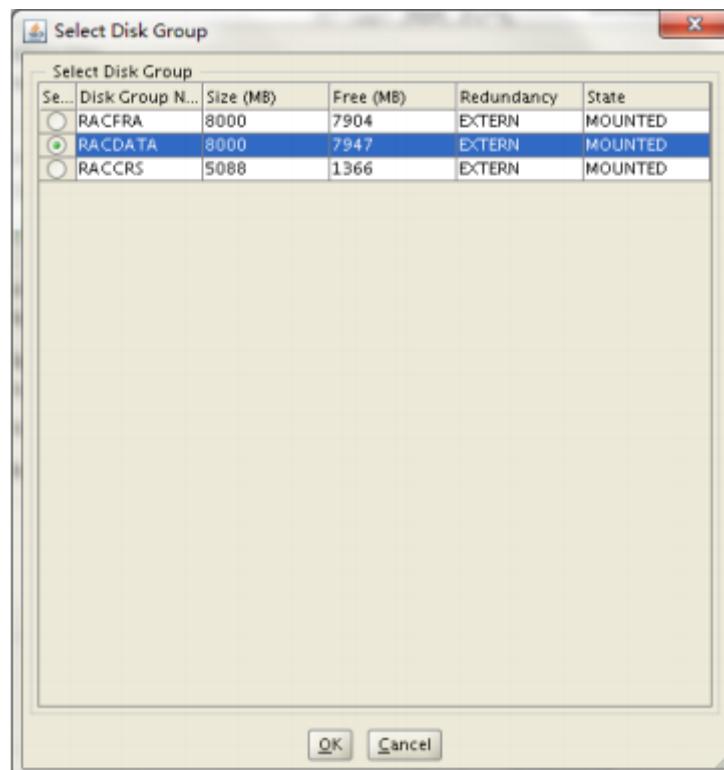
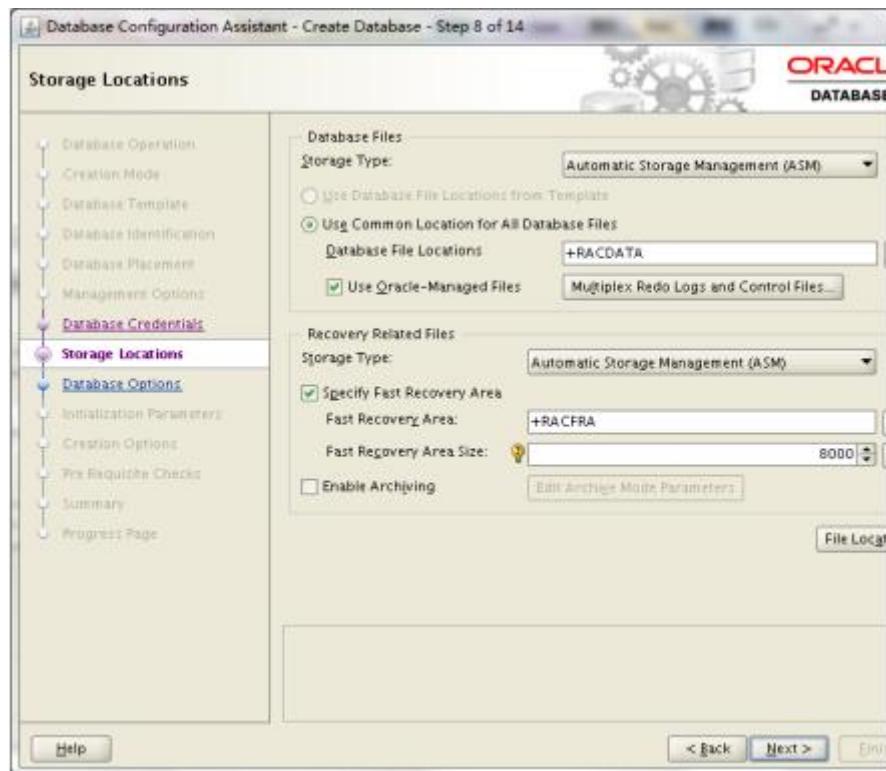
创建数据库

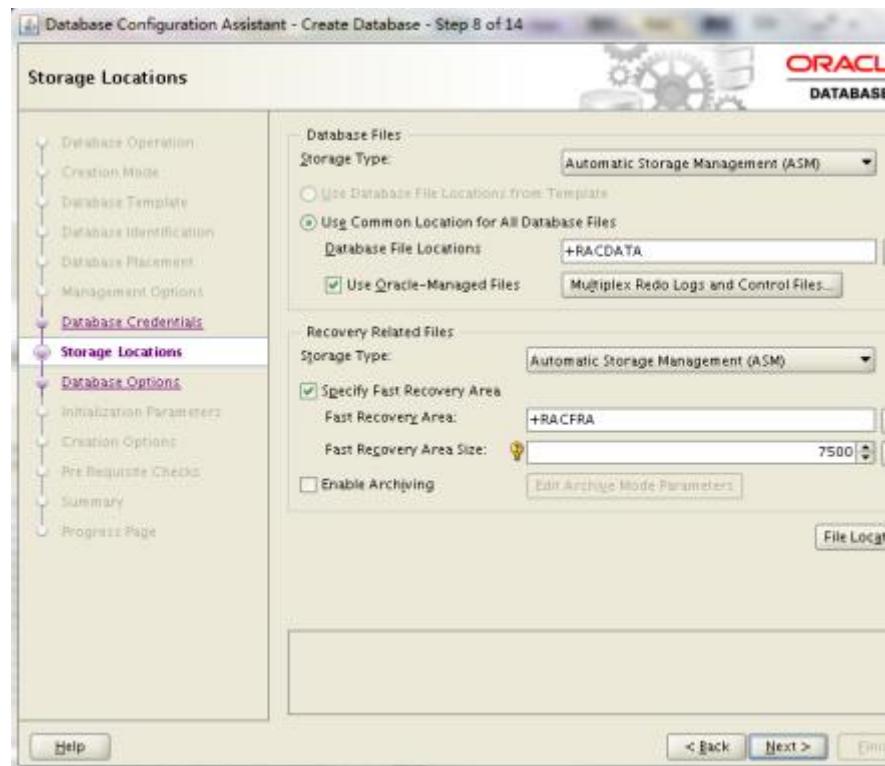
切换到 oracle 用户，输入 dbca

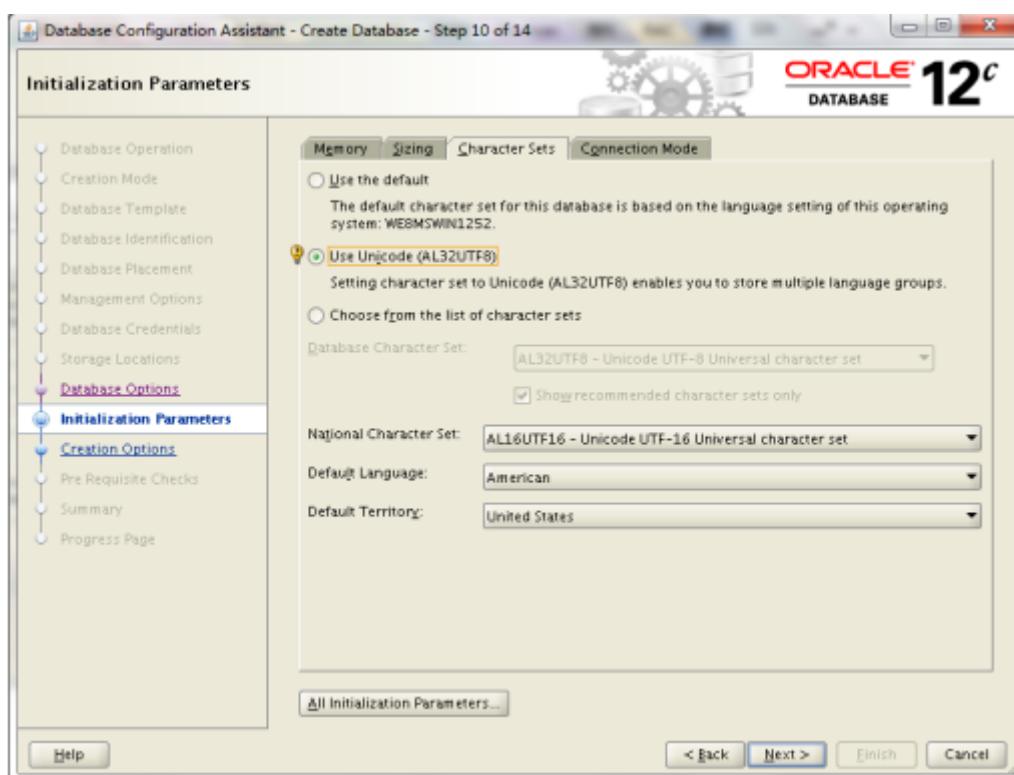
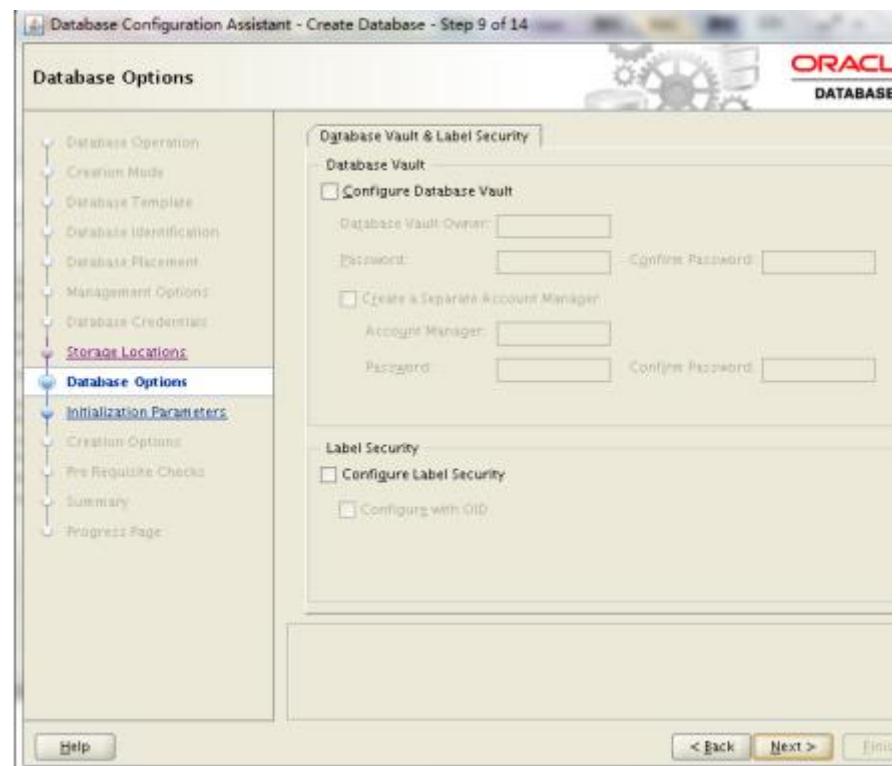


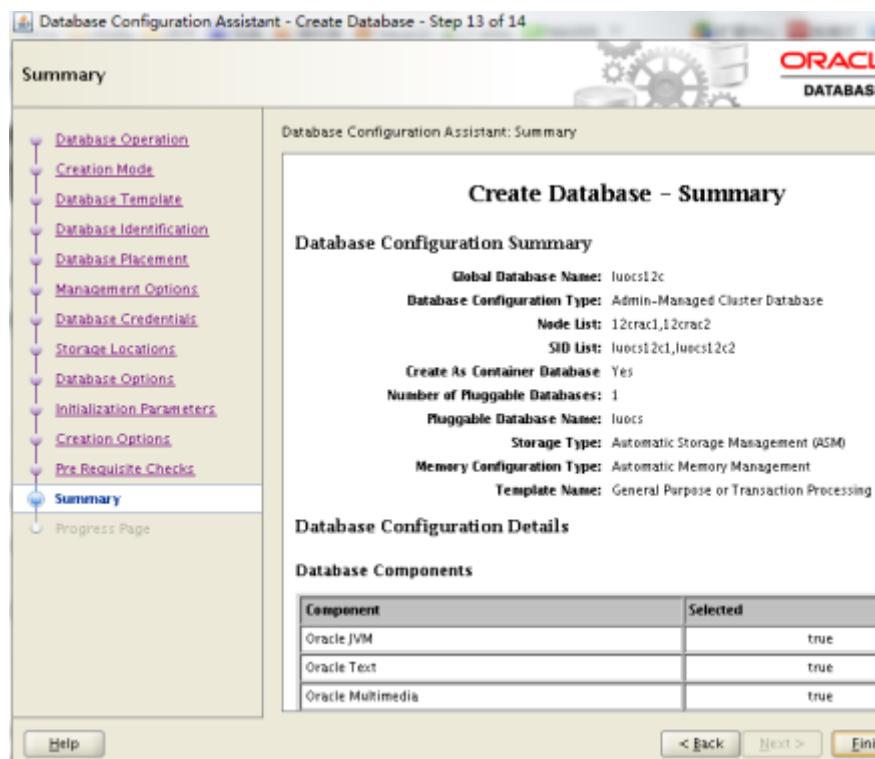
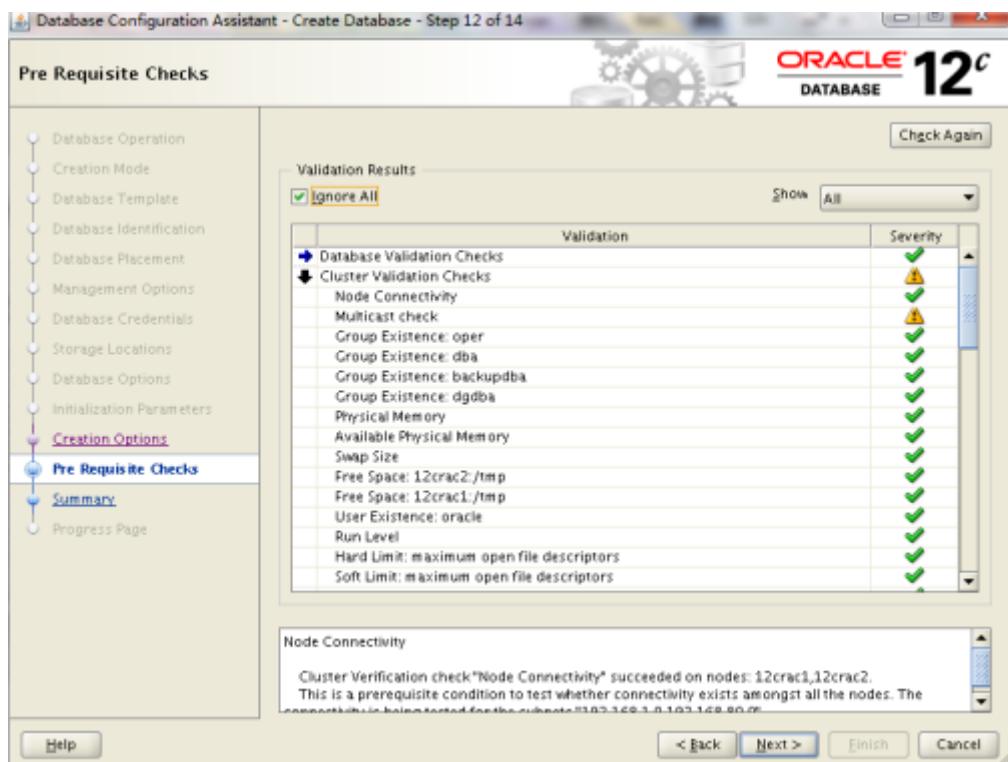


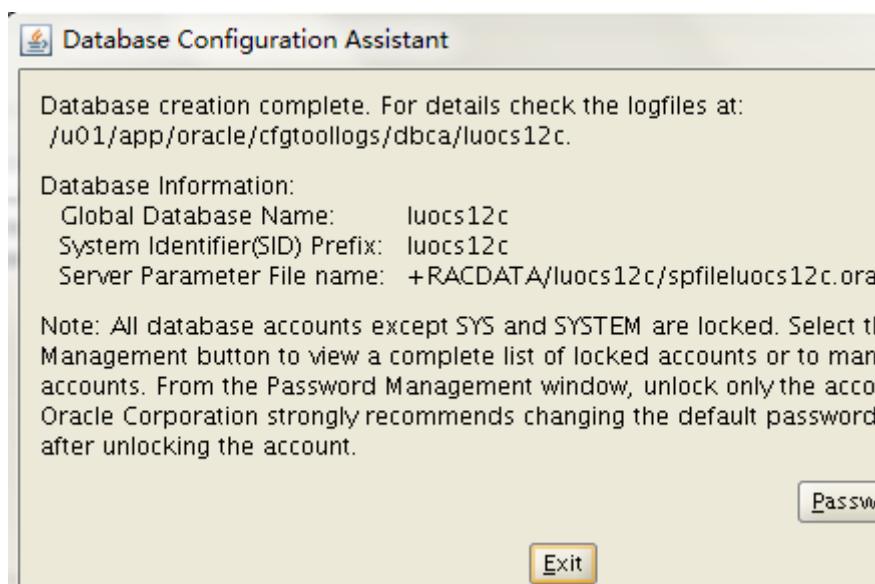
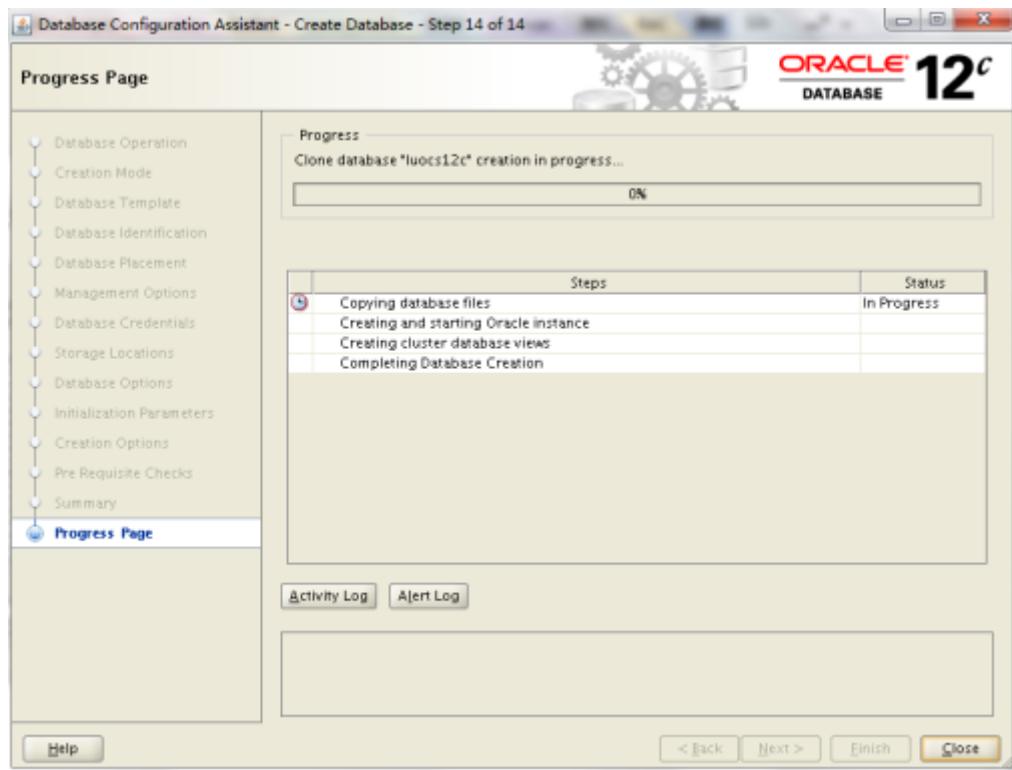


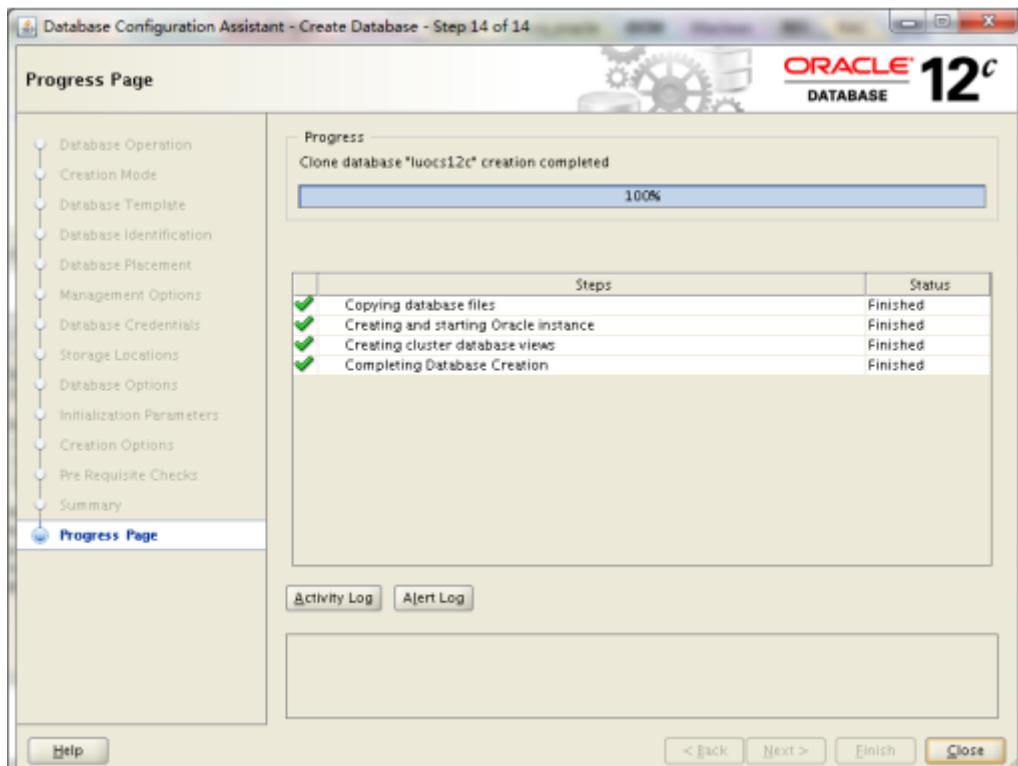












STABLE

ora.RACCRS.dg

ONLINE ONLINE

12crac1

STABLE

ONLINE ONLINE

12crac2

STABLE

ora.RACDATA.dg

ONLINE ONLINE

12crac1

STABLE

ONLINE ONLINE

12crac2

STABLE

ora.RACFRA.dg

ONLINE ONLINE

12crac1

STABLE

ONLINE ONLINE

12crac2

STABLE

ora.asm

ONLINE ONLINE

12crac1

Started, STABLE

ONLINE ONLINE

12crac2

Started, STABLE

ora.net1.network

ONLINE ONLINE

12crac1

STABLE

ONLINE ONLINE

12crac2

STABLE

ora. ons
STABLE
ONLINE ONLINE 12crac1

STABLE
ONLINE ONLINE 12crac2

--

ClusterResources

--

ora. 12crac1.vip
1
STABLE
ONLINE ONLINE 12crac1

ora. 12crac2.vip
1
STABLE
ONLINE ONLINE 12crac2

ora. LISTENER_SCAN1. lsnr
1
STABLE
ONLINE ONLINE 12crac2

ora. LISTENER_SCAN2. lsnr
1
STABLE
ONLINE ONLINE 12crac1

ora. LISTENER_SCAN3. lsnr
1
STABLE
ONLINE ONLINE 12crac1

ora. MGMTLSNR
1
169. 254. 88. 173192. 1

68. 80. 150, STABLE

ora. cvu
1
STABLE
ONLINE ONLINE 12crac1

ora. luocs12c.db
1
Open, STABLE
2
STABLE
ONLINE ONLINE 12crac2

ora. mgmtdb
1
Open, STABLE
1
STABLE
ONLINE ONLINE 12crac1

ora. oc4j

1

ONLINE ONLINE

12cracl

STABLE

ora.scan1.vip

RAC 数据库配置信息

```
[grid@12cracl ~]$ srvctl config database -d luocs12c
Database unique name: luocs12c
Database name: luocs12c
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +RACDATA/luocs12c/spfileluocs12c.ora
Password file: +RACDATA/luocs12c/orapwluocs12c
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools: luocs12c
Database instances: luocs12c1,luocs12c2
Disk Groups: RACFRA,RACDATA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
Database is administrator managed
```

```
[grid@12cracl ~]$ srvctl status database -d luocs12c
Instance luocs12c1 is running on node 12cracl
Instance luocs12c2 is running on node 12cracl2
```

```
[grid@12cracl ~]$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s): 12cracl,12cracl2
实例状态:
sys@LUOCS12C> select instance_name, status from gv$instance;
```

删除重新配置

```
/u01/12.1.0/grid/crs/install/roothas.pl -deconfig -force -verbose
/oracle/grid/crs/install/roothas.pl -delete -force -verbose
```

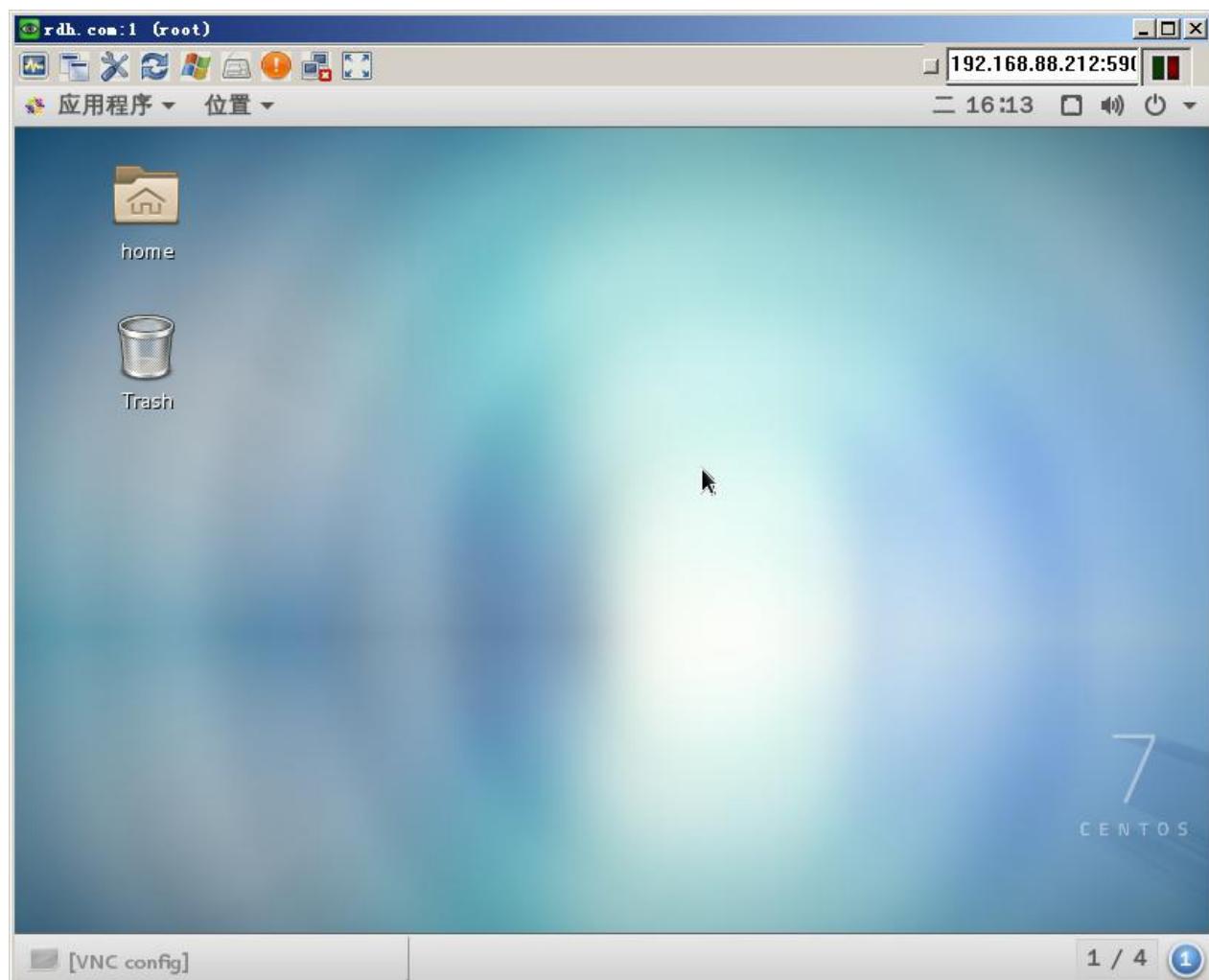
修改 IP 地址详细记录

<http://www.xuebuyuan.com/2045149.html>

十、VNC Server

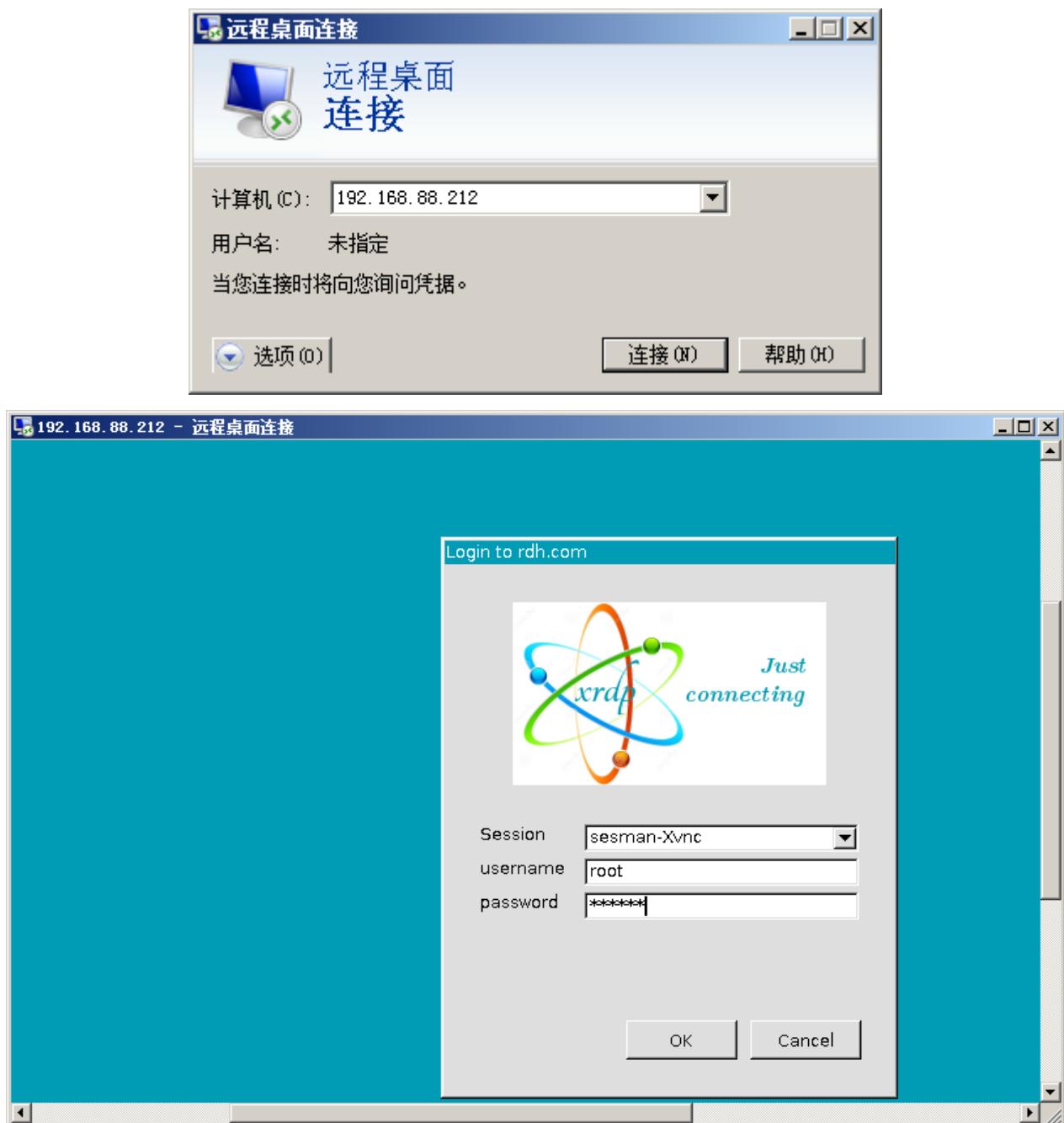
```
[root@rdh ~]# yum -y install tigervnc-server
[root@rdh ~]# vncpasswd
Password:
Verify:
```

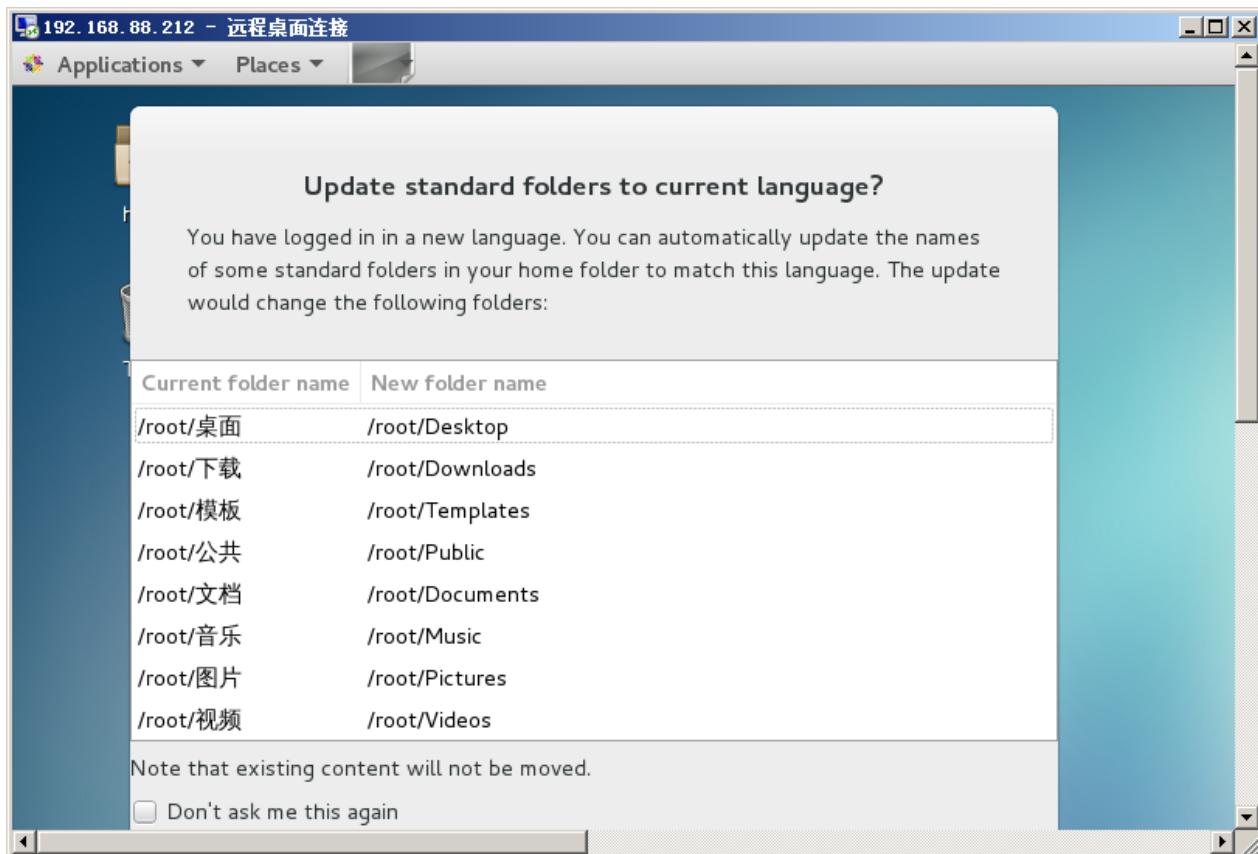
```
[root@rdh ~]# vncserver :1 -geometry 800x600 -depth 24
客户 端 下 载 地 址 1 :
http://www.uvnc.com/component/jdownloads/finish/4-setup/291-ultravnc-1210-x64-setup/0.html
客户端下载地址 2: http://dl-sh-ocn-3.pchome.net/3z/zk/UltraVNC\_1\_2\_04\_X64\_Setup.rar?tmp=1456214530384
```



十一、XRDП 远程桌面

```
[root@rdh ~]# yum -y install xrdp-0.6.1-4.el6.x86\_64.rpm
[root@rdh ~]# yum -y install xrdp
[root@rdh ~]# systemctl start xrdp
[root@rdh ~]# systemctl enable xrdp
```





十二、DRBD 分布式存储

```
[root@rdh ~]# yum -y install gcc make automake autoconf libxslt libxslt-devel flex rpm-build kernel-devel
[root@rdh ~]# wget http://oss.linbit.com/drbd/drbd-utils-latest.tar.gz \
> http://oss.linbit.com/drbd/8.4/drbd-8.4.7-1.tar.gz
[root@rdh ~]# tar -zvxf drbd-8.4.7-1.tar.gz
[root@rdh ~]# cd drbd-8.4.7-1/
[root@rdh drbd-8.4.7-1]# mkdir -p
rpmbuild/{BUILD,BUILDROOT,RPMS,SOURCES,SPECS,SRPMS}
[root@rdh ~]# mkdir -p rpmbuild/{BUILD,BUILDROOT,RPMS,SOURCES,SPECS,SRPMS}
[root@rdh drbd-8.4.7-1]# make km-rpm
[root@rdh ~]# tar -zvxf drbd-utils-latest.tar.gz
[root@rdh ~]# cd drbd-utils-8.9.6/
[root@rdh drbd-utils-8.9.6]# vim drbd.spec.in
%undefine with_sbin_symlinks
[root@rdh drbd-utils-8.9.6]# ./configure
[root@rdh drbd-utils-8.9.6]# make rpm
[root@rdh drbd-utils-8.9.6]# cd /root/rpmbuild/RPMS/x86_64
[root@rdh drbd-utils-8.9.6]# cd /root/rpmbuild/RPMS/x86_64
[root@rdh x86_64]# rpm -Uvh drbd-utils*.rpm drbd-km*.rpm
=====主从节点=====
[root@rdh ~]# vim /etc/drbd.d/global_common.conf
```

```
on-io-error detach;
[root@rdh ~]# vim /etc/drbd.d/r0.res
# DRBD device
# block device
disk /dev/sdb1;
meta-disk internal;
on rdh.com {
    # IP address:port
    address 192.168.88.212:7788;
}
on huatech.com {
    address 192.168.88.186:7788;
}
}.
```

```
[root@rdh ~]# modprobe drbd
[root@rdh ~]# lsmod|grep drbd
[root@rdh ~]# drbdadm create-md r0
[root@rdh ~]# systemctl start drbd
[root@rdh ~]# systemctl enable drbd
```

=====互为被动节点=====

```
[root@rdh ~]# cat /proc/drbd
version: 8.4.7-1 (api:1/proto:86-101)
GIT-hash: 3a6a769340ef93b1ba2792c6461250790795db49 build by root@rdh.com, 2016-02-24 08:49:03
0: cs:Connected ro:Secondary/Secondary ds:Inconsistent/Diskless C r-----
    ns:0 nr:0 dw:0 dr:0 al:8 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:10484380
```

=====设置主节点=====

```
[root@rdh ~]# drbdadm -- --overwrite-data-of-peer primary r0
[root@rdh ~]# mkfs.xfs /dev/drbd0
[root@rdh ~]# mkdir /drbd_disk
[root@rdh ~]# mount /dev/drbd0 /drbd_disk/
```

文件系统	容量	已用	可用	已用%	挂载点
/dev/mapper/centos-root	48G	6.2G	42G	14%	/
devtmpfs	474M	0	474M	0%	/dev
tmpfs	489M	84K	489M	1%	/dev/shm
tmpfs	489M	7.0M	483M	2%	/run
tmpfs	489M	0	489M	0%	/sys/fs/cgroup
/dev/sda1	497M	142M	356M	29%	/boot
tmpfs	98M	16K	98M	1%	/run/user/42
tmpfs	98M	0	98M	0%	/run/user/0
/dev/drbd0	10G	33M	10G	1%	/drbd disk

```
[root@rdh ~]# echo "test ok">>/drbd_disk/test.txt
[root@rdh ~]# ll /drbd_disk/test.txt
-rw-r--r-- 1 root root 8 2月 24 12:57 /drbd_disk/test.txt
[root@rdh ~]# umount /drbd_disk/
```

```
[root@rdh ~]# drbdadm secondary r0
[root@huatech ~]# drbdadm primary r0
[root@huatech ~]# mount /dev/drbd0 /drbd_disk/
[root@huatech ~]# ll /drbd_disk/
总用量 4
-rw-r--r--. 1 root root 8 2月 24 12:57 test.txt
[root@rdh ~]# yum -y install mariadb*
[root@rdh ~]# mysql_secure_installation
[root@rdh ~]# systemctl start mariadb
[root@huatech ~]# systemctl start mariadb
[root@huatech ~]# systemctl enable mariadb
[root@huatech ~]# mysql_secure_installation
[root@rdh ~]# mkdir /data
[root@rdh ~]# chmod 777 /data
[root@rdh ~]# chown mysql.mysql /data
[root@rdh ~]# systemctl stop mariadb
[root@rdh ~]# cp -r /var/lib/mysql/ /data/
[root@rdh ~]# chown -R mysql.mysql /data/
[root@rdh ~]# chmod -R 700 /data/
[root@rdh ~]# vim /etc/my.cnf
datadir=/data/mysql
socket=/data/mysql/mysql.sock
```

十三、DRBD+Heartbeat+MySQL 高可用

Distributed Replicated Block Device(DRBD)是一个用软件实现的、无共享的、服务器之间镜像块设备内容的存储复制解决方案。

我们可以理解为它其实就是一个网络 Raid 1，两台服务器间就算某台因断电或宕机也不会对数据有任何影响，而真正的热切换可以通过 Heartbeat 方案解决，不需要人工干预。

一、环境描述

系统版本：centos6.6 x64(内核 2.6.32-504.16.2.el6.x86_64)

DRBD 版本：DRBD-8.4.3

node1(主节点) IP: 192.168.0.191 主机名: drbd1.corp.com

node2(从节点) IP: 192.168.0.192 主机名: drbd2.corp.com

(node1) 仅为主节点配置

(node2) 仅为从节点配置

(node1, node2) 为主从节点共同配置

二、安装前准备：(node1, node2)

1、关闭 iptables 和 SELINUX，避免安装过程中报错。

```
1  
2 # service iptables stop  
3 # chkconfig iptables off  
4 # setenforce 0  
5 # vi /etc/selinux/config  
6 -----  
7 SELINUX=disabled  
-----
```

2、配置 hosts 文件

```
1  
2 # vi /etc/hosts  
3 192.168.0.191 drbd1.corp.com  
4 192.168.0.191 drbd2.corp.com
```

3、在两台虚拟机分别添加一块 10G 硬盘分区作为 DRBD 设备磁盘，分别都为 sdb1，大小 10G，并在本地系统创建/store 目录，不做挂载操作。

```
1  
2 # fdisk /dev/sdb  
3 -----  
4 n-p-1-1--+10G--w  
5 -----  
# mkdir /store
```

4、时间同步：

```
1  
# ntpdate -u asia.pool.ntp.org
```

三、DRBD 的安装配置：

1、安装依赖包：(node1, node2)

```
1  
# yum install gcc gcc-c++ make glibc flex kernel-devel kernel-headers
```

2、安装 DRBD：(node1, node2)

```
1 # wget http://oss.linbit.com/drbd/8.4/drbd-8.4.3.tar.gz  
2 # tar zxvf drbd-8.4.3.tar.gz  
3 # cd drbd-8.4.3  
4 # ./configure --prefix=/usr/local/drbd --with-km  
5 # make KDIR=/usr/src/kernels/2.6.32-504.16.2.el6.x86_64/  
6 # make install  
7 # mkdir -p /usr/local/drbd/var/run/drbd  
8 # cp /usr/local/drbd/etc/rc.d/init.d/drbd /etc/rc.d/init.d  
9 # chkconfig --add drbd
```

```
10 # chkconfig drbd on
```

3、加载 DRBD 模块: (node1, node2)

```
1  
# modprobe drbd
```

查看 DRBD 模块是否加载到内核:

```
1  
2 # lsmod |grep drbd  
3 drbd 310172 4  
libcrc32c 1246 1 drbd
```

4、参数配置: (node1, node2)

```
1  
# vi /usr/local/drbd/etc/drbd.conf
```

清空文件内容，并添加如下配置:

```
1 resource r0{  
2   protocol C;  
3  
4   startup { wfc-timeout 0; degr-wfc-timeout 120; }  
5   disk { on-io-error detach; }  
6   net{  
7     timeout 60;  
8     connect-int 10;  
9     ping-int 10;  
10    max-buffers 2048;  
11    max-epoch-size 2048;  
12  }  
13  syncer { rate 200M; }  
14  
15  on drbd1.corp.com{  
16    device /dev/drbd0;  
17    disk /dev/sdb1;  
18    address 192.168.0.191:7788;  
19    meta-disk internal;  
20  }  
21  on drbd2.corp.com{  
22    device /dev/drbd0;  
23    disk /dev/sdb1;  
24    address 192.168.0.192:7788;  
25    meta-disk internal;
```

```
26 }
27 }
```

注：请修改上面配置中的主机名、IP、和 disk 为自己的具体配置

5、创建 DRBD 设备并激活 r0 资源：(node1, node2)

```
1
2 # mknod /dev/drbd0 b 147 0
3 # drbdadm create-md r0
4
5 等待片刻，显示 success 表示 drbd 块创建成功
6 Writing meta data...
7 initializing activity log
8 NOT initializing bitmap
9 New drbd meta data block successfully created.
10
11 ---- Creating metadata ----
12 As with nodes, we count the total number of devices mirrored by DRBD
13 at http://usage.drbd.org.
14
15 The counter works anonymously. It creates a random number to identify
16 the device and sends that random number, along with the kernel and
17 DRBD version, to usage.drbd.org.
18
19 http://usage.drbd.org/cgi-bin/insert_usage.pl?
20
21 nu=716310175600466686&ru=15741444353112217792&rs=1085704704
22
23 * If you wish to opt out entirely, simply enter 'no'.
24 * To continue, just press [RETURN]
25
success
```

再次输入该命令：

```
1 # drbdadm create-md r0
2 成功激活 r0
3 [need to type 'yes' to confirm] yes
4
5 Writing meta data...
6 initializing activity log
7 NOT initializing bitmap
8 New drbd meta data block successfully created.
```

6、启动 DRBD 服务: (node1, node2)

```
1 # service drbd start
```

注: 需要主从共同启动方能生效

7、查看状态: (node1, node2)

```
1 # service drbd status
2 drbd driver loaded OK; device status:
3 version: 8.4.3 (api:1/proto:86-101)
4 GIT-hash: 89a294209144b68adb3ee85a73221f964d3ee515 build by
5 root@drbd1.corp.com, 2015-05-12 21:05:41
6 m:res cs ro ds p mounted
7 fstype
8 0:r0 Connected Secondary/Secondary Inconsistent/Inconsistent C
```

这里 ro:Secondary/Secondary 表示两台主机的状态都是备机状态, ds 是磁盘状态, 显示的状态内容为“Inconsistent 不一致”, 这是因为 DRBD 无法判断哪一方为主机, 应以哪一方的磁盘数据作为标准。

8、将 drbd1.example.com 主机配置为主节点: (node1)

```
1 # drbdsetup /dev/drbd0 primary --force
```

分别查看主从 DRBD 状态:

(node1)

```
1 # service drbd status
2 drbd driver loaded OK; device status:
3 version: 8.4.3 (api:1/proto:86-101)
4 GIT-hash: 89a294209144b68adb3ee85a73221f964d3ee515 build by
5 root@drbd1.corp.com, 2015-05-12 21:05:41
6 m:res cs ro ds p mounted fstype
7 0:r0 Connected Primary/Secondary UpToDate/UpToDate C
```

(node2)

```
1 # service drbd status
2 drbd driver loaded OK; device status:
3 version: 8.4.3 (api:1/proto:86-101)
4 GIT-hash: 89a294209144b68adb3ee85a73221f964d3ee515 build by
5 root@drbd2.corp.com, 2015-05-12 21:05:46
6 m:res cs ro ds p mounted
```

```

fstype
0:r0 Connected Secondary/Primary UpToDate/UpToDate C

```

ro 在主从服务器上分别显示 Primary/Secondary 和 Secondary/Primary
ds 显示 UpToDate/UpToDate
表示主从配置成功。

9、挂载 DRBD: (node1)

从刚才的状态上看到 mounted 和 fstype 参数为空，所以我们这步开始挂载 DRBD 到系统目录/store

```

1
2 # mkfs.ext4 /dev/drbd0
# mount /dev/drbd0 /store

```

注：Secondary 节点上不允许对 DRBD 设备进行任何操作，包括挂载；所有的读写操作只能在 Primary 节点上进行，只有当 Primary 节点挂掉时，Secondary 节点才能提升为 Primary 节点，并自动挂载 DRBD 继续工作。

成功挂载后的 DRBD 状态: (node1)

```

# service drbd status
1 drbd driver loaded OK; device status:
2 version: 8.4.3 (api:1/proto:86-101)
3 GIT-hash: 89a294209144b68adb3ee85a73221f964d3ee515 build by
4 root@drbd1.corp.com, 2015-05-12 21:05:41
5 m:res cs ro ds p mounted
6 fstype
0:r0 Connected Primary/Secondary UpToDate/UpToDate C /store ext4

```

一、Heartbeat 配置

1、安装 heartbeat

```

1
2 # yum install epel-release -y
# yum --enablerepo=epel install heartbeat -y

```

2、设置 heartbeat 配置文件

(node1)

编辑 ha.cf，添加下面配置：

```

1 # vi /etc/ha.d/ha.cf
2 logfile /var/log/ha-log

```

```

3 logfacility local0
4 keepalive 2
5 deadtime 5
6 ucast eth0 192.168.0.192      # 指定对方网卡及 IP
7 auto_failback off
8 node drbd1.corp.com drbd2.corp.com

```

(node2)

编辑 ha.cf，添加下面配置：

```

1
2 # vi /etc/ha.d/ha.cf
3 logfile /var/log/ha-log
4 logfacility local0
5 keepalive 2
6 deadtime 5
7 ucast eth0 192.168.0.191
8 auto_failback off
node drbd1.corp.com drbd2.corp.com

```

3、编辑双机互联验证文件 authkeys，添加以下内容：(node1, node2)

```

1
2 # vi /etc/ha.d/authkeys
3 auth 1
1 crc

```

给验证文件 600 权限

```

1
# chmod 600 /etc/ha.d/authkeys

```

4、编辑集群资源文件：(node1, node2)

```

1 # vi /etc/ha.d/haresources
2 drbd1.corp.com      IPAddr::192.168.0.190/24/eth0      drbddisk::r0
   Filesystem:::/dev/drbd0::/store::ext4 killnfsd

```

注：该文件内 IPAddr, Filesystem 等脚本存放路径在 /etc/ha.d/resource.d/ 下，也可在该目录下存放服务启动脚本（例如：mysql, www），将相同脚本名称添加到 /etc/ha.d/haresources 内容中，从而跟随 heartbeat 启动而启动该脚本。

IPAddr::192.168.0.190/24/eth0：用 IPAddr 脚本配置对外服务的浮动虚拟 IP

drbddisk::r0：用 drbddisk 脚本实现 DRBD 主从节点资源组的挂载和卸载

Filesystem:::/dev/drbd0::/store::ext4：用 Filesystem 脚本实现磁盘挂载和卸载

5、编辑脚本文件 killnfsd，用来重启 NFS 服务：(node1, node2)

```
1  
2 # vi /etc/ha.d/resource.d/killnfsd  
killall -9 nfsd; /etc/init.d/nfs restart;exit 0
```

赋予 755 执行权限：

```
1  
# chmod 755 /etc/ha.d/resource.d/killnfsd
```

二、创建 DRBD 脚本文件 drbddisk: (node1, node2)

编辑 drbddisk，添加下面的脚本内容

```
1 # vi /etc/ha.d/resource.d/drbddisk  
  
1 #!/bin/bash  
2 #  
3 # This script is inteded to be used as resource script by heartbeat  
4 #  
5 # Copyright 2003-2008 LINBIT Information Technologies  
6 # Philipp Reisner, Lars Ellenberg  
7 #  
8 #####  
9  
10 DEFAULTFILE="/etc/default/drbd"  
11 DRBDADM="/sbin/drbdadm"  
12  
13 if [ -f $DEFAULTFILE ]; then  
14 . $DEFAULTFILE  
15 fi  
16  
17 if [ "$#" -eq 2 ]; then  
18 RES="$1"  
19 CMD="$2"  
20 else  
21 RES="all"  
22 CMD="$1"  
23 fi  
24  
25 ## EXIT CODES  
26 # since this is a "legacy heartbeat R1 resource agent" script,  
27 # exit codes actually do not matter that much as long as we conform  
28 to  
29 # http://wiki.linux-ha.org/HeartbeatResourceAgent
```

```
30 # but it does not hurt to conform to lsb init-script exit codes,
31 # where we can.
32 # http://refspecs.linux-foundation.org/LSB_3.1.0/
33 #LSB-Core-generic/LSB-Core-generic/iniscriptact.html
34 #####
35
36 drbd_set_role_from_proc_drbd()
37 {
38 local out
39 if ! test -e /proc/drbd; then
40 ROLE="Unconfigured"
41 return
42 fi
43
44 dev=$( $DRBDADM sh-dev $RES )
45 minor=${dev#/dev/drbd}
46 if [[ $minor = *[!0-9]* ]] ; then
47 # sh-minor is only supported since drbd 8.3.1
48 minor=$( $DRBDADM sh-minor $RES )
49 fi
50 if [[ -z $minor ]] || [[ $minor = *[!0-9]* ]] ; then
51 ROLE=Unknown
52 return
53 fi
54
55 if out=$(sed -ne "/^ *$minor: cs:/ { s// /g; p; q; }" /proc/drbd);
56 then
57 set -- $out
58 ROLE=${5%/*}
59 : ${ROLE:=Unconfigured} # if it does not show up
60 else
61 ROLE=Unknown
62 fi
63 }
64
65 case "$CMD" in
66     start)
67 # try several times, in case heartbeat deadtime
68 # was smaller than drbd ping time
69 try=6
70 while true; do
71 $DRBDADM primary $RES && break
72 let "--try" || exit 1 # LSB generic error
```

```
73    sleep 1
74    done
75    ;;
76    stop)
77    # heartbeat (haresources mode) will retry failed stop
78    # for a number of times in addition to this internal retry.
79    try=3
80    while true; do
81        $DRBDADM secondary $RES && break
82        # We used to lie here, and pretend success for anything != 11,
83        # to avoid the reboot on failed stop recovery for "simple
84        # config errors" and such. But that is incorrect.
85        # Don't lie to your cluster manager.
86        # And don't do config errors...
87        let --try || exit 1 # LSB generic error
88        sleep 1
89    done
90    ;;
91    status)
92    if [ "$RES" = "all" ]; then
93        echo "A resource name is required for status inquiries."
94        exit 10
95    fi
96    ST=$( $DRBDADM role $RES )
97    ROLE=${ST%/*}
98    case $ROLE in
99        Primary|Secondary|Unconfigured)
100       # expected
101       ;;
102       *)
103       # unexpected. whatever...
104       # If we are unsure about the state of a resource, we need to
105       # report it as possibly running, so heartbeat can, after failed
106       # stop, do a recovery by reboot.
107       # drbdsetup may fail for obscure reasons, e.g. if /var/lock/ is
108       # suddenly readonly. So we retry by parsing /proc/drbd.
109       drbd_set_role_from_proc_drbd
110   esac
111   case $ROLE in
112       Primary)
113       echo "running (Primary)"
114       exit 0 # LSB status "service is OK"
115       ;;
```

```
116 Secondary|Unconfigured)
117 echo "stopped ($ROLE)"
118 exit 3 # LSB status "service is not running"
119 ;;
120 *)
121 # NOTE the "running" in below message.
122 # this is a "heartbeat" resource script,
123 # the exit code is _ignored_.
124 echo "cannot determine status, may be running ($ROLE)"
125 exit 4 # LSB status "service status is unknown"
126 ;;
127 esac
128 ;;
129 *)
130 echo "Usage: drbddisk [resource] {start|stop|status}"
131 exit 1
132 ;;
133 esac

exit 0
```

赋予 755 执行权限：

```
1
# chmod 755 /etc/ha.d/resource.d/drbddisk
```

三、启动 HeartBeat 服务

在两个节点上启动 HeartBeat 服务，先启动 node1：(node1, node2)

```
1
2 # service heartbeat start
# chkconfig heartbeat on
```

现在从其他机器能够 ping 通虚 IP 192.168.0.190，表示配置成功

四、配置 NFS：(node1, node2)

编辑 exports 配置文件，添加以下配置：

```
1
2 # vi /etc/exports
/store *(rw, no_root_squash)
```

重启 NFS 服务：

```
1 # service rpcbind restart
```

```

2 # service nfs restart
3 # chkconfig rpcbind on
4 # chkconfig nfs off

```

注：这里设置 NFS 开机不要自动运行，因为/etc/init.d/resource.d/killnfsd 该脚本会控制 NFS 的启动。

五、测试高可用

1、正常热备切换

在客户端挂载 NFS 共享目录

```

1
# mount -t nfs 192.168.0.190:/store /tmp

```

模拟将主节点 node1 的 heartbeat 服务停止，则备节点 node2 会立即无缝接管；测试客户端挂载的 NFS 共享读写正常。

此时备机 node2 上的 DRBD 状态：

```

# service drbd status
1 drbd driver loaded OK; device status:
2 version: 8.4.3 (api:1/proto:86-101)
3 GIT-hash: 89a294209144b68adb3ee85a73221f964d3ee515 build by
4 root@drbd2.corp.com, 2015-05-12 21:05:41
5 m:res cs ro ds p mounted
6 fstype
0:r0 Connected Primary/Secondary UpToDate/UpToDate C /store ext4

```

2、异常宕机切换

强制关机，直接关闭 node1 电源

node2 节点也会立即无缝接管，测试客户端挂载的 NFS 共享读写正常。

此时 node2 上的 DRBD 状态：

```

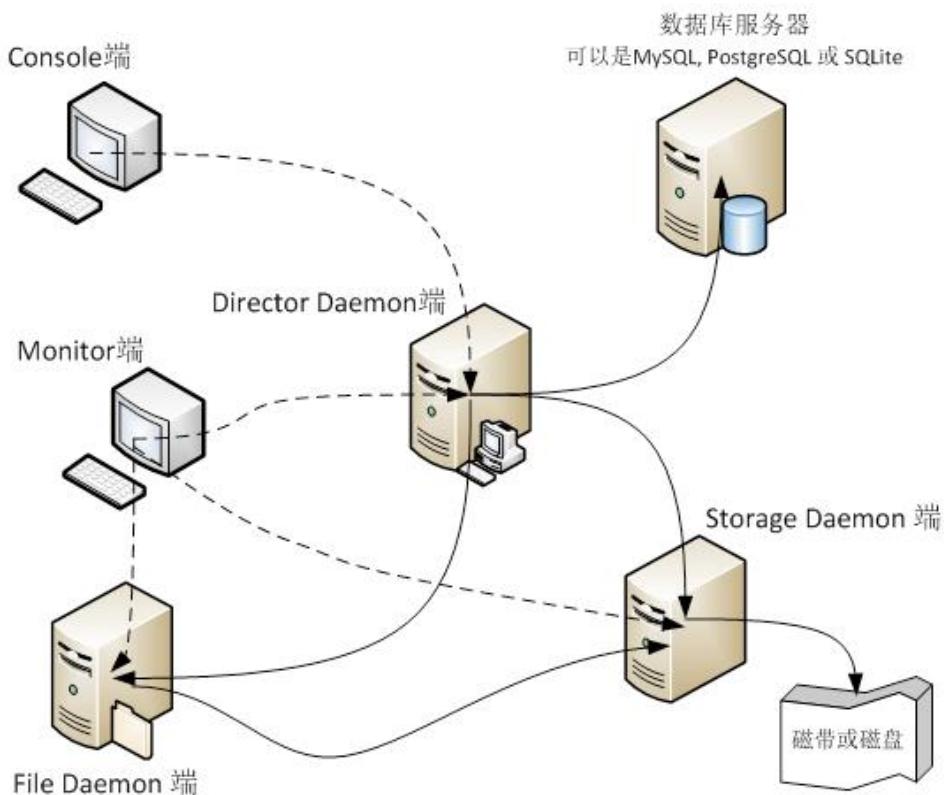
# service drbd status
1 drbd driver loaded OK; device status:
2 version: 8.4.3 (api:1/proto:86-101)
3 GIT-hash: 89a294209144b68adb3ee85a73221f964d3ee515 build by
4 root@drbd2.corp.com, 2015-05-12 21:05:41
5 m:res cs ro ds p mounted
6 fstype
0:r0 Connected Primary/Unknown UpToDate/DUnknown C /store ext4

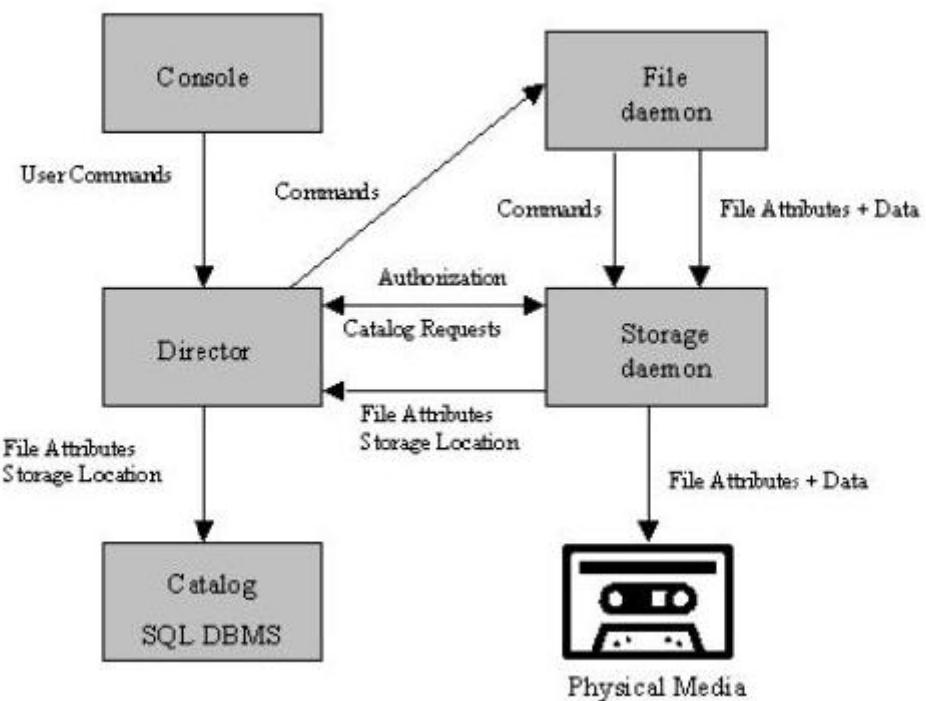
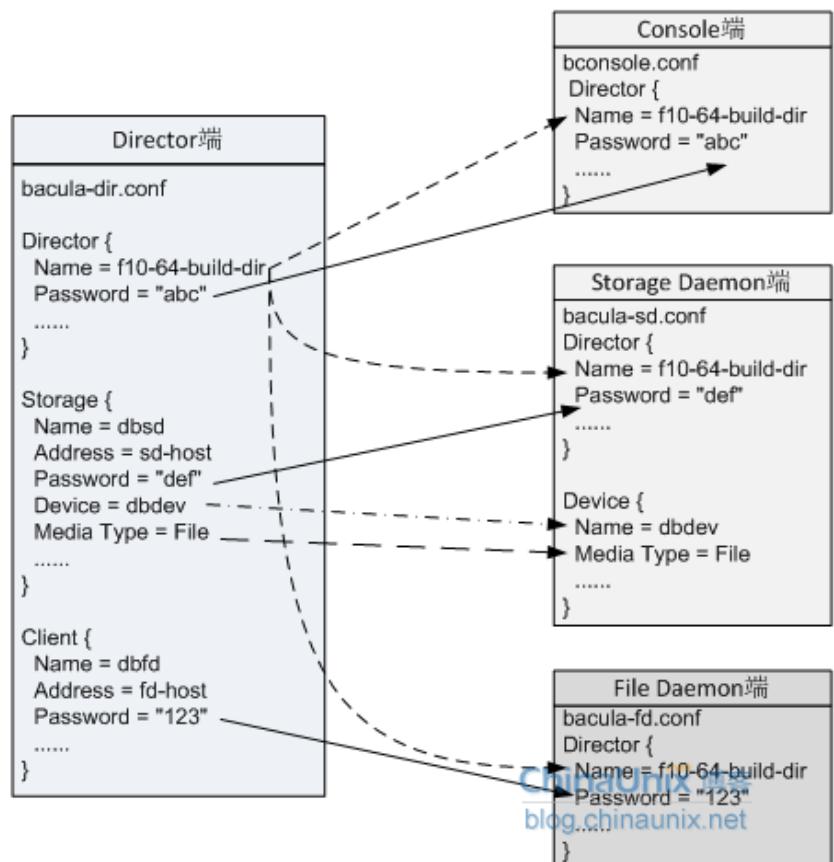
```

+-----+ [Bacula Server] 10.0.0.30 10.0.0.51 [Bacula Client] +-----+	(Backup Target)
Bacula Director +-----+ +-----+ Bacula File Daemon	
Bacula Storage	
Maria DB +-----+ +-----+	

十四、备份服务器 Bacula

架构图





服务器端

```
[root@server01 ~]# yum -y install epel*
[root@server01 ~]# vim /etc/my.cnf
character-set-server=utf8
[root@server01 ~]# service mysqld start
[root@server01 ~]# chkconfig mysqld on
[root@server01 ~]# mysql_secure_installation
Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
[root@server01 ~]# mysql -u root -p
mysql> select user,host,password from mysql.user;
+-----+-----+-----+
| user | host      | password          |
+-----+-----+-----+
| root | localhost | *EAC77AEF0F3C9DF3FCB4B89A03D80D0F0B5C1794 |
| root | server01.rdh.com | *EAC77AEF0F3C9DF3FCB4B89A03D80D0F0B5C1794 |
| root | 127.0.0.1 | *EAC77AEF0F3C9DF3FCB4B89A03D80D0F0B5C1794 |
+-----+-----+-----+
3 rows in set (0.00 sec)

[root@server01 ~]# yum -y install bacula-director-mysql bacula-storage-mysql
bacula-console
[root@server01 ~]# cd /usr/libexec/bacula/
[root@server01 bacula]# ./grant_mysql_privileges -p
[root@server01 bacula]# ./create_mysql_database -p
[root@server01 bacula]# ./make_mysql_tables -p
mysql> set password for bacula=password('password');
mysql> set password for bacula@localhost=password('password');
mysql> select user,host,password from mysql.user;
+-----+-----+-----+
| user | host      | password          |
+-----+-----+-----+
| root | localhost | *EAC77AEF0F3C9DF3FCB4B89A03D80D0F0B5C1794 |
| root | server01.rdh.com | *EAC77AEF0F3C9DF3FCB4B89A03D80D0F0B5C1794 |
| root | 127.0.0.1 | *EAC77AEF0F3C9DF3FCB4B89A03D80D0F0B5C1794 |
| bacula | %        | *2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19 |
| bacula | localhost | *2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19 |
+-----+-----+-----+
```

```
5 rows in set (0.00 sec)
mysql> show databases;
mysql> use bacula;
mysql> show tables;
[root@server01 ~]# vim /etc/bacula/bacula-dir.conf
Director {                                # define myself
    Name = bacula-dir
    DIRport = 9101                      # where we listen for UA connections
    QueryFile = "/usr/libexec/bacula/query.sql"
    WorkingDirectory = "/var/spool/bacula"
    PidDirectory = "/var/run"
    Maximum Concurrent Jobs = 1
    Password = "password"                # Console password
    Messages = Daemon
}
FileSet {
    Name = "Full Set"
    Include {
        Options {
            signature = MD5
            compression = GZIP
        }
        File = /home
    }
}
Client {
    Name = bacula-fd
    Address = server02.rdh.com
    FDPort = 9102
    Catalog = MyCatalog
    Password = "password"                # password for FileDaemon
    File Retention = 30 days             # 30 days
    Job Retention = 6 months            # six months
    AutoPrune = yes                   # Prune expired Jobs/Files
}
Storage {
    Name = File
    # Do not use "localhost" here
    Address = server01.rdh.com          # N.B. Use a fully qualified name here
    SDPort = 9103
    Password = "password"
    Device = FileStorage
    Media Type = File
}
Catalog {
```

```
Name = MyCatalog
# Uncomment the following line if you want the dbi driver
# dbdriver = "dbi:sqlite3"; dbaddress = 127.0.0.1; dbport =
  dbname = "bacula"; dbuser = "bacula"; dbpassword = "password"
}
Pool {
  Name = File
  Pool Type = Backup
  Recycle = yes          # Bacula can automatically recycle Volumes
  AutoPrune = yes        # Prune expired volumes
  Volume Retention = 180 days
  Maximum Volume Jobs = 1
  Label Format = Vol-    # one year
}
[root@server01 ~]# vim /usr/libexec/bacula/make_catalog_backup.pl
[root@server01 ~]# /etc/rc.d/init.d/bacula-dir start
[root@server01 ~]# vim /etc/bacula/bacula-sd.conf
Director {
  Name = bacula-dir
  Password = "password"
}
[root@server01 ~]# /etc/rc.d/init.d/bacula-sd start
[root@server01 ~]# chkconfig bacula-sd on
[root@server01 ~]# vim /etc/bacula/bconsole.conf
Director {
  Name = bacula-dir
  DIRport = 9101
  address = server01.rdh.com
  Password = "password"
}
```

客户端

```
=====client=====
[root@server02 ~]# yum -y install bacula-client bacula-console
[root@server02 ~]# vim /etc/bacula/bacula-fd.conf
Director {
  Name = bacula-dir
  Password = "password"
}
[root@server02 ~]# /etc/rc.d/init.d/bacula-fd start
[root@server02 ~]# chkconfig bacula-fd on
[root@server02 ~]# vim /etc/bacula/bconsole.conf
[root@server02 ~]# bconsole
*label
```

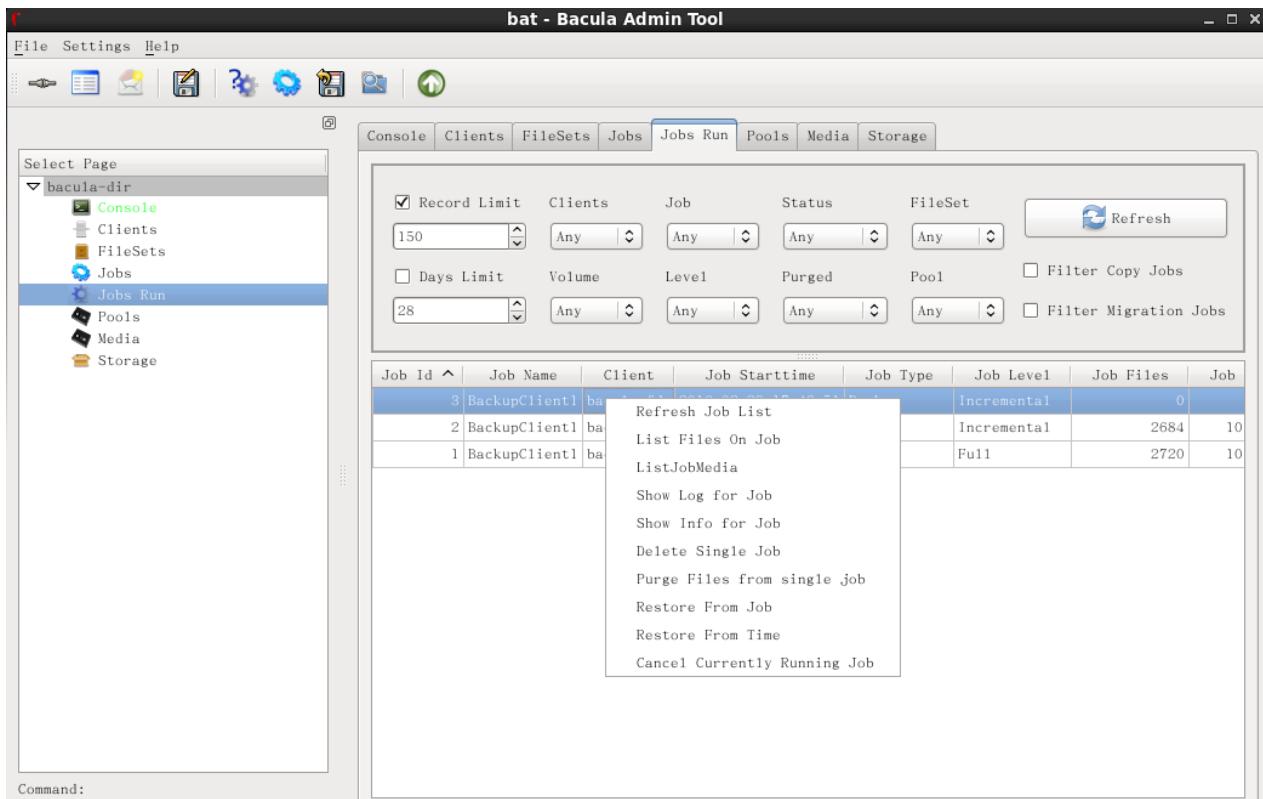
```
*run
OK to run? (yes/mod/no) : yes
*exit
[root@server01 ~]# ll /tmp
-rw-r----- 1 root root      205 2月 28 16:15 Vol-20110515
```

恢复

```
*restore
Select item: (1-13) :
5
# select 5
$ ls
$ mark home
$ lsmark
$ done
[root@file01 ~]# ll /tmp/bacula-restores
```

使用图形界面 GUI

```
[root@d1p ~]# yum -y install bacula-console-bat
[root@server02 桌面]# vim /etc/bacula/bat.conf
Director {
    Name = bacula-dir
    DIRport = 9101
    address = server01.rdh.com
    Password = "password"
}
```



客户端 GUI 图像界面

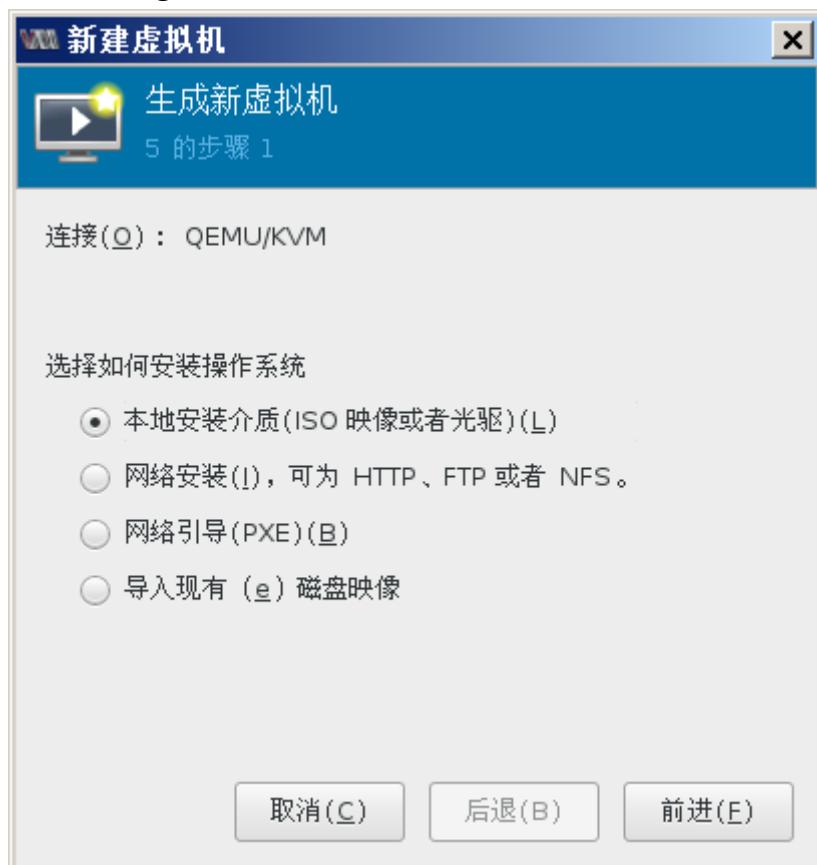
project/bacula/Win32_64/5.2.10/bacula-win64-5.2.10.exe

十五、虚拟化

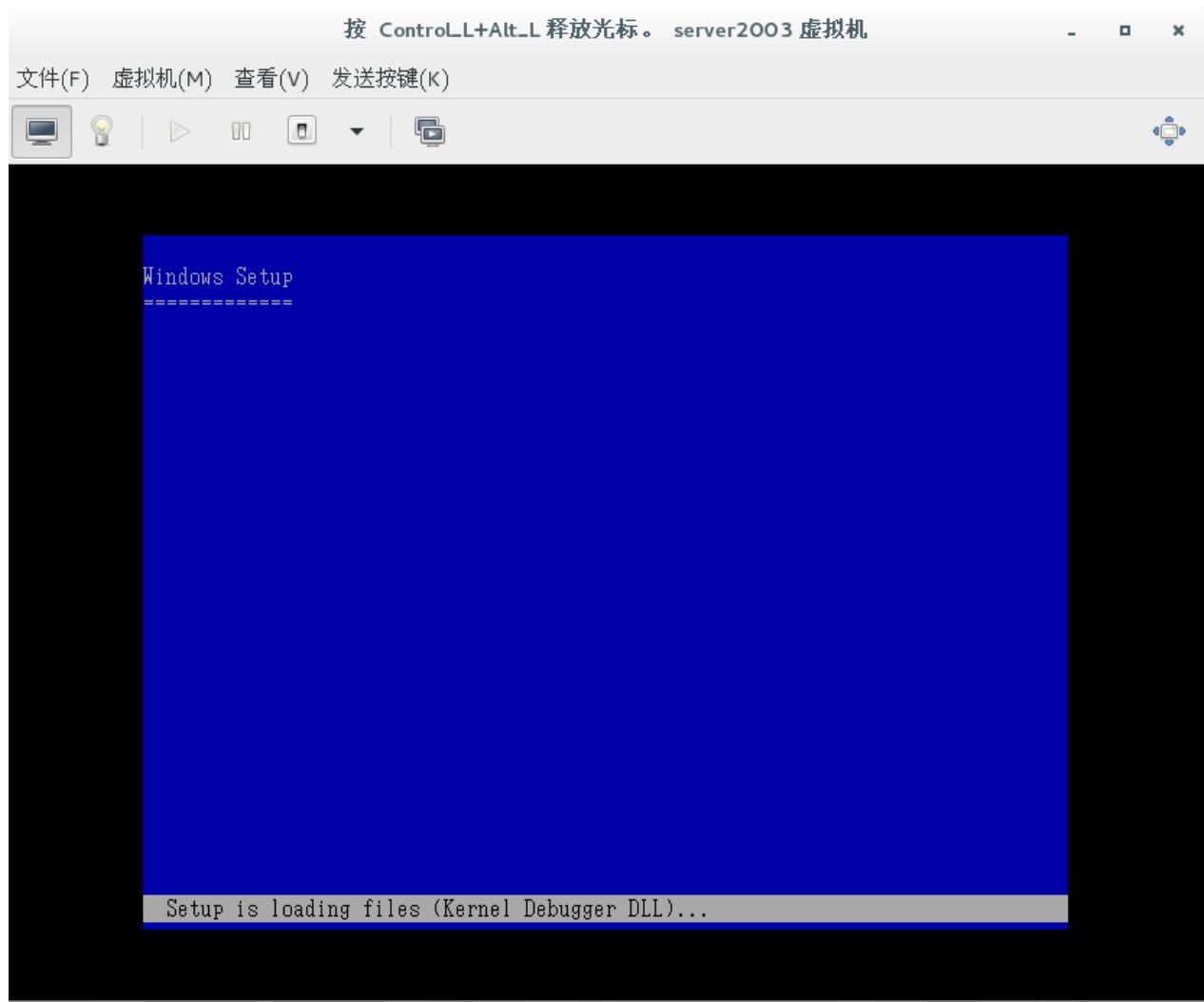
KVM

```
[root@rdh ~]# yum -y install qemu-kvm libvirt virt-install bridge-utils
[root@rdh ~]# lsmod | grep kvm
kvm_intel           162153  0
kvm                 525259  1 kvm_intel
[root@rdh ~]# systemctl start libvirtd
[root@rdh ~]# systemctl enable libvirtd
[root@rdh ~]# nmcli c add type bridge autoconnect yes con-name br0 iface br0
[root@rdh ~]# nmcli c modify br0 ipv4.addresses 192.168.1.111/24 ipv4.method manual
[root@rdh ~]# nmcli c modify br0 ipv4.gateway 192.168.1.1
[root@rdh ~]# nmcli c modify br0 ipv4.dns 192.168.1.1
[root@rdh ~]# nmcli c delete eno16777728
[root@rdh 桌面]# nmcli c add type bridge-slave autoconnect yes con-name eno16777728
iface eno16777728 master br0
[root@rdh 桌面]# systemctl stop NetworkManager;systemctl start NetworkManager
[root@rdh ~]# mkdir -p /var/kvm/images
> --name centos6 \
> --ram 1024 \
```

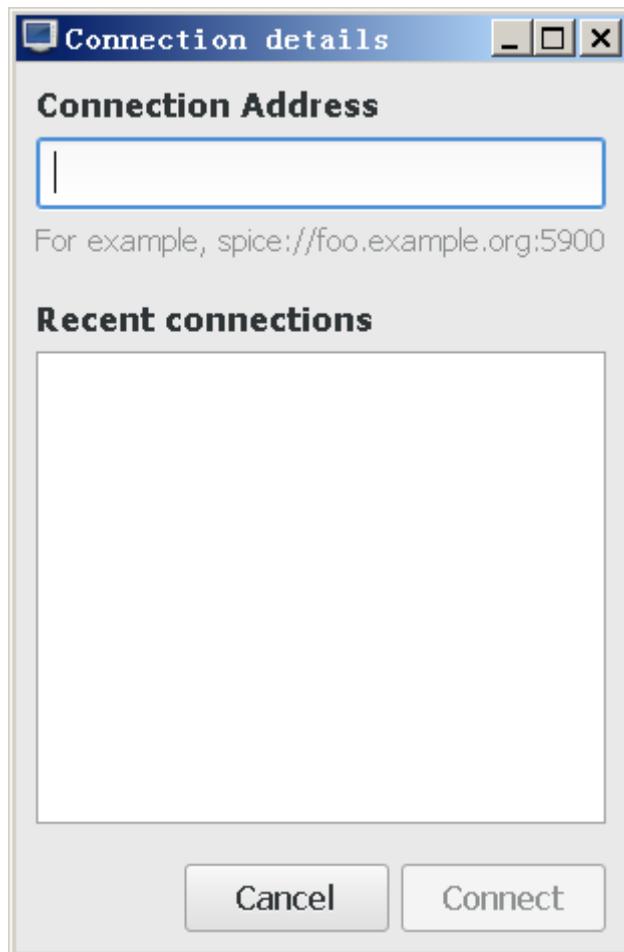
```
> --disk path=/var/kvm/images/centos6.img size=20 \
> --vcpus 2 \
> --os-type linux \
> --os-variant rhel6 \
> --network bridge=br0 \
> --graphics none \
> --console pty,target_type=serial \
> --location 'http://192.168.1.107/CentOS-6.5-x86_64-bin-DVD1.iso' \
[root@rdh ~]# yum -y install virt-manager
[root@rdh ~]# virt-manager
```



```
[root@rdh ~]# virsh start centos7
[root@rdh ~]# virsh start centos7 --console
[root@rdh ~]# virsh shutdown centos7
[root@rdh ~]# virsh destroy centos7
[root@rdh ~]# virsh autostart centos7
[root@rdh ~]# virsh list
```



```
[root@rdh ~]# yum -y install spice-server spice-protocol  
https://fedorahosted.org/released/virt-viewer/virt-viewer-x64-3.0.msi
```



```
[root@kvm01 ~]#  
virsh migrate --live centos7 qemu+ssh://10.0.0.22/system
```

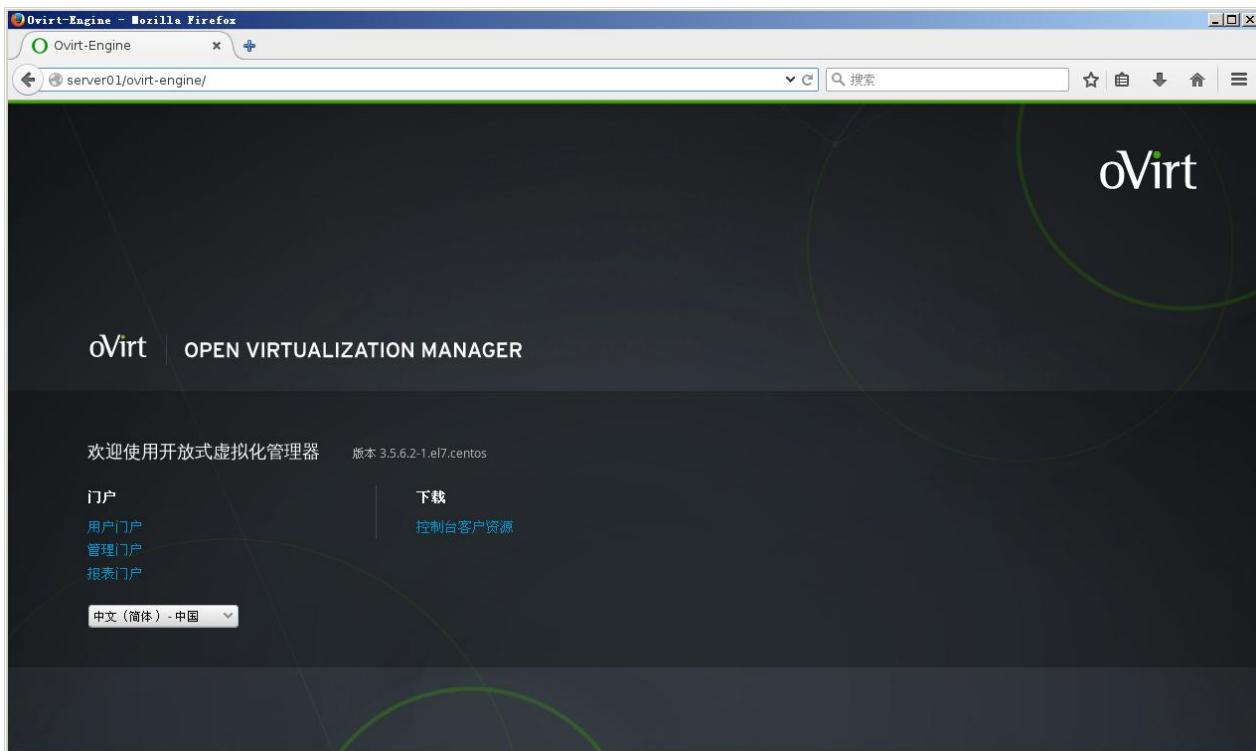
OVIRT

=====控制端=====

```
[root@server01 ~]# yum -y install  
http://resources.ovirt.org/pub/yum-repo/ovirt-release35.rpm  
[root@server01 ~]# yum -y install ovirt-engine  
[root@server01 ~]# touch /etc/exports  
[root@server01 ~]# systemctl restart rpcbind nfs-server  
[root@server01 ~]# engine-setup  
Firewall manager to configure (firewalld): firewalld  
  
Internet cx_04_50_LC_01_cr_be_i1_10_13_00_4D_00_00_00_07_12_E1_27_43_01  
Web access is enabled at:  
    http://server01:80/ovirt-engine  
    https://server01:443/ovirt-engine  
  
--- END OF SUMMARY ---  
  
[ INFO ] Starting engine service  
[ INFO ] Restarting httpd  
[ INFO ] Stage: Clean up  
    Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-setup-20160229155249-xwbzjq.log  
[ INFO ] Generating answer file '/var/lib/ovirt-engine/setup/answers/20160229155446-setup.conf'  
[ INFO ] Stage: Pre-termination  
[ INFO ] Stage: Termination  
[ INFO ] Execution of setup completed successfully
```

```
[root@server01 ~]# vim /etc/sysconfig/nfs
```

```
NFS4_SUPPORT="no"
[root@server01 ~]# mkdir /var/lib/exports/data
[root@server01 ~]# chown vdsm:kvm /var/lib/exports/data
[root@server01 ~]# vim /etc/exports.d/ovirt-engine-iso-domain.exports
/var/lib/exports/iso    192.168.88.0/24(rw)
/var/lib/exports/data   192.168.88.0/24(rw)
[root@server01 ~]# systemctl restart rpc-statd nfs-server
=====计算节点=====
[root@server02 ~]# yum -y install qemu-kvm libvirt virt-install bridge-utils
[root@server02 ~]# lsmod | grep kvm
kvm_intel           162153  0
kvm                 525259  1 kvm_intel
[root@server02 ~]# systemctl start libvirtd
[root@server02 ~]# systemctl enable libvirtd
[root@server02 ~]# nmcli c add type bridge autoconnect yes con-name br0 ifname br0
[root@server02 ~]# nmcli c modify br0 ipv4.addresses 192.168.88.238/24 ipv4.method
manual
[root@server02 ~]# nmcli c modify br0 ipv4.gateway 192.168.88.1
[root@server02 ~]# nmcli c modify br0 ipv4.dns 192.168.88.1
[root@server02 ~]# nmcli c delete eno16777728
[root@server01 桌面]# nmcli c add type bridge-slave autoconnect yes con-name
eno16777728 ifname eno16777728 master br0
[root@server01 桌面]# systemctl stop NetworkManager;systemctl start
NetworkManager
[root@server02 ~]# yum -y install vdsm
http://resources.ovirt.org/pub/yum-repo/ovirt-release35.rpm
[root@server02 ~]# yum -y install vdsm
```



This screenshot shows the 'Hosts' management screen in the oVirt Engine web interface. The URL in the address bar is 'https://server01/ovirt-engine/webadmin/?locale=zh_CN#hosts-general'. The interface includes a top navigation bar with user information ('admin') and various management links. The main area has a toolbar with tabs like '数据中心' (Datacenter), '集群' (Cluster), '主机' (Host), etc. On the left, a sidebar lists '系统' (System), '数据中心' (Datacenter), and specific hosts like 'Default', 'ovirtmgmt', and '外部供应商' (External Supplier). The central part of the screen displays a table of hosts. One host, 'server02', is highlighted with a red border. The table columns include '名称' (Name), '主机名/AP' (Hostname/AP), '集群' (Cluster), '数据中心' (Datacenter), '状态' (Status), '虚拟机' (Virtual Machines), '内存' (Memory), 'CPU' (CPU), and '网络' (Network). Below the table, detailed information for 'server02' is shown in a tabular format. At the bottom, there are status messages and a Firefox privacy notice.

国产虚拟化 CECOSI

1. 安装环境，硬件要求：

CPU: AMD 或 Intel 64 位，开启 CPU 虚拟化

内存：最小 4G，推荐 16G

网卡：千兆网卡

硬盘：最小 20G，推荐 50G

CecOS 系统光盘一张，CD 或 DVD 光驱

2.安装步骤

2.1 安装引导

现在开始安装系统，打开计算机，光盘放入光驱，进入系统安装引导界面。



选择第一个选项，开始安装。

2.2 检测光盘介质

是否检测光盘，可以根据实际情况选择 OK 或者 SKIP，选择 OK 后，开始检测光盘，检测完成后会弹出光驱，这时需要重新载入光盘才能继续安装，选择 SKIP，则直接开始安装。



显示下图所示界面，点击 NEXT，进入下一步。

2.3 安装欢迎界面



2.4 选择安装过程中的语言

选择安装语言，完成后点击 NEXT，进入下一步

© OPENFANS <http://www.openfans.org>

What language would you like to use during the installation process?

- Bulgarian (Български)
- Catalan (Català)
- Chinese(Simplified) (中文 (简体))
- Chinese(Traditional) (中文 (正體))
- Croatian (Hrvatski)
- Czech (Čeština)
- Danish (Dansk)
- Dutch (Nederlands)
- English (English)**
- Estonian (eesti keel)
- Finnish (suomi)
- French (Français)
- German (Deutsch)
- Greek (Ελληνικά)
- Gujarati (ગુજરાતી)
- Hebrew (עברית)
- Hindi (हिन्दी)

[Back](#) [Next](#)

2.5 选择键盘布局类型

选择键盘布局，完成后点击 NEXT

© OPENFANS <http://www.openfans.org>

Select the appropriate keyboard for the system.

- Portuguese
- Romanian
- Russian
- Serbian
- Serbian (latin)
- Slovak (qwerty)
- Slovenian
- Spanish
- Swedish
- Swiss French
- Swiss French (latin1)
- Swiss German
- Swiss German (latin1)
- Turkish
- U.S. English**
- U.S. International
- Ukrainian
- United Kingdom

[Back](#) [Next](#)

2.6 选择磁盘

选择需要安装的磁盘类型，确定后点击 NEXT，

What type of devices will your installation involve?

Basic Storage Devices

- Installs or upgrades to typical types of storage devices. If you're not sure which option is right for you, this is probably it.

Specialized Storage Devices

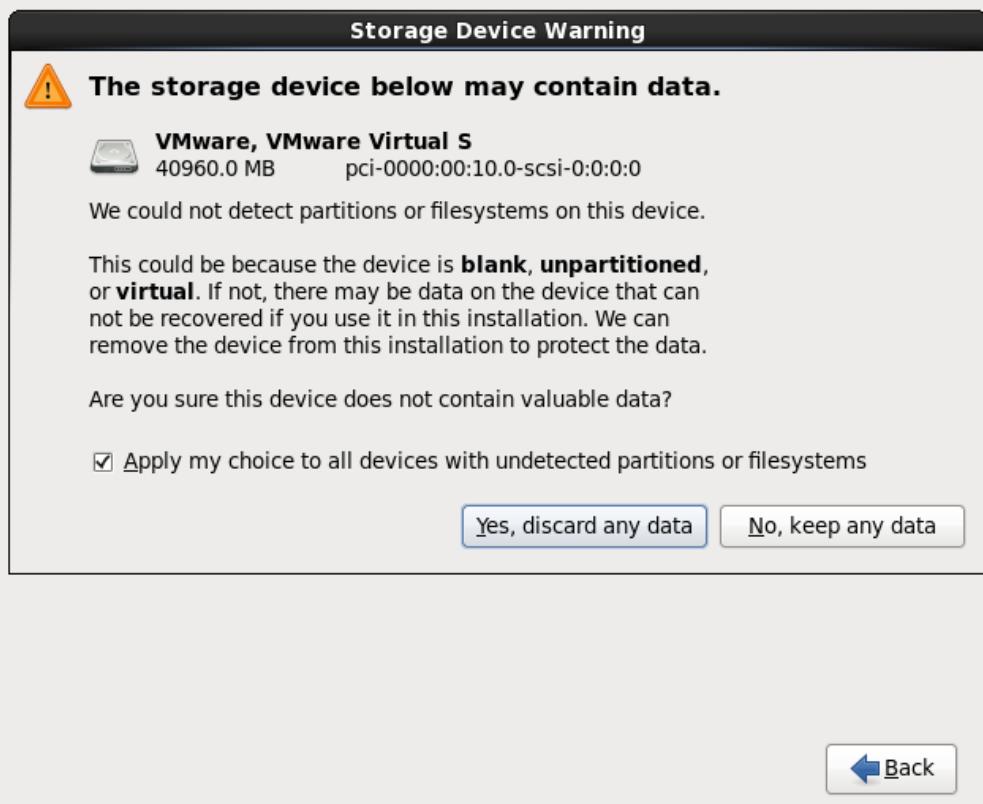
- Installs or upgrades to enterprise devices such as Storage Area Networks (SANs). This option will allow you to add FCoE / iSCSI / zFCP disks and to filter out devices the installer should ignore.

[Back](#)

[Next](#)

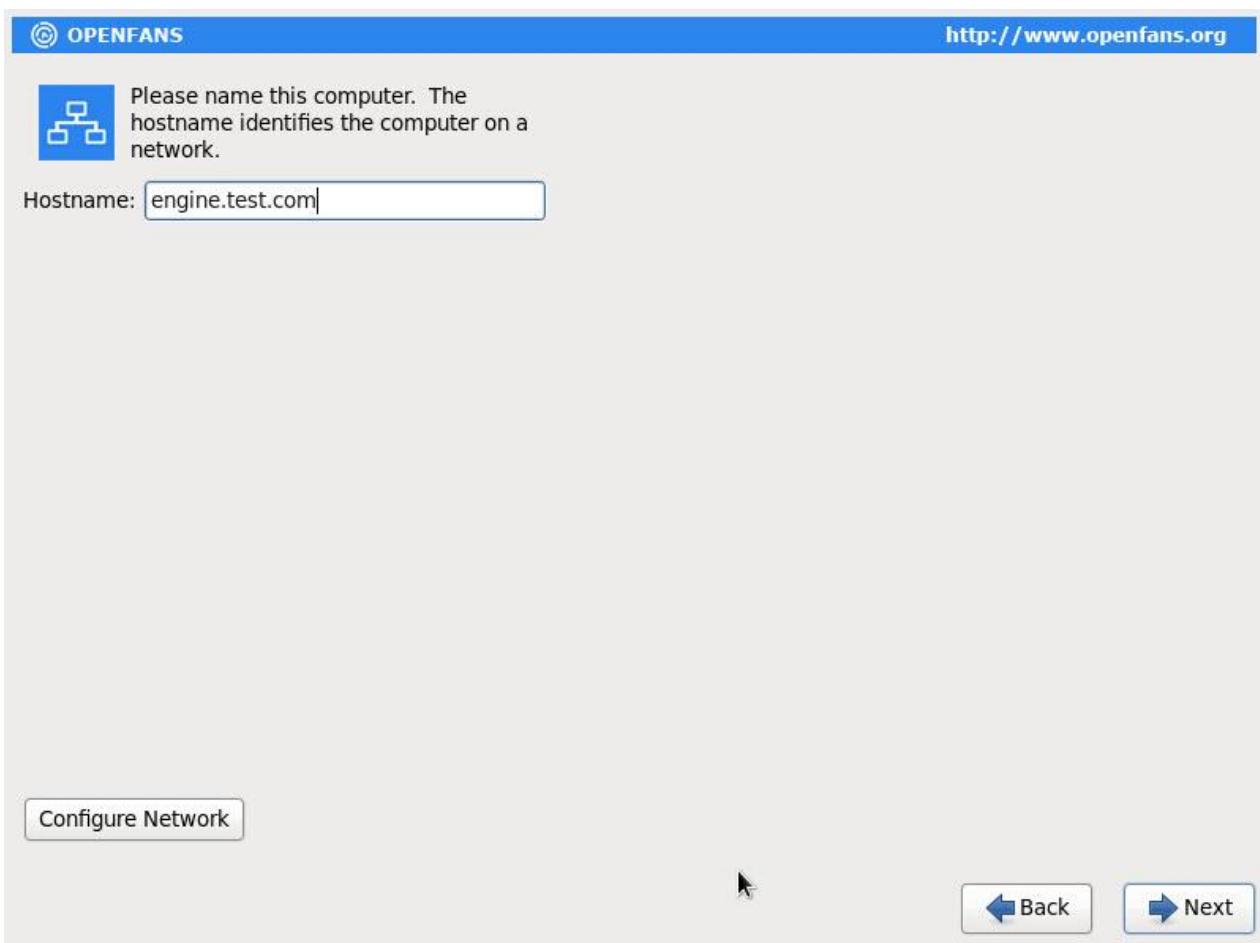
2.7 初始化硬盘

提示是否覆盖数据，根据实际选择覆盖或保留

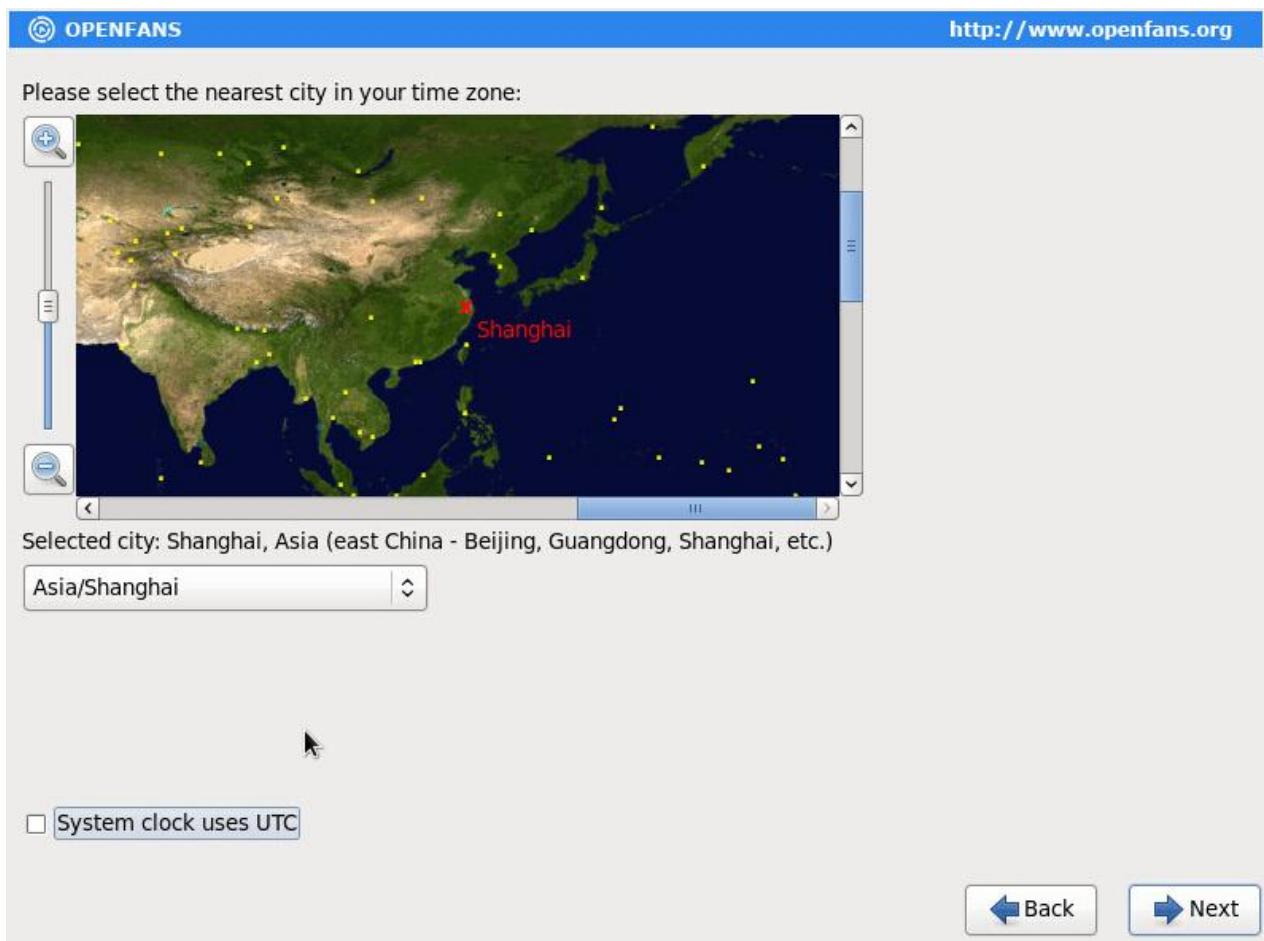


2.8 设置主机名与网络

确认选择，点击 NEXT，进入下一步，设置主机名，同时也可以选择是否配置网络

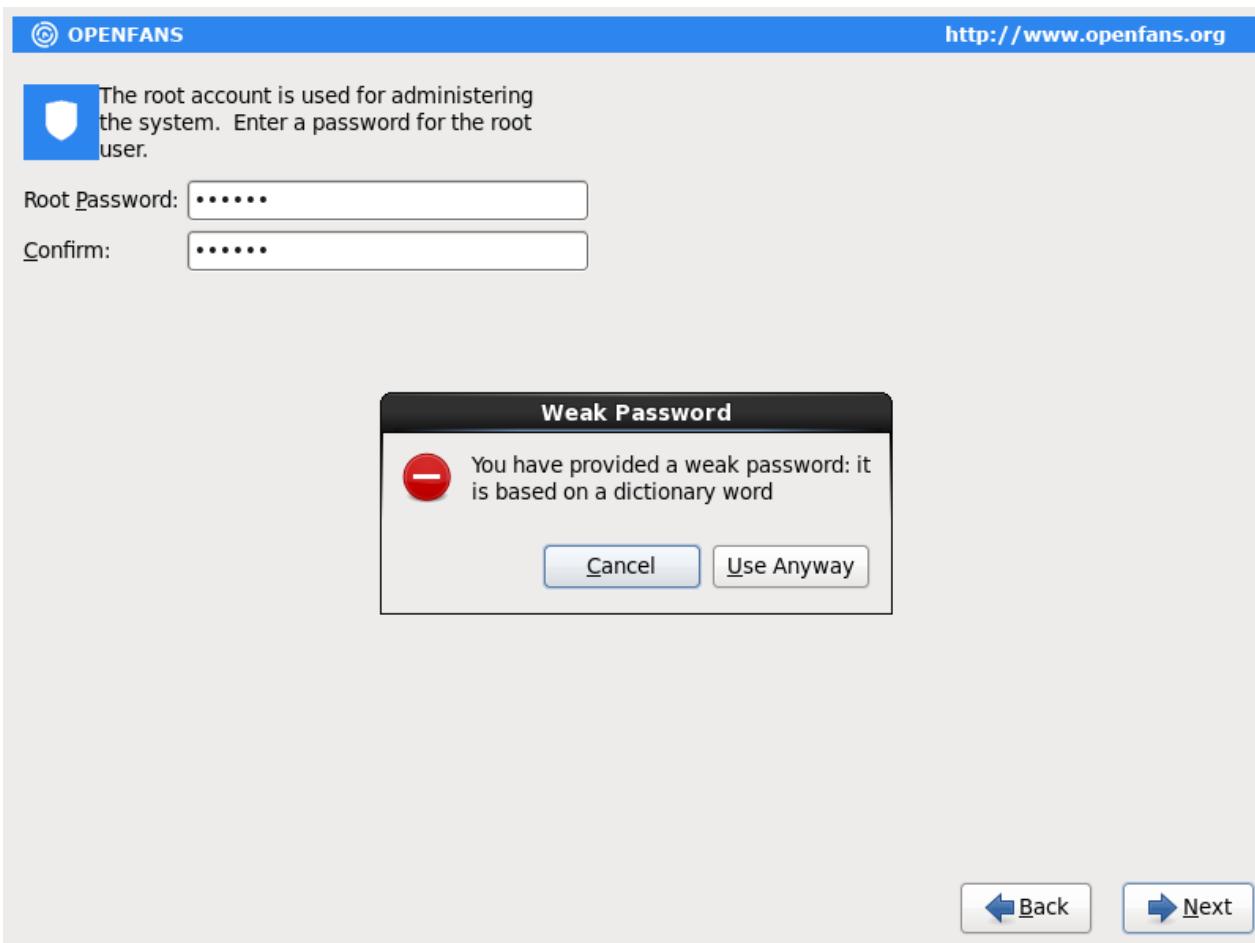


配置完成进入下一步，选择所在时区，默认为美国纽约，选择为上海并选择不使用 UTC 时间。



2.9 设置管理员密码(root 密码)

下一步，设置密码，如果密码强度不够，会显示下图内容，建议更换密码



2.10 磁盘分区配置

如果没有上图所示内容，进入下一步，配置磁盘，如果为空盘，选择第一个选项，并选中底部 Review and modify portioning layout 选项，查看磁盘分区情况。

Which type of installation would you like?

Use All Space

 ALL
Removes all partitions on the selected device(s). This includes partitions created by other operating systems.

Tip: This option will remove data from the selected device(s). Make sure you have backups.

Replace Existing Linux System(s)

 Replace
Removes only Linux partitions (created from a previous Linux installation). This does not remove other partitions you may have on your storage device(s) (such as VFAT or FAT32).

Tip: This option will remove data from the selected device(s). Make sure you have backups.

 Shrink Current System

Shrinks existing partitions to create free space for the default layout.

 Use Free Space

Retains your current data and partitions and uses only the unpartitioned space on the selected device(s), assuming you have enough free space available.

 Create Custom Layout

Manually create your own custom layout on the selected device(s) using our partitioning tool.

Encrypt system

Review and modify partitioning layout

 Back

 Next

下图为硬盘分区状态，可根据实际需求手动分区。

© OPENFANS <http://www.openfans.org>

Please Select A Device

Device	Size (MB)	Mount Point/ RAID/Volume	Type	Format
▽ LVM Volume Groups				
▽ vg_test	40456			
lv_root	36488	/	ext4	✓
lv_swap	3968		swap	✓
▽ Hard Drives				
▽ sda (/dev/sda)				
sda1	500	/boot	ext4	✓
sda2	40459	vg_test	physical volume (LVM)	✓

[Create](#) [Edit](#) [Delete](#) [Reset](#)

[Back](#) [Next](#)

如果不手动分区，则直接点击 NEXT 进入下一步

 OPENFANS <http://www.openfans.org>

Which type of installation would you like?

Use All Space
Removes all partitions on the selected device(s). This includes partitions created by other operating systems.

Tip: This option will remove data from the selected device(s). Make sure you have backups.

Replace Existing Linux System(s)
Removes only Linux partitions (created from a previous Linux installation). This does not remove other partitions you may have on your storage device(s) (such as VFAT or FAT32).

Tip: This option will remove data from the selected device(s). Make sure you have backups.

Shrink Current System
Shrinks existing partitions to create free space for the default layout.

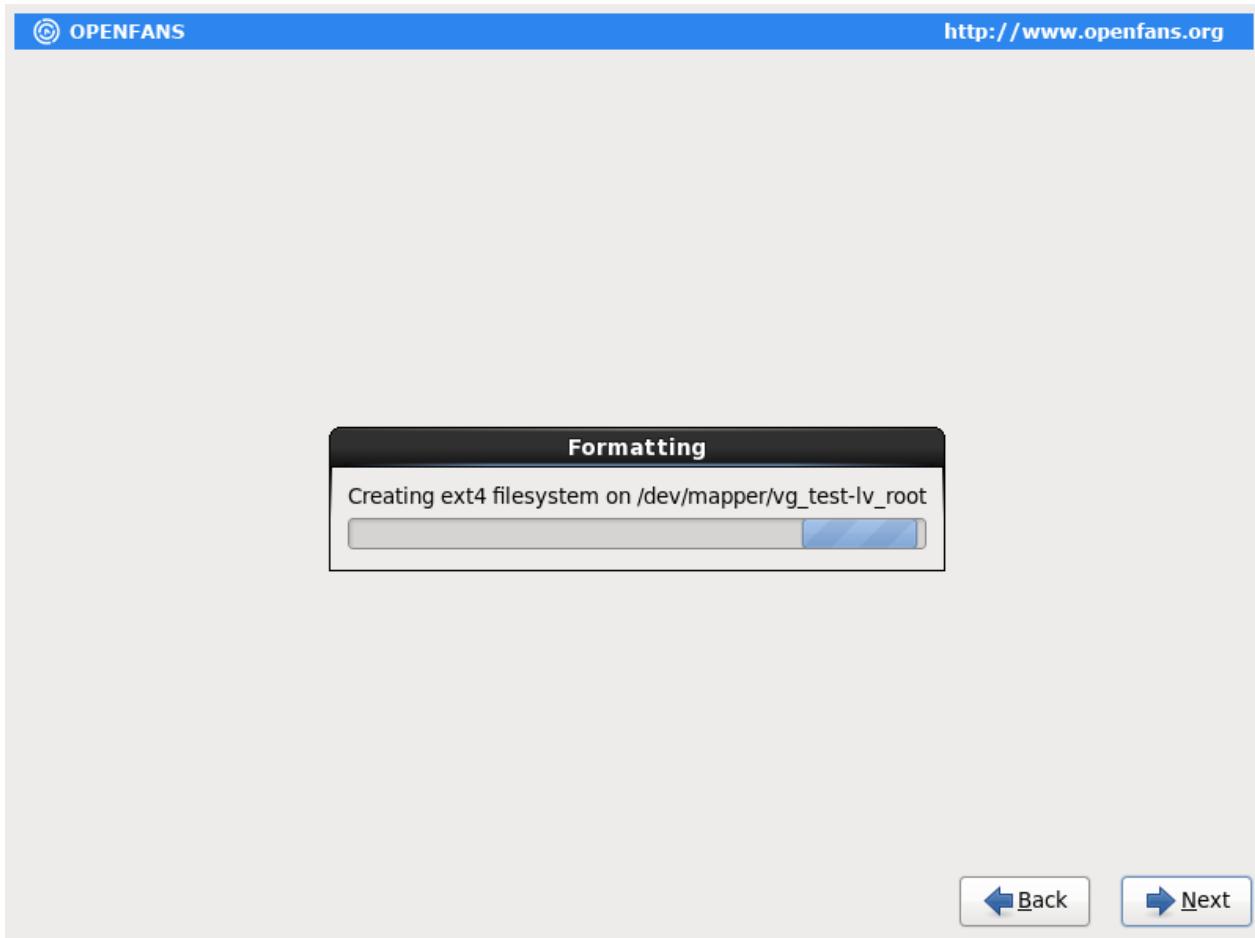
Use Free Space
Retains your current data and partitions and uses only the unpartitioned space on the selected device(s), assuming you have enough free space available.

Create Custom Layout
Manually create your own custom layout on the selected device(s) using our partitioning tool.

Encrypt system
 Review and modify partitioning layout

 Back  Next

系统自动分区并建立文件系统，完成后进入下一步。



2.11 选择安装的软件包（默认）

选择系统安装组件。

© OPENFANS

<http://www.openfans.org>

The default installation of CecOS includes a set of software applicable for general internet usage. You can optionally select a different set of software now.

Minimal

Please select any additional repositories that you want to use for software installation.

CecOS Core Installation.

You can further customize the software selection now, or after install via the software management application.

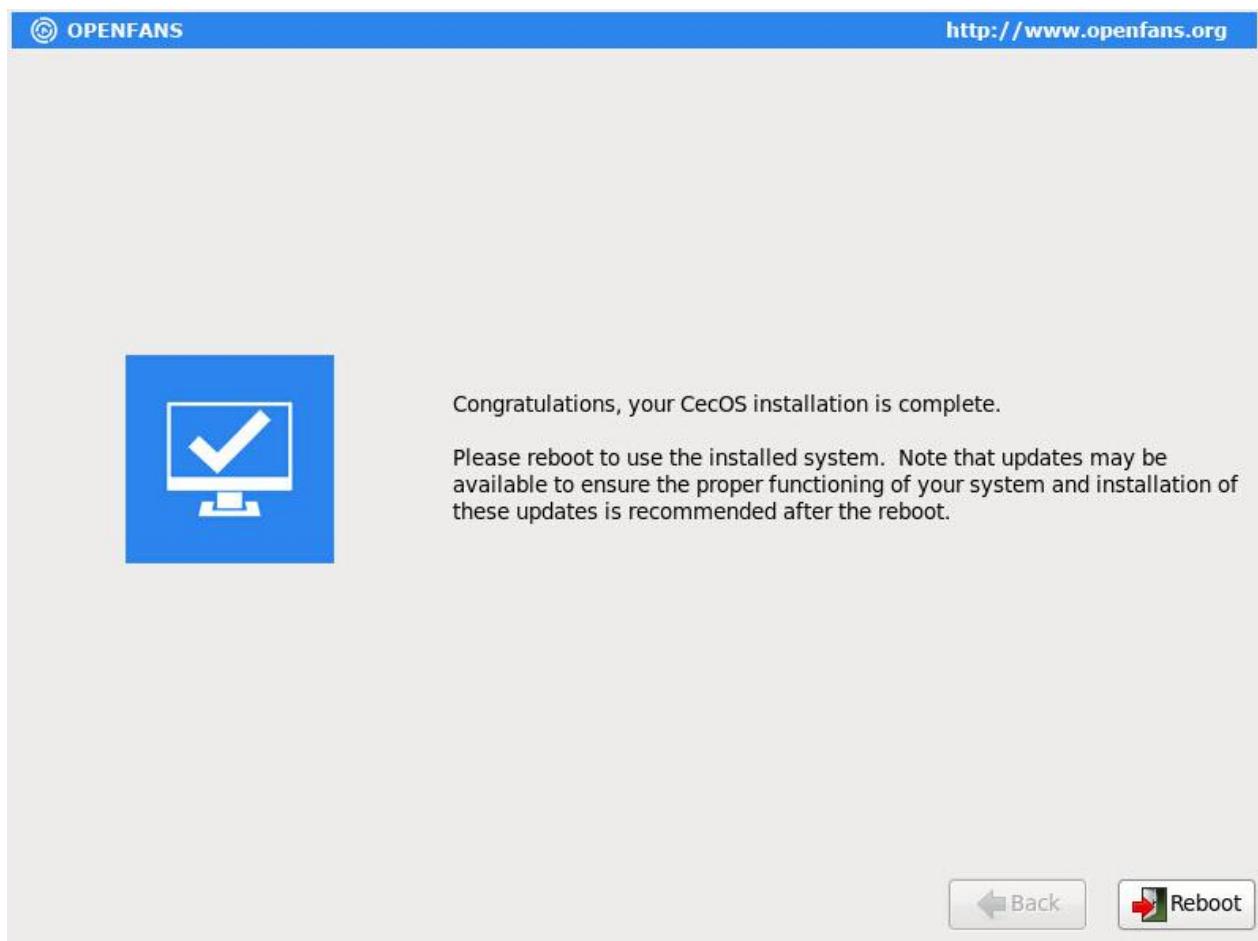
Customize later Customize now

2.12 开始安装 CecOS 系统

确认后开始安装系统。



系统安装完成，点击 Reboot，重新启动系统。



2.13 登陆界面

系统重启成功，输入用户名和密码登录系统。

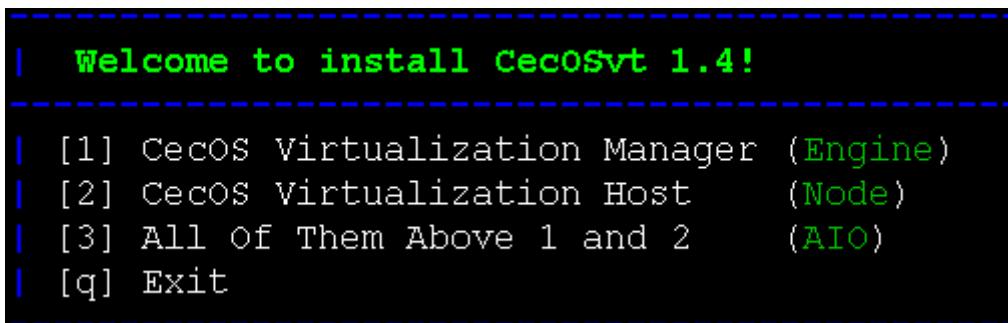
```
CecOS release 1.4 (Niu'er)
Kernel 3.14.18 on an x86_64
cecos login:
```

[点击下载](#)

=====独立安装计算节点、管理节点=====

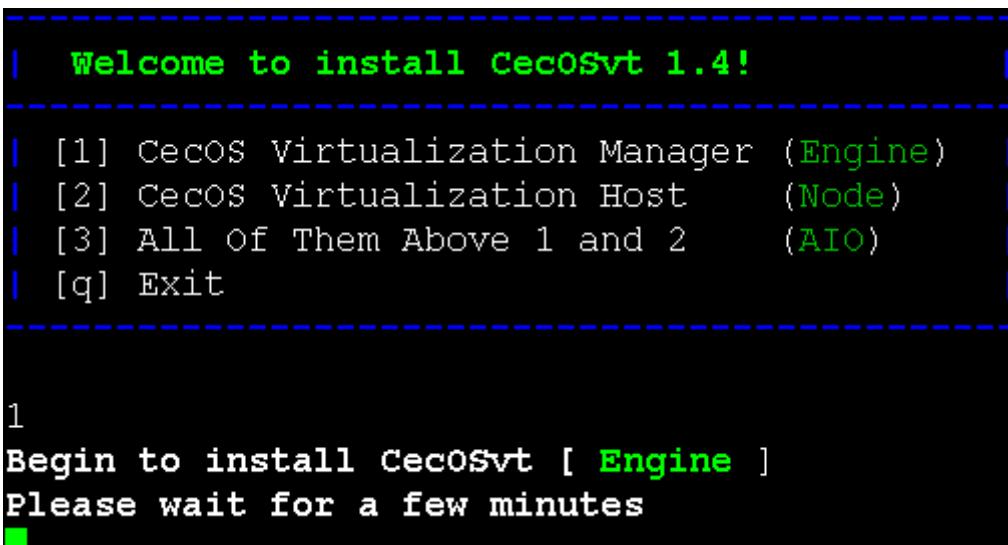
```
[root@system ~]# mount /dev/cdrom /mnt
[root@system mnt]# ./run
```

```
[root@system ~]# cecosvt-install
```



总共三个节点：

管理节点、计算节点、所有节点（计算节点+管理节点）



```
[root@system ~]# vdsm-tool configure --force
```

```
[root@system ~]# cecvm-setup
```

1. 安装 Engine 环境

准备：一台安装好 CecOS 平台的机器，物理机或虚拟机均可，设置好主机名和固定 IP，设置 DNS 或者绑定主机名。CecOSvt 镜像或光盘一份

2. 安装配置 Engine 管理节点

打开机器，进入系统，挂载 CentOSvt 镜像，打开挂载目录，执行 ./run 命令，加载 yum 源

```
CecOS release 1.4 (Niu'er)
Kernel 3.14.19 on an x86_64

cecos login: root
Password:
Last login: Thu Sep 25 16:24:03 on ttys1
[root@cecos ~]# mount /dev/dvd /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@cecos ~]# cd /mnt/
[root@cecos mnt]# ls
EULA GPL linux-guest-agent_tools Packages README RPM-GPG-KEY-OPENFANS-cecos run Script TRANS.TBL version
[root@cecos mnt]# ./run _
```

出现下图所示界面，表示 yum 源建立成功

2.1 安装 Engine 管理节点

```
CecOS release 1.4 (Niu'er)
Kernel 3.14.19 on an x86_64

cecos login: root
Password:
Last login: Thu Sep 25 16:24:03 on ttys1
[root@cecos ~]# mount /dev/dvd /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@cecos ~]# cd /mnt/
[root@cecos mnt]# ls
EULA GPL linux-guest-agent_tools Packages README RPM-GPG-KEY-OPENFANS-cecos run Script TRANS.TBL version
[root@cecos mnt]# ./run
Copy files to your system, please wait...
Loaded plugins: versionlock
CecOSvt-1.4
Metadata Cache Created
Done!
CecOSvt Local Yum Repo made!
Use command "cecosvt-install" to install CecOSvt packages.
[root@cecos mnt]# _
```

根据提示运行“cecosvt-install”命令，出现下图显示界面，选择[1]，安装 Engine 服务



现在开始安装 Engine 服务，耐心等待

```
Welcome to install CecOSvt 1.4!
[1] CecOS Virtualization Manager (Engine)
[2] CecOS Virtualization Host (Node)
[3] All Of Them Above 1 and 2 (AIO)
[q] Exit

Select installation:
1
Begin to install CecOSvt [ Engine ]
Please wait for a few minutes ...
-
```

看到下图显示内容，表示 Engine 节点已经安装完成

```
Welcome to install CecOSvt 1.4!
[1] CecOS Virtualization Manager (Engine)
[2] CecOS Virtualization Host (Node)
[3] All Of Them Above 1 and 2 (AIO)
[q] Exit

Select installation:
1
Begin to install CecOSvt [ Engine ]
Please wait for a few minutes ...
Installation completed!
Installation log: /root/cecosvt_install-140907232931246442951-I0BEFS26aAj6mUC.log
[root@cecos mnt]# -
```

2.2 配置 Engine 管理服务

接下来开始配置 Engine 服务，执行 `cecmv-setup` 命令，开始配置，首先配置报表系统，可以根据实际情况选择 yes 或 no，本文默认配置报表系统，选择 yes

```
[ INFO ] Stage: Environment setup
  Configuration files: ['/etc/ovirt-engine-setup.conf.d/10-packaging-dwh.conf', '/etc/ovirt-engine-setup.conf.d/10-packaging.conf', '/etc/ovirt-engine-setup.conf.d/20-packaging-rhevm-reports.conf']
    Log file: /var/log/ovirt-engine/setup/ovirt-engine-setup-20140907234055-gsni59.log
      Version: otopi-1.2.2 (otopi-1.2.2-1.el6ev)
[ INFO ] Stage: Environment packages setup
[ INFO ] Stage: Programs detection
[ INFO ] Stage: Environment setup
[ INFO ] Stage: Environment customization

==== PRODUCT OPTIONS ====

  Configure Reports on this host (Yes, No) [Yes]: yes
  Configure Data Warehouse on this host (Yes, No) [Yes]: yes

==== PACKAGES ====

[ INFO ] Checking for product updates...
[ INFO ] No product updates found
```

下面开始配置主机名，防火墙等，均采用默认配置即可

```
Note: automatic configuration of the firewall may overwrite current settings.
[ INFO ] Do you want Setup to configure the firewall? (Yes, No) [Yes]: [ ]
[ INFO ] iptables will be configured as firewall manager.

==== DATABASE CONFIGURATION ====

Where is the Engine database located? (Local, Remote) [Local]:
Setup can configure the local postgresql server automatically for the engine to run. This may conflict with existing applications.
Would you like Setup to automatically configure postgresql and create Engine database, or prefer to perform that manually? (Automatic, Manual) [Automatic]: [ ]
[ ] Where is the DWH database located? (Local, Remote) [Local]: [ ]
Setup can configure the local postgresql server automatically for the DWH to run. This may conflict with existing applications.
Would you like Setup to automatically configure postgresql and create DWH database, or prefer to perform that manually? (Automatic, Manual) [Automatic]: [ ]
[ ] Where is the Reports database located? (Local, Remote) [Local]:
Setup can configure the local postgresql server automatically for the Reports to run. This may conflict with existing applications.
Would you like Setup to automatically configure postgresql and create Reports database, or prefer to perform that manually? (Automatic, Manual) [Automatic]: [ ]
```

下面配置管理员密码

```
==== OVIRT ENGINE CONFIGURATION ====

Application mode (Both, Virt, Gluster) [Both]:
Default storage type: (NFS, FC, iSCSI, POSIXFS, GLUSTERFS) [NFS]:
Engine admin password: [ ]
Confirm engine admin password: [ ]
[WARNING] Password is weak: it is based on a dictionary word
Use weak password? (Yes, No) [No]: yes
```

配置 ISO 存储域和报表系统密码

```
Configure WebSocket Proxy on this machine? (Yes, No) [Yes]:
Configure an NFS share on this server to be used as an ISO Domain? (Yes, No) [Yes]:
Local ISO domain path [/var/lib/exports/iso]: [ ]
Local ISO domain ACL [0.0.0.0/0.0.0.0(rw)]: [ ]
Local ISO domain name [ISO_DOMAIN]: [ ]

==== MISC CONFIGURATION ====

Reports power users password: [ ]
Confirm Reports power users password: [ ]
```

确定以上配置是否正确，如果需要改动，输入 Cancel 取消，重新配置服务，若不改动，OK，进入下一步，开始配置系统

```

Host FQDN : cecos.test.com
NFS export ACL : 0.0.0.0/0.0.0.0(rw)
NFS mount point : /var/lib/exports/iso
Datacenter storage type : nfs
Configure local Engine database : True
Set application as default page : True
Configure Apache SSL : True
DWH installation : True
DWH database name : ovirt_engine_history
DWH database secured connection : False
DWH database host : localhost
DWH database user name : ovirt_engine_history
DWH database host name validation : False
DWH database port : 5432
Configure local DWH database : True
Reports installation : True
Reports database name : ovirt_engine_reports
Reports database secured connection : False
Reports database host : localhost
Reports database user name : ovirt_engine_reports
Reports database host name validation : False
Reports database port : 5432
Configure local Reports database : True

```

Please confirm installation settings (OK, Cancel) [OK]: _

输入 OK, 开始配置服务

Please confirm installation settings (OK, Cancel) [OK]:

```

[ INFO ] Stage: Transaction setup
[ INFO ] Stopping engine service
[ INFO ] Stopping dwh service
[ INFO ] Stopping websocket-proxy service
[ INFO ] Stage: Misc configuration
[ INFO ] Stage: Package installation
[ INFO ] Stage: Misc configuration
[ INFO ] Initializing PostgreSQL
[ INFO ] Creating PostgreSQL 'engine' database
[ INFO ] Configuring PostgreSQL
[ INFO ] Creating PostgreSQL 'ovirt_engine_history' database

```

等待服务配置完成，出现下图所示，表示 Engine 服务配置完成，这时就可以通过域名或者 IP 来访问及管理你的 Engine 服务器了

```

[ INFO ] Starting engine service
[ INFO ] Restarting httpd
[ INFO ] Restarting nfs services
[ INFO ] Starting dwh service
[ INFO ] Stage: Clean up
      Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-setup-
20140907234055-gsni59.log
[ INFO ] Generating answer file '/var/lib/ovirt-engine/setup/answers/2014090800
0452-setup.conf'
[ INFO ] Stage: Pre-termination
[ INFO ] Stage: Termination
[ INFO ] Execution of setup completed successfully
[root@cecos mnt]#

```

3.安装 NODE 计算节点

3.1 安装 NODE 环境

一台安装好 CecOS 平台的机器，物理机或虚拟机均可，设置好主机名和固定 IP，设置 DNS 或者绑定主机名。CecOSvt 镜像或光盘一张

3.2 安装 NODE 计算节点

打开机器，进入系统，挂载 CentOS 镜像，打开挂载目录，执行 sh ./run 命令，加载 yum 源

```
CecOS release 1.4 (Niu'er)
Kernel 3.14.19 on an x86_64

cecos login: root
Password:
Last login: Thu Sep 25 16:24:03 on ttym1
[root@cecos ~]# mount /dev/dvd /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@cecos ~]# cd /mnt/
[root@cecos mnt]# ls
EULA GPL linux-guest-agent_tools Packages README RPM-GPG-KEY-OPENFANS-cecos run Script TRANS.TBL version
[root@cecos mnt]# ./run _
```

出现下图所示界面，表示 yum 源建立成功

```
CecOS release 1.4 (Niu'er)
Kernel 3.14.19 on an x86_64

cecos login: root
Password:
Last login: Thu Sep 25 16:24:03 on ttym1
[root@cecos ~]# mount /dev/dvd /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@cecos ~]# cd /mnt/
[root@cecos mnt]# ls
EULA GPL linux-guest-agent_tools Packages README RPM-GPG-KEY-OPENFANS-cecos run Script TRANS.TBL version
[root@cecos mnt]# ./run
Copy files to your system, please wait...
Loaded plugins: versionlock
CecOSvt-1.4
Metadata Cache Created
Done!
CecOSvt Local Yum Repo made!
Use command "cecosvt-install" to install CecOSvt packages.
[root@cecos mnt]# _
```

根据提示运行“cecosvt-install”命令，出现下图显示界面，选择[2]，安装 NODE 服务

```
Welcome to install CecOSvt 1.4!

[1] CecOS Virtualization Manager (Engine)
[2] CecOS Virtualization Host      (Node)
[3] All Of Them Above 1 and 2    (AIO)
[q] Exit

Select installation:
2
Begin to install CecOSvt [ Node ]
Please wait for a few minutes ...
```

现在开始安装 NODE 服务，请等待，出现下图所显示内容表示 NODE 节点安装完成

```
Welcome to install CecOSvt 1.4!

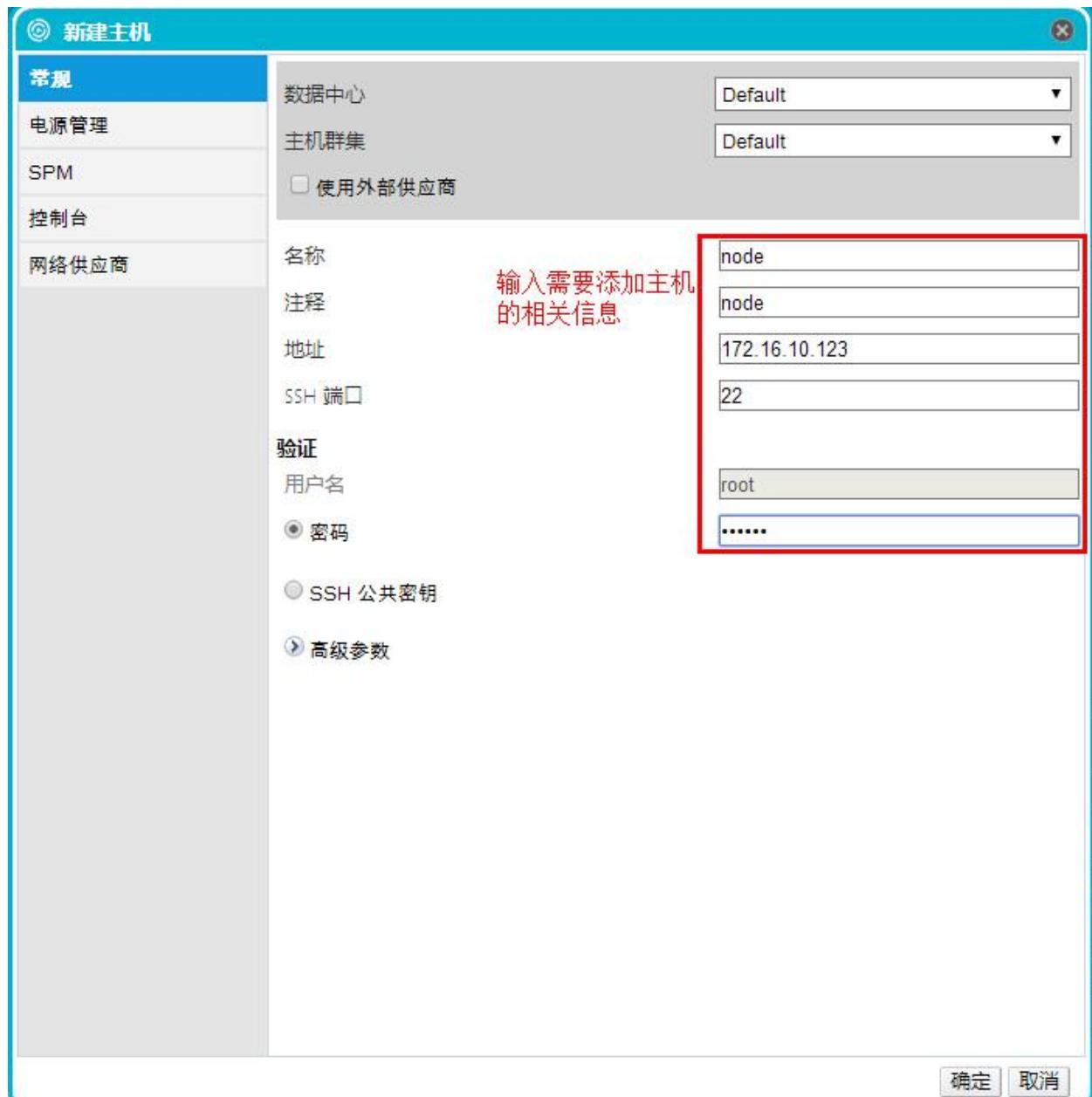
[1] CecOS Virtualization Manager (Engine)
[2] CecOS Virtualization Host      (Node)
[3] All Of Them Above 1 and 2    (AIO)
[q] Exit

Select installation:
2
Begin to install CecOSvt [ Node ]
Please wait for a few minutes ...
Installation completed!
Installation log: /root/cecosvt_install-140907232253370016520-MC2eA8oP1pYk6Fy.log
[root@cecosvt ~]#
```

现在，可以通过访问 Engine 管理界面来添加所安装好的 NODE 节点

4. Engine 添加 Node 计算结点

登录 engine 管理界面，打开主机选项卡，选择新建，打开下图所示界面。确定，开始添加主机



由于 Node 计算结点已经安装所需要的组件，管理节点会自动调用计算节点程序并添加，待出现下图所示界面，表示 Node 节点安装完成

The screenshot shows the CecOS Virtualization Platform management interface. At the top, there's a navigation bar with tabs like '数据中心', '群集', '主机', '网络', etc. Below it, a search bar says '搜索: Host:'. A main table lists two hosts: 'ClusterFS' and 'node'. Both hosts have their IP addresses (172.16.10.149 and 172.16.10.123 respectively) and names ('ClusterFS' and 'node') listed under '名称'. They are both assigned to the 'Default' cluster. The table includes columns for '状态' (Status), '虚拟机' (Virtual Machines), '内存' (Memory), 'CPU', '网络' (Network), and 'SPM'. Below the table, there's a detailed view for the 'ClusterFS' host, showing various system parameters like kernel version, CPU type, memory usage, and disk space. The 'node' host has a similar detailed view below its table entry.

=====平台使用手册 Linux 版=====

1. 登录系统管理界面

CecOS Virtualization 安装完成以后，在安装有浏览器的机器上面编辑 hosts 文件，加入 CecOS Virtualization 平台的 IP 和主机名，如：IP 地址 主机名，保存并退出。打开你的浏览器，地址栏输入设定的主机名或者 IP 地址，访问管理界面，在欢迎界面后，进入下图显示界面



点击管理用户入口，进入登录页面，如下图所示，输入用户名和密码，用户名默认为：admin



点击登录，进入管理界面，下图为进入管理界面后默认标签

A screenshot of the CecOS management interface. The top navigation bar includes tabs for "数据中心", "群集", "主机", "网络", "存储", "磁盘", "虚拟机", "池", "模板", "卷", "用户", and "事件". The "主机" tab is currently selected. On the left, a sidebar titled "系统" shows a tree view with nodes like "数据中心", "Default", "local_datacenter", and "外部供应商". The main content area displays a table with columns: "名称" (Name), "主机" (Host), "IP 地址" (IP Address), "FQDN", "群集", "数据中心", "内存", "CPU", "网络", "显示", and "状态". A message at the bottom of the table says "没有要显示的项目" (No items to display). At the bottom left, there is a "书签" (Bookmarks) section.

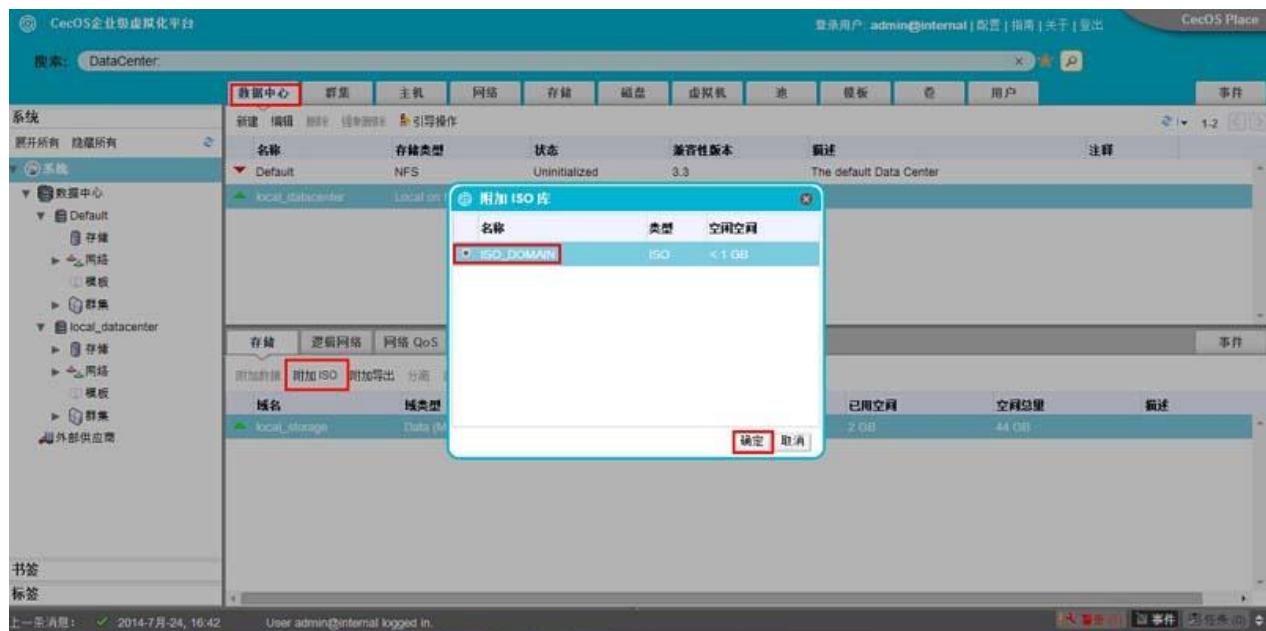
2. 查看主机状态

选中主机标签，查看主机状态是否正确，若显示为绿色三角符号，则主机添加正确，可以进行下一步操作



3. 配置数据中心

打开数据中心，选中 local_datacenter，然后选择附加 ISO，选择 ISO_DOMAIN，并确定，系统开始添加 ISO 存储域



4. 上传系统镜像

如下图所示，表示 ISO 存储域添加成功



现在可以开始上传系统镜像，安装虚拟机

镜像上传位置：

/var/lib/exports/iso/c03bf88d-4efe-4707-95c4-99b4fd77c76c/images/11111111-1111-1111-1111-111111111111 注：红色字体为系统随机产生

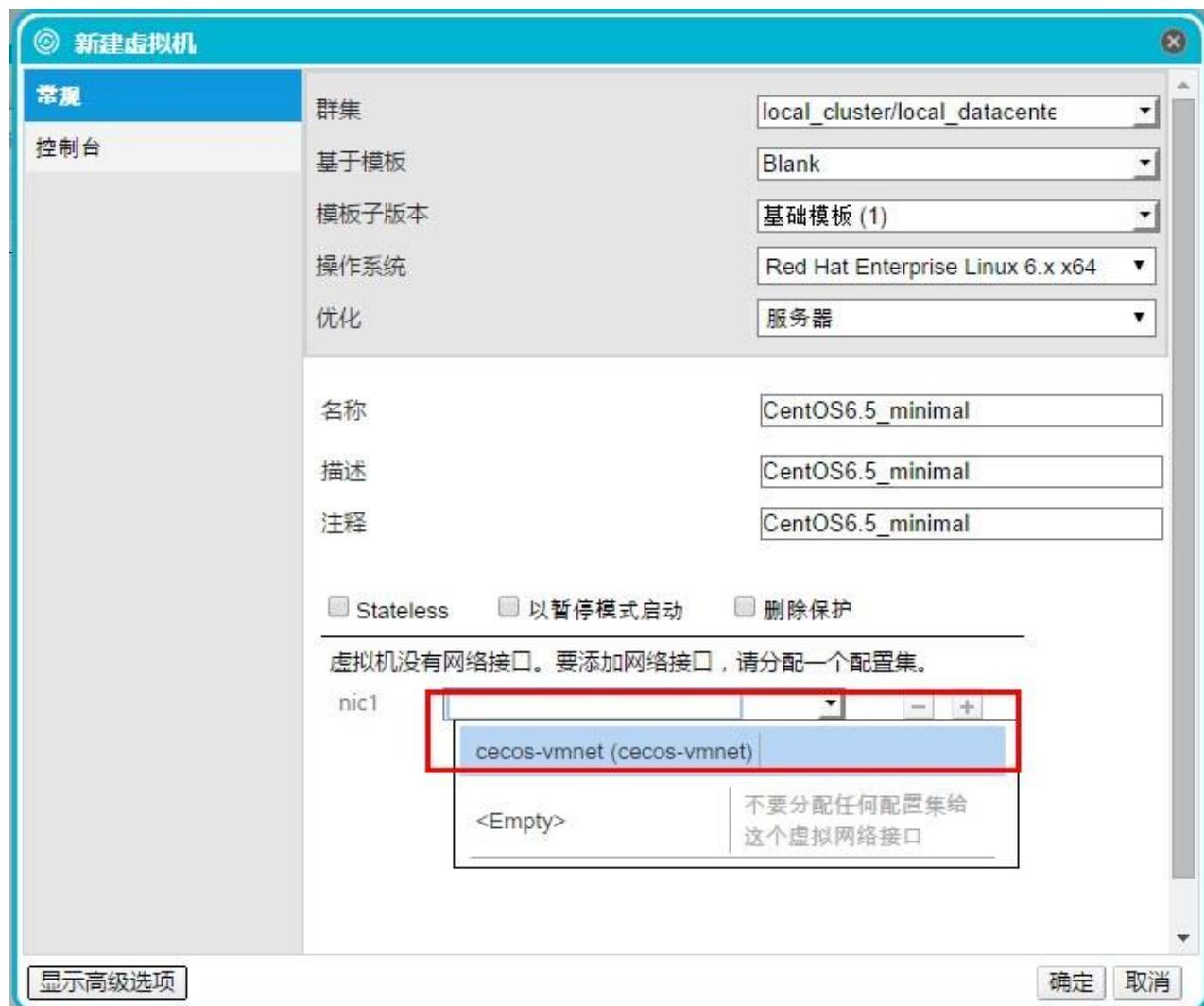
本文以 CentOS-6.4-x86_64-minimal 为例

检测镜像是否上传成功：选中 ISO_DOMAIN，点击图像，看到 CentOS 镜像，表示上传成功，现在可以安装虚拟机



5. 新建虚拟机

打开虚拟机选项卡，选择新建虚拟机，根据自己实际情况定义桌面还是服务器。本文以新建服务器为例。输入新建虚拟机名称并设置网卡



选择显示高级选项中系统选项，分配内存大小及选择操作系统类型，完成后，点击确定



下一步配置硬盘，配置磁盘，输入磁盘大小，完成后点击确定，新建虚拟机完成

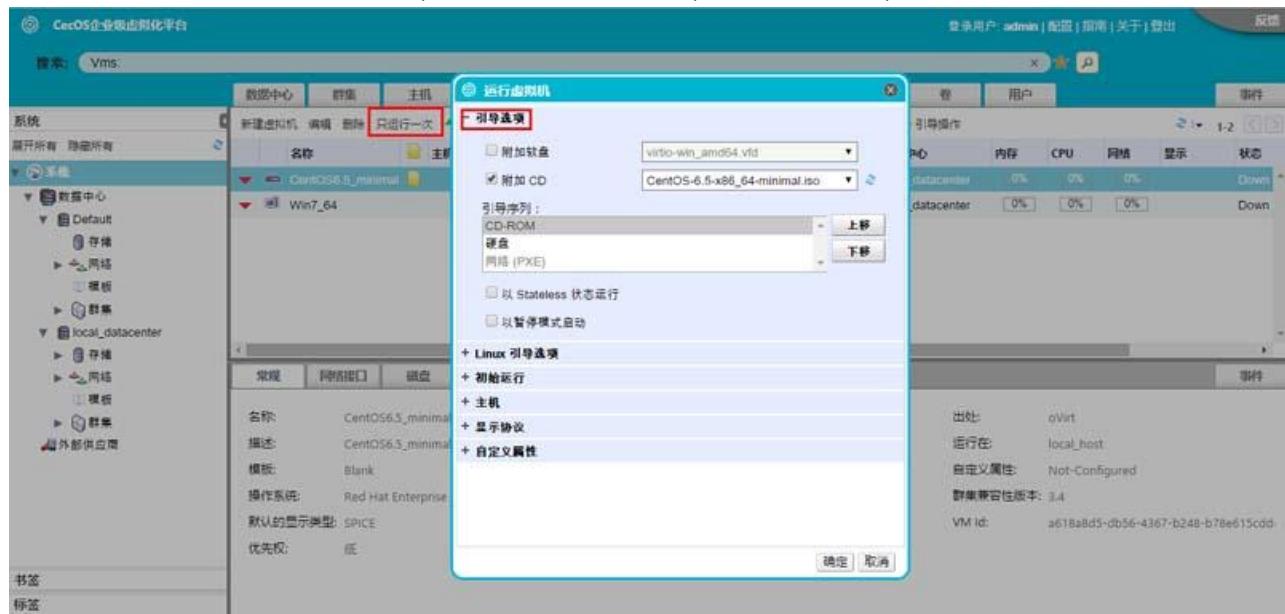


6. 安装虚拟机系统

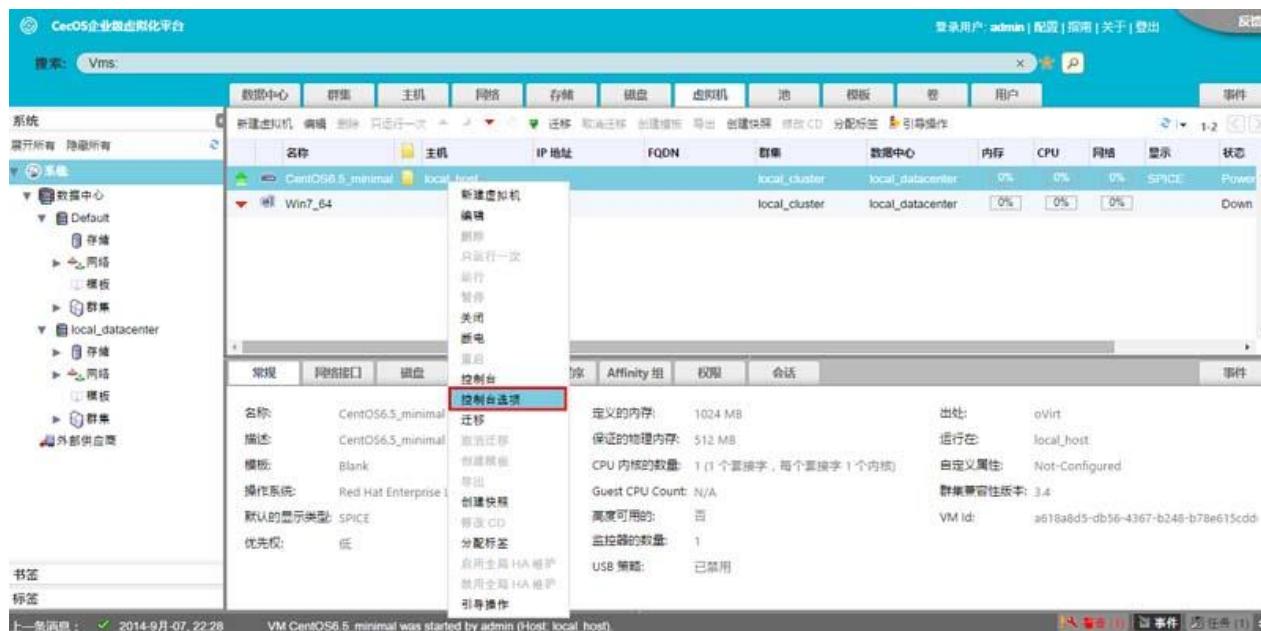
下面开始启动虚拟机，启动虚拟机之前需要安装 virt-viewer，此软件可以直接在下图所示界面资源选项中下载



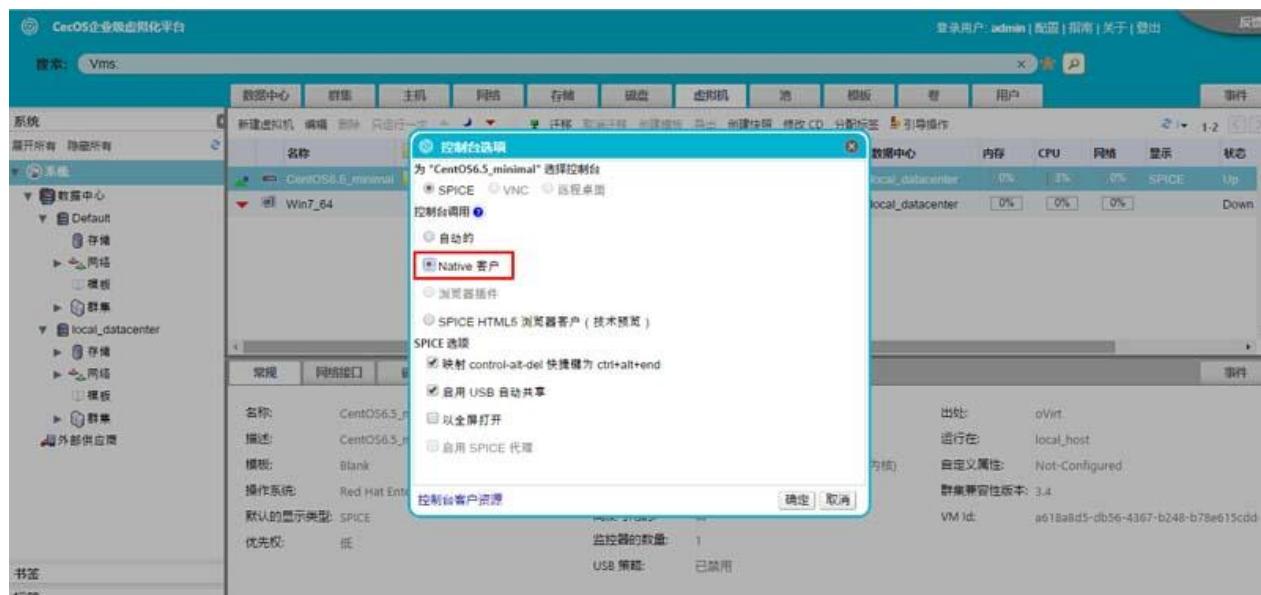
首次安装虚拟机，开机需选择只运行一次，用以选择需要安装的系统镜像，并把 CDROM 设置为第一启动盘。选中 CDROM，点击右面 UP 按钮，完成后确定，开始安装虚拟机



右击选中的虚拟机，选择控制台选项



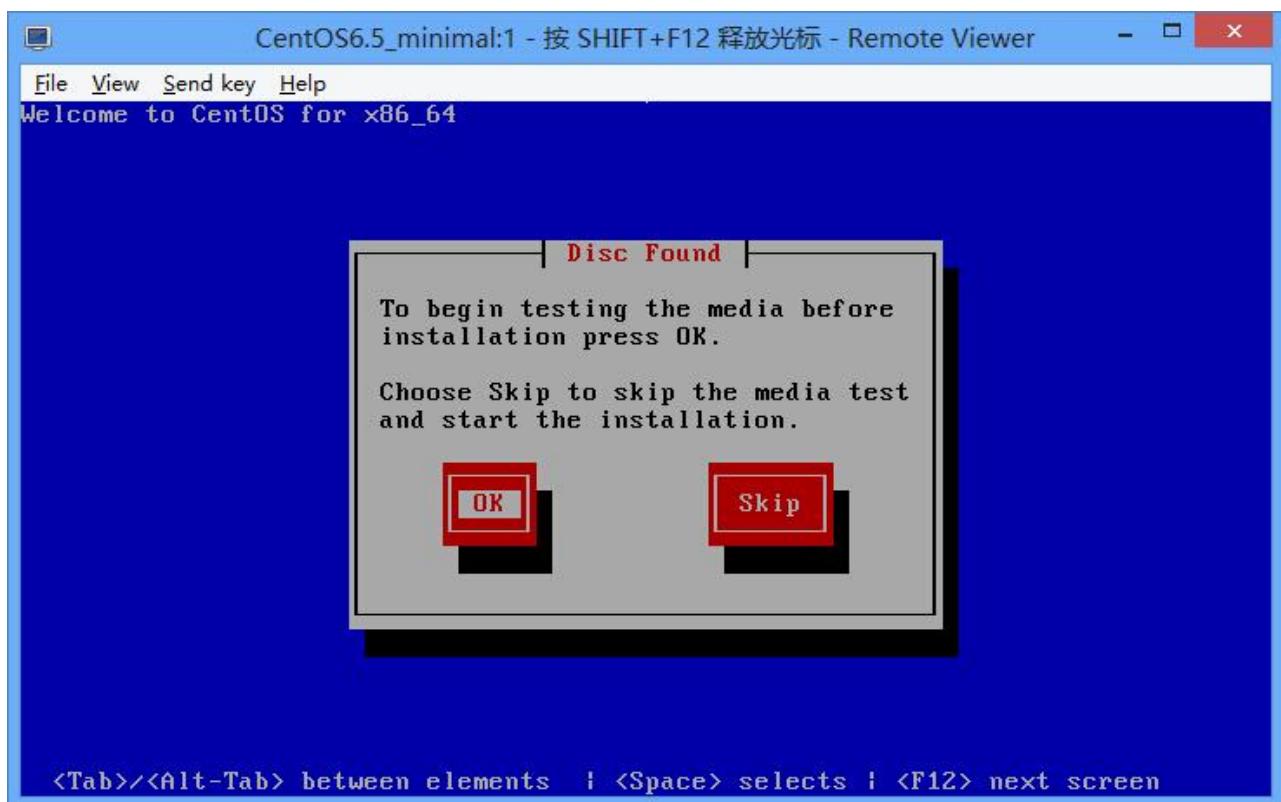
选择 Native Client 选项，确定，利用客户端打开控制台



设置完成后，点击绿色图标，浏览器会下载一个程序，点击运行，打开控制台界面



下图为控制台管理界面，可以看到虚拟机安装系统界面，现在可以像在真机安装操作系统一样来安装虚拟机的操作系统系统了



7. 操作虚拟机

系统安装完成后，可以对操作系统进行操作，如下图所示，正在执行： yum update 命令

CentOS6.5_minimal:1 - 按 SHIFT+F12 释放光标 - Remote Viewer			
(27/61): krb5-libs-1.10.3-15.el6_5.1.x86_64.rpm	761 kB	00:01	
(28/61): libblkid-2.17.2-12.el6_5.x86_64.rpm	115 kB	00:00	
(29/61): libcom_err-1.41.12-18.el6_5.1.x86_64.rpm	37 kB	00:00	
(30/61): libcurl-7.19.7-37.el6_5.3.x86_64.rpm	166 kB	00:00	
(31/61): libss-1.41.12-18.el6_5.1.x86_64.rpm	41 kB	00:00	
(32/61): libtasn1-2.3-6.el6_5.x86_64.rpm	238 kB	00:00	
(33/61): libuuid-2.17.2-12.el6_5.x86_64.rpm	68 kB	00:00	
(34/61): libxml2-2.7.6-14.el6_5.2.x86_64.rpm	800 kB	00:01	
(35/61): mdadm-3.2.6-7.el6_5.2.x86_64.rpm	337 kB	00:00	
(36/61): mysql-libs-5.1.73-3.el6_5.x86_64.rpm	1.2 MB	00:02	
(37/61): nspr-4.10.6-1.el6_5.x86_64.rpm	113 kB	00:00	
(38/61): nss-3.16.1-4.el6_5.x86_64.rpm	832 kB	00:01	
(39/61): nss-softokn-3.14.3-10.el6_5.x86_64.rpm	265 kB	00:00	
(40/61): nss-softokn-freebl-3.14.3-10.el6_5.x86_64.rpm	157 kB	00:00	
(41/61): nss-sysinit-3.16.1-4.el6_5.x86_64.rpm	41 kB	00:00	
(42/61): nss-tools-3.16.1-4.el6_5.x86_64.rpm	360 kB	00:00	
(43/61): nss-util-3.16.1-1.el6_5.x86_64.rpm	64 kB	00:00	
(44/61): openldap-2.4.23-34.el6_5.1.x86_64.rpm	265 kB	00:00	
(45/61): openssl-1.0.1e-16.el6_5.15.x86_64.rpm	1.5 MB	00:02	
(46/61): p11-kit-0.18.5-2.el6_5.2.x86_64.rpm	94 kB	00:00	
(47/61): p11-kit-trust-0.18.5-2.el6_5.2.x86_64.rpm	71 kB	00:00	
(48/61): plymouth-0.8.3-27.el6.centos.1.x86_64.rpm	89 kB	00:00	
(49/61): plymouth-core-libs-0.8.3-27.el6.centos.1.x86_64	88 kB	00:00	
(50/61): plymouth-scripts-0.8.3-27.el6.centos.1.x86_64.r	31 kB	00:00	
(51/61): postfix-2.6.6 (85%) 25% [==	1 306 kB/s	523 kB	00:05 ETA

[点击下载](#)

[点击下载](#)

=====NFS 存储=====

```
[root@system ~]# mkdir /nfs
[root@system ~]# chown -R 36:36 /nfs
[root@system ~]# vim /etc/exports
/var/lib/exports/iso    0.0.0.0/0.0.0.0(rw)
/nfs                    0.0.0.0/0.0.0.0(rw)
[root@system ~]# service nfs restart
[root@system ~]# showmount -e localhost
```

=====Linux 平台驱动安装篇=====

1. 安装环境准备

挂载 CecOSvt 镜像到 ISO 存储域的机器上面, 打开挂载路径, 进入 linux-guest-agent_tools 目录, 拷贝 GuestAgentTools-1.4.iso 到

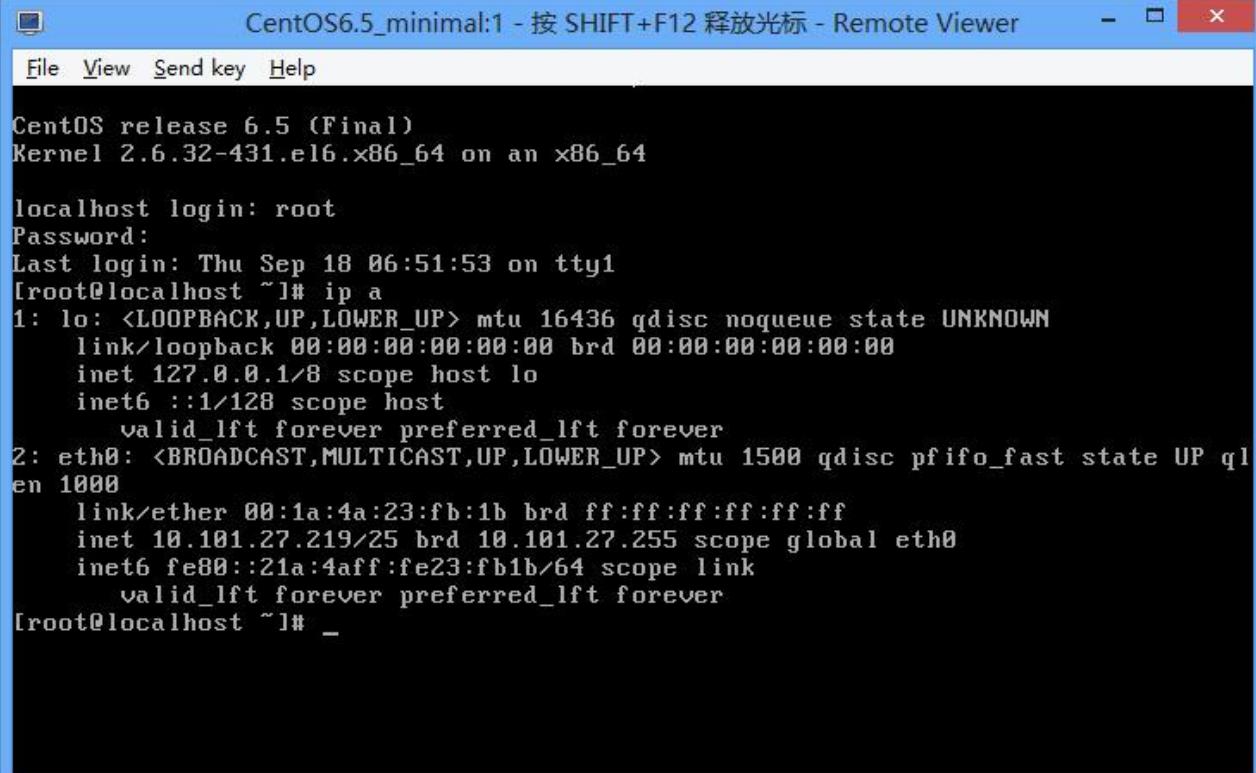
/var/lib/exports/iso/c03bf88d-4efe-4707-95c4-99b4fd77c76c/images/11111111-1111-1111-1111-111111111111 目录注: 红色字体部分为系统随机产生

2. 新建虚拟机

新建虚拟机及第一次安装虚拟机系统请参考 CecOS 虚拟化平台使用指南, 建立完成后, 启动虚拟机。

3. 安装 Linux 操作系统

安装系统按照系统正常安装步骤进行, 直到系统安装完成

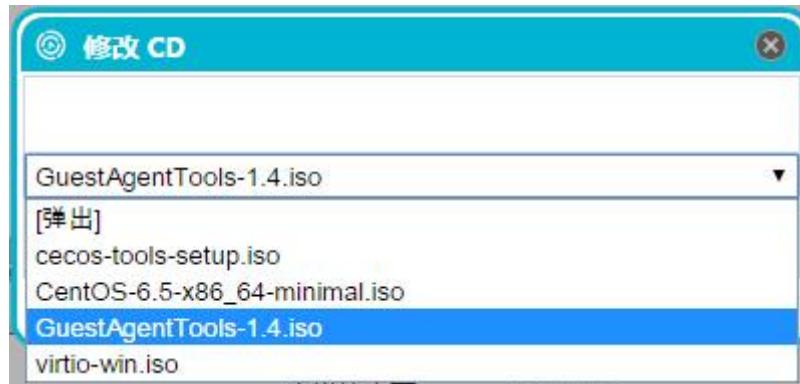


```
CentOS release 6.5 (Final)
Kernel 2.6.32-431.el6.x86_64 on an x86_64

localhost login: root
Password:
Last login: Thu Sep 18 06:51:53 on tty1
[root@localhost ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:1a:4a:23:fb:1b brd ff:ff:ff:ff:ff:ff
    inet 10.101.27.219/25 brd 10.101.27.255 scope global eth0
        inet6 fe80::21a:4aff:fe23:fb1b/64 scope link
            valid_lft forever preferred_lft forever
[root@localhost ~]# _
```

4. 安装 Linux 驱动

进入到 Web 管理界面，打开虚拟机选项，选择修改 CD，选择 GuestAgentTools-1.4.iso，确定



现在进入 Linux 系统，挂载 ISO 镜像并根据自己所安装的系统类型选择安装对应的软件包，本文档以 CentOS 为例，因为 CentOS 与红帽同源，所以选择 RHEL

```
[root@localhost ~]# mount /dev/dvd /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@localhost ~]# ls
anaconda-ks.cfg  install.log  install.log.syslog
[root@localhost ~]# cd /mnt/
[root@localhost mnt]# ls
linux-guest-agent_tools  TRANS.TBL
[root@localhost mnt]# cd linux-guest-agent_tools/
[root@localhost linux-guest-agent_tools]# ls
Debian  Fedora  OpenSUSE  RHEL  TRANS.TBL  Ubuntu
[root@localhost linux-guest-agent_tools]# _
```

```
[root@localhost linux-guest-agent_tools]# cd RHEL/
[root@localhost RHEL]# ls
cloud-init-0.7.2-2.el6.noarch.rpm
repodata
rhev-agent-2.3.16-7.el6_2.x86_64.rpm
rhev-agent-gdm-plugin-rhevcred-2.3.16-7.el6_2.x86_64.rpm
rhev-agent-kdm-plugin-rhevcred-2.3.16-7.el6_2.x86_64.rpm
rhev-agent-pam-rhev-cred-2.3.16-7.el6_2.x86_64.rpm
rhevm-guest-agent-1.0.5-13.el6ev.x86_64.rpm
rhevm-guest-agent-common-1.0.9-5.el6ev.noarch.rpm
rhevm-guest-agent-gdm-plugin-1.0.9-5.el6ev.x86_64.rpm
rhevm-guest-agent-kdm-plugin-1.0.9-5.el6ev.x86_64.rpm
rhevm-guest-agent-pam-module-1.0.9-5.el6ev.x86_64.rpm
TRANS.TBL
[root@localhost RHEL]# yum localinstall rhevm-guest-agent-
rhevm-guest-agent-1.0.5-13.el6ev.x86_64.rpm
rhevm-guest-agent-common-1.0.9-5.el6ev.noarch.rpm
rhevm-guest-agent-gdm-plugin-1.0.9-5.el6ev.x86_64.rpm
rhevm-guest-agent-kdm-plugin-1.0.9-5.el6ev.x86_64.rpm
rhevm-guest-agent-pam-module-1.0.9-5.el6ev.x86_64.rpm
[root@localhost RHEL]# yum localinstall rhevm-guest-agent-1.0.5-13.el6ev.x86_64.
rpm _
```

安装完成后启动 ovirt-guest-agent 服务并设置开机启动

```
[root@localhost ~]# /etc/init.d/ovirt-guest-agent start
Starting ovirt-guest-agent: [ OK ]
[root@localhost ~]# chkconfig ovirt-guest-agent on
[root@localhost ~]# 
```

进入 Engine 的 Web 管理界面，进入虚拟机页面，出现下图所示内容，表明安装完成

The screenshot shows the CecOS virtualization platform interface. On the left, there's a navigation tree with categories like 系统, 数据中心, 群集, 主机, 网络, 存储, 磁盘, 虚拟机, 池, 模板, 卷, 用户, and 事件. The main area displays a table of virtual machines. One row is highlighted with a red box, showing the IP address as 10.101.27.218. Below the table, there are tabs for 常规, 网络接口, 磁盘, 快照, 应用程序 (which is selected and highlighted with a red box), Affinity 相, 权限, and 会话. Under the 应用程序 tab, it lists installed applications: kernel-2.6.32-431.el6, rhev-agent-1.0.5-13.el6ev, and rhevm-guest-agent-1.0.5-13.el6ev.

[点击下载](#)

Windows 驱动安装篇

1. 新建虚拟机

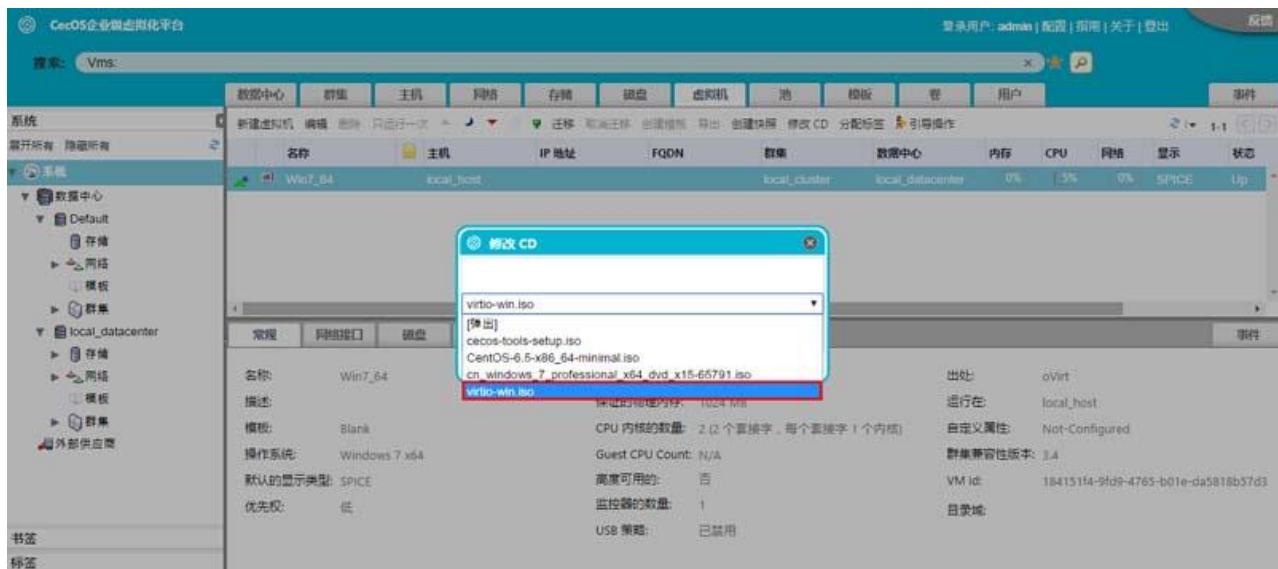
新建虚拟机及第一次安装虚拟机系统请参考 CecOS 虚拟化平台使用指南，建立完成后，启动虚拟机

2. 安装 Windows 操作系统

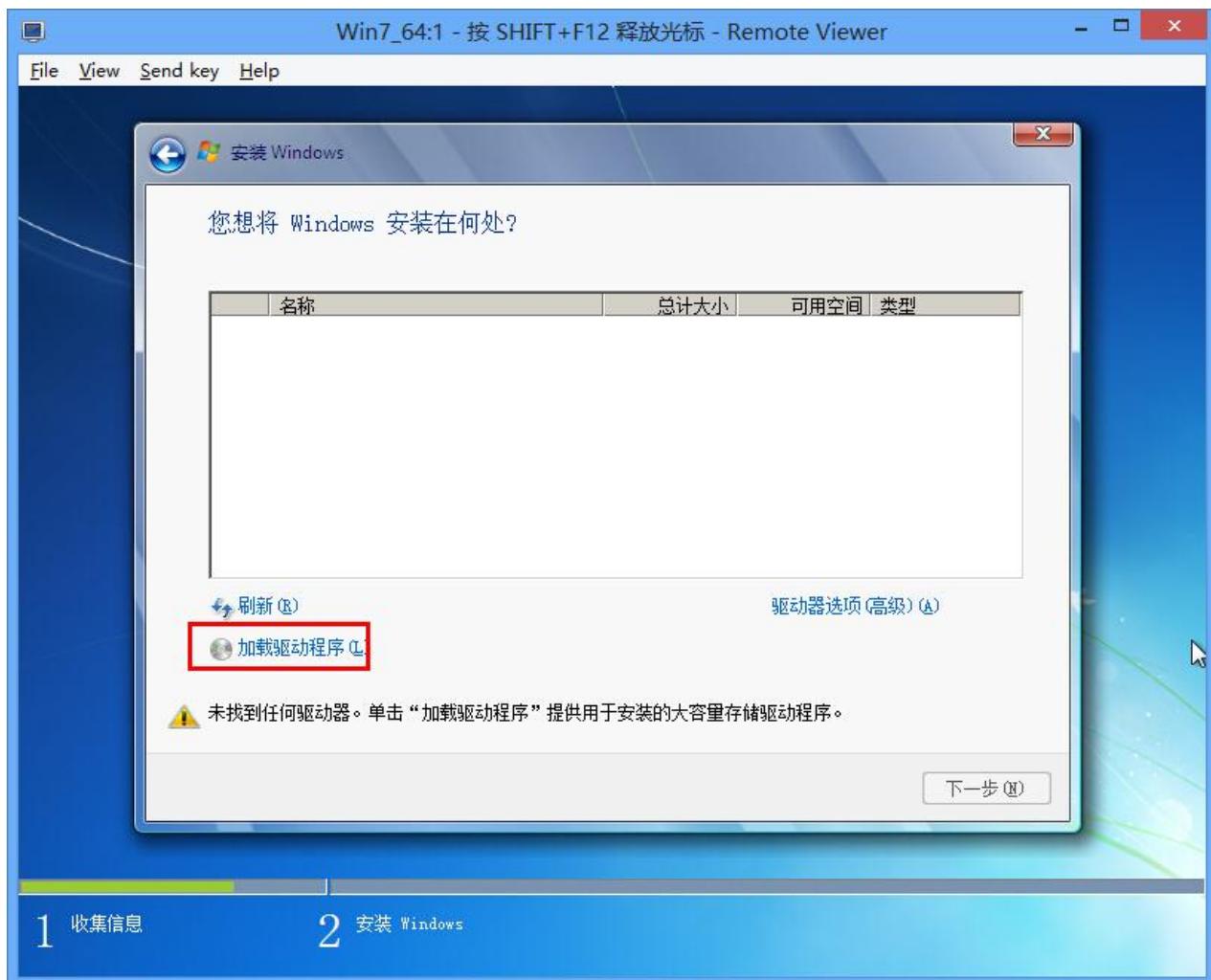
安装系统按照系统正常安装步骤进行，直到出现下图所示内容

The screenshot shows two windows side-by-side. On the left is the "Install Windows" setup window, specifically the "Where do you want to install Windows?" step. It asks "你想将 Windows 安装在何处？" and shows a table with columns: 名称, 总计大小, 可用空间, and 类型. A note at the bottom says "未找到任何启动器。单击“加载驱动程序”提供用于安装的大容量存储驱动程序。". On the right is the CecOS management interface showing the virtual machine configuration. The "修改 CD" button is highlighted with a red box. The configuration details shown include: 出处: oVirt, 运行在: local_host, 自定义属性: Not-Configured, 群集兼容性版本: 3.4, VIM id: 184151f4-9fd9-4765-b01e-da5818b57d3, and 目录域: .

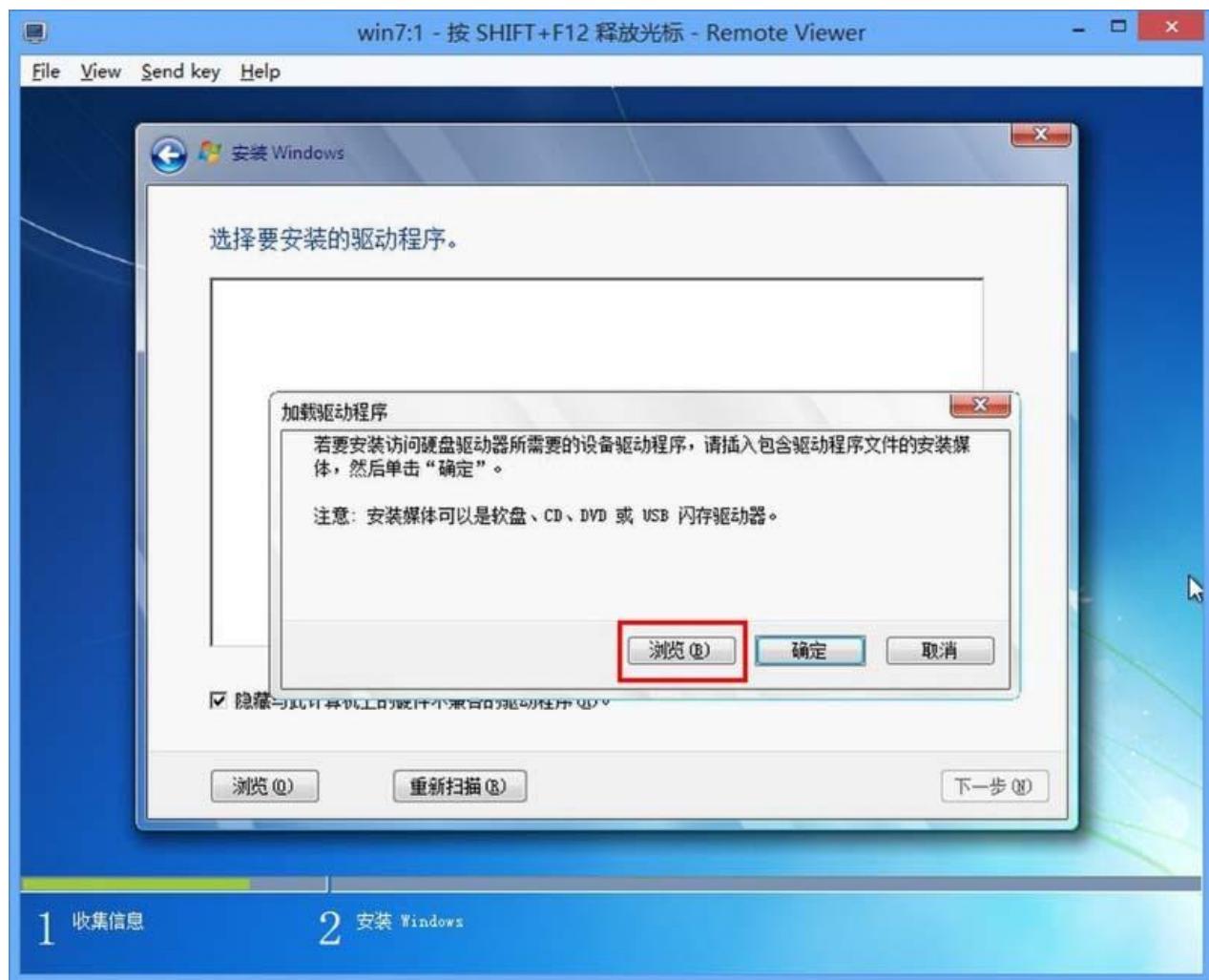
这一步需要安装虚拟磁盘驱动，点击修改 CD 按钮，选择驱动光盘，确定



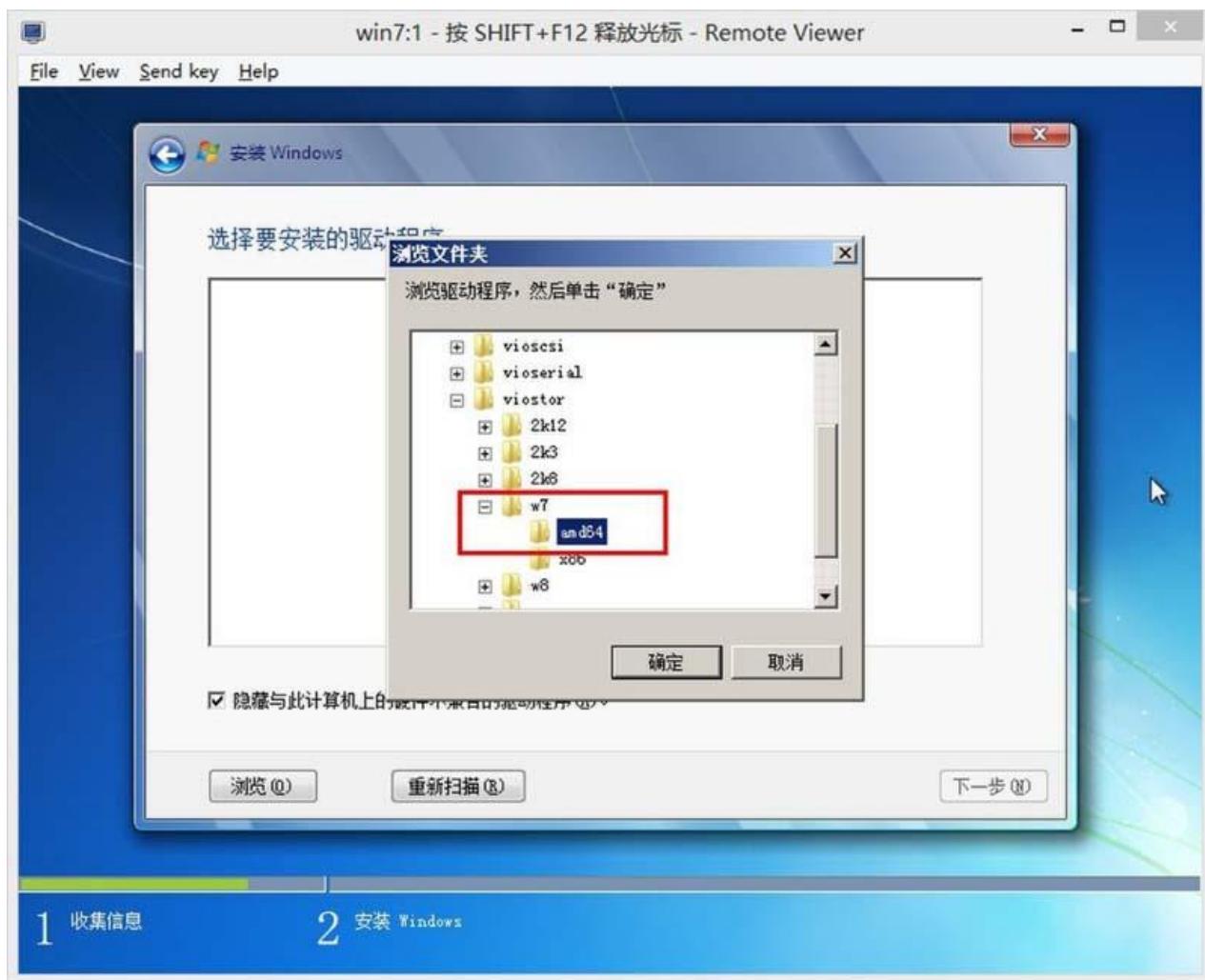
回到系统安装界面。选择加载驱动程序



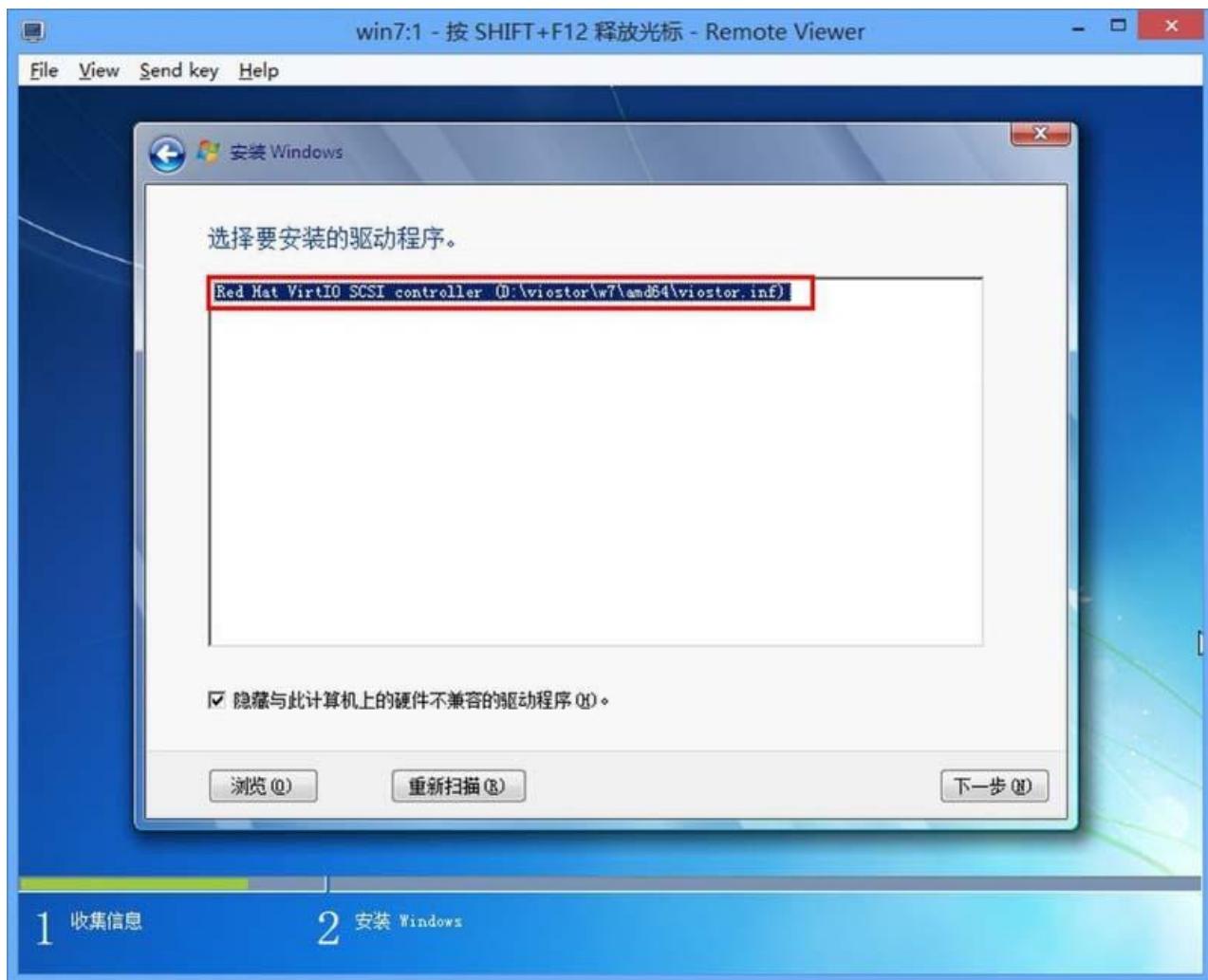
弹出对话框，选择浏览



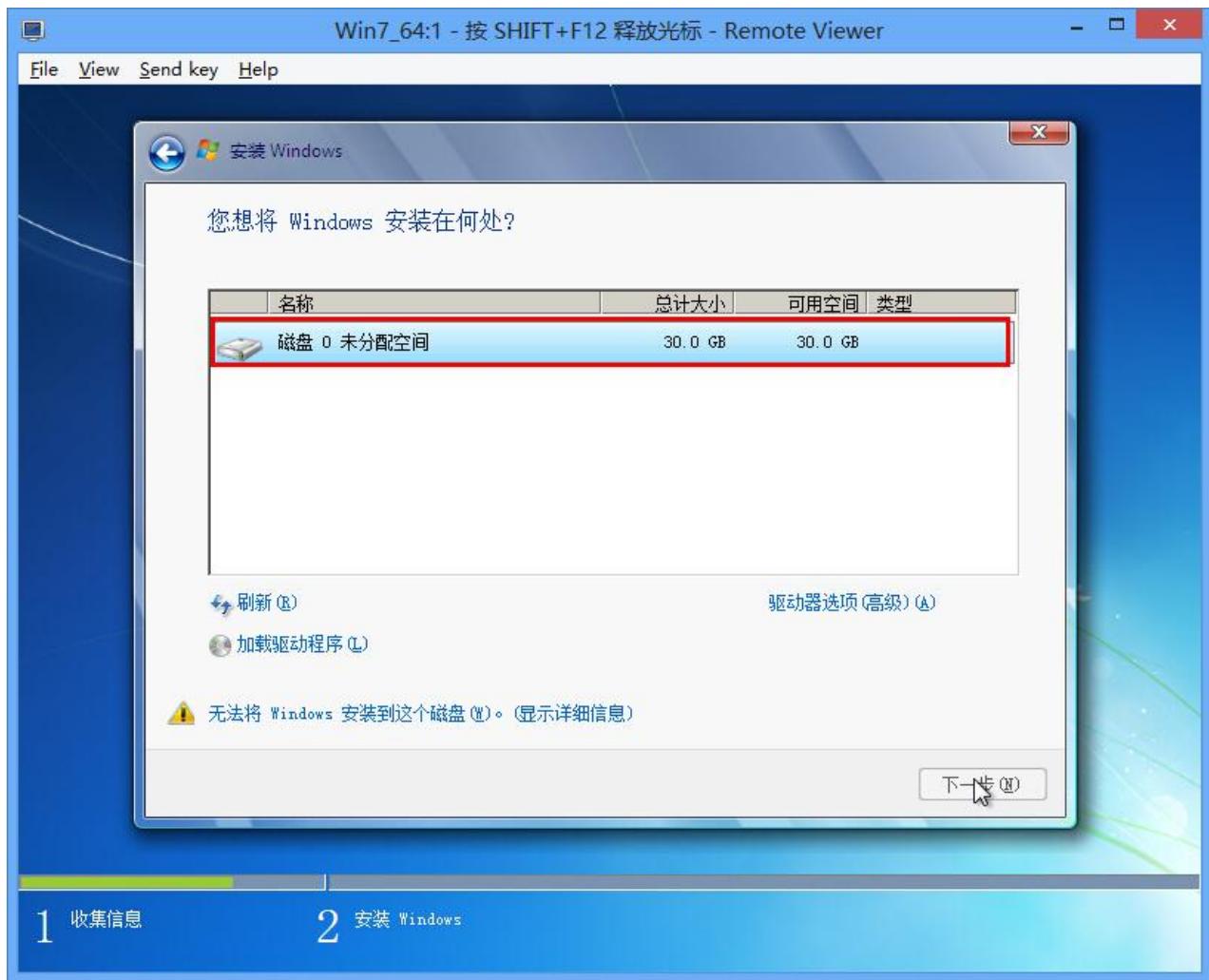
打开对话框，选择 Viostor->W7->amd64，具体驱动请根据实际安装的操作系统选择，点击确定



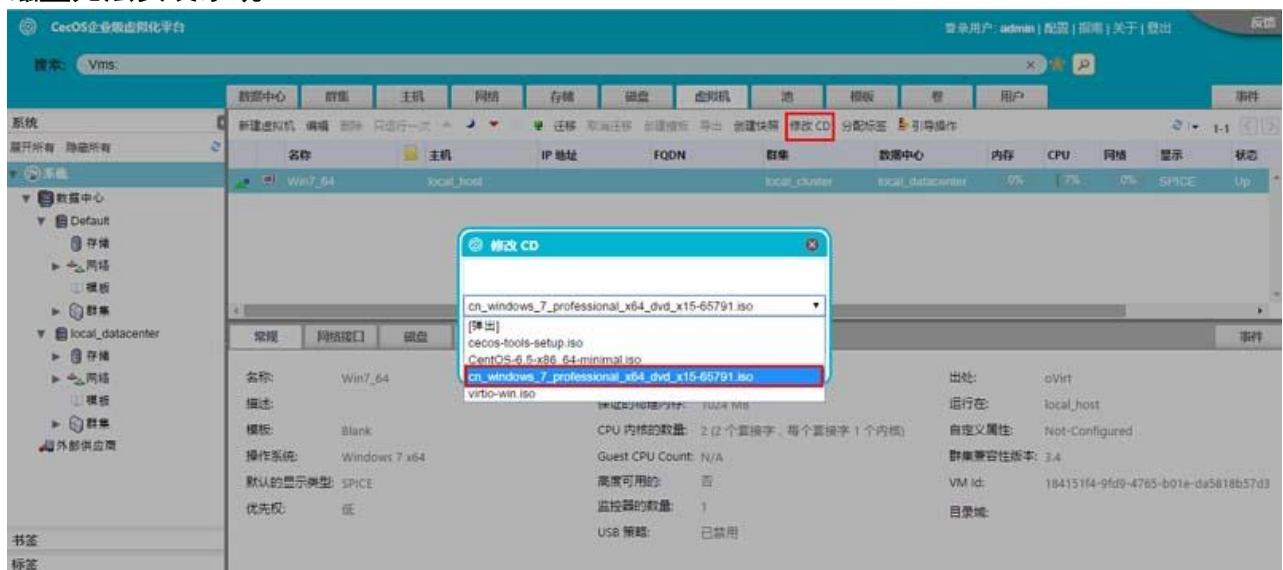
可以看到加载进去的驱动，然后点击下一步



开始搜索磁盘并发现磁盘



发现磁盘后必须点击修改 CD，把驱动光盘弹出并重新加载 Windows 系统光盘，否则会提示磁盘无法安装系统

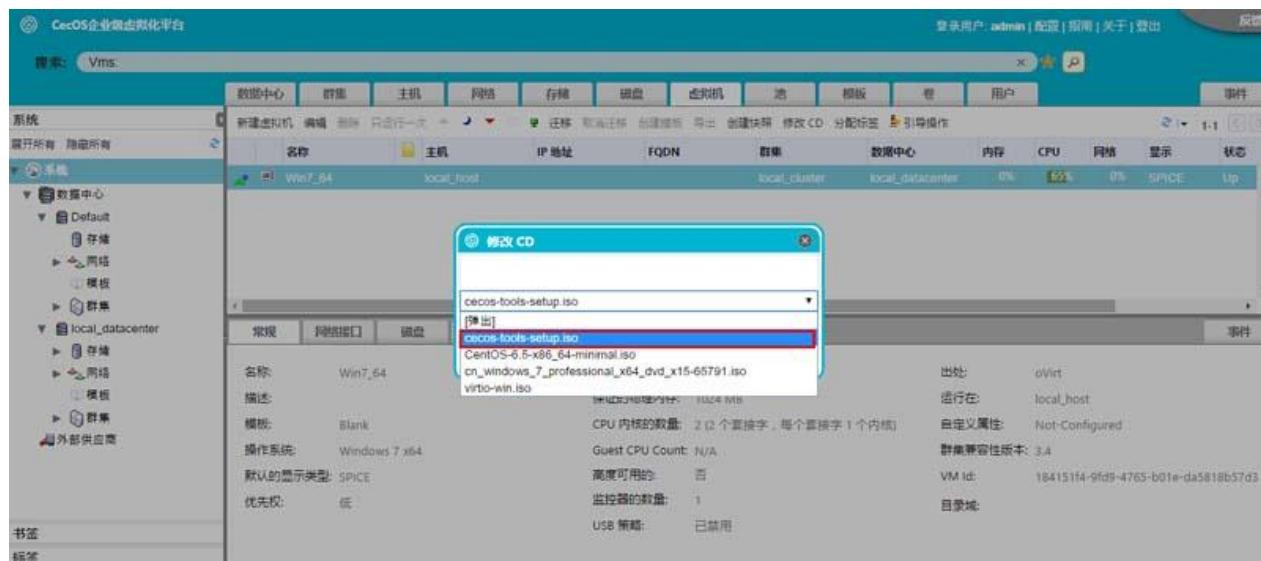


重新加载系统光盘以后，按照 Windows 系统正常安装步骤进行，直到系统安装完成并重启

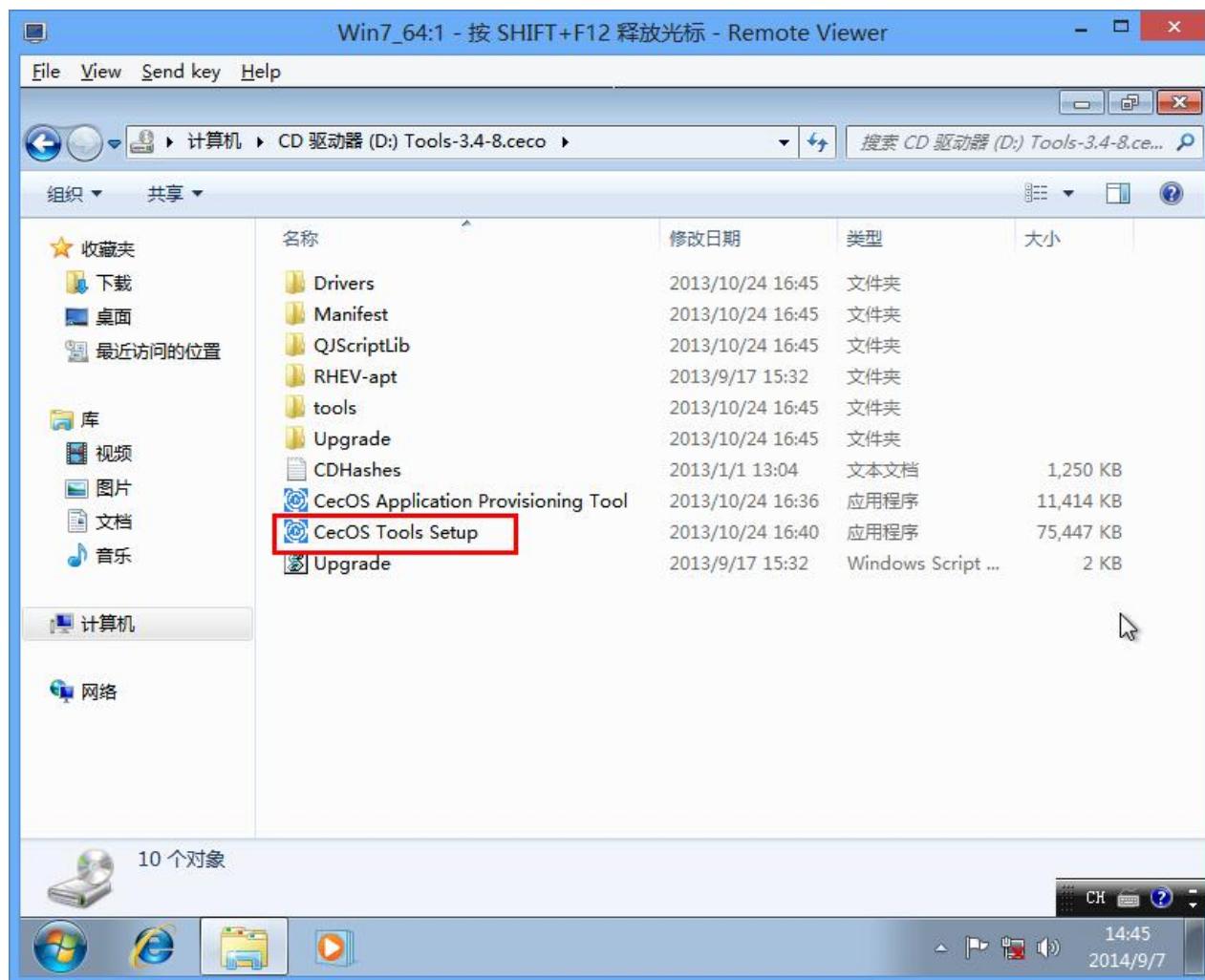
3. 安装驱动程序

重启系统以后，开始安装程序及驱动

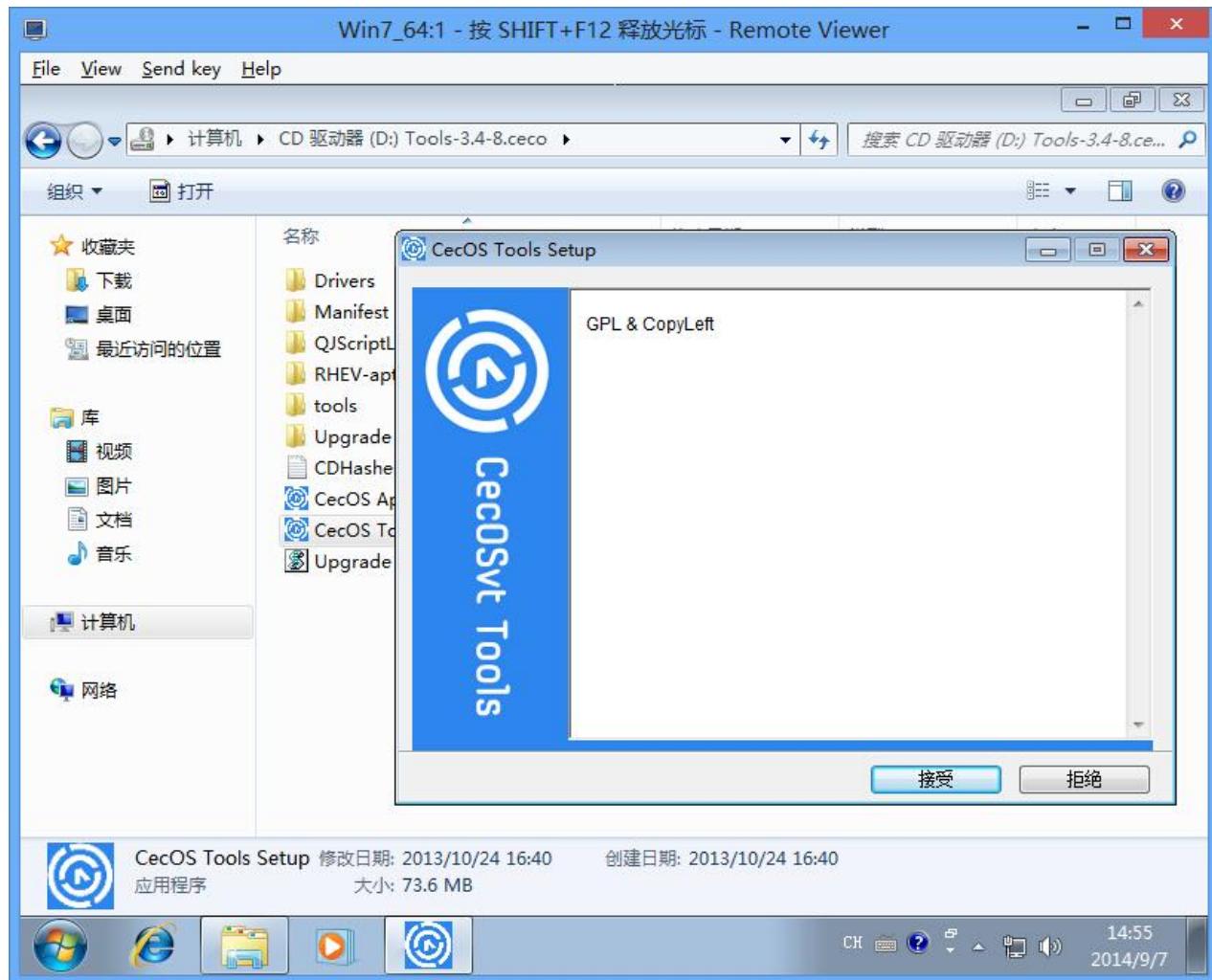
点击修改 CD，选中工具盘，确定，加载光盘镜像



加载驱动光盘完成以后，进入虚拟机，打开光驱，选择 CecOS Tools Setup 进行安装

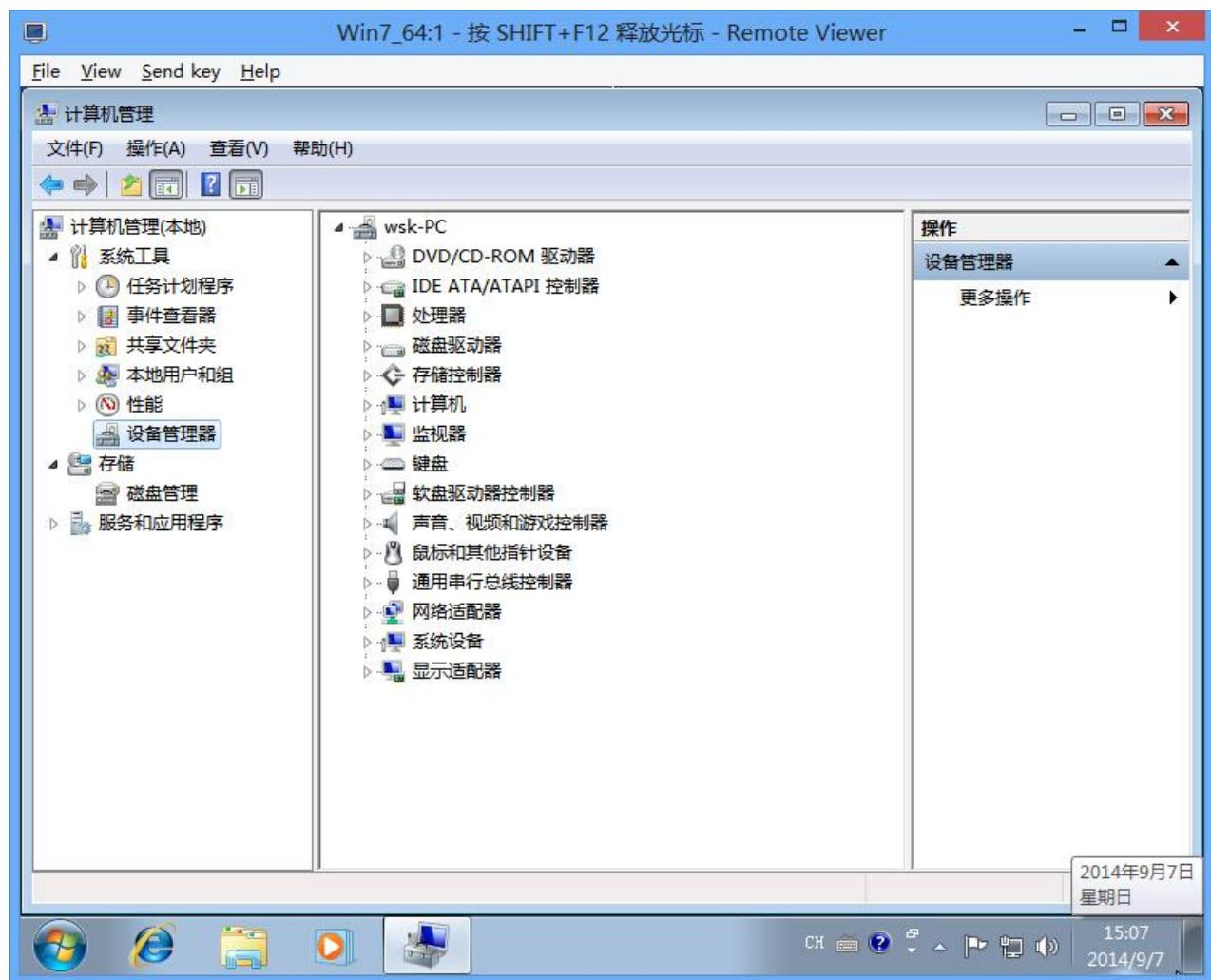


安装完成以后，开始安装驱动程序，注意，所有的驱动都必须安装



驱动程序安装完成后，重新启动虚拟机

打开设备管理器，查看驱动安装情况。如下图所示，驱动安装完成，至此，Windows 虚拟机安装完成



点击下载

=====NFS 存储=====

ecOS Virtualization 有多种存储方式，下面介绍如何使用 NFS 做存储节点。

注意事项：All in one 安装方式无法直接使用 NFS 存储方式。

1. NFS 存储配置使用环境

在 Engine 节点上面配置 NFS 服务

新建一个目录，并设置所属用户和组为 VDSM, KVM

```
[root@cecos ~]# mkdir nfs
[root@cecos ~]# chown -R 36:36 nfs/
[root@cecos ~]# vi /etc/exports
```

编辑/etc/export 文件，加入新建目录，保存并退出。

```
/var/lib/exports/iso 0.0.0.0/0.0.0.0(rw)
/nfs 0.0.0.0/0.0.0.0(rw)
```

启动 NFS 服务，执行 `showmount -e`，检查 NFS 服务是否配置成功，出现下图所示，表示 NFS 服务配置成功。

```
[root@cecos ~]# service nfs restart
关闭 NFS 守护进程: [确定]
关闭 NFS mountd: [确定]
关闭 NFS 服务: [确定]
启动 NFS 服务: exportfs: Failed to stat /nfs: No such file or directory

启动 NFS mountd: [确定]
正在启动 RPC idmapd: [确定]
正在启动 RPC idmapd: [确定]
启动 NFS 守护进程: [确定]

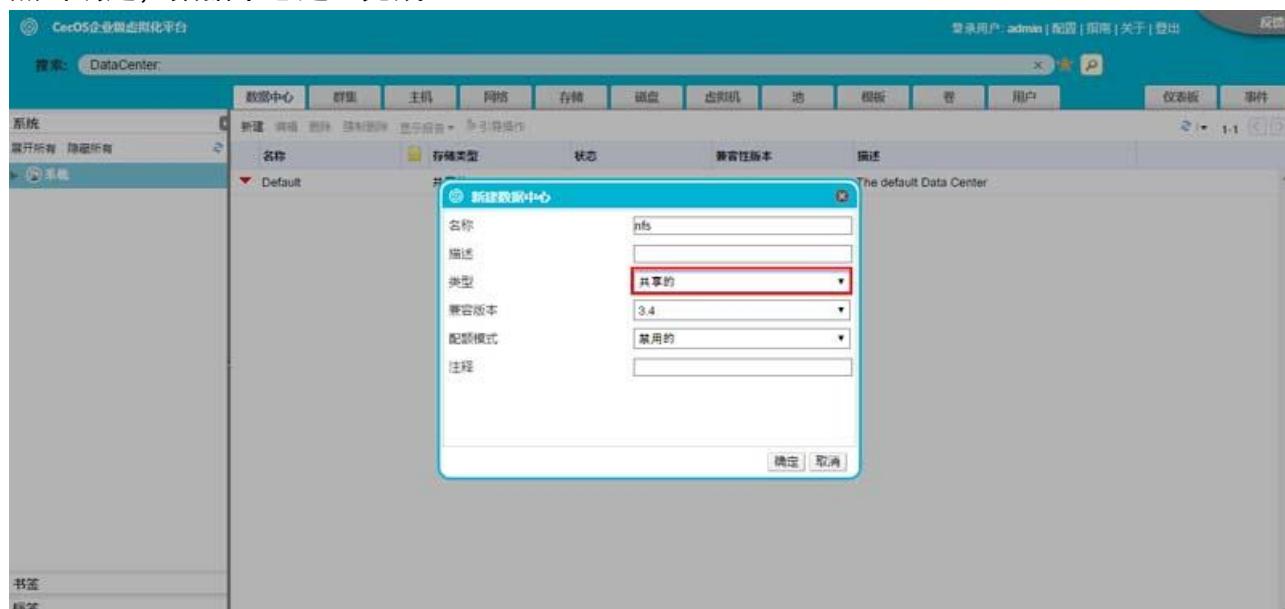
[root@cecos ~]# showmount -e
Export list for cecos.test.com:
/nfs 0.0.0.0/0.0.0.0
/var/lib/exports/iso 0.0.0.0/0.0.0.0
[root@cecos ~]#
```

2. 新建数据中心

进入数据中心管理界面，新建数据中心，打开下图界面，输入数据中心名称，选择存储类型。

注意：存储类型选择为共享的。

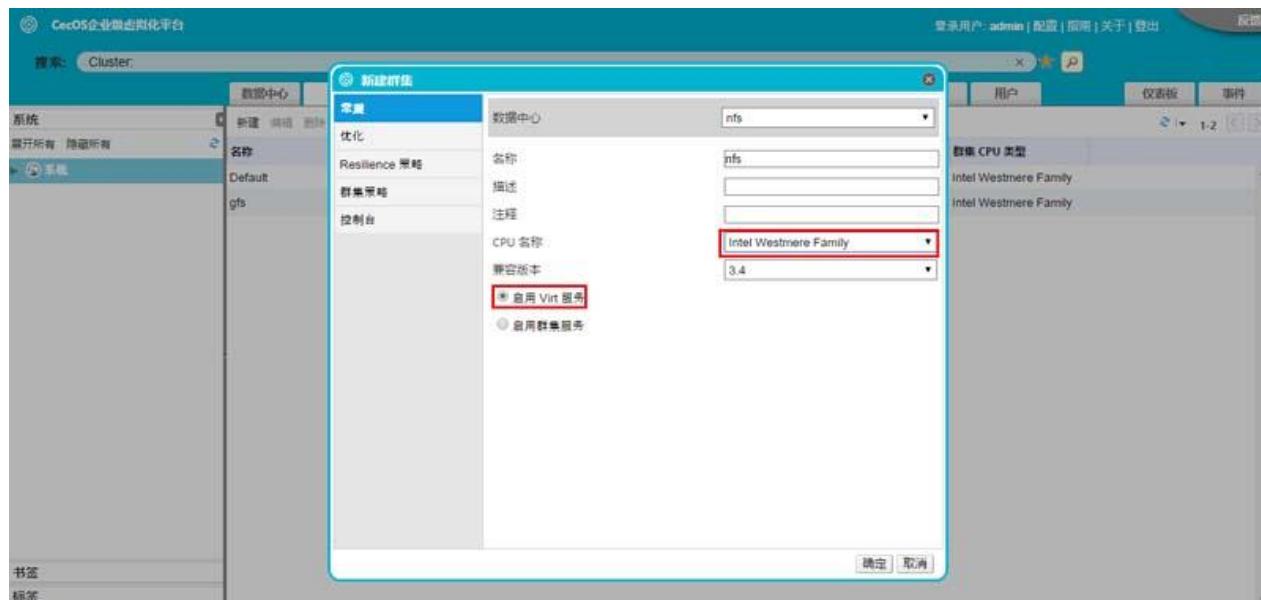
点击确定，数据中心建立完成。



3. 新建群集

打开群集选项，新建群集，选择数据中心，输入群集名称，设置 CPU 类型

注意：CPU 类型要和主机服务器 CPU 类型保持一致，并启用 virt 服务。



打开优化选项，可以看到内存优化 CPU 线程两个选项，根据实际需要进行设置。



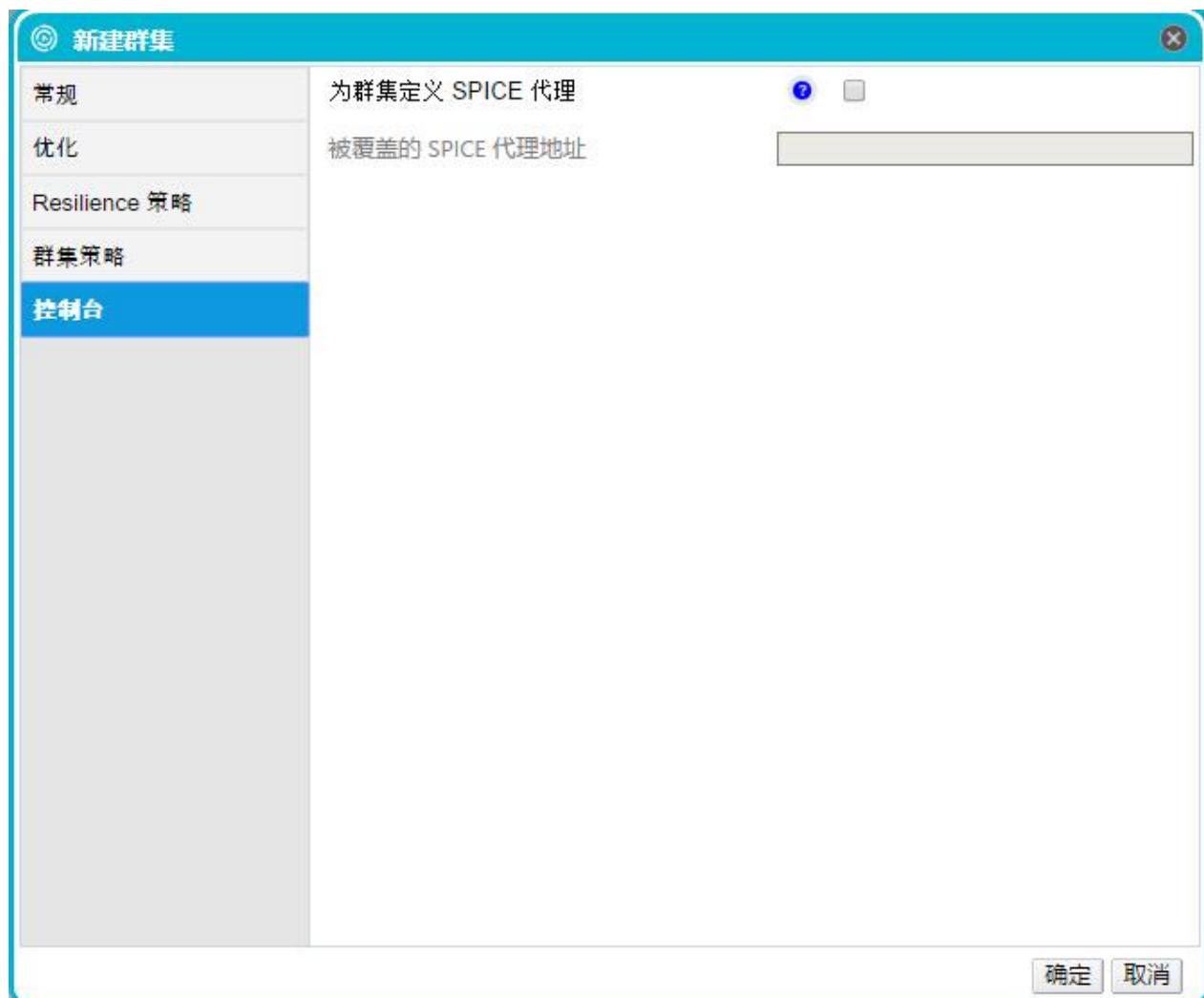
打开 Resilience 策略选项，选择是否迁移虚拟机或者只迁移高可用性虚拟机



打开群集策略选项，根据实际生产环境或测试环境进行选择。

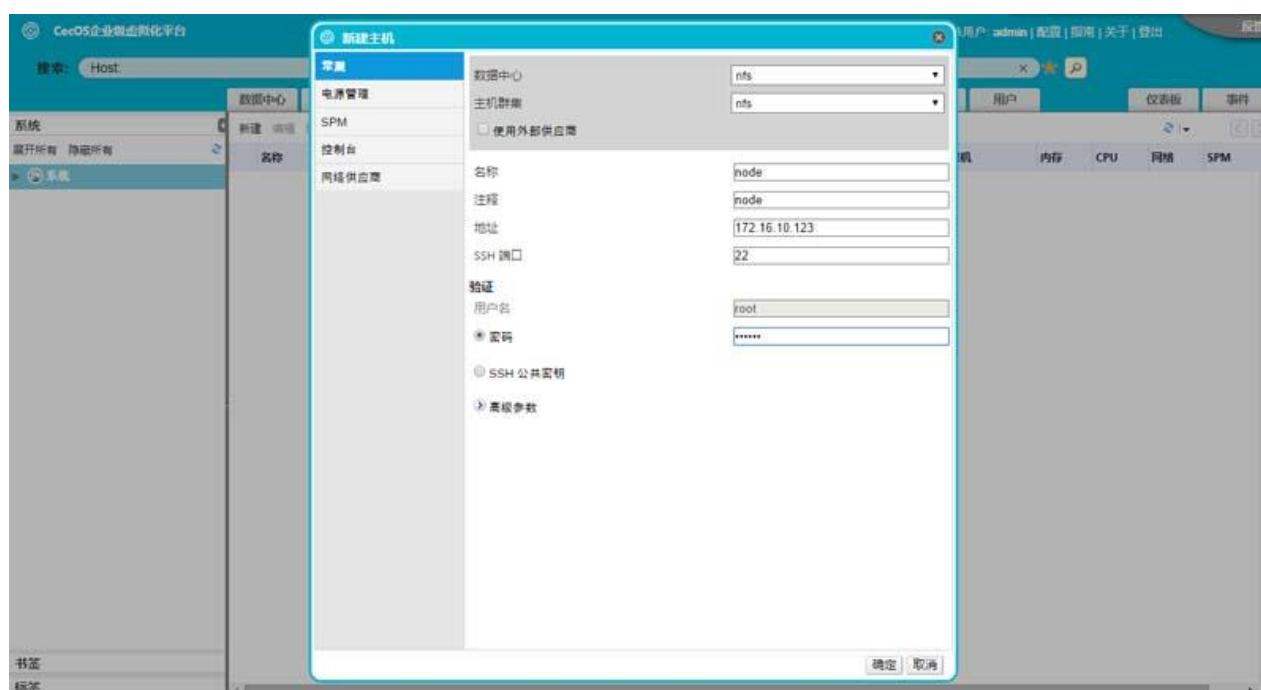


打开控制台选项，根据实际情况设置 Spice 代理。设置完成后，点击确定，完成群集配置



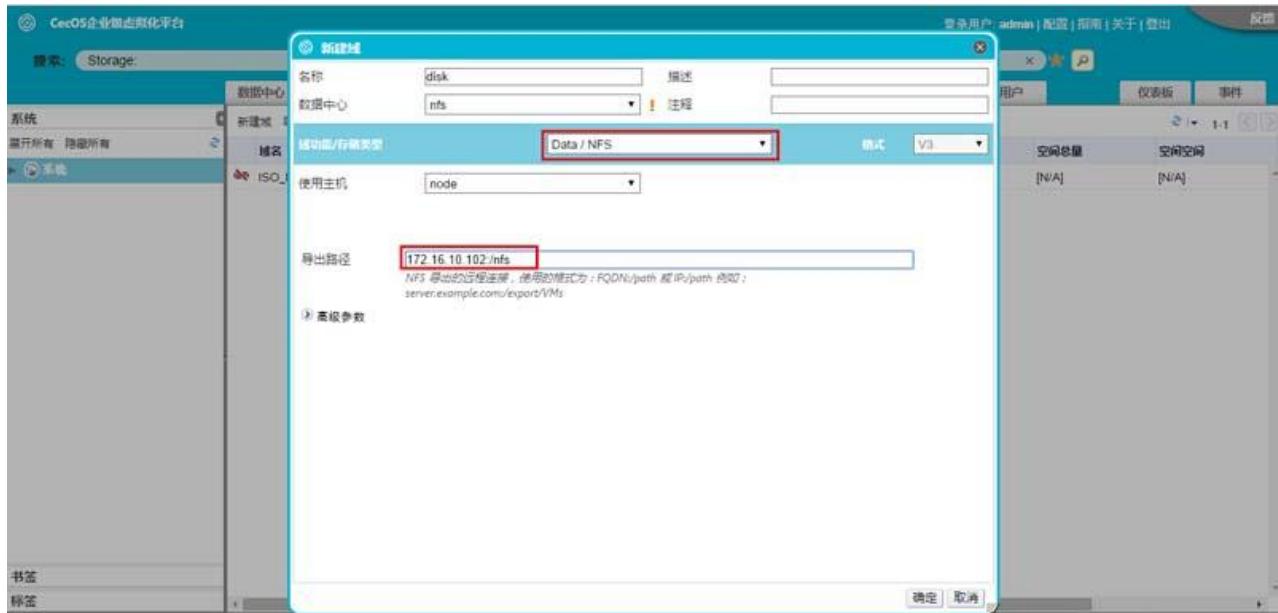
4. 添加主机

打开主机管理界面，新建主机，数据中心选择为新建的 nfs，然后输入主机名和地址及主机系统密码，完成后点击确定，待主机添加完成后配置 NFS 存储。

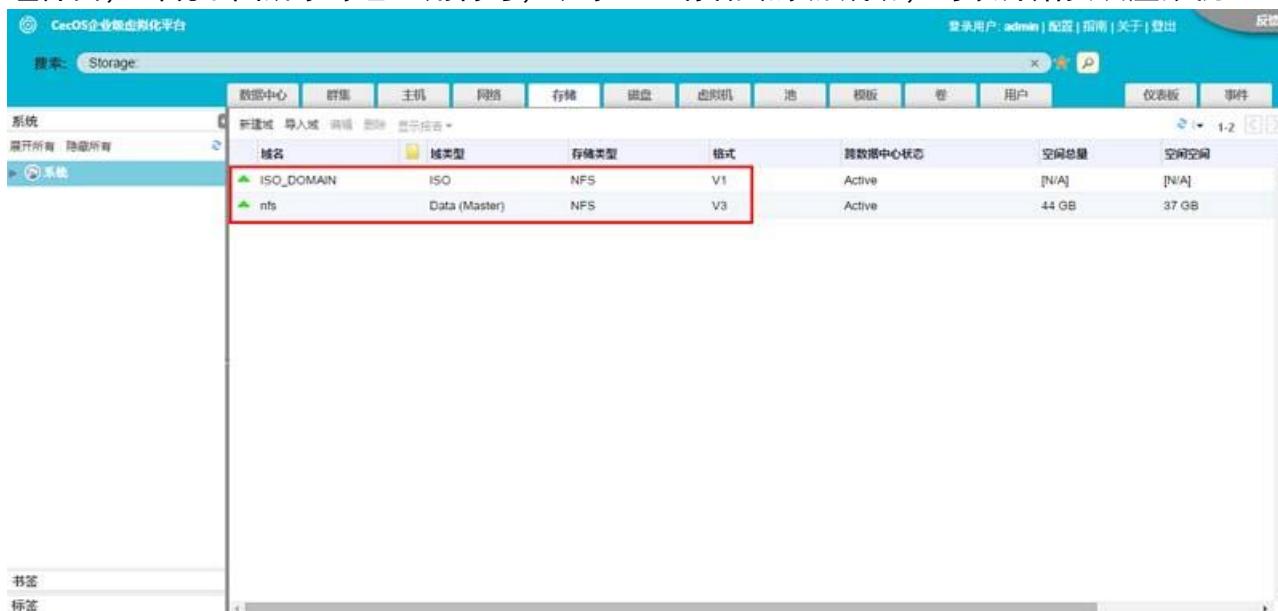


5. 新建存储

打开存储管理界面，点击新建域，打开配置界面，输入名称，选择新建的数据中心，选择域功能/存储类型，默认为 Data/NFS，选择使用主机，导出路径设置为 NFS 路径。点击确定，完成 NFS 存储域添加。



存储建立完成以后，进入数据中心，激活存储，附加 ISO 存储域并激活，然后进入存储管理界面，出现下图所示绿色三角符号，表示 NFS 存储域添加成功，可以开始安装虚拟机。



[点击下载](#)

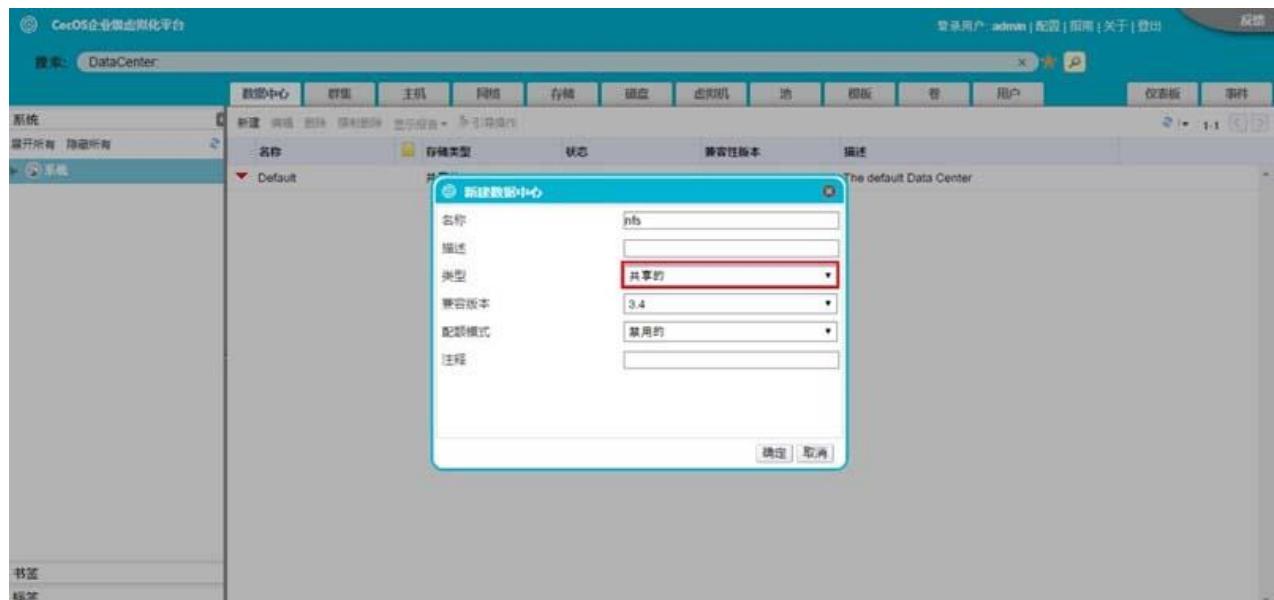
=====ISCSI 存储配置=====

1. 新建数据中心

进入数据中心管理界面，新建数据中心，打开下图界面，输入数据中心名称，选择存储类型。

注意：存储类型选择为共享的。

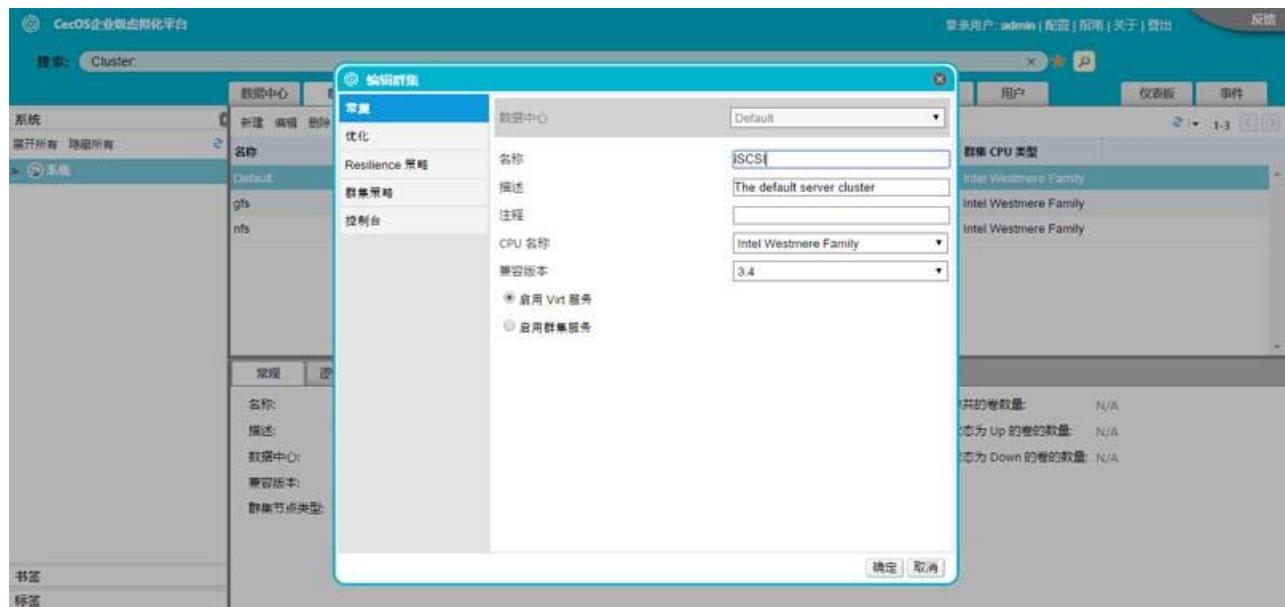
点击确定，数据中心建立完成。



2. 新建群集

打开群集选项，新建群集，选择数据中心，输入群集名称，设置 CPU 类型

注意：CPU 类型要和主机服务器 CPU 类型保持一致，启用 Virt 服务



打开优化选项，可以看到内存优化 CPU 线程两个选项，根据实际需要进行设置



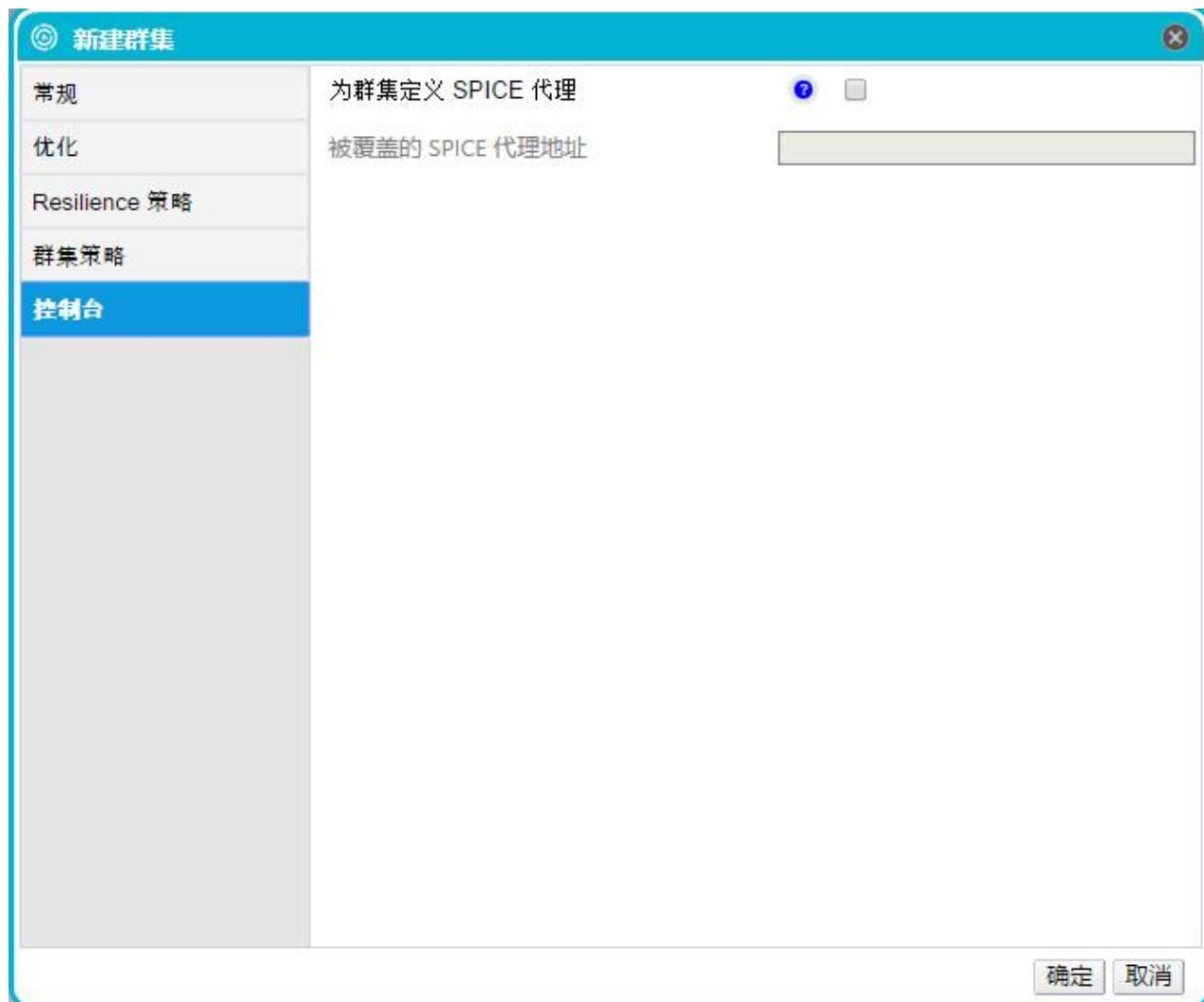
打开 Resilience 策略选项，选择是否迁移虚拟机或者只迁移高可用性虚拟机



打开群集策略选项，根据实际生产环境或测试环境进行选择。



打开控制台选项，设置 Spice 代理，具体设置根据实际情况设置。设置完成后，点击确定，完成群集配置。



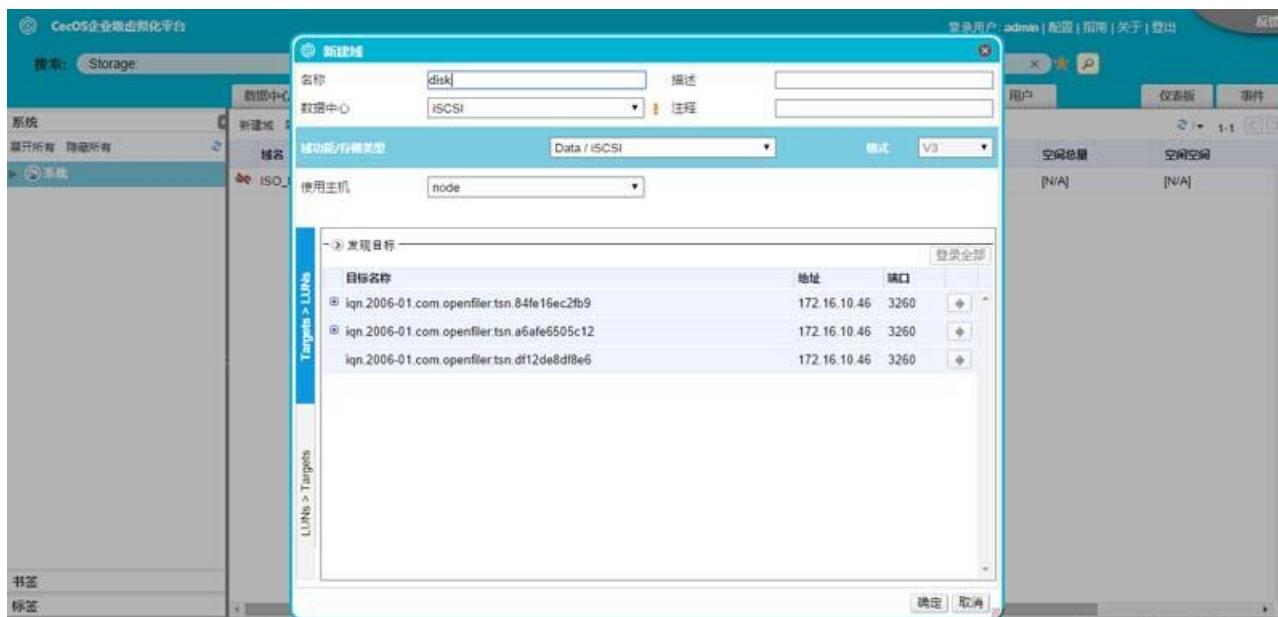
3. 添加主机

打开主机管理界面，新建主机，数据中心选择为新建的 iSCSI，然后输入主机名和地址及主机系统密码，完成后点击确定，待主机添加完成后配置 iSCSI 存储

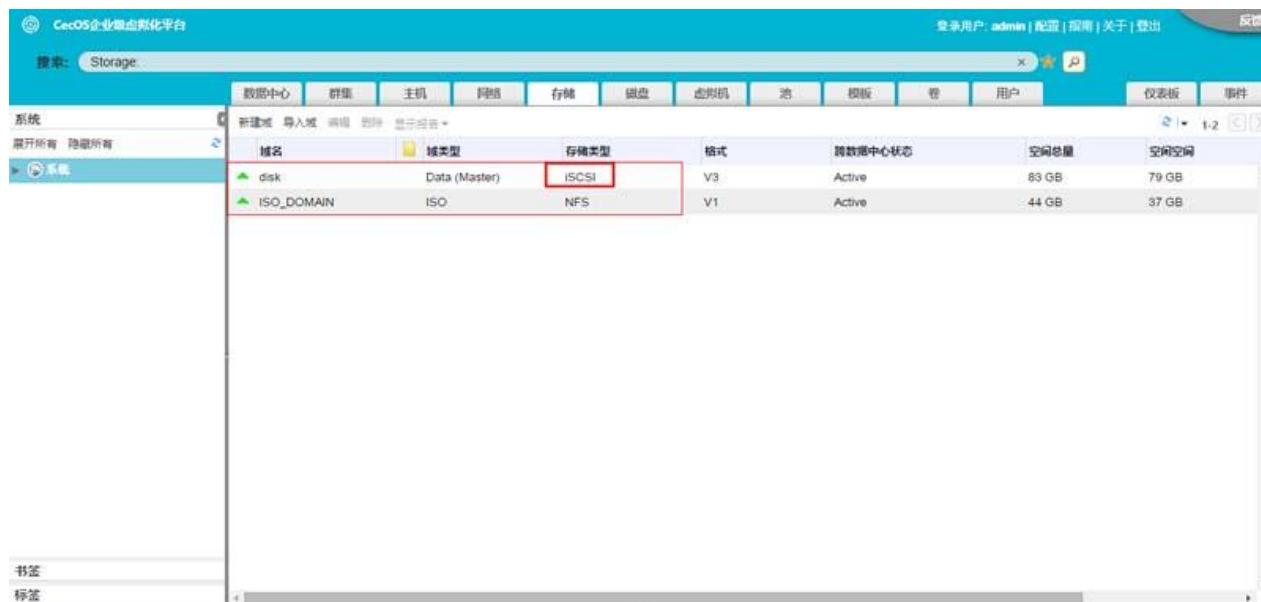


4. 新建存储

打开存储管理界面，点击新建域，打开配置界面，输入名称，选择新建的数据中心，选择域功能/存储类型，默认为 Data/iSCSI，选择作用主机。然后点击发现目标，输入 iSCSI 服务器地址，点击发现，系统会搜索出所配置好的 iSCSI 设备，点击全部登录，选中 iSCSI 设备，点击确定，等待存储建立完成



存储建立完成以后，进入数据中心，激活存储，附加 ISO 存储域并激活，然后进入存储管理界面，出现下图所示，表示 iSCSI 存储域添加成功，可以开始安装虚拟机



[点击下载](#)

卷存储

1. 卷存储安装环境

安装完成 CecOS 平台机器一台，并配置初始安装环境

```
CecOS release 1.4 (Niu'er)
Kernel 2.6.32-431.23.3.el6.x86_64 on an x86_64

cecos login: root
Password:
Last login: Sun Sep  7 23:22:59 from 192.168.1.106
[root@cecos ~]# mount /dev/dvd /mnt
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@cecos ~]# cd /mnt/
[root@cecos mnt]# ls
EULA  Packages  RPM-GPG-KEY-OPENFANS-cecos  Script      version
GPL   README     run                         TRANS.TBL
[root@cecos mnt]# sh ./run _
```

初始安装环境配置完成

```
CecOS release 1.4 (Niu'er)
Kernel 2.6.32-431.23.3.el6.x86_64 on an x86_64

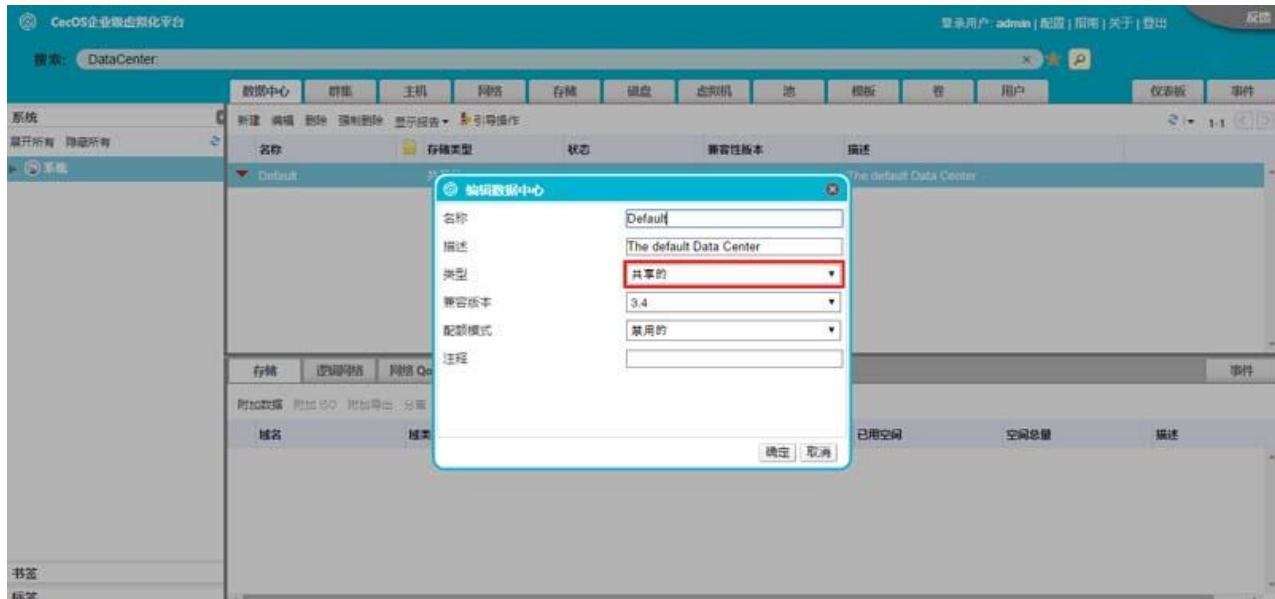
cecos login: root
Password:
Last login: Sun Sep  7 23:22:59 from 192.168.1.106
[root@cecos ~]# mount /dev/dvd /mnt
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@cecos ~]# cd /mnt/
[root@cecos mnt]# ls
EULA  Packages  RPM-GPG-KEY-OPENFANS-cecos  Script      version
GPL   README    run                           TRANS.TBL

[root@cecos mnt]# sh ./run
Copy files to your system, please wait...
CecOSvt-1.4                                | 2.9 kB    00:00 ...
CecOSvt-1.4/filelists_db                     | 281 kB    00:00 ...
CecOSvt-1.4/primary_db                       | 291 kB    00:00 ...
CecOSvt-1.4/other_db                         | 204 kB    00:00 ...
Metadata Cache Created
Done!
CecOSvt Local Yum Repo maked!
Use command "cecosvt-install" to install CecOSvt packages.
[root@cecos mnt]# _
```

下面开始配置卷存储

2. 数据中心

采用默认数据中心，类型为共享的。



3. 新建群集

建立两个群集，一个类型选择为 cluster，一个选择为 Virt，Cluster 用作存放存储机器，Virt 用于存放计算节点机器

The screenshot shows the CeeOS Cluster Management interface. On the left, there's a sidebar with '系统' (System) and '集群' (Cluster) sections. The '集群' section has '新建' (New), '编辑' (Edit), and '删除' (Delete) buttons, and a list of existing clusters: 'GlusterFS' and 'virt'. The main area is titled '编辑群集' (Edit Cluster).
集群 virt 配置:

- 数据中心: Default
- 名称: virt
- 描述: The default server cluster
- CPU 名称: Intel Westmere Family
- 兼容版本: 3.4
- 启用 Virt 服务
- 启用群集服务

集群 GlusterFS 配置:

- 数据中心: Default
- 名称: GlusterFS
- 描述:
- CPU 名称: Intel Westmere Family
- 兼容版本: 3.4
- 启用 Virt 服务
- 启用群集服务

右侧显示了 CPU 类型: Intel Westmere Family, 共的卷数量: N/A, 心为 Up 的卷的数量: N/A, 心为 Down 的卷的数量: N/A。

新建两台主机，一台启用群集服务，一台启用 Virt 服务

4. 添加主机

The screenshot shows the Cocos Enterprise Virtualization Platform interface. At the top, there's a navigation bar with tabs for Data Center, Cluster, Host, Network, Storage, Disk, Virtual Machine, Pool, Template, Volume, User, Dashboard, and Events. The 'Host' tab is selected. Below the navigation bar is a table listing hosts:

名称	主机名/IP	集群	数据中心	状态	虚拟机	内存	CPU	网络	SPM
GlusterFS	172.16.10.149	GlusterFS	Default	Installing	0	0%	0%	0%	正常
node	172.16.10.123	virt	Default	Up	0	8%	0%	3%	正常

On the left, there's a sidebar with '系统' (System) and '集群' (Cluster) sections. The '集群' section is expanded, showing '集群' (Cluster), '卷' (Volume), and '用户' (User). Below the sidebar is a '日志' (Log) section titled '上一条消息' (Last Message) with a list of log entries from August 27, 2014, at 15:58. The log entries are:

- 2014-8月-27, 15:58 Installing Host GlusterFS. Stage: Environment packages setup.
- 2014-8月-27, 15:58 Installing Host GlusterFS. Stage: Environment packages setup.
- 2014-8月-27, 15:58 Installing Host GlusterFS. Stage: Environment setup.
- 2014-8月-27, 15:58 Installing Host GlusterFS. Stage: Initializing.
- 2014-8月-27, 15:58 Installing Host GlusterFS. Connected to host 172.16.10.149 with SSH key fingerprint: 7a:24:53:a0:0a:1f:0fc8:93:6f:74:8a:9c:a4:0a:e9.
- 2014-8月-27, 15:58 Host GlusterFS was added by admin.
- 2014-8月-27, 15:57 State was set to Up for host node.
- 2014-8月-27, 15:57 Host node installed.
- 2014-8月-27, 15:57 Installing Host node. Stage: Termination.
- 2014-8月-27, 15:57 Installing Host node. Retrieving installation logs to: '/var/log/libvirt-engine/host-deploy/covrt-20140827155742-172.16.10.123-19765d0a.log'.
- 2014-8月-27, 15:57 Installing Host node. Stage: Pre-termination.

系统自动安装所需组件

This screenshot shows the same interface as the previous one, but the 'Host' tab is not selected. Instead, it shows a large list of log entries for the installation of 'Host node2'. The log entries are identical to those in the previous screenshot, detailing the installation of various system packages and dependencies. The log entries are timestamped from July 09, 2014, at 14:01 to 14:01.

添加主机完成。

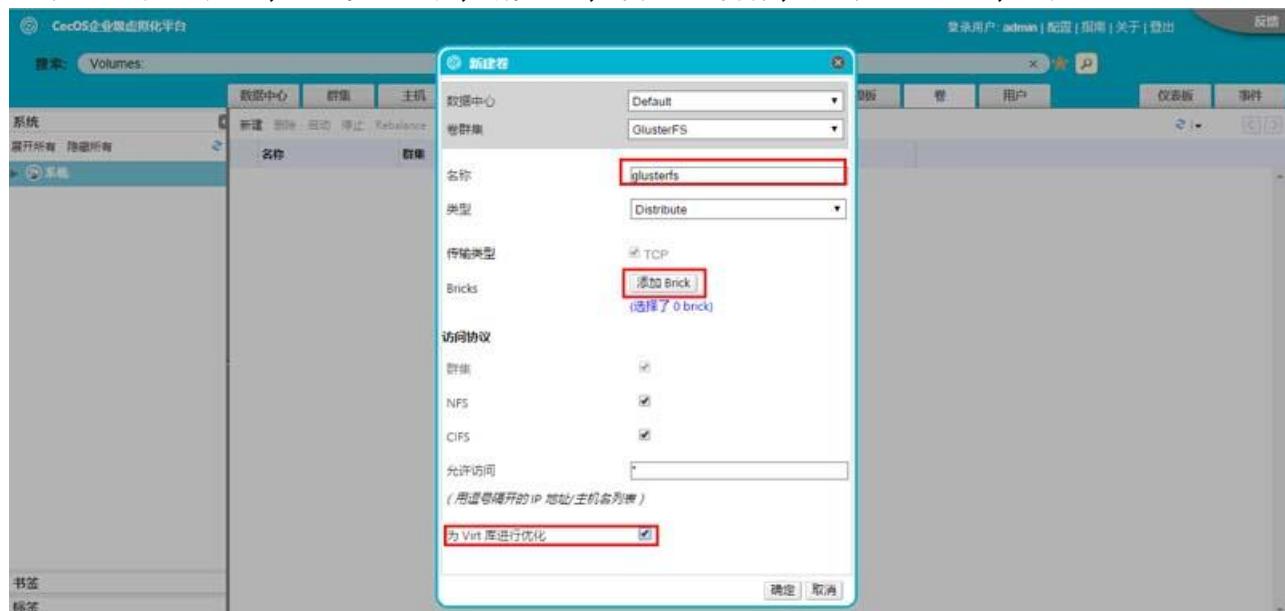
The screenshot shows the 'Storage' tab selected in the navigation bar. Below the navigation bar is a table listing storage volumes:

域名	域类型	存储类型	格式	跨数据中心状态	空间总量	空间可用
disk	Data (Master)	GlusterFS	V3	Active	74 GB	70 GB
ISO_DOMAIN	ISO	NFS	V1	Active	44 GB	37 GB

On the left, there's a sidebar with '系统' (System) and '集群' (Cluster) sections. The '集群' section is expanded, showing '集群' (Cluster), '卷' (Volume), and '用户' (User). Below the sidebar is a '日志' (Log) section titled '上一条消息' (Last Message) with a list of log entries from July 09, 2014, at 14:01. The log entries are identical to those in the previous screenshots.

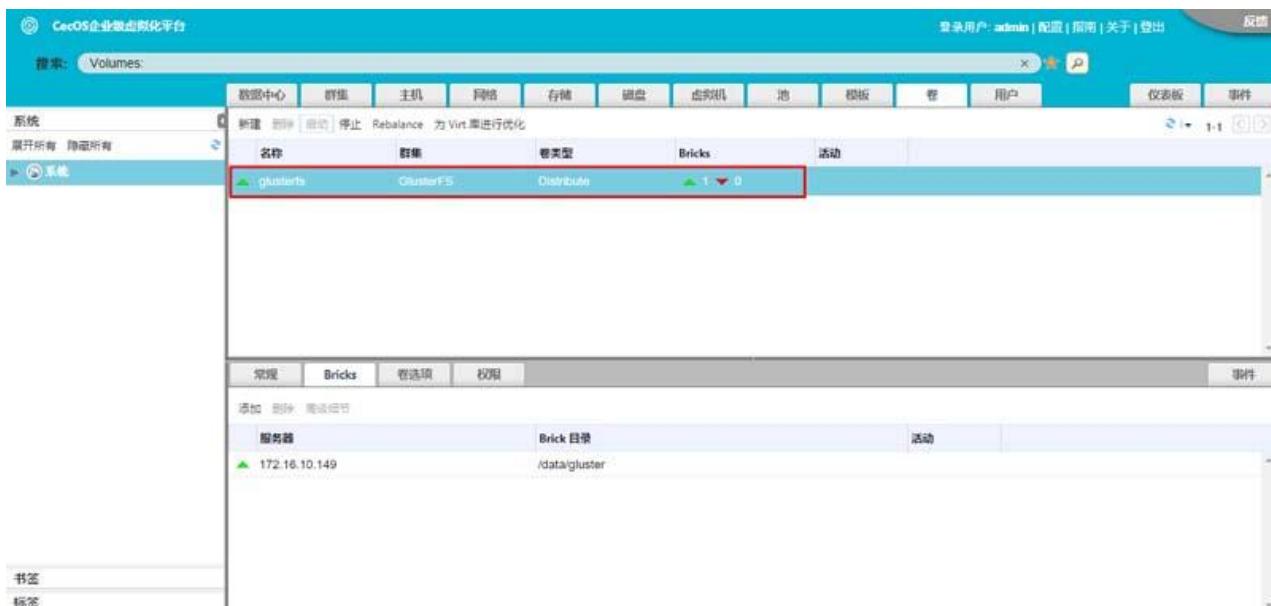
5. 创建卷存储

登录 WEB 管理界面，选择卷选项，新建卷，输入卷名称，并添加 brick，确定

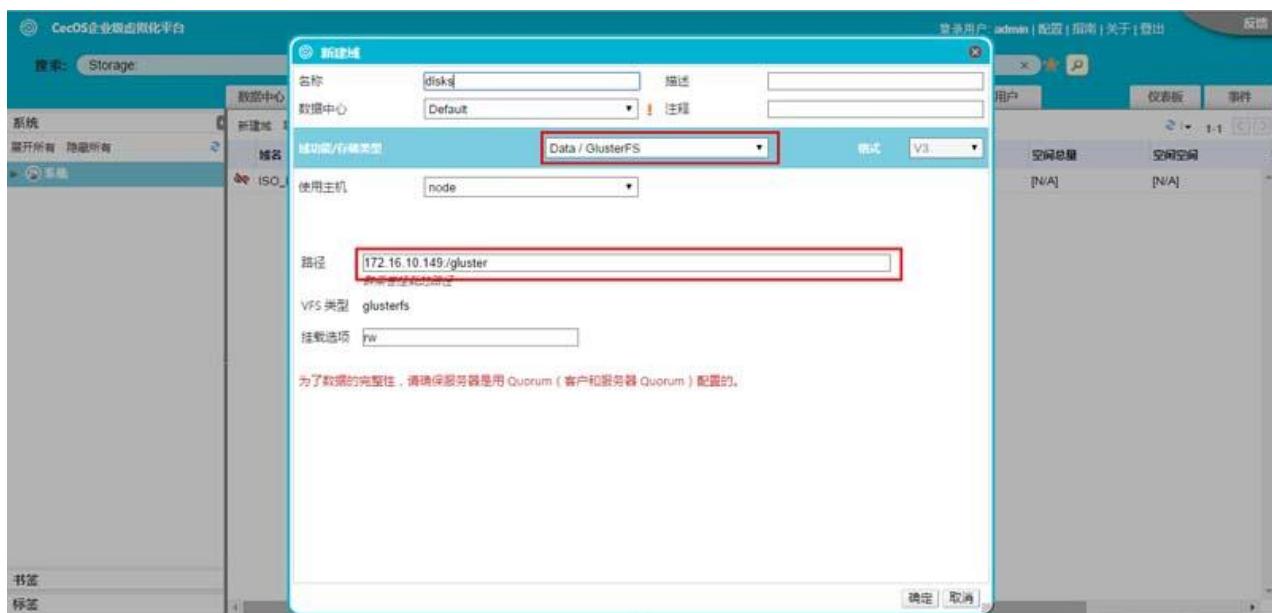




完成卷的创建，选择并启动创建的卷



然后开始添加存储，写入卷路径。确定，添加存储完成



打开数据中心，选择附加 ISO，添加 ISO 存储域并激活



域名	域类型	存储类型	格式	跨数据中心状态	空间总量	空间可用
disk	Data (Master)	GlusterFS	V3	Active	74 GB	70 GB
ISO_DOMAIN	ISO	NFS	V1	Active	44 GB	37 GB

CecOS 卷存储配置完成，现在可以开始安装虚拟机

[点击下载](#)

操作指南

管理员指南：

http://docs.openfans.org/cecos/cecos-65876863/cecos-1-4-64cd4f5c63075357/cecos-v1-4-7ba17406545863075357/at_download/file

用户指南：

http://docs.openfans.org/cecos/cecos-65876863/cecos-1-4-64cd4f5c63075357/cecos-v1-4-7528623763075357-1/at_download/file

XEN

```
[root@huatech ~]# clear
[root@huatech ~]# yum -y install centos-release-xen
[root@huatech ~]# sed -i -e "s(enabled=1/enable=0/g"
/etc/yum.repos.d/CentOS-Xen.repo
[root@huatech ~]# yum --enablerepo=centos-virt-xen -y update kernel
[root@huatech ~]# yum --enablerepo=centos-virt-xen -y install xen
[root@huatech ~]# vim /etc/default/grub
[root@huatech ~]# /bin/grub-bootxen.sh
[root@huatech ~]# nmcli c add type bridge con-name br0 ifname br0
[root@huatech ~]# nmcli connection modify br0 ipv4.addresses 192.168.88.233/24
ipv4.method manual
[root@huatech ~]# nmcli connection modify br0 ipv4.gateway 192.168.88.1
[root@huatech ~]# nmcli connection modify br0 ipv4.dns 192.168.88.1
[root@huatech ~]# nmcli connection delete eno16777728
[root@huatech 桌面]# nmcli connection add type bridge-slave autoconnect yes
con-name eno16777728 ifname eno16777728 master br0
[root@huatech 桌面]# systemctl stop NetworkManager;systemctl start NetworkManager
```

```
[root@huatech ~]# yum --enablerepo=centos-virt-xen -y install libvirt  
libvirt-daemon-xen virt-install  
[root@huatech ~]# systemctl start libvirtd  
[root@huatech ~]# systemctl enable libvirtd  
[root@huatech ~]# mkdir -p /var/xen/images  
[root@huatech ~]# virt-install --help
```

十六、分布式存储 GlusterFS

【ON ALL NODES】

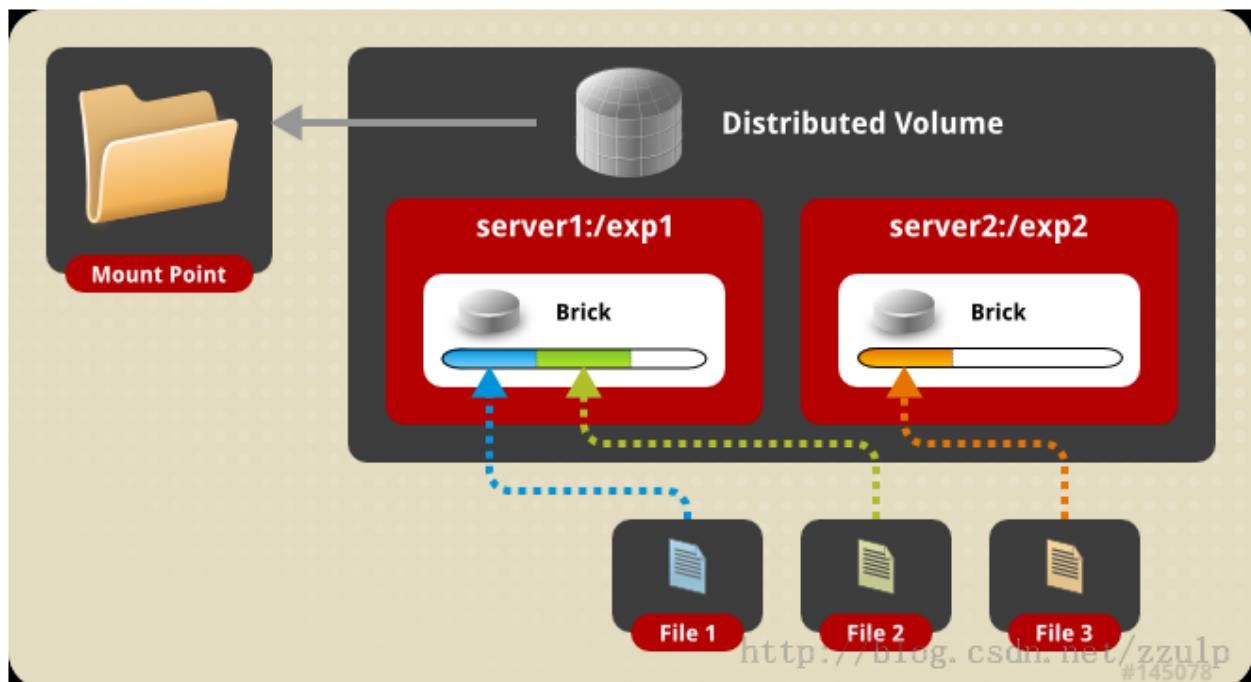
```
[root@huatech ~]# fdisk /dev/sdb  
[root@huatech ~]# mkfs -t ext4 /dev/sdb1  
[root@huatech ~]# mkdir glusterfs  
[root@huatech ~]# curl  
http://download.gluster.org/pub/gluster/glusterfs/LATEST/EPEL.repo/glusterfs-epeL.repo -o /etc/yum.repos.d/glusterfs-epeL.repo
```

```
[root@huatech ~]# yum --enablerepo=epel -y install glusterfs-server  
[root@huatech ~]# yum --enablerepo=epel -y install glusterfs-server  
[root@huatech ~]# systemctl start glusterd  
[root@huatech ~]# systemctl enable glusterd  
[root@huatech ~]# yum -y install rpcbind  
[root@huatech ~]# systemctl enable rpcbind  
[root@huatech ~]# systemctl restart glusterd
```

【node1】【先挂载 Glusterfs】

```
[root@huatech ~]# mkdir /glusterfs/distributed【all nodes】  
[root@server02 ~]# gluster peer probe server01  
peer probe: success.  
[root@server02 ~]# gluster peer status
```

3.1.1 distributed volume



分布卷可以将某个文件随机的存储在卷内的一个 brick 内，通常用于扩展存储能力，不支持数据的冗余。

```
[root@server01 ~]# gluster volume create vol_distributed transport tcp server01:/glusterfs/distributed/ server02:/glusterfs/distributed/
```

```
[root@server01 ~]# gluster volume start vol_distributed  
=====客户端处理=====
```

```
[root@server01 ~]# curl
```

```
http://download.gluster.org/pub/gluster/glusterfs/LATEST/EPEL.repo/glusterfs-epe l.repo -o /etc/yum.repos.d/glusterfs-epe l.repo
```

```
[root@server01 ~]# yum -y install glusterfs glusterfs-fuse
```

```
[root@server01 ~]# mount -t glusterfs server01:/vol_distributed /mnt
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/vg_server01-lv_root	28G	5.6G	21G	22%	/
tmpfs	491M	80K	491M	1%	/dev/shm
/dev/sda1	485M	35M	426M	8%	/boot
/dev/sr0	4.2G	4.2G	0	100%	/media/CentOS_6.5_Final
server01:/vol_distributed	20G	73M	19G	1%	/mnt

```
=====NFS V3 客户端支持=====
```

```
[root@server01 ~]# yum -y install nfs-utils
```

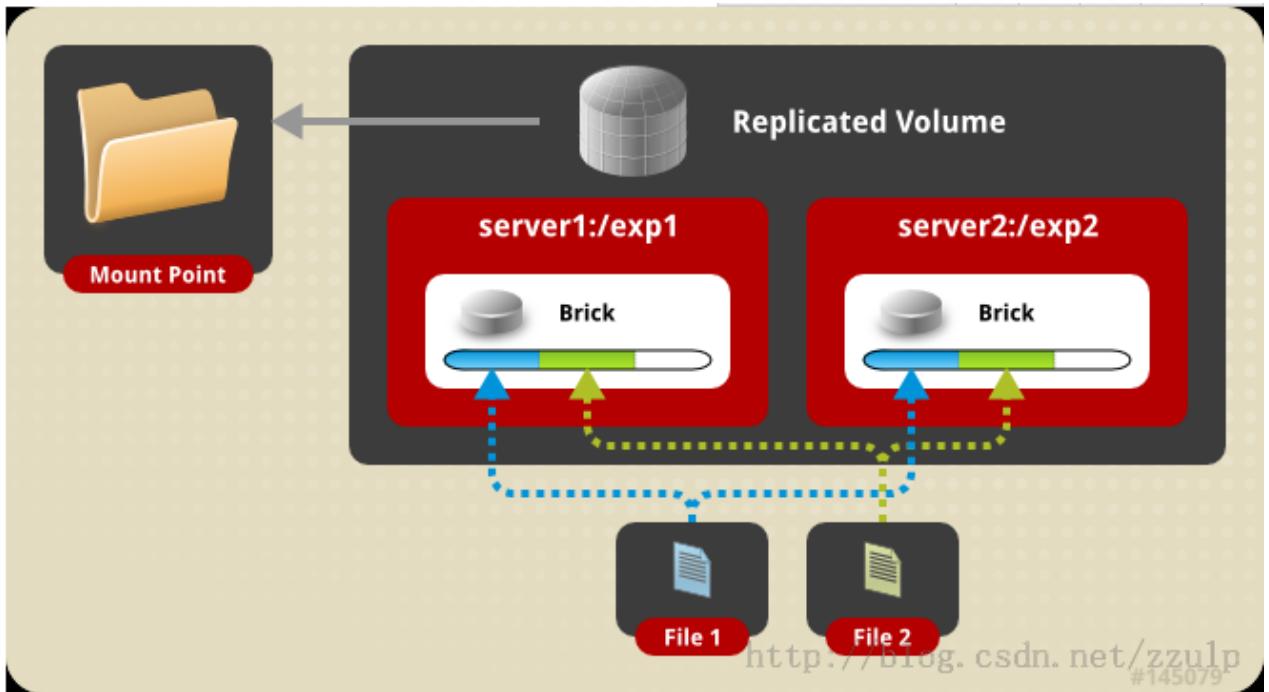
```
[root@server01 ~]# systemctl start rpcbind rpc-statd
```

```
[root@server01 ~]# systemctl enable rpcbind rpc-statd
```

```
[root@server01 ~]# mount -t nfs -o mountvers=3
```

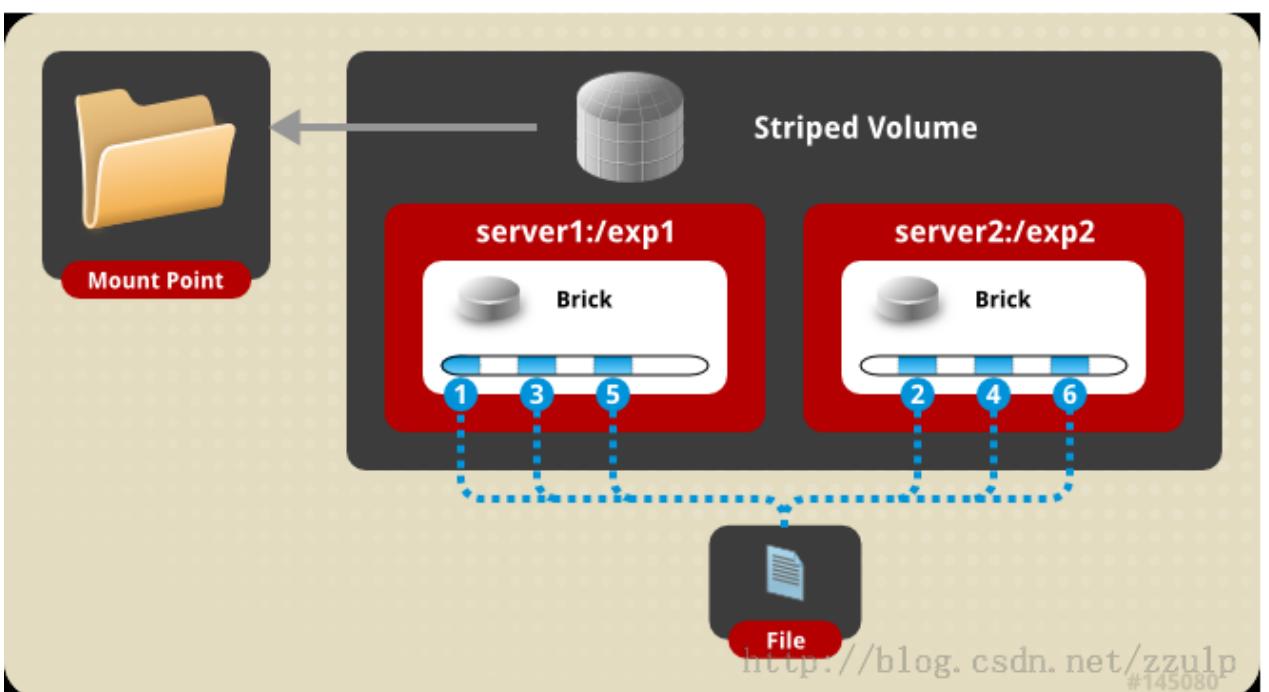
```
node01.server.world:/vol_distributed /mnt
```

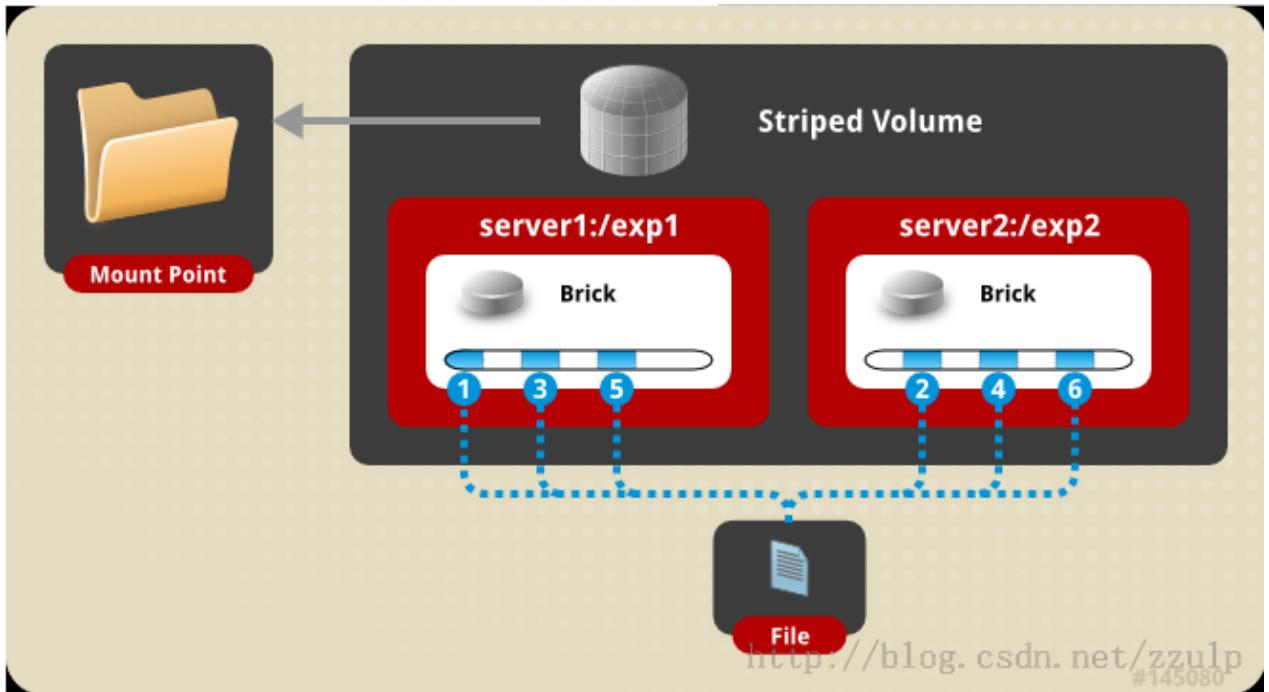
```
[root@server01 ~]# df -hT
```



```
[root@server01 ~]# mkdir /glusterfs/replica
[root@server01 ~]# gluster peer probe server02
[root@server01 ~]# gluster peer status
[root@server01 ~]# gluster volume create vol_replica replica 2 transport tcp
server01:/glusterfs/replica/ server02:/glusterfs/replica/
[root@server01 ~]# gluster volume start vol_replica
```

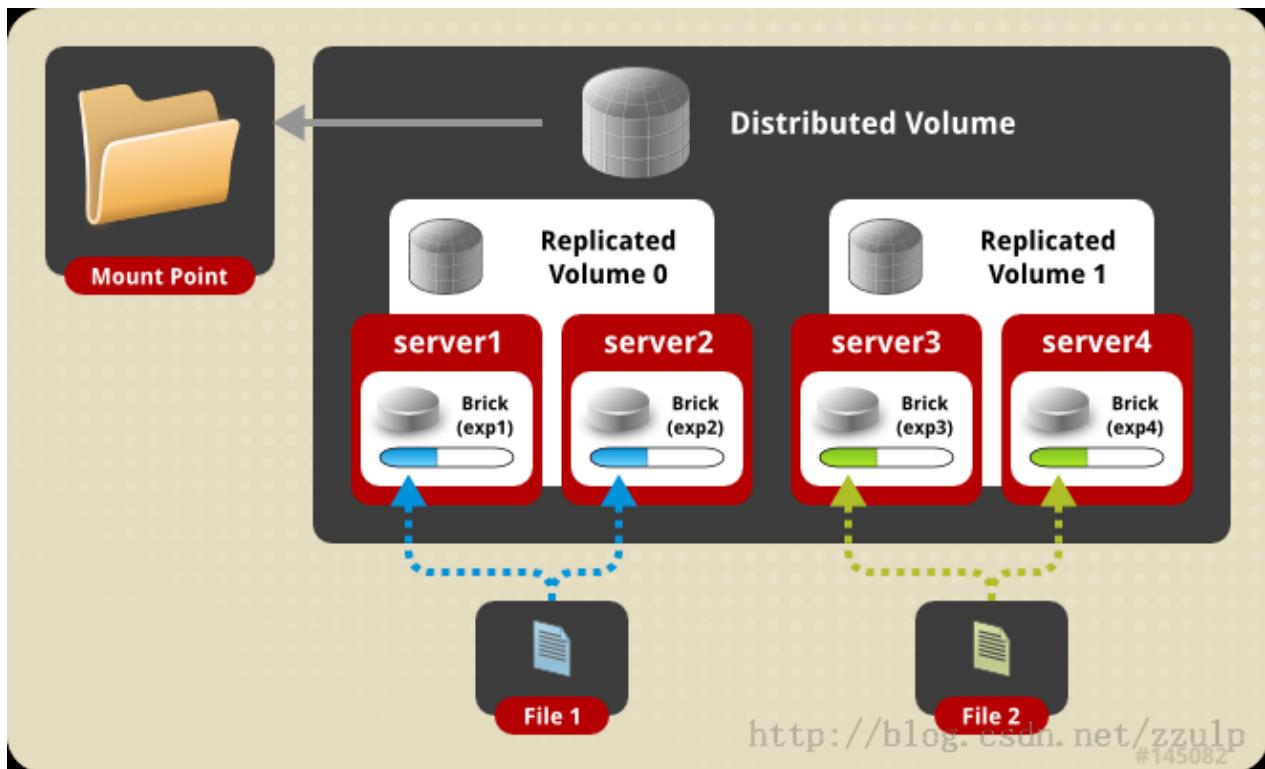
```
[root@server01 ~]# mount -t glusterfs server01:/vol_replica /mnt
[root@server01 ~]# df -hT
Filesystem           Type      Size  Used Avail Use% Mounted on
/dev/mapper/vg_server01-lv_root  ext4      28G  5.6G  21G  22% /
tmpfs                 tmpfs     491M   80K  491M   1% /dev/shm
/dev/sda1              ext4      485M   35M  426M   8% /boot
/dev/sr0                iso9660  4.2G   4.2G    0 100% /media/CentOS_6.5_Final
server01:/vol_replica  fuse.glusterfs  9.8G   37M  9.2G   1% /mnt
```





```
[root@server01 ~]# mkdir /glusterfs/stripped  
[root@server01 ~]# gluster volume create vol_stripped stripe 2 transport tcp  
server01:/glusterfs/stripped/ server02:/glusterfs/stripped/  
[root@server01 ~]# gluster volume start vol_stripped  
[root@server01 ~]# gluster volume info  
[root@server01 ~]# mount -t glusterfs server01:vol_stripped /mnt
```

```
[root@server01 ~]# df -hT  
Filesystem           Type      Size  Used Avail Use% Mounted on  
/dev/mapper/vg_server01-lv_root ext4      28G  5.6G   21G  22% /  
tmpfs                tmpfs     491M   80K  491M   1% /dev/shm  
/dev/sda1              ext4      485M   35M  426M   8% /boot  
/dev/sr0               iso9660   4.2G   4.2G    0 100% /media/CentOS_6.5_Final  
server01:vol_stripped        fuse.glusterfs  20G   73M   19G   1% /mnt
```



```
+-----+ | [GlusterFS Server#1] | 10.0.0.51 | 10.0.0.52 | [GlusterFS Server#2] |  
| node01.server.world +-----+-----+-----+ node02.server.world |  
+-----+ | | | +-----+  
+-----+ | [GlusterFS Server#3] | 10.0.0.53 | 10.0.0.54 | [GlusterFS Server#4] |  
| node03.server.world +-----+-----+-----+ node04.server.world |  
+-----+ | | | +-----+
```

- [1] [Install GlusterFS Server on All Nodes, refer to here.](#)
- [2] Create a Directory for GlusterFS Volume on all Nodes.
`mkdir /glusterfs/stripe-replica`
- [3] Configure Clustering like follows on a node. (it's OK on any node)

```
[root@node01 ~]#  
gluster peer probe node02  
peer probe: success.  
[root@node01 ~]#  
gluster peer probe node03  
peer probe: success.  
[root@node01 ~]#  
gluster peer probe node04  
peer probe: success.  
# show status  
[root@node01 ~]#  
gluster peer status
```

```
Number of Peers: 3
Hostname: node02
Uuid: 2ca22769-28a1-4204-9957-886579db2231
State: Peer in Cluster (Connected)
Hostname: node03
Uuid: 79cff591-1e98-4617-953c-0d3e334cf96a
State: Peer in Cluster (Connected)
Hostname: node04
Uuid: 779ab1b3-fda9-46da-af95-ba56477bf638
State: Peer in Cluster (Connected)
# create volume
[root@node01 ~]#
gluster volume create vol_strip-replica stripe 2 replica 2 transport tcp \
node01:/glusterfs/strip-replica \
node02:/glusterfs/strip-replica \
node03:/glusterfs/strip-replica \
node04:/glusterfs/strip-replica
volume create: vol_strip-replica: success: please start the volume to access data
# start volume
[root@node01 ~]#
gluster volume start vol_strip-replica
volume start: vol_strip-replica: success
# show volume info
[root@node01 ~]#
gluster volume info
Volume Name: vol_strip-replica
Type: Striped-Replicate
Volume ID: ec36b0d3-8467-47f6-aa83-1020555f58b6
Status: Started
Number of Bricks: 1 x 2 x 2 = 4
Transport-type: tcp
Bricks:
Brick1: node01:/glusterfs/strip-replica
Brick2: node02:/glusterfs/strip-replica
Brick3: node03:/glusterfs/strip-replica
Brick4: node04:/glusterfs/strip-replica
Options Reconfigured:
performance.readdir-ahead: on
```

[4] [To mount GlusterFS volume on clients, refer to here.](#)

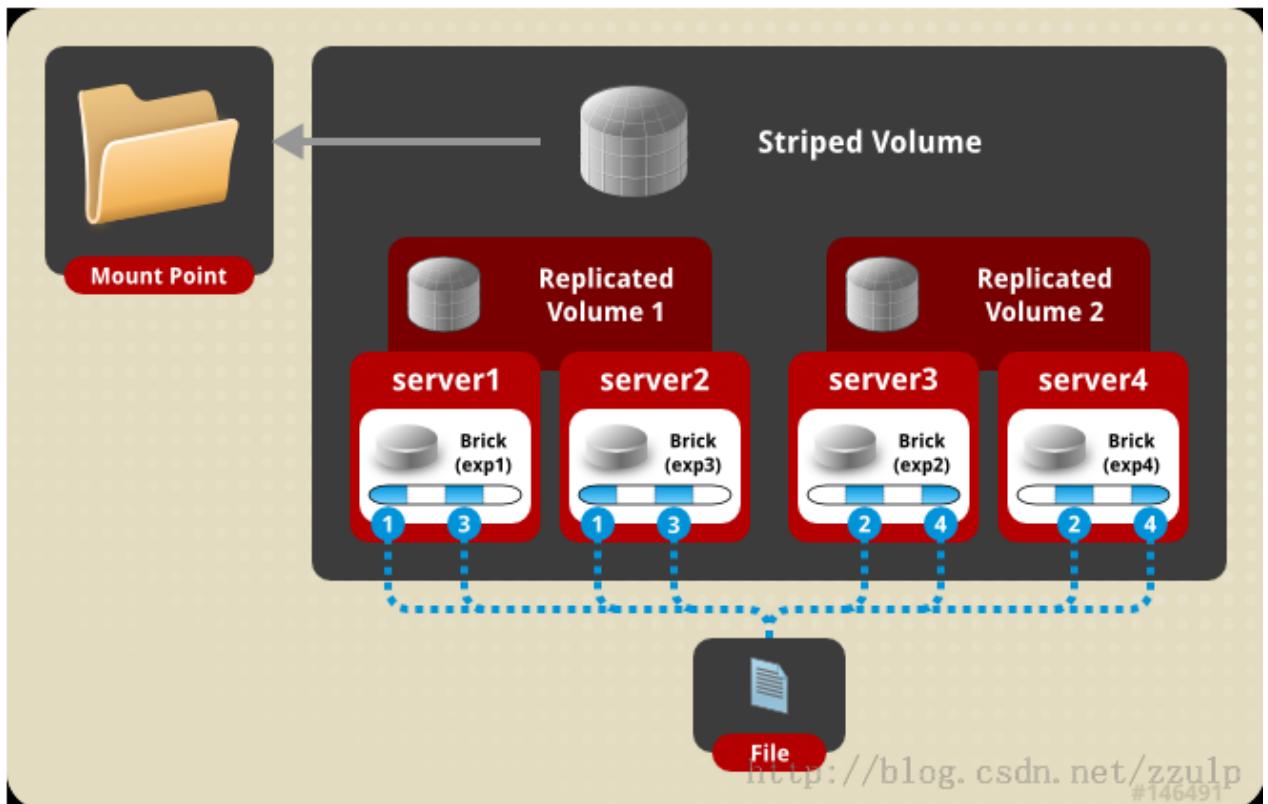
```
[root@client ~]#
curl
http://download.gluster.org/pub/gluster/glusterfs/LATEST/EPEL.repo/glusterfs-epel.re
po -o /etc/yum.repos.d/glusterfs-epel.repo
[root@client ~]#
```

```
yum -y install glusterfs glusterfs-fuse
# mount vol_distributed volume on /mnt
[root@client ~]#
mount -t glusterfs node01.server.world:/vol_distributed /mnt
[root@client ~]#
df -hT
Filesystem          Type      Size  Used Avail Use% Mounted on
/dev/mapper/centos-root    xfs       27G  1.1G  26G  5% /
devtmpfs            devtmpfs   2.0G     0  2.0G  0% /dev
tmpfs               tmpfs      2.0G     0  2.0G  0% /dev/shm
tmpfs               tmpfs      2.0G  8.3M  2.0G  1% /run
tmpfs               tmpfs      2.0G     0  2.0G  0%
/sys/fs/cgroup
/dev/vda1            xfs      497M  151M  347M  31% /boot
node01.server.world:/vol_distributed fuse.glusterfs  40G   65M  40G  1% /mnt
```

[2] NFS (v3) is also supported, so it's possible to mount with NFS.

[Configure for it on GlusterFS Servers first, refer to here.](#)

```
[root@client ~]#
yum -y install nfs-utils
[root@client ~]#
systemctl start rpcbind rpc-statd
[root@client ~]#
systemctl enable rpcbind rpc-statd
[root@client ~]#
mount -t nfs -o mountvers=3 node01.server.world:/vol_distributed /mnt
[root@client ~]#
df -hT
Filesystem          Type      Size  Used Avail Use% Mounted on
/dev/mapper/centos-root    xfs       27G  1.1G  26G  5% /
devtmpfs            devtmpfs   2.0G     0  2.0G  0% /dev
tmpfs               tmpfs      2.0G     0  2.0G  0% /dev/shm
tmpfs               tmpfs      2.0G  8.3M  2.0G  1% /run
tmpfs               tmpfs      2.0G     0  2.0G  0%
/sys/fs/cgroup
/dev/vda1            xfs      497M  151M  347M  31% /boot
node01.server.world:/vol_distributed nfs        40G   64M  40G  1% /mnt
```



- [1] Install GlusterFS Server on All Nodes, refer to here.
- [2] Create a Directory for GlusterFS Volume on all Nodes.
`mkdir /glusterfs/strip-replica`
- [3] Configure Clustering like follows on a node. (it's OK on any node)

```
# probe the node
[root@node01 ~]#
gluster peer probe node02
peer probe: success.
[root@node01 ~]#
gluster peer probe node03
peer probe: success.
[root@node01 ~]#
gluster peer probe node04
peer probe: success.

# show status
[root@node01 ~]#
gluster peer status
Number of Peers: 3
Hostname: node02
Uuid: 2ca22769-28a1-4204-9957-886579db2231
State: Peer in Cluster (Connected)
Hostname: node03
Uuid: 79cff591-1e98-4617-953c-0d3e334cf96a
State: Peer in Cluster (Connected)
```

```
Hostname: node04
Uuid: 779ab1b3-fda9-46da-af95-ba56477bf638
State: Peer in Cluster (Connected)
# create volume
[root@node01 ~]#
gluster volume create vol_strip-replica stripe 2 replica 2 transport tcp \
node01:/glusterfs/stripe-replica \
node02:/glusterfs/stripe-replica \
node03:/glusterfs/stripe-replica \
node04:/glusterfs/stripe-replica
volume create: vol_strip-replica: success: please start the volume to access data
# start volume
[root@node01 ~]#
gluster volume start vol_strip-replica
volume start: vol_strip-replica: success
# show volume info
[root@node01 ~]#
gluster volume info
Volume Name: vol_strip-replica
Type: Striped-Replicate
Volume ID: ec36b0d3-8467-47f6-aa83-1020555f58b6
Status: Started
Number of Bricks: 1 x 2 x 2 = 4
Transport-type: tcp
Bricks:
Brick1: node01:/glusterfs/stripe-replica
Brick2: node02:/glusterfs/stripe-replica
Brick3: node03:/glusterfs/stripe-replica
Brick4: node04:/glusterfs/stripe-replica
Options Reconfigured:
performance.readdir-ahead: on
```

=====客户端=====

```
[root@client ~]# curl
http://download.gluster.org/pub/gluster/glusterfs/LATEST/EPEL.repo/glusterfs-epel.re
po -o /etc/yum.repos.d/glusterfs-epel.repo
[root@client ~]#
yum -y install glusterfs glusterfs-fuse
# mount vol_distributed volume on /mnt
[root@client ~]#
mount -t glusterfs node01.server.world:/vol_distributed /mnt
[root@client ~]#
df -hT
Filesystem          Type      Size  Used Avail Use% Mounted on
/dev/mapper/centos-root    xfs       27G  1.1G   26G  5% /
```

```

devtmpfs          devtmpfs      2.0G    0  2.0G  0% /dev
tmpfs            tmpfs        2.0G    0  2.0G  0% /dev/shm
tmpfs            tmpfs        2.0G   8.3M  2.0G  1% /run
tmpfs            tmpfs        2.0G    0  2.0G  0%
/sys/fs/cgroup
/dev/vda1         xfs         497M  151M 347M 31% /boot
node01.server.world:/vol_distributed fuse.glusterfs  40G   65M  40G  1% /mnt

```

[2] NFS (v3) is also supported, so it's possible to mount with NFS.

[Configure for it on GlusterFS Servers first, refer to here.](#)

```

[root@client ~]#
yum -y install nfs-utils
[root@client ~]#
systemctl start rpcbind rpc-statd
[root@client ~]#
systemctl enable rpcbind rpc-statd
[root@client ~]#
mount -t nfs -o mountvers=3 node01.server.world:/vol_distributed /mnt
[root@client ~]#
df -hT
Filesystem           Type      Size  Used Avail Use% Mounted on
/dev/mapper/centos-root  xfs       27G  1.1G  26G  5% /
devtmpfs            devtmpfs  2.0G    0  2.0G  0% /dev
tmpfs              tmpfs     2.0G    0  2.0G  0% /dev/shm
tmpfs              tmpfs     2.0G   8.3M  2.0G  1% /run
tmpfs              tmpfs     2.0G    0  2.0G  0% /sys/fs/cgroup
/dev/vda1            xfs       497M  151M 347M 31% /boot
node01.server.world:/vol_distributed nfs      40G   64M  40G  1% /mnt

```

十七、弹性存储 LVM

```

[root@huatech ~]# fdisk /dev/sdb
[root@huatech ~]# pvdisplay
[root@huatech ~]# pvresize --setphysicalvolumesize 10G /dev/sdb1
[root@huatech ~]# pvcreate /dev/sdb1
[root@huatech ~]# pvresize --setphysicalvolumesize 10G /dev/sdb1
[root@huatech ~]# pvscan
[root@huatech ~]# pvremove /dev/sdb1
[root@huatech ~]# pvresize --setphysicalvolumesize 10G /dev/sdb1

[root@huatech ~]# vgcreate wmmvg /dev/sdb1
[root@huatech ~]# vgdisplay
[root@huatech ~]# vgscan
[root@huatech ~]# vgrename wmmvg sharevg
[root@huatech ~]# vgextend sharevg /dev/sdc1

```

```
[root@huatech ~]# vgre
vgreduce vgremove vgrenam
vgreduce: 减少 VG 量
vgremove: 移除
[root@huatech ~]# lvcreate -L 5G -n wmmlv sharevg
[root@huatech ~]# lvextend -L 6G /dev/sharevg/wmmlv 【增加容量、且针对文件系统】
  Size of logical volume sharevg/wmmlv changed from 5.00 GiB (1280 extents) to 6.00
  GiB (1536 extents).
Size of logical volume sharevg/wmmlv changed from 6.00 GiB (1536 extents) to 5.00
  GiB (1280 extents).
[root@huatech ~]# xfs_growfs /mnt 【xfs 文件系统增加】
[root@huatech ~]# lvresize /dev/sharevg/wmmlv 【调整 Ext4 文件系统容量】

[root@huatech ~]# e2fsck -f /dev/sharevg/wmmlv
[root@huatech ~]# resize2fs /dev/sharevg/wmmlv 4G
[root@huatech ~]# lvreduce -L 5G /dev/sharevg/wmmlv
[root@huatech ~]# lvremove /dev/sharevg/wmmlv
```

十八、版本控制服务器

18.1 SVN 服务器搭建和使用（一）

windows 操作系统下面的二进制文件包一共有 5 种, 如图:

Windows

-  [CollabNet](#) (supported and certified by [CollabNet](#); *requires registration*)
- [SlikSVN](#) (32- and 64-bit client MSI; maintained by [Bert Huijben](#), [SharpSvn project](#))
- [VisualSVN](#) (client and server; supported and maintained by [VisualSVN](#))
- [WANDisco](#) (32- and 64-bit client and server; supported and certified by [WANDisco](#))
- [Win32Svn](#) (32-bit client, server and bindings, MSI and ZIPs; maintained by [David Dar](#))

个人认为最好用 VisualSVN server 服务端和 TortoiseSVN 客户端搭配使用。
点开上面的 VisualSVN 连接, 下载 VisualSVN server, 如图:

The screenshot shows the VisualSVN website's download section. It features three main download links:

- Apache Subversion command line tools**: Version 1.7.4, Size: ~2 MB. Includes a link to "Read more about Subversion command line tools".
- VisualSVN for Visual Studio**: Version 2.5.4, Size: ~4 MB. Includes a link to "Learn more about VisualSVN integration for Visual Studio".
- VisualSVN Server**: Version 2.5.4, Size: ~4 MB. Includes a link to "Learn more about VisualSVN Server for Windows".

At the bottom, there is a note: "Apache Subversion, Subversion and the Apache Subversion logo are either registered trademarks or trademarks of The Apache Software Foundation. Microsoft, Windows and Visual Studio are either registered trademarks or trademarks of Microsoft Corporation." The footer includes copyright information (© 2005-2012 VisualSVN Limited), terms of service, and a VisualSVN logo.

然后下载 TortoiseSVN 客户端, 官网下载地址:[http://tortoisenv.net/downloads.html](http://tortoisevn.net/downloads.html)

The screenshot shows the TortoiseSVN website's download section. It features three main download links:

- SVN 1.7**: The Free and Most Feature-Rich Automated Subversion 1.7 Client. Includes a link to www.vercule.com.
- File Sync Software**: Sync Or Copy Folders Over Network. No Malware/Spyware. Download Free!. Includes a link to www.AllwaySync.com.
- StiCode**: Free SVN (Subversion) Repository Wiki, tickets and files hosting. Includes a link to www.sticode.com.

Below the links are navigation arrows (left, right, AdChoices), and a note: "The current version 1.7.6 is linked against the Subversion library 1.7.4." There is also a note: "Please make sure that you choose the right installer for your PC, otherwise the setup will fail."

for 32-bit OS [Download TortoiseSVN 1.7.6 - 32-bit](#) **for 64-bit OS** [Download TortoiseSVN 1.7.6 - 64-bit](#)

To verify the file integrity follow [these instructions](#)

注意下载跟你电脑匹配的安装包, 在页面的下面你还可以找到语言包, 如图:

Language packs

Country	32 Bit	64 Bit	Separate manual (PDF)
1 Arabic	Setup	Setup	Translate to Arabic
2 Bulgarian	Setup	Setup	Translate to Bulgarian
3 Catalan	Setup	Setup	Translate to Catalan
4 Chinese, simplified	Setup	Setup	T SVN TMerge
5 Chinese, traditional	Setup	Setup	Translate to trad. Chinese
6 Croatian	Setup	Setup	Translate to Croatian
7 Czech	Setup	Setup	T SVN TMerge
8 Danish	Setup	Setup	Translate to Danish

下载完成后, 应该有这些安装包, 如图:

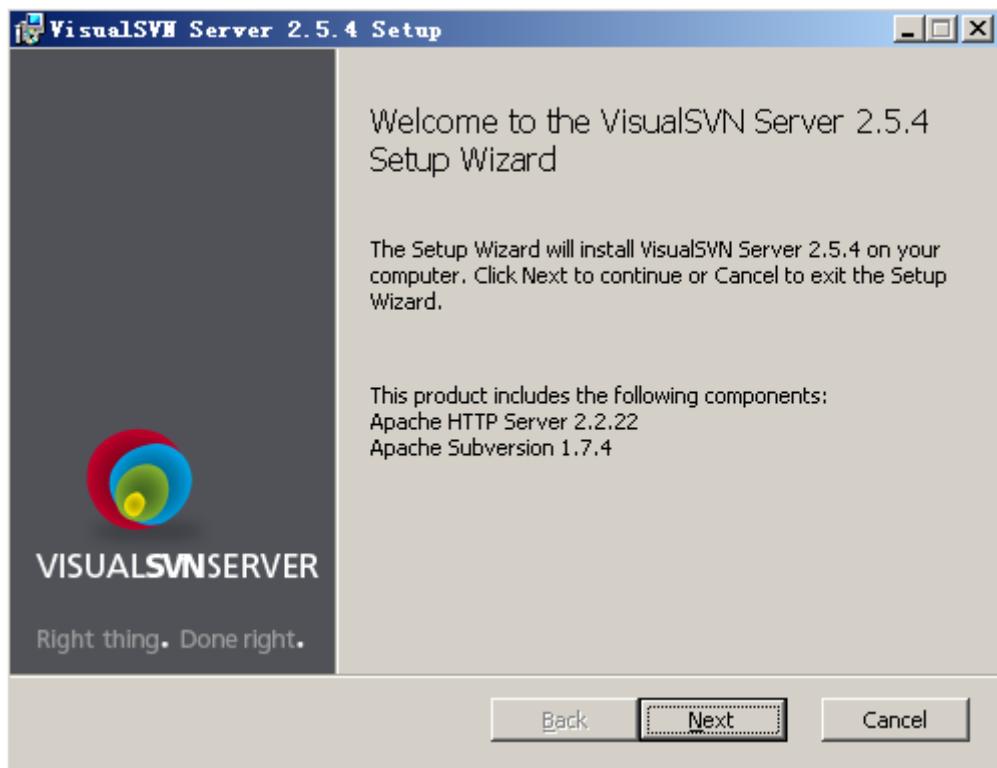
	3,708 KB
	13,648 KB

TortoiseSVN 安装包和简体中文语言包

	4,664 KB
--	----------

VisualSVN server 安装包

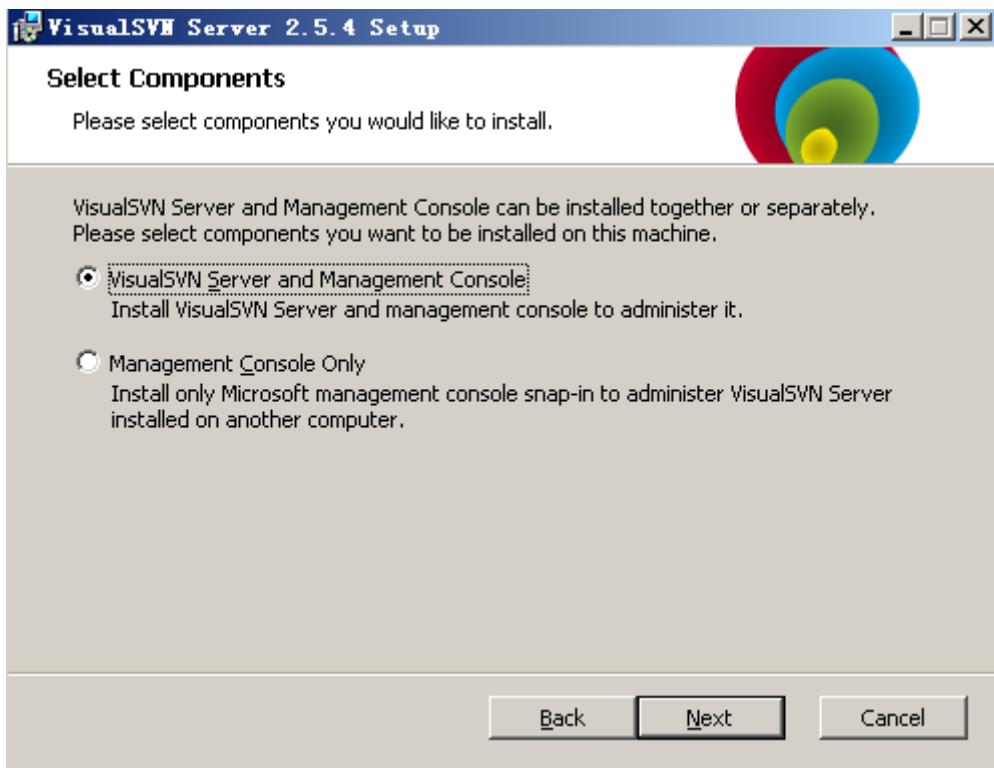
先安装 VisualSVN server 的安装包, 双击 VisualSVN server 安装包, 如图:



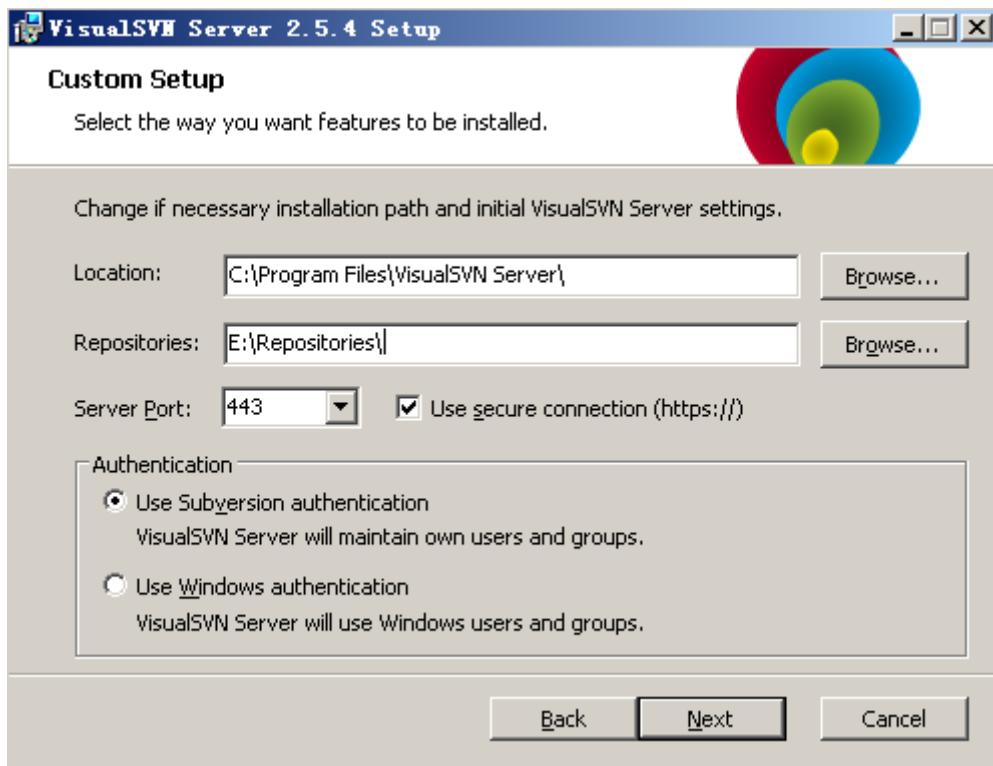
点 Next, 进入下一步, 如图:



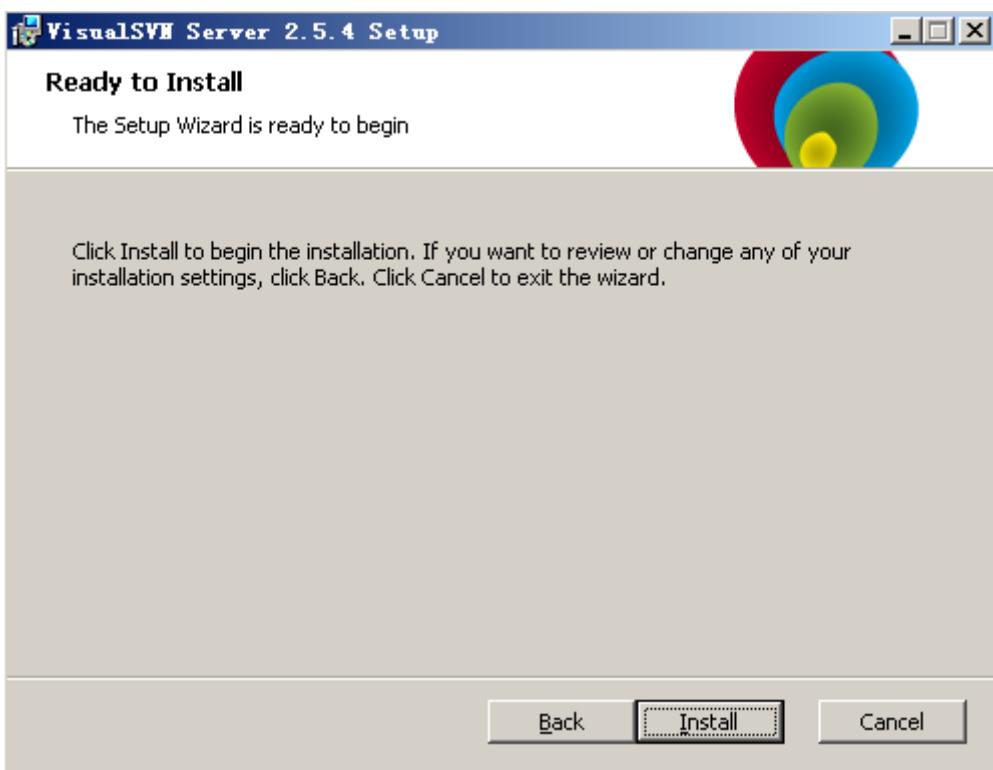
点同意, 进图下一步, 如图:



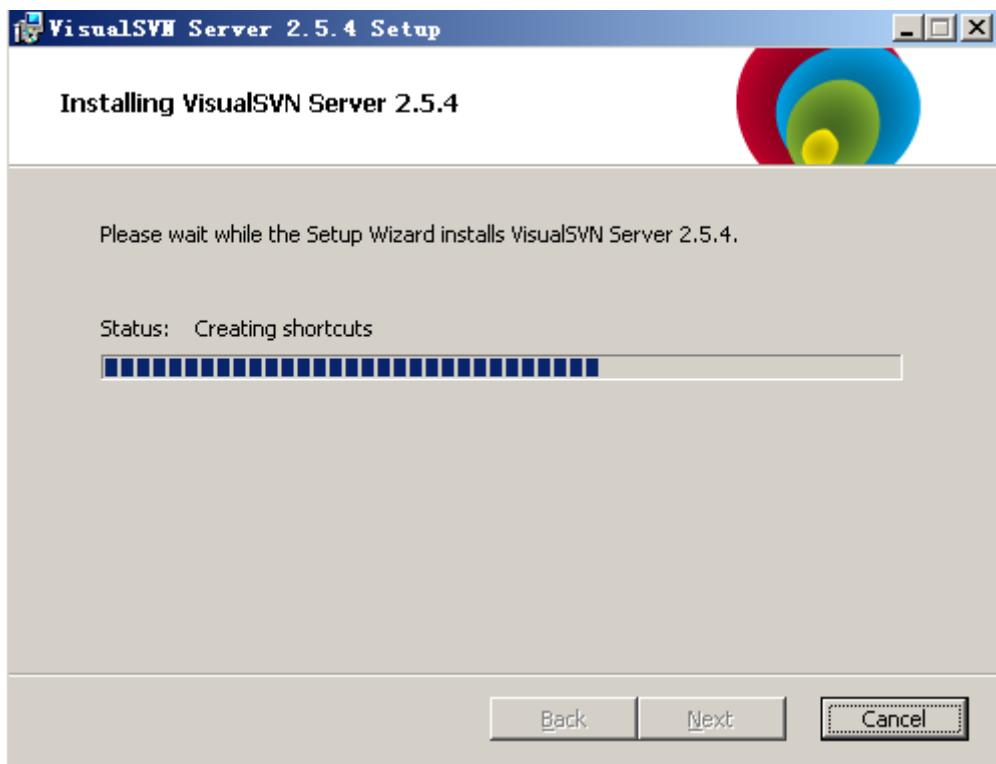
选择上面一个选项, 点 Next, 进入下一步, 如图:



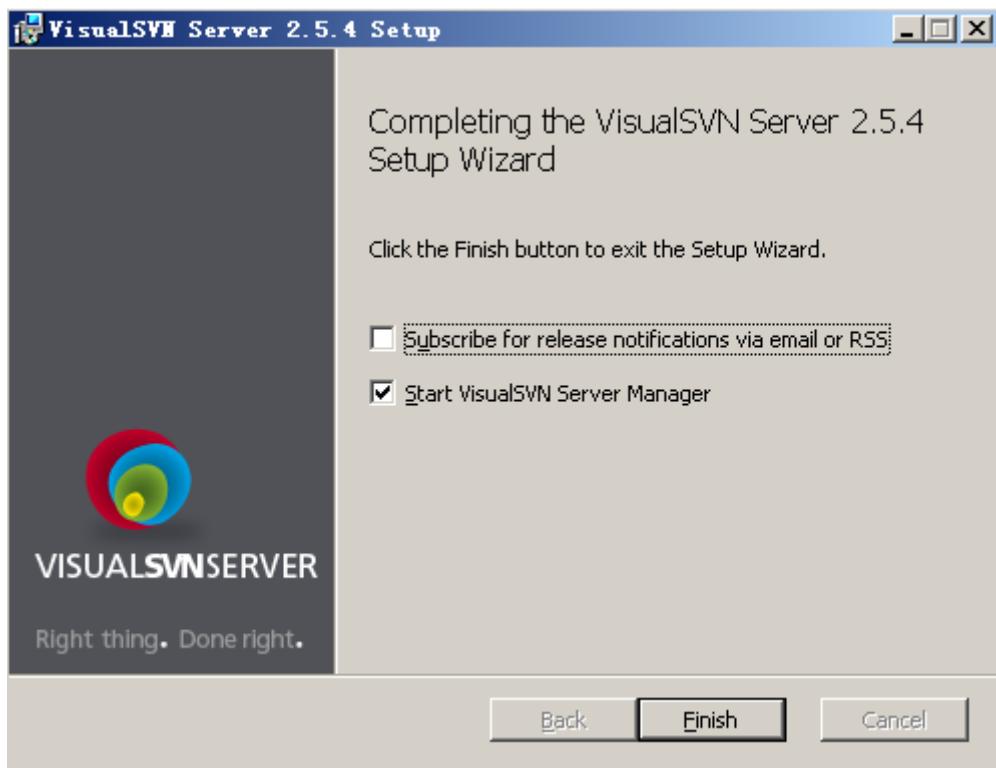
Location 是指 VisualSVN Server 的安装目录, Repositories 是指定你的版本库目录. Server Port 指定一个端口, Use secure connection 勾选表示使用安全连接, Use Subversion authentication 表示使用 Subversion 自己的用户认证. 点击 Next, 进入下一步, 如图:



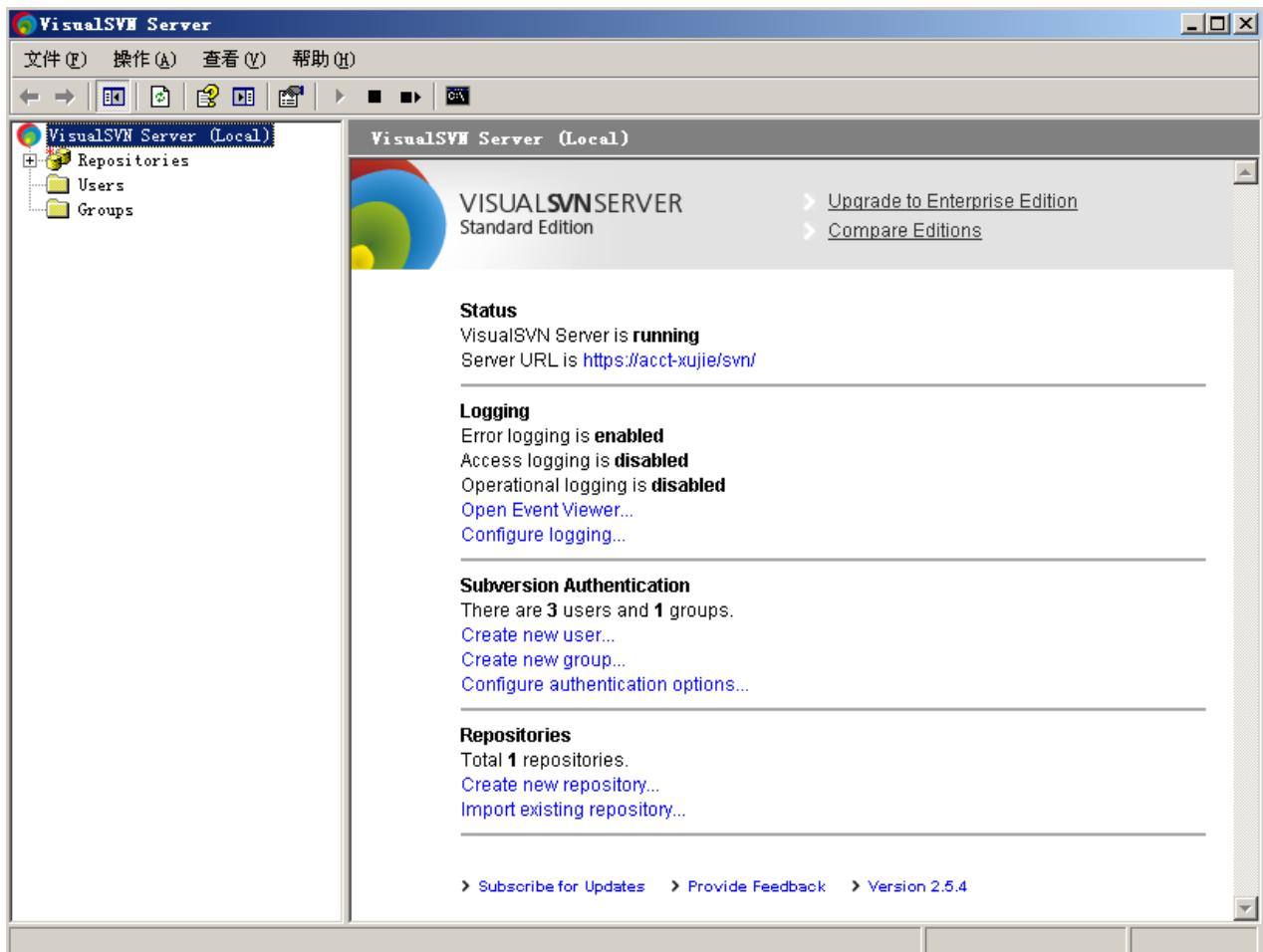
点 Install, 进入下一步, 如图:



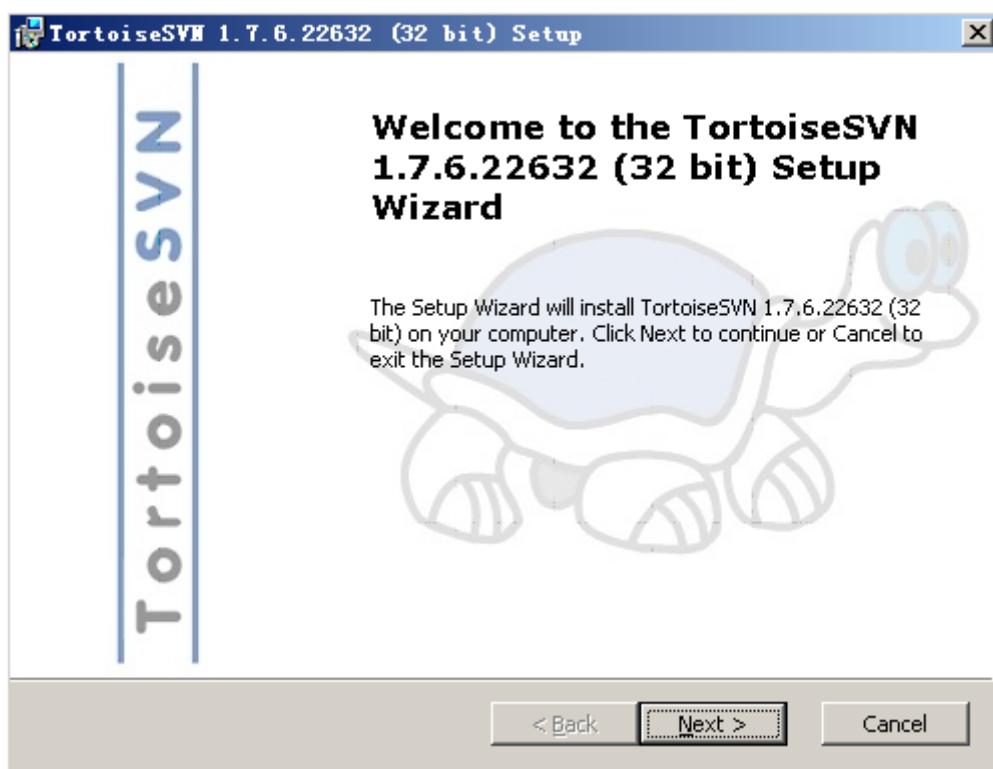
等待安装完成, 如图:



安装完成后, 启动 VisualSVN Server Manager, 如图:



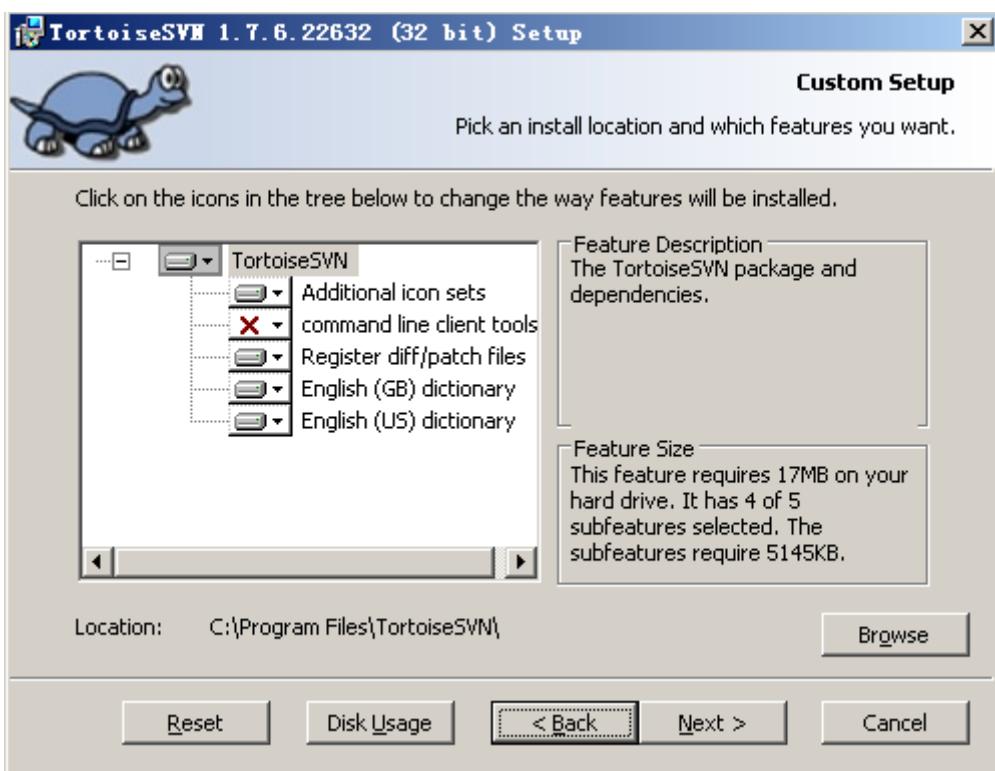
接下来我们安装 TortoiseSVN, 双击安装包, 进入下一步. 如图:



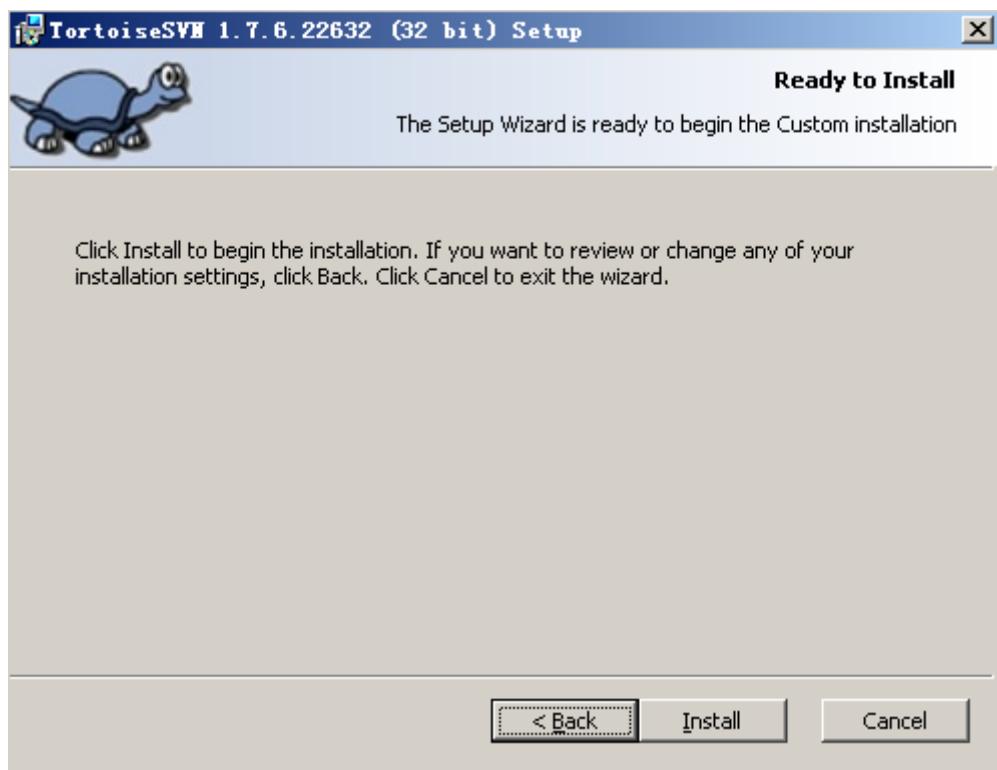
点击 Next, 进入下一步, 如图:



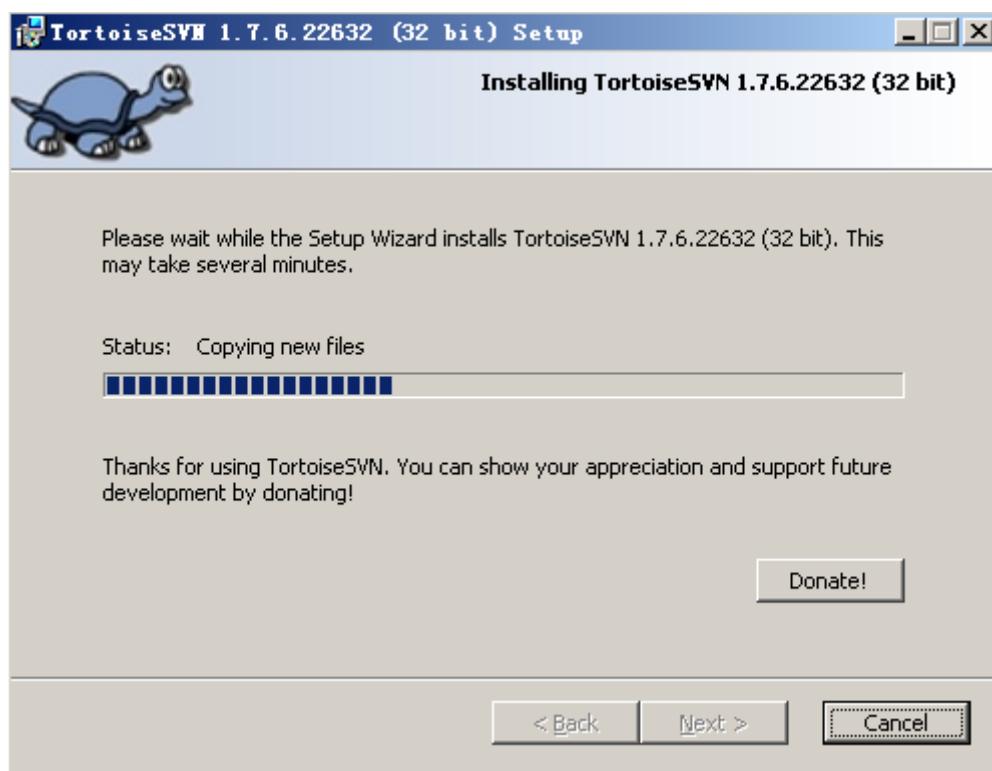
选择接受, 然后点击 Next, 进入下一步, 如图:



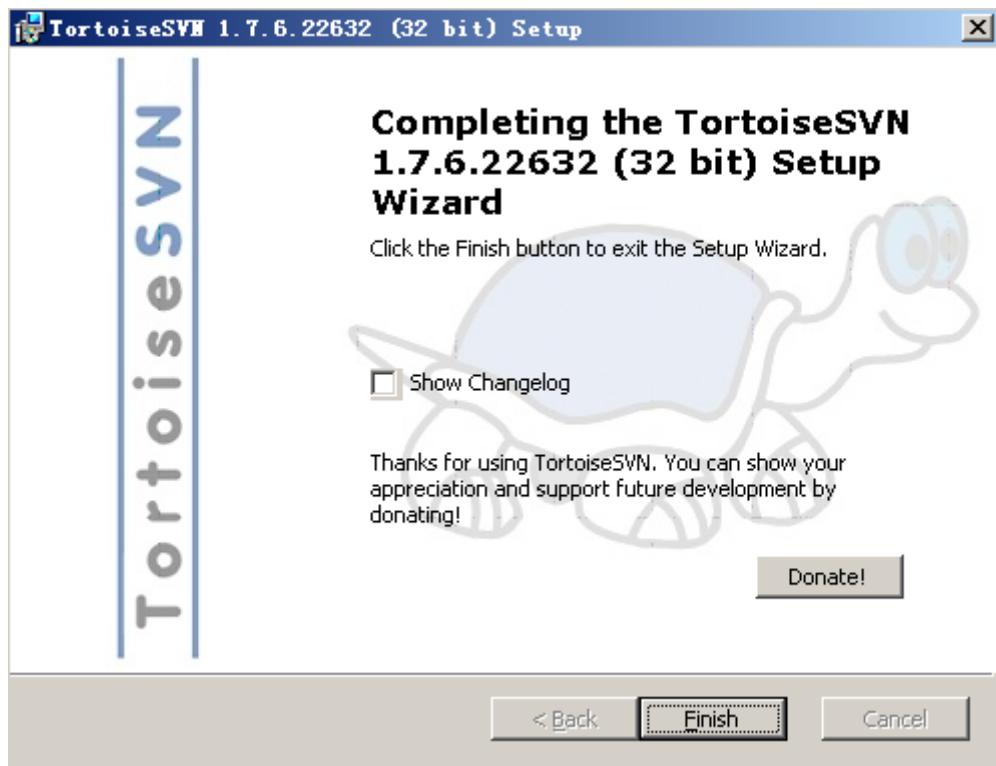
选择安装路径, 然后点击 Next, 进入下一步, 如图:



点击 **Install**, 开始安装, 如图:



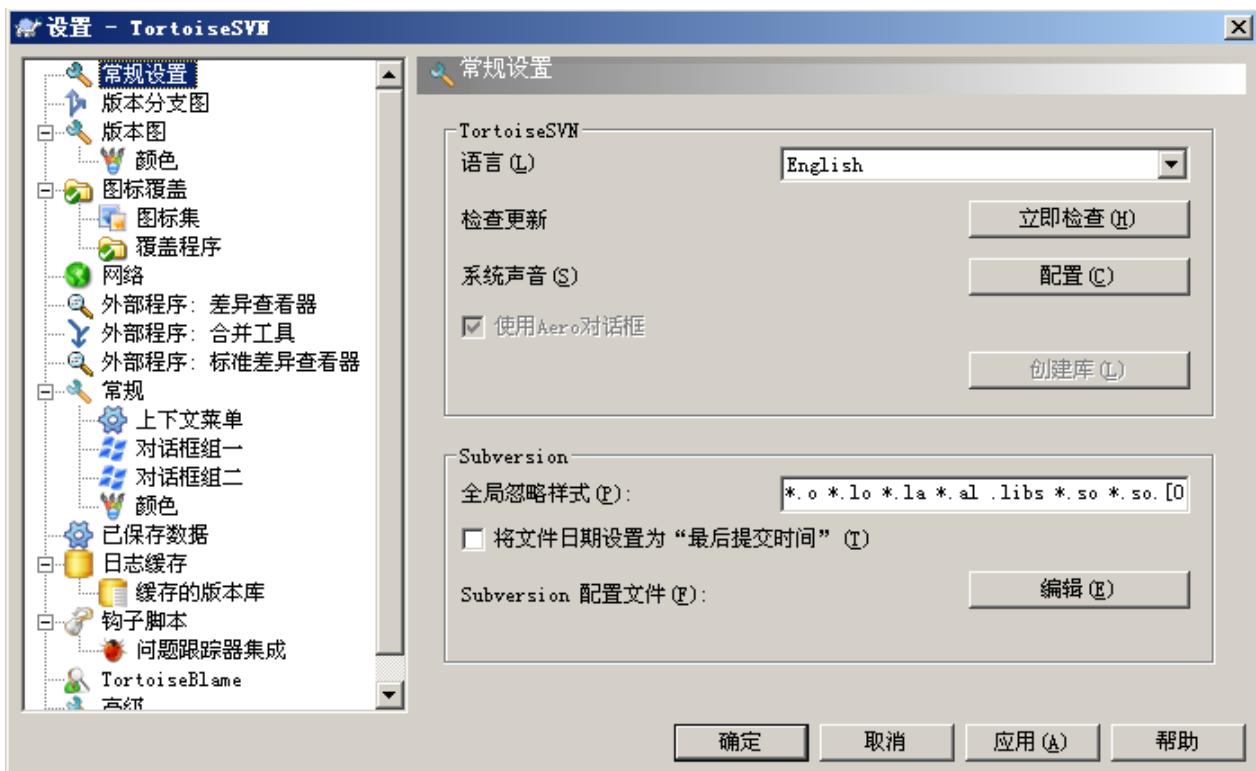
等待安装完成, 安装完成后如图:



接下来我们安装简体中文语言包，这个非常简单，一路 Next 就行，就不截图了。语言包安装完成以后在桌面任意空白地方单击鼠标右键，会在右键菜单里找到 SVN，如图：



选择设置，进入下一步，如图：



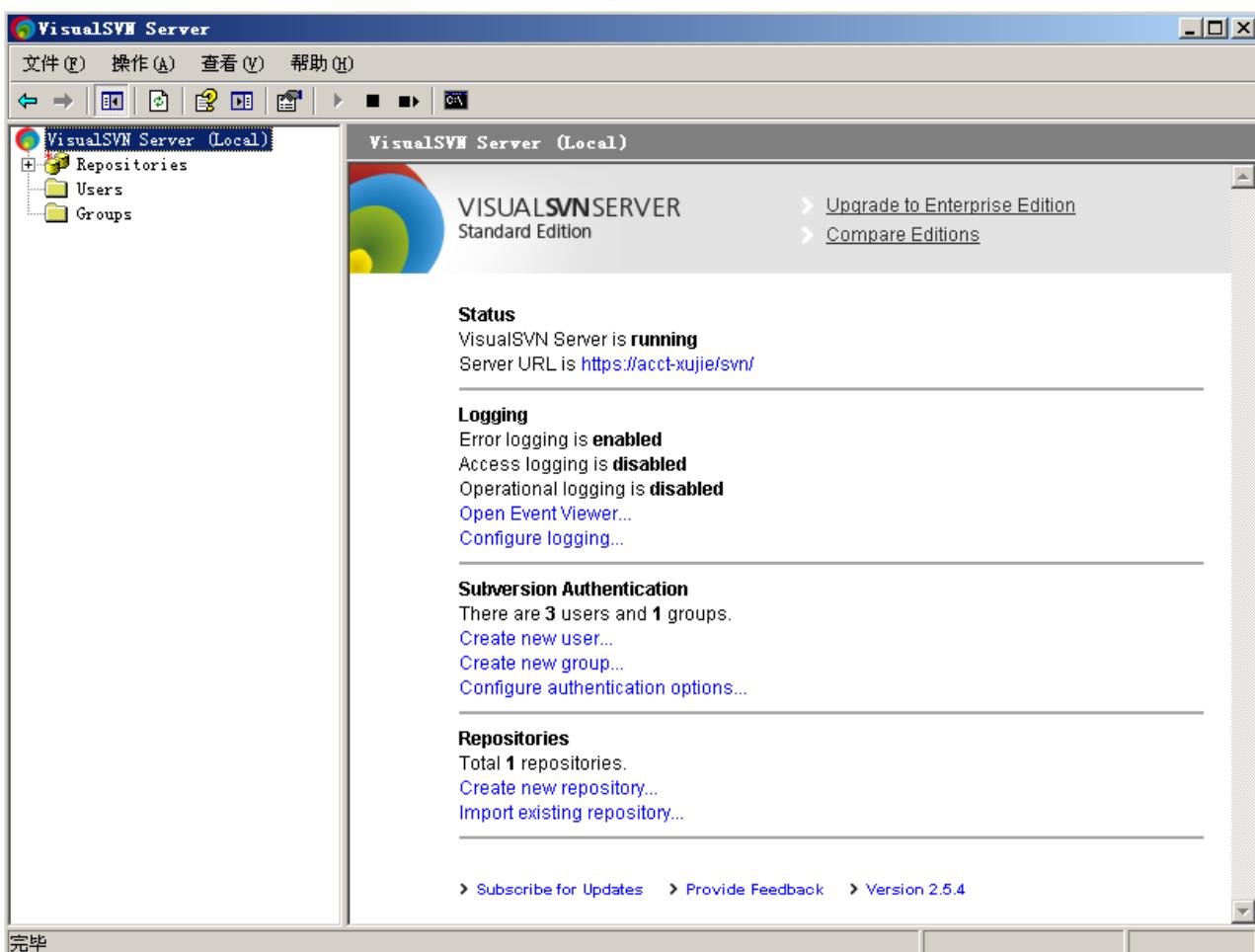
在右边的语言里面选择简体中文, 然后点击应用, 确定, 汉化即完成, 如图:



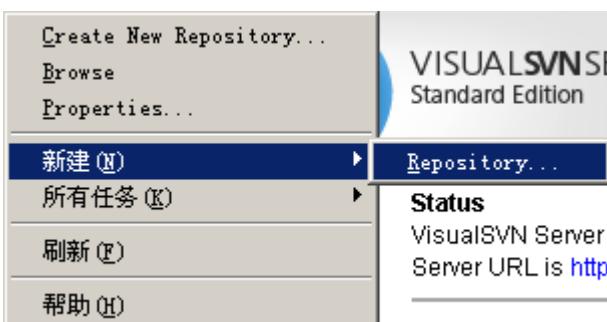
到这里, 服务端和客户端都已经安装完毕, 下一篇介绍用 VisualSVN Server 创建版本库, 以及 TortoiseSVN 的使用

18.2 SVN 服务器搭建和使用（二）

首先打开 VisualSVN Server Manager, 如图:



可以在窗口的右边看到版本库的一些信息,比如状态,日志,用户认证,版本库等.要建立版本库,需要右键单击左边窗口的 Repositories,如图:



在弹出的右键菜单中选择 Create New Repository 或者新建->Repository,进入下一步:



输入版本库名称, 勾上 Create default structure 复选框(推荐这么做). 点击 OK, 版本库就创建好了, 版本库中会默认建立 trunk, branches, tags 三个文件夹, 如图:



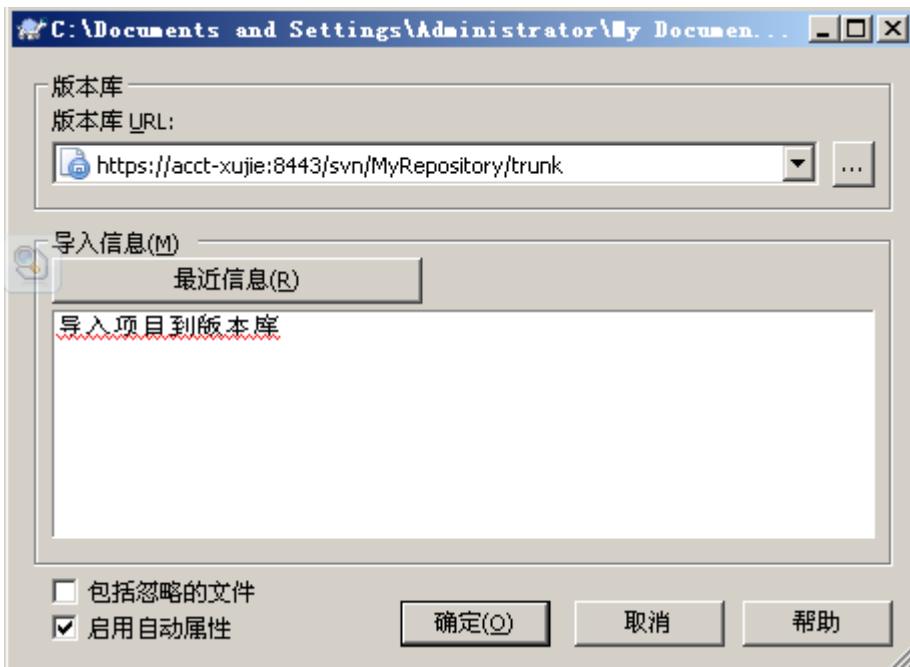
这时候我们将项目导入到版本库中, 找到你的项目文件夹, 在项目文件夹上点击鼠标右键, 找到 SVN 菜单, 选择导入, 如图:



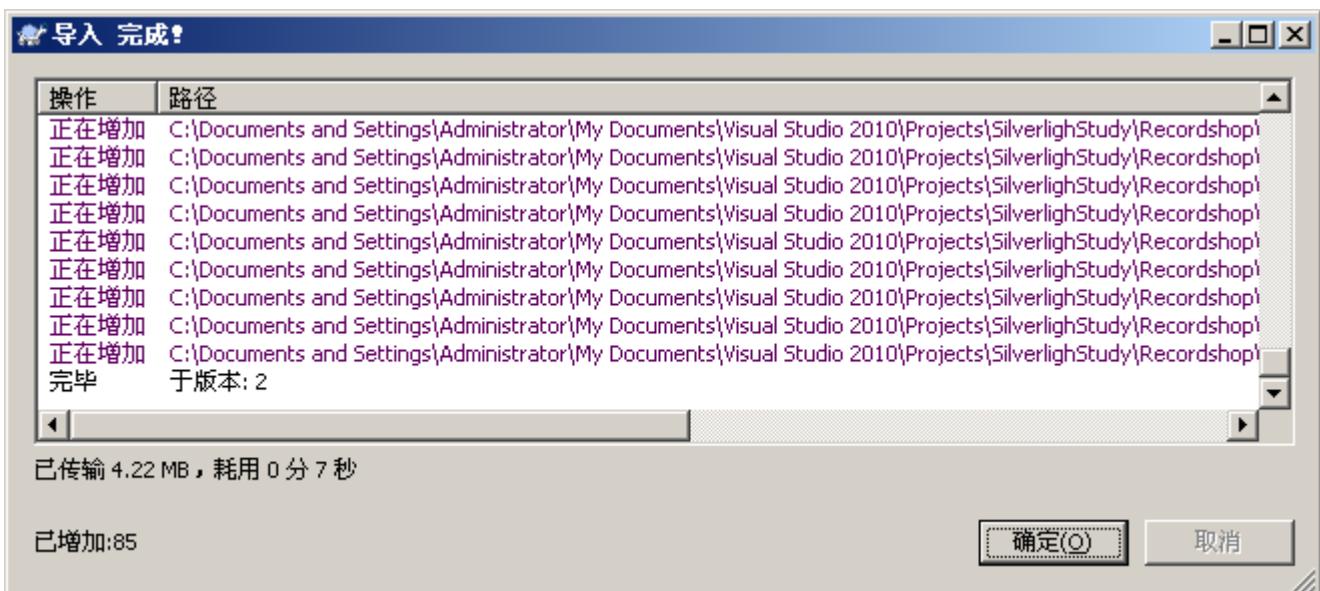
在弹出的对话框中填上版本库 URL, 这个 URL 可以从 VisualSVN Server Manager 中获取, 在你的版本库上单击右键, 选择 Copy URL to Clipboard, 这样就把版本库 URL 复制到你的剪贴版了. 如图:



将复制的版本库 URL 粘贴上, 在 URL 后面加上 trunk 子路径. 然后在导入信息里面填上导入信息 "导入项目到版本库". 如图:



点击确定, 所选中的项目就会被导入到版本库中. 如图:



项目导入到版本库以后, 不能随便让谁都能够读写版本库, 所以需要建立用户组和用户.

在 VisualSVN Server Manager 窗口的左侧右键单击用户组, 选择 Create User 或者新建->User, 如图:



在弹出的对话框中填写 User name 和 Password, 然后点击 OK, 如图:



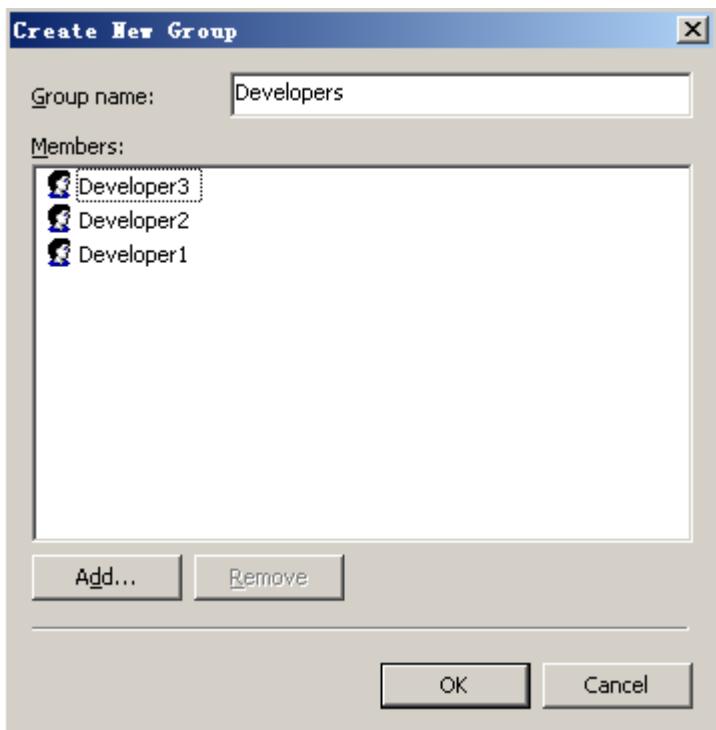
用相同的方式分别创建用户 Developer1, Developer2, Developer3, Tester1, Tester2, Manager 六个用户, 分别代表 3 个开发人员, 两个测试人员和一个项目经理, 如图:



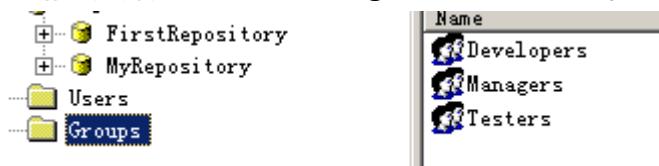
然后我们建立用户组, 在 VisualSVN Server Manager 窗口的左侧右键单击用户组, 选择 Create Group 或者新建->Group, 如图:



在弹出窗口中填写 Group name 为 Developers, 然后点 Add 按钮, 在弹出的窗口中选择三个 Developer, 加入到这个组, 然后点 Ok, 如图:



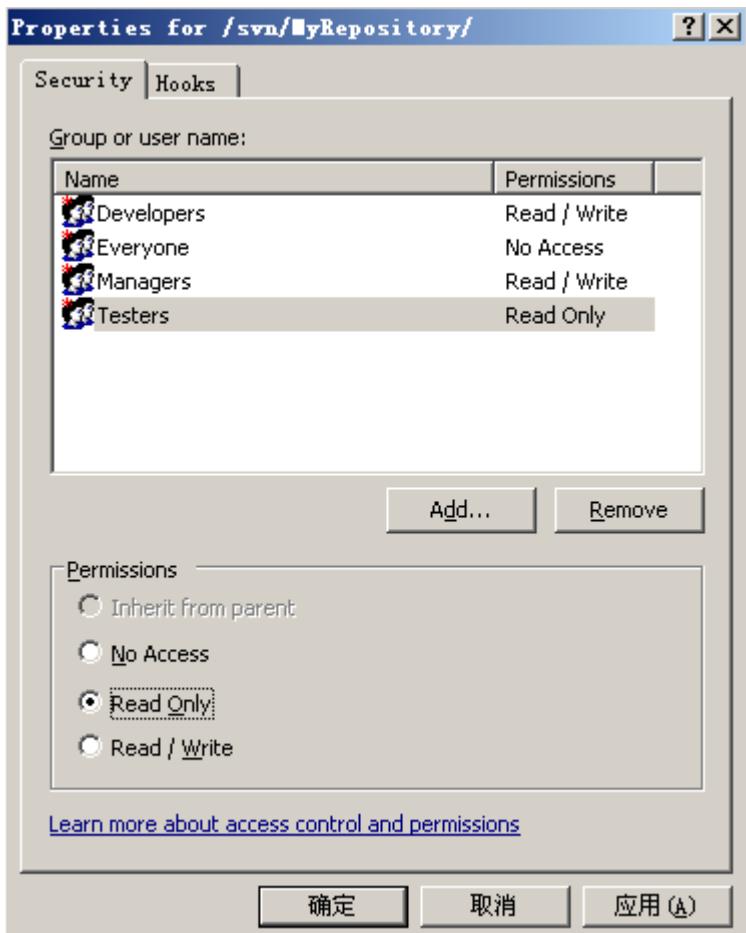
用相同的方式创建组 Managers, Testers, 如图：



接下来我们给用户组设置权限，在 MyRepository 上单击右键，选择属性，如图：

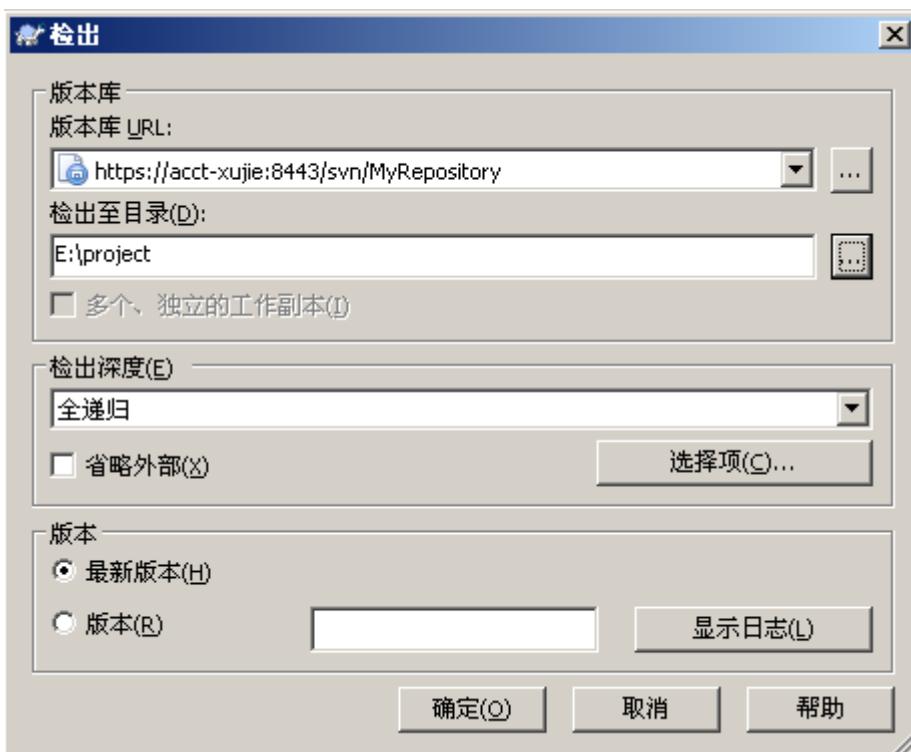


在弹出的对话框中，选择 Security 选项卡，点击 Add 按钮，选中 Developers, Managers, Testers 三个组，然后添加进来，给 Developers, Managers 权限设置为 Read/Write, Tester 权限设置为 Read Only, 如图：

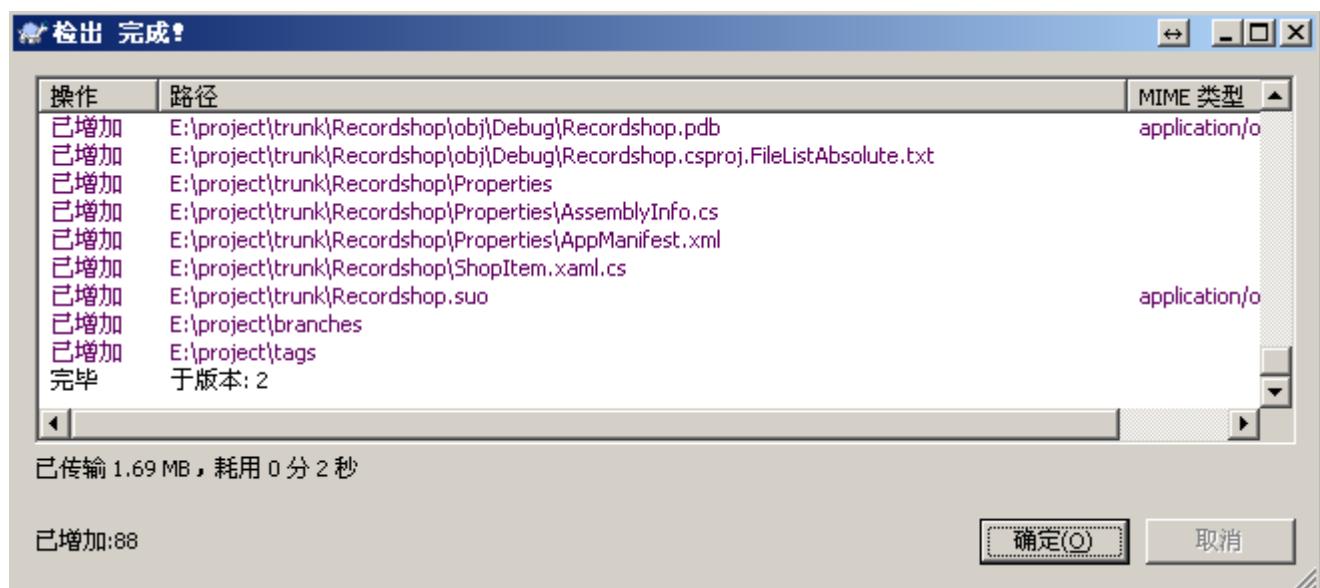


到此，服务端就完成了。

接下来，我们用客户端去检出代码，在桌面空白处单击右键，选择 SVN 检出，在弹出的对话框中填写版本库 URL（具体获取方式，上面讲上传项目到版本库的时候讲过），选择检出目录，点击确定。如图：



开始检出项目, 如图:



检出完成之后, 我们打开工作副本文件夹, 会看到所有文件和文件夹都有一个绿色的v. 如图:



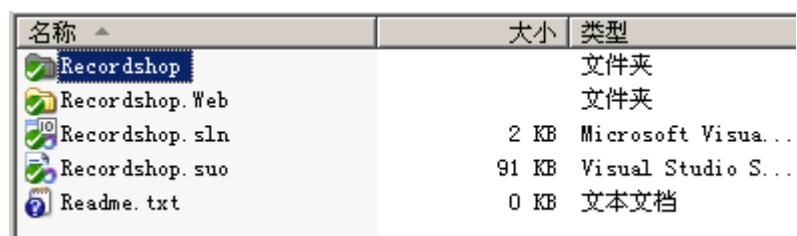
至此, 创建版本库和使用 TortoiseSVN 导入项目, 检出项目已经介绍完毕.

18.3 SVN 服务器搭建和使用 (三)

接下来, 试试用 TortoiseSVN 修改文件, 添加文件, 删除文件, 以及如何解决冲突等.

添加文件

在检出的工作副本中添加一个 `Readme.txt` 文本文件, 这时候这个文本文件会显示为没有版本控制的状态, 如图:



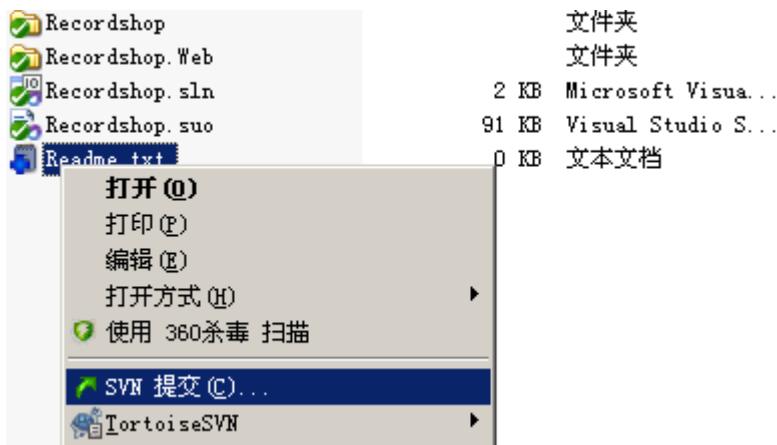
这时候, 你需要告知 TortoiseSVN 你的操作, 如图:



加入以后, 你的文件会变成这个状态, 如图:

名称	大小	类型
Recordshop		文件夹
Recordshop.Web		文件夹
Recordshop.sln	2 KB	Microsoft Visua...
Recordshop.suo	91 KB	Visual Studio S...
Readme.txt	0 KB	文本文档

这时候使用 TortoiseSVN 进行提交. 这样别人就能看到你所做的更改了, 如图.



修改文件

使用 TortoiseSVN 更新, 修改工作副本中的 Readme.txt 文件, 加入"hello world!", 然后保存, 你会发现 Readme.txt 文件的图标改变了, 如图:

名称	大小	类型
Recordshop		文件夹
Recordshop.Web		文件夹
Readme.txt	1 KB	文本文档
Recordshop.sln	2 KB	Microsoft Visua...
Recordshop.suo	91 KB	Visual Studio S...

这个红色的叹号代表这个文件被修改了, 这时候, 提交更改, 其他人即可看到你的更改.

重命名文件

使用 TortoiseSVN 更新, 重命名工作副本中的 Readme.txt 文件为 "Readme1.txt", 然后保存, 你会发现 Readme.txt 文件的图标改变了, 如图:

名称	大小	类型
Recordshop		文件夹
Recordshop.Web		文件夹
Readme1.txt	1 KB	文本文档
Recordshop.sln	2 KB	Microsoft Visua...
Recordshop.suo	91 KB	Visual Studio S...

更添加文件一个道理, 这时候你需要告诉 TortoiseSVN 你的操作, 如图:



加入以后, 提交, 这时候版本库中的 Readme.txt 文件将会被重命名为 "Readme1.txt".

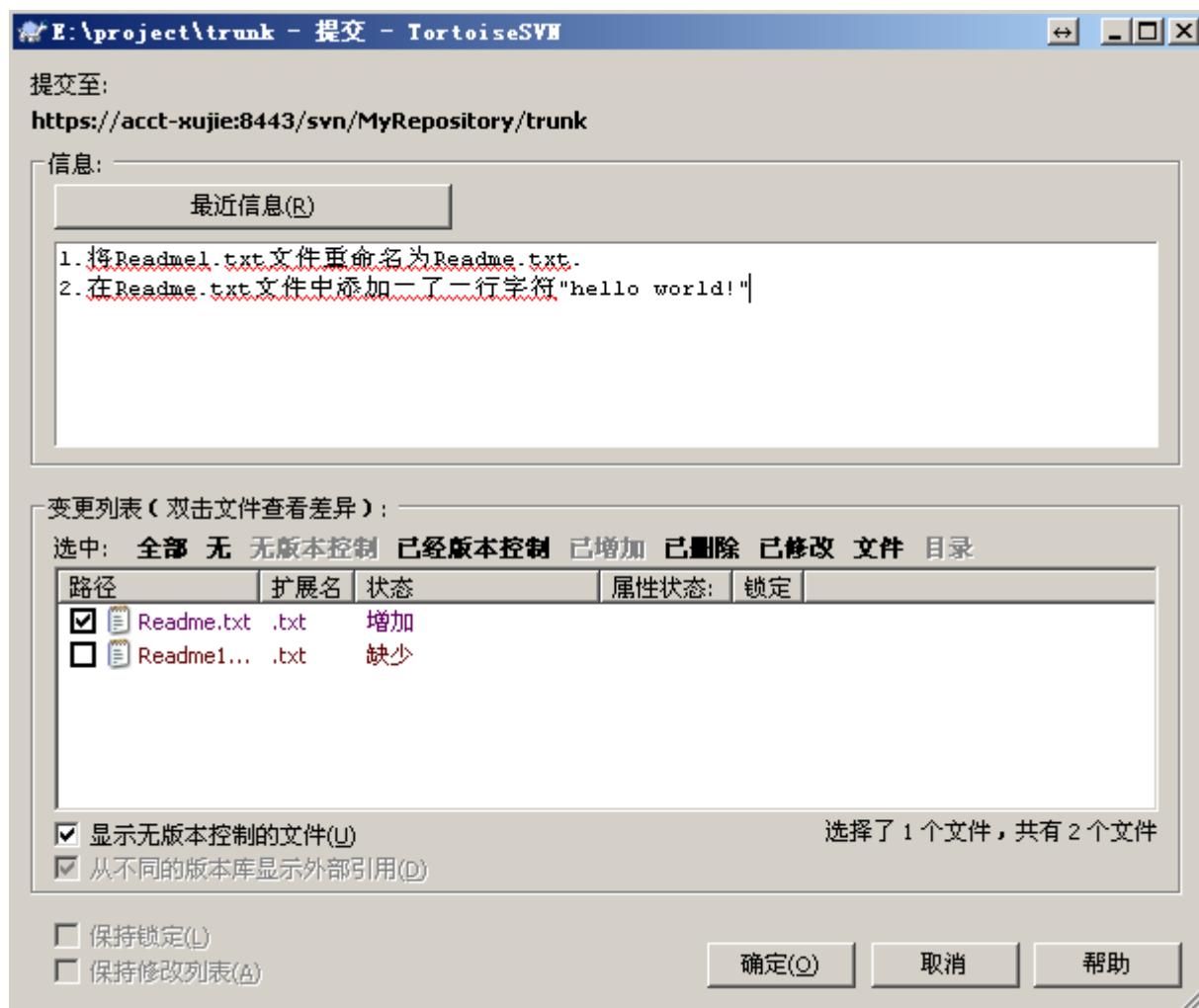
删除文件

使用 TortoiseSVN 更新, 使用 TortoiseSVN 删除工作副本中的 Readme.txt 文件, 然后提交, 版本库中的相应文件即被删除掉了, 如图:



强制写注释

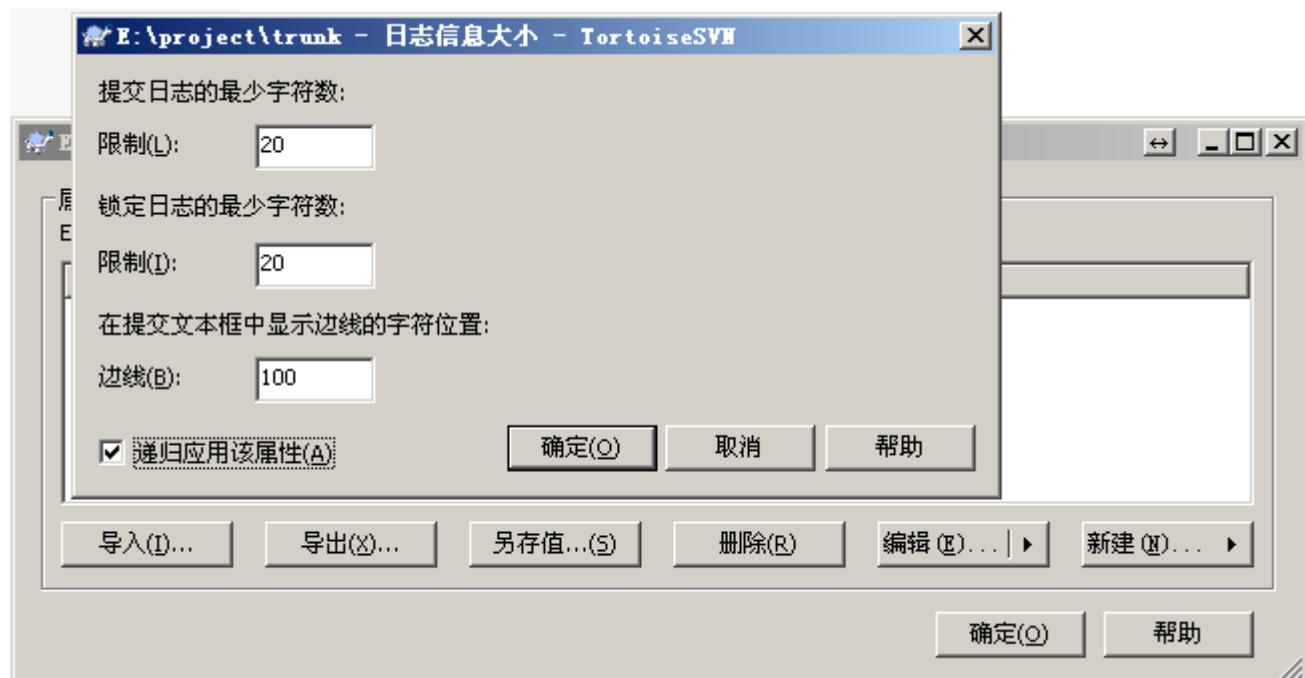
为了以后你能更清晰的看到你所做的每一次更改的原因, 你在每次提交的时候应该写上注释, 而且尽量详细. 如图:



但是, 可能有的人因为觉得太繁琐, 而不填写注释, 这不利于以后的版本控制, 可以将强制在提交的时候写注释, 首先单击右键, 选择 TortoiseSVN->属性, 如图:



在弹出的对话框中, 点击新建->日志大小, 设置提交日志的最小字符数和锁定日志的最小字符数为 20, 提交文本框中显示边线的字符位置设置为 100, 点击确定, 如图:



提交, 以后再次提交的时候, 如果输入的注释少于 20 个字符, 将无法提交.

冲突解决

冲突问题是常见的问题, 它是这样产生的, A 用户 check out 了一个工作副本 A, 接着 B 用户又 check out 了一个工作副本 B. 然后 A 用户对副本 A 中的文件 C 做了修改(可以是内容修改, 文件删除, 重命名, 以及位置移动), 并且提交. 这时候 B 用户也对文件 C 的相同部分做了修改, 这时候如果 B 用户进行提交, 会先被告知版本过时, 要求更新, 然后更新的时候会提示冲突了, 这时候可以用冲突编辑器进行手动选择。

18.4 Linux 版本控制服务器配置

```
[root@huatech ~]# yum -y install subversion  
[root@huatech ~]# mkdir -p /var/svn/repos/project  
[root@huatech ~]# svnadmin create /var/svn/repos/project
```

十九、FTP 服务器大全

19.1 VSFTPD

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

这里普及一下 YUM 的用法和基本参数

install	Install a package or packages on your system # yum install package
update	Update a package or packages on your system # yum update package
remove	Remove the packages # yum remove package
groupinstall	Install the packages in a group on your system # yum groupinstall "X Window System"
groupupdate	Update the packages in a group on your system # yum groupupdate "X Window System"
groupremove	Remove the packages in a group from your system # yum groupremove "X Window System"
list	List a package or groups of packages # yum list ⇒ display packages that is possible to install # yum list installed ⇒ display installed packages
check-update	Check for available package updates # yum check-update
info	Display details about a package or group of packages # yum info package
search	Search package details for the given string # yum search keyword
deplist	List a package's dependencies # yum deplist package
-y	answer yes for all questions # yum -y install package
--enablerepo=repo	enable one or more repositories (wildcards allowed) # yum --enablerepo=repo install package
--disablerepo=repo	disable one or more repositories (wildcards allowed) # yum --disablerepo=repo install package
--exclude=package	exclude package(s) by name or glob # yum --exclude=package update

--nopugins	disable Yum plugins # yum --nopugins update
------------	--

```
[root@huatech ~]# vim /etc/vsftpd/vsftpd.conf
```

```
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
```

```
anonymous_enable=YES
```

```
# Uncomment this to allow local users to log in.
```

```
local_enable=YES
```

```
# Uncomment this to enable any form of FTP write command.
```

```
write_enable=YES
```

```
local_root=public_html
```

```
use_localtime=YES
```

```
seccomp_sandbox=NO
```

```
[root@huatech ~]# vim /etc/vsftpd/chroot_list
```

```
[root@huatech ~]# systemctl restart vsftpd
```

```
[root@huatech ~]# systemctl enable vsftpd
```

```
=====Pfoftpd=====
```

```
[root@huatech ~]# yum --enablerepo=epel -y install proftpd
```

```
[root@huatech ~]# vim /etc/proftpd.conf
```

```
[root@huatech ~]# vim /etc/proftpd.conf
```

```
[root@huatech ~]# vim /etc/ftpusers
```

```
[root@huatech ~]# systemctl start proftpd
```

```
[root@huatech ~]# systemctl enable proftpd
```

```
=====Pure-ftpd=====
```

```
[root@huatech ~]# yum --enablerepo=epel -y install pure-ftpd
```

```
[root@huatech ~]# vim /etc/pure-ftpd/pure-ftpd.conf
```

```
[root@huatech ~]# systemctl start pure-ftpd
```

```
[root@huatech ~]# systemctl enable pure-ftpd
```

```
=====Ftp Client=====
```

Ftp Client 1 下载地址: <https://filezilla-project.org/download.php?type=client>

Ftp Client 2 下载地址:

http://dl.pconline.com.cn/html_2/1/89/id=61&pn=0&linkPage=1.html

```
=====Vsftpd+TLS=====
```

```
[root@huatech ~]# cd /etc/pki/tls/certs/
```

```
[root@huatech certs]# openssl req -x509 -nodes -newkey rsa:2048 -keyout vsftpd.pem -out vsftpd.pem -days 365
```

```
Country Name (2 letter code) [XX]:CH
```

```
State or Province Name (full name) []:HN
```

```
Locality Name (eg, city) [Default City]:ZZ
```

```
Organization Name (eg, company) [Default Company Ltd]:RDH
```

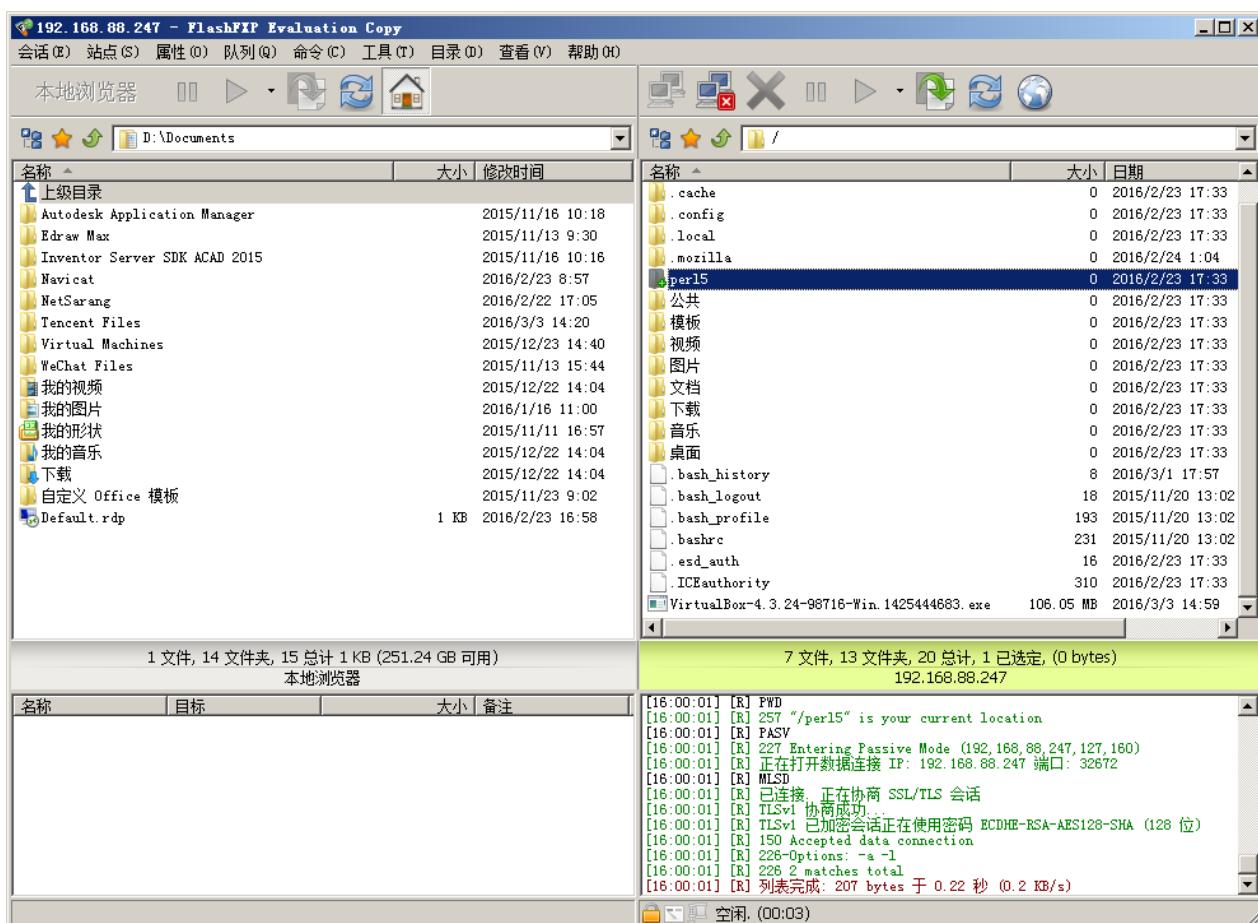
```
Organizational Unit Name (eg, section) []:IT
```

```
Common Name (eg, your name or your server's hostname) []:huatech.com
```

```
Email Address []:15136278534@163.com
```

```
[root@huatech certs]# chmod 400 vsftpd.pem
```

```
[root@huatech ~]# vim /etc/vsftpd/vsftpd.conf
local_root=public_html
rsa_cert_file=/etc/pki/tls/certs/vsftpd.pem
ssl_enable=YES
force_local_data_ssl=YES
force_local_logins_ssl=YES
=====Proftp+TLS=====
[root@huatech ~]# cd /etc/pki/tls/certs/
[root@huatech certs]# openssl req -x509 -nodes -newkey rsa:2048 -keyout proftpd.pem
-out proftpd.pem -days 365
Country Name (2 letter code) [XX]:CH
State or Province Name (full name) []:HN
Locality Name (eg, city) [Default City]:ZH
Organization Name (eg, company) [Default Company Ltd]:RDH
Organizational Unit Name (eg, section) []:IT
Common Name (eg, your name or your server's hostname) []:huatech.com
Email Address []:151@163.com
[root@huatech certs]# chmod 600 proftpd.pem
[root@huatech ~]# vim /etc/proftpd.conf
TLSEngine          on
TLSRequired        on
TLSProtocol       SSLv23
TLSLog             /var/log/proftpd/tls.log
TLSRSACertificateFile /etc/pki/tls/certs/proftpd.pem
TLSRSACertificateKeyFile /etc/pki/tls/certs/proftpd.pem
=====Pure-ftpd+TLS=====
[root@huatech ~]# cd /etc/pki/tls/certs/
[root@huatech certs]# openssl req -x509 -nodes -newkey rsa:2048 -keyout
pure-ftpd.pem -out pure-ftpd.pem -days 365
[root@huatech certs]# vim /etc/pure-ftpd/pure-ftpd.conf
TLS               1
[root@huatech ~]# systemctl restart pure-ftpd
[root@huatech ~]# chkconfig pure-ftpd on
```



二十、日志服务器

【服务器端】

```
[root@huatech ~]# vim /etc/rsyslog.conf
$AllowedSender TCP, 127.0.0.1, 192.168.88.0/24, *.rdh.com
```

```
$ModLoad imtcp
$InputTCPServerRun 514
[root@huatech ~]# systemctl restart rsyslog
【客户端】
*. * @@192.168.1.30:514
【用数据库承载日志】
[root@huatech ~]# yum -y install rsyslog-mysql
[root@huatech ~]# cat /usr/share/doc/rsyslog-7.4.7/mysql-createDB.sql | mysql -u root -p
MariaDB [(none)]> grant all privileges on Syslog.* to rsyslog@'localhost'
identified by 'password';
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.01 sec)

MariaDB [(none)]> exit;
[root@huatech ~]# vim /etc/rsyslog.conf
$ModLoad ommysql
authpriv.* :ommysql:localhost,Syslog,rsyslog,password
[root@huatech ~]# systemctl restart rsyslog
[root@huatech ~]# mysql -u rsyslog -p Syslog
```

二十一、DHCP 服务器 (PXE-SERVER)

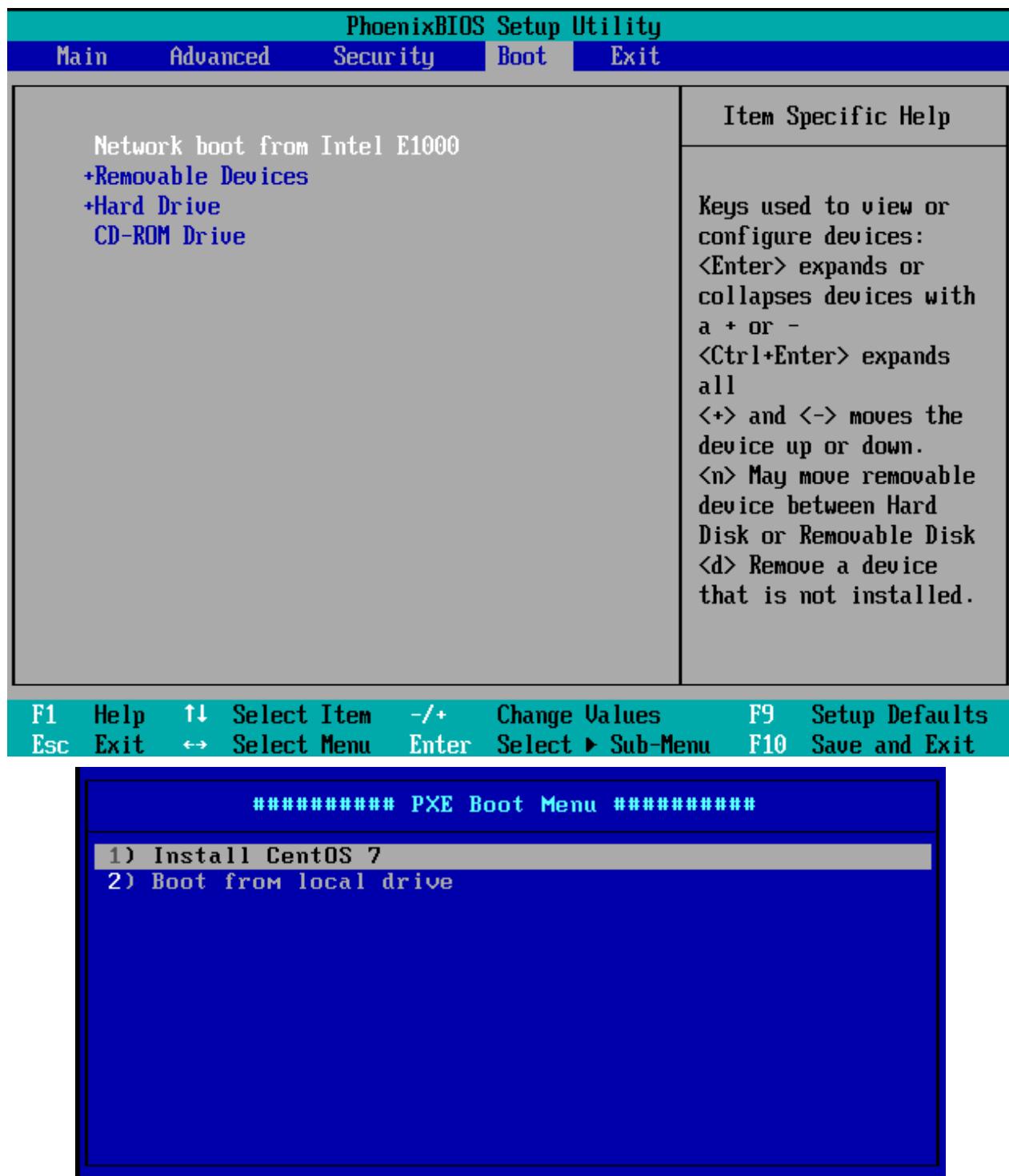
```
[root@huatech ~]# yum -y install dhcp
[root@huatech ~]# vim /etc/dhcp/dhcpd.conf
option domain-name "huatech.com";
default-lease-time 600;
max-lease-time 7200;
log-facility local7;
subnet 10.10.1.0 netmask 255.255.255.0 {
    range 10.10.10.1 10.10.1.155;
    option routers 10.10.10.1;
    option domain-name-servers 10.10.10.1;
}
[root@huatech ~]# systemctl restart dhcpd
[root@huatech ~]# systemctl enable dhcpd
[root@huatech ~]# yum -y install syslinux xinetd tftp-server
[root@huatech ~]# yum -y install syslinux xinetd tftp-server
[root@huatech ~]# mkdir /var/lib/tftpboot/pxelinux.cfg
[root@huatech ~]# cp /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/
[root@huatech ~]# vim /etc/xinetd.d/tftp
disable = no
```

```
[root@huatech ~]# systemctl start xinetd
[root@huatech ~]# systemctl enable xinetd
[root@huatech ~]# vim /etc/dhcp/dhcpd.conf
filename      "pxelinux.0";
next-server    192.168.88.247;
[root@huatech ~]# systemctl restart dhcpd

[root@huatech ~]# mkdir -p /var/pxe/centos7
[root@huatech ~]# mkdir /var/lib/tftpboot/centos7
[root@huatech ~]# mount /dev/sr0 /var/pxe/centos7/
[root@huatech ~]# cp /var/pxe/centos7/images/pxeboot/vmlinuz
/var/lib/tftpboot/centos7/
[root@huatech ~]# cp /var/pxe/centos7/images/pxeboot/initrd.img
/var/lib/tftpboot/centos7/
[root@huatech ~]# cp /usr/share/syslinux/menu.c32 /var/lib/tftpboot/
[root@huatech ~]# vim /var/lib/tftpboot/pxelinux.cfg/default
timeout 100
default menu.c32

menu title ##### PXE Boot Menu #####
label 1
  menu label ^1) Install CentOS 7
  kernel centos7/vmlinuz
  append initrd=centos7/initrd.img method=http://192.168.88.247/centos7
  devfs=nomount

label 2
  menu label ^2) Boot from local drive
  localboot
[root@huatech ~]# vim /etc/httpd/conf.d/pxeboot.conf
alias /centos7 /var/pxe/centos7
<Directory /var/pxe/centos7>
  Options Indexes FollowSymLinks
  # IP address you allow to access
  Require ip 127.0.0.1 192.168.88.0/24
</Directory>
[root@huatech ~]# systemctl restart httpd
```

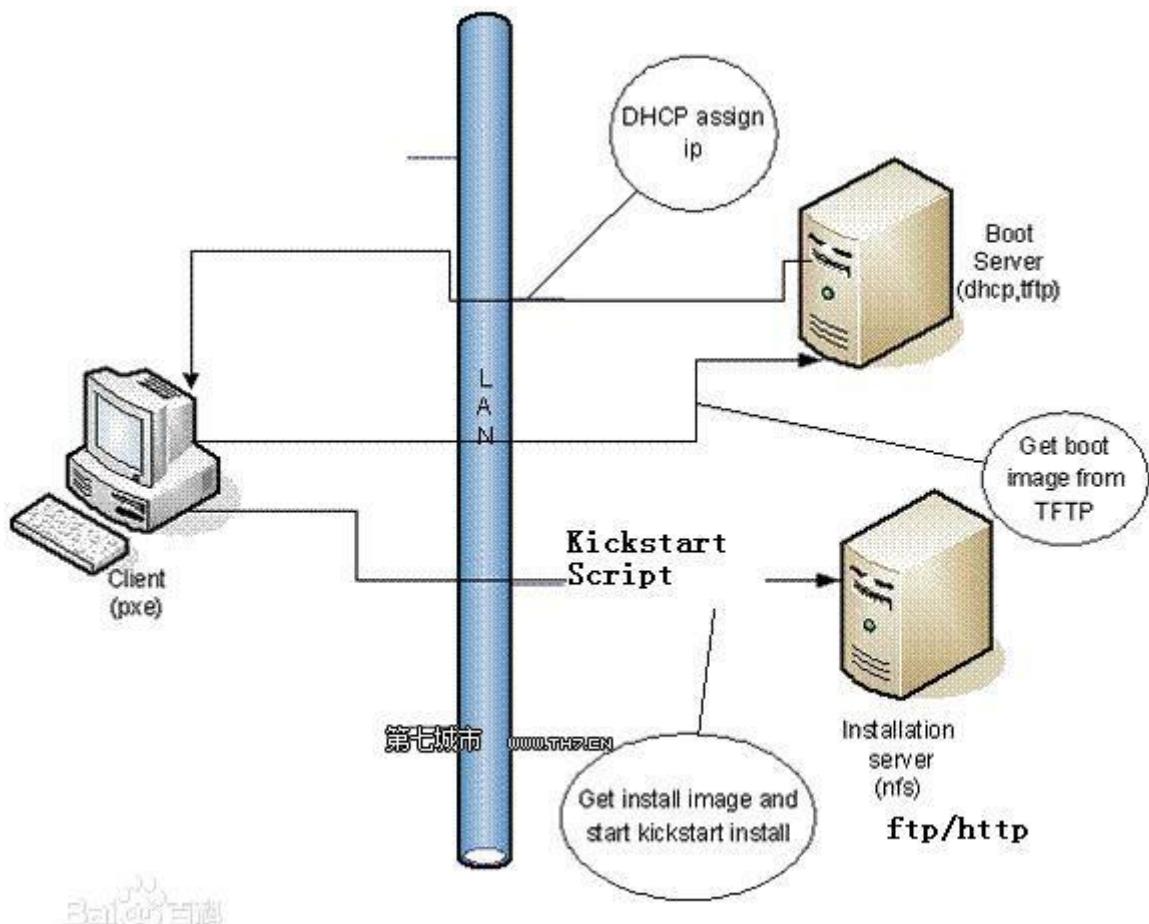


```
[root@huatech ~]# yum -y install kickstart
[root@huatech ~]# yum -y install system-config-kickstart
[root@huatech ~]# system-config-kickstart
```

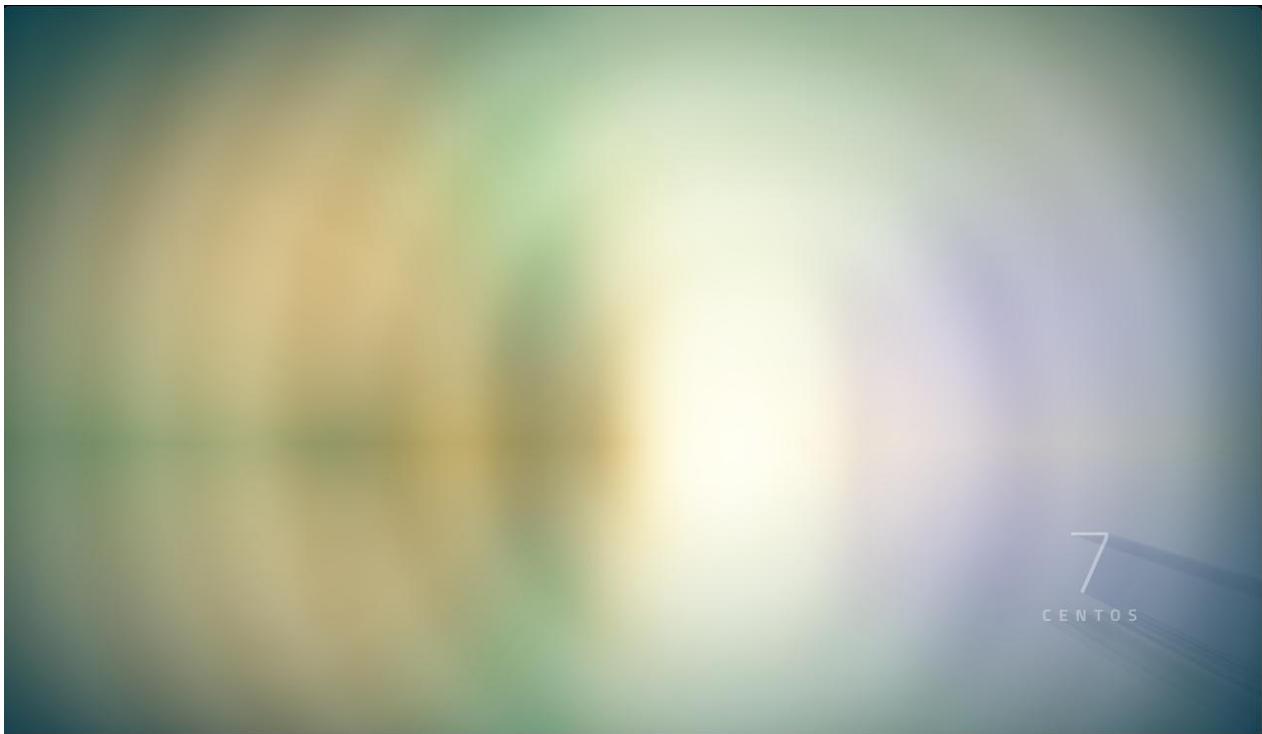




```
[root@huatech ~]# mkdir /var/www/html/ks
[root@huatech ~]# cp -p ks.cfg /var/www/html/ks/
[root@huatech ~]# vim /var/lib/tftpboot/pxelinux.cfg/default
append initrd=centos7/initrd.img method=http://192.168.88.247/ks.cfg
devfs=nomount
append initrd=centos7/initrd.img ks=http://192.168.88.247/ks.cfg devfs=nomount
append initrd=centos7/initrd.img ks=http://192.168.88.247/ks.cfg
```



```
[root@huatech ~]# yum -y install dracut-network nfs-utils
[root@huatech ~]# mkdir -p /var/lib/tftpboot/centos7/root
[root@huatech ~]# vim /var/lib/tftpboot/centos7/root/etc/shadow
[root@huatech ~]# vim /var/lib/tftpboot/centos7/root/etc/fstab
none      /tmp        tmpfs    defaults    0 0
tmpfs     /dev/shm    tmpfs    defaults    0 0
sysfs     /sys        sysfs    defaults    0 0
proc      /proc       proc    defaults    0 0
[root@huatech root]# wget -P /var/lib/tftpboot/centos7/
http://mirror.centos.org/centos/7/os/x86_64/images/pxeboot/vmlinuz
http://mirror.centos.org/centos/7/os/x86\_64/images/pxeboot/initrd.img
[root@huatech ~]# vim /var/lib/tftpboot/pxelinux.cfg/default
default centos7
label centos7
kernel centos7/vmlinuz
append initrd=centos7/initrd.img
root=nfs:192.168.88.234:/var/lib/tftpboot/centos7/root rw selinux=0
[root@huatech ~]# vim /etc/exports
/var/lib/tftpboot/centos7/root 192.168.88.0/24(rw, no_root_squash)
[root@huatech ~]# systemctl start rpcbind nfs-server
[root@huatech ~]# systemctl enable rpcbind nfs-server
```



```
[root@huatech ~]# python -c 'import crypt, getpass;print(crypt.crypt(getpass.getpass(), crypt.mksalt(crypt.METHOD_SHA512)))'
Password:
$6$g2Fxma3z5YwZPdsB$/jiorGm0vn0XUVFRJ.Fc0p1pQAE9TMt3rM15XoHs/AxdD2CKI/bZH2qP0l
Xiqy0ws1UECo5ZE1Jeuc0ID55RN1
[root@d1p ~]#vi /var/lib/tftpboot/centos7/root/etc/shadow
root:$6$EC1T.oKN5f3seb20$y1WIMQ7lh4240w0n.....:16372:0:99999:7:::
[root@huatech ~]# vim /var/lib/tftpboot/centos7/root/etc/shadow
```

二十二、ACL

```
[root@huatech ~]# setfacl -m u:root:0 test
[root@huatech ~]# getfacl test
```

二十三、RSYNC 同步

源主机：

```
[root@huatech ~]# yum -y install rsync
[root@huatech ~]# vim /etc/rsyncd_exclude.list
```

目标主机：

```
[root@server02 ~]# yum -y install rsync
[root@server02 ~]# vim /etc/rsyncd.conf
[backup]
path = /home/backup
hosts allow = 192.168.88.234
hosts deny = *
```

```
list = true
uid = root
gid = root
read only = false
[root@server02 ~]# systemctl start rsyncd
[root@server02 ~]# systemctl enable rsyncd
[root@huatech ~]# rsync -avz --delete --exclude-from=/etc/rsync_exclude.lst /192.168.88.26::backup
[root@huatech ~]# rsync -avz --delete --exclude-from=/etc/rsync_exclude.lst /192.168.88.26::backup
[root@huatech ~]# crontab -e
00 02 * * * rsync -avz --delete --exclude-from=/etc/rsync_exclude.lst /192.168.88.26::backup
```

二十四、RKHunter

```
[root@huatech ~]# yum --enablerepo=epel -y install rkhunter
[root@huatech ~]# vim /etc/sysconfig/rkhunter
[root@huatech ~]# rkhunter -update
[root@huatech ~]# rkhunter -propupd
[root@huatech ~]# rkhunter --check -sk
```

二十五、杀毒

```
[root@huatech ~]# yum --enablerepo=epel -y install clamav clamav-update
[root@huatech ~]# freshclam
[root@huatech ~]# sed -i -e "s/^Example/#Example/" /etc/freshclam.conf
[root@huatech ~]# freshclam
[root@huatech ~]# clamscan --infected --remove --recursive /home
[root@huatech ~]# curl -O http://www.eicar.org/download/eicar.com 【病毒库】
[root@huatech ~]# clamscan --infected --remove --recursive .
----- SCAN SUMMARY -----

```

```
Known viruses: 4285683
Engine version: 0.99
Scanned directories: 101
Scanned files: 118
Infected files: 0
Data scanned: 11.77 MB
Data read: 18.41 MB (ratio 0.64:1)
```

二十六、Linux 加入 windows AD 域

```
Domain Server: Windows Server 2012 R2
Domain Name: FD3S01
Realm: FD3S.SERVER.WORLD
Hostname: fd3s.server.world
```

```
[root@rdh ~]# yum -y install realmd sssd oddjob oddjob-mkhomedir adcli samba-common  
[root@rdh ~]# nmcli connection  
add delete down edit help load modify reload show up  
add: 增加网卡类型  
delete: 删除网卡  
down: 关闭网卡  
up: 启动网卡  
edit: 编辑网卡  
modify: 更改网卡参数。
```

注意：加入域的客户端的 DNS 得指向域控制器。

```
[root@rdh ~]# realm Discovery www.baidu.com 【发现域名】  
[root@rdh ~]# realm join www.baidu.com 【加入域】  
[root@rdh ~]# id wmm\server 【确定能从获取组的用户】  
[root@rdh ~]# su - wmm\server  
[root@rdh ~]# systemctl restart sssd
```

主机入侵防御系统 HIDS

AIDE: 记录主机的状态信息，也就是 INODE 里面的信息，MTIME 更改时间、ATIME 访问事件、CTIME 创建时间，一旦一个文件有所更改，都会记录。都记录在一个数据库当中，每次进行比对，检查更新。

```
[root@rdh ~]# yum -y install aide  
[root@rdh ~]# vim /etc/aide.conf  
/var/log p+u+g+i+n+acl+selinux+xattrs  
[root@rdh ~]# cp -p /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz  
CP: -p 保留权限、-r 复制目录  
[root@rdh ~]# chmod 644 anaconda-ks.cfg  
[root@rdh ~]# aide -check  
[root@rdh ~]# aide --update  
[root@rdh ~]# cp -p /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz  
[root@rdh ~]# vim /etc/cron.d/aide  
00 01 * * * /usr/sbin/aide --update | mail -s 'Daily Check by AIDE' root  
[root@rdh ~]# systemctl restart crond
```

二十七、Linux 配置 JAVA 环境

```
[root@rdh ~]# curl -L0 -H "Cookie: oraclelicense=accept-securebackup-cookie" \  
> http://download.oracle.com/otn-pub/java/jdk/8u71-b15/jdk-8u71-linux-x64.rpm
```

```
[root@rdh ~]# curl -L0 -H "Cookie: oraclelicense=accept-securebackup-cookie" \  
> "http://download.oracle.com/otn-pub/java/jdk/8u71-b15/jdk-8u71-linux-x64.rpm"  
% Total    % Received % Xferd  Average Speed   Time      Time     Current  
          Dload  Upload Total   Spent    Left Speed  
 0       0     0      0     0      0      0 --:--:--  0:00:21 --:--:--     0  
100  285  100  285     0      0     12      0  0:00:23  0:00:22  0:00:01  401  
 27 152M  27 42.3M     0      0    110k      0  0:23:35  0:06:31  0:17:04 43870
```

```
[root@rdh ~]# rpm -Uvh jdk-8u71-linux-x64.rpm
```

```
[root@rdh ~]# vim /etc/profile
export JAVA_HOME=/usr/java/default
export PATH=$PATH:$JAVA_HOME/bin
export CLASSPATH=.:$JAVA_HOME/jre/lib:$JAVA_HOME/lib:$JAVA_HOME/lib/tools.jar
[root@rdh ~]# source /etc/profile
[root@rdh ~]# alternatives --config java
[root@rdh ~]# vim day.java
import java.util.Calendar;
class day {
    public static void main(String[] args) {
        Calendar cal = Calendar.getInstance();
        int year = cal.get(Calendar.YEAR);
        int month = cal.get(Calendar.MONTH) + 1;
        int day = cal.get(Calendar.DATE);
        int hour = cal.get(Calendar.HOUR_OF_DAY);
        int minute = cal.get(Calendar.MINUTE);
        System.out.println(year + "/" + month + "/" + day + " " + hour + ":" + minute);
    }
}
[root@rdh ~]# javac day.java
[root@rdh ~]# java day
```

二十八、Linux JAVA 容器 Tomcat

```
[root@rdh ~]# curl -O
http://ftp.riken.jp/net/apache/tomcat/tomcat-8/v8.0.20/bin/apache-tomcat-8.0.20.tar.gz
[root@rdh ~]# tar -zxf apache-tomcat-8.0.20.tar.gz
[root@rdh ~]# yum -y install tomcat*
[root@rdh ~]# wget
http://ftp.riken.jp/net/apache/tomcat/tomcat-9/v9.0.0.M3/bin/apache-tomcat-9.0.0.M3-fulldocs.tar.gz
[root@rdh ~]# tar -zxf apache-tomcat-9.0.0.M3-fulldocs.tar.gz
[root@rdh ~]# mv tomcat-9.0-doc/ /usr/tomcat9
```

Useradd:

-b, --base-dir BASE_DIR	新账户的主目录的基目录
-c, --comment COMMENT	新账户的 GECOS 字段
-d, --home-dir HOME_DIR	新账户的主目录
-D, --defaults	显示或更改默认的 useradd 配置
-e, --expiredate EXPIRE_DATE	新账户的过期日期
-f, --inactive INACTIVE	新账户的密码不活动期
-g, --gid GROUP	新账户主组的名称或 ID
-G, --groups GROUPS	新账户的附加组列表
-h, --help	显示此帮助信息并推出

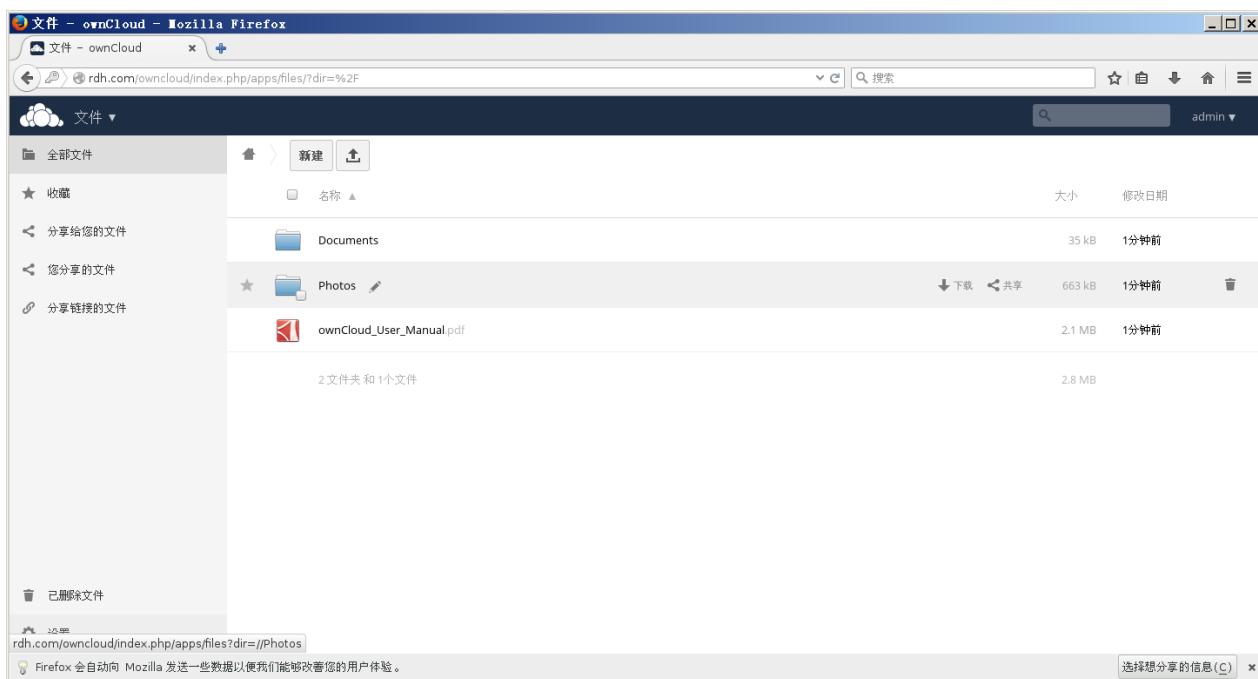
-k, --skel SKEL_DIR	使用此目录作为骨架目录
-K, --key KEY=VALUE	不使用 /etc/login.defs 中的默认值
-l, --no-log-init	不要将此用户添加到最近登录和登录失败数据库
-m, --create-home	创建用户的主目录
-M, --no-create-home	不创建用户的主目录
-N, --no-user-group	不创建同名的组
-o, --non-unique	允许使用重复的 UID 创建用户
-p, --password PASSWORD	加密后的新账户密码
-r, --system	创建一个系统账户
-R, --root CHROOT_DIR	chroot 到的目录
-s, --shell SHELL	新账户的登录 shell
-u, --uid UID	新账户的用户 ID
-U, --user-group	创建与用户同名的组
-Z, --selinux-user SEUSER	为 SELinux 用户映射使用指定 SEUSER

```
[root@rdh ~]# vim /usr/lib/systemd/system/tomcat8.service
[Service]
Type=oneshot
ExecStart=/usr/tomcat9/bin/startup.sh
ExecStop=/usr/tomcat9/bin/shutdown.sh
RemainAfterExit=yes
User=tomcat8
Group=tomcat8
[Install]
WantedBy=multi-user.target
[root@rdh ~]# systemctl restart tomcat8
[root@rdh ~]# systemctl enable tomcat8
```

二十九、Linux 建立私有云存储

```
[root@rdh ~]# yum -y install httpd
[root@rdh ~]# rm -f /etc/httpd/conf.d/welcome.conf
[root@rdh ~]# vim /etc/httpd/conf/httpd.conf
ServerName rdh.com:80
DirectoryIndex index.html index.cgi index.php
[root@rdh ~]# systemctl restart httpd
[root@rdh ~]# systemctl enable httpd
[root@rdh ~]# vim /var/www/html/index.html
[root@rdh ~]# yum -y install php php-mbstring php-pear
[root@rdh ~]# vim /etc/php.ini
date.timezone = "Asia/Tokyo"
[root@rdh ~]# systemctl restart httpd
[root@rdh ~]# vim /var/www/html/index.php
[root@rdh ~]# yum -y install mariadb-server
[root@rdh ~]# vim /etc/my.cnf
```

```
character-set-server=utf8
[root@rdh ~]# systemctl start mariadb
[root@rdh ~]# systemctl enable mariadb
[root@rdh ~]# mysql_secure_installation
[root@rdh ~]# mysql -u root -p
[root@rdh ~]# yum --enablerepo=epel -y install php-pear-MDB2-Driver-mysqli
php-pear-Net-Curl
[root@rdh ~]# wget
http://download.opensuse.org/repositories/isv:ownCloud:community/CentOS_CentOS
-7/isv:ownCloud:community.repo -P /etc/yum.repos.d
[root@rdh ~]# yum -y install owncloud
[root@rdh ~]# systemctl restart httpd
[root@rdh ~]# mysql -u root -p
MariaDB [(none)]> create database owncloud;
MariaDB [(none)]> grant all privileges on owncloud.* to owncloud@'localhost'
identified by 'password';
MariaDB [(none)]> flush privileges;
```



三十、CockPIT 系统管理面板

```
[root@rdh ~]# yum -y install cockpit
[root@rdh ~]# systemctl start cockpit
[root@rdh ~]# systemctl enable cockpit.socket
```

四十、Linux 防火墙 FirewallD 配置详解

FirewallD 提供了支持网络/防火墙区域(zone) 定义网络链接以及接口安全等级的动态防火墙管理工具。它支持 IPv4, IPv6 防火墙设置以及以太网桥接，并且拥有运行时配置和永久配置选项。它也支持允许服务或者应用程序直接添加防火墙规则的接口。以前的

system-config-firewall/lokkit 防火墙模型是静态的，每次修改都要求防火墙完全重启。这个过程包括内核 netfilter 防火墙模块的卸载和新配置所需模块的装载等。而模块的卸载将会破坏状态防火墙和确立的连接。

相反，firewall daemon 动态管理防火墙，不需要重启整个防火墙便可应用更改。因而也就没有必要重载所有内核防火墙模块了。不过，要使用 firewall daemon 就要求防火墙的所有变更都要通过该守护进程来实现，以确保守护进程中的状态和内核里的防火墙是一致的。另外，firewall daemon 无法解析由 ip*tables 和 ebttables 命令行工具添加的防火墙规则。

特性可以是预定义的防火墙功能，如：服务、端口和协议的组合、端口/数据报转发、伪装、ICMP 拦截或自定义规则等。该功能可以启用确定的一段时间也可以再次停用。通过所谓的直接接口，其他的服务(例如 libvirt)能够通过 iptables 变元(arguments)和参数(parameters)增加自己的规则。

使用 iptables 和 ip6tables 的静态防火墙规则。

```
[root@rdh ~]# yum install iptables-services  
[root@rdh ~]# systemctl mask firewalld.service  
[root@rdh ~]# systemctl enable iptables.service  
[root@rdh ~]# systemctl enable ip6tables.service
```

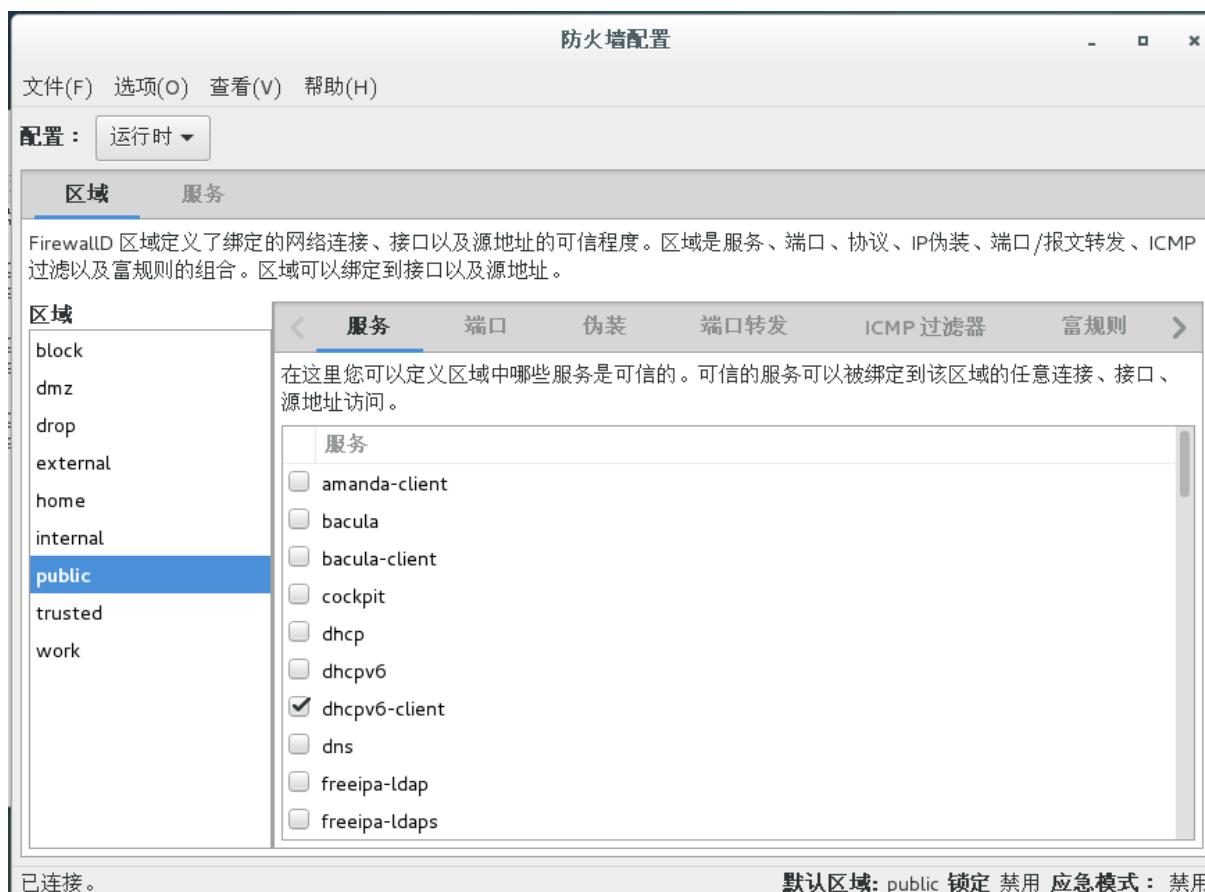
基本定义阻塞

预定义的服务：就是各种服务

端口协议：就是 TCP/IP

伪装：就是 NAT 协议

端口转发：从一个端口转移到另一个端口



丢弃：任何流入网络的包都被丢弃，不作出任何响应。只允许流出的网络连接。

阻塞：任何进入的网络连接都被拒绝，并返回 IPv4 的 icmp-host-prohibited 报文或者 IPv6 的 icmp6-adm-prohibited 报文。只允许由该系统初始化的网络连接。

公开：用以可以公开的部分。你认为网络中其他的计算机不可信并且可能伤害你的计算机。只允许选中的连接接入。

外部：用在路由器等启用伪装的外部网络。你认为网络中其他的计算机不可信并且可能伤害你的计算机。只允许选中的连接接入。

隔离区（dmz）：用以允许隔离区（dmz）中的电脑有限地被外界网络访问。只接受被选中的连接。

工作：用在工作网络。你信任网络中的大多数计算机不会影响你的计算机。只接受被选中的连接。

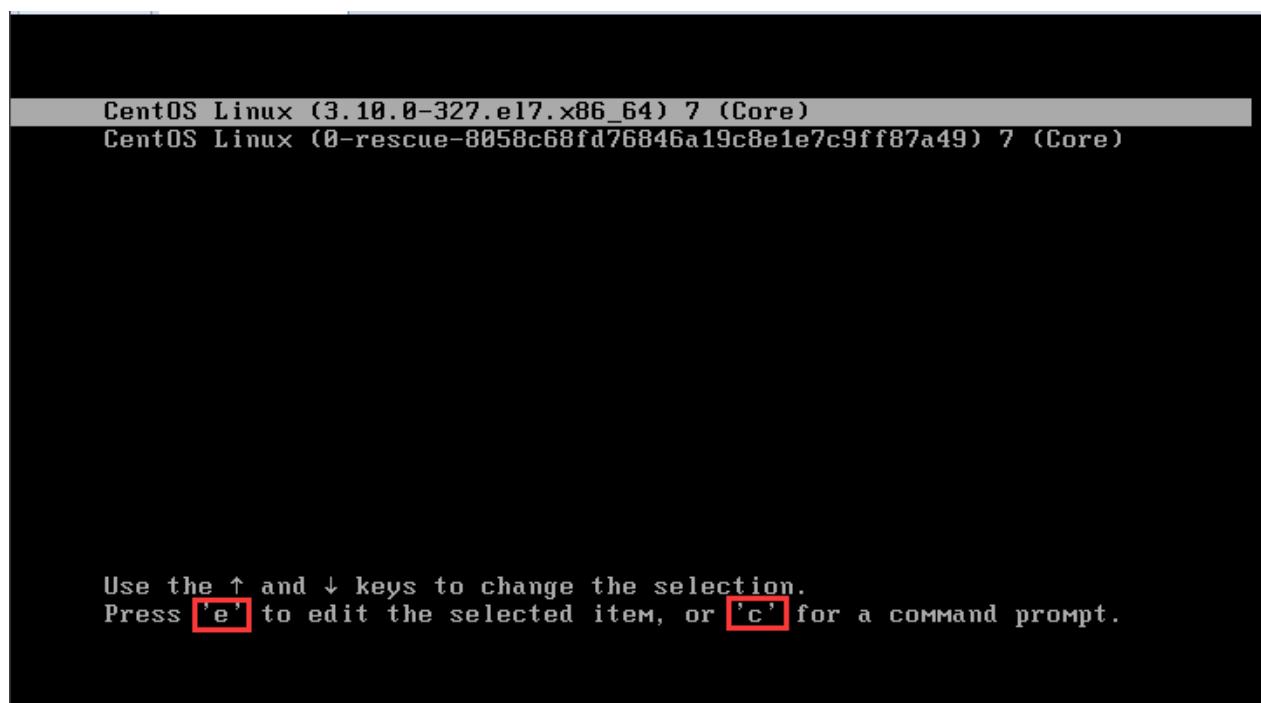
家庭：用在家庭网络。你信任网络中的大多数计算机不会影响你的计算机。只接受被选中的连接。

内部：用在内部网络。你信任网络中的大多数计算机不会影响你的计算机。只接受被选中的连接。

```
[root@huatech ~]# firewall-config
```

防火墙所有的服务器、端口、封装、转发都是根据端口设定的。

四十一、Linux 破解密码



选择内核模块，然后点击'e'编辑网卡，开始编辑。

```

insmod part_msdos
insmod xfs
set root='hd0,msdos1'
if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint\x
t-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1' ff66325f-f\x
53a-43b1-a139-fd9116d0c9b6
else
    search --no-floppy --fs-uuid --set=root ff66325f-f53a-43b1-a139-fd91\x
16d0c9b6
fi
linux16 /vmlinuz-3.10.0-327.el7.x86_64 root=/dev/mapper/centos-root ro\x
rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet LANG=zh_CN.UTF-8 rw in\x
it=/bin/bash
initrd16 /initramfs-3.10.0-327.el7.x86_64.img

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
discard edits and return to the menu. Pressing Tab lists
possible completions.

```

在后面添加 `rw init=/bin/bash` 进入可以读写的 shell, `/bin/bash`.

Press `Ctrl-x` to start, `Ctrl-c` for a command prompt or Escape to
discard edits and return to the menu. Pressing Tab lists
possible completions.

```

[ 2.055570] su 0.0.0.0: lsuid
bash-4.2# touch /.autorelabel_

```

```

bash-4.2# touch /.autorelabel
bash-4.2# passwd
[*****] root [*****]
[*****] 8 [*****]
passwd[*****]
bash-4.2# exec /sbin/init_

```

四十二、登录次数限制、限制登录

```

[root@huatech ~]# vim /etc/pam.d/system-auth
auth      required      pam_tally2.so deny=5 unlock_time=60
account   required      pam_tally2.so
[root@huatech ~]# vim /etc/pam.d/password-auth
auth      required      pam_tally2.so deny=5 unlock_time=60
account   required      pam_tally2.so
[root@huatech ~]# pam_tally2 -u root

```

四十三、磁盘配额 quota

```

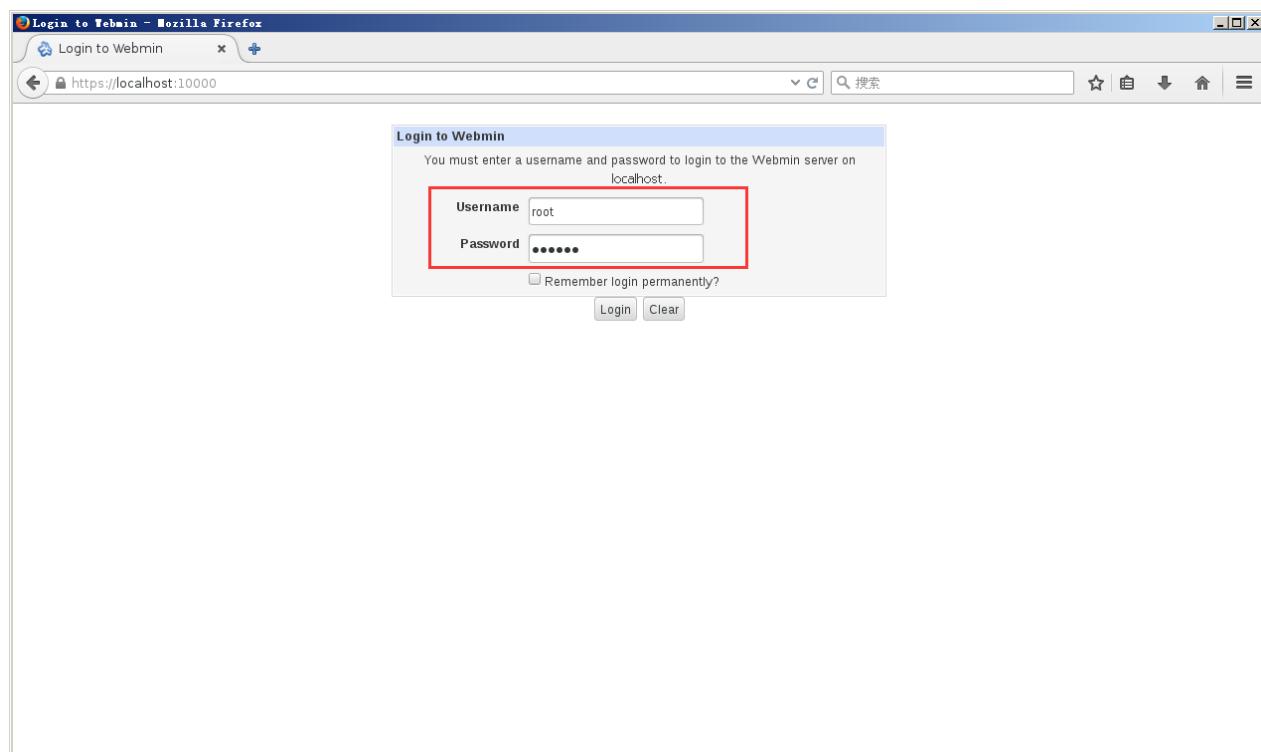
[root@huatech ~]# mount -o remount,usrquota,grpquota /home
[root@huatech ~]# vim /etc/fstab

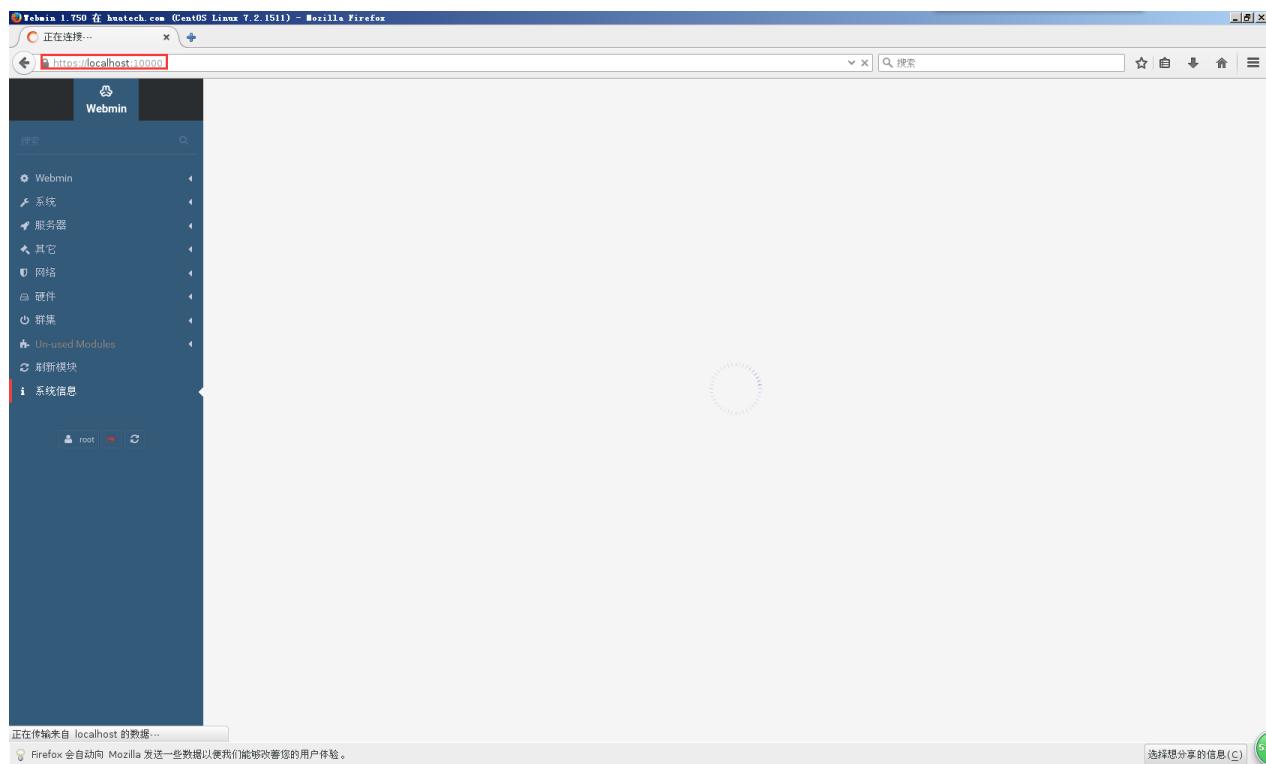
```

```
/dev/mapper/centos-home /home xfs
defaults,usrquota,grpquota 0 0
[root@huatech ~]# edquota -t 【磁盘配额时间】
[root@huatech ~]# edquota -u wmm
Filesystem          blocks    soft    hard   inodes   soft
hard
/dev/mapper/centos-home      5272     10000   20000    135     20
50
[root@huatech ~]# edquota -p wmm test 【把这个值复制给其它人】
[root@huatech ~]# edquota -t 【宽限时间】
[root@huatech ~]# quota -uvgs
[root@huatech ~]# repquota -avug
```

四十四、Webmin 管理面板

```
[root@huatech ~]# yum -y install perl-Net-SSLeay
[root@huatech ~]# yum -y install
http://download.webmin.com/download/yum/webmin-1.750-1.noarch.rpm
[root@huatech ~]# vim /etc/webmin/miniserv.conf
allow=127.0.0.1 192.168.88.0/24
[root@huatech ~]# /etc/init.d/webmin start
```





=====普通用户权限=====

```
[root@huatech ~]# yum --enablerepo=epel -y install perl-Net-SSLeay perl-Authen-PAM
```

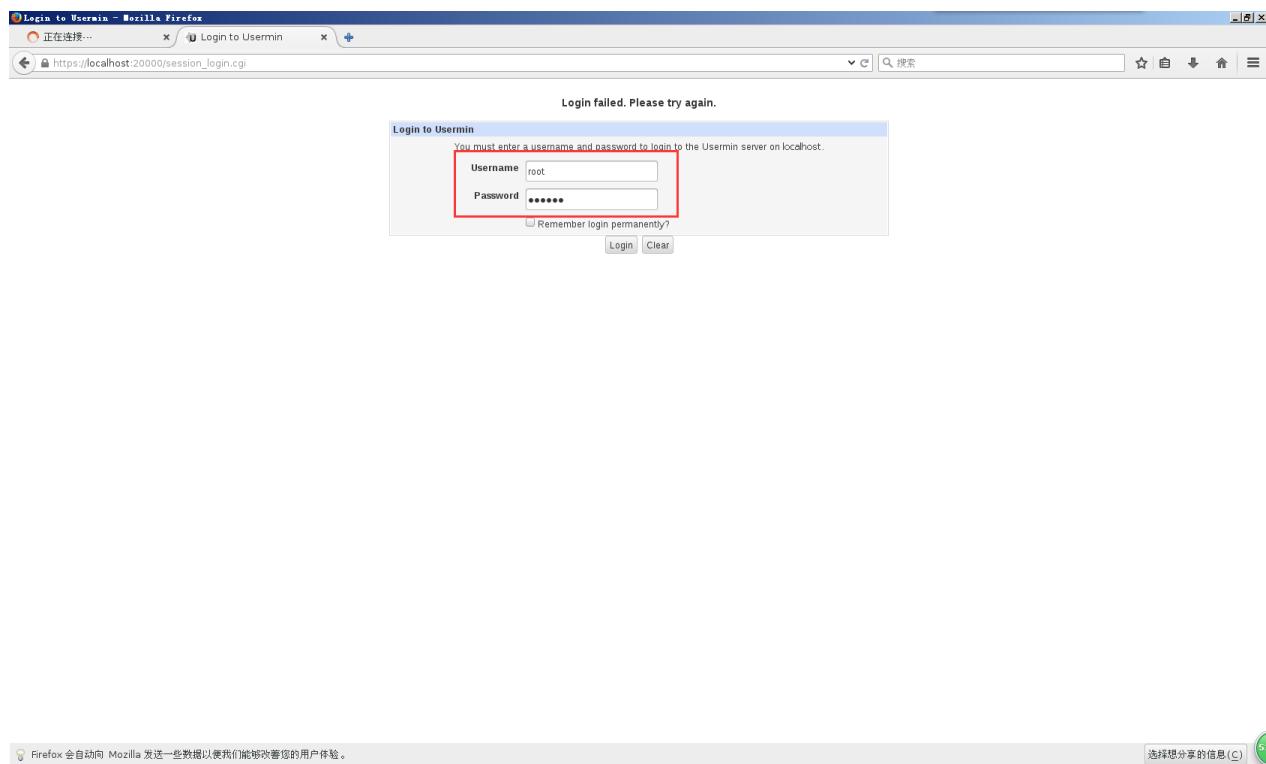
```
[root@huatech ~]# yum -y install
```

<http://nchc.dl.sourceforge.net/project/webadmin/usermin/1.700/usermin-1.700-1.noarch.rpm>

```
[root@huatech ~]# vim /etc/usermin/miniserv.conf
```

```
allow=127.0.0.1 192.168.88.0/24
```

```
[root@huatech ~]# /etc/init.d/usermin start
```



Firefox 会自动向 Mozilla 发送一些数据以便我们能够改善您的用户体验。 选择想分享的信息(C) 51

```
[root@huatech ~]# yum -y install
http://download.webmin.com/download/yum/usermin-webmail-1.700-1.noarch.rpm
=====
[root@huatech ~]# rpm -ivh usermin-webmail-1.700-1.noarch.rpm
下载地址: http://download.webmin.com/download/yum/
```

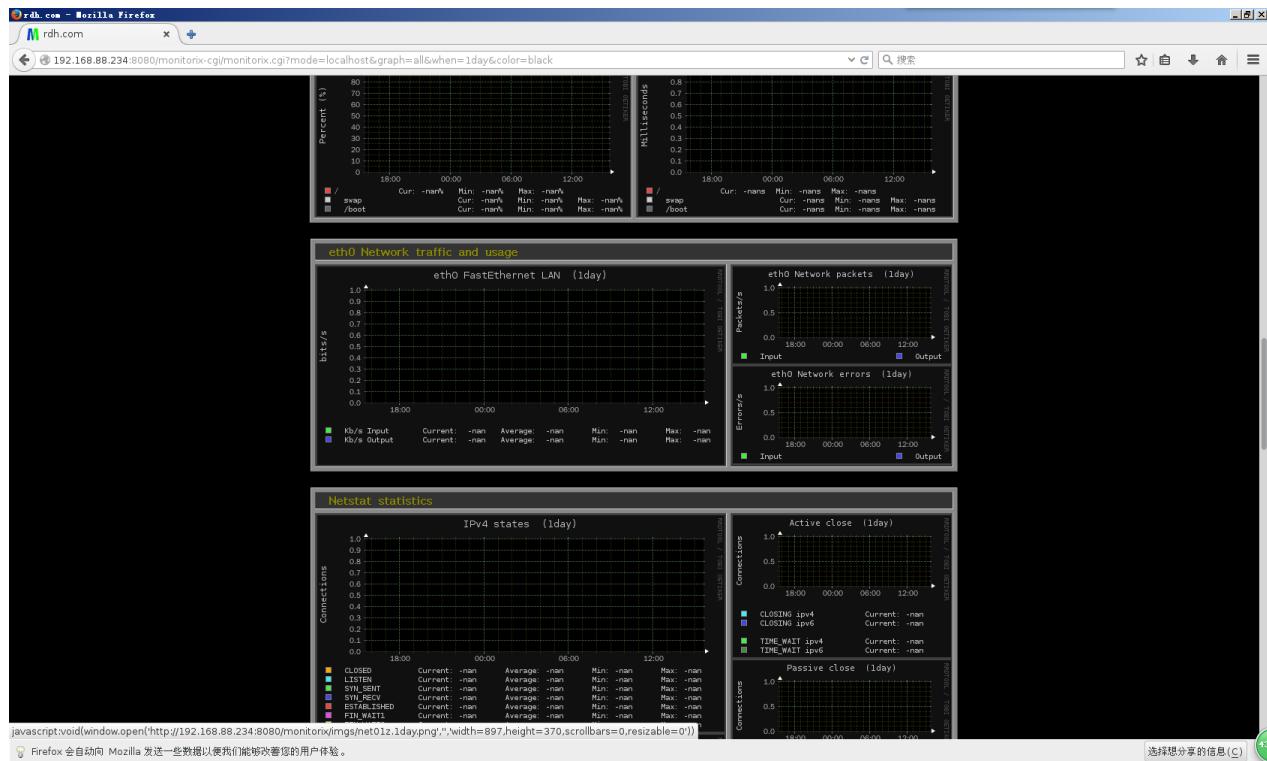
=====
[root@huatech ~]# curl -O <http://software.virtualmin.com/gpl/scripts/install.sh>
[root@huatech ~]# chmod +x install.sh

四十五、监控系列

monitorix

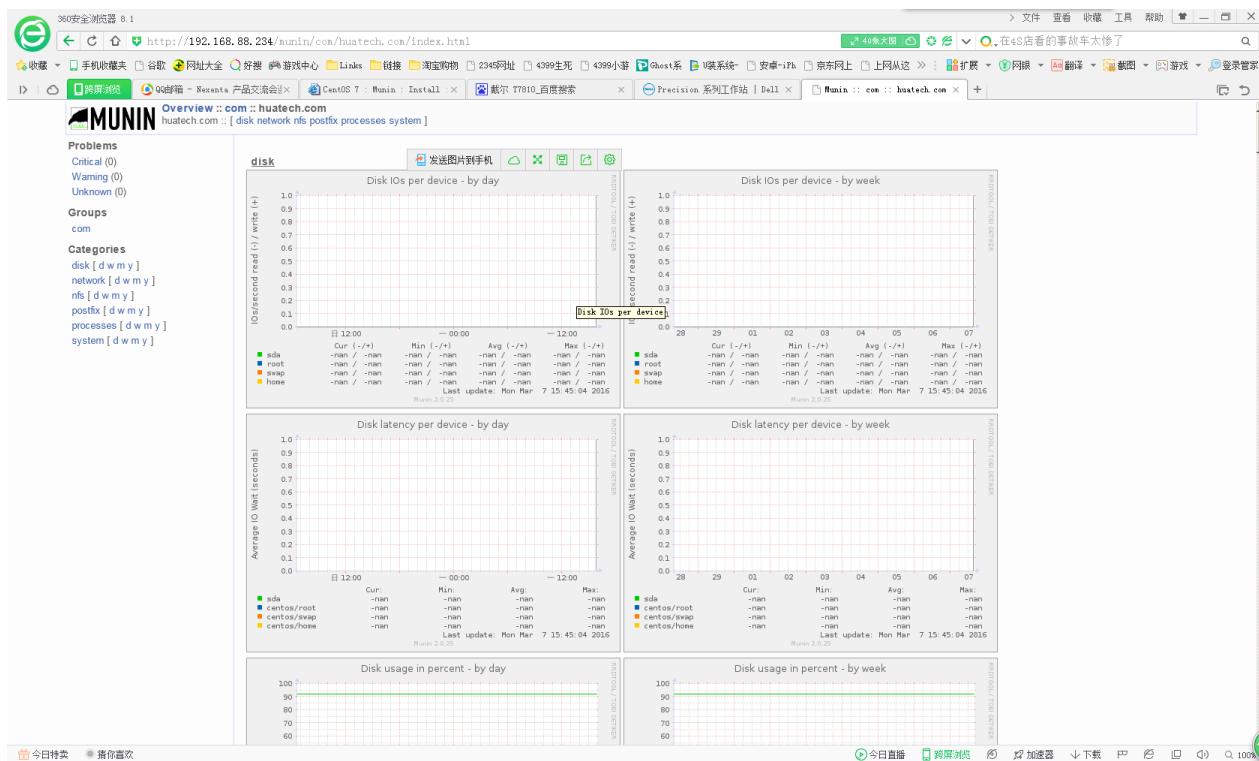
```
[root@huatech ~]# yum --enablerepo=epel -y install monitorix
[root@huatech ~]# vim /etc/monitorix/monitorix.conf
title = Monitorix
hostname = dlp.server.world
netstats_in_bps = y
hosts_deny = all
hosts_allow = 192.168.88.0/24
[root@huatech ~]# systemctl start monitorix
[root@huatech ~]# systemctl enable monitorix
```





munin

```
[root@huatech ~]# yum -y install httpd
[root@huatech ~]# rm -rf /etc/httpd/conf.d/welcome.conf
[root@huatech ~]# systemctl start httpd
[root@huatech ~]# systemctl enable httpd
[root@huatech ~]# yum --enablerepo=epel -y install munin munin-node
[root@huatech ~]# vim /etc/munin/munin.conf
[huatech.com]
    address 127.0.0.1
    use_node_name yes
[root@huatech ~]# vim /etc/httpd/conf.d/munin.conf
Order deny,allow
deny from all
allow from 192.168.88.0/24
[root@huatech ~]# systemctl restart httpd
[root@huatech ~]# htpasswd -c /etc/munin/munin-htpasswd test
New password:
Re-type new password:
Adding password for user test
[root@huatech ~]# vim /etc/munin/munin-node.conf
host_name huatech.com
```



分布式监控 Zabbix

安装 httpd 服务器

```
[root@huatech ~]# rm -rf /etc/httpd/conf.d/welcome.conf
[root@huatech ~]# yum -y install httpd
[root@huatech ~]# vim /etc/httpd/conf/httpd.conf
```

ServerAdmin root@huatech.com

ServerName huatech.com:80

AllowOverride All

DirectoryIndex index.html index.cgi index.php

PHP 环境配置

```
[root@huatech ~]# yum -y install php php-mbstring php-pear
[root@huatech ~]# vim /etc/php.ini
date.timezone = "Asia/Tokyo"
[root@huatech ~]# systemctl restart httpd
```

安装 MariaDB 数据库

```
[root@huatech ~]# yum -y install mariadb-server
[root@huatech ~]# vim /etc/my.cnf
character-set-server=utf8
[root@huatech ~]# systemctl start mariadb
[root@huatech ~]# systemctl enable mariadb
[root@huatech ~]# mysql_secure_installation
```

安装 Zabbix

```
[root@huatech ~]# yum -y install php-mysql php-gd php-xml php-bcmath
[root@huatech ~]# yum -y install
http://repo.zabbix.com/zabbix/2.4/rhel/7/x86\_64/zabbix-release-2.4-1.el7.noarch
```

h. rpm

```
[root@huatech ~]# yum -y install zabbix-get zabbix-server-mysql zabbix-web-mysql  
zabbix-agent  
[root@huatech ~]# yum -y update  
http://ftp.jaist.ac.jp/pub/Linux/CentOS/6/os/x86\_64/Packages/trousers-0.3.13-2.el6.x86\_64.rpm  
[root@huatech ~]# mysql -u root -p  
MariaDB [(none)]> grant all privileges on zabbix.* to zabbix@'localhost' identified  
by 'password';  
MariaDB [(none)]> create database zabbix;  
MariaDB [(none)]> grant all privileges on zabbix.* to zabbix@'%' identified by  
'password';  
MariaDB [(none)]> flush privileges;  
[root@huatech ~]# cd /usr/share/doc/zabbix-server-mysql-*/*/create  
[root@huatech create]# mysql -u root -p zabbix < schema.sql 【导入数据库】  
Enter password:  
[root@huatech create]# mysql -u root -p zabbix < images.sql 【导入数据库】  
Enter password:  
[root@huatech create]# mysql -u root -p zabbix < data.sql 【导入数据库】  
Enter password:  
[root@huatech create]# vim /etc/zabbix/zabbix_server.conf  
DBHost=localhost  
DBPassword=password  
[root@huatech create]# systemctl start zabbix-server  
[root@huatech create]# systemctl enable zabbix-server  
[root@huatech ~]# vim /etc/zabbix/zabbix_agentd.conf  
[root@huatech ~]# systemctl start zabbix-agent  
[root@huatech ~]# systemctl enable zabbix-agent  
[root@huatech ~]# vim /etc/httpd/conf.d/zabbix.conf  
php_value date.timezone Asia/Tokyo
```

This image shows the "Check of pre-requisites" step from the Zabbix 2.4 setup. The left sidebar has the same navigation as the previous screen. The main area is titled "2. Check of pre-requisites". It contains a table comparing current PHP settings against required values. Most entries show "OK" status. The table includes columns for "Current value", "Required", and "Status".

	Current value	Required	Status
PHP version	5.4.16	5.3.0	OK
PHP option memory_limit	128M	128M	OK
PHP option post_max_size	16M	16M	OK
PHP option upload_max_filesize	2M	2M	OK
PHP option max_execution_time	300	300	OK
PHP option max_input_time	300	300	OK
PHP time zone	Asia/Tokyo		OK
PHP databases support	MySQL SQLite3		OK
PHP bcmath	on		OK
PHP mbstring	on		OK
PHP mbstring.func_overload	off	off	OK
PHP sockets	on		OK
PHP gd	2.1.0	2.0	OK
PHP gd PNG support	on		OK

At the bottom right of the table area, there is an "OK" button. Navigation buttons "Cancel", "<< Previous", and "Next >" are located at the very bottom.

ZABBIX

3. Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database.

Press "Test connection" button when done.

Database type	MySQL
Database host	localhost
Database port	0 0 - use default port
Database name	zabbix
User	zabbix
Password	*****

Test connection

[Cancel](#) [« Previous](#) [Next »](#)

www.zabbix.com
Licensed under [GPL v2](#)

ZABBIX

4. Zabbix server details

Please enter host name or host IP address
and port number of Zabbix server,
as well as the name of the installation (optional).

Host	localhost
Port	10051
Name	huatech.com

[Cancel](#) [« Previous](#) [Next »](#)

www.zabbix.com
Licensed under [GPL v2](#)

The screenshot shows the Zabbix 2.4.7 login screen at the top, followed by the personal dashboard.

Login Screen:

- Username:
- Password:
- Remember me for 30 days
- Sign in** | **Login as Guest**

Zabbix 2.4.7 Copyright 2001-2015 by Zabbix SIA

Personal Dashboard:

- Favourite graphs**: No graphs added. [Graphs »](#)
- Favourite screens**: No screens added. [Screens »](#) | [Slide shows »](#)
- Favourite maps**: No maps added. [Maps »](#)
- Status of Zabbix**

Parameter	Value	Details
Zabbix server is running	Yes	localhost:10051
Number of hosts (enabled/disabled/templates)	39	0 / 1 / 38
Number of items (enabled/disabled/not supported)	0	0 / 0 / 0
Number of triggers (enabled/disabled [problem/ok])	0	0 / 0 [0 / 0]
Number of users (online)	2	1
Required server performance, new values per second	0	-

Updated: 17:33:03
- System status**

Host group	Disaster	High	Average	Warning	Information	Not classified
No host groups found.						

Updated: 17:33:03
- Host status**

Host group	Without problems	With problems	Total
No host groups found.			

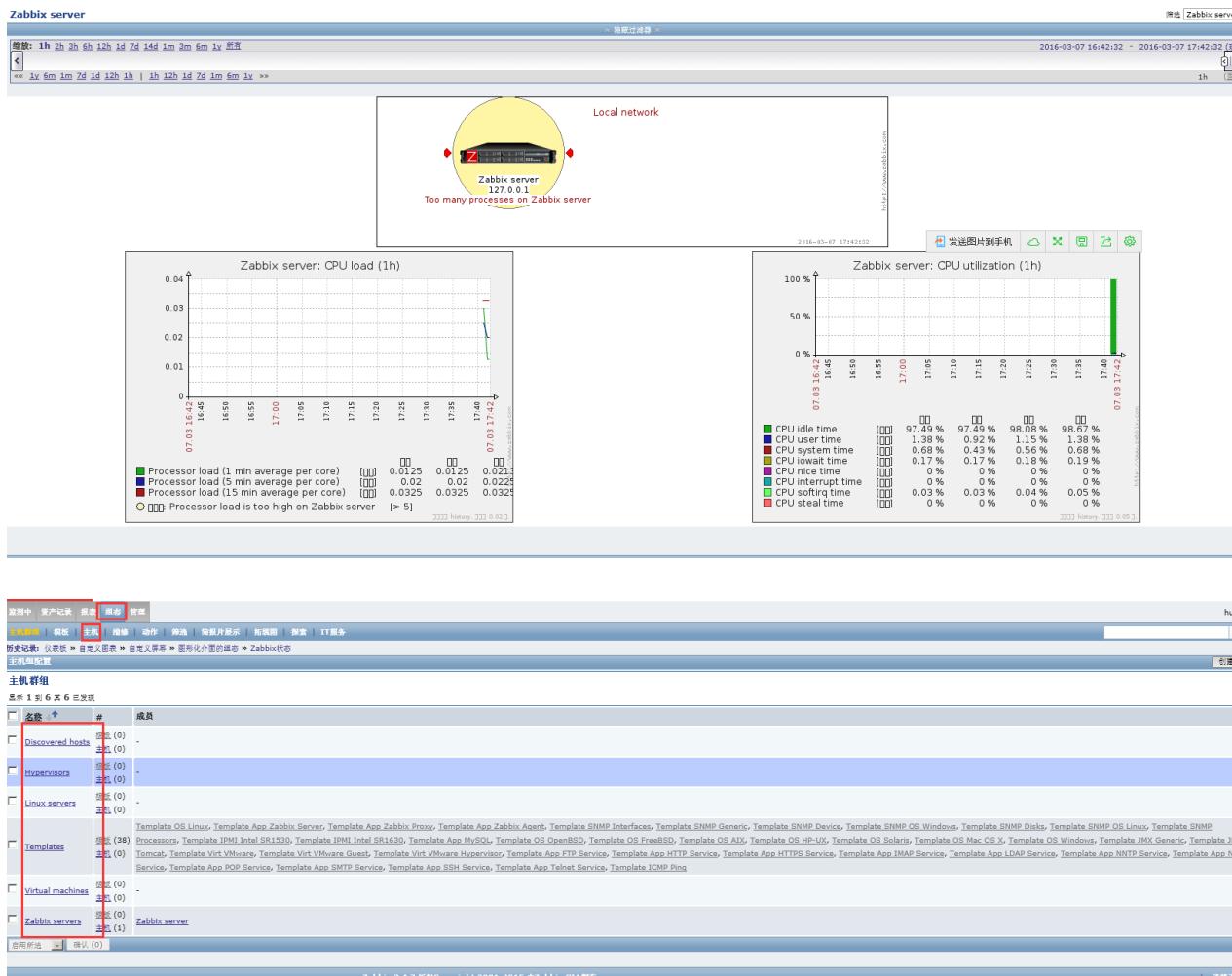
Updated: 17:33:03
- Last 20 issues**

Host	Issue	Last change	Age	Info	Ack	Actions
No events found.						

0 of 0 issues are shown
Updated: 17:33:03
- Web monitoring**: (empty table)

账户: admin

密码: Zabbix



添加目标主机: linux

```
[root@server02 ~]# yum -y install
http://repo.zabbix.com/zabbix/2.4/rhel/7/x86\_64/zabbix-release-2.4-1.el7.noarch.rpm
```

Cacti+Nagios+Mrtg

下 载 地 址 :

http://nchc.dl.sourceforge.net/project/cnyunwei/V11/Cnyunwei-X64-V11_base.iso

操作手册: <http://www.cnyunwei.com/forum.php?mod=viewthread&tid=5710>

```
[root@rdh ~]# yum -y install httpd
[root@rdh ~]# rm -rf /etc/httpd/conf.d/welcome.conf
[root@rdh ~]# vim /etc/httpd/conf/httpd.conf
```

ServerAdmin root@rdh.com

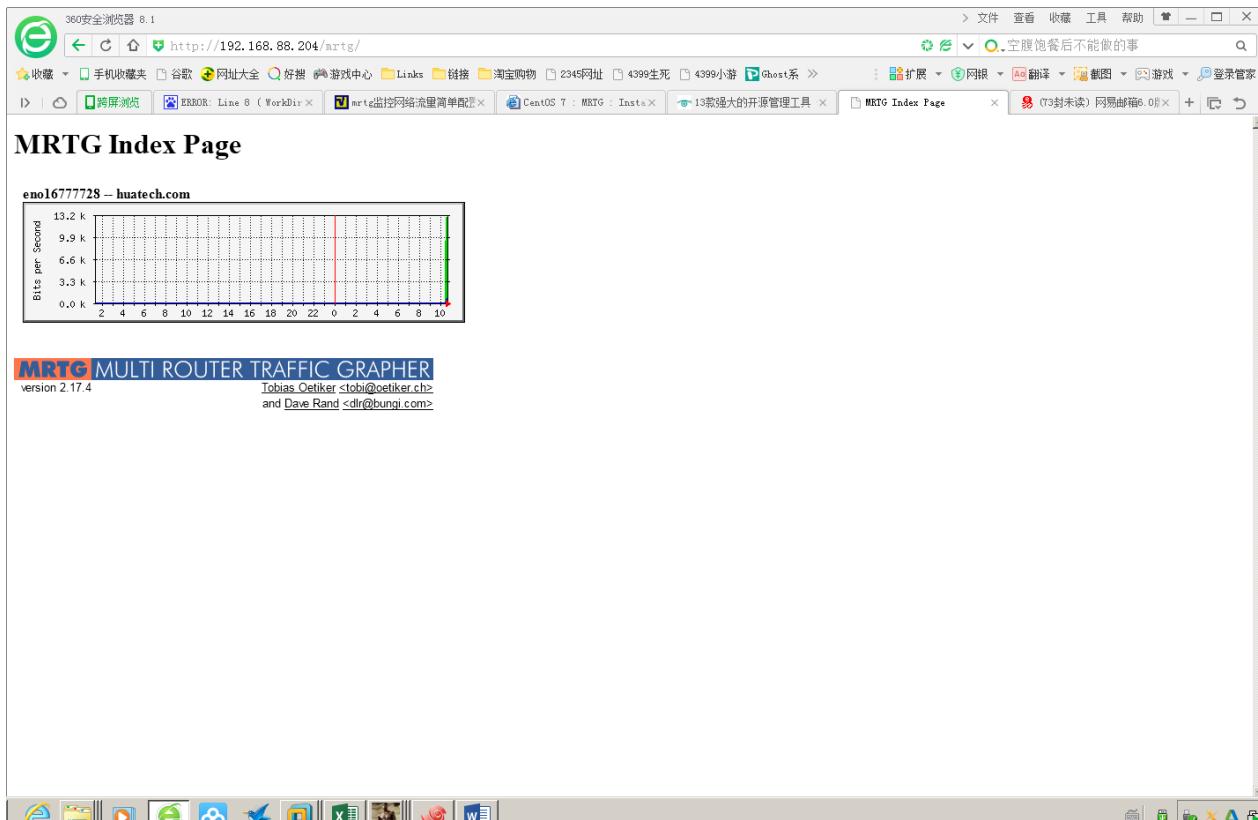
ServerName rdh.com:80

DirectoryIndex index.html index.cgi index.php

```
[root@rdh ~]# systemctl start httpd
[root@rdh ~]# systemctl enable httpd
[root@rdh ~]# yum -y install net-snmp net-snmp-utils mrtg
[root@rdh ~]# vim /etc/snmp/snmpd.conf
```

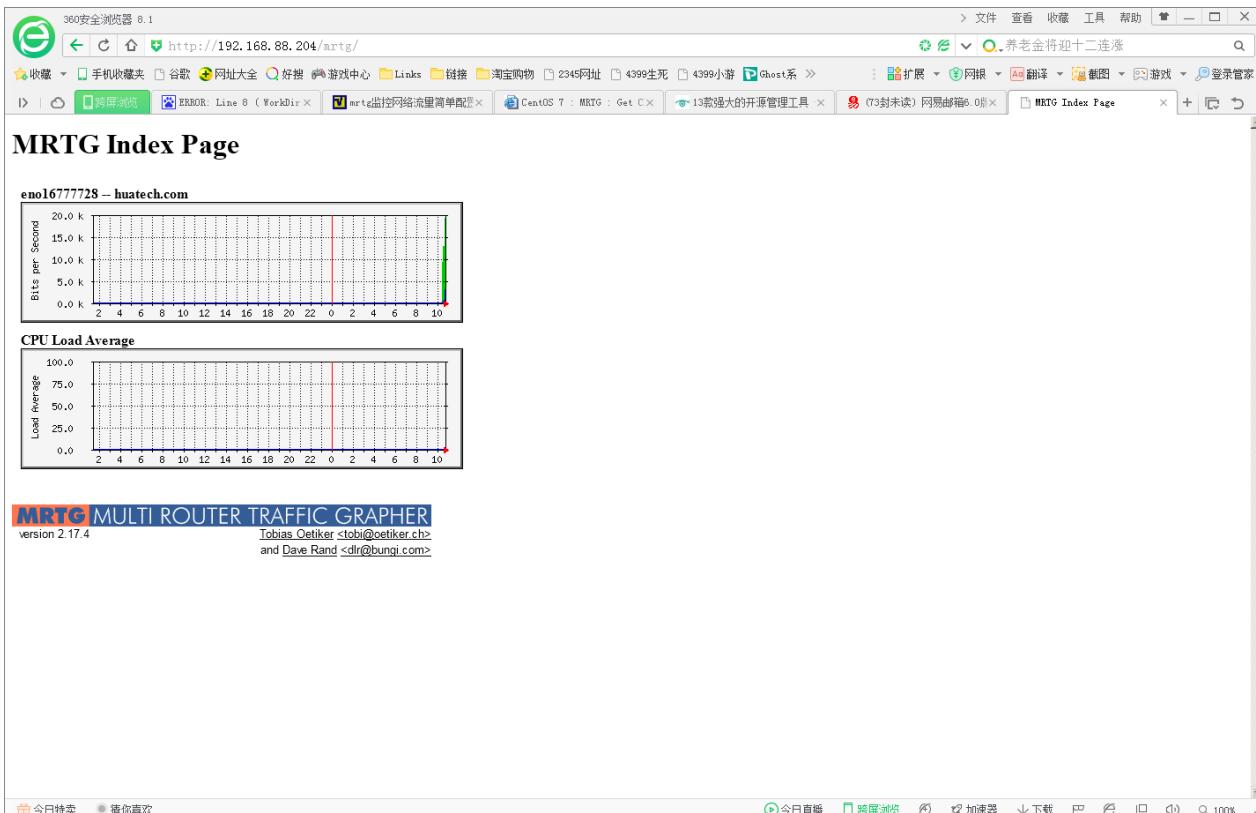
access MyROGroup "" v2c noauth exact all none none

```
access MyRWGroup "" v2c noauth exact all all all
view all included .1
group MyRWGroup v2c local
group MyR0Group v2c mynetwork
com2sec local localhost rdh
com2sec mynetwork 192.168.88.0/24 rdh
[root@huatech ~]# snmpwalk -v2c -c rdh localhost system
[root@huatech ~]# cfgmaker --snmp-options=:::::2 --ifref=descr --ifdesc=descr
rdh@192.168.88.204 > /etc/mrtg/mrtg.cfg
[root@huatech ~]# vim /etc/mrtg/mrtg.cfg
WorkDir: /var/www/mrtg
Options[_]: growright, bits
【顶格写，不然会报错】
[root@huatech www]# for (( i=1 ; i <= 3 ; i++ )); do env LANG=C mrtg
/etc/mrtg/mrtg.cfg; done
[root@huatech www]# indexmaker --columns=1 /etc/mrtg/mrtg.cfg >
/var/www/mrtg/index.html
[root@huatech ~]# vim /etc/cron.d/mrtg
*/5 * * * * root LANG=C LC_ALL=C /usr/bin/mrtg /etc/mrtg/mrtg.cfg --lock-file
/var/lock/mrtg/mrtg_l --confcache-file /var/lib/mrtg/mrtg.ok
[root@huatech ~]# vim /etc/httpd/conf.d/mrtg.conf
<Location /mrtg>
    # Require local
    Require ip 192.168.88.0/24
    DirectoryIndex index.html
</Location>
[root@huatech ~]# systemctl restart httpd
http://192.168.88.204/mrtg/
```



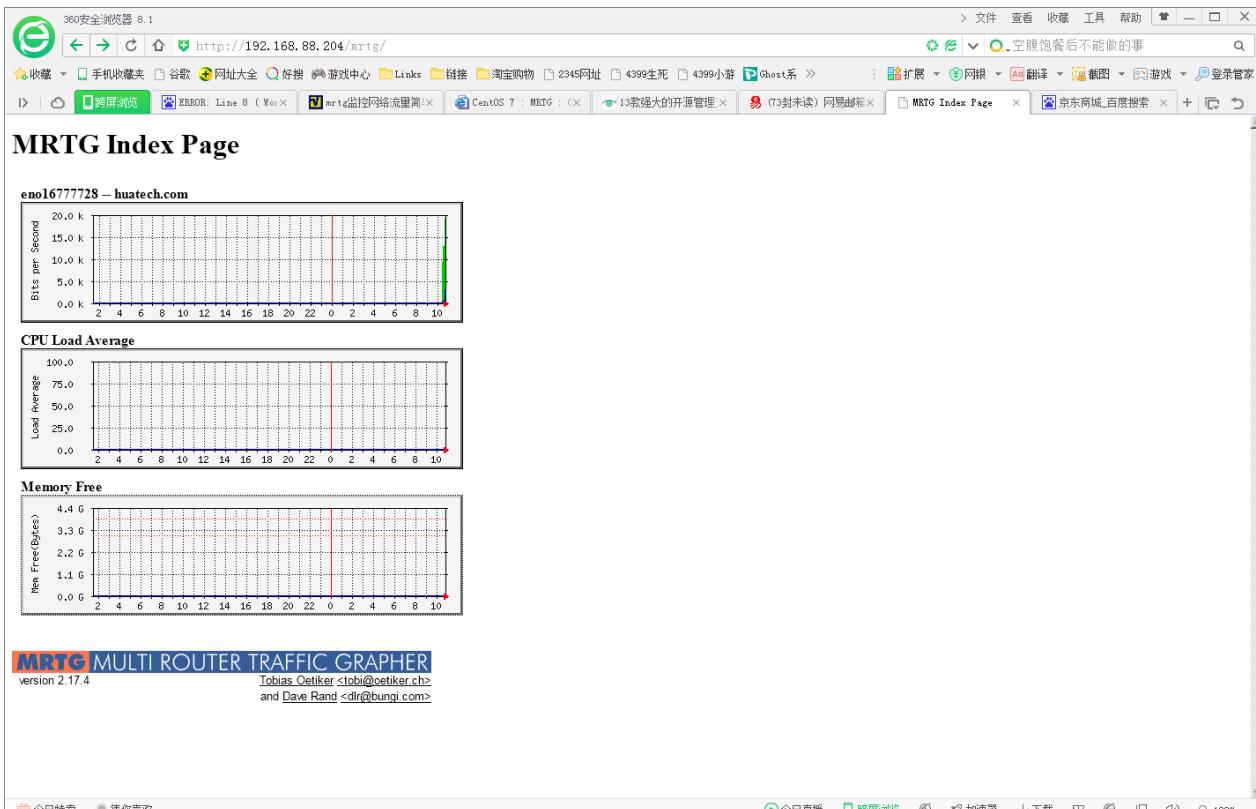
===== 获取 CPU 负载 =====

```
[root@huatech ~]# vim /etc/mrtg/mrtg.cfg
Target[CPU]: .1.3.6.1.4.1.2021.10.1.5.1&.1.3.6.1.4.1.2021.10.1.5.2:rdh@127.0.0
.1:::::2
MaxBytes[CPU]: 100
Unscaled[CPU]: dwmy
Options[CPU]: gauge, growright, nopercent
YLegend[CPU]: Load Average
ShortLegend[CPU]: (%)
Legend1[CPU]: Load Average 1 min
Legend0[CPU]: Load Average 5 min
Legend1[CPU]: Load Average 1 min
Legend2[CPU]: Load Average 5 min
Title[CPU]: CPU Load Average
PageTop[CPU]: <h1>CPU Load Average</h1>
[root@huatech ~]# for (( i=1 ; i <= 3 ; i++ )); do env LANG=C mrtg /etc/mrtg/mrtg.cfg;
done
[root@huatech ~]# indexmaker --columns=1 /etc/mrtg/mrtg.cfg >
/var/www/mrtg/index.html
```



=====获取内存负载=====

```
[root@huatech ~]# vim /etc/mrtg/mrtg.cfg
Target[mem]: . 1. 3. 6. 1. 4. 1. 2021. 4. 6. 0&. 1. 3. 6. 1. 4. 1. 2021. 4. 4. 0:rdh@127. 0. 0. 1::::
:2
# total memory
MaxBytes1[Mem]: 4047620
# total swap
MaxBytes2[Mem]: 3145724
Unscaled[Mem]: dwmly
Options[Mem]: gauge, growright
YLegend[Mem]: Mem Free (Bytes)
ShortLegend[Mem]: Bytes
kilo[Mem]: 1024
kMG[Mem]: k, M, G, T, P
Legend1[Mem]: Real
Legend0[Mem]: Swap
Legend1[Mem]: Memory Free [MBytes]
Legend2[Mem]: Swap Free [MBytes]
Title[Mem]: Memory Free
PageTop[Mem]: <H1>Memory Free</H1>
[root@huatech ~]# for (( i=1 ; i <= 3 ; i++ )) ; do env LANG=C mrtg /etc/mrtg/mrtg.cfg;
done
[root@huatech ~]# indexmaker --columns=1 /etc/mrtg/mrtg.cfg >
/var/www/mrtg/index.html
```



=====磁盘使用空间=====

```
[root@huatech ~]# vim /etc/snmp/snmpd.conf
disk / 10000
[root@huatech ~]# systemctl restart snmpd
[root@huatech ~]# snmpwalk -v2c -c rdh localhost .1.3.6.1.4.1.2021.9.1.6
[root@huatech ~]# vim /etc/mrtg/mrtg.cfg
Target[Disk]: .1.3.6.1.4.1.2021.9.1.7.1&.1.3.6.1.4.1.2021.9.1.7.1:rdh@127.0.0.1::::2
MaxBytes[Disk]: 27740944
kMG[Disk]: k, M, G, T, P
Unscaled[Disk]: dwmy
Options[Disk]: gauge, absolute, growright, nopercent
YLegend[Disk]: Disk Free (Bytes)
ShortLegend[Disk]: Bytes
Legend1[Disk]: / Disk Free [Bytes]
Legend0[Disk]:
Legend1[Disk]: / Disk Free [Bytes]
Legend2[Disk]:
Title[Disk]: Disk Free
PageTop[Disk]: <H1>Disk Free</H1>
[root@huatech ~]# for (( i=1 ; i <= 3 ; i++ )) ; do env LANG=C mrtg /etc/mrtg/mrtg.cfg; done
[root@huatech ~]# indexmaker --columns=1 /etc/mrtg/mrtg.cfg > /var/www/mrtg/index.html
```

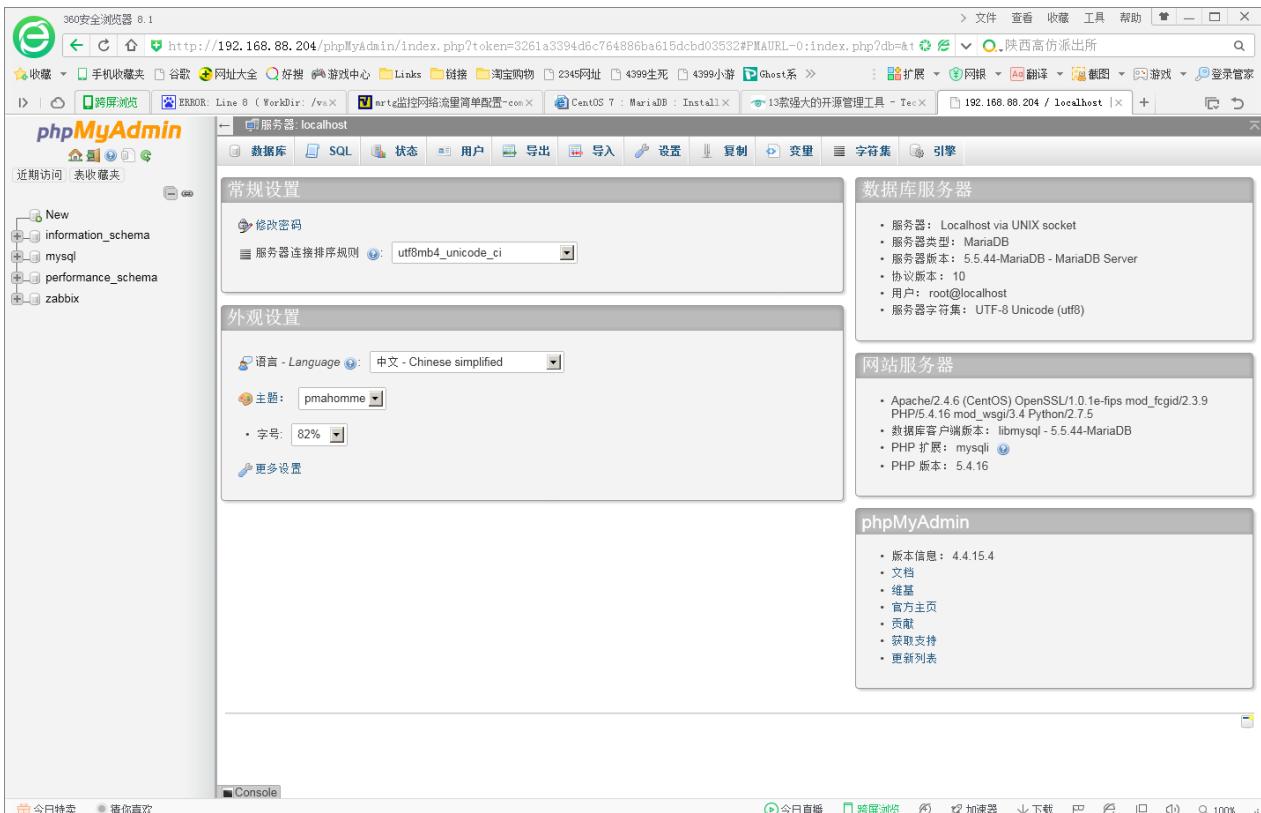
=====http 服务=====

```
[root@huatech ~]# vim /etc/snmp/snmpd.conf
proc httpd
[root@huatech ~]# systemctl restart snmpd
[root@huatech ~]# vim /etc/mrtg/mrtg.cfg
Target[httpd]: .1.3.6.1.4.1.2021.2.1.5.1&.1.3.6.1.4.1.2021.2.1.4.1:rdh@127.0.0
.1:::::2
MaxBytes[httpd]: 50
Unscaled[httpd]: dwmy
Options[httpd]: gauge, growright, nopercent
YLegend[httpd]: Count
ShortLegend[httpd]: proc(s).
Title[httpd]: Httpd Processes
Legend1[httpd]: Processes
Legend0[httpd]:
Legend1[httpd]: Httpd Processes
Legend2[httpd]:
PageTop[httpd]: <h1>Httpd Processes</h1>
[root@huatech ~]# for (( i=1 ; i <= 3 ; i++ )); do env LANG=C mrtg /etc/mrtg/mrtg.cfg;
done
[root@huatech ~]# indexmaker --columns=1 /etc/mrtg/mrtg.cfg >
/var/www/mrtg/index.html
```

=====Cacti 服务=====

Install httpd:

```
[root@huatech ~]# yum -y install httpd
[root@huatech ~]# systemctl restart httpd.service ;systemctl enable httpd
[root@huatech ~]# rm -rf /etc/httpd/conf/welcome.html
[root@huatech ~]# yum -y install php php-mbstring php-pear
[root@huatech ~]# yum -y install mariadb*
[root@huatech ~]# systemctl start mariadb;systemctl enable mariadb
[root@huatech ~]# mysql_secure_installation
[root@huatech ~]# yum --enablerepo=epel -y install phpMyAdmin php-mysql php-mcrypt
[root@huatech ~]# vim /etc/httpd/conf.d/phpMyAdmin.conf
Require ip 127.0.0.1 192.168.88.204/24
Require ip 127.0.0.1 192.168.88.0/24
```



```
[root@huatech ~]# yum --enablerepo=epel -y install cacti net-snmp net-snmp-utils  
php-mysql php-snmp rrdtool  
[root@huatech ~]# mysql -u root -p  
MariaDB [(none)]> create database cacti;  
Query OK, 1 row affected (0.05 sec)
```

```
MariaDB [(none)]> grant all privileges on cacti.* to catcti@'localhost' identified  
by 'password';  
Query OK, 0 rows affected (0.13 sec)
```

```
MariaDB [(none)]> flush privileges;  
Query OK, 0 rows affected (0.02 sec)
```

```
MariaDB [(none)]> exit
```

```
Bye
```

```
[root@huatech ~]# mysql -u catcti -p cacti </usr/share/doc/cacti-0.8.8b/cacti.sql  
Enter password:
```

```
[root@huatech ~]# vim /usr/share/cacti/include/config.php  
$database_type = "mysql";  
$database_default = "cacti";  
$database_hostname = "localhost";  
$database_username = "catcti";  
$database_password = "password";  
$database_port = "3306";  
$database_ssl = false;
```

```
[root@huatech ~]# vim /etc/httpd/conf.d/cacti.conf
Require ip 192.168.88.0/24
[root@huatech ~]# systemctl restart httpd
```

The screenshot shows a web browser window with the URL <http://192.168.88.204/cacti/install/>. The page title is "Cacti Installation Guide". The content includes a message of thanks for downloading and installing Cacti, instructions for reading the upgrade information file, and a license notice under the GNU General Public License. A "Next >>" button is at the bottom right.

The screenshot shows a second instance of the same web browser window, also displaying the Cacti Installation Guide page. This version includes a dropdown menu for selecting the type of installation, currently set to "New Install". The rest of the content is identical to the first screenshot.

360安全浏览器 8.1

http://192.168.88.204/cacti/install/index.php

收藏 手机收藏夹 谷歌 网址大全 好搜 游戏中心 Links 链接 淘宝购物 2345网址 4399生死 4399小游戏 Ghost系 > 扩展 网报 翻译 截图 游戏 登录管家

跨屏浏览 ERROR: Line 8 (WorkDir < /var/www/html/networkflow簡單配置 > CentOS 7 : Cacti : Setup < 163邮箱登陆_百度搜索 > (73封未读) 网易邮箱0.0MB < cacti > +)

[FOUND] RRDTool Binary Path: The path to the rrdtool binary.
/bin/rrdtool
[OK: FILE FOUND]

[FOUND] PHP Binary Path: The path to your PHP binary file (may require a php recompile to get this file).
/bin/php
[OK: FILE FOUND]

[FOUND] snmpwalk Binary Path: The path to your snmpwalk binary.
/bin/snmpwalk
[OK: FILE FOUND]

[FOUND] snmpget Binary Path: The path to your snmpget binary.
/bin/snmpget
[OK: FILE FOUND]

[FOUND] snmpbulkwalk Binary Path: The path to your snmpbulkwalk binary.
/bin/snmpbulkwalk
[OK: FILE FOUND]

[FOUND] snmpgetnext Binary Path: The path to your snmpgetnext binary.
/bin/snmpgetnext
[OK: FILE FOUND]

[FOUND] Cacti Log File Path: The path to your Cacti log file.
/usr/share/cacti/log/cacti.log
[OK: FILE FOUND]

SNMP Utility Version: The type of SNMP you have installed. Required if you are using SNMP v2c or don't have embedded SNMP support in PHP.
NET-SNMP 5.x

RRDTool Utility Version: The version of RRDTool that you have installed.
RRDTool 1.4.x

NOTE: Once you click "Finish", all of your settings will be saved and your database will be upgraded if this is an upgrade. You can change any of the settings on this screen at a later time by going to "Cacti Settings" from within Cacti.

Finish

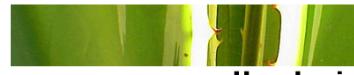
今日特卖 痞你喜欢 今日直播 跨屏浏览 加速器 下载 回到顶部 搜索 100% ..

360安全浏览器 8.1

http://192.168.88.204/cacti/index.php

收藏 手机收藏夹 谷歌 网址大全 好搜 游戏中心 Links 链接 淘宝购物 2345网址 4399生死 4399小游戏 Ghost系 > 扩展 网报 翻译 截图 游戏 登录管家

跨屏浏览 ERROR: Line 8 (WorkDir < /var/www/html/networkflow簡單配置 > CentOS 7 : Cacti : Setup < 163邮箱登陆_百度搜索 > (73封未读) 网易邮箱0.0MB < Login to Cacti > +)

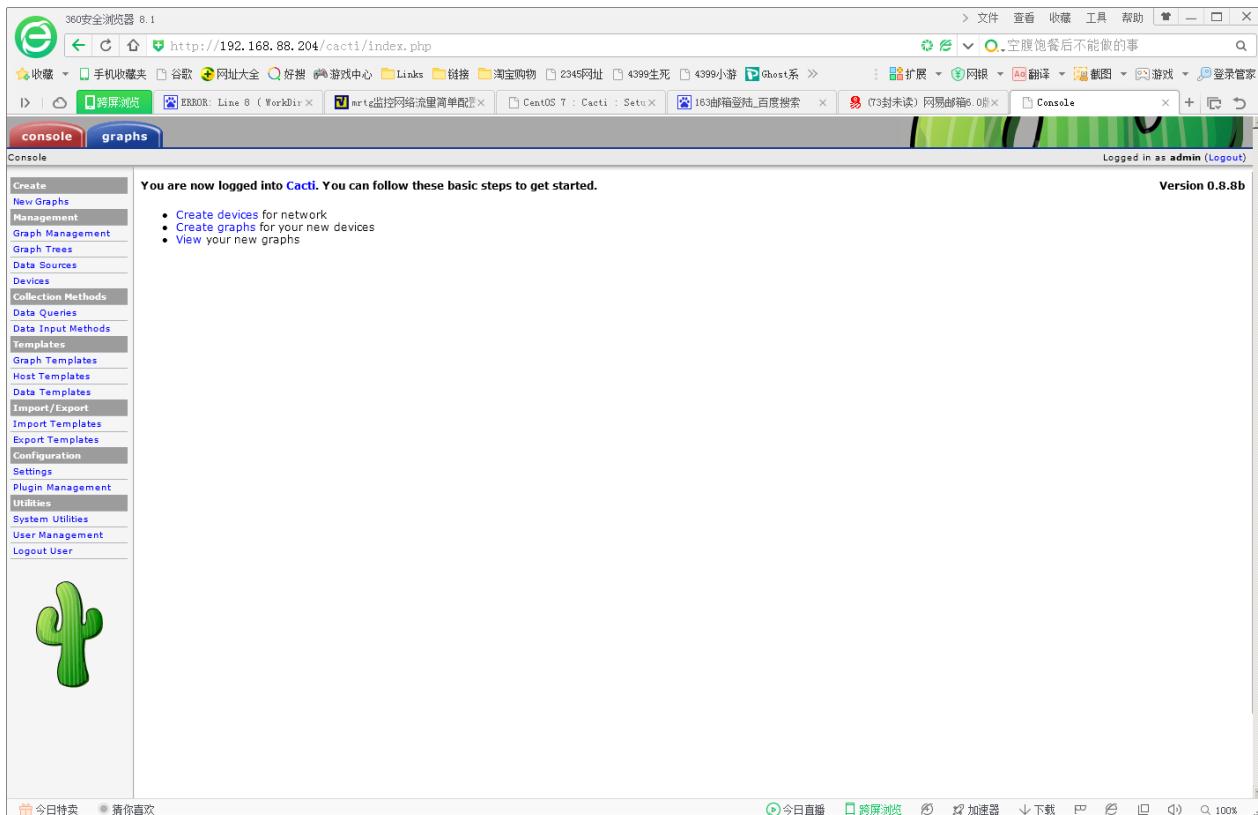
 User Login

Please enter your Cacti user name and password below:

User Name:
Password:

Login

今日特卖 痞你喜欢 今日直播 跨屏浏览 加速器 下载 回到顶部 搜索 100% ..



=====基本设置=====

360安全浏览器 8.1
http://192.168.88.204/cacti/host.php?action=edit&id=1
ERR: Line 8 (WorkDir)
Cacti : Basic
Console > Devices > (x) +

Availability/Reachability Options
Downed Device Detection
The method Cacti will use to determine if a host is available for polling.
NOTE: It is recommended that, at a minimum, SNMP always be selected.
Ping Timeout Value
The timeout value to use for host ICMP and UDP pings. This host SNMP timeout value applies for SNMP pings.
Ping Retry Count
After an initial failure, the number of ping retries Cacti will attempt before failing.
SNMP Options
SNMP Version
Choose the SNMP version for this device.
SNMP Community
SNMP read community for this device.
SNMP Port
Enter the UDP port number to use for SNMP (default is 161).
SNMP Timeout
The maximum number of milliseconds Cacti will wait for an SNMP response (does not work with php-snmp supports).
Maximum OID's Per Get Request
Specified the number of OID's that can be obtained in a single SNMP Get request.
Additional Options
Notes
Enter notes to this host.

Associated Graph Templates
Graph Template Name Status
1) Linux - Memory Usage Is Being Graphed (Edit)
2) Unix - Load Average Is Being Graphed (Edit)
3) Unix - Logged in Users Is Being Graphed (Edit)
4) Unix - Processes Is Being Graphed (Edit)
Add Graph Template: Cisco - CPU Usage Add

Associated Data Queries
Data Query Name Debugging Re-Index Method Status
1) Unix - Get Mounted Partitions (Verbose Query) None Success [6 Items, 3 Rows] ○ ×
Add Data Query: Karinet - Wireless Bridge Statistics Re-Index Method: Uptime Goes Backwards Add

今日特卖 疯狂喜欢 Return Save

360安全浏览器 8.1
http://192.168.88.204/cacti/graphs_new.php
ERR: Line 8 (WorkDir)
Cacti : Basic
Console > Create New Gx +

Logged in as admin (Logout)

Create New Graphs
Management Graph Management Graph Trees Data Sources Devices Collection Methods Data Queries Data Input Methods Templates Graph Templates Host Templates Data Templates Import/Export Import Templates Export Templates Configuration Settings Plugin Management Utilities System Utilities User Management Logout User

localhost (127.0.0.1) Local Linux Machine
Host: Localhost (127.0.0.1) Graph Types: All *Edit this Host *Create New Host

Graph Templates
Graph Template Name
Create: Linux - Memory Usage
Create: Unix - Load Average
Create: Unix - Logged in Users
Create: Unix - Processes
Create: (Select a graph type to create)

Data Query [Unix - Get Mounted Partitions]
Device Name Mount Point
/dev/mapper/centos-home /home
/dev/mapper/centos-root /
/dev/sda1 /boot

Cancel Create

今日特卖 疯狂喜欢

=====故障邮件通知=====

```
[root@huatech ~]# wget -P /usr/share/cacti/plugins
http://docs.cacti.net/media/plugin:settings-v0.71-1.tgz
[root@huatech ~]# tar zxvf /usr/share/cacti/plugins/plugin:settings-v0.71-1.tgz
-C /usr/share/cacti/plugins
```

The screenshot shows the Cacti Plugin Management interface. On the left sidebar, 'Plugin Management' is selected. In the main area, a table lists one plugin:

Action	Name	Version	Load Order	Description	Type	Status	Author
	Settings	0.71		Global Plugin Settings	System	Installed	Jimmy Conner

NOTE: Please sort by 'Load Order' to change plugin load ordering.
NOTE: SYSTEM plugins can not be ordered.

=====添加主机和设备=====

The screenshot shows the Cacti Device creation interface for a new host template. The 'Devices' section is selected in the sidebar. A red box highlights the 'SNMP Uptime' and 'Ping Timeout' fields, which are set to 400 and 1 respectively. Another red box highlights the 'SNMP Community' field, which is set to 'public'. The 'Notes' section at the bottom is also highlighted with a red box.

PEN

```
[root@rdh ~]# yum --enablerepo=epel -y install pen
[root@rdh ~]# yum --enablerepo=epel -y install pen
[root@rdh ~]# vim /etc/pen.conf
```

```
LOGFILE=/var/log/pen.log
WEBFILE=/var/www/pen/webstats.html
MAX_CONNECTIONS=256
XFORWARDEDFOR=true
ROUNDROBIN=true
PORT=80
BACKEND=2
SERVER1=192.168.88.204:80
SERVER2=192.168.88.221:80
[root@rdh ~]# vim /etc/rc.d/init.d/pend
[root@rdh ~]# vim /etc/rc.d/init.d/pend
#!/bin/bash

# pend: Start/Stop Pend
# chkconfig: - 90 10
# description: Pen is a light weight simple load balancer.
# pidfile: /var/run/pen.pid

. /etc/rc.d/init.d/functions
. /etc/pen.conf

LOCKFILE="/var/lock/subsys/pen"
PID=/var/run/pen.pid
PROG=/usr/bin/pen
PROGNAME=Pend

RETVAL=0
start() {
    SERVER=`grep "^\$SERVER" /etc/pen.conf | cut -d= -f2`"
    [ $XFORWARDEDFOR = "true" ] && SERVER="-H $SERVER"
    [ $ROUNDROBIN = "true" ] && SERVER="-r $SERVER"
    [ $SSLCERTS ] && SERVER="-E $SSLCERTS $SERVER"

    echo -n $"Starting $PROGNAME: "
    daemon $PROG $PORT -w $WEBFILE -x $MAX_CONNECTIONS -p $PID -l $LOGFILE -s
$BACKEND $SERVER
    RETVAL=$?
    echo
    [ $RETVAL -eq 0 ] && touch $LOCKFILE
    return $RETVAL
}
stop() {
    echo -n $"Stopping $PROGNAME: "
    killproc $PROG
```

```
RETVAL=$?
echo
[ $RETVAL -eq 0 ] && rm -f $PID $LOCKFILE
return $RETVAL
}
case "$1" in
start)
    start
    ;;
stop)
    stop
    ;;
status)
    status -p "$PID" -l $PROG $PROGNAME
    ;;
restart)
    stop
    start
    ;;
*)
    echo $"Usage: $0 {start|stop|status|restart}"
    exit 1
esac
exit $?
```

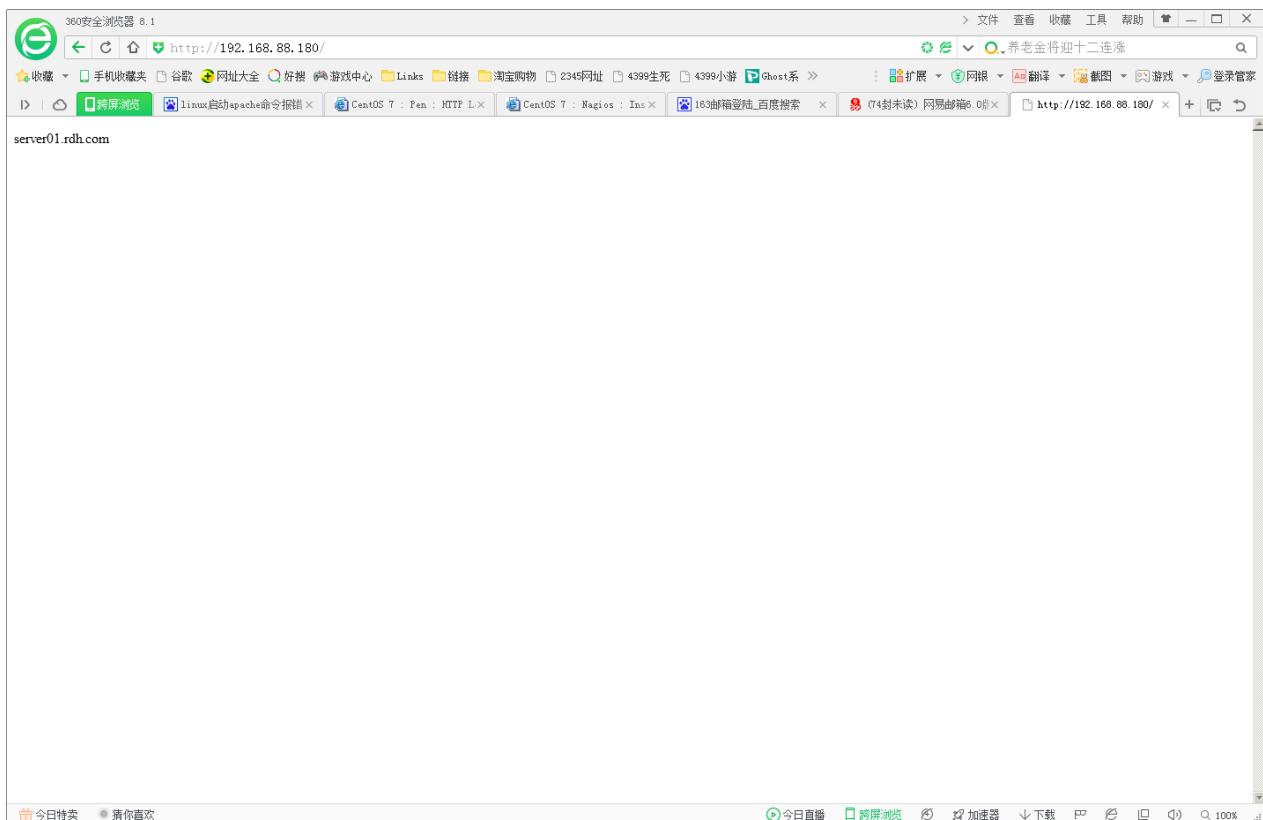
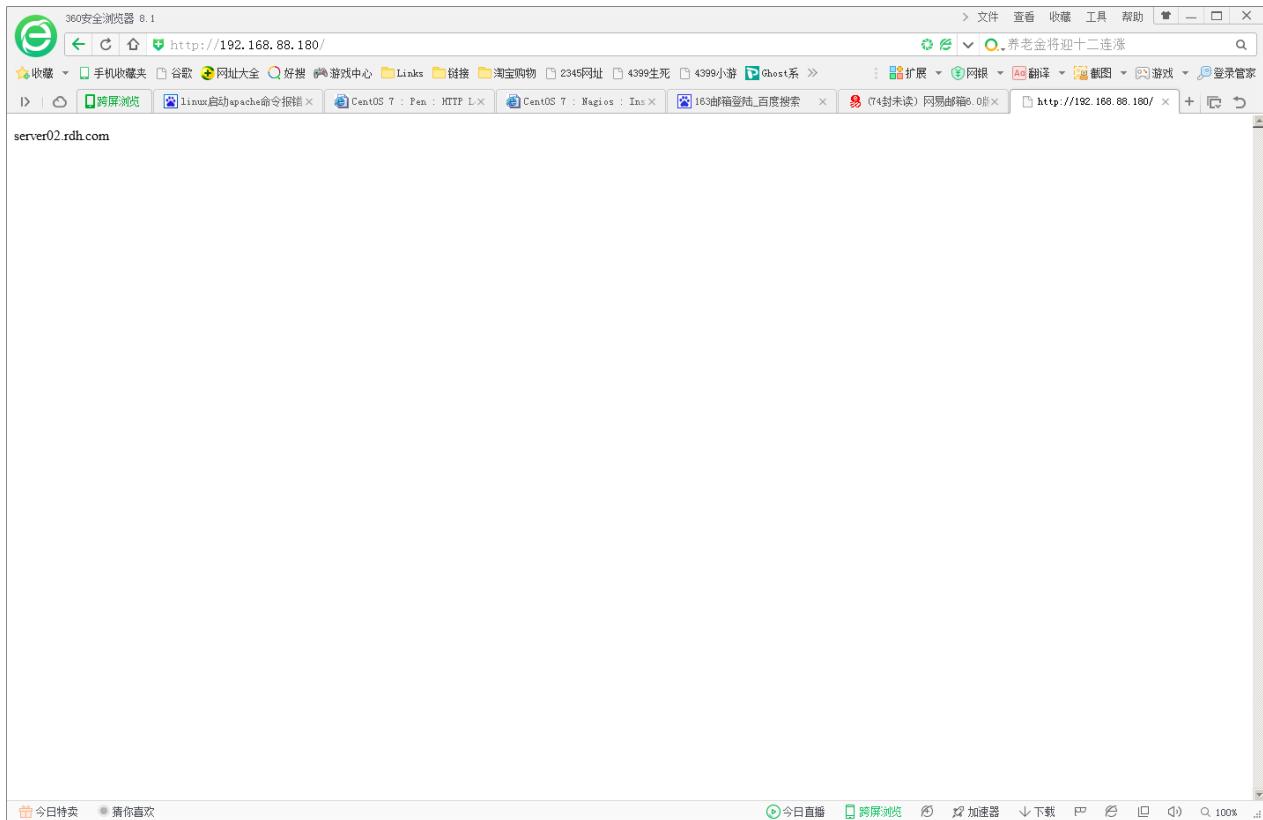
脚本地址: http://www.server-world.info/en/note?os=CentOS_7&p=pen&f=1

```
[root@rdh ~]# chmod 755 /etc/rc.d/init.d/pend
[root@rdh ~]# vim /usr/lib/systemd/system/pen.service
[Unit]
Description=Pend service
After=network.target
```

```
[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/etc/rc.d/init.d/pend start
ExecStop=/etc/rc.d/init.d/pend stop
```

```
[Install]
WantedBy=multi-user.target
[root@rdh ~]# systemctl start pen
[root@rdh ~]# systemctl enable pen
[root@rdh ~]# vim /etc/httpd/conf/httpd.conf
【客户端 http 配置】
```

```
LogFormat "\"%{X-Forwarded-For} i\" %l %u %t \"%r\" %>s %b \"%{Referer} i\" \"%{User-Agent} i\""
combined
[root@rdh ~]# systemctl restart httpd
```



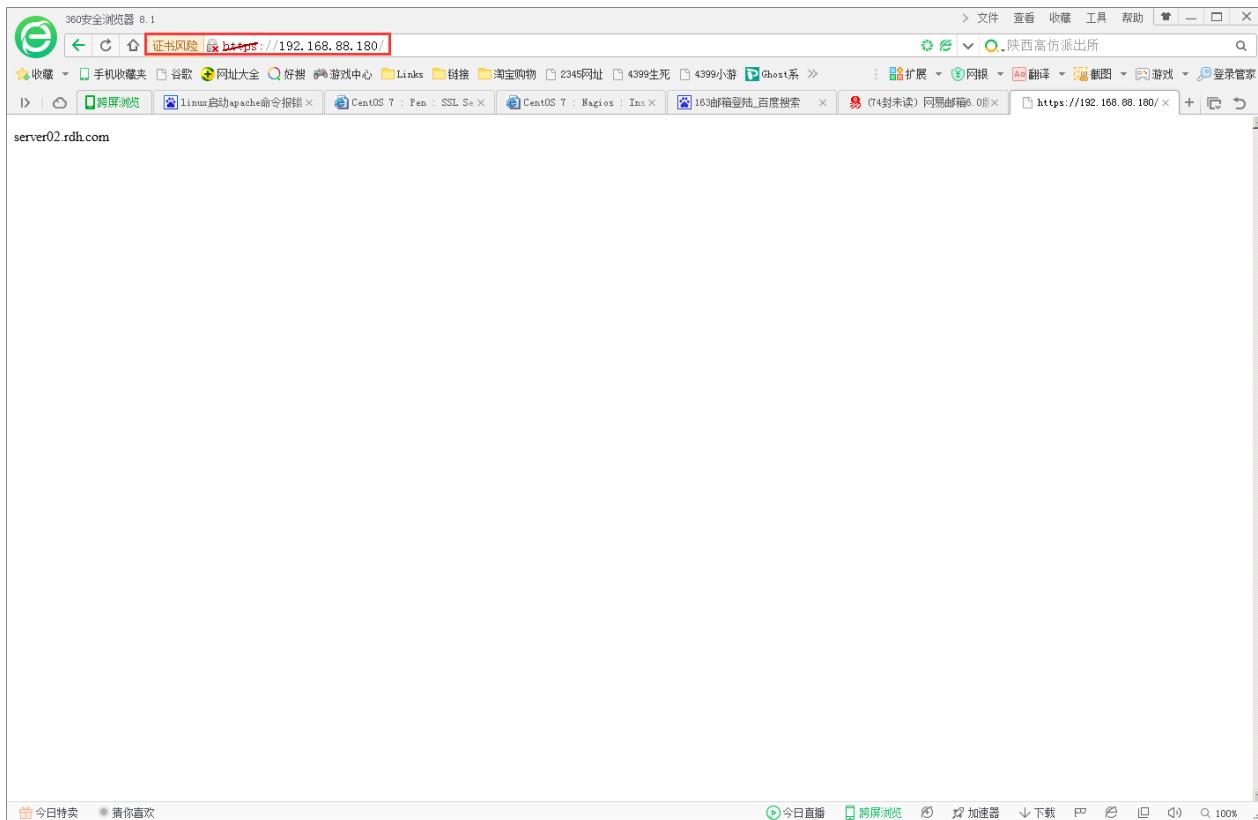
=====SSL 状态=====

```
[root@rdh ~]# cd /etc/pki/tls/certs  
[root@rdh certs]# openssl req -x509 -nodes -newkey rsa:2048 -keyout  
/etc/pki/tls/certs/pen.pem -out /etc/pki/tls/certs/pen.pem -days 365  
Generating a 2048 bit RSA private key  
.....+++  
.....+++  
writing new private key to '/etc/pki/tls/certs/pen.pem'
```

You are about to be asked to enter information that will be incorporated
into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [XX]:CH
State or Province Name (full name) []:HN
Locality Name (eg, city) [Default City]:ZZ
Organization Name (eg, company) [Default Company Ltd]:RDH
Organizational Unit Name (eg, section) []:IT
Common Name (eg, your name or your server's hostname) []:WMM
Email Address []:15136278634@163.com
[root@rdh certs]# chmod 600 pen.pem
[root@rdh ~]# vim /etc/pen.conf
PORT=443
SSLCERTS=/etc/pki/tls/certs/pen.pem



=====静态记录=====

```
[root@rdh ~]# cp /usr/share/doc/pen-*/* /var/www/pen
[root@rdh ~]# vim /var/www/pen/penstats
[root@rdh ~]# vim /etc/httpd/conf.d/pen.conf
<Directory /var/www/pen/>
    DirectoryIndex webstats.html
    Options ExecCGI
    <IfModule mod_authz_core.c>
        # Apache 2.4
        Require local
        Require ip 192.168.88.0/24
    </IfModule>
[root@rdh ~]# chmod 755 /var/www/pen/penstats
[root@rdh ~]# /var/www/pen/penstats > /dev/null
[root@rdh ~]# echo '*5 * * * * /var/www/pen/penstats > /dev/null' >
/etc/cron.d/pend
```

=====数据库负载 MariaDB =====

```
[root@huatech ~]# yum -y install mariadb*
[root@huatech ~]# systemctl start mariadb
[root@huatech ~]# systemctl enable mariadb
[root@rdh ~]# vim /etc/pen.conf
LOGFILE=/var/log/pen.log
WEBFILE=/var/www/pen/webstats.html
MAX_CONNECTIONS=256
XFORWARDEDFOR=true
```

ROUNDROBIN=true

PORT=3306

BACKEND=2

SERVER1=192.168.88.204:3306

SERVER2=192.168.88.221:3306

[root@rdh ~]# systemctl restart pen

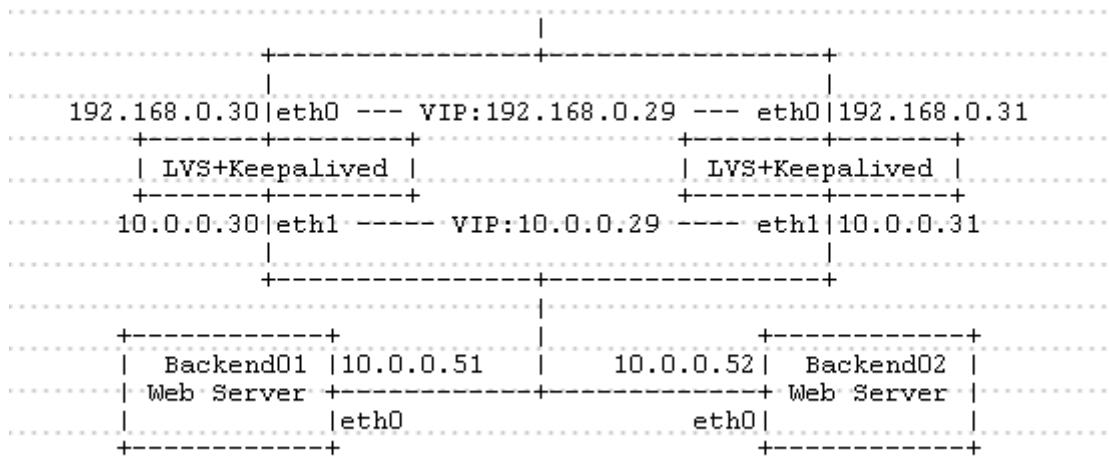
[root@rdh ~]# mysql -u keystone -p -h 10.0.0.30 keystone -e "select * from table01;"

四十六、负载均衡

LVS

```
[root@rdh ~]# yum -y install ipvsadm
[root@rdh ~]# echo 'net.ipv4.ip_forward = 1' >> /etc/sysctl.conf
[root@rdh ~]# sysctl -p
[root@rdh ~]# touch /etc/sysconfig/ipvsadm
[root@rdh ~]# systemctl start ipvsadm
[root@rdh ~]# systemctl enable ipvsadm
[root@rdh ~]# ipvsadm -C
[root@rdh ~]# ipvsadm -A -t 192.168.0.30:80 -s wlc
[root@rdh ~]# ipvsadm -a -t 192.168.88.180:80 -r 192.168.88.204:80 -m
[root@rdh ~]# ipvsadm -a -t 192.168.88.180:80 -r 192.168.88.221:80 -m
[root@rdh ~]# ipvsadm -l
```

LVS+KeepAlived



HTTP packets to the eth0 on LVS Server are forwarded to Backend01 and Backend02 Servers with NAT.

Change the default gateway to internal IP address of LVS on both Backend Web Servers first. (it's 10.0.0.29 on the example)

[root@rdh ~]# yum -y install ipvsadm keepalived

```
[root@rdh ~]# echo 'net.ipv4.ip_forward = 1' >> /etc/sysctl.conf
[root@rdh ~]# sysctl -p
[root@rdh ~]# touch /etc/sysconfig/ipvsadm
[root@rdh ~]# systemctl start ipvsadm
[root@rdh ~]# systemctl enable ipvsadm
[root@rdh ~]# mv /etc/keepalived/keepalived.conf
/etc/keepalived/keepalived.conf.org
[root@rdh ~]# vim /etc/keepalived/keepalived.conf
[root@rdh ~]# vim /etc/keepalived/keepalived.conf
[root@rdh ~]# systemctl start keepalived
[root@rdh ~]# systemctl enable keepalived
```

四十七、OpenLDAP 域服务器

四十八、FreeIPA

FreeIPA 是一款集成的安全信息管理解决方案。FreeIPA 包含 Linux (Fedora), 389 Directory Server、MIT Kerberos, NTP, DNS, Dogtag (Certificate System) 等等身份, 认证和策略功能。

```
[root@rdh ~]# yum -y install ipa-server bind bind-dyndb-ldap  
[root@rdh ~]# ipa-server-install --setup-dns  
[root@rdh ~]# yum -y install ipa-server-dns
```

```
Existing BIND configuration detected, overwrite? [no]: yes
```

```
The IPA Master Server will be configured with:  
Hostname: huatech.com  
IP address(es): 192.168.88.238, 192.168.88.238  
Domain name: com  
Realm name: COM
```

```
BIND DNS server will be configured to serve IPA domain with:  
Forwarders: No forwarders  
Reverse zone(s): 88.168.192.in-addr.arpa.
```

```
Continue to configure the system with these values? [no]:
```

Next steps:

1. You must make sure these network ports are open:

TCP Ports:

- * 80, 443: HTTP/HTTPS
- * 389, 636: LDAP/LDAPS
- * 88, 464: kerberos
- * 53: bind

UDP Ports:

- * 88, 464: kerberos
- * 53: bind
- * 123: ntp

```
[root@huatech ~]# kinit admin  
[root@huatech ~]# kinit admin  
Password for admin@COM:  
[root@huatech ~]# klist  
[root@huatech ~]# ipa config-mod --defaultshell=/bin/bash  
[root@huatech ~]# ipa user-add wmm --first=CentOS --last=Linux --password
```

```
[root@huatech ~]# ipa user-add cent --first=CentOS --last=Linux --password  
密码：  
再次输入 密码进行校验：  
-----  
新增用户 "cent"  
-----  
    用户登录名: cent  
    名: CentOS  
    姓: Linux  
    Full name: CentOS Linux  
    Display name: CentOS Linux  
    Initials: CL  
    Home directory: /home/cent  
    GECOS: CentOS Linux  
    登录shell: /bin/bash  
    Kerberos principal: cent@HUAUTCH.COM  
    邮件地址: cent@huatech.com  
    UID: 174200001  
    GID: 174200001  
    密码: True  
    Member of groups: ipausers  
    Kerberos keys available: True  
[root@huatech ~]# ipa user-find cent  
-----  
1 user matched  
-----  
    用户登录名: cent  
    名: CentOS  
    姓: Linux  
    Home directory: /home/cent  
    登录shell: /bin/bash  
    邮件地址: cent@huatech.com  
    UID: 174200001  
    GID: 174200001  
    Account disabled: False  
    密码: True  
    Kerberos keys available: True  
-----  
Number of entries returned 1  
-----
```

创建虚拟账号、而不是本地账号

```
[root@huatech ~]# useradd wmm  
useradd: 用户 “wmm” 已存在  
[root@huatech ~]# echo "jstvps" |passwd --stdin wmm  
更改用户 wmm 的密码。  
passwd: 所有的身份验证令牌已经成功更新。
```

【将本地账号导入到域账户中】

```
[root@huatech ~]# vim ipauser.sh
# extract local users who have 1000-9999 digit UID
# this is an example
#!/bin/bash

for line in `grep "x:[1-9][0-9][0-9][0-9]::" /etc/passwd`
do
    USER=`echo $line | cut -d: -f1`
    FIRST=`echo $line | cut -d: -f5 | awk {'print $1'}`` 
    LAST=`echo $line | cut -d: -f5 | awk {'print $2'}`` 

    [ ! "$FIRST" ] && FIRST=$USER
    [ ! "$LAST" ] && LAST=$USER

    echo $USER | ipa user-add $USER --first=$FIRST --last=$LAST --password
done
```

[root@huatech ~]# chmod +x ipauser.sh
[root@huatech ~]# ./ipauser.sh

```
[root@huatech ~]# ./ipauser.sh
-----
新增用户 "wmm"
-----
用户名: wmm
名: wmm
姓: wmm
Full name: wmm wmm
Display name: wmm wmm
Initials: ww
Home directory: /home/wmm
GECOS: wmm wmm
登录shell: /bin/bash
Kerberos principal: wmm@HUATECH.COM
邮件地址: wmm@huatech.com
UID: 174200003
GID: 174200003
密码: True
Member of groups: ipausers
Kerberos keys available: True
```

=====配置客户端=====

[root@server01 ~]# ipa dnsrecord-add huatech.com client01 --a-rec 192.168.88.221
【在服务器端】
[root@server01 ~]# clear
[root@server01 ~]# yum -y install ipa-client
[root@server01 ~]# nmcli connection modify eno1677728 ipv4.dns 192.168.88.238

```
[root@server01 ~]# nmcli connection down eno16777728;nmcli connection up eno16777728
[root@server01 ~]# ipa-client-install
[root@huatech ~]# ipa dnsrecord-add huatech.com server01 --a-rec 192.168.88.238
[root@server01 ~]# authconfig --enablemkhomedir --update
```

```
[root@client01 ~]# exit
logout

CentOS Linux 7 (Core)
Kernel 3.10.0-123.20.1.el7.x86_64 on an x86_64

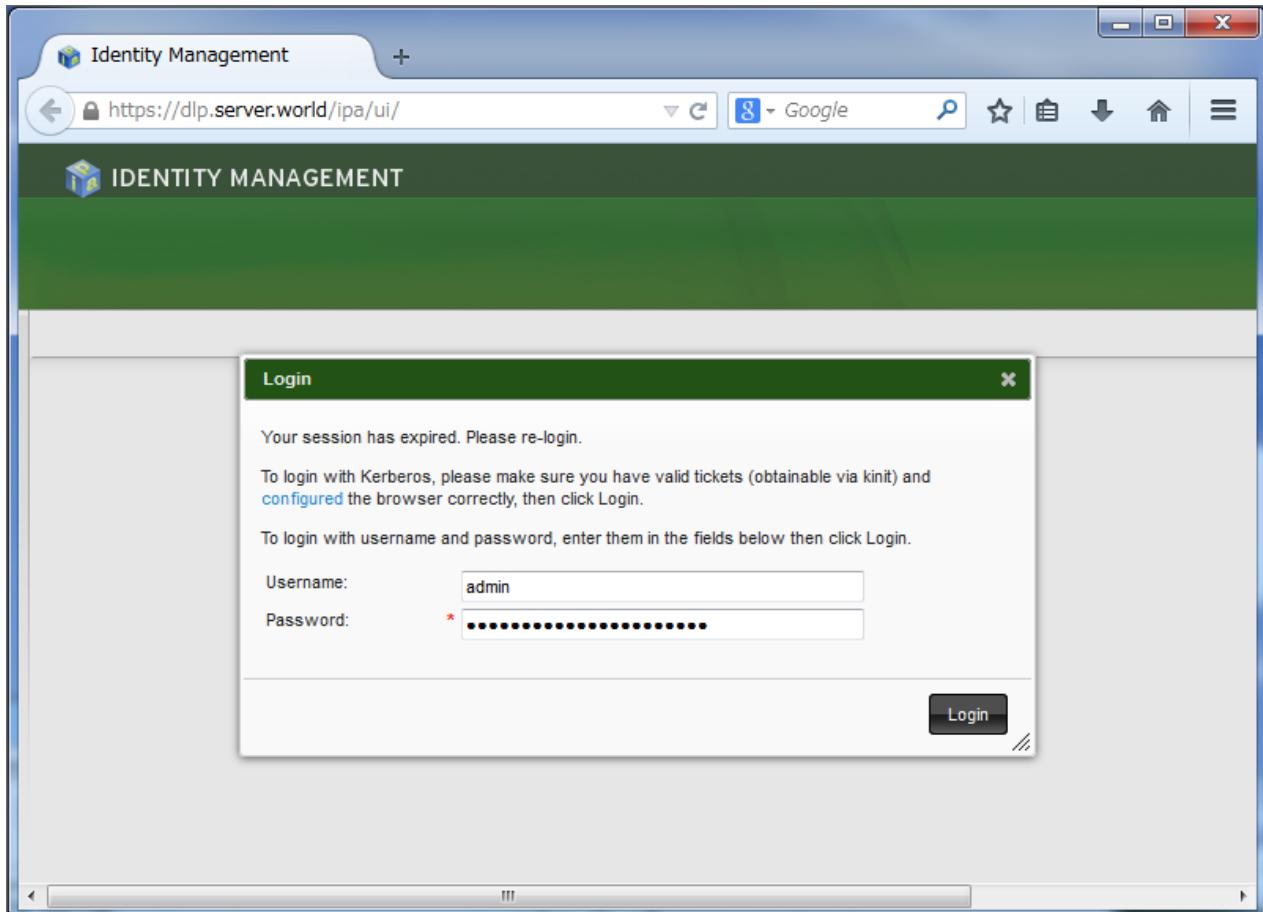
client01 login: redhat      # IPA user
Password:          # password
Password expired. Change your password now. # required to change the password when initial login
Current Password:    # current password
New password:        # new password
Retype new password:
Creating home directory for redhat.
[redhat@client01 ~]$      # just logined
```

=====基本操作=====

```
[root@server01 ~]# ipa user-add cent --first=CentOS --last=Linux --password
[root@server01 ~]# ipa user-disable cent
[root@server01 ~]# ipa user-enable cent
[root@server01 ~]# ipa user-find cent
[root@server01 ~]# ipa user-show --raw cent
[root@server01 ~]# ipa user-del cent
[root@server01 ~]# ipa group-add --desc='Development Group' development
[root@server01 ~]# ipa group-add-member --users=redhat,ubuntu development
[root@server01 ~]# ipa group-add-member --groups=development Hiroshima
[root@server01 ~]# ipa group-find development
[root@server01 ~]# ipa group-del Hiroshima
```

=====Web GUI=====

```
[root@huatech ~]# firefox
```

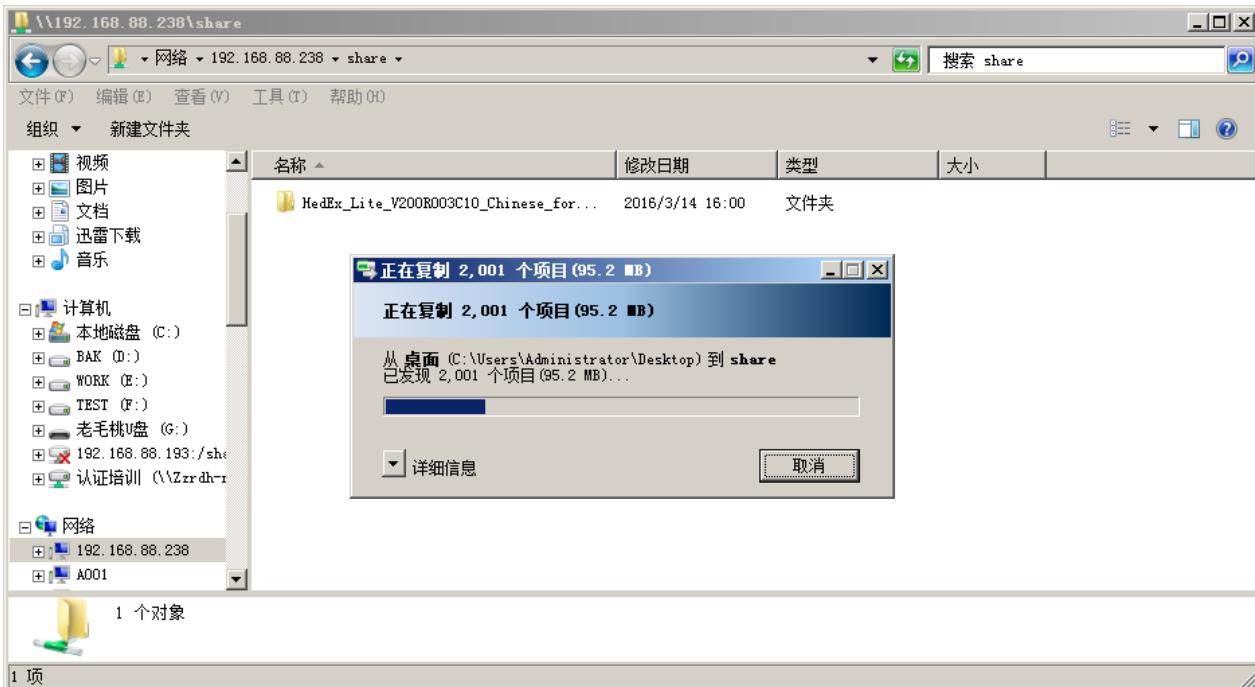


四十九、SAMBA 共享服务

=====Fully Access=====

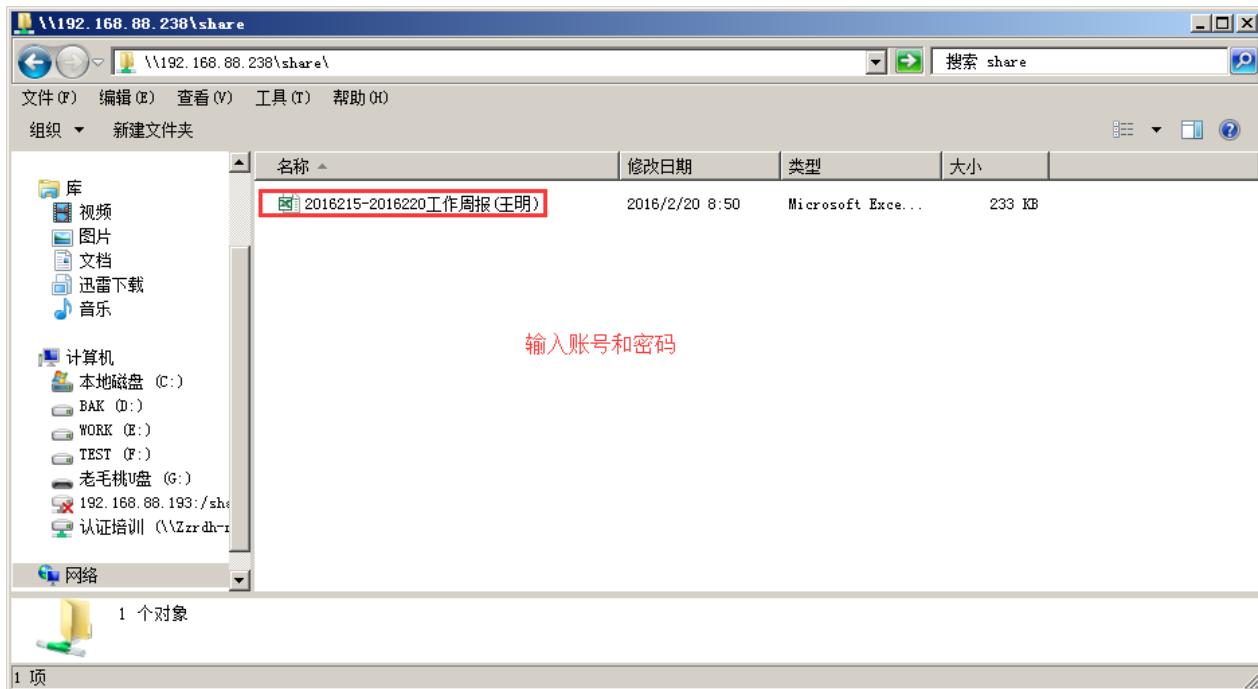
```
[root@huatech ~]# yum -y install samba samba-client
[root@huatech ~]# mkdir /share
[root@huatech ~]# chmod 777 /share
[root@huatech ~]# vim /etc/samba/smb.conf
unix charset = UTF-8
workgroup = WORKGROUP
90         server string = Samba Server Version %v
91         hosts allow = 127. 192.168.88.
security = user
124         passdb backend = tdb
125         map to guest = Bad User
[share]
322 path = /share
323 writable = yes
324 guest ok = yes
325 guest only = yes
326 create mode = 0777
327 directory mode = 0777
328 share modes = ye
```

```
[root@huatech ~]# systemctl start smb
[root@huatech ~]# systemctl start nmb
[root@huatech ~]# systemctl enable smb
[root@huatech ~]# systemctl enable nmb
```

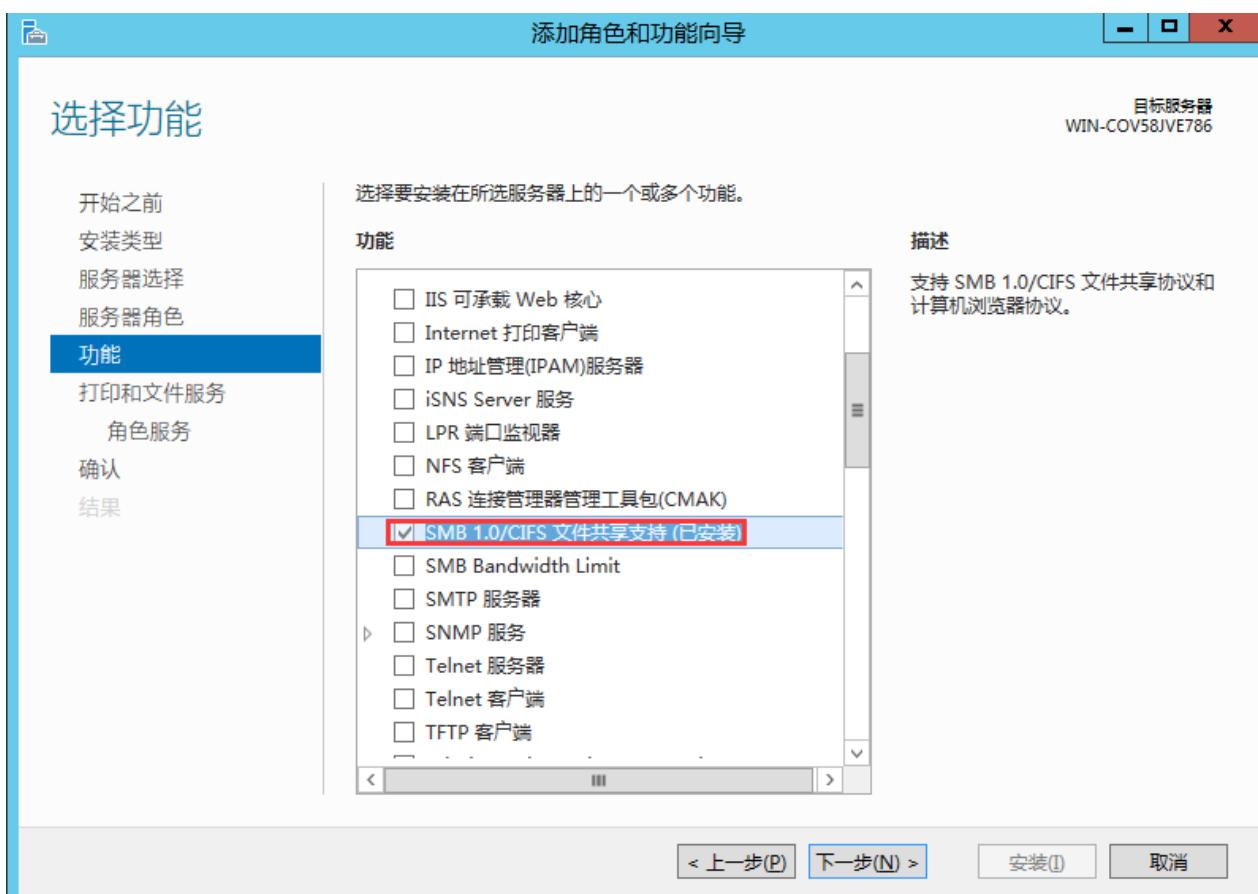


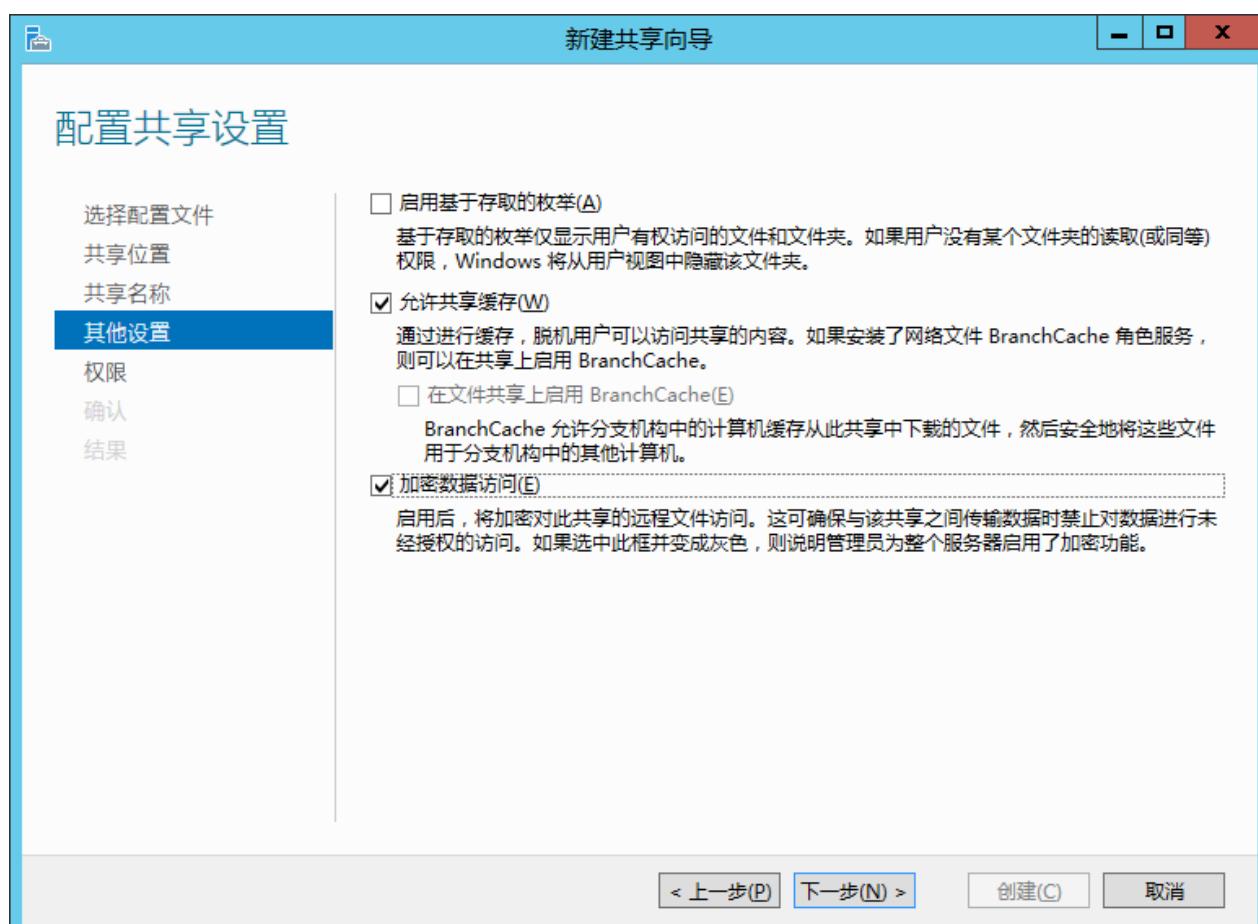
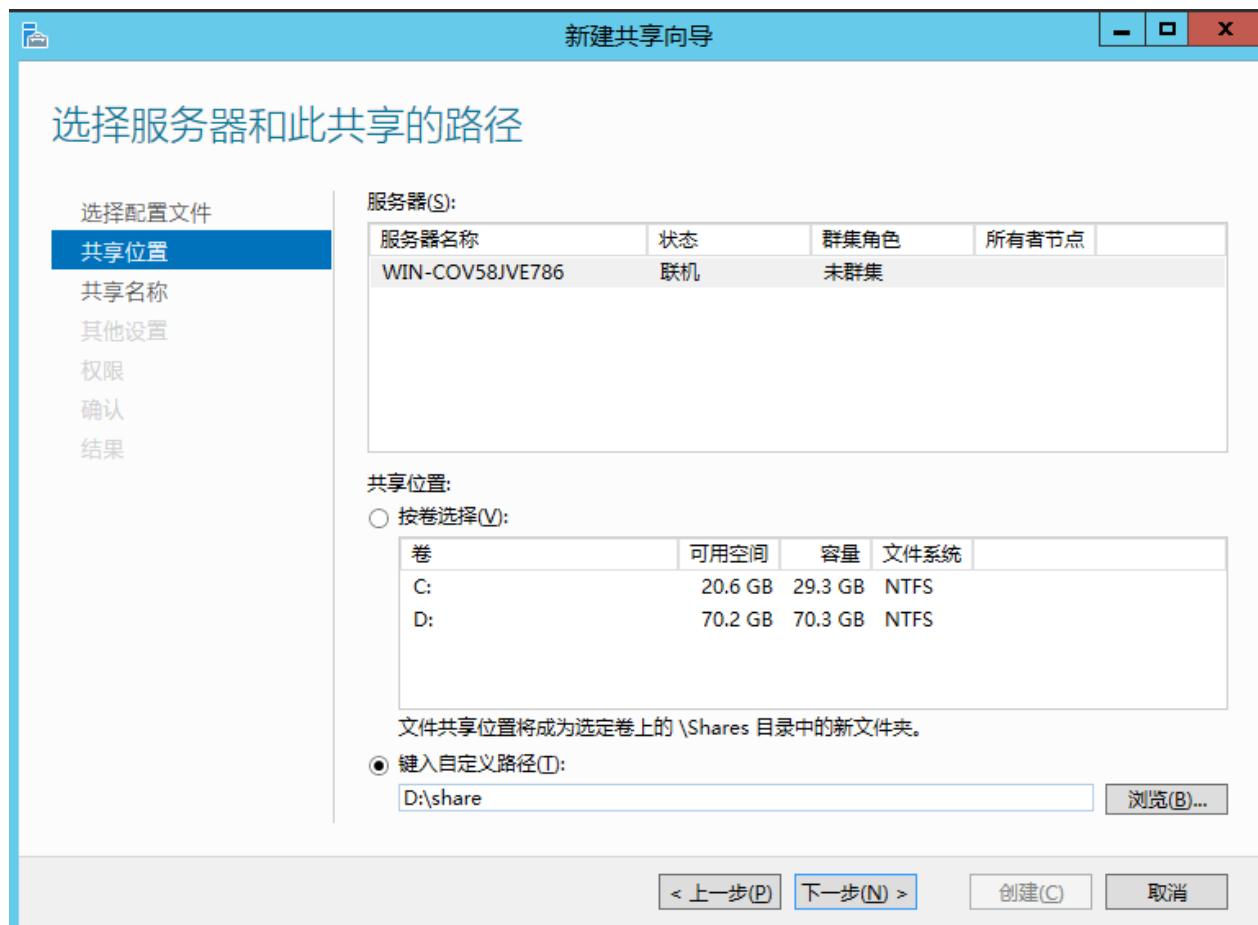
===== Limited Access =====

```
[root@huatech ~]# yum -y install samba samba-client
[root@huatech ~]# mkdir /share/security
[root@huatech ~]# groupadd security
[root@huatech ~]# chgrp security /share/security/
[root@huatech ~]# chmod 770 /share/security/
[root@huatech ~]# vim /etc/samba/smb.conf
[share]
path = /share/security
writable = yes
create mode = 0770
directory mode = 0770
share modes = yes
guest ok = no
valid users = @security
[root@huatech ~]# systemctl start smb
[root@huatech ~]# systemctl start nmb
[root@huatech ~]# systemctl enable smb
[root@huatech ~]# systemctl enable nmb
[root@huatech ~]# smbpasswd -a wmm
New SMB password:
Retype new SMB password:
Added user wmm.
[root@huatech ~]# usermod -G security wmm
```



=====windows Samba 服务器=====





=====windows 共享服务=====

下载地址: http://down9.3987.com:801/2010/ok_share.3987.com.rar

四十六、邮件服务器

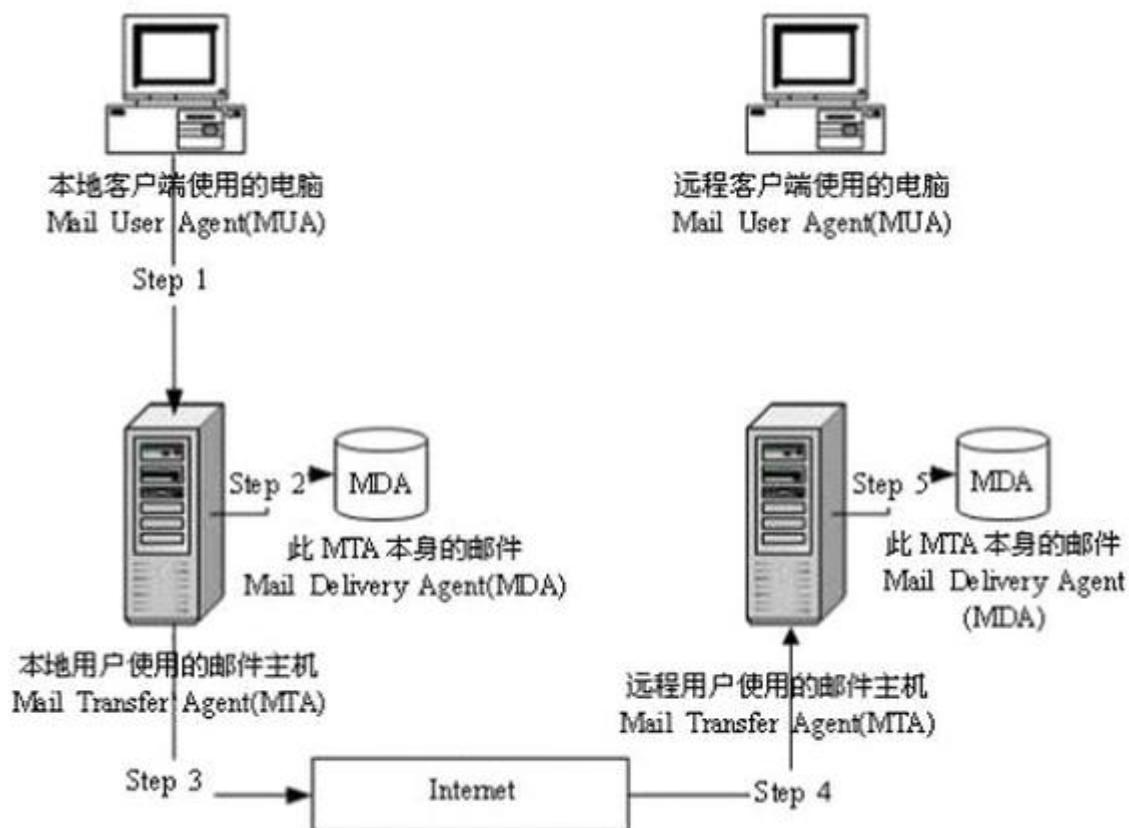


图 20-1 电子邮件从邮件主机寄送信件示意图

MUA (Mail User Agent) 接收邮件所使用的邮件客户端，使用 IMAP 或 POP3 协议与服务器通信，outlook、thunderbird、Mac mail、mutt，简单说就是一个邮件客户端。一般的客户端都接入 POP3、IMAP、SMTP，使用 IMAP、POP3、SMTP 与邮件服务器通信。

MTA (Mail Transfer Agent) 通过 SMTP 协议发送、转发邮件。

MDA (Mail Deliver Agent) 将 MTA 接收到的邮件保存到磁盘或指定地方，通常会进行垃圾邮件及病毒扫描。

MRA (Mail Receive Agent) 负责实现 IMAP 与 POP3 协议，与 MUA 进行交互；

SMTP (Simple Mail Transfer Protocol) 传输发送邮件所使用的标准协议；

IMAP (Internet Message Access Protocol) 接收邮件使用的标准协议之一；

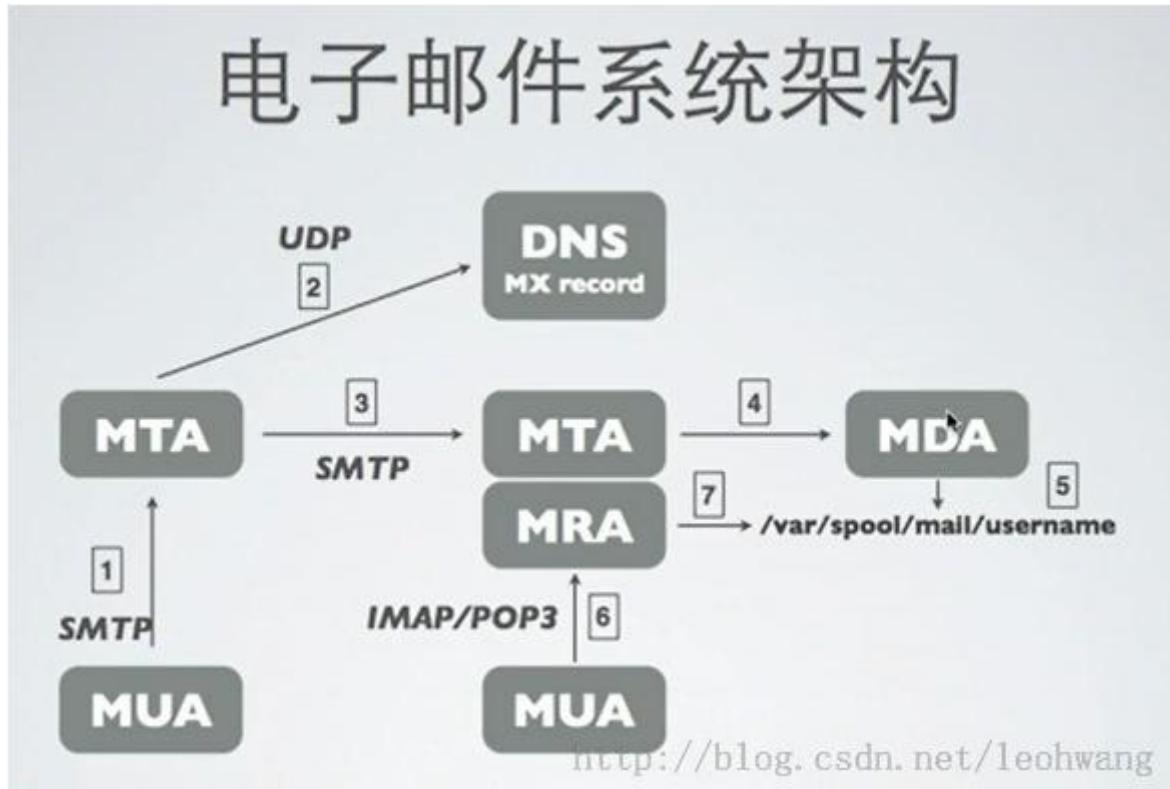
POP3 (Post Office Protocol 3) 接收邮件使用的标准协议之一。

常用的 MUA 有：outlook、thunderbird、Mac Mail、mutt；

常用的 MTA 服务有：sendmail、postfix；

常用的 MDA 有：procmail、dropmail；

常用的 MRA 有：dovecot。

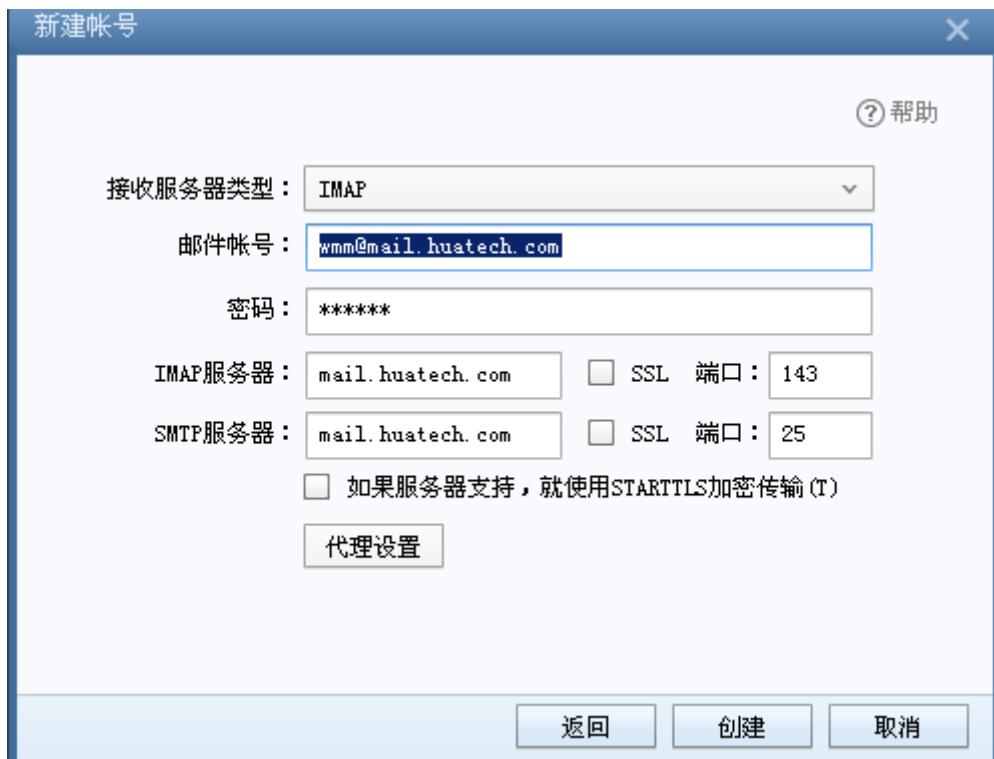


```

[root@huatech ~]# yum -y install postfix【MTA】【MRA】【MDA】
[root@huatech ~]# vim /etc/postfix/main.cf
myhostname = mail.huatech.com
mydomain = huatech.com
myorigin = $mydomain【发邮件地址，一般格式为@mail.huatech.com】
inet_interfaces = all
mydestination = $myhostname, localhost.$mydomain, localhost,$mydomain【本地邮件地址直接接收】
mynetworks = 127.0.0.1/24, 192.168.88.0/24
smtpd_banner = $myhostname ESMTP $mail_name
home_mailbox = Maildir/
message_size_limit = 10485760
mailbox_size_limit = 1073741824【限制邮件大小】
# for SMTP-Auth
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
smtpd_recipient_restrictions =
permit_mynetworks, permit_auth_destination, permit_sasl_authenticated, reject
[root@huatech ~]# systemctl restart postfix
[root@huatech ~]# systemctl enable postfix
  
```

```
[root@huatech ~]# yum -y install dovecot 【MRA】
[root@huatech ~]# vim /etc/dovecot/dovecot.conf
protocols = imap pop3 lmtp
listen = *
[root@huatech ~]# vim /etc/dovecot/conf.d/10-mail.conf
mail_location = maildir:~/Maildir
[root@huatech ~]# vim /etc/dovecot/conf.d/10-master.conf
unix_listener auth-userdb {
    mode = 0666
    user = postfix
    group = postfix
}
[root@huatech ~]# vim /etc/dovecot/conf.d/10-ssl.conf
ssl = no
[root@huatech ~]# vim /etc/dovecot/conf.d/10-auth.conf
disable_plaintext_auth = no
auth_mechanisms = plain login
[root@huatech ~]# systemctl start dovecot
[root@huatech ~]# systemctl enable dovecot
```





=====开启 SSL 支持=====

```
[root@huatech ~]# cd /etc/pki/tls/certs/
[root@huatech certs]# make server.key
[root@huatech certs]# openssl rsa -in server.key -out server.key
[root@huatech certs]# make server.csr
[root@huatech certs]# make server.csr
[root@huatech certs]# openssl x509 -in server.csr -out server.crt -req -signkey
server.key -days 3650
[root@huatech ~]# vim /etc/postfix/main.cf
# add follows to the end
smtpd_use_tls = yes
smtpd_tls_cert_file = /etc/pki/tls/certs/server.crt
smtpd_tls_key_file = /etc/pki/tls/certs/server.key
smtpd_tls_session_cache_database = btree:/etc/postfix/smtpd_scache
[root@huatech ~]# vim /etc/postfix/master.cf
smtps      inet  n       -       n       -       -           smtpd
-o syslog_name=postfix/smtps
-o smtpd_tls_wrappermode=yes
[root@huatech ~]# vim /etc/dovecot/conf.d/10-ssl.conf
ssl = yes
ssl_cert = </etc/pki/tls/certs/server.crt
ssl_key = </etc/pki/tls/certs/server.key
[root@huatech ~]# systemctl restart postfix
[root@huatech ~]# systemctl enable postfix
=====邮件防病毒=====
[root@huatech ~]# yum --enablerepo=epel -y install clamav clamav-update
```

```
[root@huatech ~]# sed -i -e "s/^Example/#Example/" /etc/freshclam.conf
[root@huatech ~]# freshclam
[root@huatech ~]# clamscan --infected --remove --recursive /home
[root@huatech ~]# clamscan --infected --remove --recursive /home
[root@huatech ~]# curl -O http://www.eicar.org/download/eicar.com
[root@huatech ~]# clamscan --infected --remove --recursive .
[root@huatech ~]# yum --enablerepo=epel -y install amavisd-new clamav-server
clamav-server-systemd
[root@huatech ~]# cp /usr/share/doc/clamav-server*/clamd.sysconfig
/etc/sysconfig/clamd.amavisd
[root@huatech ~]# vim /etc/sysconfig/clamd.amavisd
CLAMD_CONFIGFILE=/etc/clamd.d/amavisd.conf
CLAMD_SOCKET=/var/run/clamd.amavisd/clamd.sock
[root@huatech ~]# vi /etc/tmpfiles.d/clamd.amavisd.conf
d /var/run/clamd.amavisd 0755 amavis amavis -
[root@huatech ~]# vim /usr/lib/systemd/system/clamd@.service
[Install]
WantedBy=multi-user.target
[root@huatech ~]# systemctl start clamd@amavisd
[root@huatech ~]# systemctl enable clamd@amavisd
[root@huatech ~]# vim /etc/amavisd/amavisd.conf
$mydomain = 'huatech.com';
$myhostname = 'mail.huatech.com';
$notify_method = 'smtp:[127.0.0.1]:10025';
$forward_method = 'smtp:[127.0.0.1]:10025';
[root@huatech ~]# systemctl start amavisd
[root@huatech ~]# systemctl enable amavisd
[root@huatech ~]# systemctl start spamassassin
[root@huatech ~]# systemctl enable spamassassin
[root@huatech ~]# vim /etc/postfix/main.cf
content_filter=smtp-amavis:[127.0.0.1]:10024
[root@huatech ~]# vim /etc/postfix/master.cf
smtp-amavis unix - - n - 2 smtp
-o smtp_data_done_timeout=1200
-o smtp_send_xforward_command=yes
-o disable_dns_lookups=yes
127.0.0.1:10025 inet n - n - - smtpd
-o content_filter=
-o local_recipient_maps=
-o relay_recipient_maps=
-o smtpd_restriction_classes=
-o smtpd_client_restrictions=
-o smtpd_helo_restrictions=
-o smtpd_sender_restrictions=
```

```
-o smtpd_recipient_restrictions=permit_mynetworks,reject
-o mynetworks=127.0.0.0/8
-o strict_rfc821_envelopes=yes
-o smtpd_error_sleep_time=0
-o smtpd_soft_error_limit=1001
-o smtpd_hard_error_limit=1000
[root@huatech ~]# systemctl restart postfix
=====日志分析=====
[root@huatech ~]# yum -y install postfix-perl-scripts
[root@huatech ~]# perl /usr/sbin/pflogsumm -d yesterday /var/log/maillog
```

四十七、邮件服务一体机 EMOS

下载地址：http://mirror.extmail.org/iso/emos/EMOS_1.6_x86_64.iso

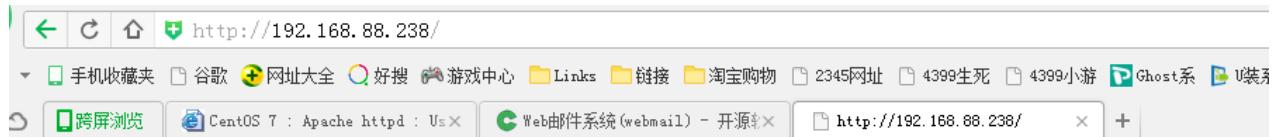


四十八、webmail 详解

SMTP SERVER:

```
[root@huatech ~]# yum -y install postfix
[root@huatech ~]# vim /etc/postfix/main.cf
myhostname = huatech.com
mydomain = mail.huatech.com
myorigin = $mydomain
inet_interfaces = all
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
mynetworks = 127.0.0.1/8, 192.168.88.0/24
home_mailbox = Maildir/
smtpd_banner = $myhostname ESMTP $mail_name
message_size_limit = 10485760 【单个邮件的大小就是 10M】
mailbox_size_limit = 1073741824 【单个邮箱的大小就是 10G】
```

```
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
smtpd_recipient_restrictions =
permit_mynetworks, permit_auth_destination, permit_sasl_authenticated, reject
[root@huatech ~]# systemctl restart postfix
[root@huatech ~]# systemctl enable postfix
POP3 SERVER|IMAP SERVER
[root@huatech ~]# yum -y install dovecot
[root@huatech ~]# vim /etc/dovecot/dovecot.conf
protocols = imap pop3 lmtp
listen = *
[root@huatech ~]# vim /etc/dovecot/conf.d/10-auth.conf
disable_plaintext_auth = no
auth_mechanisms = plain login
[root@huatech ~]# vim /etc/dovecot/conf.d/10-mail.conf
mail_location = maildir:~/Maildir
[root@huatech ~]# vim /etc/dovecot/conf.d/10-master.conf
unix_listener auth-userdb {
    mode = 0666
    user = postfix
    group = postfix
}
[root@huatech ~]# vim /etc/dovecot/conf.d/10-ssl.conf
ssl = no
[root@huatech ~]# systemctl start dovecot
[root@huatech ~]# systemctl enable dovecot
[root@huatech ~]# yum -y install httpd
[root@huatech ~]# yum -y install php php-mbstring php-pear
[root@huatech ~]# systemctl restart httpd
[root@huatech ~]# vim /var/www/html/index.php
```



2016/03/18

```
[root@huatech ~]# cd /etc/pki/tls/certs/
[root@huatech certs]# make server.key
[root@huatech certs]# openssl rsa -in server.key -out server.key
[root@huatech certs]# openssl rsa -in server.key -out server.key
[root@huatech certs]# make server.csr
[root@huatech certs]# openssl x509 -in server.csr -out server.crt -req -signkey
server.key -days 3650
[root@huatech ~]# yum -y install mod_ssl
[root@huatech ~]# vim /etc/httpd/conf.d/ssl.conf
DocumentRoot "/var/www/html"
ServerName huatech.com:44
SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/certs/server.key
```

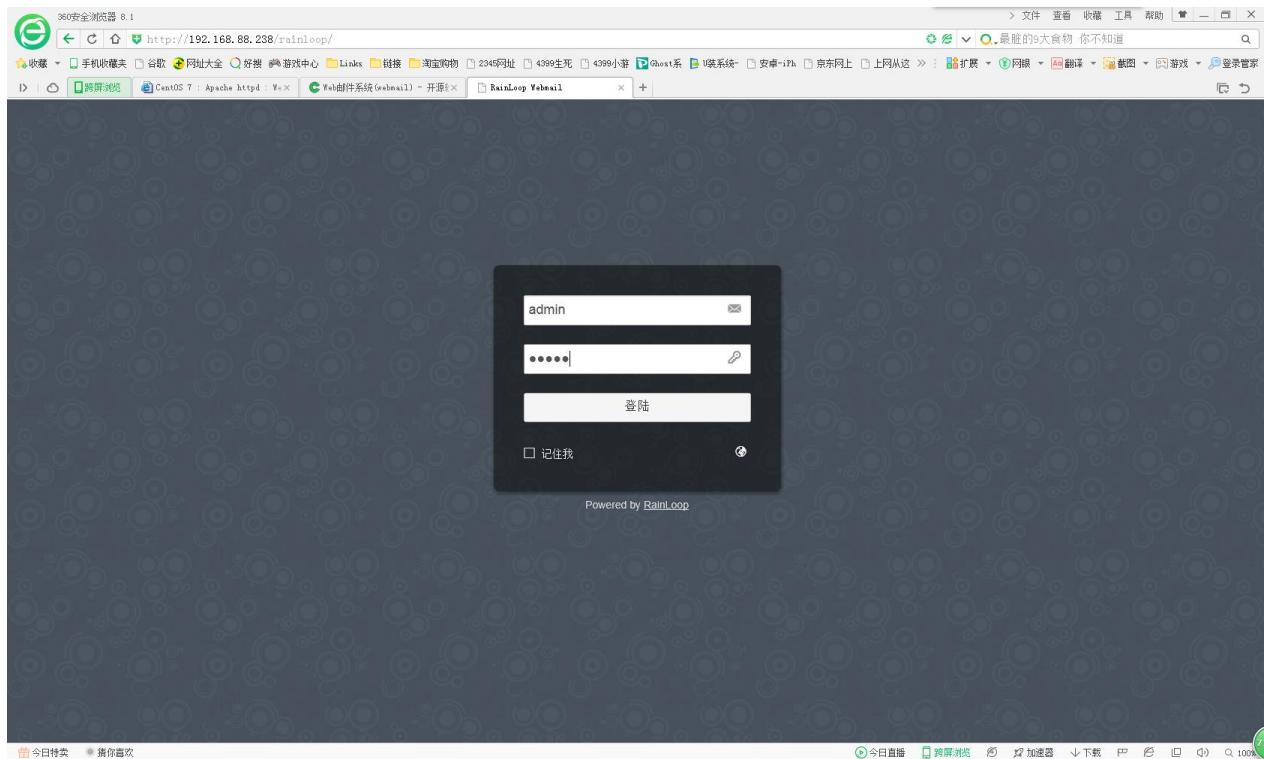


2016/03/18

```
[root@huatech ~]# yum -y install mariadb-server
[root@huatech ~]# systemctl start mariadb
[root@huatech ~]# mysql_secure_installation
[root@huatech ~]# mysql -u root -p
MariaDB [(none)]> create database roundcube;
MariaDB [(none)]> grant all privileges on roundcube.* to roundcube@'localhost'
identified by 'password';
```

```
MariaDB [(none)]> flush privileges;
[root@huatech ~]# yum --enablerepo=epel -y install roundcubemail
[root@huatech ~]# yum --enablerepo=epel -y install roundcubemail
[root@huatech ~]# cd /usr/share/roundcubemail/SQL
[root@huatech SQL]# mysql -u roundcube -p roundcube < mysql.initial.sql
[root@huatech ~]# cp -p /etc/roundcubemail/defaults.inc.php
/etc/roundcubemail/config.inc.php
[root@huatech ~]# vim /etc/roundcubemail/config.inc.php
$config['db_dsnw'] = 'mysql://roundcube:@localhost/roundcubemail';
$config['default_host'] = 'ssl://mail.huatech.com';
$config['default_port'] = 993;
$config['smtp_server'] = 'ssl://mail.huatech.com';
$config['smtp_port'] = 465;
$config['smtp_user'] = '%u';
$config['smtp_pass'] = '%p';
$config['smtp_auth_type'] = 'LOGIN';
$config['smtp_helo_host'] = 'mail.huatech.com';
$config['mail_domain'] = 'huatech.com';
$config['language'] = de_CH;
[root@huatech ~]# vim /etc/httpd/conf.d/roundcubemail.conf
Require ip 192.168.88.0/24
```

```
[root@huatech ~]# curl -O
http://repository.rainloop.net/v2/webmail/rainloop-latest.zip
[root@huatech ~]# mkdir /var/www/html/rainloop
[root@huatech ~]# unzip rainloop-latest.zip -d /var/www/html/rainloop
[root@huatech ~]# find /var/www/html/rainloop -type d -exec chmod 755 {} \;
[root@huatech ~]# find /var/www/html/rainloop -type f -exec chmod 644 {} \;
[root@huatech ~]# chown -R apache. /var/www/html/rainloop
```



```
[root@huatech ~]# yum --enablerepo=epel -y install squirrelmail
[root@huatech ~]# curl -O
http://www.squirrelmail.org/plugins/compatibility-2.0.16-1.0.tar.gz
[root@huatech ~]# curl -O
http://www.squirrelmail.org/plugins/empty\_trash-2.0-1.2.2.tar.gz
[root@huatech ~]# curl -O
http://www.squirrelmail.org/plugins/secure\_login-1.4-1.2.8.tar.gz
[root@huatech ~]# tar zxvf compatibility-2.0.16-1.0.tar.gz -C
/usr/share/squirrelmail/plugins
[root@huatech ~]# tar zxvf empty_trash-2.0-1.2.2.tar.gz -C
/usr/share/squirrelmail/plugins
[root@huatech ~]# tar zxvf secure_login-1.4-1.2.8.tar.gz -C
/usr/share/squirrelmail/plugins
[root@huatech ~]# rm -f ./*.tar.gz
[root@huatech ~]# /usr/share/squirrelmail/config/conf.pl
```

SquirrelMail Configuration : Read: config.php (1.4.0)

Main Menu —

- 1. Organization Preferences
 - 2. Server Settings
 - 3. Folder Defaults
 - 4. General Options
 - 5. Themes
 - 6. Address Books
 - 7. Message of the Day (MOTD)
 - 8. Plugins
 - 9. Database
 - 10. Languages
- D. Set pre-defined settings for specific IMAP servers
- C Turn color off
S Save data
Q Quit

Command >> 1

SquirrelMail Configuration : Read: config.php (1.4.0)

Organization Preferences

- | | | |
|---------------------------|---|---|
| 1. Organization Name | : | SquirrelMail |
| 2. Organization Logo | : | ./images/sm_logo.png |
| 3. Org. Logo Width/Height | : | (308/111) |
| 4. Organization Title | : | SquirrelMail \$version |
| 5. Signout Page | : | |
| 6. Top Frame | : | _top |
| 7. Provider link | : | http://squirrelmail.org/ |
| 8. Provider name | : | SquirrelMail |

- R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> 5

SquirrelMail Configuration : Read: config.php (1.4.0)

Organization Preferences

```
1. Organization Name      : SquirrelMail
2. Organization Logo      : ./images/sm_logo.png
3. Org. Logo Width/Height : (308/111)
4. Organization Title     : SquirrelMail $version
5. Signout Page           :
6. Top Frame               : _top
7. Provider link           : http://squirrelmail.org/
8. Provider name           : SquirrelMail
```

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> 5

When users click the Sign Out button they will be logged out and then sent to signout_page. If signout_page is left empty, (hit space and then return) they will be taken, as normal, to the default and rather sparse SquirrelMail signout page.

[]: /webmail

SquirrelMail Configuration : Read: config.php (1.4.0)

Organization Preferences

```
1. Organization Name      : SquirrelMail
2. Organization Logo      : ./images/sm_logo.png
3. Org. Logo Width/Height : (308/111)
4. Organization Title     : SquirrelMail $version
5. Signout Page           : /webmail
6. Top Frame               : _top
7. Provider link           : http://squirrelmail.org/
8. Provider name           : SquirrelMail
```

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> r

SquirrelMail Configuration : Read: config.php (1.4.0)

Main Menu —

- 1. Organization Preferences
 - 2. Server Settings
 - 3. Folder Defaults
 - 4. General Options
 - 5. Themes
 - 6. Address Books
 - 7. Message of the Day (MOTD)
 - 8. Plugins
 - 9. Database
 - 10. Languages

 - D. Set pre-defined settings for specific IMAP servers

 - C Turn color off
 - S Save data
 - Q Quit
- Command >> 2

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

- 1. Domain : localhost
- 2. Invert Time : false
- 3. Sendmail or SMTP : Sendmail

- A. Update IMAP Settings : localhost:143 (uw)
- B. Change Sendmail Config : /usr/sbin/sendmail

- R Return to Main Menu
- C Turn color off
- S Save data
- Q Quit

Command >> 1

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

1. Domain : localhost
2. Invert Time : false
3. Sendmail or SMTP : Sendmail

A. Update IMAP Settings : localhost:143 (uw)
B. Change Sendmail Config : /usr/sbin/sendmail

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> 1

The domain name is the suffix at the end of all email addresses. If for example, your email address is jdoe@example.com, then your domain would be example.com.

[localhost]: huatech.com

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

1. Domain : huatech.com
2. Invert Time : false
3. Sendmail or SMTP : Sendmail

A. Update IMAP Settings : localhost:143 (uw)
B. Change Sendmail Config : /usr/sbin/sendmail

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> 3

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

1. Domain : **huatech.com**
2. Invert Time : **false**
3. Sendmail or SMTP : **Sendmail**

A. Update IMAP Settings : **localhost:143 (uw)**
B. Change Sendmail Config : **/usr/sbin/sendmail**

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> **3**

You now need to choose the method that you will use for sending messages in SquirrelMail. You can either connect to an SMTP server or use sendmail directly.

1. Sendmail
2. SMTP
Your choice [1/2] [1]: **2**

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

1. Domain : **huatech.com**
2. Invert Time : **false**
3. Sendmail or SMTP : **SMTP**

A. Update IMAP Settings : **localhost:143 (uw)**
B. Update SMTP Settings : **localhost:25**

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> **A**

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

1. Domain : **huatech.com**
2. Invert Time : **false**
3. Sendmail or SMTP : **SMTP**

IMAP Settings

4. IMAP Server : **localhost**
5. IMAP Port : **143**
6. Authentication type : **login**
7. Secure IMAP (TLS) : **false**
8. Server software : **uw**
9. Delimiter : **/**

B. Update SMTP Settings : **localhost:25**
H. Hide IMAP Server Settings

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> **4**

IMAP Settings

4. IMAP Server : **localhost**
5. IMAP Port : **143**
6. Authentication type : **login**
7. Secure IMAP (TLS) : **false**
8. Server software : **uw**
9. Delimiter : **/**

B. Update SMTP Settings : **localhost:25**

H. Hide IMAP Server Settings

R Return to Main Menu

C Turn color off

S Save data

Q Quit

Command >> **4**

This is the hostname where your IMAP server can be contacted.

[**localhost**] : **huatech.com**

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General

```
1. Domain : huatech.com
2. Invert Time : false
3. Sendmail or SMTP : SMTP
```

IMAP Settings

```
4. IMAP Server : huatech.com
5. IMAP Port : 143
6. Authentication type : login
7. Secure IMAP (TLS) : false
8. Server software : uw
9. Delimiter : /
B. Update SMTP Settings : localhost:25
H. Hide IMAP Server Settings
```

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> 8

Command >> 8

Each IMAP server has its own quirks. As much as we tried to stick to standards, it doesn't help much if the IMAP server doesn't follow the same principles. We have made some work-arounds for some of these servers. If you would like to use them, please select your IMAP server. If you do not wish to use these work-arounds, you can set this to "other", and none will be used.

```
bincimap = Binc IMAP server
courier = Courier IMAP server
cyrus = Cyrus IMAP server
dovecot = Dovecot Secure IMAP server
exchange = Microsoft Exchange IMAP server
hmailserver = hMailServer
macosx = Mac OS X Mailserver
mercury32 = Mercury/32
uw = University of Washington's IMAP server
gmail = IMAP access to Google mail (Gmail) accounts
other = Not one of the above servers
```

[uw]: devocot

IMAP Settings

```
4. IMAP Server      : huatech.com
5. IMAP Port        : 143
6. Authentication type : login
7. Secure IMAP (TLS)   : false
8. Server software    : devcot
9. Delimiter         : /
```

```
B. Update SMTP Settings : localhost:25
H. Hide IMAP Server Settings
```

```
R  Return to Main Menu
C  Turn color off
S  Save data
Q  Quit
```

```
Command >> 9
```

```
Command >> 9
```

This is the delimiter that your IMAP server uses to distinguish between folders. For example, Cyrus uses '.' as the delimiter and a complete folder would look like 'INBOX.Friends.Bob', while UW uses '/' and would look like 'INBOX/Friends/Bob'. Normally this should be left at 'detect' but if you are sure you know what delimiter your server uses, you can specify it here.

To have it autodetect the delimiter, set it to 'detect'.

```
[/] : detect
```

IMAP Settings

4. IMAP Server : **huatech.com**
5. IMAP Port : **143**
6. Authentication type : **login**
7. Secure IMAP (TLS) : **false**
8. Server software : **devocot**
9. Delimiter : **detect**

B. Update SMTP Settings : **localhost:25**
H. Hide IMAP Server Settings

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> **B**

SMTP Settings

4. SMTP Server : **localhost**
5. SMTP Port : **25**
6. POP before SMTP : **false**
7. SMTP Authentication : **none**
8. Secure SMTP (TLS) : **false**
9. Header encryption key :

A. Update IMAP Settings : **huatech.com:143 (devocot)**
H. Hide SMTP Settings

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> **4**

SMTP Settings

```
4. SMTP Server : localhost
5. SMTP Port : 25
6. POP before SMTP : false
7. SMTP Authentication : none
8. Secure SMTP (TLS) : false
9. Header encryption key :

A. Update IMAP Settings : huatech.com:143 (devocot)
H. Hide SMTP Settings

R Return to Main Menu
C Turn color off
S Save data
Q Quit
```

Command >> 4

This is the hostname of your SMTP server.
[localhost]: **huatech.com**

General

1. Domain : **huatech.com**
2. Invert Time : **false**
3. Sendmail or SMTP : **SMTP**

SMTP Settings

4. SMTP Server : **huatech.com**
5. SMTP Port : **25**
6. POP before SMTP : **false**
7. SMTP Authentication : **none**
8. Secure SMTP (TLS) : **false**
9. Header encryption key :

A. Update IMAP Settings : **huatech.com:143 (devocot)**
H. Hide SMTP Settings

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> **7**

SMTP Settings

```
4. SMTP Server      : huatech.com
5. SMTP Port        : 25
6. POP before SMTP  : false
7. SMTP Authentication : none
8. Secure SMTP (TLS) : false
9. Header encryption key : 

A. Update IMAP Settings : huatech.com:143 (devocot)
H. Hide SMTP Settings

R Return to Main Menu
C Turn color off
S Save data
Q Quit
```

Command >> 7

If you have already set the hostname and port number, I can try to automatically detect the mechanisms your SMTP server supports.
Auto-detection is *optional* - you can safely say "n" here.

Try to detect auth mechanisms? [y/N]: n

Command >> 7

If you have already set the hostname and port number, I can try to automatically detect the mechanisms your SMTP server supports.
Auto-detection is *optional* - you can safely say "n" here.

Try to detect auth mechanisms? [y/N]: n

What authentication mechanism do you want to use for SMTP connections?
none - Your SMTP server does not require authorization.
login - Plaintext. If you can do better, you probably should.
plain - Plaintext. If you can do better, you probably should.
cram-md5 - Slightly better than plaintext.
digest-md5 - Privacy protection - better than cram-md5.

***** YOUR SMTP SERVER MUST SUPPORT THE MECHANISM YOU CHOOSE HERE *****
If you don't understand or are unsure, you probably want "none"

none, login, plain, cram-md5, or digest-md5 [**none**]: login

***** YOUR SMTP SERVER MUST SUPPORT THE MECHANISM YOU CHOOSE HERE *****
If you don't understand or are unsure, you probably want "none"

none, login, plain, cram-md5, or digest-md5 [none]: **login**
SMTP authentication uses IMAP username and password by default.

Would you like to use other login and password for all SquirrelMail
SMTP connections? [y/N]:**n**

Main Menu —

- 1. Organization Preferences
- 2. Server Settings
- 3. Folder Defaults
- 4. General Options
- 5. Themes
- 6. Address Books
- 7. Message of the Day (MOTD)
- 8. Plugins
- 9. Database
- 10. Languages

- D. Set pre-defined settings for specific IMAP servers

- C Turn color off
- S Save data
- Q Quit

Command >> **10**

SquirrelMail Configuration : Read: config.php (1.4.0)

Main Menu —

- 1. Organization Preferences
- 2. Server Settings
- 3. Folder Defaults
- 4. General Options
- 5. Themes
- 6. Address Books
- 7. Message of the Day (MOTD)
- 8. Plugins
- 9. Database
- 10. Languages

- D. Set pre-defined settings for specific IMAP servers

- C Turn color off
- S Save data
- Q Quit

Command >> 4

SquirrelMail Configuration : Read: config.php (1.4.0)

General Options

1. Data Directory	:	/var/lib/squirrelmail/prefs/
2. Attachment Directory	:	/var/spool/squirrelmail/attach/
3. Directory Hash Level	:	0
4. Default Left Size	:	150
5. Usernames in Lowercase	:	false
6. Allow use of priority	:	true
7. Hide SM attributions	:	false
8. Allow use of receipts	:	true
9. Allow editing of identity	:	true
Allow editing of name	:	true
Remove username from header	:	false
10. Allow server thread sort	:	true
11. Allow server-side sorting	:	true
12. Allow server charset search	:	true
13. Enable UID support	:	true
14. PHP session name	:	SQMSESSID
15. Location base	:	
16. Only secure cookies if poss.	:	true
17. Disable secure forms	:	false
18. Page referal requirement	:	

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> 7

Command >> 7

Hide SM attributions (y/n) [n]: y

```
[root@huatech ~]# cp
/usr/share/squirrelmail/plugins/secure_login/config.sample.php
/usr/share/squirrelmail/plugins/secure_login/config.php
[root@huatech ~]# vim /usr/share/squirrelmail/plugins/secure_login/config.php
$change_back_to_http_after_login = 0;
[root@huatech ~]# systemctl restart httpd
```

webmail

powered by Fedora and SquirrelMail

SquirrelMail Login

Name:

Password:

四十九、WEB 服务器

```
[root@rdh ~]# yum -y install httpd
```

```
[root@rdh ~]# rm -rf /etc/httpd/conf.d/welcome.conf
```

```
[root@rdh ~]# vim /etc/httpd/conf/httpd.conf
```

ServerAdmin root@rdh.com

```
ServerName rdh.com:80
```

```
AllowOverride all
```

```
DirectoryIndex index.html index.cgi index.php
```

```
[root@rdh ~]# systemctl start httpd
```

```
[root@rdh ~]# systemctl enable httpd
```

```
[root@rdh ~]# echo "test ok">>/var/www/html/index.html
```

【PERL 语言】

```
[root@rdh ~]# yum -y install perl perl-CGI
```

```
[root@rdh ~]# grep -n "^\*ScriptAlias" /etc/httpd/conf/httpd.conf
```

```
[root@rdh ~]# vim /etc/httpd/conf.d/cgi-enabled.conf
```

```
<Directory "/var/www/html/cgi-enabled">
```

```
    Options +ExecCGI
```

```
    AddHandler cgi-script .cgi .pl
```

```
</Directory>
```

```
[root@rdh ~]# systemctl restart httpd
```

```
[root@rdh ~]# vim /var/www/html/cgi-enabled/index.cgi
```

```
[root@rdh ~]# vim /var/www/html/cgi-enabled/index.cgi
```

```
[root@rdh ~]# chmod 755 /var/www/html/cgi-enabled/index.cgi
```

【PHP 脚本语言】

```
[root@rdh ~]# yum -y install php php-mbstring php-pear
```

```
[root@rdh ~]# systemctl restart httpd
```

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

```
<body>
```

```
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
```

```
<?php  
    print Date("Y/m/d");  
?>  
</div>  
</body>  
</html>
```

2016/03/18

【RUBY 脚本语言】

```
[root@rdh ~]# yum -y install ruby  
[root@rdh ~]# grep -n "^\*ScriptAlias" /etc/httpd/conf/httpd.conf  
<Directory "/var/www/html/cgi-enabled">  
    Options +ExecCGI  
    AddHandler cgi-script .cgi .rb  
</Directory>  
[root@rdh ~]# systemctl restart httpd  
[root@rdh ~]# vim /var/www/html/cgi-enabled/index.rb  
#!/usr/bin/ruby  
#  
#print "Content-type: text/html\n\n"  
#print "<html>\n<body>\n"  
#print "<div style=\"width: 100%; font-size: 40px; font-weight: bold; text-align: center;\">\n"  
#print "Ruby Script Test Page"  
#print "\n</div>\n"  
#print "</body>\n</html>\n"  
[root@rdh ~]# chmod 755 /var/www/html/cgi-enabled/index.rb
```

【

】

【

Ruby Script Test Page

【Python 脚本语言】

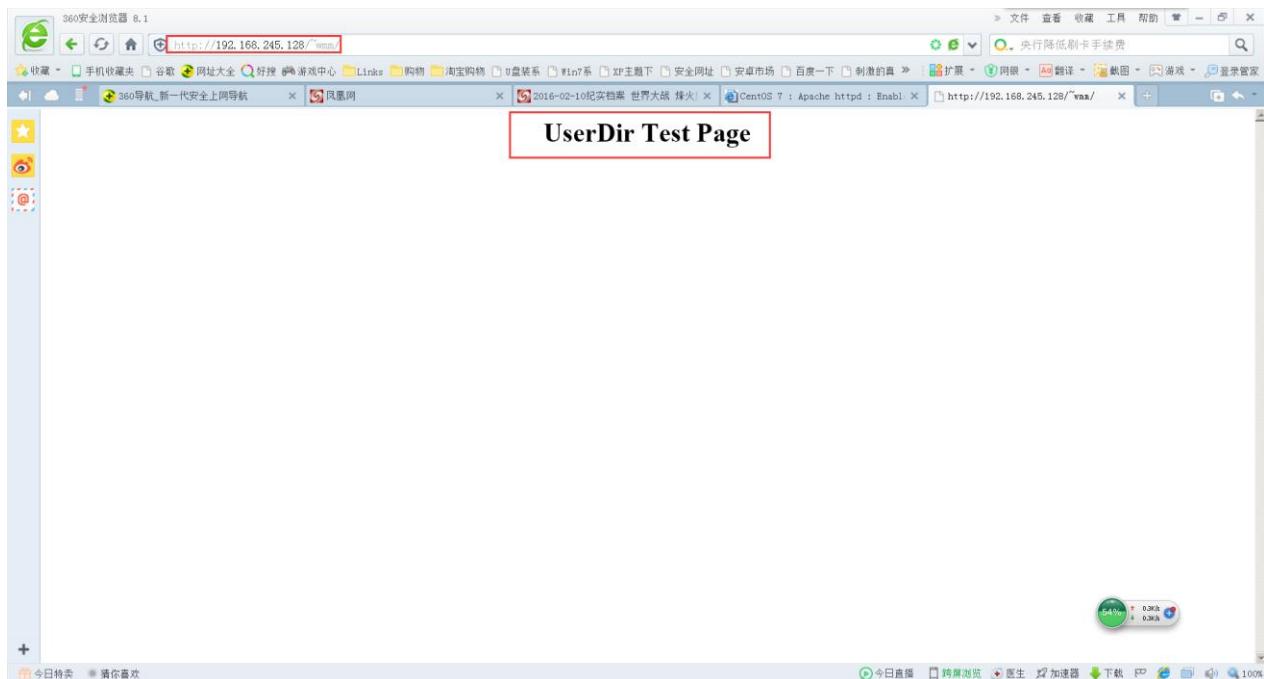
```
[root@rdh ~]# yum -y install python
[root@rdh ~]# grep -n "^\*ScriptAlias" /etc/httpd/conf/httpd.conf
[root@rdh ~]# vim /etc/httpd/conf.d/cgi-enabled.conf
<Directory "/var/www/html/cgi-enabled">
    Options +ExecCGI
    AddHandler cgi-script .cgi .py
</Directory>
[root@rdh ~]# systemctl restart httpd
[root@rdh ~]# vim /var/www/html/cgi-enabled/index.py
#!/usr/bin/env python

print "Content-type: text/html\n\n"
print "<html>\n<body>"
print "<div style=\"width: 100%; font-size: 40px; font-weight: bold; text-align: center;\">"
print "Python Script Test Page"
print "</div>\n</body>\n</html>"
[root@rdh ~]# chmod 755 /var/www/html/cgi-enabled/index.py
```

Python Script Test Page

【UserDIR】

```
[root@rdh ~]# vim /etc/httpd/conf.d/userdir.conf
#UserDir disabled
UserDir public_html
<Directory "/home/*public_html">
    AllowOverride ALL
    Options None
    Require method GET POST OPTIONS
</Directory>
[root@rdh ~]# systemctl restart httpd
[wmm@rdh ~]$ mkdir public_html
[wmm@rdh ~]$ chmod 711 /home/wmm/
[wmm@rdh ~]$ chmod 755 /home/wmm/public_html/
[wmm@rdh ~]$ vim ./public_html/index.html
<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
UserDir Test Page
</div>
</body>
</html>
```



【配置虚拟主机 VHOST】

```
[root@rdh ~]# vim /etc/httpd/conf.d/vhost.conf
<VirtualHost 192.168.245.128:8062>
DocumentRoot /var/www/html
ServerName 192.168.245.128
</VirtualHost>
<VirtualHost *:80>
DocumentRoot /home/wmm/public_html
ServerName huatech.com
ServerAdmin 151@163.com
ErrorLog logs/virtual.host-error_log
CustomLog logs/virtual.host-access_log combined
</VirtualHost>
```

【SSL 加密证书】

```
[root@rdh ~]# cd /etc/pki/tls/certs/
[root@rdh certs]# make server.key
[root@rdh certs]# openssl rsa -in server.key -out server.key
[root@rdh certs]# make server.csr
[root@rdh certs]# make server.csr
```

```
Country Name (2 letter code) [XX]:CH
State or Province Name (full name) []:HN
Locality Name (eg, city) [Default City]:ZH
Organization Name (eg, company) [Default Company Ltd]:HUATECH
Organizational Unit Name (eg, section) []:IT
Common Name (eg, your name or your server's hostname) []:Huatech.com
Email Address []:151@163.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

```
[root@rdh certs]# openssl x509 -in server.csr -out server.crt -req -signkey
server.key -days 3650
[root@rdh ~]# yum -y install mod_ssl
[root@rdh ~]# vim /etc/httpd/conf.d/ssl.conf
DocumentRoot "/var/www/html"
ServerName *:443
SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/certs/server.key
```



【BASIC 基本身份认证、虚拟账号】

```
[root@rdh ~]# vim /etc/httpd/conf.d/auth_basic.conf
<Directory /var/www/html/auth-basic>
AuthType Basic
AuthName "Basic Authentication"
AuthUserFile /etc/httpd/conf/.htpasswd
require valid-user
</Directory>
[root@rdh ~]# htpasswd -c /etc/httpd/conf/.htpasswd wmm
```

New password:

Re-type new password:

Adding password for user wmm

```
[root@rdh ~]# systemctl restart httpd
```

```
[root@rdh ~]# mkdir /var/www/html/auth-basic
```

```
[root@rdh ~]# vim /var/www/html/auth-basic/index.html
```

```
<html>
```

```
<body>
```

```
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
```

```
Test Page for Basic Auth
```

```
</div>
```

```
</body>
```

```
</html>
```

【基本身份认证系统身份认证】

```
[root@rdh ~]# yum -y install httpd-devel pam-devel gcc make
```

```
[root@rdh ~]# curl -L -0
```

https://mod-auth-external.googlecode.com/files/mod_authnz_external-3.3.2.tar.gz

```
[root@rdh ~]# curl -L -0
```

https://mod-auth-external.googlecode.com/files/mod_authnz_external-3.3.2.tar.gz

```
[root@rdh ~]# curl -L -0 https://pauth.googlecode.com/files/pauth-2.3.11.tar.gz
```

```
[root@rdh ~]# tar zxvf mod_authnz_external-3.3.2.tar.gz
```

```
[root@rdh ~]# cd mod_authnz_external-3.3.2
```

```
[root@rdh ~]# apxs -c mod_authnz_external.c
```

```
[root@rdh ~]# apxs -i mod_authnz_external.la
```

```
[root@rdh ~]# ar zxvf pauth-2.3.11.tar.gz
```

```
[root@rdh ~]# cd pauth-2.3.11
```

```
[root@rdh ~]# vim config.h
```

```
# line 126: comment out
```

```
/* #define SHADOW_SUN
```

```
# line 134: uncomment
```

```
#define PAM
```

```
# line 282: change to the httpd's ID )
```

```
#define SERVER_UIDS 48 /* user "apache" on the author's system */
```

```
[root@www pauth-2.3.11]# vi Makefile
```

```
# line 10: comment out
```

```
#LIB= -lcrypt
```

```
# line 14: uncomment
```

```
LIB=-lpam -ldl
```

```
[root@rdh ~]# make
```

```
[root@rdh ~]# cp pauth /usr/local/libexec/
```

```
[root@rdh ~]# chmod 4755 /usr/local/libexec/pauth
```

```
[root@rdh ~]# vim /etc/pam.d/pauth
```

```
# create new
# #%PAM-1.0
auth      include      system-auth
account   include      system-auth
session   include      system-auth
[root@rdh ~]# vim /etc/httpd/conf.d/auth_pam.conf
<Directory /var/www/html/auth-pam>
    SSLRequireSSL
    AuthType Basic
    AuthName "PAM Authentication"
    AuthBasicProvider external
    AuthExternal pwauth
    require valid-user
</Directory>
-- 插入 --
[root@rdh ~]# mkdir /var/www/html/auth-pam
[root@rdh ~]# vim /var/www/html/auth-pam/index.html
【身份认证 LDAP】
【1】Configure LDAP Server in your LAN
【2】create certificates
【3】编辑
[root@rdh ~]# yum -y install mod_ldap
[root@rdh ~]# vim /etc/httpd/conf.d/auth_ldap.conf
<Directory /var/www/html/auth-ldap>
    SSLRequireSSL
    AuthName "LDAP Authentication"
    AuthType Basic
    AuthBasicProvider ldap
    AuthLDAPURL
ldap://dlp.server.world/dc=server,dc=world?uid?sub?(objectClass=*)
    Require ldap-filter objectClass posixAccount
</Directory>
[root@rdh ~]# mkdir /var/www/html/auth-ldap/
[root@rdh ~]# mkdir /var/www/html/auth-ldap/
[root@rdh ~]# systemctl restart httpd
【Windows 域身份验证的方式】
[root@rdh ~]# yum -y install mod_auth_kerb
【PHP-FPM】
[root@rdh ~]# yum -y install php-fpm
[root@rdh ~]# vim /etc/httpd/conf.d/php.conf
<FilesMatch \.php$>
    # SetHandler application/x-httpd-php
    SetHandler "proxy:fcgi://127.0.0.1:9000"
</FilesMatch>
```

```
[root@rdh ~]# systemctl start php-fpm
[root@rdh ~]# systemctl enable php-fpm
[root@rdh ~]# echo '<?php phpinfo();?>'>/var/www/html/info.php
```



System	Linux rdh.com 3.10.0-327.el7.x86_64 #1 SMP Thu Nov 19 22:10:57 UTC 2015 x86_64
Build Date	Jun 23 2015 21:19:01
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/phar.ini, /etc/php.d/zip.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	API20100525,NTS

```
[root@rdh ~]# yum --enablerepo=epel -y install awstats
[root@rdh ~]# vim /etc/awstats/awstats.rdh.com.conf
LogFormat = 1
SiteDomain="rdh.com"
HostAliases="localhost 127.0.0.1 REGEX[rdh\.com\$] REGEX[^192\.168\.245\.]"
[root@rdh ~]# vim /etc/httpd/conf.d/awstats.conf
Require ip 192.168.245.0/24
[root@rdh ~]# systemctl restart httpd
[root@rdh ~]# /usr/share/awstats/wwwroot/cgi-bin/awstats.pl -config=rdh.com
--update
```

=====管理面板 WDCP=====

下载地址:

http://dl.wdlinux.cn:5180/wdos/iso/wdOS-1.0-x86_64.iso (64 位)

<http://dl.wdlinux.cn:5180/wdos/iso/wdOS-1.0-i386.iso> (32 位)

http://dl.wdlinux.cn:5180/wdos/iso/wdOS_md5sum.txt

手册教程：

http://www.wdlinux.cn/doc/wdos/wdOS-1.0-i386_install.doc

=====管理面板 HTTPS=====

下载地址：http://down4.zijidelu.org/projects/https/files/https-2.2-x86_64.iso

VPSMate 首页 功能特性 安装说明 使用手册 版本历史 在线演示 交流社区

历时一年倾心打造，全新版本 AppNode 开始公测啦，详情请看 www.appnode.com

VPSMate Linux 服务器 WEB 管理面板

- 快速在线安装、小巧且节省资源
- 当前支持 CentOS/Redhat 5.4+、6.x
- 基于发行版软件源的软件管理机制
- 轻松构建 Linux + Nginx + MySQL + PHP 环境
- 强大的在线文件管理和回收站机制
- 快速创建和安装多种站点
- 丰富实用的系统工具

VPSMate 首页 文件管理 文档管理 网站管理 FTP管理 计划任务 系统工具

欢迎使用 VPSMate！

当前版本：v1.0 b1

状态概况

服务器访问:	2012-10-11 00:12:56 CST
启动时间:	2012-10-04 11:29:43 CST
运行时间:	6天 12小时 43分 13秒
空闲时间:	6天 12小时 39分 38秒

CPU 使用率: 2.33%

当前负载:	1分钟负载: 0.15
	5分钟负载: 0.03
	15分钟负载: 0.01

内存使用:

总大小:	1.5G
已使用:	545.6M (36.2%)
剩余可用:	999.3M (63.7%)
Buffers:	60.4M

在线演示 开始安装（最新：v1.0 b10） 浏览更多功能截图 >>

[root@server02 ~]# bash -c "\$(curl <http://dl.appnode.com/install.sh>)"

=====windows 组件安装包=====

下载地址：<http://www.upupw.net/aphp52/n107.html>

The screenshot shows the UPUPW website's Apache PHP5.2 series download page. At the top, there's a navigation bar with links like '首页', 'UP简介', 'UPUPW下载' (highlighted in green), 'UPUPW教程', '技术资源', '赞助我们', 'UP论坛', and '5+技术支持' (with a red '服务' button). Below the navigation is a secondary menu with links for 'Apache环境包', 'Nginx环境包', 'Kangle环境包', 'UPUPW PHP探针', 'UPUPW扩展包', and 'UPUPW运行库'. The main content area shows the breadcrumb '首页 > UPUPW下载 > Apache环境包 > Apache PHP5.2系列'. The title 'Apache版UPUPW PHP5.2系列环境包1510' is displayed. A table header row has columns for '版本名称', '更新时间', and '软件下载'. Below this is a row for 'Apache版UPUPW PHP5.2系列环境包1510' with the date '2015-10-23 20:51:17' and a '云端下载' button. The central part of the page features a large 'APACHE' logo and 'PHP 5.2' text, with a sub-header 'UPUPW APACHE PHP 套件' and a '1510' badge. A note at the bottom left says: '特别说明：UPUPW APACHE版PHP5.2系列采用Apache+PHP+MariaDB的架构搭建，MariaDB和MySQL完全兼容使用方法一致省内存性能佳。如想更改搭配可以使用UPUPW绿色服务器平台组件扩展包<http://www.upupw.net/Expand/>'.

五十、Squid 代理服务器

```
[root@rdh ~]# yum -y install squid
[root@rdh ~]# vim /etc/squid/squid.conf
acl lan src 192.168.245.0/24
http_access allow lan
http_port 3128
request_header_access Referer deny all
request_header_access X-Forwarded-For deny all
request_header_access Via deny all
request_header_access Cache-Control deny all
# specify hostname
80 visible_hostname prox.server.world
# not display IP address
forwarded_for off
192.168.245.128 rdh.com
192.168.245.128 proxy.rdh.com
[root@rdh ~]# systemctl start squid
[root@rdh ~]# systemctl enable squid
配置客户端
[root@rdh ~]# vim /etc/profile
# add follows to the end (set proxy settings to the environment variables)
```

```
MY_PROXY_URL="http://prox.server.world:8080/"
HTTP_PROXY=$MY_PROXY_URL
HTTPS_PROXY=$MY_PROXY_URL
FTP_PROXY=$MY_PROXY_URL
http_proxy=$MY_PROXY_URL
https_proxy=$MY_PROXY_URL
ftp_proxy=$MY_PROXY_URL
export HTTP_PROXY HTTPS_PROXY FTP_PROXY http_proxy https_proxy ftp_proxy
[root@rdh ~]# source /etc/profile
[root@rdh ~]# vim /etc/yum.conf
proxy=http://prox.server.world:8080/
[root@rdh ~]# vim /etc/wgetrc
http_proxy = http://prox.server.world:8080/
https_proxy = http://prox.server.world:8080/
ftp_proxy = http://prox.server.world:8080/

【身份认证】
[root@rdh ~]# yum -y install httpd-tools
[root@rdh ~]# vim /etc/squid/squid.conf
auth_param basic program /usr/lib64/squid/basic_ncsa_auth /etc/squid/.htpasswd
auth_param basic children 5
auth_param basic realm Squid Basic Authentication
auth_param basic credentialsttl 5 hours
acl password proxy_auth REQUIRED
http_access allow password
[root@rdh ~]# htpasswd -c /etc/squid/.htpasswd cent
[root@rdh ~]# systemctl restart squid
[root@rdh ~]# vim /etc/profile
MY_PROXY_URL="http://cent:password@prox.server.world:8080/"
HTTP_PROXY=$MY_PROXY_URL
HTTPS_PROXY=$MY_PROXY_URL
FTP_PROXY=$MY_PROXY_URL
http_proxy=$MY_PROXY_URL
https_proxy=$MY_PROXY_URL
ftp_proxy=$MY_PROXY_URL
export HTTP_PROXY HTTPS_PROXY FTP_PROXY http_proxy https_proxy ftp_proxy
[root@rdh ~]# source /etc/profile
[root@rdh ~]# vim /etc/yum.conf
proxy=http://prox.server.world:8080/
proxy_username=cent
proxy_password=password
[root@rdh ~]# vim /etc/wgetrc
http_proxy = http://prox.server.world:8080/
https_proxy = http://prox.server.world:8080/
ftp_proxy = http://prox.server.world:8080/
```

```
proxy_user = cent
proxy_passwd = password
```

【反向代理服务器】

```
[root@rdh ~]# vim /etc/squid/squid.conf
http_access allow all
http_port 80 accel defaultsite=www.server.world
cache_dir ufs /var/spool/squid 100 16 256
cache_peer www.server.world parent 80 0 no-query originserver
cache_mem 256 MB
```

```
visible_hostname prox.server.world
```

```
[root@rdh ~]# systemctl start squid
[root@rdh ~]# systemctl enable squid
```

【上网行为管理】

```
[root@rdh ~]# yum --enablerepo=epel -y install squidGuard
[root@rdh ~]# mv /etc/squid/squidGuard.conf /etc/squid/squidGuard.conf.bk
[root@rdh ~]# vim /etc/squid/squidGuard.conf
#create new
dbhome /var/lib/squidGuard/db
logdir /var/log/squidGuard
# define 'deny' category
dest deny {
    # define prohibited domain list in 'deny' category
    domainlist deny/domains
    # define prohibited URL list in 'deny' category
    urllist deny/urls
}
acl {
    default {
        # permit all except 'deny' category
        pass !deny all
        # the redirected URL if matches 'deny'
        redirect http://www.server.world/error.html
    }
}
[root@rdh ~]# mkdir -p /var/lib/squidGuard/db/deny
[root@rdh ~]# vim /var/lib/squidGuard/db/deny/domains
yahoo.co.jp
example.com
[root@rdh ~]# vim /var/lib/squidGuard/db/deny/urls
www.yahoo.co.jp/deny/
www.example.com/
[root@rdh ~]# squidGuard -C all
[root@rdh ~]# chown -R squid. /var/lib/squidGuard/db/deny
[root@rdh ~]# vim /etc/squid/squid.conf
```

```
url_rewrite_program /usr/bin/squidGuard  
[root@rdh ~]# systemctl start squid  
[root@rdh ~]# systemctl enable squid
```

五十一、高速缓存服务器 Varnish

```
[root@rdh ~]# yum -y install varnish  
[root@rdh ~]# systemctl start varnish  
[root@rdh ~]# systemctl enable varnish  
[root@rdh ~]# vim /etc/varnish/default.vcl  
backend default {  
    .host = "127.0.0.1";  
    .port = "8080";  
}  
[root@rdh ~]# vim /etc/httpd/conf/httpd.conf  
Listen 8080  
[root@rdh ~]# netstat -tulnp|grep varnish  
[root@rdh ~]# varnishd -f /etc/varnish/default.vcl -s malloc,126M -T  
127.0.0.1:2000 -a 0.0.0.0:80  
[root@rdh ~]# curl -I http://192.168.245.128/  
HTTP/1.1 200 OK  
Date: Sat, 19 Mar 2016 13:51:53 GMT  
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2-fips mod_auth_kerb/5.4  
mod_fcgid/2.3.9 PHP/5.4.16 mod_wsgi/3.4 Python/2.7.5  
Last-Modified: Sat, 19 Mar 2016 11:28:18 GMT  
ETag: "8-52e65290ebddf"  
Content-Length: 8  
Content-Type: text/html; charset=UTF-8  
X-Varnish: 98319 65558  
Age: 5  
Via: 1.1 varnish-v4  
Connection: keep-alive  
参考文档：  
http://sofar.blog.51cto.com/353572/1653683/  
http://wenku.baidu.com/view/b380f09c5727a5e9846a6175.html?from=search
```

五十二、大数据 Hadoop

大数据 Hadoop 分为三个部分：namenode、datanode、高可用。

- 【1】master node huatech.com name node
- 【2】server01.rdh.com = (Slave Node)
- 【3】server02.rdh.com = (Slave Node)

=====配置 JDK 环境=====

```
[root@rdh ~]# curl -L0 -H "Cookie: oraclelicense=accept-securebackup-cookie" \  
> http://download.oracle.com/otn-pub/java/jdk/8u71-b15/jdk-8u71-linux-x64.rpm
```

```
[root@rdh ~]# rpm -Uvh jdk-8u71-linux-x64.rpm
[root@rdh ~]# vim /etc/profile
# add follows to the end
export JAVA_HOME=/usr/java/default
export PATH=$PATH:$JAVA_HOME/bin
export CLASSPATH=. :$JAVA_HOME/jre/lib:$JAVA_HOME/lib:$JAVA_HOME/lib/tools.jar
[root@rdh ~]# source /etc/profile
[root@rdh ~]# alternatives --config java
```

选项	命令
1	/usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java
2	/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.91-2.6.2.
*+ 3	/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.65-3.b17.
4	/usr/java/jdk1.8.0_71/jre/bin/java

按 Enter 保留当前选项 [+]，或者键入选项编号：4

```
[root@rdh ~]# vim day.java
import java.util.Calendar;
class day {
    public static void main(String[] args) {
        Calendar cal = Calendar.getInstance();
        int year = cal.get(Calendar.YEAR);
        int month = cal.get(Calendar.MONTH) + 1;
        int day = cal.get(Calendar.DATE);
        int hour = cal.get(Calendar.HOUR_OF_DAY);
        int minute = cal.get(Calendar.MINUTE);
        System.out.println(year + "/" + month + "/" + day + " " + hour + ":" + minute);
    }
}
[root@rdh ~]# javac day.java
[root@rdh ~]# java day
2016/3/20 10:5
```

【在所有节点上创建一个用户】

```
[root@rdh ~]# useradd -d /usr/hadoop hadoop
[root@rdh ~]# chmod 755 /usr/hadoop/
[root@rdh ~]# passwd hadoop
```

更改用户 hadoop 的密码。

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：

passwd：所有的身份验证令牌已经成功更新。

【所有节点上配置 SSH 信任】

```
[hadoop@rdh ~]$ ssh-keygen
```

```
[hadoop@rdh ~]$ ssh-copy-id localhost
[hadoop@rdh ~]$ ssh-copy-id node01.rdh.com
[hadoop@rdh ~]$ ssh-copy-id node02.rdh.com
=====安装Hadoop在所有节点上=====
[hadoop@rdh ~]$ curl -O
http://ftp.jaist.ac.jp/pub/apache/hadoop/common/hadoop-2.7.1/hadoop-2.7.1.tar.gz
[hadoop@rdh ~]$ tar zxvf hadoop-2.7.1.tar.gz -C /usr/hadoop --strip-components 1
[hadoop@rdh ~]$ vim ~/.bash_profile
# add follows to the end
export HADOOP_HOME=/usr/hadoop
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
[hadoop@rdh ~]$ source ~/.bash_profile
=====主节点配置Hadoop=====
[hadoop@rdh ~]$ mkdir ~/datanode
[hadoop@rdh ~]$ ssh node1.rdh.com "mkdir ~/datanode"
[hadoop@rdh ~]$ ssh node2.rdh.com "mkdir ~/datanode"
[hadoop@rdh ~]$ vim ~/etc/hadoop/hdfs-site.xml
<property>
  <name>dfs.replication</name>
  <value>2</value>【2个副本、一般三个副本】
</property>
<property>
  <name>dfs.datanode.data.dir</name>
  <value>file:///usr/hadoop/datanode</value>
</property>
[hadoop@rdh ~]$ scp ~/etc/hadoop/hdfs-site.xml node01.server.world:~/etc/hadoop/
[hadoop@rdh ~]$ scp ~/etc/hadoop/hdfs-site.xml node02.server.world:~/etc/hadoop/
[hadoop@rdh ~]$ vim ~/etc/hadoop/core-site.xml
<property>
  <name>fs.defaultFS</name>
  <value>hdfs://d1p.server.world:9000/</value>
</property>
[hadoop@rdh ~]$ scp ~/etc/hadoop/core-site.xml node01.server.world:~/etc/hadoop/
[hadoop@rdh ~]$ scp ~/etc/hadoop/core-site.xml node02.server.world:~/etc/hadoop/
[hadoop@rdh ~]$ sed -i -e 's/\${JAVA_HOME}/\${usr}/java/default/' ~/etc/hadoop/hadoop-env.sh
[hadoop@rdh ~]$ scp ~/etc/hadoop/hadoop-env.sh node01.server.world:~/etc/hadoop/
```

```
[hadoop@rdh ~]$ scp ~/etc/hadoop/hadoop-env.sh node02.server.world:~/etc/hadoop/
[hadoop@rdh ~]$ mkdir ~/namenode
[hadoop@rdh ~]$ vim ~/etc/hadoop/hdfs-site.xml
<property>
  <name>dfs.namenode.name.dir</name>
  <value>file:///usr/hadoop/namenode</value>
</property>
[hadoop@rdh ~]$ vi ~/etc/hadoop/mapred-site.xml
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
</property>
[hadoop@rdh ~]$ vi ~/etc/hadoop/yarn-site.xml
<property>
  <name>yarn.resourcemanager.hostname</name>
  <value>d1p.server.world</value>
</property>
<property>
  <name>yarn.nodemanager.hostname</name>
  <value>d1p.server.world</value>
</property>
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>
[hadoop@rdh ~]$ vi ~/etc/hadoop/slaves
d1p.server.world
node01.server.world
node02.server.world
=====格式化 NameNode、开启 Hadoop services=====
[hadoop@rdh ~]$ hdfs namenode -format
[hadoop@rdh ~]$ start-dfs.sh
[hadoop@rdh ~]$ start-yarn.sh
[hadoop@rdh ~]$ jps
=====测试=====
[hadoop@rdh ~]$ hdfs dfs -mkdir /test
[hadoop@rdh ~]$ hdfs dfs -copyFromLocal ~/NOTICE.txt /test
[hadoop@rdh ~]$ hdfs dfs -cat /test/NOTICE.txt
[hadoop@rdh ~]$ hadoop jar
~/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.1.jar wordcount
/test/NOTICE.txt /output01
[hadoop@rdh ~]$ hdfs dfs -ls /output01
[hadoop@rdh ~]$ hdfs dfs -cat /output01/part-r-00000
=====Web Manager=====
```

Access to "http://(server's hostname or IP address):50070/", then it's possible to see Hadoop cluster's summary.

Started:	Wed Jul 29 16:09:49 JST 2015
Version:	2.7.1, r15ecc87ccf4a0228f35af08fc56de536e6ce657a
Compiled:	2015-06-29T06:04Z by jenkins from (detached from 15ecc87)
Cluster ID:	CID-15b1101f-5301-4bd3-9c6b-bfee16bce46c
Block Pool ID:	BP-802493637-10.0.0.30-1438153777333

五十三、虚拟主机服务器 Nginx 详解

```
[root@rdh ~]# yum -y install epel*
[root@rdh ~]# yum --enablerepo=epel -y install nginx
[root@rdh ~]# vim /etc/nginx/nginx.conf
[root@rdh ~]# systemctl start nginx
[root@rdh ~]# systemctl enable nginx
```

Welcome to **nginx** on Fedora!

This page is used to test the proper operation of the **nginx** HTTP server after it has been installed. If you can read this page, it means that the web server installed at this site is working properly.

Website Administrator

This is the default `index.html` page that is distributed with **nginx** on Fedora. It is located in `/usr/share/nginx/html`.
You should now put your content in a location of your choice and edit the `root` configuration directive in the **nginx** configuration file `/etc/nginx/nginx.conf`.

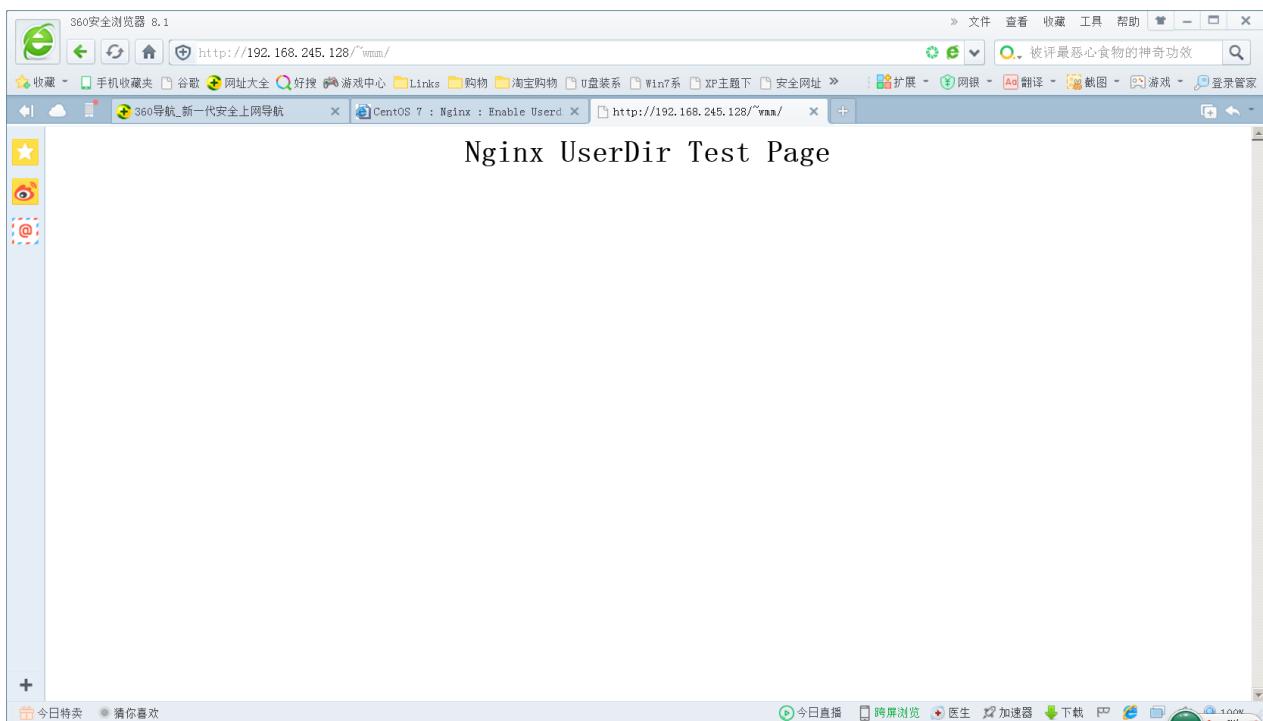
nginx Made for fedora

```
[root@rdh ~]# vim /etc/nginx/conf.d/virtualhost.conf
# create new
server {
    listen      8062;
```

```
server_name 192.168.245.128;

location / {
    root /usr/share/nginx/virtual.host;
    index index.html index.htm;
}
}

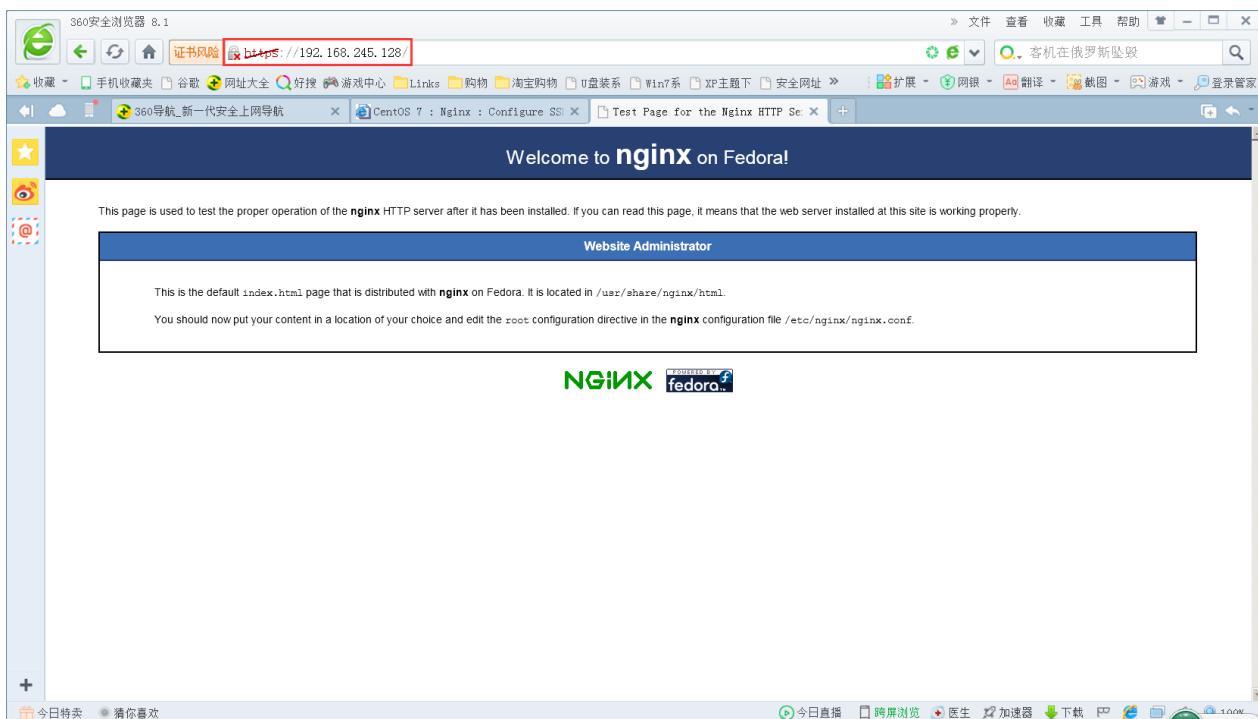
[root@rdh ~]# mkdir /usr/share/nginx/virtual.host
[root@rdh ~]# systemctl restart nginx
[root@rdh ~]# vim /usr/share/nginx/virtual.host/index.html
<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
Nginx Virtual Host Test Page
</div>
</body>
</html>
=====
虚拟主机=====
[root@rdh ~]# vim /etc/nginx/nginx.conf 【增加到 server 扩展项里面】
location ^~^(.+)($|.*$) {
    alias /home/$1/public_html$2;
    index index.html index.htm;
    autoindex on;
}
[wmm@rdh ~]$ chmod 755 /home/wmm/
[wmm@rdh ~]$ mkdir ~/public_html
[wmm@rdh ~]$ chmod 755 ~/public_html/
[wmm@rdh ~]$ vim ~/public_html/index.html
<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
Nginx UserDir Test Page
</div>
</body>
</html>
```



=====Ngnix+SSL=====

```
[root@rdh ~]# cd /etc/pki/tls/certs
[root@rdh certs]# make server.key
[root@rdh certs]# openssl rsa -in server.key -out server.key
[root@rdh certs]# make server.csr
Country Name (2 letter code) [XX]:CH
State or Province Name (full name) []:HN
Locality Name (eg, city) [Default City]:ZH
Organization Name (eg, company) [Default Company Ltd]:HUATECH
Organizational Unit Name (eg, section) []:IT
Common Name (eg, your name or your server's hostname) []:RDH.COM
Email Address []:151@163.com

[root@rdh certs]# openssl x509 -in server.csr -out server.crt -req -signkey
server.key -days 3650
[root@rdh ~]# vim /etc/nginx/nginx.conf
listen      443 ssl;
ssl_certificate      /etc/pki/tls/certs/server.crt;
ssl_certificate_key  /etc/pki/tls/certs/server.key;
[root@rdh ~]# systemctl restart nginx
```



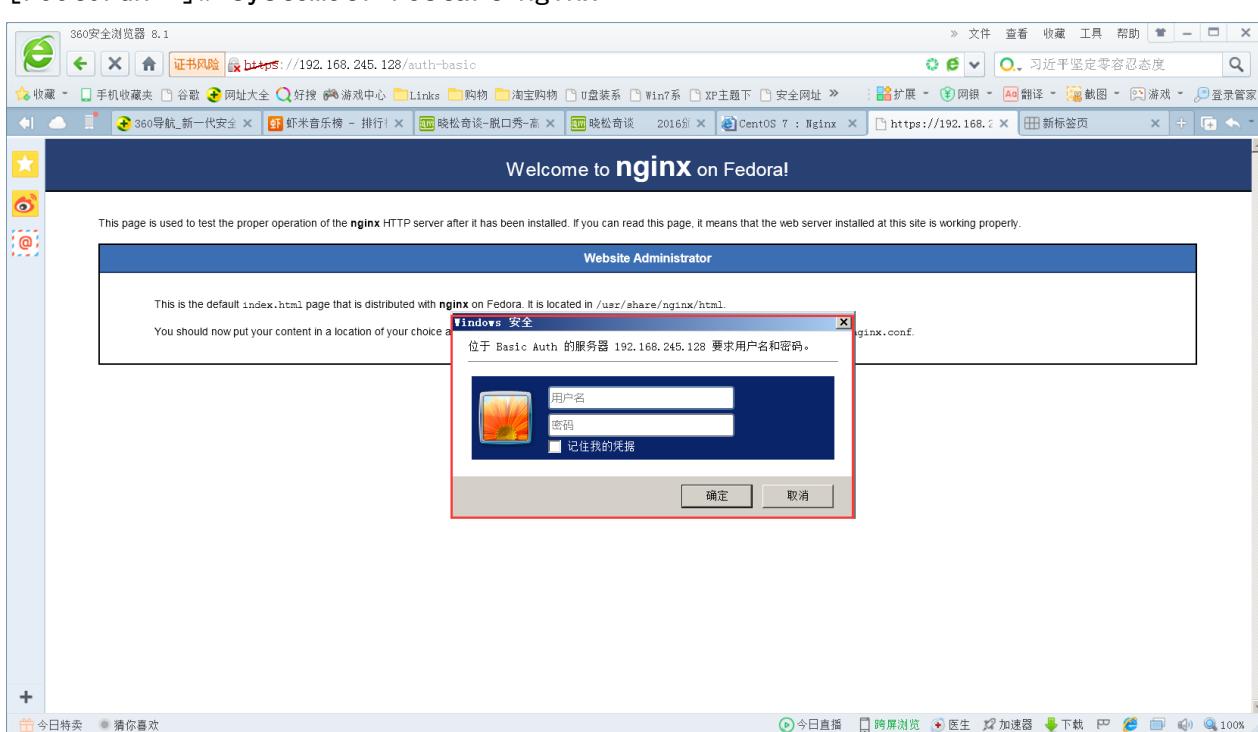
```
[root@rdh ~]# yum -y install httpd-tools
[root@rdh ~]# vim /etc/nginx/nginx.conf
location /auth-basic {
    auth_basic "Basic Auth";
    auth_basic_user_file "/etc/nginx/.htpasswd";
}
```

```
[root@rdh ~]# htpasswd -c /etc/nginx/.htpasswd wmm
```

New password:

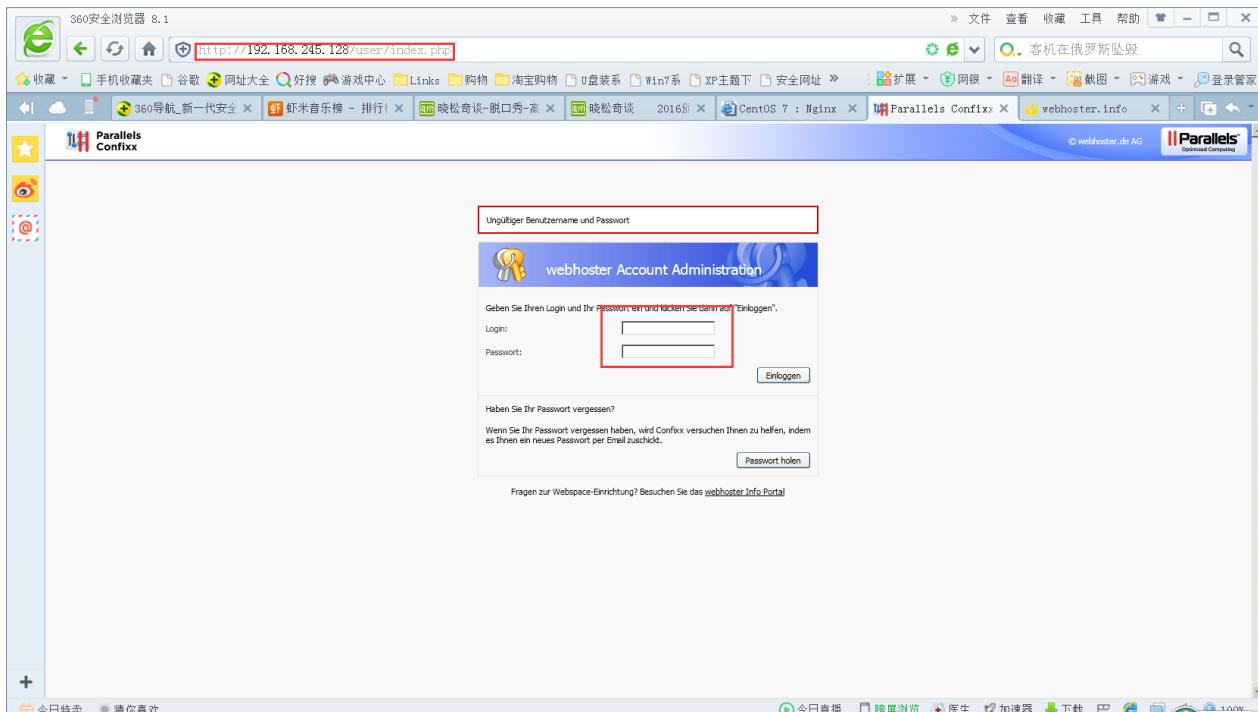
Re-type new password:

```
[root@rdh ~]# systemctl restart nginx
```



=====反向代理=====

```
[root@rdh ~]# vim /etc/nginx/nginx.conf
proxy_redirect          off;
proxy_set_header        X-Real-IP $remote_addr;
proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header        Host $http_host;
location / {
    proxy_pass http://dlp.server.world/;
}
[root@rdh ~]# systemctl restart nginx
```



【后端服务器配置】

```
[root@rdh ~]# vim /etc/httpd/conf/httpd.conf
LogFormat "%{X-Forwarded-For}i %l %u %t \"%r\" %>s %b \"%{Referer}i\""
 \"%{User-Agent}i\" combined
[root@rdh ~]# systemctl restart nginx
```

【多代理服务器】

```
[root@rdh ~]# vim /etc/nginx/nginx.conf
http {
    upstream backends {
        server node01.server.world:80 weight=3;
        server node02.server.world:80;
        server node03.server.world:80 backup;
    }
    server {
        listen      80 default_server;
        listen      [::]:80 default_server;
        server_name www.server.world;
```

```
proxy_redirect          off;
proxy_set_header        X-Real-IP $remote_addr;
proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header        Host $http_host;
location / {
    proxy_pass http://backends;
}
}

[root@rdh ~]# systemctl restart nginx
http {
    upstream backends {
        server node01.server.world:80 weight=3;
        server node02.server.world:80;
        server node03.server.world:80 backup;
    }
}

# change like follows in "server" section
server {
    listen      80 default_server;
    listen      [::]:80 default_server;
    server_name www.server.world;

    proxy_redirect          off;
    proxy_set_header        X-Real-IP $remote_addr;
    proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header        Host $http_host;

    location / {
        proxy_pass http://backends;
    }
}
=====Nginx+PHP=====
[root@rdh ~]# yum --enablerepo=epel -y install php php-mbstring php-pear php-fpm
[root@rdh ~]# vim /etc/php-fpm.d/www.conf
user = nginx
group = nginx
[root@rdh ~]# systemctl start php-fpm
[root@rdh ~]# systemctl enable php-fpm
[root@rdh ~]# vim /etc/nginx/nginx.conf
location ~ \.php$ {
    fastcgi_pass    127.0.0.1:9000;
    fastcgi_param  SCRIPT_FILENAME $document_root$fastcgi_script_name;
    fastcgi_param  PATH_INFO $fastcgi_path_info;
```

```

    include fastcgi_params;
}

[root@rdh ~]# echo "<?php phpinfo() ?>">/usr/share/nginx/html/info.php

```

System	
Build Date	Jun 23 2015 21:19:01
Server API	FFM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/dom.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d(mbstring.ini, /etc/php.d/phar.ini, /etc/php.d posix.ini, /etc/php.d/sysmsg.ini, /etc/php.d/sysvsem.ini, /etc/php.d/sysvshm.ini, /etc/php.d/wddx.ini, /etc/php.d/zip.ini)
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	API20100525.NTS
PHP Extension Build	API20100525.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled

五十四、LDAP 域控制器

```

[root@rdh ~]# yum -y install openldap-servers openldap-clients
[root@rdh ~]# cp /usr/share/openldap-servers/DB_CONFIG.example
/var/lib/ldap/DB_CONFIG
[root@rdh ~]# chown ldap. /var/lib/ldap/DB_CONFIG
[root@rdh ~]# systemctl start slapd
[root@rdh ~]# systemctl enable slapd
[root@rdh ~]# slappasswd
New password:
Re-enter new password:
{SSHA}3R0KNWJkvMVgVE91oxx/7cBrgNIkcX0i
[root@rdh ~]# vim chrootpw.ldif
dn: olcDatabase={0}config, cn=config
changetype: modify
add: olcRootPW
olcRootPW: {SSHA}3R0KNWJkvMVgVE91oxx/7cBrgNIkcX0i
[root@rdh ~]# ldapadd -Y EXTERNAL -H ldap:// -f chrootpw.ldif
[root@rdh ~]# ldapadd -Y EXTERNAL -H ldap:// -f /etc/openldap/schema/cosine.ldif
[root@rdh ~]# ldapadd -Y EXTERNAL -H ldap:// -f /etc/openldap/schema/nis.ldif
[root@rdh ~]# ldapadd -Y EXTERNAL -H ldap:// -f
/etc/openldap/schema/inetorgperson.ldif
[root@rdh ~]# slappasswd

```

New password:

Re-enter new password:

```
{SSHA}orgdGj+JqDldq4k01zBfrIG9jiWfqnuN
[root@rdh ~]# vim chdomain.ldif
# replace to your own domain name for "dc=***, dc=***" section
# specify the password generated above for "olcRootPW" section
dn: olcDatabase={1}monitor, cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by
dn. base="gidNumber=0+uidNumber=0, cn=peercred, cn=external, cn=auth"
      read by dn. base="cn=Manager, dc=rdh, dc=com" read by * none

dn: olcDatabase={2}hdb, cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=rdh, dc=com

dn: olcDatabase={2}hdb, cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=Manager, dc=rdh, dc=com

dn: olcDatabase={2}hdb, cn=config
changetype: modify
add: olcRootPW
olcRootPW: {SSHA}orgdGj+JqDldq4k01zBfrIG9jiWfqnuN

dn: olcDatabase={2}hdb, cn=config
changetype: modify
add: olcAccess
olcAccess: {0}to attrs=userPassword, shadowLastChange by
      dn="cn=Manager, dc=server, dc=world" write by anonymous auth by self write by *
      none
olcAccess: {1}to dn. base="" by * read
olcAccess: {2}to * by dn="cn=Manager, dc=rdh, dc=com" write by * read
[root@rdh ~]# ldapmodify -Y EXTERNAL -H ldapi:/// -f chdomain.ldif
[root@rdh ~]# vim basedomain.ldif
# replace to your own domain name for "dc=***, dc=***" section
dn: dc=rdh, dc=com
objectClass: top
objectClass: dcObject
objectclass: organization
o: Server World
```

dc: rdh

dn: cn=Manager, dc=rdh, dc=com
objectClass: organizationalRole
cn: Manager
description: Directory Manager

dn: ou=People, dc=rdh, dc=com
objectClass: organizationalUnit
ou: People

dn: ou=Group, dc=rdh, dc=com
objectClass: organizationalUnit
ou: Group
[root@rdh openldap]# rm -rf /etc/openldap/slapd.d/*
[root@rdh openldap]# chown -R ldap:ldap /etc/openldap/slapd.d
[root@rdh ~]# ldapadd -x -D cn=Manager, dc=rdh, dc=com -W -f basedomain.ldif
Enter LDAP Password:
adding new entry "dc=rdh, dc=com"

adding new entry "cn=Manager, dc=rdh, dc=com"

adding new entry "ou=People, dc=rdh, dc=com"

adding new entry "ou=Group, dc=rdh, dc=com"

【增加用户】

[root@rdh ~]# slappasswd
New password:
Re-enter new password:
{SSHA}g1kFZZJo/eyBB1810JcwtPPC1pSYHTe7
[root@rdh ~]# vim ldapuser.ldif
dn: uid=cent, ou=People, dc=rdh, dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
cn: Cent
sn: Linux
userPassword: {SSHA}g1kFZZJo/eyBB1810JcwtPPC1pSYHTe7
loginShell: /bin/bash
uidNumber: 1000
gidNumber: 1000
homeDirectory: /home/cent

dn: cn=cent, ou=Group, dc=rdh, dc=com

```
objectClass: posixGroup
cn: Cent
gidNumber: 1000
memberUid: cent
[root@rdh ~]# ldapadd -x -D cn=Manager,dc=rdh,dc=com -W -f ldapuser.ldif
Enter LDAP Password:
adding new entry "uid=cent, ou=People, dc=rdh, dc=com"

adding new entry "cn=cent, ou=Group, dc=rdh, dc=com"
```

【将本地用户作为账号的脚本】

```
[root@rdh ~]# vim ldapuser.sh
# extract local users and groups who have 1000-9999 digit UID
# replace "SUFFIX=***" to your own domain name
# this is an example
#!/bin/bash

SUFFIX='dc=server,dc=world'
LDIF='ldapuser.ldif'

echo -n > $LDIF
GROUP_IDS=()
grep "x:[1-9][0-9][0-9][0-9]::" /etc/passwd | (while read TARGET_USER
do
    USER_ID=$(echo "$TARGET_USER" | cut -d':' -f1)

    USER_NAME=$(echo "$TARGET_USER" | cut -d':' -f5 | cut -d' ' -f1,2)
    [ ! "$USER_NAME" ] && USER_NAME="$USER_ID"

    LDAP_SN=$(echo "$USER_NAME" | cut -d' ' -f2)

    LASTCHANGE_FLAG=$(grep "${USER_ID}:" /etc/shadow | cut -d':' -f3)
    [ ! "$LASTCHANGE_FLAG" ] && LASTCHANGE_FLAG="0"

    SHADOW_FLAG=$(grep "${USER_ID}:" /etc/shadow | cut -d':' -f9)
    [ ! "$SHADOW_FLAG" ] && SHADOW_FLAG="0"

    GROUP_ID=$(echo "$TARGET_USER" | cut -d':' -f4)
    [ ! "$echo ${GROUP_IDS[@]} | grep \"$GROUP_ID\" ] &&
    GROUP_IDS+=("$GROUP_ID")

    echo "dn: uid=$USER_ID,ou=People,$SUFFIX" >> $LDIF
    echo "objectClass: inetOrgPerson" >> $LDIF
    echo "objectClass: posixAccount" >> $LDIF
    echo "objectClass: shadowAccount" >> $LDIF
```

```
echo "sn: $LDAP_SN" >> $LDIF
echo "givenName: $(echo "$USER_NAME" | awk '{print $1}')" >> $LDIF
echo "cn: $USER_NAME" >> $LDIF
echo "displayName: $USER_NAME" >> $LDIF
echo "uidNumber: $(echo "$TARGET_USER" | cut -d':' -f3)" >> $LDIF
echo "gidNumber: $(echo "$TARGET_USER" | cut -d':' -f4)" >> $LDIF
echo "userPassword: {crypt}$(grep "${USER_ID}:" /etc/shadow | cut -d':' -f2)" >> $LDIF
echo "gecos: $USER_NAME" >> $LDIF
echo "loginShell: $(echo "$TARGET_USER" | cut -d':' -f7)" >> $LDIF
echo "homeDirectory: $(echo "$TARGET_USER" | cut -d':' -f6)" >> $LDIF
echo "shadowExpire: $(passwd -S "$USER_ID" | awk '{print $7}')" >> $LDIF
echo "shadowFlag: $SHADOW_FLAG" >> $LDIF
echo "shadowWarning: $(passwd -S "$USER_ID" | awk '{print $6}')" >> $LDIF
echo "shadowMin: $(passwd -S "$USER_ID" | awk '{print $4}')" >> $LDIF
echo "shadowMax: $(passwd -S "$USER_ID" | awk '{print $5}')" >> $LDIF
echo "shadowLastChange: $LASTCHANGE_FLAG" >> $LDIF
echo >> $LDIF
done

for TARGET_GROUP_ID in "${GROUP_IDS[@]}"
do
    LDAP_CN=$(grep ":${TARGET_GROUP_ID}:" /etc/group | cut -d':' -f1)

    echo "dn: cn=$LDAP_CN, ou=Group, $SUFFIX" >> $LDIF
    echo "objectClass: posixGroup" >> $LDIF
    echo "cn: $LDAP_CN" >> $LDIF
    echo "gidNumber: $TARGET_GROUP_ID" >> $LDIF

    for MEMBER_UID in $(grep ":${TARGET_GROUP_ID}:" /etc/passwd | cut -d':' -f1,3)
    do
        UID_NUM=$(echo "$MEMBER_UID" | cut -d':' -f2)
        [ $UID_NUM -ge 1000 -a $UID_NUM -le 9999 ] && echo "memberUid: $(echo "$MEMBER_UID" | cut -d':' -f1)" >> $LDIF
    done
    echo >> $LDIF
done
)
```

[root@rdh ~]# chmod +x ldapuser.sh

[root@rdh ~]# sh ldapuser.sh

[root@rdh ~]# ldapadd -x -D cn=Manager,dc=server,dc=world -W -f ldapuser.ldif

```
[root@rdh ~]# ldapadd -x -D cn=Manager, dc=rdh, dc=com -W -f ldapuser.ldif
Enter LDAP Password:
adding new entry "uid=wmm, ou=People, dc=rdh, dc=com"

adding new entry "uid=huatech, ou=People, dc=rdh, dc=com"

adding new entry "cn=wmm, ou=Group, dc=rdh, dc=com"

adding new entry "cn=huatech, ou=Group, dc=rdh, dc=com"
```

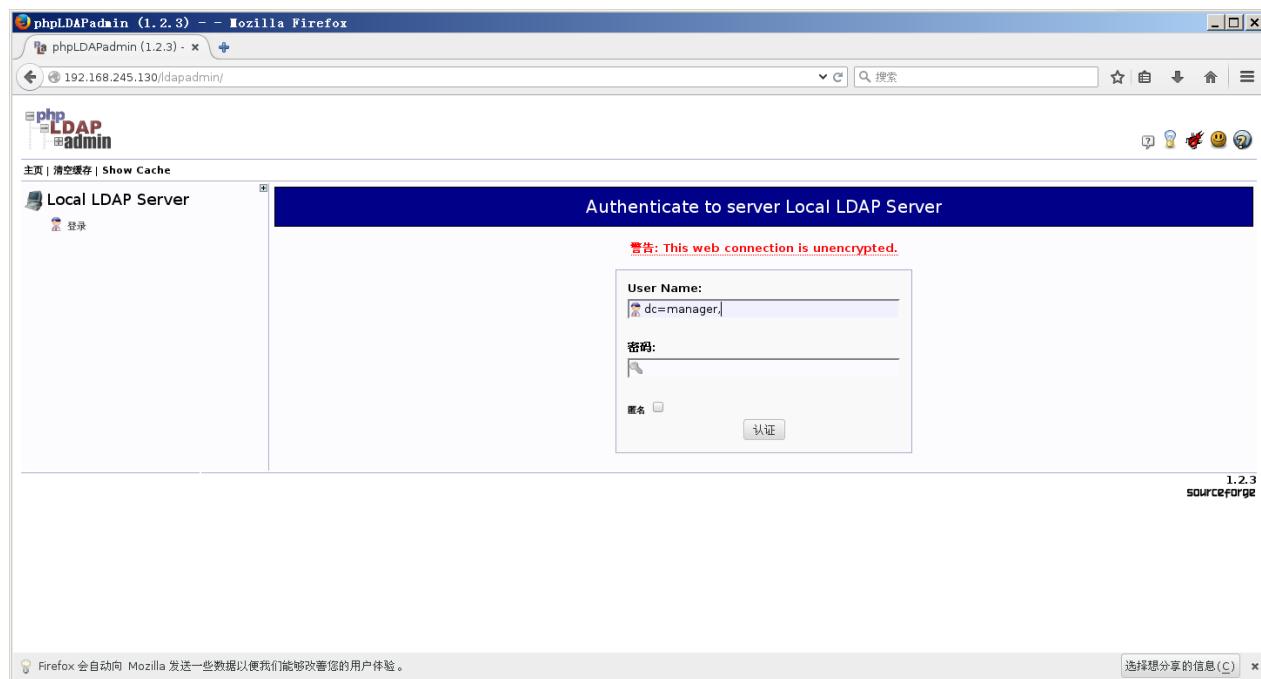
【删除】

```
[root@rdh ~]# ldapdelete -x -W -D 'cn=Manager, dc=server, dc=world'
"uid=cent, ou=People, dc=server, dc=world"
[root@rdh ~]# ldapdelete -x -W -D 'cn=Manager, dc=server, dc=world'
"cn=cent, ou=Group, dc=server, dc=world"
```

【配置客户端】

```
[root@server02 ~]# yum -y install openldap-clients nss-pam-ldapd
[root@server02 ~]# authconfig --enableldap --enableldapauth --ldapserver=rdh.com
--ldapbasedn="dc=rdh, dc=com" --enablemkhomedir --update
=====TLS=====
[root@rdh ~]# cd /etc/pki/tls/certs/
[root@rdh certs]# make server.key
[root@rdh certs]# openssl rsa -in server.key -out server.key
[root@rdh certs]# make server.crs
[root@rdh certs]# openssl x509 -in server.csr -out server.crt -req -signkey
server.key -days 3650
[root@rdh ~]# cp /etc/pki/tls/certs/server.key \
> /etc/pki/tls/certs/server.crt \
> /etc/pki/tls/certs/ca-bundle.crt \
> /etc/openldap/certs/
[root@rdh ~]# vim mod_ssl.ldif
# create new
dn: cn=config
changetype: modify
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/openldap/certs/ca-bundle.crt
-
replace: olcTLSCertificateFile
olcTLSCertificateFile: /etc/openldap/certs/server.crt
-
replace: olcTLSKeyFile
olcTLSKeyFile: /etc/openldap/certs/server.key
[root@rdh ~]# ldapmodify -Y EXTERNAL -H ldapi:/// -f mod_ssl.ldif
[root@rdh ~]# vim /etc/sysconfig/slapd
SLAPD_URLS="ldapi:/// ldap:/// ldaps:///"
```

```
[root@rdh ~]# systemctl restart slapd
[root@rdh ~]# echo "TLS_REQCERT allow" >> /etc/openldap/ldap.conf
[root@rdh ~]# echo "tls_reqcert allow" >> /etc/nslcd.conf
[root@rdh ~]# authconfig --enableldap --update
=====
[root@rdh ~]# yum -y install httpd
[root@rdh ~]# rm -f /etc/httpd/conf.d/welcome.conf
[root@rdh ~]# systemctl start httpd
[root@rdh ~]# systemctl enable httpd
[root@rdh ~]# yum -y install php php-mbstring php-pear
[root@rdh ~]# yum --enablerepo=epel -y install phpLDAPadmin
[root@rdh ~]# vim /etc/phpLDAPadmin/config.php
$servers->setValue('login', 'attr', 'dn');
[root@rdh ~]# vim /etc/httpd/conf.d/phpLDAPadmin.conf
Require local
Require ip 192.168.245.0/24
[root@rdh ~]# systemctl restart httpd
```



The screenshot shows the phpLDAPadmin interface in Mozilla Firefox. The title bar reads "phpLDAPadmin (1.2.3) - Mozilla Firefox". The address bar shows the URL "192.168.245.130/ldapadmin/cmd.php?server_id=1&redirect=true". The main content area displays the "Local LDAP Server" configuration for "cn=Manager". The top navigation bar includes "schema", "search", "刷新", "信息", "monitor", "export", and "退出". The left sidebar shows a tree structure of the LDAP schema: "dc=rdh, dc=com (3)" with "cn=Manager" selected, and "ou=Group (3)" and "ou=People (3)". The right panel shows the object details for "cn=Manager":

- cn**: Manager (必填的, rdn)
- description**: Directory Manager
- objectClass**: organizationalRole (结构化)

At the bottom, a message from Firefox states: "Firefox 会自动向 Mozilla 发送一些数据以便我们能够改善您的用户体验。" and a "选择想分享的信息(C)" button.