第 8 章　常用服务器配置与管理

本章能力目标

- 掌握 Linux 系统下常用服务器的功能与配置
- 掌握 Linux 系统下常用服务器的管理方法

本章要点内容

- Samba 服务器的安装、配置和管理
- NFS 服务器的安装、配置和管理
- Apache 服务器的安装、配置和管理
- VSFTP 服务器的安装、配置和管理
- DNS 服务器的安装、配置和管理
- DHCP 服务器的安装、配置和管理

8.1　Samba 服务器

8.1.1　Samba 概述

1. Samba 的作用

建立计算机网络的目的之一就是为了能够共享资源，如今接入网络的计算机大多数使用 Windows 操作系统。为了能让使用 Linux 操作系统的计算机和使用 Windows 操作系统的计算机共享资源，需要使用 Samba 工具。

Samba 是在 Linux/UNIX 系统上实现 SMB（Session Message Block）协议的一个免费软件，以实现文件共享和打印机服务共享，它的工作原理与 Windows 的网上邻居类似。

SMB 使 Linux 计算机在网上邻居中看起来如同一台 Windows 计算机。Windows 计算机的用户可以“登录”到 Linux 计算机中，从 Linux 中复制文件，提交打印任务。如果 Linux 运行环境中有较多的 Windows 用户，使用 SMB 将会非常方便。

如图 8-1 所示，图中的服务器运行 Samba 服务器软件，其操作系统是 Linux。该服务器通过 Samba 可以向局域网中的其他 Windows 主机提供文件共享的服务。同时，在 Linux 服务器上还连接了一个共享打印机，打印机也通过 Samba 向局域网的其他 Windows 用户提供打印服务。

2. Samba 的组成

给 Windows 客户提供文件服务是通过 Samba 实现的，这套软件由一系列的组件构成，主要的组件有：

（1）smbd（SMB 服务器）。smbd 是 Samba 服务守护进程，是 Samba 的核心，时刻侦听网络的文件和打印服务请求，负责建立对话进程、验证用户身份、提供对文件系统和打印机的

访问机制。该程序默认安装在/usr/sbin 目录下。

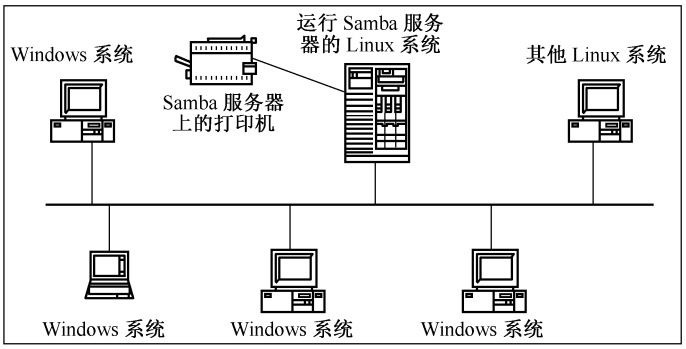

图 8-1　由 Samba 提供文件和打印共享

（2）nmbd（NetBIOS 名字服务器）。nmbd 也是 Samba 服务器的守护进程，用来实现“Network Browser”（网络浏览服务器）的功能，对外发布 Samba 服务器可以提供的服务。用户甚至可以用 Samba 作为局域网的主浏览服务器。

（3）smbclient（SMB 客户程序）。是 Samba 的客户端程序，客户端用户使用它可以复制 Samba 服务器上的文件，还可以访问 Samba 服务器上共享的打印机资源。

（4）testparm。该程序用来快速检查和测试 Samba 服务器配置文件 smb.conf 中的语法错误。

（5）smbtar。smbtar 是一个 Shell 脚本程序，它通过 smbclient 使用 tar 格式备份和恢复一台远程 Windows 的共享文件。

还有其他工具命令用来配置 Samba 的加密口令文件、配置用于 Samba 国际化的字符集。在 Linux 上，Samba 还提供了挂载和卸载 SMB 文件系统的工具程序 smbmount 和 smbumount。

8.1.2　Samba 服务器的安装

用户在安装 Red Hat Linux 9 的时候，如果选择了安装所有软件包，那么 Samba 就已经安装上了；如果系统没有安装，则可以从光盘的 Red Hat/RPMS 目录下安装。

1. 查询 Samba 是否已经安装

Red Hat Linux 9 中提供了 Samba 服务器的 RPM 软件安装包，这里可以使用 rpm 命令来检查是否安装或已安装。安装 Samba 服务器需要以下软件包：samba-2.2.7a-7.9.0.i386.rpm，Samba 服务器软件。samba-common-2.2.7a-7.9.0.i386.rpm，Samba 服务器与客户端都需要的文件。samba-client-2.2.7a-7.9.0.i386.rpm，Samba 客户端软件。

```
[root@rh9   root]# rpm   -qa |grep   samb   //检查 Samba 的相关软件是否已经安装。
samba-2.2.7a-7.9.0
samba-common-2.2.7a-7.9.0
samba-client-2.2.7a-7.9.0            //Samba 客户端软件。
```

2. 安装 Samba

如果输出如上所示的软件名称，则说明已经安装，否则可以使用下面的命令安装 Samba 服务器软件。

注意：要先安装 samba-common-2.2.7a-7.9.0 软件包，才能顺利完成另外 2 个软件包的安装。

```
[root@rh9 dhcp]# mount    /mnt/cdrom
[root@rh9 dhcp]# cd    /mnt/cdrom/Red Hat/RPMS
[root@rh9 root]# rpm    -ivh    samba-common-2.2.7a-7.9.0.i386.rpm
warning: samba-common-2.2.7a-7.9.0.i386.rpm: V3 DSA signature:
NOKEY, key ID db42a60e
Preparing...                ########################################### [100%]
    1:samba-common      ########################################### [100%]
[root@rh9 root]#
[root@rh9 root]# rpm    -ivh    samba-2.2.7a-7.9.0.i386.rpm
warning: samba-2.2.7a-7.9.0.i386.rpm: V3 DSA signature: NOKEY,
key ID db42a60e
Preparing...                ########################################### [100%]
    1:samba                 ########################################### [100%]
[root@rh9 root]# rpm    -ivh    samba/samba-client-2.2.7a-7.9.0.i386.rpm
warning: samba-client-2.2.7a-7.9.0.i386.rpm: V3 DSA signature:
NOKEY, key ID db42a60e
Preparing...                ########################################### [100%]
    1:samba-client          ########################################### [100%]
```

安装了 Samba 的上述公用软件包、服务器软件包和客户端软件包后就可以了，但为了配置方便以及利用 Red Hat Linux 9 的新特性，建议再安装 redhat-config-samba-1.0.4-1 和 samba-swat-2.2.7a-7.9.0 两个软件包。这两个软件包在 Red Hat Linux 9 安装光盘里都有，其中 redhat-config-samba-1.0.4-1 在第 1 张光盘里，samba-swat-2.2.7a-7.9.0 在第 2 张光盘里，安装方法和上面的相同。redhat-config-samba-1.0.4-1 是 Samba 配置工具，使用它可以很方便地配置 Samba。samba-swat-2.2.7a-7.9.0 是用来修改 Samba 配置文件的。

8.1.3 Samba 服务器的启动/停止

安装并配置好 Samba 后，可以在 Linux 终端将 Samba 启动，也可通过终端命令行将已经启动的 Samba 服务器关闭。若要启动 Samba，必须以管理员身份登录 Linux，如果是以普通用户身份登录 Linux，可以在终端使用命令“su -”暂时切换到系统管理员身份。

Samba 服务器的启动、停止，以及当前所处状态的查询等操作，都可以通过 service 命令来实现。

```
[root@rh9   root]# service   smb
用法：/etc/init.d/smb {start|stop|restart|reload|status|condrestart}
[root@rh9   root]# service   smb   start
启动 SMB 服务：                                          [  确定  ]
启动 NMB 服务：                                          [  确定  ]
[root@rh9   root]# service   smb   stop
关闭 SMB 服务：                                          [  确定  ]
关闭 NMB 服务：                                          [  确定  ]
[root@rh9   root]# service   smb   status
smbd 已停
nmbd 已停
```

2. 使用 chkconfig 命令

若要系统每次启动时自动开启 Samba 服务，可以使用 chkonfig 命令，下面的例子表示在

系统进入第 3 和第 5 个级别时自动开启 Samba 服务。

```
[root@rh9   root]# chkconfig
chkconfig 版本 1.3.8 - 版权 （C） 1997-2000 Red Hat, Inc.
在 GNU 公共许可的条款下，本软件可以被自由发行。
用法：      chkconfig --list [name]
            chkconfig --add <name>
            chkconfig --del <name>
            chkconfig [--level <levels>] <name> <on|off|reset>）
[root@rh9   root]# chkconfig   --level   35   smb   on
[root@rh9   root]# chkconfig   --list   smb
smb         0:关闭   1:关闭   2:关闭   3:启用   4:关闭   5:启用   6:关闭
```

3. 使用 ntsysv 命令

也可以使用命令 ntsysv 打开图形化的命令行界面来设置，如图 8-2 所示。使用“Tab”键可以在“服务”、“确定”和“取消”之间切换，在“服务”窗口中使用方向键“↓”和“↑”可以将光标移动到想要设置的服务，然后使用“空格键”设置或者取消需要自动启动的服务（前面有“*”标志的服务将在每次开机时自动启动）。另外，按照界面下方的提示按“F1”键，可以获得有关某个服务的详细说明。

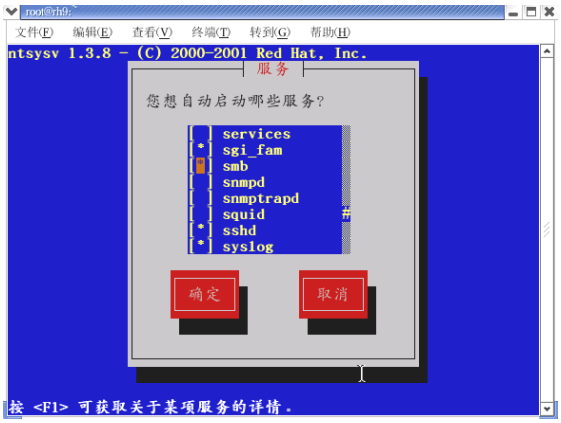

图 8-2　设置系统服务(1)

如果是在图形界面下，除了使用上面介绍的方法外，还可依次单击“主菜单”→“系统设置”→“服务器设置”→“服务”选项，打开图 8-3 所示的界面，在该图形界面下用户也可以很方便地设置选中的服务。

8.1.4　Samba 服务器的配置文件

Samba 服务器最主要的配置文件是/etc/smb/smb.conf，该文件中以“;”或“#”符号开头的都为注释，该行的内容会被忽略而不生效。文件中以“#”开头的行是指说明，而以“;”开头的行则是表示目前该项停用，但可以根据今后的需要去掉前面的“;”使之生效。配置文件中的每一行都是以“设置项目＝设置值”的格式来表示。

配置文件 smb.conf 主要是由两个部分所组成：Global settings 和 Share Definition。前者是与 Samba 整体环境有关的选项，这里的设置项目适用于每个共享的目录；后者是针对不同的共享目录的个别设置。在开始修改配置文件之前，必须先了解下列重点内容。

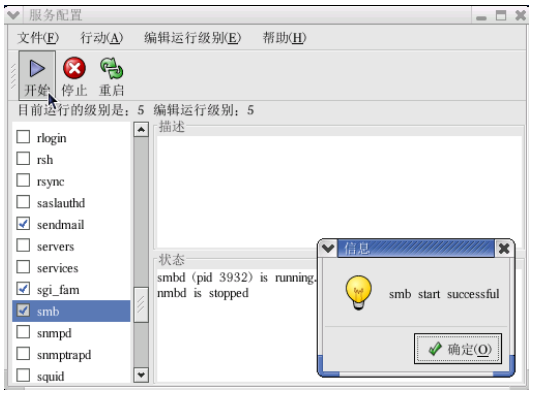

图 8-3　设置系统服务(2)

1. Global settings

本部分参数主要有基本设置参数、安全设置参数、网络设置参数、文件设置参数、打印机设置参数、用户权限设置参数和日志设置参数等。

（1）workgroup = MYGROUP

此项用来设置在 Windows 操作系统的“网上邻居”中将会出现的 Samba 服务器所属的群组，默认为 MYGROUP，不区分大小写。

（2）server string = Samba Server

此项用来设置 Samba 服务器的文字说明，以方便客户端的识别，默认为“Samba Server”。

（3）hosts allow = 192.168.1.　192.168.2.　127.

该项用来设置哪些主机允许访问 Samba 服务器，默认为全部。如果设置的项目超过一个，必须以逗号、空格或制表符来分隔开。“hosts allow = 192.168.1.　192.168.2.　127.”表示允许来自 192.168.1.*、192.168.2.*和 127.*.*.*的主机连接。

另外，也可以采用其他的一些表示方法，如：“hosts allow =192.168.1. except　192.168.1.5”表示允许来自 192.168.1.*的所有主机连接，但排除了 192.168.1.5。“hosts allow = 192.168.1.0/255.255.255.0”表示允许来自 192.168.1.0 子网的所有主机连接。“hosts allow = host1,host2”表示允许名字是 host1 和 host2 的主机连接。“hosts allow = @cqcet.cn”表示允许来自 cqcet.cn 网域的所有主机连接。

（4）printcap name = /etc/printcap

此项是用来设置开机时自动加载的打印机配置文件的名称和路径。

（5）load printers = yes

表示是否允许打印机在开机时自动加载到浏览列表（Browsing List），以支持客户端的浏览功能，即是否共享打印机。

（6）printing = cups

此项用来指定打印系统的类型，在一般情况下并不需要修改此项。目前支持的打印系统有 bsd、sysv、plp、lprng、aix、hpux、qnx 和 cups。

（7）guest account = pcguest

pcguest 为用户名，可更改去掉前边的“;”让用户以 pcguest 身份匿名登录，但要保证 /etc/passwd 中有此账号。

（8）log file = /var/log/samba/%m.log

此项为所有连接到 Samba 服务器的计算机建立独立的记录日志。其默认保存的位置是/var/log/samba/。

（9）max log size = 0

此项设置每个记录日志的大小上限，单位是 KB。默认值为 0，表示没有大小的限制。Samba 服务器会定期检查其上限，如果超过此设置值，就会重新命名此文件，并加上“.old” 扩展名。

（10）security = user

此项指定 Samba 服务器使用的安全等级为 user 级。此处可以用的安全等级有 share、user、server 和 domain 共 4 类。它们的区别，请参看本书稍后的内容。

（11）password server = <NT-Server-Name>

此项默认不使用，而且只有在上个选项设置为“security = server”时才生效。它用来指定密码验证服务器的名称，此处必须使用 NetBIOS 名称，默认是网络中的域控制器。也可以使用“password server=*”的方式自动寻找网络中可用的域控制器。

（12）password level = 8

这个选项是为了避免 Samba 服务器和客户端之间允许的密码最大位数不同，从而产生的错误。

（13）username level = 8

这个选项是为了避免服务器和客户端之间允许账号最大位数不同而产生的错误。

（14）encrypt passwords = yes

此项表示是否指定用户密码以加密的形式发送到 Samba 服务器。由于目前 Windows 操作系统都已经使用加密的方式来发送密码，因此建议启用该选项。否则，要改变 Windows 注册表。

（15）smb passwd file = /etc/samba/smbpasswd

该选项用来指定 Samba 服务器使用的密码文件路径。默认情况下，该文件并不存在，需要用户自己建立。

（16）ssl CA certFile = /usr/share/ssl/certs/ca-bundle.crt

该选项用来指定包含所有受信任 CA 名称的文件。当需要配置 Samba 服务器支持 SSL 时，必须读取这个文件的内容。在默认情况下，该选项未被启用。

（17）unix password sync = yes

当 Samba 服务器密码文件 smbpasswd 中的加密密码内容修改时，用这个选项将 Samba 和 UNIX 中的密码进行同步。在默认情况下使用该功能，在运行密码同步之后，旧的 UNIX 密码将不能再用于登录系统。

（18）passwd program = /usr/bin/passwd %u

此选项默认时启用，用来指定设置 UNIX 密码的程序，默认值是“/usr/bin/passwd %u”，其中“%u”表示用户的名称。

（19）passwd chat = *New*password* %n\n *Retype*new*password* %n\n *passwd:*all *authentication*tokens*updated*successfully*

设置用户把 Linux 密码转换为 Samba 服务器密码时，屏幕出现的指示字符串，以及与用户产生的交互窗口。如果使用默认值，则屏幕上显示如下字符串。

```
New password:
```

```
Retype new password:
passwd: all authentication tokens successfully.
```

（20）pam password change = yes

此项表示可以使用 PAM 来修改 Samba 客户端的密码。PAM 可以允许管理员设置多种验证用户身份的方式，而不需要重新编译用于验证的程序。

（21）username map = /etc/samba/smbusers

此项指定一个配置文件，在此文件中包含客户端与服务器端上用户的对应数据。可以将同一个 UNIX 账号名对应到多个 Samba 账号，账号之间用空格间隔。如果每个客户端用户在 Samba 服务器上都拥有单独的 Samba 账号，则该项不需设置。默认情况下，该选项未启用。

（22）include = /etc/samba/smb.conf.%m

允许 Samba 服务器使用其他的配置文件，可以方便管理员事先为不同的主机设计合适的配置文件。默认情况下，该选项未启用。

（23）obey pam restrictions = yes

该选项用来设定是否采用 PAM 账号及会话管理。默认情况下，该选项未启用。

（24）socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192

此选项在编写 TCP/IP 程序时相当重要，可以借此调整 Samba 服务器运行时的效率。可以使用“man setsockopt”命令来得到详细的内容。

（25）interfaces = 192.168.12.2/24 192.168.13.2/24

该选项可以使 Samba 服务器监视多个网络接口。在设置时，等号右边可以使用 IP 地址、网络接口名、或者是 IP/子网掩码的组合。

（26）remote announce = 192.168.1.255 192.168.2.44

此项允许 NMDB（NetBIOS 域名服务器）定期公布 Samba 服务器的 IP 地址和群组名称到远程的网络或主机。默认情况下，该选项未启用。

（27）local master = no

此选项表示是否允许 nmbd 担任 Local Master 浏览器的角色，在默认的配置下并不使用此功能。如果设置值为“no”，则 nmbd 将不会成为子网中的 Local Master 浏览器，但是如果设置值为“yes”，也不是表示 nmbd 一定会是 Local Master 浏览器，而是指 nmbd 将会参加 Local Master 浏览器的选举。

（28）os level = 33

此选项的设置值是用来决定 Local Master 浏览器选举时的优先次序，数值越高表示优先次序越高，一般来说，Samba 服务器都具有很高的优先权，但是在默认的配置下并不使用此功能。

（29）domain master = yes

使用这个选项后，表示这台 Samba 服务器可担任网络中的 Domain Master Browser，它便可以集中来自所有子网的浏览列表。但如果网络中已有域控制器在担任此工作，则不可使用这个选项，以避免发生错误，在默认的配置下并不使用此功能。

（30）preferred master = yes

使用这个选项后，Preferred Master 可以在 Samba 服务器启动时，强制进行 Local Browser 选择，同时 Samba 也会享有较高的优先级，但在默认配置下并不使用此功能。

（31）domain logons = yes

这个选项可以决定是否将 Samba 服务器当成 Windows 95 工作站登录时的账号验证主机，默认并不使用此功能。

（32）logon script = %m.bat

这个选项可以设置主机登录时自动运行的批处理文件。这个文件需符合 DOS 兼容的换行格式，同时也需要先使用“domain logons”选项才可生效，但在默认的配置下并不使用此功能。

（33）logon script = %U.bat

这个选项可以设置用户登录时，自动运行的批处理文件。这个文件需符合 DOS 兼容的换行格式，同时也需先使用“domain logons”选项才可生效，但在默认的配置下并不使用此功能。

（34）logon path = \\%L\Profiles\%U

这个选项可以设置用户在登录 Samba 服务器时所使用的个人配置文件（Profile）的位置。默认值中的“%L”表示这台服务器的 NetBIOS 名称，而“%U”是指用户名称，但在默认的配置下并不使用此功能。

（35）wins support = yes

这个选项可用来决定是否将这台 Samba 服务器当成 WINS 服务器，除非环境中包含多个子网，否则不建议使用此项目。另外，在同一个网络中最多可以使用一台 WINS 服务器，而默认并不使用此功能。

（36）wins server = w.x.y.z

这个选项可用来设置 WINS 服务器的 IP 地址，而这台 WINS 服务器必须已在 DNS 服务器中登记，在默认的配置下并不使用此功能。注意一点，Samba 服务器可作为 WINS 服务器或 WINS 客户端，但不可同时担任这两种角色。

（37）wins proxy = yes

这个选项可用来决定是否将此 Samba 服务器当成 WINS 代理，在默认的配置下并不使用此功能。WINS 代理是指代替非 WINS 客户端向 WINS 服务器请求域名解析查询的计算机，因此在每个网段中至少要有一台 WINS 服务器。

（38）dns proxy = no

该选项可用来决定是否将此 Samba 服务器当成 DNS 代理，默认的配置下并不使用此功能。DNS 代理如果发现尚未注册的 NetBIOS 名称，可以决定是否将由 DNS 得到的名称当成 NetBIOS 名称。

（39）preserve case = no

该选项可以用来决定新建文件时，文件名称大小写是否与用户输入相同，或自己指定，在默认的配置下并不使用此功能。

（40）short preserve case = no

该选项可用来决定新建文件名符合 DOS 8.3 格式的文件时，文件名是否都用大写，或自己指定，在默认的配置下并不使用此功能。

（41）default case = lower

该选项可用来决定新建文件时文件名称的大小写，在默认的配置下并不使用此功能。

（42）case sensitive = no

该选项可以决定是否将大小写的文件名称视为不同，在默认的配置下并不使用此功能。但如果在 Samba 中，要以中文名称来为资源命名，则此处设置的值必须是“no”。

2. Share Definitions

以下包含许多以中括号（[]）开头的区域，而每个区域各代表一个共享资源，也就是在 Windows 客户端上启动“网上邻居”时，会出现的共享文件夹，以下将以配置文件中默认的内容来说明配置选项的功能。

（1）[home]

当用户请求一个共享时，服务器将在存在的共享资源段中去寻找，如果找到匹配的共享资源段，就使用这个共享资源段。如果找不到，就将请求的共享名看成是用户的用户名，并在本地的 password 文件里找这个用户，如果用户名存在且用户提供的密码是正确的，则以这个 home 段克隆出一个共享提供给用户。这个新的共享的名称是用户的用户名，而不是 home，如果 home 段里没有指定共享路径，就把该用户的主目录（home directory）作为共享路径。

通常的共享资源段能指定的参数基本上都可以指定给[home]段。但一般情况下[home]段有如下配置就可以满足普通的应用。

```
comment = Home Directories      //共享目录的文字描述。
browseable = no                 //不允许浏览主目录，即该目录内容只对有权限的用户可见。
writable = yes                  //允许用户写入目录。
valid users = %S                //允许访问该目录的用户，%S 表示当前登录的用户。
create mode = 0664              //新建文件的缺省许可权限。
directory mode = 0775           //新建目录的默认权限。
map to guest = bad user
```

当用户输入不正确的账号和密码时，可以利用“map to guest”选项来设置处理的方式，但在使用此选项前，必须将“security”选项设为“user”、“server”或“domain”。可用的设置值如表 8.1 所示。

表 8.1　map to guest 的可选值

设置值	说明
never	拒绝访问，该项最安全
bad user	如果输入的用户名正确，但密码错误，允许以 guest 身份访问
bad password	如果输入的用户名和密码都错误，仍然允许以 guest 身份访问

注意，如果在[home]段里加了“guess access = ok”，所有的用户都可以不要密码就能访问所有的主目录。

（2）[netlogon]

```
comment = Network Logon Service     //共享目录的文字描述。
path = /usr/local/samba/lib/netlogon //共享目录的本机路径。
guest ok = yes                      //连接时不需要输入密码。
writable = no                       //不允许写入共享目录。
share modes = no                    //是否允许目录中的文件在不同的用户之间共享。
```

（3）[Profiles]

```
path = /usr/local/samba/profiles    //共享目录的本机路径。
browseable = no                     //是否允许浏览目录。
guest ok = yes                      //连接时是否需要密码。
```

（4）[printers]

该段用于提供打印服务。如果定义了[printers]这个段，用户就可以连接在/etc/printcap 文件里指定的打印机。

当一个连接请求到来时，smbd 去查看配置文件 smb.conf 里已有的段，如果和请求匹配就用那个段，如果找不到匹配的段，但[home]段存在，就用[home]段。否则请求的共享名就当作是个打印机共享名，然后去寻找适合的 printcap 文件，看看请求的共享名是不是有效的打印共享名，如果是就克隆出一个新的打印机共享提供给客户。

```
comment = All Printers          //共享打印机的文字描述。
path = /var/spool/samba         //假脱机目录所在的位置。
browseable = no                 //不允许浏览与打印服务相关的假脱机目录。
public = yes                    //所有用户可以访问共享资源。
guest ok = no                   //连接共享资源时不需要输入密码。
writable = no                   //不允许写入与打印服务相关的假脱机目录。
printable = yes                 //实现打印共享。
```

注意：实现打印共享的配置段中，必须是“printable = yes”，如果指定为其他，则服务器将拒绝加载配置文件。通常，公共打印机的打印队列路径应该是任何人都有写入的权限。

另外，public=yes or no 都不是针对限定用户的，而是针对未限定的用户。设置成 yes 就是所有的用户都能够访问，no 就是仅限于限定的用户能够访问。

（5）[tmp]

```
comment = Temporary file space  //共享资源的文字描述。
path = /tmp                     //共享资源的本机路径。
read only = no                  //不只是允许读取。
public = yes                    //所有用户都可以访问共享资源。
```

（6）[public]

```
comment = Public Stuff          //共享资源的文字描述。
path = /home/samba              //共享资源的本机路径。
public = yes                    //所有用户都可以访问共享资源。
writable = yes                  //目录允许写入。
printable = no                  //不允许打印共享。
write list = @staff             //拥有写入权限的用户或群组（以@开头表示）。
```

另外，要让 Samba 服务器使用主机名能够正确的访问到相关的其他主机，如提供客户端身份验证的另一台服务器，还必须修改/etc/samba/lmhost 文件。该配置文件的唯一功能是提供主机名与 IP 地址的对应关系，应该将网络中所有和 Samba 服务器有关的主机名与 IP 地址的对应关系都记录到里面，每条记录占用一行。默认情况下，该文件的内容只有一条记录“127.0.0.1 localhost”。

8.1.5　Samba 服务器的安全等级

smb.conf 配置文件的“security”选项，可以设置 Samba 服务器的安全性等级，直接影响客户端访问服务器的方式，是配置中最重要的项目之一。在 Samba 服务器中，共分为 4 种安全等级，分别如下：

1. share 安全等级

当客户端连接到具有 Share 安全等级的 Samba 服务器时，不需要输入账号和密码等数据，就可访问主机上的共享资源，这种方式是最方便的连接方式，但是却无法保障数据的安全性。

其实在此安全等级中，用户并非不需要任何的账号和密码就可登录，而是 smbd 会自动提供一个有效的 UNIX 账号来代表客户端身份，这个原理就和 Web 服务器及 FTP 服务器上的“匿名”（anonymous）访问相同。为了提供客户端有效的 UNIX 账号，smbd 会自动决定最适合客户端的账号，并将这些可用的账号列成表，来满足不同用户的需求。

2. user 安全等级

在 Samba 2.2 中默认的安全等级是“user”，它表示用户在访问服务器的资源前，必须先用有效的 Samba 账号和密码进行登录，如图 8-4 所示。在服务器尚未成功验证客户端的身份前，可用的资源名称列表并不会发送到客户端上。在此模式中，通常使用加密的密码来提高验证数据传送的安全性。

图 8-4　Samba 服务器登录

应该特别注意的是，Samba 服务器与 Linux 操作系统使用不同的密码文件，所以无法以 Linux 操作系统上的账号密码登录 Samba 服务器。因此应该自己建立原来在“smb passwd file”选项中指定的/etc/samba/smbpasswd 文件。

建立 Samba 密码文件并不需要手动的输入数据，只要先以管理员的账号登录 Linux 系统，然后利用名为“mksmbpasswd.sh”的 Script 程序来读取 Linux 操作系统使用的密码文件（/ete/passwd），最后再转换成 Samba 密码文件即可。建立 Samba 密码文件的方法如下。

```
[root@rh9 root]# cat /etc/passwd   | mksmbpasswd.sh > /etc/samba/smbpasswd
[root@rh9 root]# ls   -l   /etc/samba/smbpasswd
-rw-r--r--      1 root        root              3978   7 月  30 00:11 /etc/samba/smbpasswd
```

在 Samba 密码文件建立后，接下来的工作就是利用 smbpasswd 命令来设置 Samba 密码文件中每个 Samba 用户对应的密码。因此，该命令可以将已经存在的 Linux 系统登录账号转变为 Samba 用户账号，这里以用户 root 为例。

```
[root@rh9 root]# smbpasswd   root          //生成 root 的 Samba 服务器登录密码。
New SMB password:
Retype new SMB password:
Password changed for user root.
Password changed for user root.
```

如果是添加新的用户，这时必须首先确保要添加的用户名在/etc/passwd 文件中存在，否则将有“Failed to find entry for user xxxx”的提示信息出现。因此，要先使用 useradd 命令添加该账号为 Linux 系统登录账号，然后再用 smbpasswd 命令将其设置为 Samba 账号。smbpasswd 常用的命令格式是：“smbpasswd　[选项]　[用户名]”。其中，常用的选项及其含义如表 8.2 所

示。从表 8.2 可知，该命令还可以对 Samba 账号进行管理和维护。

表 8.2　smbpasswd 的主要选项及作用

选项	作用
-a	添加 Samba 用户账号
-x	删除 Samba 用户账号
-d	关闭、停用 Samba 用户账号
-e	开放 Samba 用户账户
-h	显示该命令的帮助

```
[root@rh9 root]# smbpasswd      lihh                    //不存在的 Linux 系统登录账号 lihh。
New SMB password:
Retype new SMB password:
build_sam_account: smbpasswd database is corrupt!  username lihh not in unix passwd database!
//用户名 lihh 在密码文件/etc/passwd 中不存在。
Failed to find entry for user lihh.
Failed to modify password entry for user lihh        //设置 lihh 的 Samba 登录密码失败。
[root@rh9 root]# useradd   lihh                      //创建 Linux 系统登录账号。
[root@rh9 root]# smbpasswd   -a   lihh
New SMB password:
Retype new SMB password:
Added user lihh.
[root@rh9 root]# smbpasswd   -d   lihh
Disabled user lihh.
[root@rh9 root]# smbpasswd   -e   lihh
Enabled user lihh.
[root@rh9 root]# smbpasswd   -x   lihh
Deleted user lihh.
```

3. server 安全等级

如果连接到 Server 安全等级，用户在访问服务器资源前，同样也需先用有效的账号和密码进行登录，但是客户端身份的验证会由另一台服务器负责。因此，在设置 server 安全等级时，必须同时指定“password server”选项。如果验证失败，服务器会自动将安全性等级降为“user”，但如果使用加密密码，那 Samba 服务器将无法反向检查原有的 UNIX 密码文件，所以必须指定另一个有效的 smbpasswd 文件来进行客户端的身份验证。因此，应该设置 smb.conf 文件中的以下选项。

```
security = server                          //设置 Samba 服务器使用 server 安全等级。
password server = <NT-Server-Name>         //设置 Samba 服务器的 NetBIOS 计算机名。
smb passwd file = /etc/samba/smbpasswd     //SMB 服务器使用的密码文件路径。
```

4. domain 安全等级

如果目前的网络结构为网域（Domain）而不是工作组（Workgroup），这时可以使用 domain 安全等级，以将 Samba 服务器加入现有的网域中。也就是说，不担任账号与密码的验证工作，而是由网络中的域控制器（Domain Controller , DC）统一处理。

要将 Samba 服务器加入现有的网域，可以使用以下的指令格式。

```
[root@rh9 root]#smbpasswd   -j   Samba 主机名   -r   DC
```

在运行以上的指令后，还需要修改 smb.conf 文件中[global]部分的以下配置选项。

```
workgroup = domain_name          //指定 Samba 服务器要加入的网域。
security = domain                //设置 Samba 服务器使用的安全性等级为 domain。
password server = DC             //指定进行身份验证的网域控制器名。
```

8.1.6 Samba 服务器的配置

下面以实例的方法，说明 Samba 服务器的配置方法。在某局域网中，当前 Linux 主机的 NetBIOS 名称为 rh9，主机所在的工作组为 Workgroup，现在想在该 Linux 主机上创建目录/var/work，并使用 Samba 服务器进行共享，使得客户机上所有用户都可以通过 Samba 服务匿名访问该目录，无需输入任何账号与口令，并且对该目录拥有可读可写的权限。配置过程如下。

1. 创建目录共享目录

执行命令“rmdir -m 777 /var/work”，创建目录/mnt/work 并设置其权限对所有用户都是可读、可写、可执行。

2. 编辑配置文件

使用文本编辑程序（如 VI）修改配置文件/etc/samba/smb.conf，确保里面的相关设置设定为如下内容，其他各项使用默认设定值即可。

```
[global]                                    //设置全局配置。
workgroup = MYGROUP                         //设定工作组名称。
server string = Samba Server                //对该主机的注释。
security = share                            //必须设定为 share 级，否则无法匿名登录。
netbios name = rh9                          //设定在网络中的主机名
[work]                                      //设置共享目录。
comment = A publicly accessible directory !     //共享目录的注释。
path = /var/work                                //共享资源的路径。
writable = yes                                  //允许写入目录。
guest ok = yes                                  //连接时不需要密码。
public = yes                                    //所有未指定用户都可以访问。
```

3. 语法查错

使用 Samba 安装后包含的工具——testparm，来测试 smb.conf 配置文件内的语法是否正确。如果设置时的语法都正确，那在运行 testparm 程序后，系统会出现以下画面。

```
[root@rh9 root]# testparm
Load smb config files from /etc/samba/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[work]"
Loaded services file OK.
Press enter to see a dump of your service definitions
```

在出现以上信息后，如果想查看详细的 smb.conf 配置文件内容，可以按 Enter 键，系统就会出现所有的选项设置。

测试的结果正常，也不保证 Samba 服务器就一定可以正常运行，因为这个程序仅针对语法来进行测试。

4. 重新启动 Samba 服务器

执行“service smb restart”命令重新启动 Samba 服务器进程，使得修改后的配置文件生效。

5. 访问 Samba 服务器

在 Windows 客户端上，双击“我的电脑”打开文件浏览器，然后在地址栏中输入“\\Samba 服务器名称或 IP”，按 Enter 键后在窗口中将看到 Samba 服务器上的共享资源，如图 8-5 所示。双击共享目录 work 的图标，不用输入用户名和密码即可进入共享目录，进行权限许可范围内的各种操作。

在 Linux 客户端上，双击用户主目录图标，打开文件浏览器，然后在地址栏中输入“smb://Samba 服务器名称或 IP”，在窗口中将看到 Samba 服务器上的共享资源，如图 8-6 所示。双击共享目录 work 的图标，不用输入用户名和密码即可进入共享目录，进行权限范围内的相关操作。

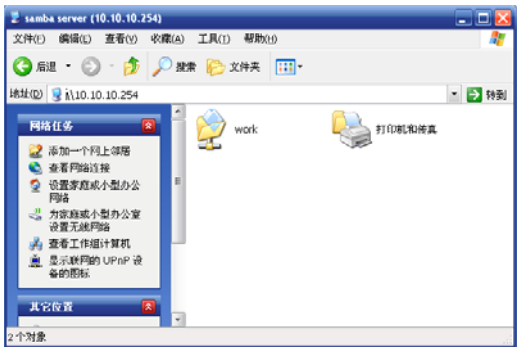

图 8-5　从 Windows 访问 Samba 服务器

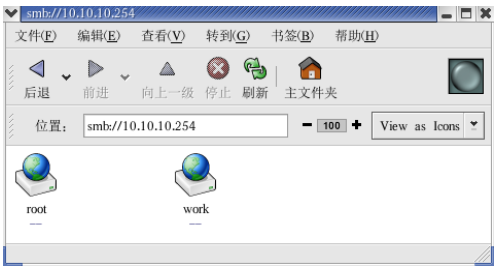

图 8-6　从 Linux 访问 Samba 服务器

8.1.7　图形界面下配置 Samba 服务器

对于初学者用户，也可以在图形界面下配置 Samba 服务器。图形界面的配置虽然简单、直观，但对于某些高级选项，图形界面下的配置工具并不能够地很好实现。因此，要想对 Samba 服务进行精细化的管理，还是要采取直接编辑配置文件的方法来实现。

依次单击“主菜单”→“系统设置”→“服务器设置”→“Samba 服务器”菜单，或者直接执行“redhat-config-Samba”命令，系统会自动弹出如图 8-7 所示的“Samba 服务器配置”窗口。

图 8-7　Samba 服务器配置主窗口

1. 配置基本和安全性选项

Samba 服务器的第一步是配置服务器的基本选项和安全选项。在图 8-7 所示界面中，选择“首

选项”→“服务器设置”选项，打开“服务器设置”窗口，选择“基本”选项卡，如图 8-8 所示。

图 8-8　“基本”选项卡

图 8-9　“安全性”选项卡

在“基本”选项卡上，指定计算机所属的工作组及计算机的简短描述。它们分别与配置文件 smb.conf 中的 workgroup 和 server string 选项相对应。

在“安全性”选项卡中，可以分别对验证模式、验证服务器、加密口令以及来宾账号进行设置，如图 8-9 所示。它们分别与 smb.conf 中的 security、password server、encrypt passwords、guest account 选项相对应。

2. 管理 Samba 用户

Samba 服务器配置工具要求在添加 Samba 用户前，在充当 Samba 服务器的 Red Hat Linux 9 系统上必须存在一个活跃的用户账号。Samba 用户和这个 Linux 用户账号相关联。

要添加 Samba 用户，可以在图 8-7 中选择“首选项”→“Samba 用户”，然后单击“添加用户”按钮，弹出图 8-10 所示的“创建新 Samba 用户”窗口。

在“创建新 Samba 用户”窗口中，从本地系统上的现存用户列表中选择“UNIX 用户名”用来将其转换为 Samba 用户，如果用户想从 Windows 机器上登录并且使用一个不同的名字，就需要在“Windows 用户名”字段中指定 Windows 用户名。还需要为 Samba 用户配置一个 Samba 口令，并再输入一次来确认这个口令。即便选择了为 Samba 使用加密口令，仍建议为所有用户设置的 Samba 口令不同于他们的 Linux 系统登录口令。

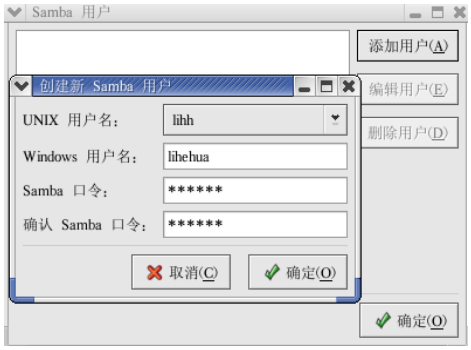

图 8-10　创建新 Samba 用户

另外，“服务器设置”—“安全”选项卡上的“验证模式”必须被设置为“用户”，才能使这里的设置在访问 Samba 服务器时有效。

要编辑某个现存 Samba 用户，可以从 Samba 用户列表中将其选中，然后单击“编辑用户”

按钮。要删除某个现存的 Samba 用户，先选择这个用户，然后单击“删除用户”按钮。需要注意的是删除 Samba 用户并不会删除与其相关的 Linux 用户。

3. 添加共享目录

要添加共享，可在图 8-7 所示界面中单击“添加共享”按钮，打开“创建 Samba 共享”窗口，在“基本”选项卡中配置以下选项，如图 8-11 所示。

图 8-11　“基本”选项卡

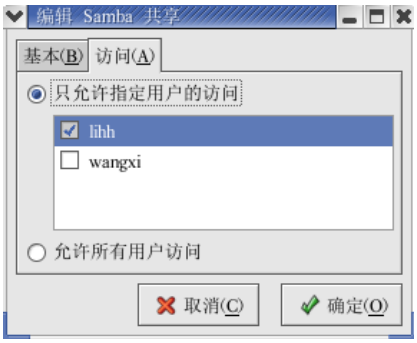

图 8-12　“访问”选项卡

（1）目录。通过 Samba 服务器共享的目录。这个目录必须存在，这里配置为“/home/lihh/share”。

（2）描述。是对共享资源的简短描述，可以设置也可以不设置。

（3）在“描述”下面是用户访问该共享的基本权限，设置用户能读/写共享目录中的文件还是仅仅只能读取。

在“访问”选项卡上，设置仅允许指定的用户访问共享还是允许所有的用户访问。如果选择“只允许指定用户的访问”，就从用户列表中选定用户，如图 8-12 所示。“基本”选项卡和“访问”选项卡设置完毕后，单击“确定”按钮后，共享就被添加。成功添加共享目录后的 Samba 服务器配置主窗口如图 8-13 所示。选中窗口中某一个共享项，还可以将其删除或修改其属性。修改后的设置，要重启 Samba 服务后才能生效。

图 8-13　添加共享后的 Samba 服务器配置窗口

8.2　NFS 服务器

8.2.1　NFS 概述

Samba 服务器主要用来解决 Windows 与 Linux 之间的资源共享，那么 Linux 与 Linux 之间

的资源共享又如何实现呢？实际上，也可以使用 Samba 服务器，不过这里将介绍另外一种更为便捷的服务——NFS。

NFS（Network File System，网络文件系统）是由 Sun 公司（Sun Microsystem,Inc.）于 1984 年推出的一个 RPC（Remote Procedure Call，远程过程调用）服务系统，它使 Linux、UINX 系统之间能够共享文件，类似 Windows 系统中的资源共享。

NFS 是基于客户端/服务器模式工作的。输出文件的计算机称为 NFS 服务器，而 NFS 客户端是访问文件的计算机。NFS 服务器上的目录被远程用户访问的过程叫“导出（export）”，客户端访问服务器则被称为“导入（import）”。

客户端和服务器通过 RPC 通信，当客户端上的应用程序访问远程文件时，客户端内核向远程 NFS 服务器发送一个请求，等待服务器响应，而 NFS 服务器一直处于等待状态，如果接收到客户端请求，就处理请求并将结果返回客户端。

当用户想使用远程文件时要使用 mount 命令把远程文件系统挂载到本地文件系统下，挂载后就像使用本地计算机上的文件一样。这样做可以使网络中的不同主机直接访问同一个文件，而不必在每台主机上都维护一个副本。

NFS 服务器是由一组守护进程在后台运行的，用以完成服务器的功能。4 个服务器守护进程如下。

（1）inetd。网络服务进程，启动 inetd.conf 配置文件所设置的网络服务，应答客户端的网络服务请求。

（2）portmap。将 TCP/IP 通信协议端口数字转化为 RPC 程序数字，使客户端能够进行 RPC 调用。

（3）nfsd。NFS 服务守护进程，启动文件系统请求服务，响应客户端对文件系统的请求。

（4）mountd。负责响应远程客户端的安装请求。

8.2.2 NFS 服务器安装

在 Red Hat Linux 9 安装时，可以选择安装 NFS 服务器，其内置的 NFS 服务器版本为 nft-utils-1.0.1-2.9，如果不知道是否已经安装了此版本的软件，可以使用以下的方法判断。

```
[root@rh9 root]# rpm   -qa |grep   nfs
redhat-config-nfs-1.0.4-5        //Linux 图形界面下的 NFS 配置工具，选装。
nfs-utils-1.0.1-2.9
```

如果看到上面的结果，则表示该软件已经安装。否则，可以找出第一张安装光盘，redhat-config-nfs-1.0.4-5.rpm 和 nfs-utils-1.0.1-2.9.rpm 软件包都在 RedHat/RPMS 目录的下面。可以使用下面的命令进行安装。

```
[root@rh9 dhcp]# mount   /mnt/cdrom
[root@rh9 dhcp]# cd   /mnt/cdrom/Red Hat/RPMS
[root@rh9 root]# rpm   -ivh   nft-utils-1.0.1-2.9.i386.rpm
[root@rh9 root]# rpm   -ivh   redhat-config-nfs-1.0.4-5.i386.rpm
```

8.2.3 NFS 服务器的启动停止

NFS 服务器也可以像前面介绍的其他服务器一样，采用同样的几种方法进行启动、关闭，或者设置自动启动等操作。下面仅以 service 和 chkconfig 命令来说明。

1. 使用 service 命令

NFS 服务器需要 portmap 服务的配合，所以需要先启动 portmap 服务，再启动 nfs 服务。

```
[root@rh9 root]# service    nfs    status
rpc.mountd    已停
nfsd    已停
rpc.rquotad    已停
[root@rh9 root]# service    portmap    start
启动  portmapper:                                    [    确定    ]
[root@rh9 root]# service    nfs    start
启动  NFS  服务:                                     [    确定    ]
Starting NFS quotas:                                 [    确定    ]
启动  NFS  守护进程:                                 [    确定    ]
启动  NFS mountd:                                    [    确定    ]
```

2. 使用 chkconfig 命令

若要系统每次启动时自动开启 nfs 服务，可以使用 chkconfig 命令，下面的例子表示在系统进入第 3 和第 5 个级别时自动开启 portmap 和 vsftpd 服务。

```
[root@rh9 root # chkconfig    --level    35    portmap    on
[root@rh9 root # chkconfig    --level    35    nfs    on
[root@rh9 root # chkconfig    --list    portmap
portmap         0:关闭    1:关闭    2:关闭    3:启用    4:启用    5:启用    6:关闭
[root@rh9 root # chkconfig    --list    nfs
nfs               0:关闭    1:关闭    2:关闭    3:启用    4:关闭    5:启用    6:关闭
```

8.2.4　NFS 服务器的配置

NFS 只有一个配置文件/etc/exports，该文件在默认情况下只允许 root 用户更改，当 NFS 启动时会自动读取该文件中的配置，向网络中的其他 Linux 主机共享资源。默认该文件为空，共享文件时，可以按照如下语法添加内容。

/etc/exports 文件中的每一行代表一个不同的共享资源，用户可以根据情况自行设定。其语法格式为：[共享目录］[客户机 1（选项 1,选项 2…)］[客户机 2（选项 1,选项 2…）]。

（1）共享目录。是要导出的文件系统或目录名称，也就是要共享给客户机使用的目录，该目录必须是绝对路径。

（2）客户机。同一共享目录可以针对不同的客户机设置不同的参数，客户机可以是 IP 地址也可以是 NetBIOS 主机名。需要注意的是，这些主机名必须是在/etc/hosts 文件中已经定义过的。否则，NFS 系统可能会无法找到指定名称的主机。

（3）选项。用于设置 NFS 客户机使用导出目录的权限，这些选项分为性能选项和安全选项，数量众多。限于篇幅，下面仅简单介绍几个常用的选项。

① rw：读/写权限，只读权限的参数为 ro。

② sync：数据同步写入内存和硬盘，也可以使用 async，此时数据会先暂存于内存中，而不立即写入硬盘。

③ root_squash：登录 NFS 主机使用共享目录的用户如果为 root，那么这个用户的权限将被压缩为匿名用户，通常他的 UID 与 GID 都会变成 nobody 系统账号的身份。

④ no_root_squash：NFS 服务器共享目录用户的属性，如果用户是 root 用户，那么对于

这个共享目录来说就具有 root 用户的权限。这个参数非常不安全，建议不要使用。

⑤ all_squash：不论登录 NFS 的用户身份为何，该身份都会变成 nobody，也就是匿名用户。

下面是一个共享出两个目录/home/work 和/tmp 的 NFS 服务器配置实例。这时，需要在/etc/exports 文件中增加两行。

```
/tmp    rh9（rw,sync）    *（ro,sync）
```

表示名称为 lihost 的主机对共享目录/tmp 有读写的权限，其他所有主机对共享目录/tmp 的权限是只读。

```
/home/work    192.168.0.*（rw,sync,no_root_squash）
```

表示允许 IP 地址范围在“192.168.0.*”的所有计算机以读写的权限来访问/home/work 目录。

8.2.5 维护共享目录列表

当修改了/etc/exports 文件的内容后，要想让新的配置文件生效，可以重新启动 NFS 服务。实际上，也可以在不重新启动 NFS 服务的情况下，直接使用 exportfs 命令使新的设置立即生效。

exportfs 命令是用来维护 NFS 服务输出目录列表的，命令的基本格式是：“exportfs [选项]”，该命令的选项及作用如表 8.3 所示。

表 8.3 命令 exportfs 的选项及含义

选项	作用
-a	输出在文件/etc/exports 中设置的所有共享目录
-r	重新读取/etc/exports 文件中的设置，并使设置生效，而不需要重新启动 NFS 服务
-u	停止输出某一目录
-v	显示 exportfs 命令执行的过程

```
[root@rh9 root]# more    /etc/exports
/tmp    rh9（rw,sync）    *（ro,sync）
[root@rh9 root]# vi     /etc/exports
/tmp    rh9（rw,sync）    *（ro,sync）
/home/work    192.168.0.*（rw,sync,no_root_squash）        //增加该行，保存、退出 vi。
[root@rh9 root]# exportfs            //不带选项时，显示 NFS 服务器当前的输出目录。
/tmp                    rh9
/tmp                    <world>
[root@rh9 root]# exportfs     -rv
exporting rh9:/tmp
exporting 192.168.0.*:/home/work
exporting *:/tmp
reexporting rh9:/tmp to kernel
[root@rh9 root]# exportfs
/tmp                    rh9
/home/work          192.168.0.*
/tmp                    <world>
```

8.2.6　图形界面下配置 NFS 服务器

除直接编辑配置文件/etc/exports 外，也可以使用图形界面下的配置工具配置 NFS 服务器。图形界面下的配置虽然简单、直观，但对于某些高级选项，图形界面并不能够实现。

依次单击“主菜单”→“系统设置”→“服务器设置”→“NFS 服务器”菜单，或者直接在终端中执行“redhat-config-nfs”命令，系统会自动弹出如图 8-14 所示的“NFS 服务器配置”窗口。

图 8-14　NFS 服务器配置主窗口

在“NFS 服务器配置”窗口中，单击“增加”按钮，将弹出“添加 NFS”共享窗口，其中包含 3 个选项卡，即“基本”、“常规选项”和“用户访问”。以下分别说明这些选项卡中的选项。

1. “基本”选项卡

如图 8-15 所示，在“基本”选项卡中包含的是 NFS 共享目录最重要的选项。其中“目录”用来指定共享目录的绝对路径；“主机”用来指定 NFS 服务器的主机名称或别名；“基本权限”用来设置此共享目录的默认访问权限，可以使用的选项是“只读”或“读/写”。

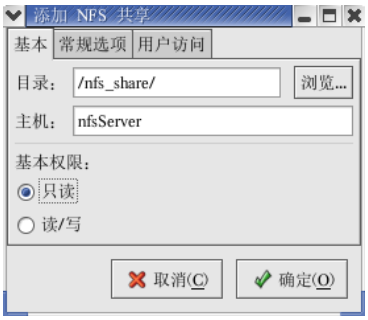

图 8-15　“基本”选项卡

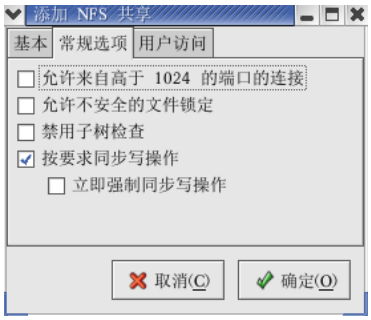

图 8-16　“常规选项”选项卡

2. “常规选项”选项卡

在“常规”选项卡中，包含的是 NFS 共享目录较高级的设置，在一般情形下并不需要修改此处的内容，如图 8-16 所示。以下是这些选项的说明。

（1）允许来自高于 1024 的端口的连接。在默认的情况下，NFS 只允许使用小于 1024 的连接端口进行连接，如果要开放大于 1024 的连接端口连接，需选择此项。

（2）允许不安全的文件锁定。为了兼容较早版本的 NFS 服务器，因为它们并不支持文件

锁定的功能。

（3）禁用子树检查。通常在客户端请求共享目录中的文件时，NFS 服务器不仅会检查客户端对此文件的访问权限，还会检查该共享目录所在的整个文件系统，属于中等的安全性设计。在选择此选项后，NFS 服务器将会停用此类的检查。

（4）按要求同步写操作。这个选项与“sync”选项的功能相同，如果不使用此功能，在运行“exportfs”命令时会出现警告信息。

（5）强制立即同步操作。立即将修改后的信息进行同步写入磁盘中。

3. “用户访问”选项卡

在“用户访问”选项卡中，包含的是客户端在访问 NFS 共享目录时的安全性设置，如图 8-17 所示。以下是这些选项的说明：

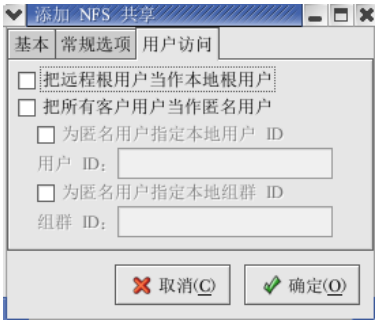

图 8-17 “用户访问”选项卡

（1）把远程根用户当作本地根用户。与“no_root_squash”选项相同，但为了提高安全性，并不建议使用此项设置。

（2）把所有客户用户当作匿名用户。与“all_squash”选项相同，为了提高安全性，建议使用此项设置。

（3）为匿名用户指定本地用户 ID。除非使用“把所有客户用户当作匿名用户”选项，否则无法选择此项目，同时必需在“用户 ID”字段中输入匿名用户所使用的本机用户 ID。

（4）为匿名用户指定本地组群 ID。除非使用“把所有客户用户当作匿名用户”选项，否则无法选择此项目，同时必需在“组群 ID”字段中输入匿名用户所属的本机组群 ID。

将上述 3 个选项卡按照实际需要设置后，单击“确定”按钮，新添加的共享就会出现在图 8-14 所示的 NFS 服务器配置主窗口中。如果要针对某一共享目录进行修改，首先选择此共享目录名称，然后单击上方工具栏中的“属性”按钮，系统会出现上面所讲的“NFS 共享”窗口，可以对相应的设置进行修改。修改后的设置将在 NFS 服务重启后生效。

8.2.7 NFS 客户机链接

在 NFS 服务器设置完成后，客户端就可以依据本身所拥有的权限来访问服务器上的共享资源，并且将远程共享目录安装到本机的文件系统中。下面将讨论有关客户端连接时的内容。

1. 查看 NFS 服务器上的共享资源

客户端如果要查看 NFS 服务器上的共享资源，可以使用 NFS 软件包中的“showmount”命令。实现该功能的命令格式是：“showmount -e [NFS 服务器]”。

```
[root@rh9 root]# showmount    -e      10.10.10.254
Export list for 10.10.10.254:
/home/work 192.168.0.*
/tmp              （everyone）
[root@rh9 root]# showmount    -e    10.10.10.1
mount clntudp_create: RPC: Port mapper failure - RPC: Unable to receive
```

如果指定的主机没有提供 NFS 服务，或 NFS 服务还没有启动，在执行该命令后将显示“mount clntudp_create: RPC: Port mapper failure - RPC: Unable to receive”的提示结果。如果在命令中没有指定 NFS 服务器，则默认 NFS 服务器是本机。

2. 安装共享资源到客户机

在利用 showmount 命令得知远程 NFS 服务器上的共享资源后，接下来就是进行实际的安装工作，在此使用“mount”命令，其格式是：“mount　NFS 服务器:共享目录　本机安装目录”。如果已经不再需要访问共享目录，可以使用“umount”命令来卸载已经挂载到本地的远程目录。

以下的范例将把 NFS 服务器（rh9）上的共享目录（/home/work）挂载到客户机上的/mnt/nfs 目录。但是在挂载前，必须先确定客户端对共享目录有足够的访问权限，并且在本机上已经提前建立了用于挂载的目录。

```
[root@rh9 root]# mkdir    /mnt/nfs    -v                    //创建远程共享目录的本地挂载点。
mkdir: 已创建目录‘/mnt/nfs’
[root@rh9 root]# ls    /mnt/nfs
[root@rh9 root]# mount    10.10.10.254:/tmp    /mnt/nfs
[root@rh9 root]# ls    /mnt/nfs                          //查看 NFS 服务器共享出来的文件。
evolution    orbit-root    ssh-XXrmBbob
[root@rh9 root]# umount    /mnt/nfs                      //卸载挂载的远程共享目录。
[root@rh9 root]# ls    /mnt/nfs
[root@rh9 root]# mount    10.10.10.254:/home/work    /mnt/nfs
mount: 10.10.10.254:/home/work failed, reason given by server: 权限不够
```

8.3　Apache 服务器

8.3.1　Apache 概述

Apache 是 Internet 上最流行的 Web 服务器软件，它安全、高效、稳定、适用于各种平台，关键它还是免费的和开放源代码的。从 1995 年开始直到今天，经过不断的发展，Apache 小组开发的 Apache httpd 服务器软件已经成为市场的领导者。在所有的 Web 服务器软件中，Apache 占有绝对优势，远远领先于 Microsoft 的 IIS 服务器软件。

许多世界知名的网站都是基于 Linux 操作系统的，如 Yahoo、Hotmail 等网站。在 Linux 系统上架设 Web 服务器所使用的最多、最广泛的软件就是 Apache。它功能强大，高度稳定，与 Linux 配合得十分完美，是在 Linux 上构建 Web 站点时首选的方案。另外，也有 Windows 平台下的 Apache 软件。可以到官方网站 http://www.apache.org 免费下载 Apache 最新的版本。

8.3.2　Apache 服务器的安装

在安装 Red Hat Linux 9 时，可以选择安装 Apache 服务器，而在 Red Hat Linux 9 中内置的

Apache 服务器版本为 httpd-2.0.40-21，如果不知道是否已安装此版本的软件，可以使用下面的方法来判断。

```
[root@rh9 root]# rpm   -qa   httpd
httpd-2.0.40-21.i386.rpm
```

如果看到上面的结果，则表示该软件已经安装。否则，需要找出第二张安装光盘，保存在 RedHat/RPMS 目录的 httpd-2.0.40-21.i386.rpm。可以使用下面的命令进行安装。

```
[root@rh9 root]# mount   /mnt/cdrom
[root@rh9 root]# cd   /mnt/cdrom/Red Hat/RPMS
[root@rh9 root]# rpm   -ivh   httpd-2.0.40-21.i386.rpm
```

另外，为了使用图形化的 Apache 服务器管理工具，建议用户也安装 redhat-config-httpd-1.0.1-18.i386.rpm 软件包。

8.3.3 Apache 服务器的基本配置

Apache 的配置文件是包含了若干指令的纯文本文件，其文件名为 httpd.conf，在 Apache 启动时，会自动读取配置文件中的内容，并根据配置指令影响 Apache 服务器的运行。配置文件改变后，只有在下次启动或重新启动后才会生效。

配置文件中的内容分为注释行和服务器配置命令行。行首有“#”的即为注释行，注释不能出现在指令的后边，除了注释行和空行外，服务器会认为其他的行都是配置命令行。配置文件中的指令不区分大小写，但指令的参数通常是对大小写敏感的。对于较长的配置命令，行末可使用反斜杠“\”换行，但反斜杠与下一行之间不能有任何其他字符（包括空白）。可以使用 apachectl 或者 httpd 的命令行参数-t 来检查配置文件中的错误，而无需启动 Apache 服务器。

```
[root@rh9 root]# httpd   -t
httpd: Could not determine the server's fully qualified domain name, using 127.0.0.1
for ServerName
Syntax OK
[root@rh9 root]# apachectl    -t
httpd: Could not determine the server's fully qualified domain name, using 127.0.0.1
for ServerName
Syntax OK
```

整个配置文件总体上划分为 3 部分（section），第 1 部分为全局环境设置；第 2 部分是服务器的主要配置；第 3 部分用于设置和创建虚拟主机。下面介绍一些常用的配置命令。

1. 常规配置指令

（1）ServerRoot。所谓 ServerRoot 是指整个 Apache 目录结构的最上层，在此目录下可包含服务器的配置、错误和日志等文件。如果安装时使用 rpm 版本的方式，则默认目录是/etc/httpd，一般不需要修改。注意，这里不能在目录路径的后面加上斜线（/）。

（2）ServerName。设置服务器用于辨识自己的主机名和端口号，该设置仅用于重定向和虚拟主机的识别。命令用法为：“ServerName　完全合格的域名[:端口号]”。

对于 Internet 的 Web 服务器，应保证该名称是 DNS 服务器中的有效记录。默认配置文件中对此没有设置，应根据服务器的实际情况进行设置。比如当前 Web 服务器的域名为 www.cqcet.cn，则可设置为：ServerName www.cqcet.cn 或 ServerName www.cqcet.cn:80。

当没有指定 ServerName 时，服务器会尝试对 IP 地址进行反向查询来获得主机名。如果在

服务器名中没有指定端口号，服务器会使用接受请求的端口。为了加强可靠性和可预测性，应使用 ServerName 显式地指定一个主机名和端口号。

（3）Listen。Listen 命令告诉服务器接受来自指定端口或者指定地址的某端口的请求。如果 Listen 仅指定了端口，则服务器会监听本机的所有地址；如果指定了地址和端口，则服务器只监听来自该地址和端口的请求。利用多个 Listen 指令，可以指定要监听的多个地址和端口，比如在使用虚拟主机时，对不同的 IP、主机名和端口需要作出不同的响应，此时就必须明确指出要监听的地址和端口。其命令用法为：“Listen　[IP 地址]:端口号”。Web 服务器使用标准的 80 号端口，若要对当前主机的 80 端口进行侦听，则配置命令为：Listen 80，假设当前服务器绑定了 61.186.160.104 和 61.186.160.105 IP 地址，现需要对其 80 端口和 8080 端口进行监听，则配置命令如下。

```
Listen　61.186.160.104：80
Listen　61.186.160.104：8080
Listen　61.186.160.105：80
Listen　61.186.160.105：8080
```

（4）ServerAdmin。用于设置 Web 站点管理员的 E-mail 地址。当服务器产生错误时（如指定的网页找不到），服务器返回给客户端的错误信息中将包含该邮件地址，以告诉用户该向谁报告错误。其命令用法为：“ServerAdmin　E-mail 地址”。

（5）DocumentRoot。用于设置 Web 服务器的站点根目录，其命令用法为：“DocumentRoot 目录路径名”，默认设置为：DocumentRoot　"/var/www/html"，注意，目录路径名的最后不能加“/”，否则将会发生错误。

（6）ErrorDocument。用于定义当遇到错误时，服务器将给客户端什么样的回应，通常是显示预设置的一个错误页面。其命令用法为：“ErrorDocument 错误号 所要显示的网页”。在默认的配置文件中，预定义了一些对不同错误的响应信息，但都注释掉了，只需去掉前面的“#”号即可开启。

（7）DirectoryIndex。用于设置站点主页文件的搜索顺序，各文件间用空格分隔。例如，要将主页文件的搜索顺序设置为 index.php、index.html、index.htm、default.htm，则配置命令为：“DirectoryIndex index.php index.html index.htm default.htm”。

（8）User 和 Group。User 用于设置服务器以哪种用户身份来响应客户端的请求。Group 用于设置将由哪一组来响应用户的请求。User 和 Group 是 Apache 安全的保证，千万不要把 User 和 Group 设置为 root。

（9）AddDefaultCharset。用于指定默认的字符集。在 HTTP 的回应信息中，若在 HTTP 头中未包含任何关于内容字符集类型的参数时，此指令将指定的字符集添加到 HTTP 头中，此时将覆盖网页文件中通过 META 标记符所指定的字符集。命令用法为：“AddDefaultCharset 字符集名称”。

Apache 默认的字符集为 ISO-8859-1，对于含有中文字符的网页，若网页中没有指定字符集，则在显示中文的时候会出现乱码，解决的办法就是将默认字符集设置为 GB2312，其配置命令为：“AddDefaultCharset GB2312”。

2. 性能配置指令

一般情况下，每个 HTTP 请求和响应都使用一个单独的 TCP 连接，服务器每次接受一个

请求时，都会打开一个 TCP 连接并在请求结束后关闭该连接。若能对多个处理重复使用同一个连接，则可减小打开 TCP 连接和关闭 TCP 连接的负担，从而提高服务器的性能。

（1）Timeout。用于设置连接请求超时的时间，单位为秒。默认设置值为 300，超过该时间，连接将断开。若网速较慢，可适当调大该值。

（2）KeepAlive。用于启用持续的连接或者禁用持续的连接。其命令用法：“KeepAlive on|off”，配置文件中的默认设置为 KeepAlive on。

（3）MaxKeepAliveRequests。用于设置在一个持续连接期间允许的最大 HTTP 请求数目。若设置为 0，则没有限制；默认设置为 100，可以适当加大该值，以提高服务器的性能。

（4）KeepAliveTimeout。用于设置在关闭 TCP 连接之前，等待后续请求的秒数。一旦接受请求建立了 TCP 连接，就开始计时，若超出该设定值还没有接收到后续的请求，则该 TCP 连接将被断开。默认设置为 10 秒。

（5）控制 Apache 进程。对于使用 prefork 多道处理模块的 Apache 服务器，对进程的控制，可在 prefork.c 模块中进行设置或修改。配置文件的默认设置如下。

```
<IfModule prefork.c>
StartServers          8
MinSpareServers       5
MaxSpareServers      20
MaxClients          150
MaxRequestsPerChild   1000
</IfModule>
```

在配置文件中，属于特定模块的指令要用<IfModule>指令包含起来，使之有条件地生效。<IfModule prefork.c>表示如果 prefork.c 模块存在，则在<IfModule prefork.c>与</IfModule>之间的配置指令将被执行，否则不会被执行。下面分别介绍各配置项的功能。

① Startservers

用于设置服务器启动时启动的子进程的个数。

② MinSPareservers

用于设置服务器中空闲子进程（即没有 HTTP 处理请求的子进程）数目的下限。若空闲子进程数目小于该设置值，父进程就会以极快的速度生成子进程。

③ MaxSPareservers

用于设置服务器中空闲子进程数目的上限。若空闲子进程超过该设置值，则父进程就会停止多余的子进程。一般只有在站点非常繁忙的情况下，才有必要调大该设置值。

④ Maxclient

用于设置服务器允许连接的最大客户数，默认值为 150，该值也限制了 httpd 子进程的最大数目，可根据需要进行更改，比如更改为 500。

⑤ MaxRequestsPerChild

用于设置子进程所能处理请求的数目上限。当到达上限后，该子进程就会停止。若设置为 0，则不受限制，子进程将一直工作下去。

3. 日志配置指令

日志对于 Web 站点必不可少，它记录着服务器处理的所有请求、运行状态和一些错误或警告等信息。要了解服务器上发生了什么，就必须检查日志文件，虽然日志文件只记录已经发

生的事件，但是它会让管理员知道服务器遭受的攻击，并有助于判断当前系统是否提供了足够的安全保护等级。

（l）ErrorLog。用于指定服务器存放错误日志文件的位置和文件名，默认设置为：“ErrorLog logs/error_log”。

此处的相对路径是相对于 ServerRoot 目录的路径。在 error_log 日志文件中，记录了 Apache 守护进程 httpd 发出的诊断信息和服务器在处理请求时所产生的出错信息。在 Apache 服务器出现故障时，可以查看该文件以了解出错的原因。

（2）LogLevel。用于设置记录在错误日志中信息的数量，其中可能出现的记录等级依照重要性降序排列，分别是：debug、info、notice、warn、error、crit、alert 和 emerg。

当指定了某个特定级别后，所有级别高于它的信息也将被记录在日志文件中。配置文件中的默认配置级别为 warn，可根据需要进行调整。级别设置过低，将会导致日志文件的急剧增大。以下是/var/log/httpd/error_log 记录中的部分内容。

```
[root@rh9 root]# more    /var/log/httpd/error_log
[Mon Nov 09 14:11:01 2009] [notice] Digest: generating secret for digest authentication ...
[Mon Nov 09 14:11:01 2009] [notice] Digest: done
[Mon Nov 09 14:11:02 2009] [notice] Apache/2.0.40 (Red Hat Linux) configured --
resuming normal operations
[Mon Nov 09 14:11:21 2009] [warn] child process 6052 still did not exit, sending a SIGTERM
[Mon Nov 09 14:11:21 2009] [warn] child process 6053 still did not exit, sending a SIGTERM
[Mon Nov 09 14:11:21 2009] [notice] caught SIGTERM, shutting down
--More--(25%)
```

（3）LogFormat。此选项用来定义“CustomLog”指令中使用的格式名称，以下是系统默认的格式，可以直接使用这些默认值。

```
[root@rh9 root]# grep    LogFormat    /etc/httpd/conf/httpd.conf
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
```

（4）CustomLog。此选项可以用来设置记录文件的位置和格式，默认值是：“CustomLog logs/access_log combined”。

（5）PidFile。用于指定存放 httpd 主（父）进程号的文件名，便于停止服务。其默认值是：“PidFile run/httpd.pid”。

4. 容器与访问控制指令

（1）容器指令简介。容器指令通常用于封装一组指令，使其在容器条件成立时有效，或者用于改变指令的作用域。容器指令通常成对出现，具有以下格式特点。

<容器指令名 参数>

</容器指令名>

例如：

```
<IfModule mod_ssl.c>
Include    conf/ssl.conf
</IfModule>
```

<IfModule>容器用于判断指定的模块是否存在，若存在（被静态地编译进服务器，或是被

动态地装载进服务器），包含于其中的指令将有效，否则会被忽略。此处的配置指令的含义是：若 mod_ssl.c 模块存在，则用 Include 指令，将 conf/ssl.conf 配置文件包含进当前的配置文件中。

<IfModule>容器可以嵌套使用。若要使模块不存在时所包含的指令有效，只需在模块名前加一个“!”即可。比如配置文件中的以下配置。

```
<IfModule ! mpm_winnt.c>
<IfModule ! mpm_netware.c>
User    nobody
</IfModule>
</IfModule>
```

除了<IfModule>容器外，Apache 还提供了<Directory>、<Files>、<Location>、<VirtualHost>等容器指令。其中，<VirtualHost>用于定义虚拟主机；<Directory>、<Files>、<Location>等容器指令主要用来封装一组指令，使指令的作用域限制在容器指定的目录、文件或某个以 URL 开始的地址。在容器中，通过使用访问控制指令可实现对这些目录、文件或 URL 地址的访问控制。

（2）访问控制指令。访问控制指令由 Apache 的内建模块 mod_access 提供，它能实现基于 Internet 主机名的访问控制，其主机名可以是域名，也可以是一个 IP 地址，建议尽量使用 IP 地址，以减少 DNS 域名解析。相关的指令主要有 Allow、Deny 和 Order。

① Allow 命令

用法：Allow from hostlist。命令功能：指定允许访问的主机。hostlist 代表主机名列表，各主机名之间用空格分隔。该指令常用于<Directory>、<Files>、<Location>等容器中，以设置允许访问指定目录、文件或 URL 地址的主机。比如允许 61.186.160.104 主机访问，则实现命令为：Allow from 61.156.160.104。若要允许所有的主机访问，则实现命令可表达为：Allow from a11。

② Deny 命令

用法：Deny from host list。命令功能：该命令与 Allow 刚好相反，用于指定禁止访问的主机名。

③ Order 命令

用于指定 Allow 和 Deny 语句，哪一个被先执行。其具体用法有以下三种：Order Allow,Deny；Order Deny, Allow；Order mutual-failure。

“Order Allow,Deny”：表示 Allow 语句在 Deny 之前执行，若主机没有被特别指出允许访问，则该主机将被拒绝访问资源。

“Order Deny,Allow”：Deny 在 Allow 之前进行控制。若主机没有被特别指出拒绝访问，则该资源将被允许访问。

“Order mutual-failure”：只有那些在 Allow 语句中被指定，同时又没有出现在 Deny 语句中的主机，才允许访问。若主机在两条指令中都没有出现，则将被拒绝访问。

（3）对目录、文件和 URL 操作的容器。

① <Directory>容器

<Directory>容器用于封装一组指令，使其对指定的目录及其子目录有效。该指令不能嵌套使用，其命令用法如下。

```
<Directory   目录名>
……
```

```
</Directory>
```

容器中所指定的目录名可以采用文件系统的绝对路径，也可以是包含通配符的表达式。比如要设置所有主机均能访问/var/www/html 目录，则容器指令的表达如下。

```
<Directory　/var/www/html>
Order allow,deny
Allow from all
</Directory>
```

若要禁止所有主机通过 Apache 服务访问文件系统的根目录，则配置指令如下。

```
<Directory　/>
Order deny,allow
Deny from all
</Directory>
```

目录名可使用“*”或“?”通配符，“*”代表任意个字符，但不能通配“/”符号。“?”代表一个任意的字符。比如要对所有普通用户主目录下的 public_html 子目录进行配置，则此时的容器指令表达如下。

```
<Directory　/home/*/public_html>
Order allow,deny
Allow from all
</Directory>
```

如果有多个<Directory>容器配置段符合包含某文档的目录（或其父目录），那么指令将以最短目录、最先应用的规则进行应用。另外还提供了一个名为<DirectoryMatch>的容器，其用法与<Directory>相同，只是在指定目录名时，可直接使用正则表达式。<Directory>若要使用正则表达式，则需要在正则表达式前加“~”符号。

② <Files>容器

<Files>容器作用于指定的文件，而不管该文件实际存在于哪个目录。其命令用法如下。

```
<Files 文件名>
……
<Files>
```

文件名可以是一个具体的文件名，也可以使用“*”和“?”通配符。另外，还可使用正则表达式来表达多个文件，此时要在正则表达式前多加一个“~”符号。

比如配置文件中的以下配置，将拒绝所有主机访问位于任何目录下的以.ht 开头的文件，如.htaccess 和.htpasswd 等系统重要文件。

```
<Files ~ "\.ht">
Order allow,deny
Deny from all
<Files>
```

该容器通常嵌套在<Directory>容器中使用，以限制其所作用的文件系统范围。比如：

```
<Directory　/var/www/html>
<Files private.html>
order allow,deny
Deny from all
</Files>
</Directory>
```

以上配置将拒绝对 html 目录及其所有子目录下的 private.html 文件进行访问。

<FilesMatch>容器与<Files>用法相同，只是可以直接使用正则表达式来通配多个文件。

③ <Location>容器

<Location>容器是针对 URL 地址进行访问限制的，而不是 Linux 的文件系统。其命令用法如下。

```
<Location URL>
……
</Location>
```

比如要拒绝除 61.186.160.105 以外的主机对 URL 以/assistant 开头的访问，则配置命令为：

```
<Location /assistant>
Order deny,allow
Deny from all
Allow from 61.186.160.105
</Location>
```

在<Location>容器中，/assistant 代表 Web 站点根目录下的 assistant 目录。而在<Directory>容器中，最左边的“/”代表的是 Linux 文件系统的根目录。通过以上设置后，除 61.186.160.105 主机外，对 Web 站点根目录下的 assistant 目录，以及对其下子目录中的页面访问，都将被禁止。

5. 其他配置指令

（1）.htaccess 文件。.htaccess 文件也称为分布式配置文件，在该文件中也可放置一些配置指令，以作用于该文件所在的目录以及目录下的所有子目录。该文件可位于多个目录中，以分别对这些目录进行控制。功能上类似于<Directory>容器，但<Directory>和<Location>容器不能用在.htaccess 文件中，<Files>容器可以用于该文件。.htaccess 文件是在 httpd.conf 配置文件中，由以下命令配置指定的。

```
AccessFileName  .htaccess
```

在配置文件指定分布式配置文件.htaccess 后，若站点的根目录为/var/www/html，当用户在浏览器中请求 index.html 页面时，服务器会在该文档的各个路径中去查找第一个存在的.htaccess 配置文件，即 Apache 会试图打开/.htaccess、/var/.htaccess、/var/www/.htaccess、/var/www/html/.htacces。

对系统的配置均可在 httpd.conf 文件中实现，搜寻.htaccess 文件会降低系统性能，可通过在<Directory />中使用 Allowoverride None 命令，来禁止系统查找该文件。

```
<Directory  />
Allowoverride None
</Directory>
```

Allowoverride 命令用于设置原来设定的权限是否可以被.htaccess 文件中的权限所覆盖，其选项有 All 和 None 两个。若设置为 None，则不受.htaccess 的权限覆盖，此时系统就不再寻找和读取.htaccess 文件中的配置内容。

（2）Options 命令。Options 命令控制在特定目录中将使用哪些服务器特性，通常用在<Directory>容器中，其命令用法为“Options 功能选项列表”。可用的选项及功能如表 8.4 所示。对于 Linux 系统的根目录和 Web 站点根目录的访问控制，通常可设置为以下形式。

表 8.4　Options 命令可用的选项

选项	功能描述
None	不启用任何额外特性
All	除 Multiviews 之外的所有特性，默认设置
ExecCGI	允许执行 CGI 脚本
FollowSymLinks	服务器允许在此目录中使用符号连接。在<Location>字段中无效
Includes	允许服务器端包含 SSI（Server-side includes）
IncludesNOEXEC	允许服务器端包含，但禁用＃exec 和＃exe CGI 命令。但仍可以从 ScriptAliass 目录使用＃include 虚拟 CGI 脚本
Indexes	如果一个映射到目录的 URL 被请求，而此目录中又没有 DirectoryIndex（例如：index.html），那么服务器会返回一个格式化后的目录列表
MultiViews	允许内容协商的多重视图
SymLinksIfOwnerMatch	服务器仅在符号连接与其目的目录或文件拥有者具有同样的用户 id 时才使用它

```
<Directory />
Options FollowSymLinks
Allowoverride None
Order allow,deny
Deny from all
</Directory>
<Directory   "/var/www/html">
Options Indexes FollowSymLinks
Allowoverride None
Order allow,deny
Allow from all
</Directory>
```

8.3.4　配置虚拟主机

虚拟主机（Virtual Host）是指在一台主机上运行的多个 Web 站点，每个站点均有自己独立的域名，虚拟主机对用户是透明的，就好像每个站点都在单独的一台主机上运行一样。如果每个 Web 站点拥有不同的 IP 地址，则称为基于 IP 的虚拟主机；若每个站点的 IP 地址相同，但域名不同，则称为基于名字或主机名的虚拟主机，使用这种技术，不同的虚拟主机可以共享同一个 IP 地址，以解决 IP 地址缺乏的问题。

要实现虚拟主机，首先必须用 Listen 指令告诉服务器需要监听的地址和端口，然后为特定的地址和端口建立一个<VirtualHost>段，并在该段中配置虚拟主机。

1. 基于主机名的虚拟主机

基于主机名（域名）的虚拟主机是根据客户端提交的 HTTP 头中，关于主机名部分决定的。配置虚拟主机之前，应首先配置 DNS 服务器，让每个虚拟主机的域名都能解析到当前服务器所使用的 IP 地址，然后再配置 Apache 服务器，使其能辨识不同的主机名即可。由于 SSL 协议自身的原因，基于主机名的虚拟主机不能做成 SSL 安全服务器。

（l）虚拟主机的创建步骤。

① 在 DNS 服务器中为每个虚拟主机所使用的域名进行注册，让其能解析到服务器所使用的 IP 地址。

② 在配置文件中使用 Listen 指令，指定要监听的地址和端口。Web 服务器使用标准的 80 号端口，因此一般可配置为 Listen 80，让其监听当前服务器的所有地址上的 80 端口。

③ 使用 NameVirtualHost 指令，为一个基于域名的虚拟主机指定将使用哪个 IP 地址和端口来接受请求。如果对多个地址使用了多个基于域名的虚拟主机，则对每个地址均要使用此指令。命令用法：“Name VirtualHost 地址[：端口]”，端口号为可选项，若虚拟主机使用的是非标准的 80 号端口，则应明确指定所使用的端口号。比如基于域名的虚拟主机使用 61.186.160.104 这个 IP 地址，则指定方法为：“NameVirtualHost 61.186.160.104”。

另外也可表达为 NameVirtualHost *。此处的“*”通配任意的 IP 地址。当 IP 地址无法确定时，使用“*”是很方便的，比如服务器使用的是动态 IP 地址，而域名也是使用动态域名解析时，因为“*”匹配任何 IP 地址，无论 IP 地址如何变化，都不需要修改虚拟主机的配置。

如果希望在一个 IP 地址上运行一个基于域名的虚拟主机，而在另外一个地址上运行一个基于 IP 的或是另外一套基于域名的虚拟主机，此时就必须使用具体的 IP 地址，而不能使用“*”。

④ 使用<VirtualHost>容器指令定义每一个虚拟主机。<VirtualHost>容器的参数必须与 NameVirtualHost 后面所使用的参数保持一致。在<VirtualHost>容器中至少应指定 ServerName 和 DocumentRoot，另外可选的配置还有 ServerAdmin、DirectoryIndex、ErrorLog、CustomLog、TransferLog、ServerAlias、ScriptAlias 等，大部分的配置命令都可用在<VirtualHost>容器中，但与进程控制相关的 PidFile、TypesConfig、ServerRoot、Listen 和 NameVirtual 不能使用。

（2）虚拟主机的匹配方式。当一个请求到达时，服务器会首先检查它是否使用了一个能和 NameVirtualHost 相匹配的 IP 地址。如果匹配，就会查找每个与这个 IP 地址相对应的<VirtualHost>配置段，并尝试找出一个 ServerName 或 ServerAlias 配置项与请求的主机名（域名）相同的，若找到则使用该虚拟主机的配置，并响应其访问请求，否则将使用符合这个 IP 地址第一个列出的虚拟主机。从中可见，排在最前面的虚拟主机成为默认虚拟主机。当请求的 IP 地址与 NameVirtualHost 指令中的地址匹配时，主服务器中的 DocumentRoot 将永远不会被用到，因此，若要在现有的 Web 服务器上增加虚拟主机，必须也要为主服务器提供的 Web 站点创建一个<VirtualHost>配置块，在该虚拟主机中 ServerName 和 DocumentRoot 的内容应该与全局的 ServerName 和 DocumentRoot 保持一致，还要把这个虚拟主机放在所有<VirtualHost>的最前面，让其成为默认主机。

下面以几个具体的例子来说明基于 IP 地址的虚拟主机的应用，以及配置过程。

例子 1：假设当前服务器的 IP 地址为 192.168.168.154，现要在该服务器上创建两个基于域名的虚拟主机，使用端口为标准的 80，其域名分别为 www.mywebl.com 和 www.myweb2.com，站点根目录分别为/var/www/mywebl 和/var/www/myweb2，日志文件分别放在/var/rhlogs/mywebl/和/var/rhlogs/myweb2/目录下面，Apache 服务器原来的主站点采用域名 www.myweb.com 进行访问。服务器配置步骤如下：

① 注册虚拟主机所要使用的域名。

对于测试，可直接使用/etc/hosts 名称解析文件来进行域名的注册。对用于 Internet 的虚拟主机域名，则应在位于 Internet 的 DNS 服务器上进行注册登记。编辑/etc/host 文件，添加以下行“192.168.168.154 www.mywebl.com www.myweb2.com www.myweb.com”。然后使用命

令 ping 判断域名是否解析正常，如果能 ping 通则表示域名解析正常。

```
[root@rh9 root]# vi    /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1               rh9 localhost.localdomain      localhost
192.168.168.154    www.myweb1.com    www.myweb2.com    www.myweb.com
[root@rh9 root]# ping www.myweb.com
[root@rh9 root]# ping www.myweb1.com
[root@rh9 root]# ping www.myweb2.com
```

② 创建所需的目录。

```
[root@rh9 root]# mkdir    -p    /var/rhlogs/myweb1
[root@rh9 root]# mkdir    -p    /var/rhlogs/myweb2
[root@rh9 root]# mkdir    -p    /var/www/myweb1
[root@rh9 root]# mkdir    -p    /var/www/myweb2
```

③ 编辑 httpd.conf 配置文件，设置 Listen 指令侦听的端口。

```
Listen    192.168.168.154:80
```

④ 在 httpd.conf 配置文件的第三部分中，添加对虚拟主机的定义。添加的配置内容为：

```
NameVirtualHost 192.168.168.154
<VirtualHost 192.168.168.154>
ServerName www.myweb.com
DocumentRoot /var/www/html
ServerAdmin webrnaster@myweb.com
</VirtualHost>
<VirtualHost 192.168.168.154>
ServerName www.mywebl.com
DocumentRoot /var/www/mywebl
DirectoryIndex index.php index.html index.htm default.html default.html
ServerAdmin webmaster@mywebl.com
ErrorLog /var/vhlogs/mywebl/error_log
TransferLog /var/vhlogs/mywebl/access_log
</VirtualHost>
<VirtualHost 192.168.168.154>
ServerName www.myweb2.com
DocumentRoot /var/www/myweb2
Directorylndex index.php index.htm index.html default.htm default.html
ServerAdmin webmaster@myweb2.com
ErrorLog /var/vhlogs/myweb2/error_log
TransferLog /var/vhlogs/myweb2/access_log
</VirtualHost>
```

利用 DirectoryIndex 可为每个虚拟主机独立设置各主页文件的解析顺序。

⑤ 对用于存放 Web 站点的目录，设置访问控制。

```
<Directory /var/www>
Options FollowSymLinks
Allowoverride None
Order deny , allow
Allow from all
</Directory>
```

⑥ 保存 httpd.conf 配置文件，利用命令“apachectl -t”检查并确保虚拟主机配置正确。使用命令“service httpd restart”重启 Apache 服务器，以使配置生效。

⑦ 测试虚拟主机。

利用 VI 编辑器，在虚拟主机的站点根目录，分别创建 index.html 页面文件，并在页面的<body>与</body>之间输入不同的正文内容，以示区别。在图形界面下启动浏览器，然后在地址栏中分别键入 http://www.myweb.com、http://www.mywebl.com 和http://www.myweb2.com 并按 Enter键，若能看到 index.html 页面的内容，则虚拟主机创建成功。

在<VirtualHost>配置段中，可以单独为每个虚拟主机指定日志文件，若未指定，则日志统一记录在主日志文件中。主日志文件默认位于/etc/httpd/logs 目录下，文件名分别为 error_log 和 access_log。

当虚拟主机很多，而且每个主机又都使用了不同的日志文件时，Apache 可能会遇到耗尽 file handles（文件勾柄）的情况，从而无法创建文件。Linux/UNIX 操作系统限制了每个进程可以使用的 file handles 的数量，典型上限为 64 个，但可以扩充，直至到达一个很大的限制为止。若服务器上的虚拟主机很多，为防止日志文件过多，也可在<VirtualHost>配置段中不指定，而统一记录在系统的主日志文件中，此时为了记录该日志是哪一个虚拟主机产生的，应修改日志的记录内容和格式，并在日志内容的最开头添加“%v”变量，以记录虚拟主机的名称。

当需要分析阅读日志文件时，可使用 split-logfile 程序，将日志按虚拟主机名分组，拆分成一个个独立的日志文件，每个日志文件采用“虚拟主机名.log”形式命名，其中包含了该虚拟主机所产生的日志记录。

例子 2：现有某企业的服务器，配有两块网卡，IP 地址分别为内网地址 192.168.168.10 和外网地址 61.186.160.104，在 Internet 网中，企业域名 www.example.com 指向 61.186.160.104 地址，在企业内网的 DNS 服务器中，同样的域名指向 192.168.168.10。现要求为来自内网和外网的请求提供同样的 Web 服务。实现方法：在企业内网中应配置一个 DNS 服务器，并对 www.example.com 域名进行注册，使其解析到 192.168.168.10 地址。对于不能解析的域名，该 DNS 服务器应向上级 DNS 服务器提交域名解析请求。内网中的主机应配置 DNS 客户为内网 DNS 服务器的地址，以便在请求域名 www.example.com 时，能优先解析为 192.168.168.10，这样才能通过内网访问服务器。

要实现上述功能，用户需要在 httpd.conf 配置文件中，添加以下虚拟主机配置。

```
NameVirtualHost 192.168.168.10
NameVirtualHost 61.186.160.104
<VirtualHost 192.168.168.10 61.186.160.104>
DocumentRoot /var/www/Serverl
ServerName www.example.com
</VirtualHost>
```

例子 3：当前服务器的 IP 地址为 192.168.168.154，现要在该服务器上创建两个基于域名（主机名）的虚拟主机，域名分别为 www.myweb3.com 和 www.myweb4.com，每个虚拟主机的 80 端口和 8080 端口分别服务一个 Web 站点，其站点根目录分别为/var/www/myweb3-80、/var/www/myweb3-8080、/var/www/myweb4-80、/var/www/myweb4-8080。www.myweb3.com 的 80 端口作为默认 Web 站点。服务器配置步骤如下。

① 在/etc/hosts 文件中注册虚拟主机所要使用的域名，并测试是否能够解析。

② 分别在/var/www 下面创建所需的子目录：myweb3-80、myweb3-8080、myweb4-80、myweb4-8080。

③ 编辑 httpd.conf 配置文件，设置 Listen 指令侦听的端口。

```
Listen  192.168.168.154:80
Listen  192.168.168.154:8080
```

④ 在 httpd.conf 配置文件的第三部分中，添加对虚拟主机的定义。添加的配置内容如下。

```
NameVirtualHost 192.168.168.154:80
NameVirtualHost 192.168.168.154:8080
<VirtualHost 192.168.168.154:80>
ServerName www.myweb3.com
DocumentRoot /var/www/myweb3-80
</VirtualHost>
<VirtualHost 192.168.168.154:8080>
ServerName www.myweb3.com
DocumentRoot /var/www/myweb3-8080
</VirtualHost>
<VirtualHost 192.168.168.154:80>
ServerName www.myweb4.com
DocumentRoot /var/www/myweb4-80
</VirtualHost>
<VirtualHost 192.168.168.154:8080>
ServerName www.myweb4.com
DocumentRoot /var/www/myweb4-8080
</VirtualHost>
```

⑤保存 httpd.conf 配置文件，利用命令“apachectl　-t”检查并确保虚拟主机配置正确。使用命令“service httpd restart”重启 Apache 服务器，以使配置生效。

⑥利用 VI 编辑器，在虚拟主机的站点根目录，分别创建 index.html 页面文件，并在页面的<body>与</body>之间输入不同的正文内容，以示区别。

⑦ 测试虚拟主机。

利用域名虚拟主机，结合使用不同的端口，同一个 IP 地址可以创建出很多 Web 站点。在目前实际应用的虚拟主机服务中，对于同一个域名，通常开放了几个端口，80 端口服务于该域名的主 Web 站点，而将 8080 或其他端口提供的网站，用作对主网站的后台管理或用于提供该域名的基于 Web 的 E-mail 服务。

2. 基于 IP 地址的虚拟主机

基于 IP 的虚拟主机拥有不同的 IP 地址，这就要求服务器必须同时绑定多个 IP 地址。这可以通过在服务器上安装多块网卡，或通过虚拟 IP 接口（Red Hat Linux 将其称为 IP 别名）来实现，即在一张网卡上绑定多个 IP 地址。有两种配置方法使 Apache 支持基于 IP 地址的虚拟主机，一是为每个主机运行一个 httpd 守护进程，各守护进程的配置文件不同，分别以不同的 User、Group、Listen 和 ServerRoot 来运行，并通过 Listen 指令来指定为哪个 IP 地址和端口的虚拟主机服务。该方法适合于虚拟主机彼此之间的安全性要求很高的场合。启动 httpd 守护进程时，可使用命令“httpd　-f　配置文件名及路径”来指定所要加载的配置文件。

另一种方法是使用一个 httpd 守护进程来支持所有的虚拟主机。在服务器需要为大量请求服务的情况下，该方法可以获得较高的性能。下面主要针对该方法介绍基于 IP 地址的虚拟主机的实现方法。

例子 4：当前服务器有 192.168.167.156 和 192.165.167.157 两个 IP 地址，对应的域名分别为 www.example2.com 和 www.example3.com，试为其创建基于 IP 地址的虚拟主机，端口使用 80。这两个站点的根目录分别为/var/www/example2 和/var/www/example3。

服务器配置步骤如下。

① 注册虚拟主机所要使用的域名。编辑/etc/hosts 文件，在文件中添加以下两行内容。

```
192.168.167.156    www.example2.com
192.168.167.157    www.example3.com
```

② 创建 Web 站点根目录/var/www/example2 和/var/www/example3。

③ 编辑 httpd.conf 配置文件，保证有以下 Listen 指令。

```
Listen 80
```

④ 配置虚拟主机。

```
<VirtualHost 192.168.167.156>
ServerName www.example2.com
DocumentRoot /var/www/example2
</VirtualHost>
<VirtualHost 192.168.167.157>
ServerName www.example3.com
DocumentRoot /var/www/example3
</VirtualHost>
```

⑤ 在/var/www/example2 和/var/www/example3 目录中，利用 VI 编辑器创建 index.html 主页文件。

⑥ 重启 Apache 服务器，然后测试虚拟主机。若键入 http://localhost，返回的将是服务器的主站点的主页内容。

基于 IP 地址的虚拟主机，可以使用域名访问，也可使用 IP 地址访问。基于主机名的虚拟主机，应采用域名访问，若使用 IP 地址，则访问的是服务器的主站点。另外，在同一台主机上，还可以混用基于域名的虚拟主机和基于 IP 地址的虚拟主机。

8.3.5 Apache 服务器的启停与测试

1. *启动* Apache 服务器

在安装后，可以利用以下的方法来启动，并查看是否启动成功。

```
[root@rh9 root]# /etc/rc.d/init.d/httpd
{start|stop|restart|condrestart|reload|status|fullstatus|graceful|help|configtest}
[root@rh9 root]# /etc/rc.d/init.d/httpd     start
启动 httpd:                                                [  确定  ]
[root@rh9 root]# ps   -aux | grep httpd
(USER PID% CPU% MEM   VSZ   RSS TTY   STAT START   TIME   COMMAND)
root      6336   1.8    2.4   19516   8188   ?       S    14:36    0:00    /usr/sbin/httpd
apache   6339   0.0    2.4   19540   8216   ?       S    14:36    0:00    [httpd]
…  …
apache   6346   0.0    2.4   19540   8216   ?       S    14:36    0:00    [httpd]
root      6348   0.0    0.1   4816     640   pts/5   S    14:36    0:00    grep httpd
```

2. *重启* Apache 服务器

可以利用以下的方法重新启动 Apache 服务器。

```
[root@rh9 root]# service httpd restart
```

```
停止 httpd:                                                [  确定  ]
启动 httpd:                                                [  确定  ]
```

3. 停止 Apache 服务器

可以利用下面的方法关闭 Apache 服务器。

```
[root@rh9 root]# /etc/rc.d/init.d/httpd     stop
停止 httpd:                                                [  确定  ]
```

4. 开机时自动启动 Apache 服务器

因为 Web 服务在服务器中是相当重要的，所以在一般情况下应该设置为开机时自动启动，以节约每次手动启动的时间，并且避免因为忘记启动而导致的服务器停止。这里仅介绍使用命令 chkconfig 的实现方法。

```
[root@rh9 root]# chkconfig   --level   35   httpd    on
[root@rh9 root]# chkconfig   --list   httpd
httpd              0:关闭   1:关闭   2:关闭   3:启用   4:关闭   5:启用   6:关闭
```

5. 测试 Apache 服务器

在客户端使用的 Web 浏览器中输入 Apache 服务器的 IP 地址进行访问，如出现 Apache 的测试页面，如图 8-18 所示，则表示 Apache 服务器安装正确并且已经正常工作。

8.3.6　图形化配置 Apache 服务器

Apache 简单的配置可以使用图形配置工具，依次单击“系统”→“管理”→“服务器设置”→“HTTP 服务器”选项，或者在 X Window 环境下直接执行“apacheconf”命令，就可启动如图 8-19 所示的图形配置管理工具。

图 8-18　Apache 的测试页面

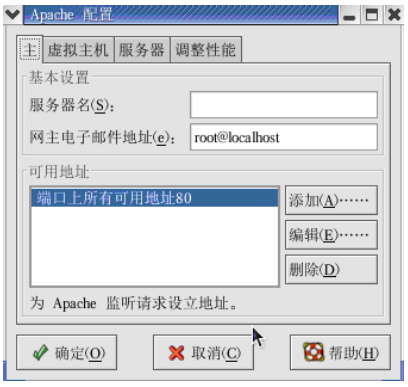

图 8-19　Apache 图形配置工具

Apache 服务器图形配置工具包括主、虚拟主机、服务器、调整性能等几个选项卡，每个选项卡都包含不同功能的配置。

1. “主”选项卡

（1）服务器名。在服务器名输入框中输入服务器的名字，这个名字应该是一个完全符合域名规则的名字，这里服务器名的设置与 Apache 的主配置 httpd.conf 中的 ServerName 指令对应。这个名字是 Web 服务器的主机名。如果不指定服务器名，则 Apache 服务器使用从系统获得的 IP 地址表示 Apache 服务器。Apache 服务器的名字不一定必须是计算机的 IP 地址所对应的实际域名。

（2）网主电子邮件地址。输入负责维护 Apache 服务器的管理员的 E-mail 地址。这个选项对应于 httpd.conf 配置文件中的 ServerAdmin 指令。如果在服务器的出错页面上包含了这个 E-mail 地址作为联系用电子邮件，则发现错误的用户可以向 Apache 服务器的管理员发送电子邮件报告问题。默认地址是 root@localhost。

（3）可用地址。设置 Apache 服务器进程监听客户端时使用的 IP 地址和端口号码。默认会在所有 IP 地址上的 80 连接端口进行监听。这个选项对应于配置文件 httpd.conf 中的 listen 指令。

单击“添加”按钮，弹出如图 8-20 所示的对话框，可以添加其他接受请求的端口。也可以选择监听所有地址上的某个端口，或者选择监听指定地址上的指定端口，不过每个端口只能指定一个 IP 地址。如果要为多个 IP 地址指定相同的监听端口，则必须为每个 IP 地址添加一条监听端口记录。推荐使用 IP 地址而不是域名，避免 DNS 解析失败造成的服务器不能正常工作和访问。如果在地址框中输入“*”，则与选择“监听所有地址”作用相同。

单击“编辑”按钮，弹出选中监听端口记录的编辑窗口，可以修改 IP 地址或者端口设置。单击“删除”按钮，可以删除选中的监听端口记录。如果是 1024 以上的端口作为监听端口，则可以由普通用户启动 HTTP 服务器，否则需以 root 账户启动 HTTP 服务器。

2. “虚拟主机”选项卡

该选项卡是用来管理虚拟主机的一大利器，在窗口中的“虚拟主机”列表里列出了目前 Apache 服务器上建立的所有虚拟主机的名称及地址，如图 8-21 所示。

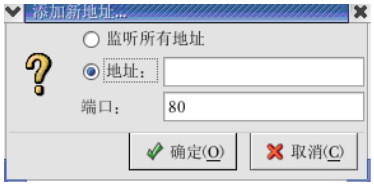

图 8-20　添加新的监听端口

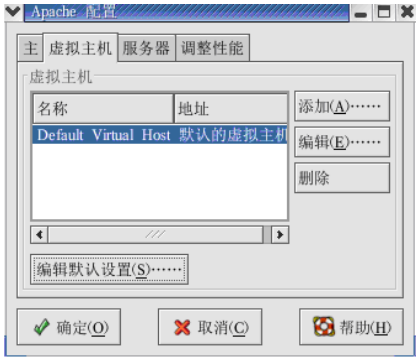

图 8-21　“虚拟主机”选项卡

（1）编辑默认设置。在新建虚拟主机前，必须事先定义虚拟主机的默认内容，而这些设置也会自动套用在此后所有新建的虚拟主机中。如果要编辑虚拟主机的默认内容，首先单击“虚拟主机”选项卡左下角的“编辑默认设置”按钮，则系统会出现“虚拟主机的属胜”窗口，如图 8-22 所示。这也是一个包含多个选项的窗口，包括站点配置、记录日志、环境变量和目录。下面分别说明这些选项的作用。

①　“站点配置”选项卡

在右侧的“目录页搜寻列表”中，可用定义客户端连接到服务器时默认启动的网页，这与 httpd.conf 文件中的“DirectoryIndex”功能相同，可利用右侧的按钮来添加、编辑或删除列表内容项目，如图 8-22 所示。

另一个列表中包含的是网页发生错误时响应客户端的信息，它与 httpd.conf 文件中的

“ErrorDocument”功能相同。如果希望修改错误信息的内容，首先由列表选中错误信息，然后单击“编辑”按钮，系统会出现“ApacheConf.py”对话框，如图 8-23 所示。在“ApacheConf.py”对话框中，可从“行为”下拉列表中选择“默认”（使用原来错误信息）、“文件”（使用指定的文件内容）或“URL”（导向外面的 URL），然后在“位置”字段中输入对应的文件路径。

而最后的“默认错误页页脚”下拉列表，可用来选择错误信息中的页尾内容，可以选择加入默认页尾、管理员电子邮件地址或不显示页尾等。

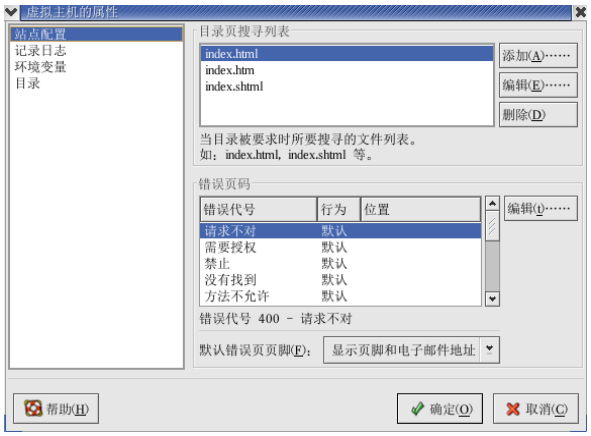

图 8-22　“站点配置”选项卡

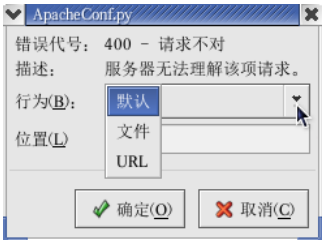

图 8-23　“ApacheConf.py”对话框

② “记录日志”选项卡

“记录日志”选项卡如图 8-24 所示。在右侧的“传输日志”选项组中，可以设置记录客户端访问的方式，这与 httpd.conf 中的“TransferLog”功能相同，可以选择记录到文件、记录到程序或使用系统日志。“使用定制记录设施”选项与 httpd.conf 中的“LogFormat”功能相同，可以将定制的错误格式输入到“定制日志字串”字段中。

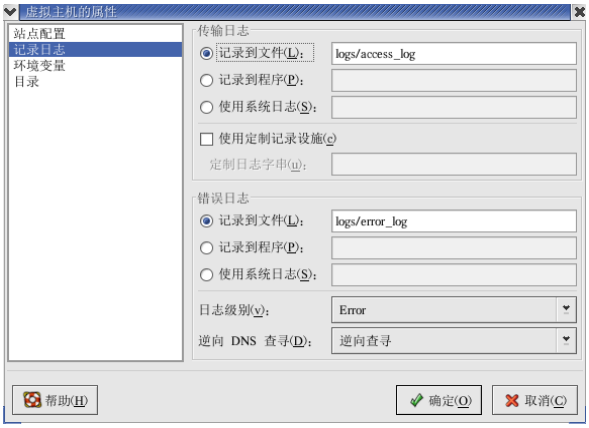

图 8-24　“记录日志”选项卡

在“错误日志”选项组中，可以设置记录错误事件的方式，这个与 httpd.conf 中的“LogLevel”功能相同，可以选择记录到文件、记录到程序或使用系统日志。

“日志级别”下拉列表可以用来选择记录时使用的类型，这与 httpd.conf 文件中的

“TransferLog”功能相同，可以选择的类型有 Emergency、Alert、Critical、Error、Warm、Notice、Info 和 Debug 等。

“逆向 DNS 查寻”下拉列表与 httpd.conf 中的“HostnameLookups”功能相同，可以在此设置是否允许 Apache 服务器在接受客户端请求时，向 DNS 服务器请求反向解析客户端的 IP 地址。

③ “环境变量”选项卡

在右侧“为 CGI 脚本设置”对话框中会列出目前系统已经设置的环境变量，这些环境变量是提供给 CGI 或 SSI 网页使用的，如图 8-25 所示。

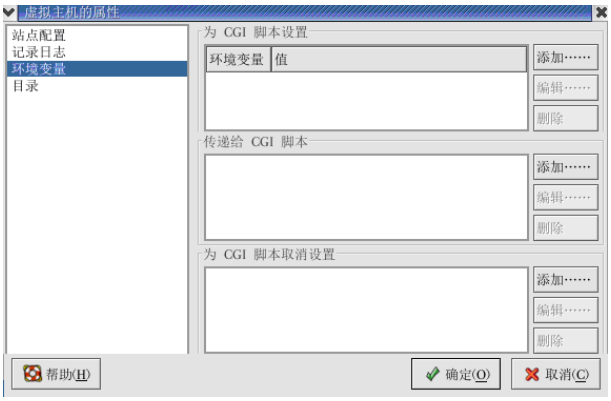

图 8-25 “环境变量”选项卡

如果要新建环境变量，可以单击“新建”按钮，则系统会出现“环境变量”对话框，以供输入环境变量及其要设置的值，这与 httpd.conf 文件中的“SetEnv”功能相同，如图 8-26 所示。

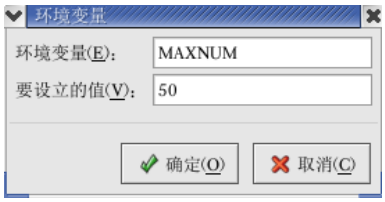

图 8-26 新建环境变量及赋值

“传递给 CGI 脚本”列表中显示的环境变量值，会在 Apache 服务器第一次启动 CGI 程序时传入，它与 httpd.conf 文件中的“PassEnv”功能相同。可以输入“env”指令来查看已传入的环境变量值。

“解除 CGI 程序代码的设置”列表，用来设置要删除的环境变量。这样，这些环境变量便不再提供给 CGI 或 SSI 网页使用，这与 httpd.conf 文件中的“UnsetEnv”功能相同。

④ 目录

这个选项卡的功能同 httpd.conf 文件中的<Directory>…</Directory >区块相同，都是用来设置指定目录的配置内容，如图 8-27 所示。

在窗口上方的“默认目录选项”下，显示的是所有目录的默认选项（但下方列表中的目录选项可以覆盖此处的设置），其默认值为 ExecCGI、FollowSymLinks、Includes、IncludesNOEXEC、Indexes 和 SymLinksIf OwnerMatch。如果要修改默认值，可以单击右侧的

“编辑”按钮，然后在出现的“目录选项”对话框中选择或取消指定的选项，如图 8-28 所示。

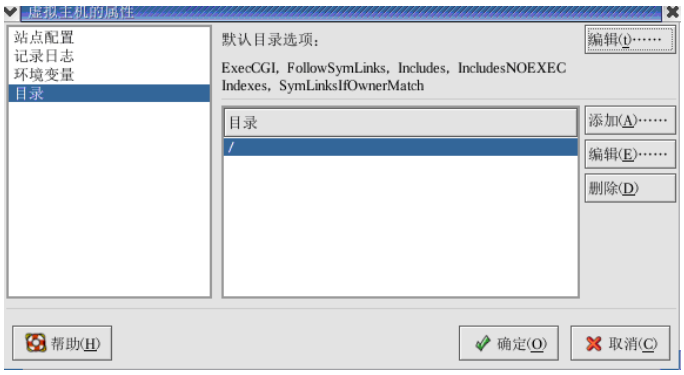

图 8-27　“目录”选项卡

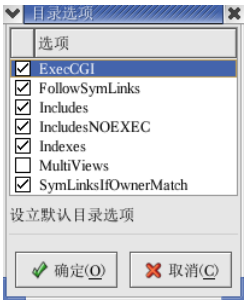

图 8-28　“目录选项”窗口

在“目录”下显示的是定制选项的目录名称，如果要在此处新建目录及选项，可以单击右侧的“添加”按钮，然后系统会出现“目录选项”窗口，如图 8-29 所示。在“目录选项”对话框中，必需设置访问此目录时的顺序、目录名称、目录选项及其他内容。

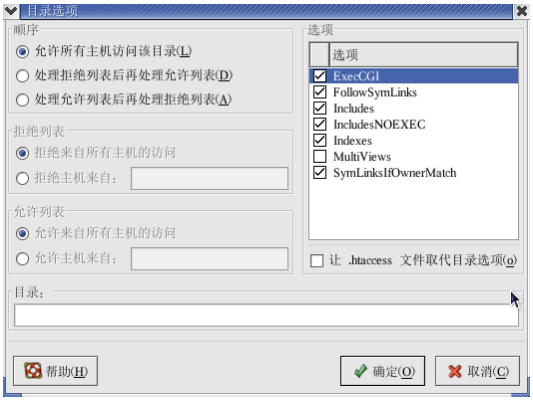

图 8-29　新建目录及选项

（2）虚拟主机设置。“虚拟主机”选项卡中的另一项功能为虚拟主机的设置，在“虚拟主机”列表中显示目前 Apache 服务器上已经建立的虚拟主机，而要新建其他虚拟主机，可以单击窗口右侧的“添加”按钮，则系统会出现“虚拟主机的属性”窗口，如图 8-30 所示。在

出现的“虚拟主机的属性”窗口左侧包含 6 个选项，即常规选项、站点配置、SSL、记录日志、环境变量和目录。本书在此仅介绍“常规选项”和“SSL”选项卡的功能。

① “常规选项”选项

在此选项卡中的设置，只会套用在目前新建的虚拟主机，其中可供设置的内容有：虚拟主机名（ServerName）、文档根目录（DocumentRoot）、网主电子邮件地址（ServerAdmin）与主机信息等，如图 8-30 所示。

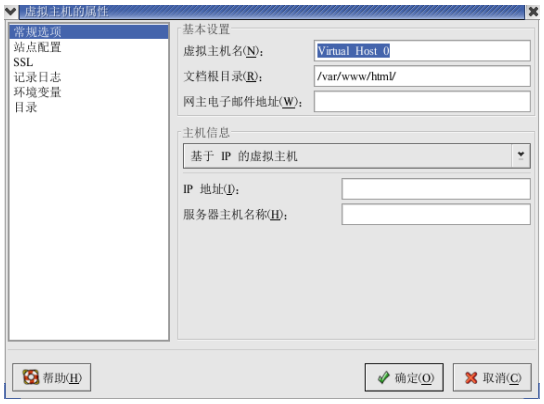

图 8-30 “常规选项”选项卡

值得注意的是，在“主机信息”的下拉列表中，可供选择的选项有“基于 IP 的虚拟主机”、“基于名称的虚拟主机”和“默认虚拟主机”。如果选择“基于 IP 的虚拟主机”选项，则需要在窗口下方输入此虚拟主机使用的 IP 地址（如果使用多个 IP 地址需以空格加以分开）以及实际的主机名。如果选择的是“基于名称的虚拟主机”选项，除了 IP 地址和主机名称外，尚需输入此虚拟主机的别名，如图 8-31 所示。

当客户端请求的 IP 地址并不存在任何虚拟主机时，Apache 服务器会使用“默认虚拟主机”的设置来响应，如果没有设置“默认虚拟主机”，则会由主要服务器接受此请求。如果选择“默认虚拟主机”选项，则只需要指定监听的连接端口号码，或接受所有未指定 IP 地址的请求。注意一点，每台主机上只可建立一台默认虚拟主机，如图 8-32 所示。

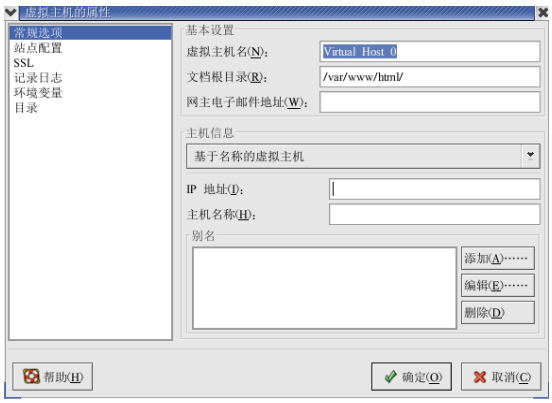

图 8-31 基于名称的虚拟主机

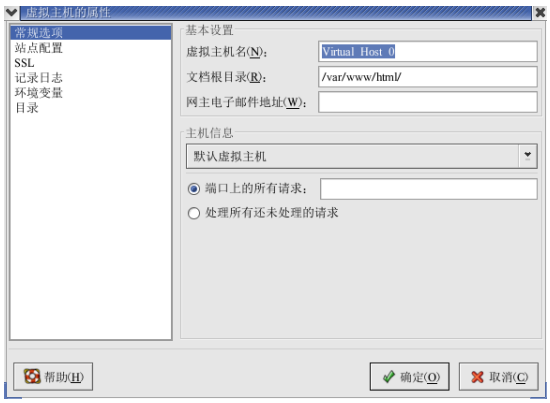

图 8-32 默认虚拟主机

② “SSL”选项卡

如果要在虚拟主机上使用 SSL 安全通信，首先必须选择右侧窗口中的“SSL”选项，然后再设置其他相关文件的路径，如图 8-33 所示。

3. “服务器”选项卡

此处的设置内容比较简单，如图 8-34 所示。可用来设置的项目如下。

（1）锁文件。指定锁文件的路径，与 http.conf 文件中的“LockFile”功能相同，此路径必需为本机路径。

（2）PID 文件。指定 PID 文件的路径，与 http.conf 文件中的“PidFile”功能相同，此路径必需为本机路径。

（3）核心转储目录。指定 Core Dump 目录的路径，与 http.conf 文件中的“Core Dump Directory”功能相同。

当程序运行时遇到一些错误，而操作系统无法处理时，就会产生 Core Dump 事件。然后操作系统会将程序运行到该错误时的所有配置，包括变量值以及程序中的各个函数的调用设置，然后保存到一个文件，这个动作称为 Core Dump。

（4）用户。指定运行 Apache 服务器时的用户账号，与 http.conf 文件中的“User”功能相同。

（5）组群。指定运行 Apache 服务器时的组群账号，与 http.conf 文件中的“Group”功能相同。

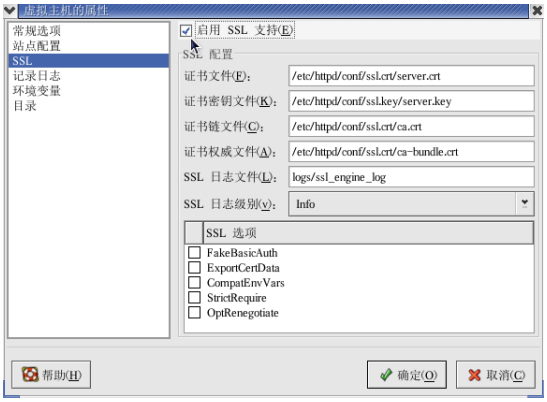

图 8-33　“SSL”配置窗口

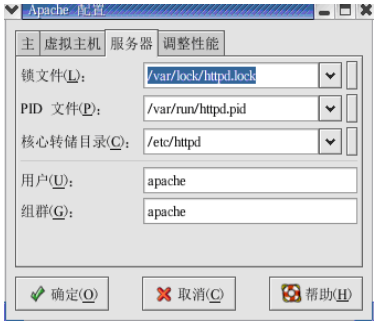

图 8-34　“服务器”选项卡

4. “调整性能”选项卡

“调整性能”选项卡如图 8-35 所示，可用来调整 Apache 服务器运行时的性能，尤其是以带宽为主要的考虑重点。

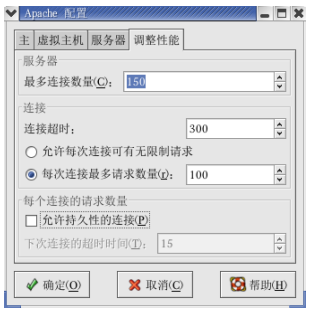

图 8-35　“调整性能”选项卡

8.4 VSFTP 服务器

8.4.1 FTP 概述

FTP 是 TCP/IP 协议组中的协议之一，是 File Transfer Protocol 的缩写。在 Internet 中，大部分的文件传送采用 FTP 协议。FTP 采用“客户机/服务器”的架构，FTP 服务器能在网络上提供文件传输服务，可以供客户端上的用户上传和下载文件；用户需要在本地计算机上安装 FTP 客户端程序。FTP 可以提供跨平台的数据交换，如安装 Linux 和 Windows 操作系统的计算机之间。

FTP 与其他通信协议最大的不同是，它使用两个连接端口来和客户端连接：TCP 20 和 TCP 21。其中，连接端口 TCP 20 用来传递数据，而 TCP 21 用来负责传输流程的控制。这种设计可以支持多个客户端同时连接 FTP 服务器，并且具有稳定的优点。

一般来说，用户在访问 FTP 服务器的资源前需要先经过认证，FTP 服务器会要求用户输入用户名及密码。如果管理员允许匿名进入，用户需要输入“anonymous”作为用户名，而密码则不进行验证。

目前在 Linux 中常用的免费 FTP 服务器软件主要是 Wu-FTP、ProFTP、VSFTP 三种。其中，Wu-FTP 使用最广泛。在 Red Hat Linux 7.2 中，Wu-FTP 是默认的 FTP 服务器软件。Red Hat Linux 8.0 中同时自带 Wu-FTP 和 VSFTP。可能由于 Wu-FTP 被发现安全漏洞比较多的原因，到了 Red Hat Linux 9，只有 VSFTP 了。VSFTP 是 Very Secure FTP 的缩写，意思是“非常安全的 FTP”。如果需要使用 Wu-FTP 或 ProFTP，用户也可以自己安装，配置的方法和 VSFTP 接近。

8.4.2 VSFTP 服务器的安装

Red Hat Linux 9 提供的 FTP 服务器软件为 vsftpd-1.1.3-8.i386.rpm。建立 FTP 服务器需要首先安装该软件，可以使用以下命令检查该软件是否已经安装。

```
[root@rh9 root]# rpm   -qa |grep   vsftp
vsftpd-1.1.3-8                    //该软件包已经安装。
```

由于 Red Hat Linux 9 自带 VSFTP 服务器软件，通常不需要另行安装。如果在安装系统时没有安装，可以在图形界面的“主菜单”中选择“系统设置”→“添加删除应用程序”选项，在出现的“软件包管理”对话框里面选中“FTP 服务器”选项，然后单击“更新”按钮，安装屏幕提示插入第 2 张安装光盘即可开始安装。用户也可以用 rpm 命令来安装。

```
[root@rh9 dhcp]# mount   /mnt/cdrom
[root@rh9 dhcp]# cd   /mnt/cdrom/Red Hat/RPMS
[root@rh9 root]# rpm   -ivh   vsftpd-1.1.3-8.i386.rpm
```

8.4.3 VSFTP 服务器的配置文件

VSFTP 的配置文件主要有 3 个，分别是/etc/vsftpd/vsftpd.conf 和/etc/vsftpd.ftpuser、/etc/vsftpd.user_list。这些文件中的每一行都表示单一的注释或指令，如果以“#”开头表示该行是注释行，同时会被服务器忽略。除注释以外的都属于指令，采用“选项=设置值”的格式，

每条指令都有一个默认值，可以在配置 VSFTP 服务器的时候根据具体的用户需要进行修改。但要注意的是，只有重启 VSFTP 服务器进程（vsftpd）后，修改后的配置才会生效。

1. /etc/vsftpd/vsftpd.conf

该文件是 VSFTP 最主要的配置文件，需要对它进行配置以发挥 VSFTP 服务器的最大功能与安全性，该文件的默认内容及说明如下。

```
[root@rh9 root]# more    /etc/vsftpd.conf
# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are very paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
#
# Allow anonymous FTP?
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
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
#anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
#anon_mkdir_write_enable=YES
#
# Activate directory messages - messages given to remote users when they
# go into a certain directory.
dirmessage_enable=YES
#
# Activate logging of uploads/downloads.
xferlog_enable=YES
#
# Make sure PORT transfer connections originate from port 20 (ftp-data).
connect_from_port_20=YES
#
# If you want, you can arrange for uploaded anonymous files to be owned by
# a different user. Note! Using "root" for uploaded files is not
# recommended!
#chown_uploads=YES
#chown_username=whoever
#
```

```
# You may override where the log file goes if you like. The default is shown
# below.
#xferlog_file=/var/log/vsftpd.log
#
# If you want, you can have your log file in standard ftpd xferlog format
xferlog_std_format=YES
#
# You may change the default value for timing out an idle session.
#idle_session_timeout=600
#
# You may change the default value for timing out a data connection.
#data_connection_timeout=120
#
# It is recommended that you define on your system a unique user which the
# ftp server can use as a totally isolated and unprivileged user.
#nopriv_user=ftpsecure
#
# Enable this and the server will recognise asynchronous ABOR requests. Not
# recommended for security (the code is non-trivial). Not enabling it,
# however, may confuse older FTP clients.
#async_abor_enable=YES
#
# By default the server will pretend to allow ASCII mode but in fact ignore
# the request. Turn on the below options to have the server actually do ASCII
# mangling on files when in ASCII mode.
# Beware that turning on ascii_download_enable enables malicious remote parties
# to consume your I/O resources, by issuing the command "SIZE /big/file" in
# ASCII mode.
# These ASCII options are split into upload and download because you may wish
# to enable ASCII uploads (to prevent uploaded scripts etc. from breaking),
# without the DoS risk of SIZE and ASCII downloads. ASCII mangling should be
# on the client anyway..
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
#ftpd_banner=Welcome to blah FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
#
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
#chroot_list_enable=YES
# (default follows)
#chroot_list_file=/etc/vsftpd.chroot_list
```

```
#
# You may activate the "-R" option to the builtin ls. This is disabled by
# default to avoid remote users being able to cause excessive I/O on large
# sites. However, some broken FTP clients such as "ncftp" and "mirror" assume
# the presence of the "-R" option, so there is a strong case for enabling it.
#ls_recurse_enable=YES

pam_service_name=vsftpd
userlist_enable=YES
#enable for standalone mode
listen=YES
tcp_wrappers=YES
```

下面对该文件中的一些主要配置项进行说明。

（1）anonymous_enable=YES

是否允许匿名登录 FTP（YES/NO）服务器，默认值是 YES。

（2）local_enable=YES

若启用该功能，则允许本机使用者登录，默认值是 YES。

（3）write_enable=YES

是否允许客户端使用修改 FTP 服务器文件系统的 FTP 指令，如 MKD、DELE 等。默认值是 YES，但从安全控制的角度建议设置为 NO。

（4）local_umask=022

本机登录者新增文档时的 umask 数值。umask 是通过八进制的数值来定义用户创建文件或目录的默认权限，umask 表示的是禁止权限，其默认值为 077，这里设置为 022（大多 ftpd 都设置为 022）。

（5）anon_upload_enable=YES

取值为 YES/NO。设置是否允许匿名用户上传文档的权限。需要先为 FTP 用户创建一个可写的目录。

（6）anon_mkdir_write_enable=YES

取值为 YES/NO。设置是否允许匿名用户拥有创建新目录的权限。一般不建议开放该权限。

（7）dirmessage_enable=YES

取值为 YES/NO。如果启用该选项，当远程用户访问一个指定的目录时，系统将检查该目录下是否有.message 文件。如果有，显示该文件内容。通常这个文件放置欢迎词或该项目录的说明。

（8）connect_from_port_20=YES

取值为 YES/NO。若为 YES，表示所有 FTP 传递的数据（不是 FTP 的控制信息）都会通过 20 号端口来进行。默认会使用该选项。

（9）chown_uploads=YES

取值为 YES/NO。若设置为 YES，所有匿名上传数据的所有者被更换为 chown_username 中设定的用户。这样的设置对于 FTP 的安全及管理很有用。

（10）chown_username=whoever

定义匿名登录者上传文档时，这些文档的所有者将被置换成的用户名称。

（11）xferlog_file=/var/log/vsftpd.log

定义日志文件（log file）的存放位置。

（12）xferlog_std_format=YES

将日志文件定义为标准 ftpd xferlog 的格式。

（13）idle_session_timeout=600

空闲时间超时设定，单位为秒。如果超出该时间还没有数据传送或指令输入，则连接中断。

（14）data_connection_timeout=120

数据连接的超时设定，单位为秒。

（15）nopriv_user=ftpsecure

定义运行 vsftd 进程的独立而非特权的系统用户。

（16）async_abor_enable=YES

取值为 YES/NO。若设置为 YES，FTP 服务器将认可异步 ABOR 请求。一般不推荐。

（17）ascii_upload_enable=YES

取值为 YES/NO。控制是否可用 ASCII 模式上传。

（18）ascii_download_enable=YES

取值为 YES/NO。控制是否可用 ASCII 模式下载。

（19）ftpd_banner=Welcome to blah FTP service.

定制登录欢迎词。

（20）deny_email_enable=YES

若启动该项功能，可以指定一个文件/etc/vsftpd.banner_email，其中包含电子邮件地址列表。若用户用匿名登录，系统将要求输入邮件地址；如果输入的邮件地址在该项文档中，则不允许链接。该项功能主要用于防范 DoS 攻击。

（21）banned_email_file=/etc/vsftpd.banned_emails

设置文件 vsftpd.banned_emails 的存放位置。

（22）chroot_list_enable=YES

取值为 YES/NO。如果启动该项功能，所有本机使用者登录均可进到根目录之外的目录，列在/etc/vsftpd.chroot_list 中的使用者除外。

（23）chroot_list_file=/etc/vsftpd.chroot_list

设置文件 vsftpd.chroot_list 的存放位置。

（24）ls_recurse_enable=YES

取值为 YES/NO。若启动该项功能，允许登录用户使用 ls_R 指令。

（25）pam_service_name=vsftpd

定义 PAM 使用的名称。

（26）userlist_enable=YES

取值为 YES/NO。如果设置为“YES”，则/etc/vsftpd.user_list 文件会生效。

（27）userlist_deny=YES

取值为 YES/NO。该项只有在 userlist_enable 启动时才有效。如果将该选项设置为 YES，则在/etc/vsftpd.user_list 中的用户无法登录；若设置为 NO，则只有在/etc/vsftpd.user_list 中的用户才能登录。

（28）listen=YES

取值为 YES/NO。设置 FTP 服务器是否工作在 standalone 模式。若是工作在 xinetd 模式下，必须设置为 NO。

（29）tcp_wrappers=YES

如果使用该选项（YES），会将 vsftpd 与 wrapper 相结合，这样可以利用/etc/hosts.allow 与/etc/hosts.deny 文件来定义允许或拒绝的来源地址。默认不使用该项。

（30）max_clients = 0

如果 vsftpd 在 standalone 模式下启动，该处的设置表示可同时连接的最大客户端数量，其后的任何客户端尝试连接时都会收到错误信息，“0”表示不受限制。

```
[lihh@rh9 lihh]$ ftp    host1.cqcet.cn
Connected to host1.cqcet.cn.
421 There are too many connected users, please try later.
```

（31）max_per_ip= 0

如果 vsftpd 在 standalone 模式下启动，该处的设置表示在同一个 IP 地址上可以进行的连接上限，其后任何的连接都会收到错误的信息，“0”表示不受限制。

（32）local_max_rate= 0

表示本地已验证的用户进行连接时，允许的最大数据传输速率，单位是 B/s，默认为 0（无限制）。

2. /etc/vsftpd.ftpuser

该配置文件记录了不允许访问 FTP 服务器的用户名单，管理员可以把一些对系统安全有可能造成威胁的用户账号记录到这个文件中，以免这些用户账号被用于登录系统后获得大于文件上传和下载操作的权利，从而对系统造成破坏。

下面是该配置文件的初始内容，可以看到默认情况下 root 是被禁止用来登录 FTP 服务器的。如果想让 root 可以登录，只需要把 root 删除，或者在 root 的前面加上“#”号，然后重新启动 FTP 服务器就可以了。

```
[root@rh9 root]# more    /etc/vsftpd.ftpuser
# Users that are not allowed to login via ftp
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
```

3. /etc/vsftpd.user_list

该配置文件只有在主配置文件/etc/vsftpd/vsftpd.conf 中的 userlist_enable 被设置为 YES 时

才起作用。同时，系统会根据/etc/vsftpd/vsftpd.conf 文件中的 userlist_deny 为 Yes 还是 No 来决定是禁止/etc/vsftpd.user_list 中的用户登录，还是仅仅允许该文件中的用户登录。

下面是/etc/vsftpd.user_list 的初始内容，默认情况下里面列举出来的用户是被禁止登录 FTP 服务器的，可以看到 root 也是被禁止的。

```
[root@rh9 root]# more   /etc/vsftpd.user_list
#vsftpd userlist
# If userlist_deny=NO, only allow users in this file
# If userlist_deny=YES （default）, never allow users in this file, and
# do not even prompt for a password.
# Note that the default vsftpd pam config also checks /etc/vsftpd.ftpusers
# for users that are denied.
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
```

8.4.4 VSFTP 服务器的启停

VSFTP 服务器也可以像前面介绍的其他服务器一样，采用同样的几种方法进行启动、关闭，以及设置为自动启动等操作。下面仅以 service 和 chkconfig 命令来说明。

1. 使用 service 命令

VSFTP 服务器的启动、停止等操作可以通过 service 命令来实现。

```
[root@rh9 root]# service   vsftpd   start
为 vsftpd 启动 vsftpd：                                    [   确定   ]
[root@rh9 root]# service   vsftpd   stop
关闭 vsftpd：                                              [   确定   ]
[root@rh9 root]# service   vsftpd   status
vsftpd 已停
```

2. 使用 chkconfig 命令

若要系统每次启动时自动开启 vsftpd 服务，可以使用如下命令，下面的例子表示在系统进入第 3 和第 5 个级别时自动开启 vsftpd 服务。

```
[root@rh9 root]# chkconfig   --level   35   vsftpd   on
[root@rh9 root]# chkconfig   --list   vsftpd
vsftpd     0:关闭   1:关闭   2:关闭   3:启用   4:关闭   5:启用   6:关闭
```

8.4.5 访问 VSFTP 服务器

在成功安装以及启动 VSFTP 服务器后，客户端可以进行服务器的访问，常用的 FTP 客户

端工具有很多，下面仅介绍通过浏览器访问 FTP 服务器的方法。

浏览器除了浏览网页的功能外，也是连接 FTP 服务器最方便的客户端工具之一。使用浏览器连接 FTP 服务器和连接 Web 服务器的方式很相似，唯一不同是使用的协议 ftp 而非 http。

1. 匿名访问

默认情况下，安装好的 VSFTP 服务器就可以使用匿名用户（anonymous 或 ftp）浏览和下载 FTP 服务器发表目录（默认是/var/ftp）下的文件。匿名访问需要在地址栏中输入以下格式的信息：“ftp://服务器的 IP 或域名”，如图 8-36 所示。

2. 指定用户名访问

默认情况下，安装好的 VSFTP 服务器也允许本机用户（如 wangxi）浏览和下载自己主目录下的文件（lihh 的主目录是/home/wangxi），如图 8-37 所示。指定用户名访问需要在地址栏中输入以下格式的信息：“ftp://用户名@服务器的 IP 或域名”。

首次访问时，会弹出如图 8-38 所示的对话框，提示用户输入密码，以验证用户的身份。通过认证后，浏览器中会显示该用户主目录下的文件。

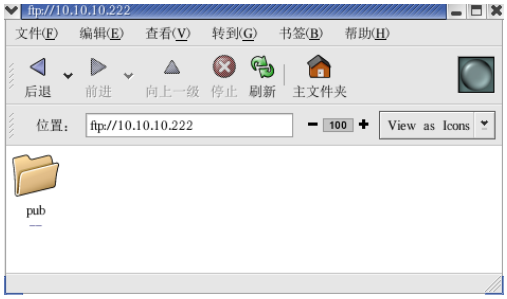

图 8-36　匿名访问 VSFTP

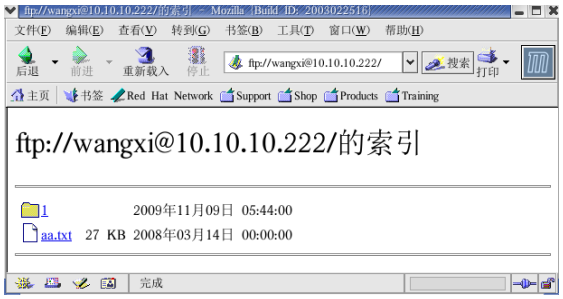

图 8-37　指定用户名访问 VSFTP

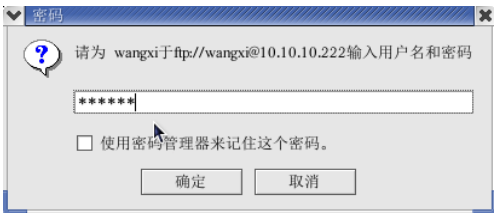

图 8-38　提示输入密码

8.5　DNS 服务器

8.5.1　DNS 概述

在 Internet 上浏览网站时，使用的大都是便于用户记忆的称之为主机名的友好名字。例如，搜狐的主机名为 www.sohu.com，用户在访问搜狐的时候一般用 www.sohu.com 访问，而很少有人使用其 IP 地址去访问。用户计算机使用 www.sohu.com 访问时，要先设法找到该服务器相应的 IP 地址，客户与服务器之间仍然是通过 IP 地址进行连接的。用于存储该 Web 域名和 IP 地址并接受客户查询的计算机，称为 DNS 服务器。

DNS 是 Internet 和 TCP/IP 网络中广泛使用的、用于提供名字登记和名字到地址转换的一组协议和服务。DNS 服务免除了用户记忆枯燥的 IP 地址的烦恼，可以使用具有层次结构的“友好”的名字来定位本地 TCP/IP 网络和 Internet 上的主机及其他资源。

DNS 通过分布式名字数据库系统，为管理大规模网络中的主机名和相关信息提供了一种可靠的方法。DNS 的命名系统是一种叫做域名空间（Domain Name space）的层次性的逻辑树形结构，其犹如一棵倒立的树，树根在最上面。域名空间的根由 Internet 域名管理机构 InterNIC 负责管理。InterNIC 负责划分数据库的名字信息，使用名字服务器（DNS 服务器）来管理域名，每个 DNS 服务器中有一个数据库文件，其中包含了域名树中某个区域的记录信息。Internet 将所有联网主机的名字空间划分为许多不同的域。树根（也称根域）下是顶级域（或称一级），再往下是二级、三级域。

DNS 域名是按组织来划分的，Internet 中最初规定的一级域名有 7 个，其中 com 代表商业机构，edu 代表教育机构，mil 代表军事机构，gov 代表政府部门，net 代表提供网络服务的部门，org 代表非商业机构，xx 代表国家或者地区。此外，ICANN 还在 2000 年新增了 7 个域名，分别是 info（提供信息服务的单位）、biz（公司）、name（个人）、Pro（专业人士）、museum（博物馆）、coop（商业合作机构）和 aero（航空业）。

一般情况下，域名可以向提供域名注册服务的网站进行在线申请。例如，可以到中国互联网络信息中心（CNNIC）的网站 http://www.cnnic.net.cn 查看并注册域名。企业如果需要部署自己的 DNS 服务器、需要安装 Active Directory，或希望 Internet 用户对企业内部计算机进行访问时，必须架设 DNS 服务器。

DNS 客户端向 DNS 服务器提出查询，DNS 服务器作出响应的过程称为域名解析。DNS 域名的解析方式有以下两种。

（1）正向查询。正向查询就是根据域名，搜索出对应的 IP 地址，其查询方法为：当 DNS 客户机（也可以是 DNS 服务器）向首选 DNS 服务器发出查询请求后，如果首选 DNS 服务器中不含所需的数据，则会将查询请求转发给另一台 DNS 服务器，依此类推，一直找到所需的数据为止，如果到最后一台 DNS 服务器中也没有所需的数据，则通知 DNS 客户机查询失败。

（2）反向查询。反向查询与正向查询刚好相反，它是依据 DNS 客户端提供的 IP 地址，来查询该 IP 地址对应的主机域名。实现反向查询，必须在 DNS 服务器内创建一个反向查询的区域。一旦创建的区域进入 DNS 数据库中，就会增加一个指针记录，将 IP 地址与相应的主机名相关联。换句话说，当查询 IP 地址为 192.168.1.1 的主机名时，解析程序将向 DNS 服务器查询 1.1.168.192.in-addr.arpa 的指针记录。如果该 IP 地址在本地域之外，DNS 服务器将从根开始顺序地解析节点，直到找到 1.1.168.192.in-addr.arpa。当创建反向查询区域时，系统会自动为其创建一个反向查询区域文件。

8.5.2 DNS 服务器的安装

Red Hat Linux 9 自带有版本号为 9.2.1 的 BIND（Berkeley Internet Name Domain）服务器软件，它是目前使用最广泛的域名服务器软件，使用 named 守护进程提供域名解析服务。BIND 软件是由 Nominum 公司开发，由 ISC（Internet Software Consortium）负责维护，其最新软件包采用源代码方式发布，可访问 http://www.isc.org/products/blnd/网站下载。

第 1 张安装光盘提供了 BIND 服务的主要安装软件包 bind-9.2.1-16.1386.rpm，另一部分与 DNS 相关的实用程序（dig、host、nslookup 和 nsupdate）由 bind-utils-9.2.1-16.i386.rpm 软件包提供。BIND 的主配置文件（named.conf）、本地域文件和根域文件（named.ca）等重要配置文件，则由第 2 张安装光盘中的 caching-nameserver-7.2-7.noarch.rpm 软件包提供。在配置 DNS 服务器之前，应首先检查当前 Linux 系统是否安装了 BIND 软件包。

```
[root@rh9 root]# rpm    -qa |grep    bind
bind-9.2.16-16
bind-utils-9.2.1-16
```

通过输出的 bind-9.2.16-16，说明当前系统已安装了该软件包。否则，可以使用如下命令（假定这些软件已经存放在/root 目录内）进行安装。

```
[root@rh9 root]# rpm    -ivh    bind/bind-utils-9.2.1-16.i386.rpm
warning:    bind-utils-9.2.1-16.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
Preparing...                ########################################### [100%]
    1:bind-utils            ########################################### [100%]
[root@rh9 root]# rpm    -ivh      /bind-9.2.1-16.i386.rpm
warning: bind-9.2.1-16.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
Preparing...                ########################################### [100%]
    1:bind                  ########################################### [100%]
[root@rh9 root]# rpm    -ivh    caching-nameserver-7.2-7.noarch.rpm
warning: caching-nameserver-7.2-7.noarch.rpm: V3 DSA signature:
NOKEY, key ID db42a60e
Preparing...                 ########################################### [100%]
    1:caching-nameserver     ########################################### [100%]
[root@rh9 root]# rpm    -ivh    redhat-config-bind-1.9.0-13.noarch.rpm
warning: redhat-config-bind-1.9.0-13.noarch.rpm: V3 DSA signature: NOKEY, key ID db42a60e
Preparing...                ########################################### [100%]
    1:redhat-config-bind    ########################################### [100%]
```

BIND 软件包安装后，系统将创建名为 named 的用户和用户组，并自动设置相关目录的权属关系。named 守护进程默认使用 named 用户身份运行。

```
[root@rh9 root]# grep    named /etc/passwd
named:x:25:25:Named:/var/named:/sbin/nologin
[root@rh9 root]# grep    named /etc/group
named:x:25:
```

如果是利用源代码安装，还应该手工创建 named 用户和用户组，并设置好工作目录（/var/named）和用于存放进程号文件的目录（/var/run/named）的所有者和权限设置。

```
[root@rh9 root]# ll /var    |grep named
drwxr-xr-x       2 named       named             4096 11 月  10 06:47 named
[root@rh9 root]# ll /var/run/    |grep named
drwxr-xr-x       2 named       named             4096 11 月  10 07:00 named
```

BIND 的主配置文件为 named.conf，默认位置在/etc 目录，默认的工作目录为/var/named，相关的主要实用程序默认安装在/usr/sbin 目录中。bind-utils-9.2.1-16.i386.rpm 软件包提供的实用程序默认安装在/usr/bin 目录中。

检查 BIND 的样本配置文件和区文件是否安装。检查/etc/目录下是否有 named.conf 文件，

/var/named 目录下面是否有本地域文件和根域文件，若都没有则说明软件包 caching-nameserver-7.2-7.noarch.rpm 未安装。对于一个新的软件包，若不知其安装位置，可通过查询其安装文件的列表来获知。

8.5.3 DNS 服务器的配置文件

BIND 的配置文件包括主配置文件/etc/named.conf、根域文件（named.ca）、区域文件以及用于管理 BIND 守护进程的 rndc 程序的配置文件/etc/rndc.conf。根域文件 named.ca 用于提供位于顶层根域名服务器的列表。

named.ca、rndc.conf 和用于保存密钥的 rndc.key 文件，在安装好 BIND 软件包后就已提供，用户一般不需要编辑或修改，配置 DNS 服务器主要是对 named.conf 和区域文件进行编辑和修改。

1. /etc/named.conf

/etc/named.conf 是 BIND 中最重要的配置文件，在该文件中除了设置的一些参数外，同时也会指出该服务器管辖的区域域名和相关文件的存放位置。

该配置文件为文本文件，由若干配置段构成，每个配置段由若干语句构成，语句以分号作为结束符，行注释可使用“#”或“//”，对一段文字的注释采用/* … */。下面是该文件的内容以及说明。

```
[root@rh9 root]# more    /etc/named.conf
# named.conf - configuration for bind
#
## Generated automatically by redhat-config-bind, alchemist et al.
# Any changes not supported by redhat-config-bind should be put
# in /etc/named.custom
#
controls {
            inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
//controls 类型的记录，其中的“inet”表示利用 TCP/IP Socket 来访问 Internet 资源，它是由指定的 IP 地址和端口号所产生，而此处表示可允许本机（localhost）利用 rndckey 进行访问。

include "/etc/named.custom"
//表示将文件/etc/named.custom 包含到该文件中。

include "/etc/rndc.key";
//表示将文件/etc/rndc.key 包含到该文件中。

zone    "0.0.127.in-addr.arpa" {
        type master;
        file    "0.0.127.in-addr.arpa.zone";
        allow-update{none; }
};
//zone 类型的记录，用来定义一个 DNS 区域。其中，“0.0.127.in-addr.arpa”表示此区域是用来定义本机所在的网域的反向解析内容，而它是属于“IN”（Internet）的区域类别（Class），同时此区域中服务器的种类是“master”，使用 0.0.127.in-addr.arpa.zone 文件记录此区域的名称记录。“allow-update{none; }”表示不允许客户端或服务器自己更新此 DNS 记录。
```

```
zone    "localhost"    IN{
        type master;
        file    "localhost.zone";
allow-update{none; }
};
//zone 类型的记录，用来定义一个 DNS 区域。其中，“localhost”表示此区域是本机所在的网域，而它是属于“IN”（Internet）的区域类别（Class），同时此区域中服务器的种类是“master”，使用 localhost.zone 文件记录此区域的名称记录。“allow-update{none; }”表示不允许客户端或服务器自己更新此 DNS 记录。


};
```

2. /etc/named.custom

配置 DNS 服务器一般需要手工修改配置文件，在 Red Hat Linux 9 中，也提供了图形配置工具（redhat-config-bind），使用该工具配置完 BIND 后系统会自动修改相应的配置文件。在该配置文件中的相应选项不能够通过图形界面下的配置工具进行修改。

```
[root@rh9 root]# more    /etc/named.custom
## named.custom - custom configuration for bind
#
# Any changes not currently supported by redhat-config-bind should be put
# in this file.
#

zone    "." IN {
        type hint;
        file    "named.ca";
};
options {
            directory "/var/named/";
};
```

其中，zone 类型的记录用来定义一个 DNS 区域。其中，“.”表示此区域是根域（root），而它是属于“IN”（Internet）的区域类别（Class），同时此区域的类型是“hint”。使用 named.ca 文件记录此区域的名称记录，named.ca 文件中记录着 13 个根服务器的地址。扩展名“ca”表示这是 Cache 文件，也就是说系统启动时必须将此文件加入缓存中，以提高访问时的效率。

options 类型的记录，用来定义这台 DNS 服务器中的通用选项。“directory "/var/named/"”表示将 DNS 服务器的资源记录保存在/var/named 目录下，包括 DNS 日志记录等。

3. /etc/rndc.key

这个文件是 BIND 9.X 版的新功能，它可以用来进行区域转移或 DNS 更新时的加密处理。以下是该文件的默认内容。

```
[root@rh9 root]# more    /etc/rndc.key
key "rndckey" {
            algorithm            hmac-md5;
            secret "A6pglMPu7IhxspmoUcOfChpTV274wCzqOFxJlvhIyHPpR10EidcGdst9tLmp";
};
```

4. /var/named/named.ca

在 DNS 的域名解析流程中，如果 DNS 服务器的数据库中没有包含请求的记录，那么此服务器就会通过 Internet 的根服务器进行逐级查询。而根服务器的相关信息就记录在 named.ca

中，因此它是 DNS 中一个很重要的文件。

根服务器的地址一般不会发生多大的变化，但部分根服务器地址发生变化是有可能的，另外也可能添加了新的根服务器，因此，更新根服务器的最新地址列表对 DNS 服务器的正常解析是很有必要的，可以到ftp://rs.internic.net/domain/named.root下载最新的文件，改名为named.ca 并复制到/var/named/目录下。以下是 Red Hat Linux 9 中该文件的默认内容。

```
[root@rh9 root]# more    /var/named/named.ca
;           This file holds the information on root name servers needed to
;           initialize cache of Internet domain name servers
;           （e.g. reference this file in the "cache    .    <file>"
;           configuration file of BIND domain name servers）.
;
;           This file is made available by InterNIC
;           under anonymous FTP as
;                file                        /domain/named.cache
;                on server                FTP.INTERNIC.NET
;
;           last update:       Nov 5, 2002
;           related version of root zone:      2002110501
;
;
; formerly NS.INTERNIC.NET
;
.                                3600000    IN    NS       A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.          3600000          A        198.41.0.4
;
; formerly NS1.ISI.EDU
;
.                                3600000          NS       B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.          3600000          A        128.9.0.107
;
; formerly C.PSI.NET
;
.                                3600000          NS       C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET.          3600000          A        192.33.4.12
;
; formerly TERP.UMD.EDU
;
.                                3600000          NS       D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET.          3600000          A        128.8.10.90
;
; formerly NS.NASA.GOV
;
.                                3600000          NS       E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET.          3600000          A        192.203.230.10
;
; formerly NS.ISC.ORG
;
.                                3600000          NS       F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET.          3600000          A        192.5.5.241
```

```
;
; formerly NS.NIC.DDN.MIL
;
.                        3600000      NS    G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET.      3600000      A     192.112.36.4
;
; formerly AOS.ARL.ARMY.MIL
;
.                        3600000      NS    H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET.      3600000      A     128.63.2.53
;
; formerly NIC.NORDU.NET
;
.                        3600000      NS    I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET.      3600000      A     192.36.148.17
;
; operated by VeriSign, Inc.
;
.                        3600000      NS    J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET.      3600000      A     192.58.128.30
;
; housed in LINX, operated by RIPE NCC
;
.                        3600000      NS    K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET.      3600000      A     193.0.14.129
;
; operated by IANA
;
.                        3600000      NS    L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET.      3600000      A     198.32.64.12
;
; housed in Japan, operated by WIDE
;
.                        3600000      NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.      3600000      A     202.12.27.33
; End of File
```

该文件中，如果行首出现“;”符号，表示该行是注释行，这和 named.conf 文件中使用的“//”不同。除了“;”符号开头的之外，这个文件中还包含“.”为行首的设置内容，而这些就是属于根服务器的记录。每一条此类记录都以 4 个字段组成。下面以“3600000　IN　NS　A.ROOT-SERVERS.NET.”为例说明。

（1）3600000。表示 TTL，也就是说此记录在缓存中停留的时间为 3600000 秒，相当于 1000 个小时。

（2）IN。表示此服务器记录为 Internet 的区域类型。

（3）NS。是 Name Server 的缩写，它是 DNS 资源记录的一种。

（4）A.ROOT-SERVERS.NET.。是根网域 DNS 服务器的主机名称。

虽然我们利用“NS”来指定这些根服务器的主机名（同时也表示它们都是域名服务器），但是其他的 DNS 服务器必须知道它们的 IP 地址才可以与它们进行沟通。为了解决这个问题，

这些记录都会紧接着另一种记录，例如，"A.ROOT-SERVERS.NET.　3600000 A　198.41.0.4"，它是用来记录此 ROOT 服务器域名和 IP 地址的对应，而此类的记录称为 A（Address）资源记录。因为在整个 DNS 域名解析的流程中，第一个步骤就是连接到根服务器，所以这些记录都必须加入缓存中，让 DNS 服务器在开机后就可使用这些记录的内容。

5. /var/named/localhost.zone

正向解析用于实现域名到 IP 地址的转换。实现主机名 localhost 到 IP 地址正向解析的文件为 localhost.zone，该文件不需要修改，可以直接使用。

下面以该文件为例，介绍正向解析文件的格式和各个部分的含义。使用命令 more 可以查看该文件的内容。

```
[root@rh9 root]# more    /var/named/localhost.zone
$TTL 86400
$ORIGIN localhost.
@     1D    IN    SOA @    root.localhost   （
                    1 ; serial
                    28800 ; refresh
                    7200 ; retry
                    604800 ; expire
                    86400 ; ttl
                    ）
      1D    IN    NS    localhost.
      1D    IN    A     127.0.0.1
```

下面依次介绍各行的含义：

（1）$TTL 86400。用来设置域的默认生存时间 TTL（time to live），其默认值为 86400 秒（24 小时），如果缺少这条记录，在开机时会出现警告信息。在记录日志中的所有资源记录都需要设置 TTL 值，如果某个资源记录没有设置 TTL 时间，就会以此处的设置为默认值。

（2）$ORIGIN localhost.。用来指出这个文件中的记录适用于哪个区域，而此处代表的区域为“1ocalhost.”，需格外留意，在 localhost 后包含一个小数点（.），它在 DNS 中表示域名称，也就是 FQDN。如果此处没有设置 ORIGIN，系统会以/etc/named.conf 定义的区域名称为设置值。

（3）@　1D　IN　SOA@　root.localhost　（

```
                    1 ; serial
                    28800 ; refresh
                    7200 ; retry
                    604800 ; expire
                    86400 ; ttl
                    ）
```

① @代表当前的域。

② 1D 是指 TTL 时间的设置，1D 表示一天（Day），3H 代表 3 个小时（hour），15M 代表 15 分钟（Minute），1W 代表 1 周（Week）。如果没有设置该值，就会套用以上定义的“$TTL”设置，同时也可以为每个资源记录设置个别的 TTL 时间。

③ 在 TTL 后的“IN”表示目前的记录类型是属于 Internet 类别。

④ IN 后就是这行资源记录的类别名称，也就是 SOA。所谓 SOA 是指“Start Of Authority”

的缩写，它是每一个标准区域中的第一条记录，而且在每个区域文件中都必须存在一个唯一的 SOA 记录。

⑤ 在 SOA 后面的内容，是指此区域的授权主机和管理者电子邮箱。此处的“@”和“root”分别表示“localhost.”主机和 root 用户邮箱。

注意：在 DNS 记录中的“@”属于保留字，它代表本机，因此如果要表示原来的电子邮件地址，必须用“.”代替“@”，比如原来的“root@localhost.”必须写成“root.localhost.”。

⑥ 括号中的选项表示 SOA 的设置内容，这些内容会在 Master 和 Slave DNS 服务器之间复制。其中部分选项的说明如表 8.5 所示。

表 8.5　选项说明

选项	说明
serial	区域文件的修订号码（序号），每次修改区域资源记录时次数会增加。当 Slave DNS 要进行资料同步时，它会比较这个号码，如果此处的值比较大，则会进行更新，反之则忽略。该值是一个正整数
refresh	Slave DNS 服务器与 Master DNS 服务器同步的时间间隔，在到达该时间后，Slave DNS 服务器会比较 Master DNS 服务器中的 SOA 序号
retry	如果与 Master DNS 服务器的同步没有成功，则 Slave DNS 服务器会在此处设置的时间间隔后再次尝试
expire	如果与 Master DNS 服务器的同步一直无法成功，则在此设置的时间后 Slave DNS 服务器便会放弃同步作业。expire 必须不小于 refresh 加上 retry 的时间，也必须不小于 10 倍的 retry 时间

（4）1D IN　NS　localhost.。表示一条 NS（Name Server）记录，即名称服务器记录。该语句用于指定域名服务器，NS 之后应放置当前域名服务器的名称。

（5）1D IN　A　127.0.0.1。表示一条 A（Address）记录，即地址记录。用于指定一个名称所对应的 IP 地址，域名的正向解析就是通过 A 记录来实现的，有多少个域名需要解析，就添加多少条 A 记录。该条记录的含义就是将 localhost 解析为 127.0.0.1。

通常，DNS 资源记录的设置都是使用“[名称][TTL][类型][资料]”的格式。下面是其中每个部分的说明。

名称：设置所用的名称，如主机名、IP 地址或网址等。

TTL：定义此记录在保存区的保留时间，空白表示默认值。

IN：固定格式，表示这是网络上的一条记录。

类型：此资源记录的类型，如 NS 或 A。

资料：每个资源记录类型的设置值，如 A 资源记录就必须在此输入 IP 地址。

6. /var/named/named.local

反向解析文件用于实现从 IP 到域名的转换。/var/named/named.local 用于实现从 127.0.0.1 到 localhost 的转换。

```
[root@rh9 root]# more    /var/named/named.local
$TTL86400
@          IN          SOA          localhost. root.localhost.    （
                                                1997022700 ; Serial
```

```
                                        28800       ; Refresh
                                        14400       ; Retry
                                        3600000     ; Expire
                                        86400 )       ; Minimum
                IN      NS      localhost.

1       IN      PTR     localhost.
```

在该文件中，时间全部采用的是以秒为单位进行表达，也可以用 D、H、M、W 为单位来表达，两种表达法等效。

最后一行的 PTR 用于定义一个 PTR 记录，即定义一条反向解析记录。该行前面的 1 代表当前网段（127.0.0.）内的第 1 台主机，即 IP 地址为 127.0.0.1 的主机，最后的 localhost 代表将该 IP 地址解析为 localhost 域名。

8.5.4 实现泛域名解析

泛域名解析是指一个域名下的所有主机和子域名都被解析成同一个 IP 地址。例如，使用命令“ping marry.cqcet.cn”和“ping jack.cqcet.cn”均能解析并返回同一个 IP 地址。也就是说，在域名 cqcet.cn 前面加上任意主机名，DNS 服务器都可以解析到同一个 IP 地址上去，这是因为负责解析 cqcet.cn 的 DNS 服务器使用了泛域名解析技术。

泛域名解析在实际使用中的作用是非常广泛的，除了可以将泛域名解析到默认 Web 网站，以方便用户的访问外，还可以实现基于数据库的二级域名管理。很多企业都为员工架设了个人 Web 站点来满足工作的需要，为了节省费用，这些 Web 网站通常采用虚拟主机技术，即在同一台服务器上架设多个网站，员工使用二级域名访问这些站点。由于员工数比较多，因此在 DNS 服务器上维护这些二级域名的工作量非常大，采用泛域名解析技术就可以很好地解决这个问题。

它可以将泛域名解析到一台 Web 服务器上，Web 服务器上的默认主页程序（如 index.jsp、index.php、default.cgi）对用户浏览器发送的 HTTP 访问请求信息进行分析，并分隔出二级子域名；然后查询数据库得到子域名对应的 URL，并利用重定向技术将访问的用户带到目标 URL，如网易免费二级域名转向系统 yeah.net 就是这样实现的。

要实现泛域名解析，可以在/etc/named.conf 文件末尾加入“* IN A 192.168.1.101”或“*.cqcet.cn IN A 192.168.1.101”这样一条特殊的 A 资源记录，以便支持实现泛域名解析功能。

8.5.5 DNS 服务器的启停

DNS 服务器的守护进程名字是 named，也可以像前面介绍的其他服务器一样，采用同样的几种方法进行启动、关闭，以及设置为自动启动等操作。下面仅以 service 和 chkconfig 命令来说明。

1. *启动与停止*

DNS 服务器的启动、停止等操作可以通过 service、ps 和 kill 命令来实现。

```
[root@rh9 root]# ps   -e | grep   named
 7558    ?          00:00:00   named                  //DNS 服务已经启动。
```

```
[root@rh9 root]# kill    7558                          //杀死 DNS 服务器进程。
[root@rh9 root]# ps    -e | grep    named
[root@rh9 root]# service    named    start             //启动 DNS 服务。
[root@rh9 root]# ps –e | grep    named
 7675    ?            00:00:00    named
```

2. 设置为开机自动启动

若要系统每次启动时自动开启 DNS 服务，可以使用如下命令，下面的例子表示在系统进入第 3 和第 5 个级别时自动开启 DNS 服务。

```
[root@rh9 root]# chkconfig    --level    35    named    on
[root@rh9 root]# chkconfig    --list    named
named      0:关闭    1:关闭    2:关闭    3:启用    4:关闭    5:启用    6:关闭
```

8.5.6　DNS 客户端设置

完成 DNS 服务器的配置后，接下来进行 DNS 客户端的联机测试，以判断服务器是否正常提供 DNS 服务，这里仅介绍 Linux 客户端的设置。

1. 直接修改配置文件的方法

字符界面下，通过文本编辑器 VI 修改配置文件/etc/resolv.conf 和/etc/nsswitch.conf 来实现。其中，/etc/resolv.conf 用于指定域名解析器所使用的默认域、域名查找顺序，以及域名服务器的 IP 地址，DNS 客户端就是利用其中设定的 IP 地址查找到 DNS 服务器的。按照以下格式修改该文件。

```
search    abc.com.cn
nameserver    192.168.1.1
nameserver    192.168.1.2
```

第一行：表示在没有指定计算机所在域时，默认该计算机属于指定的域。例如，要查找计算机 www，默认它是 abc.com.cn 中的一个成员，即 www 和 www.abc.com.cn 是一样的。

第二、三行：用于指定客户端使用的域名服务器的 IP 地址。可以为客户机指定多个域名服务器。当进行域名查询时，首先查找第一台域名服务器，如果第一台没有响应，这时就查找第二台域名服务器。

另外，/etc/nsswitch.conf 这个文件，用来决定是先使用/etc/hosts 还是先用/etc/resolv.conf 里面的设置进行名称解析。该文件中，“hosts: files dns”行规定了该主机上进行域名解析的顺序。files 就是使用/etc/hosts，后面的 dns 则是使用/etc/resolv.conf 的 DNS 主机 IP 查找。Linux 的默认名称搜索是先从/etc/hosts 开始的，一般无需修改。

2. 使用图形化配置工具

以 root 用户从 GNOME 图形界面登录系统，依次单击“主菜单”→“系统工具”→“网络”菜单，弹出“网络配置”对话框，选择“DNS”选项卡，如图 8-39 所示。

可以设置主机名，以及用于域名解析的 DNS 服务器地址。设置完成后，重启主机或网卡（禁用然后再启动），新的设置就生效了。

8.5.7　图形化配置 DNS 服务器

Red Hat Linux 9 提供了 redhat-config-bind 图形用户界面工具配置 DNS 服务器，选择桌面

面板上的“主菜单”→“系统设置”→“服务器设置”→“域名服务器”菜单命令，进入如图 8-40 所示的“域名服务”窗口。

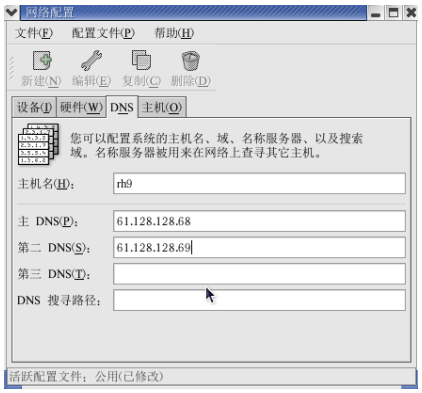

图 8-39　配置 DNS 客户端

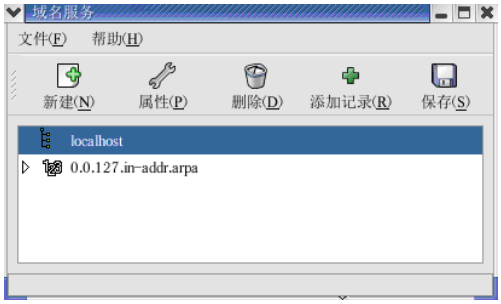

图 8-40　域名服务配置主窗口

下面通过实例说明域名服务器的配置与启用。假如有一个 IP 网络号为 192.168.0.0，掩码为 255.255.255.0 的局域网，要将其中一台 IP 地址为 192.168.0.1 的主机设置为域名服务器，同时将局域网的域名设置为 abc.com，局域网中的各主机已配置了同一段的静态 IP 地址，即 IP 地址前三段为 192.16.0，掩码为 255.255.255.0，现在要为它们分别设置形如 xxx.abc.com（其中的 xxx 分别设置为不同的主机名称）的主机域名，假设有一台 IP 地址为 192.168.0.15 的计算机名为 www，则域名为 www.abc.com。

1. 正向解析配置

在图中单击“新建”按钮，进入“选择一个区块类型”对话框，选中“正向主区块”单选按钮，在“域名”文本框中输入 abc.com（用户也可以根据自己的情况设置域名名称），如图 8-41 所示。

单击“确定”按钮，进入“名称到 IP 的翻译”对话框，如图 8-42 所示。不同的区块对应着不同的服务类型，正向区块负责完成域名到 IP 地址的正向解析过程，逆向区块负责完成 IP 地址到域名的反向解析过程，而从区块则是完成辅助 DNS 服务器的工作，如果局域网中还没有主 DNS 服务器是不能单独设置辅助服务器的。

图 8-41　“选择一个区块类型”对话框

图 8-42　“名称到 IP 的翻译”对话框

在“主名称服务器（SOA）”栏添入“@”。在“记录”框中已经有一个记录 abc.com，该记录是一个域名服务器记录，即标注区域 abc.com 的域名服务器（DNS 服务器）记录。单击“编辑”按钮，弹出“abc.com 设置”对话框，如图 8-43 所示。

在“名称服务器”配置组，单击“增加”按钮弹出图 8-44 所示“名称服务器的属性”对话框，在“服务器”文本框中输入 DNS 服务器所在的主机名（rh9），然后单击“确定”按钮，返回上一界面。中间的“邮件交换器”配置组是用来设置负责接收外部邮件的邮件服务器，由于暂时没有邮件服务器，因此该栏不填。在下面“IP 地址”文本框中输入 DNS 服务器所在主机的 IP 地址。

图 8-43　abc 设置对话框

图 8-44　“名称服务器的属性”对话框

设置完 abc.com 后，单击“确定”按钮，返回“名称到 IP 的翻译”对话框。单击“增加”按钮，添加主机 www 服务器域名和 IP 地址的对应关系，如图 8-45 所示。

添加好记录之后，单击“确定”按钮，一条记录将被添加到 DNS 服务器中。采用同样的方法，依次添加其他的所有需要的记录后，就完成了 DNS 服务器对 abc.com 域的正向主区块的设置。

2. 反向解析配置

在“域名服务”对话框（如图 8-40 所示）中单击“新建”按钮，弹出“选择一个区块类型”对话框。选中“逆向主区块”单选按钮，在“IP 地址”栏输入要添加区域数据 IP 地址的前三部分 192.168.0，如图 8-46 所示。完成后，单击“确定”按钮，弹出“IP 到名称的翻译”对话框，如图 8-47 所示。

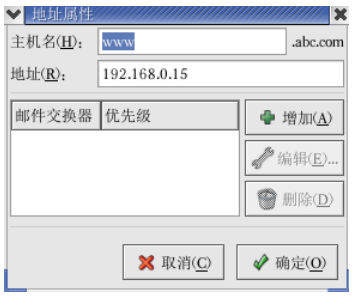

图 8-45　添加一条主机记录

图 8-46　“选择区块类型”对话框

在“主名称服务器（SOA）”栏添入“@”。在“名称服务器”栏中单击“增加”按钮，打开“名称服务器的属性”对话框，添加 DNS 服务器名称，如图 8-48 所示。

图 8-47　“IP 到名称的翻译”对话框

图 8-48　“名称服务器的属性”对话框

填完 DNS 服务器的 IP 地址后，单击“确定”按钮，返回“IP 到名称的翻译”对话框。单击“逆向地址表”处的“添加”按钮，弹出“新逆向区块指针”对话框，如图 8-49 所示，依次填写要解析的 IP 地址与域名的对应记录（要填写完整的域名并且最后要加一个点）。

反向解析区块也配置完毕后，就完成了对 DNS 服务器的配置，这时的 DNS 服务器配置主窗口如图 8-50 所示。然后，在该窗口的右上角单击“保存”按钮，将设置保存到配置文件并退出，DNS 服务器就配置完成了。

图 8-49　“新逆向区块指针”对话框

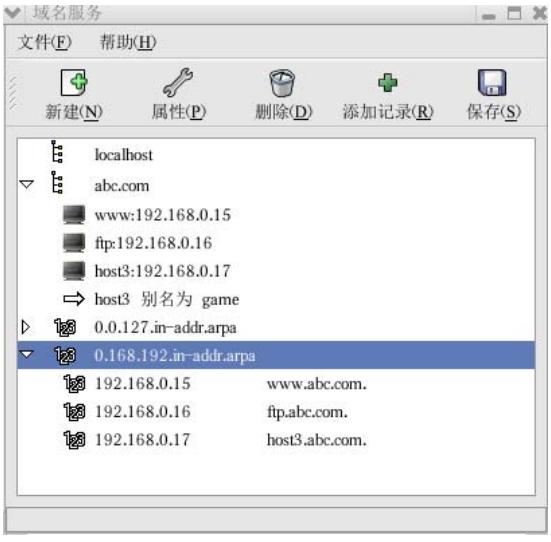

图 8-50　配置完毕后的域名服务窗口

8.5.8　DNS 服务器的测试

测试 DNS 服务器是否正常的方法很多，但首先应保证 DNS 客户机已在网络配置中正确设置了 DNS 服务器的地址。常用的测试命令有 ping、host、nslookup、dig 等命令。这里以 nslookup 命令为例介绍。其余命令的使用方法请读者自己查找有关资料或网站。

使用 nslookup 命令不仅可以测试 DNS 服务器的正向解析，也能测试反向解析。在命令行提示符“#”或“$”后按照“nslookup　主机域名”格式输入命令，如果在返回的结果中出现相应的 IP 地址，表示正向解析测试成功。按照“nslookup　主机 IP”格式输入命令，如果在返回的结果中出现相应的主机域名，表示反向解析测试也成功。

```
[root@rh9 root]# nslookup www.abc.com
Note:    nslookup is deprecated and may be removed from future releases.
Consider using the `dig' or `host' programs instead.    Run nslookup with
the `-sil[ent]' option to prevent this message from appearing.
Server:              192.168.0.1
Address:             192.168.0.1#53

Name:     www.abc.com
Address: 192.168.0.15
[root@rh9 root]# nslookup game.abc.com
Note:    nslookup is deprecated and may be removed from future releases.
Consider using the `dig' or `host' programs instead.    Run nslookup with
the `-sil[ent]' option to prevent this message from appearing.
Server:              192.168.0.1
Address:             192.168.0.1#53

game.abc.com       canonical name = host3.abc.com.
Name:     host3.abc.com
Address: 192.168.0.17
[root@rh9 root]# nslookup 192.168.0.17
Note:    nslookup is deprecated and may be removed from future releases.
Consider using the `dig' or `host' programs instead.    Run nslookup with
the `-sil[ent]' option to prevent this message from appearing.
Server:              192.168.0.1
Address:             192.168.0.1#53

17.0.168.192.in-addr.arpa          name = host3.abc.com.
```

8.6　DHCP 服务器

8.6.1　DHCP 简介

在使用 TCP/IP 协议的网络中的每台计算机，都必须至少有一个 IP 地址，这样才能与其他计算机通信。当计算机移动到不同的网段时，就必须修改有关的配置内容，否则会造成无法使用网络资源的情况。但是，这种修改设置的工作在大型或经常变动的网络中，例如客户端大多使用笔记本电脑的环境，就显得相当繁重，同时也增加了手动输入的错误机会。

动态主机配置协议（Dynamic Host Configuration Protocol，DHCP）的产生，很好地解决了这个问题。它是利用 DHCP 服务器所包含的 IP 地址数据库，以租用的方式来为客户端动态指

派 IP 地址及其他相关配置，如默认网关（路由）和域名服务器等，以便客户机能与其他网段的计算机通信或实现对 Internet 的访问，大大降低系统管理员重新设置计算机的复杂性和工作量。

网络中的大部分主机都可使用 DHCP 的配置，但对于一些特殊的主机却无法成为 DHCP 客户端，这些主机包括 DHCP 服务器本身、路由器、DNS 服务器或其他服务器等。

DHCP 使用客户机/服务器体系结构，因此系统管理员至少需要建立一台 DHCP 服务器，在 DHCP 服务器中应至少配置一个作用域，作用域是指 DHCP 服务器可分配租用给 DHCP 客户机的 IP 地址范围。各作用域的 IP 地址范围（IP 地址池）不能发生重叠。

在安装配置好 DHCP 服务器后，对于 Windows 系统的 DHCP 客户，只需在 TCP/IP 协议的属性对话框中，设置为“自动获得 IP 地址”，然后重启系统，即可从 DHCP 服务器的 IP 地址池中自动分配到 IP 地址。对于 Linux 系统的 DHCP 客户机，则应安装配置 DHCP 客户程序（dhclient），以便能动态分配到 IP 地址。

在网络上安装并设置 DHCP 服务器后，所有的 DHCP 客户端在每次启动时，都会发送广播的数据包来寻找 DHCP 服务器，以便取得 IP 地址以及相关的设置参数入网。

8.6.2 DHCP 服务器的安装

Red Hat Linux 9 自带有 DHCP 的安装软件包，相关文件共有三个，分别是：dhcp-3.0p11-23.i386.rpm，DHCP 服务器软件包，位于第 2 张光盘；dhclient-3.op11-23.i386.rpm，DHCP 客户端软件包，位于第 1 张光盘；dhcp-devel-3.0pl1-23.i386.rpm，DHCP 开发包，位于第 2 张光盘，提供库和头文件，一般不需要安装。

```
[root@rh9 dhcp]# rpm   -q   dhcp          //检查 dhcp 的相关软件是否已经安装。
dhcp-3.0p11-23
```

在安装 DHCP 服务器之前，应先使用上面的方法检查当前系统是否已经安装，如果输出如上所示的软件名称，则说明已经安装，否则可以使用下面的命令进行安装。

```
[root@rh9 dhcp]# mount   /mnt/cdrom
[root@rh9 dhcp]# cd   /mnt/cdrom/Red Hat
[root@rh9 dhcp]# rpm   -ivh   dhcp-3.0p11-23.i386.rpm
warning: dhcp-3.0pl1-23.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
Preparing...                ########################################### [100%]
   1:dhcp                   ########################################### [100%]
[root@rh9 dhcp]# rpm   -pql   dhcp-3.0pl1-23.i386.rpm      //查询安装文件列表。
warning: dhcp-3.0pl1-23.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
/etc/rc.d/init.d/dhcpd                          //DHCP 服务器启动脚本。
/etc/rc.d/init.d/dhcrelay                       //DHCP 中继服务启动脚本。
/etc/sysconfig/dhcpd
/etc/sysconfig/dhcrelay
/usr/bin/omshell
/usr/sbin/dhcpd                                 //DHCP 服务器守护进程程序。
/usr/sbin/dhcrelay                              //DHCP 中继服务守护进程程序。
/usr/share/doc/dhcp-3.0pl1
/usr/share/doc/dhcp-3.0pl1/CHANGES
/usr/share/doc/dhcp-3.0pl1/README
/usr/share/doc/dhcp-3.0pl1/RELNOTES
/usr/share/doc/dhcp-3.0pl1/dhcpd.conf.sample
/usr/share/man/man1/omshell.1.gz
```

```
/usr/share/man/man5/dhcp-eval.5.gz
/usr/share/man/man5/dhcpd.conf.5.gz
/usr/share/man/man5/dhcpd.leases.5.gz
/usr/share/man/man8/dhcpd.8.gz
/usr/share/man/man8/dhcrelay.8.gz
/var/lib/dhcp
/var/lib/dhcp/dhcpd.leases                    //DHCP 守护进程启动必须的文件。
```

DHCP 中继服务（dhcrelay）可以在没有 DHCP 服务器的网段上接收 DHCP 和 BOOTP（Internet Bootstrap Protocol）网络地址请求，并将其转发到另一个有 DHCP 服务器的网段。

8.6.3　DHCP 服务器的配置文件

有关 DHCP 服务器的配置几乎都集中在/etc/dhcp.conf 中，但由于系统安装后并不会自动生成该文件，所以建议先将该配置文件的模板/usr/share/doc/dhcp-3.0p11/dhcp.conf. sample，复制到/etc 目录下并改名为 dhcp.conf 然后再根据需要进行修改，下面显示了该文件的默认内容。

```
[root@rh9 dhcp]# more   /usr/share/doc/dhcp-3.0p11/dhcp.conf. sample
ddns-update-style interim;
ignore client-updates;
subnet 192.168.0.0 netmask 255.255.255.0 {

# --- default gateway
        option routers                  192.168.0.1;
        option subnet-mask              255.255.255.0;
        option nis-domain               "domain.org";
        option domain-name              "domain.org";
        option domain-name-servers 192.168.1.1;
        option time-offset              -18000;     # Eastern Standard Time
#       option ntp-servers              192.168.1.1;
#       option netbios-name-servers  192.168.1.1;
# --- Selects point-to-point node （default is hybrid）. Don't change this unless
# -- you understand Netbios very well
#       option netbios-node-type 2;

        range dynamic-bootp 192.168.0.128 192.168.0.255;
        default-lease-time 21600;
        max-lease-time 43200;

        # we want the nameserver to appear at a fixed address
        host ns {
                next-server marvin.redhat.com;
                hardware ethernet 12:34:56:78:AB:CD;
                fixed-address 207.175.42.254;
        }
}
```

DHCP 配置文件使用 subnet 定义作用域，一个网段应该定义一个作用域，作用域的定义采用以下的格式。

```
subnet   子网 1   netmask   子网掩码 {
  option   routers   默认网关地址;
  range   [dynamic-bootp]   low-address      [high-address];
   [其他可选的设置]
}
subnet   子网 2   netmask   子网掩码 {
```

```
    option  routers  默认网关地址;
    range  [dynamic-bootp]  low-address      [high-address];
     [其他可选的设置]
}
…  …
```

DHCP 配置文件使用 group 将多个需要特殊设置的主机归结为一个组，便于集中设置共同的项，如果要特别设置的主机很少，也可以不用 group，而直接使用 host 语句来指定。

group 组的定义采用以下的格式。

```
group {
    [组中的全局设置]              //对一个组中的所有主机都起作用的选项。
   host  主机名 1 {
   hardware  ethernet  MAC1;
   fixed-address  IP1;
    [其他可选的设置]
}
   host  主机名 2 {
   hardware  ethernet  MAC2;
   fixed-address  IP2;
    [其他可选的设置]
}
…  …
}
```

另外，在 subnet 和 group 声明的段落之前，通常还有一些配置参数，用来指定所有网段或主机的共同特性，即全局特性。以下是一些全局参数的设置范例。

```
ddns-update-sytle    interim    //设置 DNS 的动态更新方式（interim 或 ad-hoc）。
ignore client-updates   //不允许动态更新 DNS，如要允许则设置为 allow client-updates。
default-lease-time      259200          //默认的 IP 地址租约期限，单位为秒。
max-lease-time          777600          //IP 租约的最长期限，单位为秒。
server-identifer      192.168.0.1       //指定 DHCP 服务器的 IP 地址。
…  …
```

dhcpd 里有一个语法分析器，能对 dhcpd.conf 文件进行语法分析，获得必要的配置参数。这些参数可以通过 dhcpd.conf 中的 option 语句来定义，常见 option 选项如表 8.6 所示。语句以“;”结尾，注释以“#”开头直到该行结束。

表 8.6　dhcpd.conf 中常见的 option 选项及其含义

选项（option）	作用
routers	指定所在网段的默认网关
subnet-mask	指定所在网段的子网掩码
nis-domain	指定 NIS 服务器的地址
domain-name	指定所在的域名
domain-name-servers	指定 DNS 地址
time-offset	东部标准时间
range [dynamic-bootp]	指定可分配的 IP 地址范围，dynamic-bootp 可以不写
netbios-name-servers	指定 WINS 服务器的地址
netbios-node-type	指明客户端的 NetBIOS 节点类型
broadcast-address	指定所在网段的广播地址
time-servers	指定时间服务器

DHCP 服务器的配置比较简单，下面给出一个具体的实例，来说明 DHCP 服务器的配置方法和过程。假设要在某局域网中配置一台 DHCP 服务器，为两个网段 192.168.168.0/24 和 192.168.167.0/24 的用户提供 IP 地址动态分配服务。两个网段默认的主、次域名服务器都是 61.128.128.68 和 61.128.128.69。

192.168.168.0/24 网段动态分配的 IP 地址范围是 192.168.168.60~192.168.168.240，该网段的其余地址保留或用于静态分配，其中物理地址为 00:0C:29:04:FB:E2 的网卡固定分配 192.168.168.8，物理地址为 00:0C:29:04:ED:35 的网卡固定分配 192.168.168.9。该网段的默认网关是 192.168.168.1。

用于给 192.168.167.0/24 网段动态分配的 IP 地址范围是 192.168.167.20~192.168.167.100 和 192.168.167.140~192.168.167.240。其中，物理地址为 00:0C:29:1E:2F:4A 的网卡，固定分配 IP 地址 192.168.167.100。该网段的默认网关是 192.168.167.1。

根据需求分析可知，要提供动态 IP 地址分配的网段有两个，因此在 DHCP 服务器中，需要定义两个 DHCP 作用域。两个作用域都有相同的域名服务器和子网掩码，可将这两项指定为全局配置，在各作用域中就不用再单独设置了。下面是使用文本编辑程序，按照需求修改后的配置文件。

```
[root@rh9 dhcp]# vi    /etc/dhcpd.conf
//以下是全局设置。
ddns-update-style    interim ;
ignore    client-update s ;
default-lease-time        28800 ;
max-lease-time            43200 ;
option    subnet-mask      255.255.255.0 ;
option    domain-name-servers    61.128.128.68 , 61.128.128.69 ;
//以下分别是两个作用域的单独设置。
subnet    192.168.168.0    netmask 255.255.255.0 {
range    192.168.168.60    192.168.168.240 ;
option    broadcast-address    192.168.168.255 ;
option routers    192.168.168.1 ;
}
subnet    192.168.167.0    netmask 255.255.255.0 {
range    192.168.167.20    192.168.167.100 ;
range    192.168.167.140    192.168.167.240 ;
option    broadcast-address    192.168.167.255 ;
option    routers    192.168.167.1 ;
}
//以下是对特殊主机进行的设置。
group {
//以下是组中的全局设置。
default-lease-time    36000 ;                  //可以为该组中的主机单独设置默认租期。
option    routers    192.168.168.1 ;          //可以为该组中的主机单独指定默认网关。
//以下分别是对 3 个主机的单独设置。
host    host1 {
hardware    ethernet    00:0C:29:04:FB:E2 ;
fixed-address    192.168.168.8 ;
}
host    host2 {
hardware    ethernet    00:0C:29:04:ED:35 ;
fixed-address    192.168.168.9 ;
```

```
}
host  host3 {
hardware  ethernet  00:0C:29:1E:2F:4A ;
fixed-address  192.168.167.100 ;
option routers  192.168.167.1 ;   //可以为该主机单独指定默认网关。
}
}
```

8.6.4 DHCP 服务器的启停

1. 使用 service 命令

DHCP 服务器也可以像前面介绍的其他服务器一样，使用 service 命令进行启动、关闭、重新启动，以及查看运行状态等操作。DHCP 服务器网卡绑定的静态 IP 地址应该和其分配出去的 IP 地址在同一个网段。否则，DHCP 服务器进程（dhcpd）将无法启动成功。

```
[root@rh9 root]# service  dhcpd  start
启动 dhcpd：                                                [  确定  ]
[root@rh9 root]# service  dhcpd  status
dhcpd （pid 5147） 正在运行...
[root@rh9 root]# service  dhcpd  stop
关闭 dhcpd：                                                [  确定  ]
[root@rh9 root]# service  dhcpd  restart
关闭 dhcpd：                                                [  失败  ]
启动 dhcpd：                                                [  确定  ]
```

2. 使用 chkconfig 命令

若要系统每次启动时自动开启 DHCP 服务，可以使用 chkconfig 命令，下面的例子表示在系统进入第 3 和第 5 个级别时自动开启 DHCP 服务。

```
[root@rh9 root]# chkconfig  --level  35  dhcpd  on
[root@rh9 root]# chkconfig  --list  dhcpd
dhcpd    0:关闭  1:关闭  2:关闭  3:启用  4:关闭  5:启用  6:关闭
```

8.6.5 DHCP 的客户端设置

DHCP 的客户端有多种类型，可以是 Windows 操作系统，也可以是 Linux/UNIX。Linux 下的 DHCP 客户端需要安装 dhclient-3.op11-23.i386.rpm 软件包，它位于第 1 张 Linux 系统光盘上，默认情况下已经安装。下面分别介绍 DHCP 服务器的 Linux 客户端以及 Windows XP 客户端的设置。

1. Linux 客户端的设置

以 root 用户从 GNOME 图形界面登录系统，依次单击“主菜单”→“系统工具”→“网络”菜单，弹出“网络配置”对话框，在“设备”选项卡中双击要配置的网卡，弹出图 8-51 所示的窗口，这里选择通过 DHCP 方式自动获取 IP 地址，并自动从网络提供商（ISP）处获取 DNS 信息。

另外，在 Linux 命令行界面下设置 DHCP 客户端，要使用文本编辑器修改配置文件的方法：执行命令“vi /etc/sysconfig/network-scripts/ifcfg-eth0”，打开配置文件并找到其中的“BOOTPROTO = none”行，将其修改为“BOOTPROTO = dhcp”即可。

设置完成后，先关闭客户端网卡然后再启动，然后使用 ifconfig 命令查看从 DHCP 服务器获得 IP 地址的情况，判断 DHCP 服务器以及客户端是否已经能够正常工作。

DHCP 服务器会把出租出去的 IP 地址信息存放在/var/lib/dhcp/dhcpd.leases 文本文件中，通过查看该文件的内容，也可在一定程度上判断其是否已经正常工作。

2. Windows 客户端的设置

在Windows XP的控制面板中双击网络连接图标，单击要设置的网卡，在弹出的“TCP/IP属性”窗口的“常规”选项卡中，选择“自动获得 IP 地址”，如图 8-52 所示。如果 DHCP 还提供 DNS、WINS 等设置，也可以把它们设置为自动获得。

图 8-51　Linux 下配置 DHCP 客户端

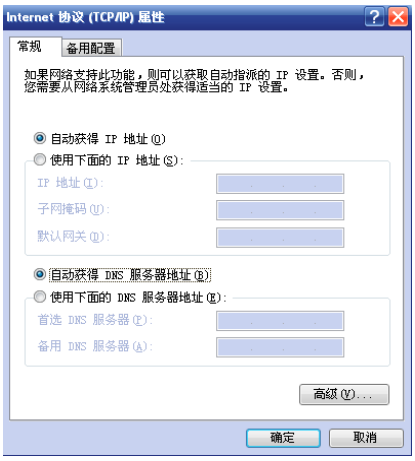

图 8-52　Windows 下配置 DHCP 客户端

设置完成后，先关闭客户端网卡然后再启用，然后执行“ipconfig　/all”命令查看从 DHCP 服务器自动获得配置的情况，判断 DHCP 服务器以及客户端是否已经能够正常工作。另外，可以执行“ipconfig　/renew”命令更新 IP 地址，执行“ipconfig　/release”命令释放从 DHCP 服务器获得的 IP。

8.6.6　DHCP 中继代理

如果网络跨越了多个子网，子网之间是以具有多个网络接口的同一台主机相连，在此种情形下 DHCP 服务器可能无法为远在其他子网的客户机提供服务，因为 DHCP 客户端是靠广播形式请求 DHCP 服务器的，但是一般情形下，网络接口之间并不会自动转发广播的数据包。要解决以上的问题，就必须使用 DHCP 中继代理（Relay Agent）的功能。

通过 DHCP 中继代理，可以使用同一台 DHCP 服务器为多个子网提供 IP 地址，而不需要在每个子网中配置 DHCP 服务器。默认情况下，在安装 DHCP 服务器程序时，已经安装了 DHCP 中继代理程序（/usr/sbin/dhcrelay）。

一般情况下，dhcrelay 启动后将监听所有网络接口上的 DHCP 请求并提供中继代理服务，当然也可以让其只监听部分网络接口。完成这一工作，要使用命令 dhcrelay，并且拥有 root 的权限，其语法格式为：“dhcrelay　[-p　连接端口]　-i　网络接口　DHCP 服务器 IP”。

其中，选项-p 用来指定中继代理服务使用的连接端口号，如果没有指定，系统默认使用 UDP 67 号端口；选项-i 最为重要，用来指定为哪些直接与 DHCP 服务器网卡相连的子网提供 DHCP 中继服务。

比如，命令“dhcrelay　-i　eth1　-i　eth2　192.168.168.33”，表示 DHCP 服务器的 IP 地

址是 192.168.168.33，位于与网卡 eth0 直接相连的子网 A 中，通过 DHCP 中继代理服务，使 DHCP 服务器也能够向与网卡 eth1、eth2 相连的子网 A 与子网 B 提供服务。

多个子网相连，更多的情况下采用硬件路由器，而并非具有多个网络接口的主机。在这种情况下，要选用具有 DHCP 中继功能的路由器来支持，来实现跨网段的 IP 地址自动分配。

8.7 本章习题

一、填空题

1．在启动 NFS 服务器之前，一定要先启动_____服务，否则 NFS 不能启动成功。

2．Samba 服务守护进程是 Samba 的核心，时刻侦听网络的文件和打印服务请求，该进程的名字是_____。

3．DNS 服务器的正向解析用于实现从_____到_____的转换。

4．_____服务器用来实现给网络的客户端自动分配 IP 地址。_____是指一个域名下的所有主机和子域名都被解析成同一个 IP 地址。

5．能让 Windows 主机访问 Linux 系统中共享文件的服务器是_____。

二、判断题

1．NFS 服务器不能实现 Windows 和 Linux 主机之间的文件共享。（　　）

2．Samba 服务器与 Linux 操作系统使用不同的密码文件，所以无法以 Linux 用户的系统登录密码登录 Samba 服务器。（　　）

3．DHCP 服务器只能给和服务器同在一个网段的主机自动分配 IP 地址。（　　）

4．使用图形化的配置工具对服务器进行配置不但方便，还可以对服务器实现更精细化的管理。（　　）

5．所有的 Linux 服务器都可以通过直接修改配置文件的方法实现配置。（　　）

三、选择题

1．使用 Samba 服务器，一般来说，可以提供（　　）。

A．域名服务　　B．文件服务

C．打印服务　　D．IP 地址解析

2.在使用 Samba 服务时，由于客户机查询 IP 地址不方便，可能需要管理员手工设置（　　）文件。

A．smb.conf　　B．lmhosts

C．fstab　　D．mtab

3．一个完整的 smb.conf 文件中关于 Linux 打印机的设置条目有（　　）。

A．browseable　　B．public

C．path　　D．guest ok

4．Samba 所提供的安全级别包括（　　）。

A．share　　B．user

C．serve　　D．domain

5．Samba 服务器的默认安全级别是（　　）。

A．share　　B．user

C．server　　D．domain

6．可以通过设置条目（　　）来控制可以访问 Samba 共享服务的合法主机名。

A．allowed　　B．hosts valid

C．hosts allow　　D．public

7．下列（　　）命令允许修改 Samba 用户的口令。

A．passwd　　B．mksmbpasswd

C．password　　D．smbpasswd

8．Samba 后台的两个核心进程是（　　）。

A．smbd 和 nmbd　　B．inetd 和 smbd

C．inetd 和 httpd　　D．nmbd 和 inetd

9．要检查当前 Linux 系统是否已经运行了 DNS 服务器，以下命令中正确的是（　　）。

A．rpm - q | grep dns　　B．rpm -q bind

C．ps -aux | grep bind　　D．ps –aux | grep named

10．若使用 vsftpd 的默认配置，使用匿名账户登录 FTP 服务器，所处的目录是（　　）。

A．/home/ftp　　B．/var/ftp

C．/home　　D．/home/vsftpd

11．若要设置 Web 站点根目录的位置，应在配置文件中通过（　　）配置语句来实现。

A．ServerRoot　　B．ServerName

C．DocumentRoot　　D．Directorylndex

12．若要设置网页默认使用的字符集为简体中文，则应在配置文件中添加（　　）配置项。

A．DefaultCharset GB2312　　B．AddDefaultcharset GB2312

C．DefaultCharset ISO-8859-1　　D．AddDefaultCharset GB5

13．若要设置 Apache 服务器允许持续连接，则设置命令为（　　）。

A．KeepAlive On　　B．KeepAliveTimeout 10

C．MaxKeepAliveRequests 100　　D．KeepConnect On

14．设置站点的默认主页，可在配置文件中通过（　　）配置项来实现。

A．RootIndex　　B．ErrorDocument

C．DocumentRoot　　D．DirectoryIndex

四、综合题

1．vsftpd 是 Red Hat Linux 9 中默认采用的 FTP 服务器程序，其主要的配置文件有 3 个：/etc/vsftpd.ftpusers 、 /etc/vsftpd.user_list 和 /etc/vsftpd/vsftpd.conf 。现在其主配置文件 /etc/vsftpd/vsftpd.conf 中有如下的设置：

```
anonymous_enable=YES
local_enable=YES
```

```
write_enable=YES
local_umask=022
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
xferlog_std_format=YES
pam_service_name=vsftpd
userlist_enable=YES
userlist_deny=YES
listen=YES
tcp_wrappers=YES
```

① 请问该配置是否允许匿名用户登录？是否允许本机使用者登录？

② 如果想禁止匿名用户登录应如何设置？怎么才能开启匿名用户上传文件的权限？

③ 配置文件中的 userlist_enable=YES 与 userlist_deny=YES 分别起什么作用？

④ 怎么配置才能使得只有/etc/vsftpd.user_list 文件中列出的用户才能登录？

⑤ 使用 service 命令可以在不重启主机的情况下重启服务器进程，写出重启该 FTP 服务器进程的命令？

2．根据下列要求配置 Apache 服务器，写出服务器配置文件 httpd.conf 中能够满足相应要求的部分。要求：① Apache 服务器域名是 www.xyz.com；② 服务器允许的最大客户请求数为 100；③ 不限制每次连接的最大请求数；④ Apache 服务器的默认网页放置在/var/www/html 目录中；⑤ 服务器 IP 地址是 221.124.8.100，使用 80 端口；⑥ 默认主页的搜索顺序是 index.html、index.htm、index.asp。

注：Apache 服务器配置文件 httpd.conf 中具有的部分配置项如下。

```
ServerRoot
ServerName
Timeout
Lisen
MaxKeepAliveRequests
KeepAliveTimeout
StartServers
MaxClients
MaxRequestsPerChild
DocumentRoot
UserDir
DirectoryIndex
```

3．请自己架设 Samba 服务器，并共享一个目录，使得同一个网段其他主机上的用户只能浏览和下载该目录中的文件。