《开源 IT 系统及应用架构宝典—系统、工具、案例》

作者：华清远见

第 3 章 文件服务配置与应用

本章目标

本章讲述内容中涉及以下开源软件。

- ProFTPd：官方网站 http://www.proftpd.org/
- ClamAV：官方网站 http://www.clamav.net/
- MySQL：官方网站 http://www.mysql.com/
- vsftpd：官方网站 http://vsftpd.beasts.org/
- Samba：官方网站 http://www.samba.org/
- Samba-vscan：官方网站 http://www.openantivirus.org/projects.php/

文件服务应该算得上是企业信息化建设中第一个会使用到的服务。文件服务器为网络上各主机提供了完整的数据、文件、目录等信息的共享，实现了统一管理。文件服务器能进行文件建立、删除、打开、关闭、读写等操作。在企业网络中，为了有效地进行各项文件的管理功能，通常都需要一台或多台服务器提供数据、文件、目录等的信息共享。文件服务器位于网络上的中心位置，当用户需要文件时，可以访问文件服务器上的文件，而不必在各自独立的计算机之间传送文件。

使用 Linux 作为文件服务器不管从总体成本还是可靠性来说都是一个非常不错的选择。

Linux 下可以作为文件服务器的服务也比较多，比如 FTP、NFS、Samba 等，在本章中主要介绍 FTP、Samba 这两种目前比较流行的文件服务。

3.1 ProFTPd：易于配置的 FTP 服务器

ProFTPd（ProFTPd 官方网站：http://www.proftpd.org/）是一套可配置性强的类 UNIX 平台上的 FTP 服务器软件，是在自由软件基金会的版权声明（GPL）下开发、发布的免费软件。ProFTPd 设计目标是实现一个安全且易于配置的 FTP 服务器。ProFTPd 主要包含以下特点。

（1）一个与 Apache 的 httpd.conf 类似的配置文件。
（2）每个目录下的.ftpaccess 文件和 Apache 的.htaccess 文件类似。
（3）可以单独运行也可以从 inetd/xinetd 启动。
（4）匿名 FTP 的根目录不需要特别的目录结构。
（5）系统的二进制文件和其他系统文件没有 SITE EXEC 命令。
（6）在单独运行方式下，以非特权用户运行，降低攻击风险。
（7）提供日志以及 utmp/wtmp 支持。
（8）提供 Shadow 口令支持。

3.1.1 ProFTPd 安装

目前 ProFTPd 最新稳定版是 1.3.2a，在一台安装了 RHEL/CentOS 5.4 的计算机上安装并配置 ProFTPd 的操作步骤如下。

CentOS（Community Enterprise Operating System，社区企业操作系统）并不是全新的 Linux 发行版，而是 RHEL 的派生版本。在众多的 RHEL 的派生版本中，CentOS 是比较出众的一个。CentOS（CentOS 官方网站：http://www.centos.org/）将 RHEL 发行的源代码重编译一次，形成一个可使用的二进制版本。由于 Linux 的源代码是 GNU，所以从获得 RHEL 的源代码到编译成新的二进制都是合法的。只是 Red Hat 是注册商标，所以在新的发行版里不能出现 Red Hat 的商标。Red Hat 公司对这种发行版并不反对，Red Hat 公司认为真正付费的用户，重视的并不是系统本身，而是 Red Hat 公司所提供的商业服务。所以 CentOS 可以得到 RHEL 的所有功能，甚至是更好的软件，但 CentOS 并不向用户提供商业支持，当然也不承担任何商业责任。

（1）使用如下命令下载并安装 ProFTPd。

```
cd /usr/src/
wget ftp://ftp.proftpd.org/distrib/source/proftpd-1.3.2a.tar.bz2
tar -xvjf proftpd-1.3.2a.tar.bz2
cd proftpd-1.3.2a
./configure --sysconfdir=/etc/proftpd
make && make install
```

（2）使用如下命令建立 FTP 匿名用户使用目录。

```
mkdir /var/ftp
```

ProFTPd 在安装完成后，默认是开启匿名用户访问的，与 vsftpd 相同默认用户为 ftp，但在 ProFTPd 安装时并没有将 ftp 用户所指定的家目录（在 RHEL/CentOS 中默认为 /var/ftp）建立，为了保证匿名用户的正常访问需要手动建立该目录。

（3）使用如下方法制作 ProFTPd 在 RHEL/CentOS 中的启动脚本。

- 复制启动脚本例子文件。

```
cd /usr/src/proftpd-1.3.2a
cp contrib/dist/rpm/proftpd.init.d /etc/rc.d/init.d/proftpd
```

- 使用以下命令修改启动脚本权限。

```
chmod +x /etc/rc.d/init.d/proftpd
```

- 增加 proftpd 服务。

```
chkconfig --add proftpd
```

（4）修改 ProFTPd 服务启动组。由于在 RHEL/CentOS 中 nobody 用户是属于 nobody 组，而不是 ProFTPd 配置文件中默认的 nogroup 组，所以为了在 RHEL/CentOS 中可以正常启动服务需要修改 /etc/proftpd/proftpd.conf 文件，内容如下。

```
Group        nobody
```

（5）ProFTPd 在默认情况下不支持断点续传功能，但目前大部分用户在下载时都希望服务器可以支持断点续传功能，因此需要在/etc/proftpd/proftpd.conf 文件中增加如下内容。

```
AllowRetrieveRestart on
AllowStoreRestart on
```

在安装 ProFTPd 服务后，可通过“service proftpd start|stop|restart|reload”命令将 ProFTPd 服务启动、停止、重新启动、重新载入配置文件。

3.1.2 ProFTPd 配置

根据在安装时的定义，在本章中 ProFTPd 的配置文件位于/etc/proftpd 目录，文件名为 proftpd.conf，该配置文件（如图 3-1 所示）与 Apache 的配置文件非常类似。ProFTPd 的配置文件与其他大多数 Linux 配置一样，以井号“#”开始的是注释行（在执行时将被忽略），对大小写敏感，所有参数的配置形式均使用“参数 值”的方式，如果某个参数有多个值时使用空格分隔多个参数。

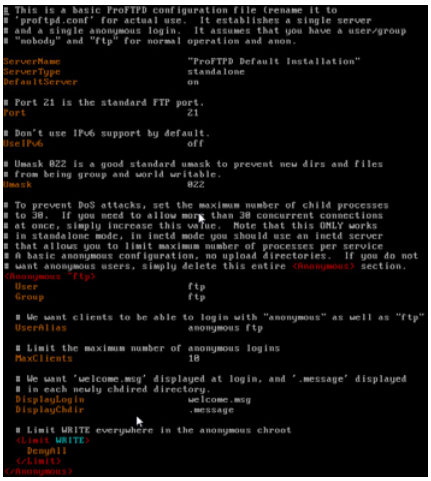


图 3-1 /etc/proftpd/proftpd.conf

proftpd.conf 文件可以分为全局配置、目录配置 2 个部分，其中全局配置主要针对 ProFTPd 运行状态进行调整，而目录配置主要针对 FTP 中某个目录的权限等内容进行调整。在 ProFTPd 所提供的全局配置参数中常用的主要包括以下几个。

（1）ServerName：当客户端连接到 FTP 服务器时，ProFTPd 显示在客户端的一些信息。

（2）port：ProFTPd 使用的 FTP 端口。

（3）UseIPv6 on|off：是否支持 IPv6。

（4）DefaultRoot 目录 用户|组：限制用户或组登录到 FTP 服务器后所在的目录，在默认情况下每个用户登录到 ProFTPd 后，会被引导到该用户的家目录。

（5）RootLogin on|off：是否允许 root 用户登录，默认是不允许的，安全起见不推荐将此选项设置为 on。

（6）ServerIdent on|off：是否屏蔽服务器的版本等相关信息，为安全起见推荐将此选项设置为 on。

（7）DefaultServer on|off：默认主机，只有在需要配置虚拟主机时需要将该参数配置为 off。

（8）IdentLookups on|off：是否开启反向查询客户端的用户名的操作。

（9）UseReverseDNS on|off：是否开启 DNS 反向查询。

（10）DeleteAbortedStores on|off：是否自动删除未传完的文件。

（11）DirFakeUser on|off：是否显示真实的文件所有者信息。

（12）DirFakeGroup on|off：是否显示真实的文件的拥有组信息。

（13）DirFakeMode on|off：是否显示真实的文件的读写操作信息。

（14）HideNoAccess on|off：是否隐藏没有访问权限的文件。

（15）RequireValidShell on|off：用户是否可以执行 shell。

（16）AllowRetrieveRestart on|off：下载时是否允许断点续传。

（17）AllowStoreRestart on|off：上传时是否允许断点续传。

（18）WtmpLog on|off：是否要把 FTP 记录在日志中。

（19）Umask Umask 值：通过 ProFTPd 上传文件的默认 Umask 值。

（20）UserAlias 别名 用户名：为用户名指定一个别名。

（21）TransferRate STOR|RETR 速率（Kbit）user|group 用户或组：指定用户或组上传（STOR）或下载（RETR）传输速率，user 表示其后指定的是用户，group 表示其后指定的是组。

```
#表示 hr 组所有用户最大下载速率为 50Kbit/s
TransferRate RETR 50 group hr
#表示用户 tonyzhang、davidxu 最大上传速率为 100Kbit/s
TransferRate STOR 100 user tonyzhang,davidxu
```

（22）UserRatio 用户 fileratio filequota byteratio bytequota：指定用户上传、下载的比例。fileratio 指定以文件为准的比例（该参数一般很少使用，所以通常设置为 0），filequota 指定可以下载多少文件（设置为 0 表示不限制），byteratio 指定上传、下载比例（该设置一个整数，表示当上传为 1 时，下载的比例），bytequota 可以下载多少 KB 的数据。

```
#表示 tonyzhang 上传、下载比率为 1:2，最多允许下载 100000 的数据
UserRatio        tonyzhang0 0 2  100000
#表示 davidxu 上传、下载比率为 1:1，最多允许下载 30000 的数据
UserRatio        davidxu  0 0 1  30000
#表示 janeli 上传、下载比率为 2:1，最多允许下载 30000 的数据
UserRatio        janeli   0 0 -2 30000
```

（23）GroupRatio 组 fileratio filequota byteratio bytequota：指定一个组的所有用户上传、下载的比例。该参数配置方法与 UserRatio 相同。

（24）MaxHostsPerUser 数量 [返回信息]：每个用户最多允许来源 IP 的个数，为了提高 FTP 用户账号的安全，将此参数配置为 1。

（25）MaxClientsPerUser 次数 [返回信息]：每个用户在每个客户端最多可以同时登录的次数，可以防止多线程软件下载对服务器的破坏。

（26）MaxClientsPerHost 数量 [返回信息]：同一个客户端最多允许多少个线程连接到 FTP 服务器。

（27）MaxClients 数量 [返回信息]：指定最大客户端数。

（28）MaxLoginAttempts 次数 [返回信息]：指定最大尝试连接次数。
（29）TimeoutIdle 秒数：客户端空闲断开时间设置，默认就是 600 秒。
（30）TimeoutNoTransfer 秒数：客户端数据传输超时设置。
（31）PassivePorts 最小端口 最大端口：被动模式端口范围。
（32）DisplayLogin 文件名：指定记录 FTP 客户端登录时显示欢迎信息的文件。
（33）DisplayChdir 文件名：指定记录当 FTP 客户端进入某个目录时显示欢迎信息的文件。

在 DisplayLogin、DisplayChdir 两个参数所指定的文件中，可以使用一些 ProFTPd 预设的变量，通过这些变量可以显示一些更友好的欢迎信息。这些变量主要包括%T（当前时间）、%F（所在硬盘剩下的空间）、%C（当前所在的目录）、%R（客户端的主机名称）、%L（服务器端的主机名称）、%U（用户账户名称）、%M（允许最大连接数）、%N（目前服务器连接数）、%E（FTP 服务器管理员的 Email）、%i（本次上传的文件数量）、%o（本次下载的文件数量）、%t（本次上传+下载的文件数量）。

在 ProFTPd 中的目录配置采用如下方式，如果对 Apache 的配置比较了解的话应该不难发现 ProFTPd 与 Apache 的配置方法非常相似。

```
<Directory 目录名>
    参数
    <Limit 限制动作>
      适用对象
    </Limit>
</Directory>
```

其中参数使用的就是 ProFTPd 全局配置中的参数（全局配置中大部分的参数都可以使用），当某个目录中的参数与全局配置中的参数发生冲突时，以这个目录的参数为准。

Limit 主要用于指定需要限定的动作，常用的限制动作主要包括 CMD（改变目录）、MKD（建立目录）、RNFR（更改目录名）、DELE（删除文件）、RMD（删除目录）、RETR（下载）、STOR（上传）、READ（可读，不包括列目录的权限）、WRITE（写文件或者目录的权限，包括 MKD 和 RMD）、DIRS（是否允许列目录）、ALL（所有权限）、LOGIN（登录，该参数对目录配置无效，只能在全局配置中使用），适用对象主要包括 AllowUser（针对某个用户允许的 Limit）、DenyUser（针对某个用户禁止的 Limit）、AllowGroup（针对某个用户组允许的 Limit）、DenyGroup（针对某个用户组禁止的 Limit）、AllowAll（针对所有用户组允许的 Limit）、DenyAll（针对所有用户禁止的 Limit）、Allow from（针对客户端 FQDN 或 IP 地址允许的 Limit，客户端的表示方法如表 3-1 所示）、Deny from（针对客户端 FQDN 或 IP 地址拒绝的 Limit，客户端的表示方法如表 3-1 所示）。下面来看一个目录配置的例子。

表 3-1 客户端定义

客户端指定方法	示　例	满足示例的客户端
IP 指定单一主机	192.168.159.30	客户端 IP 地址为 192.168.159.30
指定网段	192.168.159.0/255.255.255.0	客户端所在网段为 192.168.159.0/24
指定网段	192.168.159.0/24	
域名单一主机	client.example.com	客户端 FQDN 为 client.example.com
域名指定范围	.example.com	客户端 FQDN 的 DNS 后缀为 example.com
所有客户端	all	

```
#在/home/tonyzhang 目录，拒绝 tonyzhang 用户写并限制下载速率为 50Kbit/s、上传速
#率为 10050Kbit/s
<Directory ~tonyzhang>
    <Limit WRITE>
        DenyUser tonyzhang
    </Limit>
    TransferRate RETR 50 user tonyzhang
```

```
        TransferRate STOR 100 user tonyzhang
    </Directory>
```

除了上述适用对象外，ProFTPd 可以通过 order allow，deny 或 order deny，allow 指定默认的动作及冲突优先级，当使用 order allow、deny 时默认除了明确允许的客户端以外拒绝所有客户端，当允许与拒绝发生冲突时允许优先。当使用 order deny、allow 时默认除了明确拒绝的客户端以外允许所有客户端，当允许与拒绝发生冲突时拒绝优先。下面看几个例子。

```
    #对于/home/tonyzhang 目录只允许 192.168.159.0/24 网段的计算机可写
    <Directory~tonyzhang>
        <Limit WRITE>
            order allow,deny
            Allow from 192.168.159.
            Deny from all
        </Limit>
    </Directory>
    #对于/home/davidxu 目录允许除了 192.168.159.0/24 网段以外的计算机可写
    <Directory~davidxu>
        <Limit WRITE>
            order deny,allow
            Deny from 192.168.159.
            Allow from all
        </Limit>
    </Directory>
```

在 ProFTPd 默认的目录配置中包括一个特殊的目录配置，即匿名用户的目录配置。匿名用户的配置方法如下。

```
    <Anonymous 匿名用户根目录>
        参数
        <Limit 限制动作>
            适用对象
        </Limit>
    </Anonymous>
```

在 ProFTPd 所提供的配置文件例子中默认就包括如下的匿名用户配置，这些配置的作用如下。

```
    <Anonymous~ftp>     #指定匿名用户登录后，进入 ftp 用户家目录
    #登录后，该进程权限为 ftp:ftp
        User          ftp
        Group             ftp
        UserAlias         anonymous ftp    #为 ftp 用户指定别名 anonymous
        MaxClients    10                #指定最大客户端连接数为 10
        DisplayLogin welcome.msg        #指定记录欢迎信息的文件名为 welcome.msg
        DisplayChdir .message           #指定记录切换目录显示信息的文件名为.message
        <Limit WRITE>
            DenyALL
        </Limit>
    </Anonymous>
```

与 Apache 相同，在 ProFTPd 的主配置文件中也可以通过 Include 将另一个配置文件的内容引入到主配置文件中，比如将下面的内容加入到/etc/proftpd.proftpd.conf 文件后，ProFTPd 在启动时，会将/etc/proftpd/tls.conf 文件的内容加入到/etc/proftpd.conf 文件中。

```
    Include /etc/proftpd/tls.conf
```

ProFTPd 应该算是 FTP 服务器中功能最丰富的一个，下面将介绍搭建生产环境中使用的 FTP 服务所需的常用功能在 ProFTPd 中的实现方法。

1. 虚拟用户

当用户访问 ProFTPd 时，可以要求输入用户及密码。输入的用户及密码来源可以是本地用户（ProFTPd 安装完成默认就支持）、数据库或 LDAP 中。相对于使用 FTP 的本地用户形式来说，虚拟用户只是 FTP 服务器的专有用户，虚拟用户只能访问 FTP 服务器所提供的资源，这大大增强了系统本身的安全性。在 ProFTPd 中不但可以将虚拟用户保存到 MySQL 中，而且还可以对这些虚拟用户配置磁盘配额。下面介绍通过 MySQL 保存 ProFTPd 虚拟用户及虚拟用户空间配额的配置方法，操作步骤如下。

（1）使用以下命令安装 MySQL。

```
yum -y install mysql-devel mysql-server
```

（2）启动 mysqld 服务，并设置为下次启动自动加载。

```
service mysqld restart
chkconfig mysqld on
```

（3）使用 mysqladmin 创建 MySQL 管理员及密码。

```
#建立名为 root 的 MySQL 管理员，并将密码设置为 redhat
mysqladmin -u root password redhat
```

（4）使用 root 用户登录 MySQL 数据库，建立用于保存虚拟用户的数据库。

```
create database proftpd;
```

（5）在 MySQL 环境中执行以下语句，建立保存虚拟用户、组及磁盘配额所需的表。

```
CREATE TABLE 'ftpgroups' (
 'groupname' varchar(30) NOT NULL default '',
 'gid' int(11) NOT NULL default '1000',
 'members' varchar(255) NOT NULL default ''
)ENGINE=MyISAM DEFAULT CHARSET=latin1;
CREATE TABLE 'ftpusers'(
 'userid' varchar(30) NOT NULL default '',
 'passwd' varchar(80) NOT NULL default '',
 'uid' int(10) unsigned NOT NULL default '1000',
 'gid' int(10) unsigned NOT NULL default '1000',
 'homedir' varchar(255) NOT NULL default '',
 'shell' varchar(255) NOT NULL default '/sbin/nologin',
 'count' int(10) unsigned NOT NULL default '0',
 'host' varchar(30) NOT NULL default '',
 'lastlogin' varchar(30) NOT NULL default '',
 UNIQUE KEY 'userid' ('userid')
)ENGINE=MyISAM DEFAULT CHARSET=latin1;
CREATE TABLE 'quotalimits'(
 'name' varchar(30) default NULL,
 'quota_type' enum('user','group','class','all') NOT NULL default 'user',
 'per_session' enum('false','true') NOT NULL default 'false',
 'limit_type' enum('soft','hard') NOT NULL default 'soft',
 'bytes_in_avail' float NOT NULL default '0',
 'bytes_out_avail' float NOT NULL default '0',
 'bytes_xfer_avail' float NOT NULL default '0',
 'files_in_avail' int(10) unsigned NOT NULL default '0',
 'files_out_avail' int(10) unsigned NOT NULL default '0',
 'files_xfer_avail' int(10) unsigned NOT NULL default '0'
)ENGINE=MyISAM DEFAULT CHARSET=latin1;
CREATE TABLE 'quotatallies'(
 'name' varchar(30) NOT NULL default '',
 'quota_type' enum('user','group','class','all') NOT NULL default 'user',
 'bytes_in_used' float NOT NULL default '0',
 'bytes_out_used' float NOT NULL default '0',
 'bytes_xfer_used' float NOT NULL default '0',
```

```
 'files_in_used' int(10) unsigned NOT NULL default '0',
 'files_out_used' int(10) unsigned NOT NULL default '0',
 'files_xfer_used' int(10) unsigned NOT NULL default '0'
)ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

（6）为了提高安全性，在 ProFTPd 读取 MySQL 数据库时，不应使用 root 用户，通过以下方式在 MySQL 建立一个专门用于读取 proftpd 数据库中表的用户，在 MySQL 环境中执行以下语句，本章中让 proftpd 用户使用 redhat 作为密码。

```
grant select,insert,update,delete,create,drop,index,alter,create temporary tables, \
            lock tables on proftpd.* to proftpd@localhost Identified by "redhat";
flush privileges;
```

（7）由于在前面的安装中并没开启 ProFTPd 对 MySQL 的支持，因此使用如下命令编译 ProFTPd。

```
tar -xvjf proftpd-1.3.2a.tar.bz2
cd  /usr/src/proftpd-1.3.2a
./configure --with-modules=mod_sql:mod_sql_mysql:mod_quotatab:mod_quotatab_sql \
              --with-includes=/usr/include/mysql --with-libraries=/usr/lib/mysql \
              --sysconfdir=/etc/proftpd
make && make install
```

（8）使用如下方法制作 ProFTPd 在 RHEL/CentOS 中的启动脚本。

- 复制启动脚本例子文件。

```
cd /usr/src/proftpd-1.3.2a
cp contrib/dist/rpm/proftpd.init.d /etc/rc.d/init.d/proftpd
```

- 使用以下命令修改启动脚本权限。

```
chmod +x /etc/rc.d/init.d/proftpd
```

- 增加 proftpd 服务。

```
chkconfig --add proftpd
```

（9）修改/etc/proftpd/proftpd.conf 文件如下内容。

```
Group        nobody
#连接 MySQL 数据的用户名、数据库、密码
SQLConnectInfo proftpd@localhost proftpd  redhat
SQLAuthTypes Plaintext     #定义密码验证的方式
#定义从 Mysql 获取用户的资料
#用户的信息必须按照规定的顺序：表、用户名、密码、uid、gid、主目录、shell，
#信息的定义与 Mysql 中的列名相同，home 和 shell 可为空(NULL)
SQLUserInfo ftpusers userid passwd uid gid homedir shell
#定义从 Mysql 获取用户组的资料。
SQLGroupInfo ftpgroups groupname gid members
SQLAuthenticate users groups         #定义支持用户和用户组的校验方式
SQLNegativeCache on
SQLLogFile /var/log/proftpd.sql.log  #定义 SQL 的日志记录
SQLNamedQuery getcount SELECT "count from ftpusers where userid='%u'"
SQLNamedQuery getlastlogin SELECT "lastlogin from ftpusers where userid='%u'"
SQLNamedQuery updatelogininfo UPDATE                          \
          "count=count+1,host='%h',lastlogin=current_timestamp() \
          WHERE userid='%u'" ftpusers
SQLShowInfo PASS "230" "You've logged on %{getcount} times, last login    \
          at %{getlastlogin}"
SQLLog PASS updatelogininfo
QuotaDirectoryTally on      #开启磁盘配额功能
QuotaDisplayUnits "Mb"
QuotaEngine on
QuotaShowQuotas on
SQLNamedQuery get-quota-limit SELECT "name, quota_type, per_session,   \
       limit_type, bytes_in_avail,bytes_out_avail, bytes_xfer_avail,  \
       files_in_avail, files_out_avail, files_xfer_avail FROM quotalimits \
```

```
        WHERE name = '%{0}' AND quota_type = '%{1}'"
SQLNamedQuery get-quota-tally SELECT "name, quota_type, bytes_in_used,  \
        bytes_out_used,bytes_xfer_used, files_in_used, files_out_used,    \
        files_xfer_used FROM quotatallies \
        WHERE name = '%{0}' AND quota_type = '%{1}'"
SQLNamedQuery update-quota-tally UPDATE "bytes_in_used   \
        = bytes_in_used + %{0},bytes_out_used = bytes_out_used + %{1}, \
        bytes_xfer_used = bytes_xfer_used + %{2}, \
        files_in_used = files_in_used + %{3}, files_out_used =   \
        files_out_used + %{4}, files_xfer_used = files_xfer_used + %{5} \
        WHERE name = '%{6}' AND quota_type = '%{7}'" quotatallies
SQLNamedQuery insert-quota-tally INSERT      \
        "%{0}, %{1}, %{2}, %{3}, %{4}, %{5}, %{6}, %{7}" quotatallies
QuotaLimitTable sql:/get-quota-limit
QuotaTallyTable sql:/get-quota-tally/update-quota-tally/insert-quota-tally
```

（10）启动 ProFTPd 服务，并设置为下次启动自动加载。

```
service proftpd restart
chkconfig proftpd on
```

（11）在 MySQL 环境中执行以下语句，建立两个用于测试的虚拟用户（tonyzhang、davidxu，密码都是 redhat）。其中/home/tonyzhang、/home/davidxu 是用户默认的家目录。

```
INSERT INTO ftpusers VALUES ('tonyzhang', 'redhat', 1000, 1000, '/home/tonyzhang', \
        '/sbin/nologin',0,'','');
INSERT INTO ftpusers VALUES ('davidxu', 'redhat', 1000, 1000, '/home/davidxu', \
        '/sbin/nologin',0,'','');
```

该语句是往 ProFTPd 建立的虚拟用户表（ftpusers）中插入用户信息，该表在建立后的结构如图 3-2 所示，在该表中各字段作用如下。

- userid：虚拟用户名，这个是必填写字段。
- passwd：虚拟用户的密码，这个是必填写字段。
- uid：虚拟用户 uid，这个字段默认值为 1000。
- gid：虚拟用户 gid，这个字段默认值为 1000。
- homedir：虚拟用户家目录，这个是必填写字段。
- shell：虚拟用户是否能登录系统，这里默认的是不能登录，因为是虚拟用户，所以不能让虚拟用户来登录系统，所以默认是/sbin/nologin。
- count：虚拟用户访问次数，默认是 0。
- host：虚拟用户登录 FTP 服务器的 IP 地址记录，可以不设置，服务器会自动记录。
- lastlogin：虚拟用户最后登录时间，这个也是自动生成，服务器会自动记录。

（12）使用以下命令手动建立虚拟用户所需家目录。

```
mkdir /home/tonyzhang
mkdir /home/davidxu
chmod 777 /home/tonyzhang
chmod 777 /home/davidxu
```

（13）在 MySQL 环境下执行以下语句可以为 tonyzhang 用户增加一个 100MB 的磁盘配额的限制。

```
insert   into   quotalimits   VALUES ('tonyzhang','user','false','soft','104857600','0','0',
'0','0','0');
```

该语句是往 ProFTPd 建立的磁盘配额表（quotalimits）中插入配额信息，该表在建立后的结构如图 3-3 所示，在该表中各字段作用如下。

```
mysql> describe ftpusers;
+----------+------------------+------+-----+---------------+-------+
| Field    | Type             | Null | Key | Default       | Extra |
+----------+------------------+------+-----+---------------+-------+
| userid   | varchar(30)      | NO   | PRI |               |       |
| passwd   | varchar(80)      | NO   |     |               |       |
| uid      | int(10) unsigned | NO   |     | 1000          |       |
| gid      | int(10) unsigned | NO   |     | 1000          |       |
| homedir  | varchar(255)     | NO   |     |               |       |
| shell    | varchar(255)     | NO   |     | /sbin/nologin |       |
| count    | int(10) unsigned | NO   |     | 0             |       |
| host     | varchar(30)      | NO   |     |               |       |
| lastlogin | varchar(30)     | NO   |     |               |       |
+----------+------------------+------+-----+---------------+-------+
9 rows in set (0.00 sec)
```

图 3-2　ftpusers 表

```
mysql> describe quotalimits;
+------------------+-----------------------------------+------+-----+----------+-------+
| Field            | Type                              | Null | Key | Default  | Extra |
+------------------+-----------------------------------+------+-----+----------+-------+
| name             | varchar(30)                       | YES  |     | NULL     |       |
| quota_type       | enum('user','group','class','all') | NO   |     | user     |       |
| per_session      | enum('false','true')              | NO   |     | false    |       |
| limit_type       | enum('soft','hard')               | NO   |     | soft     |       |
| bytes_in_avail   | float                             | NO   |     | 0        |       |
| bytes_out_avail  | float                             | NO   |     | 0        |       |
| bytes_xfer_avail | float                             | NO   |     | 0        |       |
| files_in_avail   | int(10) unsigned                  | NO   |     | 0        |       |
| files_out_avail  | int(10) unsigned                  | NO   |     | 0        |       |
| files_xfer_avail | int(10) unsigned                  | NO   |     | 0        |       |
+------------------+-----------------------------------+------+-----+----------+-------+
10 rows in set (0.00 sec)
```

图 3-3　quotalimits 表

- name：可以是用户名或组名，如果在 quota_type（限额类型）中是 group 那这里设置组名；如果 quota_type（限额类型）中是 user 那这里设置组名；如果设置使用默认的 NULL，则针对所在 quota_type 中设置的类型，比如在 quota_type 中设置为 user，就是针对所有 ftpusers 中的用户起作用，如果是 group 名，也是对 ftpgroups 所有组作用。
- quota_type：磁盘限额类型，可以设置为 user、group，默认为 user 认证。
- limit_type：可以设置 hard、soft，默认为 soft。
- bytes_in_avail：用户占用空间大小，也就是家目录的空间最大可以让用户占用多少，单位是 byte，默认为 0（不受限制）。
- bytes_out_avail：所有下载文件的总和，单位是 byte，默认为 0。
- bytes_xfer_avail：一个用户上传下载流量总和，单位是 byte，默认为 0。
- files_in_avail：限制上传文件总数，默认为 0。
- files_out_avail：限制下载文件个数总计，默认为 0。
- files_xfer_avail：允许下载和上传的文件总和，默认为 0。

（14）在 MySQL 环境下执行以下语句可以将 tonyzhang、davidxu 加入到一个名为 hr 的虚拟用户组中。

```
insert into quotalimits VALUES ('tonyzhang','user','false','soft','104857600','0','0','0','0','0');
```

该语句是往 ProFTPd 建立的用户组表（ftpgroups）中插入用户组信息，该表在建立后的结构如图 3-4 所示，在该表中各字段作用如下。

- groupname：组名称。
- gid：组 gid，默认为 1000。
- members：该组成员。

2．虚拟主机

图 3-5 所示为通过 ProFTPd 实现虚拟主机的功能，只需要修改 ProFTPd 主配置文件中如下内容后，重新启动 ProFTPd 服务即可实现。

```
mysql> describe ftpgroups;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| groupname | varchar(30)  | NO   |     |         |       |
| gid       | int(11)      | NO   |     | 1000    |       |
| members   | varchar(255) | NO   |     |         |       |
+-----------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

图 3-4　ftpgroups 表

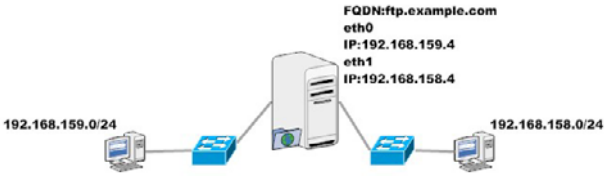


图 3-5　虚拟主机

```
DefaultServer            off
<VirtualHost 192.168.159.4>
     DefaultRoot ~
     #以下配置可以使虚拟主机 192.168.159.4 只允许匿名用户访问
     <Limit LOGIN>
         DenyAll
     </Limit>
     <Anonymous ~ftp>
         user ftp
         group ftp
```

```
            <Limit LOGIN>
                AllowAll
            </Limit>
        </Anonymous>
    </VirtualHost>
    <VirtualHost 192.168.158.4>
        #以下配置可拒绝匿名用户访问
         <Limit LOGIN>
            DenyUser ftp
        </Limit>
        #以下配置可以使虚拟主机192.168.158.4所有用户登录都被引导到/var/ftpsite目录
        DefaultRoot /var/ftpsite
    </VirtualHost>
```

3．FTPS 配置

FTPS 是一种多传输协议，相当于加密版的 FTP。FTPS 是在安全套接层使用标准的 FTP 协议和指令的一种增强型 FTP 协议，为 FTP 协议和数据通道增加了 SSL 安全功能。FTPS 也称作“FTP-SSL”和“FTP-over-SSL”。在 ProFTPd 下配置 FTPS 的操作步骤如下。

（1）由于在前面的安装中并没开启 ProFTPd 对 FTPS 的支持，因此使用如下命令安装 ProFTPd。

```
tar -xvjf proftpd-1.3.2a.tar.bz2
cd /usr/src/proftpd-1.3.2a
./configure --with-modules=mod_tls --sysconfdir=/etc/proftpd
make && make install
```

（2）建立一个用于存放证书的目录。

```
mkdir /etc/proftpd/.sslkey
```

（3）使用如下命令创建证书。在建立证书时需要输入相关信息，这些信息可根据需要输入，但 Common Name 必须是客户端访问 FTP 服务器时的 FQDN，如图 3-6 所示。

```
cd /etc/proftpd/.sslkey
openssl req -new -x509 -days 365 -nodes -out /etc/proftpd/.sslkey/proftpd.cert.pem   \
            -keyout /etc/proftpd/.sslkey/proftpd.key.pem
```

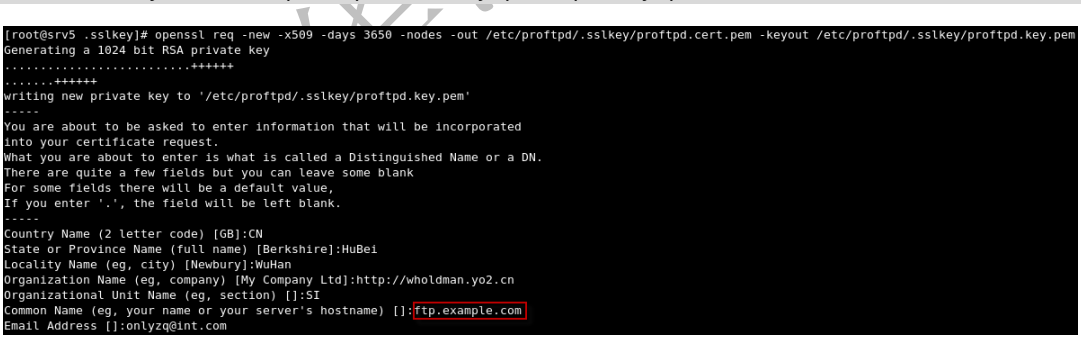


图 3-6　建立证书

（4）为了保证证书文件安全，可修改证书存放目录的权限。

```
chmod -R 400 /etc/proftpd/.sslkey
```

（5）修改/etc/proftpd/proftpd.conf 文件如下内容。

```
Group        nobody
<IfModule mod_tls.c>
     TLSEngine                    on
     TLSLog                   /var/log/proftpd/tls.log
     TLSProtocol              SSLv23
     TLSOptions               NoCertRequest
     TLSRSACertificateFile    /etc/proftpd/.sslkey/proftpd.cert.pem
     TLSRSACertificateKeyFile     /etc/proftpd/.sskey/proftpd.key.pem
```

```
        TLSVerifyClient             off
        TLSRequired             on
    </IfModule>
```

3.1.3 ProFTPd 防病毒

ProFTPd 提供了与常见防病毒软件的接口，当用户上传文件时 ProFTPd 可以调用防病毒软件对上传的文件进行病毒扫描。通过 ProFTPd 调用 ClamAV 对上传的文件进行病毒扫描的操作步骤如下。

ClamAV 全名 Clam AntiVirus（Clamav 官方网站：http://www.clamav.net/），是一个开源（GPL）杀毒软件包，这个软件最主要的目的是集成在 Linux 服务器，查杀邮件附件中的病毒。软件中主要包含一个灵活可升级的多线程后台程序、一个命令行扫描程序、一个自动升级程序。Clam AV 通过 rus 软件包以及同时发布的其他共享库文件进行病毒查杀，如果需要也可以在其他软件中使用这些共享库文件。ClamAV 本身是在字符界面下运行，但也有许多图形接口的前端工具可用，另外由于其开放源代码的特性，在 Windows 与 Mac OS 平台都有其移植版。

（1）下载 ClamAv 相关 RPM 包后，使用如下命令安装。

```
wget http://packages.sw.be/clamav/clamav-db-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-devel-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamd-0.95.3-1.el5.rf.i386.rpm
rpm -ivh *.rpm
```

（2）使用如下命令启动 clamd 服务，并设置为下次启动自动加载。

```
service clamd restart
chkconfig clamd on
```

（3）执行以下命令升级 ClamAV 病毒库，如图 3-7 所示。

```
freshclam
```

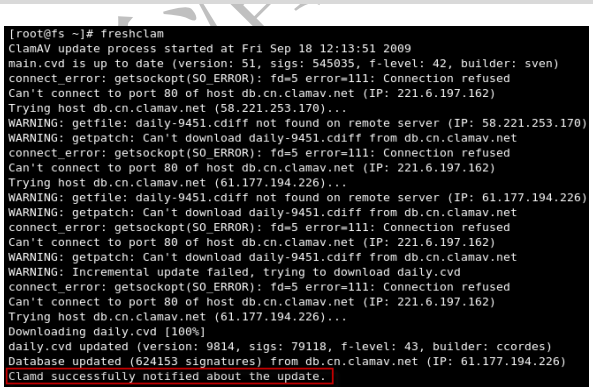


图 3-7　升级 ClamAV

（4）使用 crontab -e 命令增加如下自动化任务。

```
*/50 * * * * /usr/bin/freshclam --quiet --daemon
```

（5）使用如下命令安装相关软件包。

```
yum -y install pcre-devel gmp-devel
```

（6）下载 ProFTPd 的 ClamAV 插件并使用以下命令解压。

```
cd /usr/src
wget        https://secure.thrallingpenguin.com/redmine/attachments/download/1/mod_clamav-0.11rc.tar.gz
tar -xvzf mod_clamav-0.11rc.tar.gz
```

（7）由于在前面的安装中并没开启 ProFTPd 对 ClamAV 的支持，因此使用如下命令编译、安装 ProFTPd。

```
cd /usr/src
```

```
wget ftp://ftp.proftpd.org/distrib/source/proftpd-1.3.2a.tar.bz2
tar -xvjf proftpd-1.3.2a.tar.bz2
cd /usr/src/proftpd-1.3.2a
cp ../mod_clamav-0.11rc/mod_clamav.* proftpd-1.3.2a/contrib.
patch -p1 <../mod_clamav-0.11rc/proftpd.patch
./configure --with-modules=mod_clamav --sysconfdir=/etc/proftpd
make && make install
```

（8）使用如下方法制作 ProFTPd 在 RHEL/CentOS 中的启动脚本。

- 复制启动脚本例子文件。

```
cd /usr/src/proftpd-1.3.2a
cp contrib/dist/rpm/proftpd.init.d /etc/rc.d/init.d/proftpd
```

- 使用以下命令修改启动脚本权限。

```
chmod +x /etc/rc.d/init.d/proftpd
```

- 增加 proftpd 服务。

```
chkconfig --add proftpd
```

（9）修改/etc/proftpd/proftpd.conf 文件如下内容。

```
Group       nobody
<IfModule mod_clamav.c>
    ClamAV on
    ClamLocalSocket /var/run/clamav/clamd.sock
</IfModule>
```

（10）启动 ProFTPd 服务，并设置为下次启动自动加载。

```
service proftpd restart
chkconfig proftpd on
```

3.2 vsftpd：小巧轻快/安全的 FTP 服务器

vsftpd 是 Very Secure FTP（非常安全的 FTP）的缩写（vsftpd 官方网站：http://vsftpd.beasts.org/），是一个基于 GPL 发布的类 UNIX 系统上使用的 FTP 服务器软件。安全是 vsftpd 的开发者 Chris Evans 考虑的首要问题之一。vsftpd 目前已经被许多大型 FTP 站点所采用（比如 Red Hat、Suse、Debian、OpenBSD）。

3.2.1 vsftpd 安装

由于 vsftpd 的广泛应用，很多 Linux 的发行版都包括了 vsftpd 的软件包，一般没有特殊需求的情况下可以直接通过系统自带的 vsftpd 软件包括进行安装。在 RHEL/CentOS 5.4 中就包含有 vsftpd 的软件包，可直接使用以下命令安装 vsftpd。

```
yum -y install vsftpd
```

在安装 vsftpd 服务后，可通过“service vsftpd start|stop|restart|reload”命令将 vsftpd 服务启动、停止、重新启动、重新载入配置文件。

如果希望使用 vsftpd 最新版或者有一些特殊需求，也可以在 vsftpd 官方网站下载源码包后，编译安装。在一台安装了 RHEL/CentOS 5.4 的计算机上通过源码包编译安装 vsftpd 的操作步骤如下。

（1）使用如下命令建立 FTP 匿名用户使用目录。

```
mkdir /var/ftp
```

（2）使用如下命令下载并安装 vsftpd。

```
cd /usr/src
wget ftp://vsftpd.beasts.org/users/cevans/vsftpd-2.2.2.tar.gz
tar -xzvf vsftpd-2.2.2.tar.gz
cd vsftpd-2.2.2
make && make install
```

```
cp vsftpd.conf /etc/
```

（3）vsftpd 在编译安装完成后，默认采用超级服务的方式，可直接通过以下命令启动 vsftpd 服务。

```
service xinetd restart
chkconfig xinetd on
```

如果希望使用 System V 服务则通过以下方式将 vsftpd 改为 System V 服务。

（1）修改/etc/xinetd.d/vsftpd 文件内容如下。

```
service ftp
{
        socket_type         = stream
        wait                = no
        user                = root
        server                  = /usr/local/sbin/vsftpd
#       server_args         =
#       log_on_success      += DURATION USERID
#       log_on_failure      += USERID
        nice                = 10
        disable             = yes
}
```

（2）使用如下命令复制 vsftpd 配置文件模板。

```
cd /usr/src/vsftpd-2.2.2
cp vsftpd.conf /etc
```

（3）修改/etc/vsftpd.conf 文件，内容如下。

```
listen=YES
listen_port=21
```

如果使用超级服务方式，必须将/etc/vsftpd.conf 文件删除。

（4）执行以下命令启动 vsftpd 服务，并将其加入/etc/rc.d/rc.local 文件中。

```
/usr/local/sbin/vsftpd /etc/vsftpd.conf &
```

3.2.2 vsftpd 配置

在 vsftpd 服务安装完成后，默认情况下其配置文件位于/etc/vsftpd 目录（如果是使用源码包编译安装默认位于/etc 目录），其中主配置文件为 vsftpd.conf。vsftpd 的大多数功能都是通过修改主配置文件完成的。vsftpd 的配置文件与其他大多数 Linux 配置一样以井号“#”开始的是注释行（在执行时将被忽略），对大小写敏感，所有参数的配置形式均使用“参数=值”的方式。

vsftpd 作为一款在类 UNIX 平台中非常受欢迎的 FTP 服务，完全可以满足用户对 FTP 服务的各种要求，下面将介绍搭建生产环境中使用的 FTP 服务所需的常用功能在 vsftpd 中的实现方法。

1. 用户认证

当 FTP 客户端连接到 vsftpd 时，vsftpd 需要对客户端的身份进行验证，在 vsftpd 中可访问的用户有以下几种。

（1）匿名用户。

直接使用“ftp”或“anonymous”作为用户名访问 FTP 服务器。vsftpd 默认就允许匿名用户以只读方式（可以下载，但不能上传）登录。

（2）本地授权用户。

当用户访问 vsftpd 时，必须输入用户及密码。输入用户及密码来源于 vsftpd 所在主机中“/etc/passwd”中的用户名及密码。

（3）虚拟授权用户。

当用户访问 vsftpd 时，必须输入用户及密码。输入的用户及密码并不来自“/etc/passwd”文件，而是以其他形式保存，这其中包括本地数据文件、数据库或 LDAP 等。

相对于 FTP 的本地用户形式来说，虚拟用户只是 FTP 服务器的专有用户，虚拟用户只能访问 FTP 服务器所提供的资源，增强系统本身的安全性。相对于匿名用户而言，虚拟用户需要用户名和密码才能获取 FTP 服务器中的文件，增加了对用户和下载的可管理性。对于需要提供下载服务，但又不希望所有人都可以匿名下载又考虑到主机安全和管理方便的 FTP 站点来说，虚拟用户是一种很好的解决方案。

在本章中讲述通过以下 3 种方式实现授权用户访问 vsftpd 服务功能。

（1）使用系统用户访问。

vsftpd 在默认情况下允许系统账户（“/etc/passwd”）使用，只要本地内置的账户都可以登录到 FTP 服务器。

在一些情况下使用 FTP 服务的时候，只为了让用户通过 FTP 方式访问服务器，而不需要登录到系统，从安全方法考虑可以采用下面介绍的方法。如果大部分用户都可以登录系统，只需提供少量用户不能登录到系统但可以访问 FTP 服务器的时候，可以采用下述方法（在建立用户的时候指定使用 nologin 脚本）实现。

```
useradd tonyzhang -s /sbin/nologin
```

（2）通过本地数据文件实现虚拟用户访问。

通过本地数据文件实现虚拟用户访问时，首先需要手动建立一个文件，将所有用户和密码保存到该文件中，这种方法主要适用于用户比较少及变化不频繁的情况下。在 vsftpd 中配置本地数据文件实现虚拟用户访问的操作步骤如下。

（1）使用如下命令安装用于生成数据库的软件包 db4-utils。

```
yum -y install db4-utils
```

（2）使用如下命令创建本地映射用户，修改本地映射用户家目录权限。

```
useradd -d /var/ftp/ftpuserdir  -s /sbin/nologin ftpuserdir
chmod o=rwx /var/ftp/ ftpuserdir
```

（3）修改/etc/vsfptd/vsftpd.conf 文件，内容如下。

```
guest_enable=YES          #允许虚拟用户登录
guest_username= ftpuser  #将虚拟用户映射为本地的 ftpuser 用户
```

（4）生成虚拟用户文件，在该文件中用户及密码各一行。本章中是建立/etc/vsftpd/vftpuser.txt 文件，内容如下。

```
tonyzhang            #虚拟用户 1
123456           #虚拟用户 1 密码
davidxu          #虚拟用户 2
123456           #虚拟用户 2 密码
```

（5）生成虚拟用户数据文件。

```
db_load -T -t hash -f /etc/vsftpd/vftpuser.txt/etc/vsftpd/vftpuser.db
```

（6）为了提高安全性，应修改生成的用户数据文件权限。

```
chmod 600 /etc/vsftpd/vftpuser.db
```

（7）修改 PAM 认证文件/etc/pam.d/vsftpd，将原有内容注释并加入以下内容。通过以下两行的配置可以将认证用户及用户其他检查工作的数据来源改变为本地数据文件（/etc/vsftpd/vftpuser）。

```
auth    required     /lib/security/pam_userdb.so db=/etc/vsftpd/vftpuser
account required     /lib/security/pam_userdb.so db=/etc/vsftpd/vftpuser
```

（8）重新启动 vsftpd 后，即可使用 tonyzhang、davidxu 登录 FTP 服务器。

2．通过 MySQL 实现虚拟用户访问

当访问 FTP 服务器的虚拟用户数量非常多或变化比较频繁时（比如对 Internet 开放的 FTP 服务），使用上述的方法维护工作比较繁重。使用 MySQL 数据库的方式，可以发挥数据库操作灵活等优势。在 vsftpd 中配置 MySQL 实现虚拟用户访问的操作步骤如下。

（1）使用以下命令安装 MySQL。

```
yum -y install mysql-devel mysql-server
```

（2）启动 MySQL 服务，并设置为下次启动自动加载。

```
service mysqld restart
chkconfig mysqld on
```

（3）使用 mysqladmin 创建 MySQL 管理员及密码。

```
#建立名为 root 的 MySQL 管理员，并将密码设置为 redhat
mysqladmin -u root password redhat
```

（4）使用 root 用户登录 MySQL 数据库，建立用于保存虚拟用户的数据库、表并加入虚拟用户。本章中是建立了名为 tonyzhang、davidxu 的 2 个用户，将它们的密码都设置为 111，在设置密码时使用了 MySQL 的加密函数 password（④、⑤），这个可对存入数据库的用户密码进行加密，如图 3-8 所示。

其中各行含义如下。

①：建立用于保存虚拟用户的数据库，本章中为 vftpuser。

②：进入 vftpuser 数据库。

③：在数据库建立表，本章中为 users。

④、⑤：在表中插入 2 条用户信息的记录。

⑥：通过 select 语句查询用户信息是否被插入 users 表中。

（5）为了提高安全性，在 vsftpd 读取 MySQL 数据库时，不应使用 root 用户。通过以下方式在 MySQL 建立一个专门用于读取 vftpuser 数据库中 users 表的用户，在 MySQL 环境中执行以下语句，本章中让 vsqluser 用户使用 redhat 作为密码读取。

```
grant select on vftpuser.users to vsqluser@localhost identified by 'redhat';
flush privileges;
```

（6）测试 MySQL 用户 vsqluser 是否可以访问 vftpuser 数据库，如图 3-9 所示。

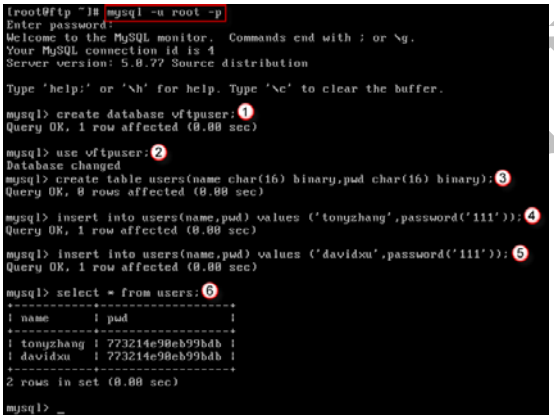


图 3-8　建立虚拟用户

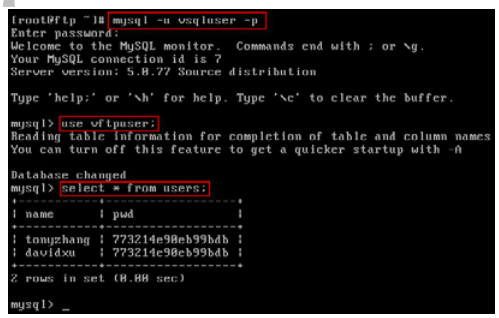


图 3-9　测试用户

（7）创建本地映射用户，修改本地映射用户家目录权限。

```
useradd -d /var/ftp/vuserdir  -s /sbin/nologin vuser
chmod o+rw /var/ftp/vuserdir
```

（8）修改/etc/vsfptd/vsftpd.conf 文件，内容如下。

```
guest_enable=YES      #允许虚拟用户登录
guest_username=vuser      #将虚拟用户映射为本地的 vuser 用户
```

（9）使用如下命令下载并安装基于 MySQL 的 PAM 认证模块。

```
cd /usr/src
wget http://nchc.dl.sourceforge.net/sourceforge/pam-mysql/pam_mysql-0.7RC1.tar.gz
tar -xvzf pam_mysql-0.7RC1.tar.gz
cd pam_mysql-0.7RC1
./configure
make && make install
```

（10）修改 PAM 认证文件/etc/pam.d/vsftpd，将原有内容注释并加入以下 2 行内容。

```
auth required /lib/security/pam_mysql.so user= vsqluser  \
          passwd=redhat host=localhost db=vftpuser table=users  \
          usercolumn=name passwdcolumn=pwd crypt=2
account required /lib/security/pam_mysql.so user= vsqluser passwd=redhat \
```

```
host=localhost db=vftpuser table=users usercolumn=name \
passwdcolumn=pwd crypt=2
```

通过以上 2 行的配置可以将认证用户及用户其他检查的工作数据来源改变为 MySQL，其中 user、passwd 是指定读取 MySQL 数据库时使用的用户名，host 是指定 MySQL 所在的主机，db、table 是指定存储用户信息的数据库和表，usercolumn、passwdcolumn 是指定表中存储用户名及密码的字段。

crypt 指定密码字段是以什么方式保存到数据库中，当 crypt = 0 时表示以明文保存密码，当 crypt = 1 时表示使用 crpyt()函数加密保存密码(对应 SQL 数据库里的 encrypt()函数)，当 crypt = 2 时表示使用 MySQL 中的 password()函数加密保存密码，当 crypt = 3 时表示使用 md5 的散列方式保存密码。

（11）重新启动 vsftpd 服务后即可使用 tonyzhang、davidxu 登录 FTP 服务器。

从使用本地数据库和使用 MySQL 数据库的配置方式可以看出，二者并没有多大区别，对于一般企业内部用户使用的 FTP 服务器，推荐使用本地数据文件验证，其他安全性比 MySQL 更高。

3．行为控制

当用户登录到 vsftpd 服务器时，如果使用系统用户则被引导到用户的家目录，如果是匿名用户则被引导到/var/ftp 目录，如果是使用虚拟用户则被引导到所映射的系统用户的家目录。在 vsftpd 中对用户使用 FTP 服务器时的行为进行控制的参数主要有以下几个。

（1）anonymous_enable = YES|NO：是否允许使用匿名账户登录。

（2）allow_anon_ssl = YES|NO：是否允许匿名账户通过 SSL 连接。

（3）local_enable = YES|NO：是否允许本地用户登录。

（4）write_enable = YES|NO：是否允许用户上传文件到 FTP 服务器，该参数只对非匿名用户有效。

（5）anon_upload_enable = YES|NO：是否允许匿名用户上传文件到 FTP 服务器。

（6）anon_mkdir_write_enable = YES|NO：是否允许匿名用户在 FTP 服务器上建立目录。

（7）anon_other_write_enable = YES|NO：是否允许匿名用户执行创建目录之外的写操作（如删除、重命名）。

（8）download_enable = YES|NO：是否允许用户下载文件。

（9）local_umask：授权用户上传文件的 umask，比如 local_umask = 022。

（10）anon_umask：匿名用户上传文件的 umask，比如 anon_umask = 022。

（11）chown_uploads = YES、chown_username = whoever：修改匿名用户上传文件的所有者，当 chown_uploads = YES 时，可通过 chown_username 指定一个系统用户，这样用户上传的所有文件所有者都被自动改为该系统用户。当然前题是 anonymous_enable = YES、anon_upload_enable = YES。

（12）ls_recurse_enable = YES|NO：是否允许用户在登录到 FTP 服务器后使用 ls-R 等比较占用系统资源的命令。

（13）dirlist_enable = YES|NO：是否允许使用 dir 之类的列目录命令。

（14）userlist_file：指定在 vsftpd 中，记录被允许或禁止登录用户的文件名。在文件中一行一个用户名。

（15）userlist_enable = YES|NO：该参数为 YES 时，vsftpd 将读取 userlist_file 参数所指定的文件中的用户列表，当列表中的用户登录 FTP 服务器时，该用户在提示输入密码之前就被禁止了（即该用户名输入后，vsftpd 检查到所输入的用户名在列表中，vsftpd 就直接禁止该用户，不会再进行询问密码等后续步骤）。

（16）userlist_deny = YES|NO：该参数决定禁止还是只允许由 userlist_file 指定文件中的用户登录 FTP 服务器（该参数只有在 userlist_enable = YES 时才生效）。当 userlist_deny = YES（默认值）时，禁止文件中的用户登录，同时也不向这些用户发出输入密码的提示，当 userlist_deny = NO 时只允许在文件中的用户登录 FTP 服务器。

（17）chroot_local_user = YES|NO：是否允许所有用户登录到 FTP 服务器离开自己的家目录。

（18）chroot_list_enable = YES|NO：是否允许指定不能离开家目录的用户，只有当 chroot_local_user = YES 时，chroot_list_enable = YES 才有效。

（19）chroot_list_file：指定不能离开家目录的用户，如 chroot_list_file = /etc/vsftpd/chroot_list，可将用户名一个一行写在该文件里，只有当 chroot_list_enable = YES 时，该参数才有效。

vsftp 在默认情况当用户通过 FTP 方式连接到服务器，允许用户离开自己的家目录，甚至允许用户进入“/etc/”这样一些重要的目录，强烈推荐使用 chroot_local_user = NO 禁止用户离开家目录或使用 chroot_list_enable 只允许特定用户离开家目录。

（20）local_root：指定所有用户的根目录。默认情况下 vsftpd 会根据登录用户的不同，引导到各自的家目录，通过 local_root 参数指定一个目录，比如 local_root = /home/ftp 后，所有登录的用户将被引导到 /home/ftp 目录，该参数对匿名用户无效。

（21）anon_root：指定匿名用户根目录。默认情况下 vsftpd 会将匿名用户引导到 ftp 用户的家目录，通过 anon_root 参数指定一个目录，比如 local_root = /ftp，匿名用户登录后将被引导到/ftp 目录。

（22）anon_max_rate：匿名用户的最大传输速度（单位：Byts/s）。

（23）local_max_rate：授权用户的最大传输速度（单位：Byts/s）。

（24）async_abor_enable = YES|NO：是否允许客户端使用 sync 等命令。

（25）ascii_upload_enable = YES|NO：是否在上传文件时使用 ASCII 传输模式。

（26）ascii_download_enable = YES|NO：是否在下载文件时使用 ASCII 传输模式。

（27）idle_session_timeout：指定会话超时（客户端连接到 FTP 但未操作）的时间（单位：秒）。

（28）data_connection_timeout：指定数据传输超时的时间（单位：秒）。

（29）deny_file：不允许上传的文件类型，比如 deny_file = {*.exe,*.dll}。

（30）pam_service_name = vsftpd：指定 vsftpd 使用 PAM 模块的配置文件，默认的 vsftpd 文件在/etc/pam.d 目录下，该文件的默认内容（如图 3-10 所示）主要指定使用系统用户作为认证来源。在①处，默认是可以指定允许或拒绝登录到 FTP 的用户或组，其中 item 指定允许或拒绝的是什么对象，可以为用户 user（用户）或 group（组）；sense 指定 allow（允许）或 deny（拒绝）的操作；file 指定文件，在该文件中定义组用户或组时一个一行。

```
session    optional    pam_keyinit.so    force revoke
auth       required    pam_listfile.so item=user sense=deny file=/etc/vsftpd/ftpusers onerr=succeed   ①
auth       required    pam_shells.so
auth       include     system-auth
account    include     system-auth
session    include     system-auth
session    required    pam_loginuid.so
```

图 3-10　/etc/pam.d/vsftpd

当用户登录到 FTP 服务器后如想上传文件，除了在 vsftpd 的配置文件中允许相应的上传操作外，该用户对目录的系统权限也必须是可写。比如当 write_enable = YES 时，用户以 tonyzhang 的用户名登录到 FTP 服务器后进入/home/tonyzhang/test 目录，但/home/tonyzhang/test 目录的所有者及拥有组均为 root，系统权限为 700，这时 tonyzhang 是无法上传文件的。

4．虚拟主机

在默认情况下，vsftpd 不像其他 FTP 服务器那样可以在同一台主机上建立多个 FTP 站点，不过并不是没有方法让 vsftpd 在同一台主机上建立多个 FTP 站点，具体操作步骤如下（在下述配置过程中使用如图 3-5 所示的网络拓扑）。

（1）创建虚拟 FTP 服务用户。

```
useradd -d /var/ftp2 -s /sbin/nologin ftp2
chmod -R 755 /var/ftp2
chown -R root:root /var/ftp2
mkdir -m 755 /var/ftp2/pub
```

```
chown ftp2:root /var/ftp2/pub
```

（2）准备虚拟 FTP 服务器的配置文件。

```
cp /etc/vsfptd/vsftpd.conf /etc/vsfptd/vsftpd1.conf
```

（3）修改/etc/vsftpd/vsfptd.conf 文件，内容如下。

```
listen_address=192.168.159.4
```

（4）修改/etc/vsftpd/vsfptd1.conf 文件，内容如下。

```
listen_address=192.168.158.4
ftp_username=ftp2
local_root=/var/ftp2
```

（5）重新启动 vsftpd 后即可使 192.168.159.0/24 和 192.168.158.0/24 两个网络的客户端访问不同的 FTP 站点。

5．客户端权限控制

在使用 FTP 服务器的过程中，一般根据用户进行权限的判断，比如说某些用户允许上传而某些用户只允许下载，但在一些特殊的情况下，希望权限客户端所处的网段进行这样的权限判断。vsftpd 本身没有提供这样的功能，但可以通过 TCP Wrappers 配合 vsftpd 实现。通过在如图 3-11 所示的网络拓扑实现只有客户端来自 192.168.159.0/24 网段时才可以上传数据的操作步骤如下。

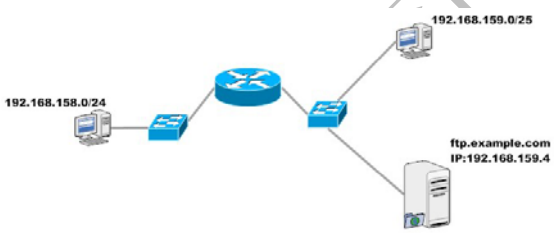


图 3-11　指定网段权限

TCP Wrappers（IP 层存取控制过滤）为由 inetd 生成的服务提供了增强的安全性。TCP Wrappers 软件扩展了 inetd 为受其控制的服务程序实施控制的能力。通过使用这种方法能够提供日志支持、返回消息给联入的连接、使得服务程序只接受内部连接等。尽管防火墙也能够完成其中的某些功能，但 TCP Wrappers 不仅增加了一层额外的保护，也提供了防火墙无法提供的功能。然而由 TCP Wrappers 提供的一些额外的安全功能，不应被视为好的防火墙的替代品。TCP Wrappers 应结合防火墙或其他安全加强设施一并使用，为系统多提供一层安全防护。

（1）修改/etc/hosts.allow 文件，内容如下。

```
vsftpd: 192.168.159.*:setenv VSFTPD_LOAD_CONF /etc/vsftpd/update.class
vsftpd: ALL:setenv VSFTPD_LOAD_CONF /etc/vsftpd/other.class
```

（2）在/etc/vsftpd 目录建立 update.class 文件后，加入如下内容。

```
write_enable=YES
anon_upload_enable=YES
anon_mkdir_write_enable=YES
```

（3）在/etc/vsftpd 目录建立 other.class 文件后，加入如下内容。

```
write_enable=NO
anon_upload_enable=NO
anon_mkdir_write_enable=NO
```

（4）修改/etc/vsftpd/vsftpd.conf 文件，内容如下。

```
#local_enable=YES
#write_enable=YES
```

```
userlist_deny=YES
```

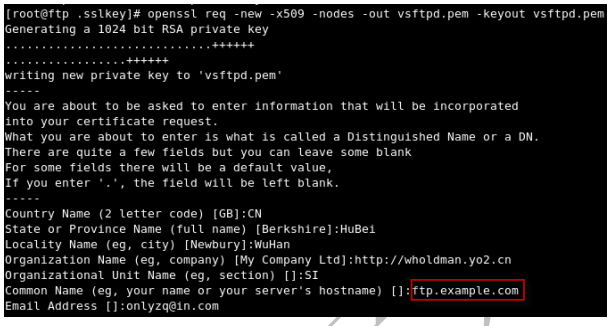
6. FTPS 配置

FTP 和 HTTP 一样在默认情况下使用明文进行数据传输，如果希望保证 FTP 服务器与客户端数据通信的安全，可以使用 SSL 保护其数据通信。在 vsftpd 2.0.1 以后的版本中提供了对 FTPS 功能的支持，这也使得 vsftpd 变得更加安全。通过 vsftpd 实现 FTPS 的操作步骤如下。

（1）建立一个用于存放证书的目录。

```
mkdir /etc/vsftpd/.sslkey
```

（2）使用如下命令创建证书。生成的 vsftpd.pem 文件中即包含私钥也包含证书。在建立证书时需要输入相关信息，这些信息可根据需要输入，但 Common Name 必须是客户端访问 FTP 服务器时的 FQDN，如图 3-12 所示。



```
[root@ftp .sslkey]# openssl req -new -x509 -nodes -out vsftpd.pem -keyout vsftpd.pem
Generating a 1024 bit RSA private key
.............................++++++
................++++++
writing new private key to 'vsftpd.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:CN
State or Province Name (full name) [Berkshire]:HuBei
Locality Name (eg, city) [Newbury]:WuHan
Organization Name (eg, company) [My Company Ltd]:http://wholdman.yo2.cn
Organizational Unit Name (eg, section) []:SI
Common Name (eg, your name or your server's hostname) []:ftp.example.com
Email Address []:onlyzq@in.com
```

图 3-12　vsftpd 生成证书

```
cd /etc/vsftpd/.sslkey
openssl req -new -x509 -nodes -out vsftpd.pem -keyout vsftpd.pem
```

（3）为了保证证书文件安全，可修改证书存放目录的权限。

```
chmod -R 400 /etc/vsftpd/.sslkey
```

（4）修改/etc/vsftpd/vsftpd.conf 文件，内容如下。

```
ssl_enable=YES                                ①
ssl_sslv2=YES                                     ②
ssl_sslv3=YES                                     ③
ssl_tlsv1=YES                                     ④
force_local_logins_ssl=YES                        ⑤
force_local_data_ssl=YES                      ⑥
rsa_cert_file=/etc/vsftpd/.sslkey/vsftpd.pem  ⑦
```

其中各行含义如下。

①：指定 vsftpd 支持加密协议。

②：指定 vsftpd 支持安全套接字层 v2。

③：指定 vsftpd 支持安全套接字层 v3。

④：指定 vsftpd 支持 tls 加密方式 v1。

⑤、⑥：指定 vsftpd 强制非匿名用户使用加密登录和数据传输，如果配置为 NO 时，则用户可以选择加密，也可以不加密。

⑦：指定证书的存放路径。

如果在 vsftpd 官方网站下载最新版源码包进行编译安装时，需要将源码包中 builddefs.h 文件的#undef VSF_BUILD_SSL 改为#define VSF_BUILD_SSL(#号必须保留)。这样编译的 vsftpd 才支持 FTPS。

重新启动 vsftpd 后，FTPS 配置完成。FTP 客户端软件连接 FTPS 服务器的操作步骤如下（以 CuteFTP 为例）。

（1）打开 CuteFTP 后，在“文件”→“新建”→“FTPS（SSL）站点”，如图 3-13 所示。

（2）在“一般”标签页输入 FTP 服务器 IP 地址（或 FQDN）、用户名、密码等信息，如图 3-14 所示。

（3）在“类型”标签页选择协议类型为“使用 TLS/SSL 进行 FTP（AUTH TLS-显示）”，如图 3-15 所示。

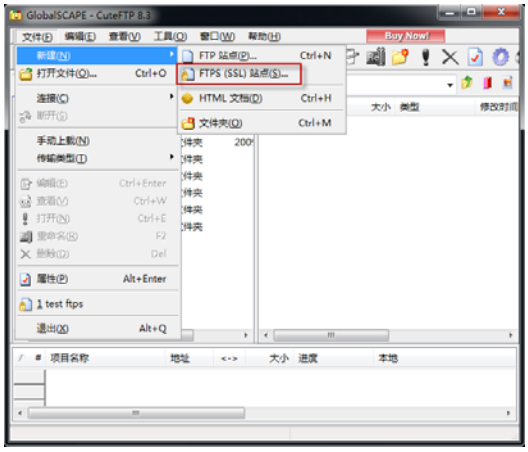


图 3-13　连接 FTPS 站点

图 3-14　输入 FTPS 服务器信息

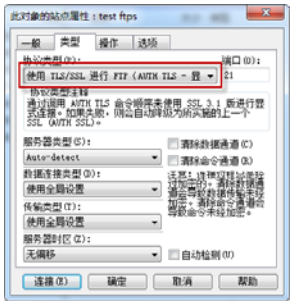


图 3-15　选择协议类型

（4）单击“连接”按钮即可连接到 FTPS 站点，如图 3-16 所示。

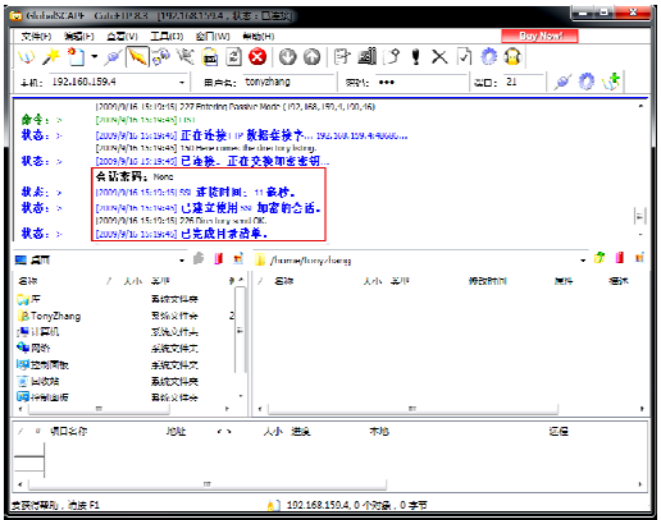


图 3-16　连接 FTPS 成功

由于 vsftpd 使用的是自颁发证书，因此在连接过程中会出现如图 3-17 所示的警告，单击“接受”按钮即可。

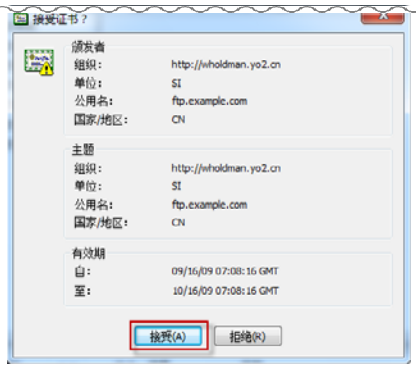


图 3-17　证书警告

7．其他配置

在 vsftpd 中除了上述参数以外，还可以通过以下参数对其运行方式进行调整。

（1）listen_address：指定 vsftpd 侦听的 IP 地址，当 vsftpd 有多个 IP 地址时，可通过该参数让 vsftpd 只接受某个 IP 地址侦听到的 FTP 请求。

（2）listen_port：指定 vsftpd 侦听的端口，默认 TCP 的 21 端口。

（3）max_clients：vsftpd 允许的最大连接数，比如 max_client = 3000。

（4）max_per_ip：vsftpd 允许相同 IP 的最大连接数，比如 max_per_ip = 10。

（5）use_localtime = YES|NO：是否在目录列表中使用本地时间。

（6）ftpd_banner：指定登录到 FTP 服务时显示的欢迎信息（并不是所有访问 FTP 服务器的方式都支持欢迎信息的，比如通过 IE 浏览器访问时就不会显示这些信息）。

（7）dirmessage_enable = YES|NO：是否当用户在 FTP 服务器切换目录时显示欢迎信息，当 dirmessage_enable = YES 时，可以在每个目录下建立一个名为.message 文件，在该文件中写入欢迎信息。

（8）banner_fail：指定当连接失败时显示的信息，比如 banner_fail = /etc/vsftpd/errinfo，当有用户连接失败后，会显示 errinfo 文件中的内容。

（9）xferlog_enable = YES|NO：是否在用户上传/下载文件时记录日志。

（10）nopriv_user：指定 vsftpd 服务的运行账户，默认为 ftp。

（11）xferlog_file：指定使用的日志文件。

（12）xferlog_std_format = YES|NO：是否使用标准日志文件来记录日志。

（13）listen = YES|NO：开启 IPv4 支持。

（14）listen_ipv6 == YES|NO：开启 IPv6 支持。

（15）tcp_wrappers = YES|NO：是否允许 tcp_wrappers 管理。

（16）connect_from_port_20 = YES|NO：是否使用 20 端口传输数据。

（17）pasv_min_port、pasv_max_port：指定被动模式时，客户端的数据连接端口范围。比如 pasv_min_port = 50000、pasv_max_port = 70000。

3.3 Samba：功能强大的文件打印共享服务器

1991 年 Andrew Tridgwell 编写了 Samba 这个自由软件（Samba 官方网站：http://www.samba.org），只要在类 UNIX 上启用 Samba 服务，类 UNIX 系统就好像成为了一台 Windows 系统，可利用 SMB 协议与 Windows 系统之间实现资源共享等相关功能。

SMB 又称为 Common Internet File System（CIFS），是由微软公司开发的一种软件程序级的网络传输协议，主要用来使得一个网络上的机器共享计算机文件、打印机、串行端口和通信等资源。SMB 也提供了认证的进程间通信机能。SMB 最初是 IBM 的贝瑞·费根鲍姆研制的，其目的是将 DOS 操作系统中的本地文件接口“中断 13”改造为网络文件系统。后来微软对这个发展进行了重大更改，这个更改后的版本也是最常见的版本。微软将 SMB 协议与其和 3Com 一起发展的网络管理程序结合在一起，并在 Windows for Workgroups 和后来的 Windows 版本中不断加入新的功能。

Samba 的出现彻底解决了类 UNIX 与 Windows 之间的资源共享与访问问题，Samba 以其简洁、实用、灵活配置、功能强大的特点受到越来越广泛的关注。也是因为这个原因几乎所有的类 UNIX 系统都可以使用 Samba 服务。Samba 是作为类 UNIX 系统和 Windows 的通信的桥梁，在设计上是让类 UNIX 系统加入到 Windows 网络中，而不是让 Windows 加入类 UNIX 网络中。

3.3.1 Samba 安装

由于 Samba 的广泛应用，很多 Linux 的发行版都包括了 Samba 的软件包括，一般没有特殊需求的情况下可以直接通过以下命令安装 Samba 软件包。

```
yum -y install samba
```

在安装 Samba 服务后，可通过“service smb start|stop|restart|reload”命令将 Samba 服务启动、停止、重新启动、重新载入配置文件。

3.3.2 Samba 配置

在 Samba 服务安装完成后其主配置文件是/etc/samba/smb.conf，该文件内容非常多，由以下几部分组成。

1. Global Settings

以“[global]”标识开始。该部分主要涉及 Samba 服务的全局配置，比如用户映射、引用子配置文件等，在该标签内的配置会影响整个 Samba 服务器。

（1）username map：指定用于定义用户映射关系的文件。

（2）use sendfile：当客户端访问 Samba 服务器读取文件时，默认情况首先由 kernel 读取数据，然后将数据传送给 Samba 服务，再将 Samba 服务回传给 kernel，最后再由 kernel 发送给客户端。当 use sendfile = yes 时，将直接由 kernel 读取数据后发送给客户端，大大提高效率。该参数默认为 use sendfile = no。

（3）max connections：设置同时允许访问 Samba 服务器或 Samba 服务器某一共享资源的客户端数量，该参数放在此部分是一个全局配置，对 Samba 服务器所有共享资源有效，如果该参数位于定义某共享资源的标识内时，只对这一个共享资源有效。如果全局设置与某一共享资源在该参数发生冲突时，共享资源内的该参数值优先。在下面的例子中，share1 同时允许 5 个客户端连接，而 share2 则同时允许 10 个客户端连接。

```
[global]
    workgroup = MYGROUP
    server string = Samba Server
    max connections = 5
[share1]      #该共享资源同时客户端的访问数由上面的 max connections 决定
    comment = This is smb share1
    path = /share1
[share2]      #该共享资源同时客户端的访问数由下面的 max connections 决定
    comment = This is smb share2
    path = /share2
    max connections = 10
```

2. Network Related Options

该部分包括 Samba 服务器接受网络访问相关配置，如图 3-18 所示。

（1）workgroup：设置 Samba 服务器所在的工作组或所在域的名称，默认设置工作组名称为“MYGROUP”。

（2）server string：设置 Samba 服务器的说明文字，用于描述 Samba 服务器。其默认值为“Samba Server Version %v”，其中%v 是 Samba 服务器预设变量，表示当前 Samba 版本。由于有可能在 Samba 发行版本中存在一些安全漏洞，所以推荐不要使用该变量让客户端访问时获得有关 Samba 版本的信息。

（3）netbios name：设置 Samba 服务器的 NetBIOS 名。

（4）interface：设置允许 Samba 服务侦听的本地网络接口。比如 Samba 服务器有多个网络接口时可通过此参数指定只接受指定接口的 SMB/CIFS 请求，该参数默认没有启用（被注释）。在编辑该参数时当使用 interface 时推荐保留 lo（表示本地回环地址），在指定网络接口时，除了通过 IP 地址指定外，也可直接通过网络接口名。如图 3-18 所示内容中的 eth0 就表示侦听该网络接口接收到的请求，不考虑该网络接口的 IP 地址及请求客户端的 IP 地址是多少。

（5）bind interfaces only：当 bind interfaces only = yes 时，interface 只对 Samba 服务器提供文件服务有效，对浏览服务的广播无效，也就是说 Windows 客户端可以在网上邻居中查看到 Samba 服务器，但不能访问。该参数默认为 bind interfaces only = no。

（6）hosts allow：指定允许连接到 Samba 服务器客户端。该参数默认没有启用。当然对应的还有 hosts deny 参数，指定拒绝连接到 Samba 服务器客户端。

```
59 # ----------------------- Network Related Options -------------------------
60 #
61 # workgroup = NT-Domain-Name or Workgroup-Name, eg: MIDEARTH
62 #
63 # server string is the equivalent of the NT Description field
64 #
65 # netbios name can be used to specify a server name not tied to the hostname
66 #
67 # Interfaces lets you configure Samba to use multiple interfaces
68 # If you have multiple network interfaces then you can list the ones
69 # you want to listen on (never omit localhost)
70 #
71 # Hosts Allow/Hosts Deny lets you restrict who can connect, and you can
72 # specifiy it as a per share option as well
73 #
74         workgroup = MYGROUP
75         server string = Samba Server Version %v
76
77 ;       netbios name = MYSERVER
78
79 ;       interfaces = lo eth0 192.168.12.2/24 192.168.13.2/24
80 ;       hosts allow = 127. 192.168.12. 192.168.13.
81
```

图 3-18　Network Related Options

3．Logging Options

该部分包括 Samba 服务器运行时如何记录日志相关配置，如图 3-19 所示。

```
82 # --------------------------- Logging Options -----------------------------
83 #
84 # Log File let you specify where to put logs and how to split them up.
85 #
86 # Max Log Size let you specify the max size log files should reach
87
88         # logs split per machine
89 ;       log file = /var/log/samba/%m.log
90         # max 50KB per log file, then rotate
91 ;       max log size = 50
92
```

图 3-19　Logging Options

（1）log file：定义 Samba 日志文件，其默认值是为每一个与服务器连接的客户端定义一个单独的日志文件。此参数可根据 Samba 服务器预设变量灵活设置。

（2）log level：设置日志记录等级从 0 到 10。等级 0 时日志记录最简单，等级 10 时日志记录最详细。

（3）max log size：设置最大的日志文件的大小（单位为 KB）。

在 Standalone Server Options、Domain Members Options、Domain Controller Options 3 部分均需要配置合适的 security（安全级别，用于配置 Samba 服务器的认证方式）和 passdb backend 用户密码存放方式，其中 security 可用参数包括 share（可匿名方式访问 Samba 服务器共享资源，该方式只对 Windows 客户端有效，一般不推荐使用）、user（访问 Samba 服务器共享资源需要输入用户密码，认证用户来源为本机）、server（访问 Samba 服务器共享资源需要输入用户密码，认证用户来源为另一台 Samba 服务器或 Windows 服务器）、domain（如果 Samba 服务器在一个基于 Windows NT 平台的 Windows 域中，访问共享资源需要输入用户密码，认证用户来源为 Windows 域）和 ads（如果 Samba 服务器在一个基于 Windows200x 平台的 Windows 活动目录中，访问共享资源需要输入用户密码，认证用户来源为 Windows 活动目录）；passdb backend 常见可用参数包括 smbpasswd（该方式是使用 Samba 提供的工具 smbpasswd 来给系统用户设置一个 Samba 密码，客户端就用这个密码来访问 Samba 的资源，如果使用此方式还需加一个 smb passwd file = /etc/samba/smbpasswd 用于指定保存用户密码文件，该文件需要手动建立，不推荐使用这种方法）、tdbsam（该方式则是使用一个数据库文件来建立用户数据库。数据库文件名为 passdb.tdb，默认在/etc/samba 目录下。passdb.tdb 用户数据库可以使用 smbpasswd 来建立 Samba 用户）、ldapsam（该方式则是基于LDAP的账户管理方式来验证用户。首先要建立LDAP服务）、mysql

（该方式则是将 Samba 服务器用户密码估计到 mysql 数据库中）。Standalone Server Options、Domain Members Options、Domain Controller Options 3 部分均与 Samba 的认证方式及工作角色相关。

4．Standalone Server Options

使用独立服务器作为 Samba 服务器认证用户来源，也就是当访问 Samba 服务器时输入的用户密码的验证工作由 Samba 服务器本机系统内置账户完成，如图 3-20 所示。

```
# ---------------------- Standalone Server Options ------------------------
#
# Scurity can be set to user, share(deprecated) or server(deprecated)
#
# Backend to store user information in. New installations should
# use either tdbsam or ldapsam. smbpasswd is available for backwards
# compatibility. tdbsam requires no further configuration.

        security = user
        passdb backend = tdbsam

```

图 3-20　Standalone Server Options

（1）security：在此部分中该参数只能设置为 share（不推荐）、user 或 server（不推荐），当该参数设置为 share 时，客户端在连接到 Samba 服务器可不需要输入用户密码，但能不能访问某个共享资源还要根据共享资源中是否允许匿名用户访问。

（2）passdb backend：默认为 tdbsam，一般不用修改，除非想使用老版本 Samba 服务器的 smbpasswd 文件方式。

5．Domain Members Options

将 Samba 服务器加入 Windows NT 平台域或 Windows 2000 Server/Windows Server 2003 活动目录中，也就是当访问 Samba 服务器时输入的用户密码的验证工作由域控制器完成，如图 3-21 所示。

```
# ---------------------- Domain Members Options ------------------------
#
# Security must be set to domain or ads
#
# Use the realm option only with security = ads
# Specifies the Active Directory realm the host is part of
#
# Backend to store user information in. New installations should
# use either tdbsam or ldapsam. smbpasswd is available for backwards
# compatibility. tdbsam requires no further configuration.
#
# Use password server option only with security = server or if you can't
# use the DNS to locate Domain Controllers
# The argument list may include:
#   password server = My_PDC_Name [My_BDC_Name] [My_Next_BDC_Name]
# or to auto-locate the domain controller/s
#   password server = *


;       security = domain
;       passdb backend = tdbsam
;       realm = MY_REALM

;       password server = <NT-Server-Name>

```

图 3-21　Domain Members Options

（1）security：在此部分中该参数只能设置为 domain、ads。

（2）passdb backend：默认为 tdbsam，一般不用修改。

（3）password server：指定进行身份验证的域控制器 IP 地址或主机名。

6．Domain Controller Options

将 Samba 服务器配置为一台域控制器，如图 3-22 所示。

（1）security：在此部分中该参数只能设置为 user。

（2）passdb backend：默认为 tdbsam，一般不用修改。

（3）domain master：让 Samba 作为主域控制器（PDC），在此部分此参数必须为 yes。

```
130 # ---------------------- Domain Controller Options ------------------------
131 #
132 # Security must be set to user for domain controllers
133 #
134 # Backend to store user information in. New installations should
135 # use either tdbsam or ldapsam. smbpasswd is available for backwards
136 # compatibility. tdbsam requires no further configuration.
137 #
138 # Domain Master specifies Samba to be the Domain Master Browser. This
139 # allows Samba to collate browse lists between subnets. Don't use this
140 # if you already have a Windows NT domain controller doing this job
141 #
142 # Domain Logons let Samba be a domain logon server for Windows workstations.

143 #
144 # Logon Scrpit let yuou specify a script to be run at login time on the client
145 # You need to provide it in a share called NETLOGON
146 #
147 # Logon Path let you specify where user profiles are stored (UNC path)
148 #
149 # Various scripts can be used on a domain controller or stand-alone
150 # machine to add or delete corresponding unix accounts
151 #
152 ;       security = user
153 ;       passdb backend = tdbsam
154
155 ;       domain master = yes
156 ;       domain logons = yes
157
158         # the login script name depends on the machine name
159 ;       logon script = %m.bat
160         # the login script name depends on the unix user used
161 ;       logon script = %u.bat
162 ;       logon path = \\%L\Profiles\%u
163         # disables profiles support by specifing an empty path
164 ;       logon path =
165
166 ;       add user script = /usr/sbin/useradd "%u" -n -g users
167 ;       add group script = /usr/sbin/groupadd "%g"
168 ;       add machine script = /usr/sbin/useradd -n -c "Workstation (%u)" -M -d /nohome -s /bin/false "%u"
169 ;       delete user script = /usr/sbin/userdel "%u"
170 ;       delete user from group script = /usr/sbin/userdel "%u" "%g"
171 ;       delete group script = /usr/sbin/groupdel "%g"
```

图 3-22　Domain Controller Options

主域控制器（PDC）是一个 Windows NT 平台的概念，在 Windows 2000 Server（或 Windows Server 2003/2008）活动目录中由活动目录为 PDC 模拟器的操作主机代替，其主要功能包括管理来自客户端（Windows NT/95/98）的密码更改、最小化密码变化的复制等待时间、同步整个域内所有域控制器上的时间、收集分发活动目录中主浏览服务器列表信息。

（4）domain logons：允许旧的 Windows 客户端提交验证信息，只有网络环境中存在 Windows 9x 客户端，此参数必须为 yes。

（5）logon script：当用户登录到域时执行的启动脚本（相当于 Windows 组策略中用户开机脚本）。

（6）logon path：当用户登录到域后的配置文件存放位置，用这个来初始化工作环境（相当于 Windows 中漫游配置文件）。

（7）logon script：当域中客户端主机的开机启动脚本（相当于 Windows 组策略中计算机开机脚本）。

（8）add user script：指定同步 Windows 与 Linux 中用户信息同步脚本，当 Windows 域中新建用户后指定脚本会将该用户的信息复制到 Linux 中。

（9）add group script：指定同步 Windows 与 Linux 中组信息同步脚本，当 Windows 域中新建组后指定脚本会将该组信息复制到 Linux 中。

（10）add machine script：指定同步 Windows 与 Linux 中计算机信息同步脚本，当 Windows 域中新加入域中后指定脚本会将该计算机信息复制到 Linux 中。

（11）delete user script：指定同步 Windows 与 Linux 中用户信息同步脚本，当 Windows 域中删除用户后指定脚本会将该用户的信息复制到 Linux 中。

（12）delete user from group script：指定同步 Windows 与 Linux 中用户信息同步脚本，当 Windows 域中将用户从组中删除后指定脚本会将该信息复制到 Linux 中。

（13）delete group script：指定同步 Windows 与 Linux 中用户信息同步脚本，当 Windows 域中删除组后指定脚本会将该组的信息复制到 Linux 中。

7．Browser Control Options

配置浏览服务器，如图 3-23 所示。

```
# ----------------------- Browser Control Options ------------------------
#
# set local master to no if you don't want Samba to become a master
# browser on your network. Otherwise the normal election rules apply
#
# OS Level determines the precedence of this server in master browser
# elections. The default value should be reasonable
#
# Preferred Master causes Samba to force a local browser election on startup
# and gives it a slightly higher chance of winning the election
;       local master = no
;       os level = 33
;       preferred master = yes
```

图 3-23　Browser Control Options

（1）local master：是否允许 Samba 服务器作为主浏览服务器。

（2）os level：该数字越大被选举成为主浏览服务器可能性越高。

（3）preferred master：当 yes 时被选为主浏览服务器可能性越高。

主浏览服务器功能主要是实现 Windows 中的网上邻居。计算机浏览服务是一系列分布式的含有可用的网络资源列表，这些列表分布在一些计算机上，提出浏览请求的计算机充当浏览工作站，而提供浏览列表的计算机充当浏览服务器。当运行 Windows 中网上邻居时，将会显示域和计算机的显示列表。该操作通过计算机从同一子网中的主浏览服务器获得浏览列表副本完成。网络上的大部分计算机均为非浏览浏览器，但运行浏览服务的计算机可作为每个子网潜在的浏览器。理论上来讲，网络上的每台计算机都可以作为主浏览服务器提供浏览列表，但这样一来会造成浏览工作站提出查询请求时，众多计算机同时向浏览工作站提供浏览列表，产生过多的网络流量，降低了网络的性能，同时也会增加 CPU 的负担。为了减轻网络和计算机 CPU 的负担，同时为了方便对资源列表进行管理，就需要对提供资源浏览服务的计算机定义各种角色，以便明确分工，各负其责，尽量减少重复无益的流量产生。浏览服务器有域主浏览服务器、主浏览服务器、备份浏览服务器、潜在浏览服务器、非浏览服务器几种。选举主浏览服务器时，主域控制器（PDC）有主浏览器的优先权。但当一个计算机不能定位主浏览器时，或具备更优先条件的计算机开机时，或主域控制器启动时，选举过程可以简化为如下几步。

- 选举是通过发广播来实现的，如果哪个计算机的选举条件比自身收到的报文要好，则将广播自己的选举条件，收到别人的选举条件后每个计算机根据自己在域中的角色延迟不等的时间后再做反应，这样能减少选举条件较差的计算机发送选举报文。
- 当一个计算机选举成为主浏览器并且其浏览列表是空时，将广播一个请求通知的报文，强迫所有的计算机必须在 30s 内给予答复，这个 30s 的时间是为了防止服务器过载或报文丢失。
- 除了承担主浏览器和备份浏览器任务的计算机外，其他计算机将向主浏览器周期性地发布通知，告知自己是可利用的资源。这个时间开始是 1 分钟、2 分钟、4 分钟、8 分钟，以后就是每隔 12 分钟一次。
- 如果某个计算机关机了，主浏览器连续 3 个周期也就是 36 分钟没有收到该计算机的消息，将认定其不可用，并从浏览列表中删除。但是还留在备份浏览器的计算机里，备份浏览器每隔 15 分钟呼叫主浏览器一次以获得更新的网络资源列表，也就是说不可用的资源最多要等到 36 + 15 = 51 分钟后才会从网上彻底消失。这就是为什么有的计算机改了名，但旧名字依旧留在网上一段时间的原因。

8．Name Resolution

该部分包括 Samba 服务器名称解析方法相关配置（如图 3-24 所示），在这部分中可以设置的参数如下。

```
188 #----------------------------- Name Resolution -----------------------------
189 # Windows Internet Name Serving Support Section:
190 # Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
191 #
192 # - WINS Support: Tells the NMBD component of Samba to enable it's WINS Server
193 #
194 # - WINS Server: Tells the NMBD components of Samba to be a WINS Client
195 #
196 # - WINS Proxy: Tells Samba to answer name resolution queries on
197 #   behalf of a non WINS capable client, for this to work there must be
198 #   at least one         WINS Server on the network. The default is NO.
199 #
200 # DNS Proxy - tells Samba whether or not to try to resolve NetBIOS names
201 # via DNS nslookups.
202
203 ;       wins support = yes
204 ;       wins server = w.x.y.z
205 ;       wins proxy = yes
206
207 ;       dns proxy = yes
208
```

图 3-24　Name Resolution

WINS（Windows Internet Name Server，Windows 网络名称服务）是由微软公司开发的一种名字解析协议，为 NetBIOS 名字提供名字注册、更新、释放、转换及解析等服务。

（1）wins support：设置 nmbd 进程支持 WINS 服务器。

（2）wins server：设置 WINS 服务器 IP 地址。

（3）wins proxy：Samba 服务器是否可作为 WINS 代理。

（4）dns proxy：Samba 服务器是否在无法联系 WINS 服务器时通过 DNS 去解析主机的 NetBIOS 名。

（5）name cache timeout：设置 Samba 服务器解析主机名缓存的保存时间，单位是秒。该参数默认值为 name cache timeout=660。

（6）name resolve order：设置 Samba 服务器名称解析的方法及顺序。该参数可设置为 lmhosts（使用/etc/samba/lmhosts 文件对 NetBIOS 名称与 IP 地址对应关系进行解析，此方式是用于解析 NetBIOS 名）、host（使用主机名方式解析 IP 地址，该方式可使用 NIS、DNS 及/etc/hosts 文件 3 种方法完成解析，这 3 种方式的使用顺序是在/etc/nsswitch.conf 文件中 hosts 参数定义的）、wins（使用 WINS 服务器进行名称解析，使用此方式时 wins server 参数必须已指明 WINS 服务器的 IP 地址）、bcast（使用广播方式进行名称解析），也可以同时指定多个值。该参数默认为 name resolve order：lmhosts host wins bcast，也就说 Samba 服务器名称解析的顺序为 lmhosts、host、wins、bcast，在一种方式解析不到 IP 地址时，自动选择下一种。

当在 Samba 服务器中某个参数定义的计算机名是 fileserver 时，按默认的名称解析顺序，会先查看 lmhosts 文件有没有对应的 NetBIOS 名，如果没有再通过主机名方式解析。这里要注意了 fileserver 并不是一个合法的 FQDN，Samba 服务器会根据/etc/resolve.conf 文件 search 参数的定义将 fs 补全为一个合法的 FQDN。当在 Samba 服务器中某个参数定义的计算机名是 fs.example.com 时，按默认的名称解析顺序，会先查看 lmhosts 文件有没有对应的 NetBIOS 名，这时会将一个 FQDN 的最左边作为 NetBIOS 名进行解析，即 fs。

9．Printing Options

该部分包括 Samba 服务器打印机相关配置（如图 3-25 所示），在这部分中可以设置的参数如下。

```
209 # ----------------------------- Printing Options -----------------------------
210 #
211 # Load Printers let you load automatically the list of printers rather
212 # than setting them up individually
213 #
214 # Cups Options let you pass the cups libs custom options, setting it to raw
215 # for example will let you use drivers on your Windows clients
216 #
217 # Printcap Name let you specify an alternative printcap file
218 #
219 # You can choose a non default printing system using the Printing option
220
221         load printers = yes
222         cups options = raw
223
224 ;       printcap name = /etc/printcap
225         #obtain list of printers automatically on SystemV
226 ;       printcap name = lpstat
227 ;       printing = cups
228
```

图 3-25 Printing Options

（1）load printers：是否允许打印机设置文件中的所有打印机在引导时自动加载浏览列表，以支持客户端的浏览功能。

（2）cups options：配置为 raw 表示可以使用 Windows 客户端的打印驱动。printcap name：设置获取打印机描述信息的文件位置，该参数默认设置为 /etc /printcapFile。

10．Filesystem Options

该部分包括 Samba 服务器如何保留从 Windows 客户端复制或移动到 Samba 服务器共享目录文件的 Windows 文件属性的相关配置（如图 3-26 所示），在这部分中可以设置的参数如下。

```
229 # ---------------------------- Filesystem Options ----------------------------
230 #
231 # The following options can be uncommented if the filesystem supports
232 # Extended Attributes and they are enabled (usually by the mount option
233 # user_xattr). Thess options will let the admin store the DOS attributes
234 # in an EA and make samba not mess with the permission bits.
235 #
236 # Note: these options can also be set just per share, setting them in global
237 # makes them the default for all shares
238
239 ;       map archive = no
240 ;       map hidden = no
241 ;       map read only = no
242 ;       map system = no
243 ;       store dos attributes = yes
244
```

图 3-26 Filesystem Options

（1）map archive：当 Windows 客户端将文件复制或移动到 Samba 服务器共享目录时，是否保留文件在 Windows 中的存档属性，当 map archive = yes 时，将保留；当 map archive = no 时，将不保留。默认为 map archive = yes。不过如果 store dos attributes = yes 时，Samba 服务器将忽略该参数的设置。

（2）map hidden：当 Windows 客户端将文件复制或移动到 Samba 服务器共享目录时，是否保留文件在 Windows 中的隐藏文件属性，当 map hidden = yes 时，将保留；当 map hidden = no 时，将不保留。该参数没有默认值，也就是在不设置该参数时，是否保留隐藏文件属性根据 store dos attributes 的值决定。

（3）map read only：当 Windows 客户端将文件复制或移动到 Samba 服务器共享目录时，是否保留文件在 Windows 中的只读属性，当 map read only = yes 时，将保留；当 map read only = no 时，将不保留。默认为 map read only = yes。不过如果 store dos attributes = yes 时，Samba 服务器将忽略该参数的设置。

（4）map system：当 Windows 客户端将文件复制或移动到 Samba 服务器共享目录时，是否保留文件在 Windows 中的系统文件属性，当 map system = yes 时，将保留；当 map system = no 时，将不保留。默认为 map system = no。不过如果 store dos attributes = yes 时，Samba 服务器将忽略该参数的设置。

（5）store dos attributes：当 Windows 客户端将文件复制或移动到 Samba 服务器共享目录时，是否保留文件在 Windows 中的相关属性（只读文件、系统文件、隐藏文件、存档属性），当 store dos attributes = yes 时，将保留；当 store dos attributes = no 时，将不保留。默认为 store dos attributes = no。

11．Share Definitons

该部分主要涉及 Samba 服务器需要共享的资源（如图 3-27），这部分中默认已配置用户家目录（从“[home]”标识开始）、打印机共享（从“[printers]”标识开始）、登录脚本及登录域中有关用户家目录的配置，用户自定义配置信息也在此部分定义。

```
247 #========================== Share Definitions ==============================
248
249 [homes]
250         comment = Home Directories
251         browseable = no
252         writeable = yes
253 ;       valid users = %S
254 ;       valid users = MYDOMAIN\%S
255
256 [printers]
257         comment = All Printers
258         path = /var/spool/samba
259         browseable = no
260 ;       guest ok = no
261 ;       writeable = no
262         printable = yes
263
264 # Un-comment the following and create netlogon directory for Domain Logons
265 ;       [netlogon]
266 ;       comment = Network Logon Service
267 ;       path = /var/lib/samba/netlogon
268 ;       guest ok = yes
269 ;       writable = no
270 ;       share modes = no
271
272
273 # Un-comment the following to provide a specific roving profile share
274 # the default is to use the user's home directory
275 ;       [Profiles]
276 ;       path = /var/lib/samba/profiles
277 ;       browseable = no
278 ;       guest ok = yes
279
280
281 # A publicly accessible directory, but read only, except for people in
282 # the "staff" group
283 ;       [public]
284 ;       comment = Public Stuff
285 ;       path = /home/samba
286 ;       public = yes
287 ;       writable = yes
288 ;       printable = no
289 ;       write list = +staff
```

图 3-27　Share Definitons

下面将介绍具体的实现方法。

1．资源共享

在编辑 Samba 主配置文件时 smb.conf 文件中并不要求参数缩进，但推荐在编写时对参数进行缩进，这样便于以后阅读及修改。对于用户自定义的共享资源配置内容放在 smb.conf 文件的最后，这样也是为了便于以后阅读及修改。

在全局配置中的内容主要是针对 Samba 服务自身相关状态。在这些工作完成后就需要配置希望共享的资源，而这部分配置参数非常多，在本节中只是简单配置一个共享实现对一个目录资源的共享，其他有关参数及其作用将在后续章节中讲述。

Samba 配置共享目录的语法如下。

```
[共享名]                ①
    comment = 描述      ②
    path = 本地目录路径  ③
```

其参数含义如下。

①：客户端访问 Samba 服务器时浏览到的目录名，该名称不要求与本地目录名相同，但在当前 Samba 服务器必须唯一。

②：客户端访问 Samba 服务器时浏览到的目录描述信息，该参数不是必须的。

③：需要共享的本地目录，必须使用绝对路径。

下面通过一个例子来看一下使用 Samba 服务器简单的共享一个目录。

（1）在根目录下建立一个名为 share 的目录。

（2）在文件/etc/samba/smb.conf 尾部加入如下内容。

```
[share]
    comment = This is smb share
    path = /share
```

（3）使用 testparm 命令测试 smb.conf 配置是否正确，如果配置正确应出现如图 3-28 所示提示。

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

[global]
        workgroup = MYGROUP
        server string = Samba Server Version %v
        passdb backend = tdbsam
        cups options = raw

[homes]
        comment = Home Directories
        read only = No
        browseable = No

[printers]
        comment = All Printers
        path = /var/spool/samba
        printable = Yes
        browseable = No

[share]
        comment = This is smb share
        path = /share
```

图 3-28　testparm 执行结果

2．访问控制

Samba 可以根据访问的客户端来源进行访问控制，Samba 的访问控制通过 hosts allow（配置允许访问的客户端）、hosts deny（配置拒绝访问的客户端）两个参数实现。

在 Samba 中使用 hosts allow、hosts deny 参数时，该参数可以出现在全局配置部分（如图 3-29 所示），用于允许或拒绝可连接到 Samba 服务器的客户端，也可以出现在具体的共享资源配置中（如图 3-30 所示），用于允许或拒绝可访问该资源的客户端。如果在全局配置部分的 hosts allow、hosts deny 与具体共享资源的配置发生冲突时会怎么样呢？通过 Samba 的工作过程不难看出 Samba 客户端首先要可以连接到 Samba 服务器，才能访问其共享资源，所有全局配置部分的 hosts allow、hosts deny 优先级与具体共享资源的配置发生冲突时使用以下规则。

（1）当全局配置中 hosts deny 指定客户端无法访问 Samba 服务器任何共享资源。

（2）当全局配置中 hosts allow 指定客户端，分以下几种情况。

① 如具体共享资源中只指定了 hosts deny 且与全局配置不冲突时，客户端可以访问具体共享资源。

② 如具体共享资源中只指定了 hosts allow 且是全局配置的子集时，只有具体共享资源中指定的客户端可以访问。

```
#======================= Global Settings =====================================

[global]

# ----------------------- Network Related Options -------------------------
#
# workgroup = NT-Domain-Name or Workgroup-Name, eg: MIDEARTH
#
# server string is the equivalent of the NT Description field
#
# netbios name can be used to specify a server name not tied to the hostname
#
# Interfaces lets you configure Samba to use multiple interfaces
# If you have multiple network interfaces then you can list the ones
# you want to listen on (never omit localhost)
#
# Hosts Allow/Hosts Deny lets you restrict who can connect, and you can
# specifiy it as a per share option as well
#
        workgroup = workgroup
        server string = Samba Server

;       netbios name = MYSERVER

;       interfaces = lo eth0 192.168.12.2/24 192.168.13.2/24
        hosts allow = 127. 192.168.12. 192.168.13.
        hosts deny = 192.168.1.
```

图 3-29　hosts allow、hosts deny 出现在[global] 标签

```
[smbtest]
        comment = This is test
        path = /test
        hosts allow = 192.168.1.
        hosts deny = 129.168.1.3
```

图 3-30　hosts allow、hosts deny 出现在共享资源

③ 如具体共享资源中即指定了 hosts allow 又指定了 hosts deny 时，首先根据下面有关 hosts allow 与 hosts deny 生效规则得出具体共享资源允许或拒绝的客户端，再根据上面两条规则得出最终的结果。

如果全局配置内或具体共享资源内的 hosts allow 与 hosts deny 发生冲突时会使用以下规则。

（1）如果 hosts deny 与 hosts allow 发生冲突时，hosts allow 优先。

（2）如果只有 hosts allow，除了 hosts allow 中指定的客户端外其他所有客户端都不能访问。

（3）如果只有 hosts deny，除了 hosts deny 中指定的客户端外其他所有客户都可以访问。

在 Samba 中使用 hosts deny 及 hosts allow 进行访问控制，可以采用以几种方式表示客户端。

（1）使用 IP 地址控制。

在 hosts allow 及 hosts deny 时，可通过使用 IP 地址精确允许或拒绝特定客户端访问 Samba 服务器，下面看几个例子。

① 不允许 IP 地址为 192.168.159.200 的客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = 192.168.159.200
```

② 只允许 IP 地址为 192.168.259.250 的客户端访 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = / share
    hosts allow = 192.168.159.250
```

③ 下面的例子中，IP 地址为 192.168.159.200 的客户端可以访问 Samba 服务器上的 share 目录吗？当然是允许访问，为什么呢，因为在 Samba 中 hosts allow 比 hosts deny 优先级要高。

```
[share]
    comment = This is smb share
    path = /share
    hosts allow = 192.168.159.200
    hosts deny = 192.168.159.200
```

（2）使用网段控制。

在 hosts allow 及 hosts deny 时，可通过使用子网允许或拒绝特定客户端访问 Samba 服务器，在表示子网时可以使用 192.168.159.0/24、192.168.159.或 192.168.159.0/255.255.255.0 表示 192.168.159.0 子网掩码 24 位子网。下面看几个例子。

① 不允许 192.168. 159.0/24 所有客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = 192.168.159.
```

② 只允许 192.168.159.0/24 所有客户端访 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts allow = 192.168.159.
```

③ 不允许 192.168.159.0/24 但不包括 192.168.159.200 的客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = 192.168.159.
    hosts allow = 192.168.159.200
```

④ 只允许 192.168.159.0/24 但不包括 192.168. 159.200 的客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts allow = 192.168.159. EXCEPT 192.168.159.200  #EXCEPT 参数表示不包括
```

（3）使用域名控制。

在 hosts allow 及 hosts deny 时，可通过使用域名允许或拒绝特定客户端访问 Samba 服务器，在表示域名时可以使用 FQDN 表示某个具体的客户端或用域名表示某个域的所有客户端。下面看几个例子。

① 不允许 FQDN 为 client1.example.com 的客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = client1.example.com.
```

② 只允许 example.com 域的所有客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts allow =.example.com
```

③ 不允许 example.com 区域但不包括 192.168.159.200 的客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = .example.com
    hosts allow = 192.168.0.200
```

④ 只允许 example.com 但不包括 IP 地址为 192.168.159.200 的客户端访问 Samba 服务器上 smbtest 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts allow = .example.com EXCEPT 192.168.159.200
```

（4）使用通配符控制。

在 hosts allow 及 hosts deny 时，可通过使用通配符代表特定客户端集。可以使用的通配符主要有：ALL 表示所有客户端，*表示任何个字符，？表示一个字符，LOCAL 表示本地计算机。下面看几个例子。

① 拒绝除了 192.168.159.200 及 192.168. 159.250 以外的所有客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = ALL
    hosts allow = 192.168.159.200 192.168.159.250  #多个匹配条件区用空格分隔
```

② 除了 192.168.159.0/24 网段的客户端（不包括 192.168.159.200）以外拒绝所有客户端访问 Samba 服务器上 share 目录。

```
[share]
    comment = This is smb share
    path = /share
    hosts deny = ALL
    hosts allow = 192.168.159. EXCEPT 192.168.159.200
```

对于使用 hosts allow 及 hosts deny 的各种形式，看一个综合例子：只允许 192.168.159.0/24、192.168.158.0/24 及 192.168.157.0/24 连接到 Samba 服务器，只允许.example.com 域，但不包括 192.168.159.200 的客户端访问名为 share 的共享目录，只允许 192.168.158.0/24 的客户端访问名为 share1 的共享。

（1）在“[global]”标签下加入如下参数。

```
[global]
    hosts allow = lo 192.168.159. 192.168.1. 192.168.158.
```

（2）在具体共享资源中加入如下参数。

```
[share]
    comment = This is smb share
    path = /share
    hosts allow = .example.com EXCEPT 192.168.159.200
[share1]
    comment = This is smb share1
    path = /share1
    hosts allow = 192.168.158.
```

3．用户认证

客户端在访问时Samba服务器根据全局配置中的参数选择合适的认证方式以及认证用户来源进行认证。Samba 服务器认证用户来源可以基于 Samba 服务器本身的用户信息，也可以基于 Windows 活动目录中的用户信息。

虽然 Samba 服务器运行在 Linux 平台，但 Samba 服务器认证用户来源并不能直接读取/etc/passwd 中的用户及密码信息。如果希望 Samba 服务器认证用户信息来自其所在系统中的用户需要通过 smbpasswd 命令将系统用户添加到 Samba 服务中。图 3-31 所示内容中通过 smbpasswd 命令将系统中现有用户 tonyzhang 添加到 Samba 认证用户中，此处被添加的用户必须是在系统中存在，也就是在/etc/passwd 中已有的用户，如果是一个系统中不存在的用户就会出现如图 3-32 所示的错误。在使用 smbpasswd 需要为用户设置一个密码，这个密码仅在客户端访问 Samba 服务器时有效，与用户登录系统密码无关。

```
[root@fs samba]# smbpasswd -a tonyzhang
New SMB password:
Retype new SMB password:
Added user tonyzhang.
```

图 3-31　smbpasswd 执行正确

```
[root@fs samba]# smbpasswd -a tomwu
New SMB password:
Retype new SMB password:
Failed to modify password entry for user tomwu
```

图 3-32　smbpasswd 执行错误

如果希望某人可以访问 Samba 服务器的共享资源时，必须让他知道访问 Samba 的用户及密码，从系统安全角度来说这样是存在一些安全隐患的，因为他至少已经知道了一个 Linux 系统中的用户名（访问 Samba 的密码是使用 smbpasswd 设置的，不是该用户登录系统的密码）。Samba 服务器也考虑到了这一点，通过 Samba 服务器提供的用户映射功能可以很好地解决这个问题。

用户映射功能实际就是给系统用户在 Samba 服务器中起一个别名，当访问 Samba 服务器时用户输入一个别名，这样就无法得知 Linux 系统中的用户名。配置用户映射功能使用以下两个步骤。

（1）通过/etc/samba/smbusers 文件设置用户映射关系。smbusers 文件在安装 Samba 服务器端时已默认配置了 root、nobody 的用户映射关系，如图 3-33 所示。在该文件第一行的注释中已标明该文件的语法。假设已有两个系统用户 tonyzhang 及 davidxu 通过 smbpasswd 添加到 Samba 服务器，希望将 tonyzhang 映射为 tony 或 zhangqin，将 davidxu 映射为 david 则在 smbusers 文件中加入以下两行。

```
# Unix_name = SMB_name1 SMB_name2 ...
root = administrator admin
nobody = guest pcguest smbguest
```

图 3-33　/etc/samba/smbusers 文件

```
tonyzhang = tony zhangqin
davidxu = david
```

（2）修改 Samba 主配置文件。在“[global]”标签下加入如下参数。

```
[global]
    username map = /etc/samba/smbusers
```

在 Samba 的配置文件默认会将用户的家目录共享，如图 3-34 所示，虽然已经将系统用户 tonyzhang 映射为 tony，但是当使用 tony 浏览时会查看到 tonyzhang 的家目录，如果不需要用户使用其家目录的共享，可将图 3-35 中所示的代码改为注释。

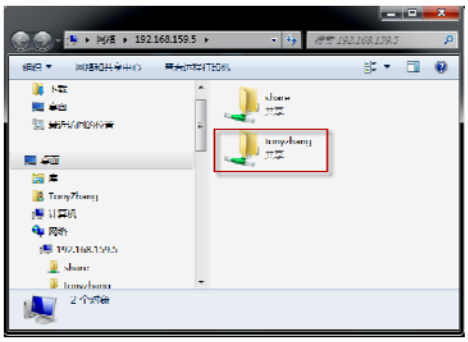


图 3-34　Samba 默认共享用户家目录

```
247 #========================= Share Definitions =====
248
249 [homes]
250         comment = Home Directories
251         browseable = no
252         writable = yes
253 ;       valid users = %S
254 ;       valid users = MYDOMAIN%S
255
```

图 3-35　smb.conf 中共享用户家目录配置

4. 行为控制

在 Samba 服务器的认证用户来源已建立完成，接下来很自然的就需要根据认证用户来源获得的用户信息，对特定用户使用共享资源的权限进行配置。

在 Windows 中共享文件夹可以针对不同的用户或组进行权限控制，在 Samba 服务器中也同样有相应的功能，而且有很多 Samba 服务器权限控制的功能 Windows 在默认情况下都无法实现。

在使用了 Samba 用户映射功能后，所有涉及用户的参数，必须使用系统用户名。比如通过/etc/samba/smbusers 文件将用户 tonyzhang 映射为 tony 后，在用户权限设置时还是需要使用 tonyzhang。

（1）用户浏览权限控制。

通过 browseable 参数可控制用户浏览权限，当 browseable = yes 时，将显示共享资源，当 browseable = no 时，将隐藏共享资源，默认为 browseable = yes。共享资源被隐藏后只是在浏览 Samba 服务器时不可见，与可不可以访问该共享资源无关。如下面的例子中 share 设置为隐藏共享，通过 Windows 客户端或 smbclient-L 浏览时查看不到，如图 3-36 所示，但还是可以通过直接输入共享资源名称访问，如图 3-37 所示（此功能与 Windows 中创建共享文件夹时在共享名后加$功能相同）。

```
[share]
    comment = This is smb share
    path = /share
    browseable = no
```

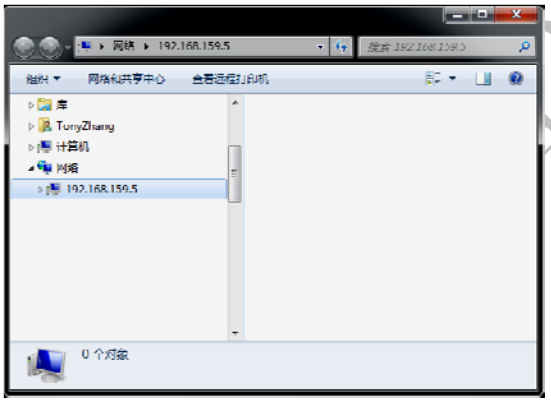


图 3-36　隐藏共享时无法查看

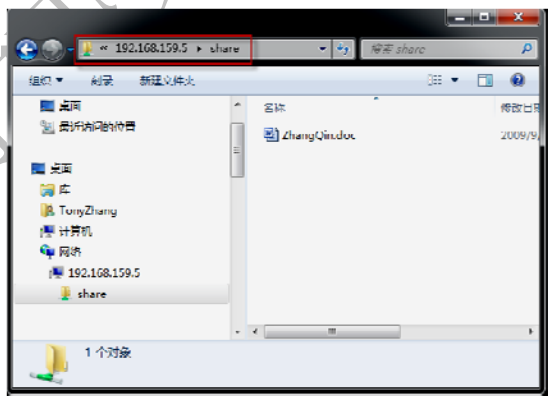


图 3-37　隐藏共享时直接输入共享名

（2）用户访问权限控制。

用户在成功访问 Samba 服务器后，能否访问某个共享资源由以下几个参数决定。

- public：设置是否允许匿名用户访问共享资源，当 public = yes 时表示允许，public = no 时表示不允许，默认为 public = no。在 Samba 服务器中 guest ok 与 public 有相同的功能。当在一个共享资源参数中 guest ok 与 public 参数发生冲突时，在后面的参数优先。如下面的例子中就不允许匿名用户访问 share。

```
[share]
    comment = This is smb share
    path = /share
    guest ok = yes
    public = no
```

- valid users：设置允许访问共享资源的用户或组（如果使用组时，需要在组名称前加@）。设置该参数后，未指定的所有用户将不能访问共享资源。

```
[share]
    comment = This is smb share
```

```
        path = /share
        valid users = tonyzhang        #允许 tonyzhang 用户访问该共享目录
        valid users = @sales           #允许 sales 组的所有用户访问该共享目录
    [share1]
        comment = This is smb share1
        path = /share1
        invalid users = tonyzhang      #不允许 tonyzhang 用户访问该共享目录
        invalid users = @sales    #不允许 sales 组的所有用户访问该共享目录
```

- wide links：设置否允许共享外符号连接。如某共享资源内有个连接指向非共享资源的文件或目录，如果设置 wide links = yes 将使该连接可用，如果设置 wide links = no 将使该连接不可用。

（3）共享资源可用性。

通过 available 参数定义共享资源是否可用。下面例子中 share 客户端可正常使用，但 share1 客户端将无法使用。

```
    [share]
        comment = This is smb share
        path = /share
        available = yes
    [share1]
        comment = This is smb share1
        path = /test1
        available = no
```

（4）用户读写权限控制。

用户在允许访问某共享资源的情况下，可以通过 Samba 对共享资源读写权限进行控制。在 Windows 中对一个位于 NTFS 分区中的共享文件夹有怎样的操作权限，由该文件夹的 NTFS 权限及共享权限共同起作用，两种权限更严格的优先。如某用户对一个共享文件夹的 NTFS 权限为完全控制，共享权限为只读，那么这个用户通过网络访问该共享文件夹时的权限为只读。在 Samba 服务器也是这样，通过网络访问 Samba 服务器资源时，能否读写是通过文件或目录自身在文件系统中的权限与该文件或目录在 Samba 服务器中的配置共同决定。如果希望某用户可以读或者写 Samba 服务的共享资源，首先要正确配置该用户对文件或目录自身文件系统中的权限。在图 3-38 所示的例子中，从 Samba 服务的配置来看，客户端使用 tonyzhang 用户通过认证后，应该是可以访问 share 的共享目录的，但是请注意 share 目录自身文件系统中的权限中其他用户是无读取权限的。通过测试会发现 tonyzhang 用户无法访问这个共享目录。

图 3-38 Samba 权限与系统权限结合

配置目录自身权限以配合 Samba 服务器读写权限时可以直接使用 chmod 配置目录其他用户权限或通过 ACL 配置指定用户或组权限。通过 chmod 配置比较方便，但不够精细，因为所有通过 Samba 服务器访问的客户端 Linux 都会认为是其他用户，通过 ACL 配置时相对复杂一些，但对权限控制更加精细，所以推荐使用 ACL 配置。

用户能否写某个共享资源由以下几个参数决定。

- readonly：是否将共享资源设置为只读。当 readonly = yes 时表示只读共享，readonly = no 时表示不使用只读方式共享。
- read list：设置只读的用户或组（如果使用组时，需要在组名称前加@）。

```
    [share]
        comment = This is smb share
        path = /share
        #允许 tonyzhang 及 davidxu 用户访问该共享目录时只读
        read list = tonyzhang davidxu
        read list = @sales        #允许 sales 组的所有用户访问该共享时只读
```

- writable：是否允许共享资源设置为可写。当 writable = yes 时表示可写，writable = no 时表示不可写。

- write list：设置可写的用户或组（如果使用组时，需要在组名称前加@）。

```
[share]
    comment = This is smb share
    path = /share
    #允许 tonyzhang 及 davidxu 用户访问该共享目录时可写
    write list = tonyzhang davidxu
    write list = @sales        #允许 sales 组的所有用户访问该共享时可写
```

- force user：指定通过 Samba 服务器访问共享资源建立的文件或目录时的拥用者。下面的例子中任何有写权限的用户在通过 Samba 服务器访问共享资源时，建立的文件或目录的拥有组均为 tonyzhang。

```
[share]
    comment = This is smb share
    path = /share
    force user = tonyzhang
```

- group：指定通过 Samba 服务器访问共享资源建立的文件或目录时的拥有组。下面的例子中任何有写权限的用户在通过 Samba 服务器访问共享资源时，建立的文件或目录的拥有组均为 sales。这个功能还可通过使用 SGID 实现。

```
[share]
    comment = This is smb share
    path = /share
    readonly = no
    group = sales
```

当 readonly、read list、writable 及 write users 在对某一共享资源的设置发生冲突时，使用以下规则。

① readonly、writeable 发生冲突时，在后面的参数优先。

② readonly、write list 发生冲突时，除 write list 指定用户可写外，其他用户只读。

③ read list、writable 发生冲突时，除 read list 指定用户只读外，其他用户可写。

④ read list、write list 发生冲突时，write list 优先。

⑤ writeable = no 时，write list 的配置无效。

⑥ 同时配置了 writeable = yes 时 write list 时 writeable = yes 无效。

当 writeable、write list 及 readonly 同时出现在一个共享资源时，最后的结果可根据上面提到的几个规则来判定是否可写。下面来看一个综合的例子，在这个例子中假设/share 及/share1 的自身权限为 777。

```
[share]
    comment = This is smb share
    path = /share
    writeable = yes
    readonly = yes
    write list = tonyzhang davidxu
[share1]
    comment = This is smb share1
    path = /share1
    readonly = yes
    writeable = yes
    write list = @sales
```

当以 tonyzhang 或 davidxu 身份访问 share 时，是可写的。只有 sales 组的成员访问 share1 时是可写。

（5）上传文件类型控制。

在 Windows Server 2003 R2 之前，Windows 无法阻止特定文件上传到文件服务器，而 Samba 服务器可以通过 veto files 参数阻止客户端上传含有特定关键字的文件或目录到 Samba 服务器共享资源。在参数值中可以使用“*”或“?”作为通配符，使用时必须通过“/”进行转义。当参数放在“[global]”标签时，是一个全局配置，对 Samba 服务器所有共享资源有效，如果该参数位于定义某共享资源的标识内时，只对这

一个共享资源有效。如果全局设置与某一共享资源在该参数发生冲突时，共享资源内的该参数值优先。在下面的例子中，客户端不允许在 share 中上传含有“root”关键字的文件或目录，在 share1 中则是不允许上传含有“.mp3”或“.avi”关键字的文件或目录。

```
[global]
    veto files = /*root*/
 [share]
    comment = This is smb share
    path = /share
    writable = yes

[share1]
    comment = This is smb share1
    path = / share1
    writable = yes
    veto files = /*.mp3/*.avi*/
```

当配置 veto files 参数后，可以阻止客户端上传含有特定关键字的文件或目录到 Samba 服务器共享资源（可以通过 hide files 参数隐藏指定类型的文件，配置方法与 veto files 参数相同），但对于配置该参数之前已经存在于共享资源的含有特定关键字的文件或目录到 Samba 服务器共享资源，可以通过 delete veto files 参数处理。当 delete veto files = yes 时，所有满足 veto files 参数条件的文件或目录将被删除，当 delete veto files = no 时，将允许已经存在与共享资源的含有特定关键字的文件或目录，这也是默认值。

（6）使用子配置文件。

在前面讲述过 browseable 可以实现类似与 Windows 隐藏共享的功能，但是如果将该参数配置为 browseable = yes 时，所有用户都无法浏览到，如果希望某个共享资源只允许特定的用户浏览，其他用户都无法浏览或访问时怎么办呢？Windows 的共享在默认情况下没有这个功能（在微软网站下载一个免费工具 Windows Server 2003 Access-based Enumeration 后也可实现），Samba 可以通过引用子配置文件的方法来实现。下面来看一个引用子配置文件的例子。

在 Samba 服务器有一个共享目录 smbtest（/test 的自身权限为 777）只希望用户 davidxu 可见、可读、可写，另一个共享目录 smbtest1（/test1 的自身权限为 777）只希望用户 tonyzhang 可见、可读、可写。通过子配置文件实现方法如下。

① 在/etc/samba 下建立一个文本文件其名称为 davidxu.smb.conf，在该文件中加入如下内容。

```
[share]
    comment = This is smb share
    path = /share
    write list = davidxu
```

② 在/etc/samba 下建立一个文本文件其名称为 tonyzhang.smb.conf，在该文件中加入如下内容。

```
[share1]
    comment = This is smb share1
    path = /share1
    write list = tonyzhang
```

③ 引用子配置文件。在引用子配置文件时有两种方法：

- config file：在“[global]”标签下加入如下参数。

```
[global]
    config file = /etc/samba/%U.smb.conf
```

- include：在“[global]”标签下入如下参数。

```
[global]
    include = /etc/samba/%U.smb.conf
```

config file 和 include 的区别是：使用 config file 时，当以 davidxu 的身份访问 Samba 服务器，只能浏览到 share，其他在 smb.conf 中定义的共享资源都无法查看；当以 tonyzhang 的身份访问 Samba 服务器，只能浏览到 share1，其他在 smb.conf 中定义的共享资源都无法查看。使用 include 时，当以 davidxu 的身份访问 Samba 服务器，除了可以浏览到 share，其他在 smb.conf 中定义的共享资源也可以浏览到；当以 tonyzhang 的身份访问 Samba 服务器，除了可以浏览到 share，其他在 smb.conf 中定义的共享资源也可以浏览到。

（7）文件及目录默认权限。

create mask、directory mask 分别用于设置客户端在访问 Samba 服务器时建立文件及目录默认的基于文件系统的权限。客户端通过网络访问 Samba 服务器共享资源时建立的文件或目录默认的基于文件系统的权限必须比通过本机访问时建立的文件或目录默认的基于文件系统的权限要低。在下面的例子中，非 root 用户通过本机访问/share 及/share1 时建立的文件默认的基于文件系统的权限是 664，那么在 smbtest 中设置的 create mask 是有效的，而在 smbtest1 中设置的 directory mask 则无效。

```
[share]
    comment = This is smb share
    path = /share
    writable = yes
    create mask = 444
 [share1]
    comment = This is smb share1
    path = /share1
    writable = yes
    create mask = 777
```

（8）共享资源管理员。

admin users 参数用于设置共享资源管理员，通过 admin users 参数指定的用户或组连接到 Samba 服务器共享资源操作时会忽略系统自身权限。当参数放在“[global]”标签时，是一个全局配置，对 Samba 服务器所有共享资源有效，如果该参数位于定义某共享资源的标识内时，只对这一个共享资源有效。如果全局设置与某一共享资源在该参数发生冲突时，共享资源内的该参数值优先。下面例子中当客户端使用 davidxu 用户连接到 share 时，即使/share 系统自身权限不允许写入，仍然可进行修改及删除操作；当客户端使用 sales 组用户连接到 share1 时，即使/ share1 系统自身权限不允许写入，仍然可进行修改及删除操作。

```
[global]
    admin users = davidxu
[share]
    comment = This is smb share
    path = /share
    writable = yes
[share1]
    comment = This is smb share1
    path = /share1
    writable = yes
    admin users = @sales
```

（9）用户与客户端访问控制。

在/etc/samba/smb.conf 文件中可以通过 hosts allow、hosts deny 对访问的客户端进行控制，也可以使用 valid users 对访问用户进行控制，但如果希望对特定用户在特定客户端进行控制就必须使用 PAM 模块了。

下面例子允许 tonyzhang 在位于 192.168.159.0/24 的客户端访问，拒绝 davidxu 在位于 192.168.158.0/24 的客户端访问。

① 在/etc/samba/smb.conf 文件“[global]”标签中加入如下内容。

```
obey pam restrictions=Yes
```

② 编辑配置文件/etc/pad.d/samba，在第一 accout 前添加如下语句。

```
account required pam_access.so accessfile=/etc/samba/myacl
```

③ 在/etc/samba 下建立名为 myacl 的文件，在该文件中增加如下内容。

```
+:tonyzhang:192.168.159.
-:davidxu:192.168.158.
```

（10）回收站配置。

为 Samba 服务器配置回收站，需要使用虚拟文件系统（VFS）模块，Samba 可以使用的多种 VFS 模块，这些模块存放在/usr/lib/samba/vfs 目录中。本章主要讲述通过 recycle.so 模块实现。

为 Samba 服务器配置回收站时，需要针对每个共享资源进行配置。在下面的例子中将为 Samba 服务器的共享资源 share 配置一个回收站。

```
[share]
    comment = This is smb share
```

```
        path = /share
        writable = yes
        vfs object = recycle
             recycle:repository = .deleted/%U
             recycle:keeptree = Yes
             recycle:versions = Yes
             recycle:maxsixe = 0
             recycle:exclude = *.tmp|*.log
             recycle:noversions = *.doc
```

在上面配置中，各参数含义如下。

- vfs object = recycle：载入 recycle.so 模块，recycle 名称不能为其他。
- recycle:repository = . recycle/%U：回收站的相对路径，这个选项指定删除的文件将被储存在什么目录，该目录和共享资源的实际路径有关。在上面的例子中，share 使用/sharet 路径，因此任何被删除的内容都被移动到这个目录下的. recycle 目录中。%U 变量是当前浏览共享用户的用户名，因此，每个用户删除的文件都会存放在以它用户名命名的目录下。此参数只能使用相对路径。所配置的目录其他用户必须有写权限。当有文件需在放入此目录时，如出现问题 Samba 服务器会将相关情况写入日志，所需文件删除。
- recycle:keeptree = Yes：在将文件移入回收站时，要建立相对应的目录结构。
- recycle:versions = Yes：如果在回收站所在目录中存在同名文件，则以“Copy #x of 文件名”的形式加以区分。
- recycle:maxsixe = 0：回收站的最大使用空间，以字节为单位。0 表示没有最大使用空间的限制。
- recycle:exclude = *.tmp|*.log：不放入回收站的文件类型。
- recycle:noversions = *.doc：如果在回收站所在目录中存在同名文件，覆盖原有文件的文件类型。

（11）自定义 Windows 客户端显示文件系统。

如果 Windows 客户端将一个 Samba 服务器映射为一个网络驱动器，当查看其文件系统时会显示为 NTFS，通过 fstype 参数可以修改显示的内容，该参数在“[global]”标签中定义。在通过下面的修改后，Windows 客户端显示为 Samba System，如图 3-39 所示。

图 3-39 自定义文件系统名

```
[global]
    fstype = Samba System
```

（12）Samba 服务器常见预设变量。

常见预设变量如表 3-2 所示。

表 3-2 **Samba 预设变量**

变 量 名	作 用	变 量 名	作 用
%S	当前服务名（如果存在）	%L	Samba 服务器的 NetBIOS 名
%P	当前服务的根目录（如果存在）	%N	NIS 服务器主机名
%u	当前服务的用户名（如果存在）	%p	NIS 服务器家目录
%g	当前用户的初始组	%R	采用协议等级
%U	当前连接的用户名	%d	Samba 服务的进程 ID
%G	当前连接用户的初始组	%a	访问 Samba 服务器客户端系统

变 量 名	作 用	变 量 名	作 用
%D	当前用户所属域或工作组名称	%I	访问 Samba 服务器客户端 IP 地址
%H	当前服务用户的家目录	%M	访问 Samba 服务器的客户端主机名
%v	Samba 服务器的版本	%m	访问 Samba 服务器客户端 NetBIOS 名
%h	Samba 服务器的主机名	%T	Samba 服务器日期及时间

下面来看一个例子，同一个共享资源名，当不同用户访问时，对应的 Samba 服务器本地目录会不同。

```
[share]
    comment = This is smb share
    path = /share/%U
```

3.3.3 Samba 防病毒

Samba-vscan（Samba-vscan 官方网站：http://www.openantivirus.org/projects.php）是 Samba 的模块，Samba-vscan 提供了 Samba 环境的 Server-based 防病毒功能，能在第一时间防止用户将感染病毒的文件存放到服务器中。配置 ClamAV、Samba-vscan 为 Samba 提供防病毒功能的操作步骤如下。

（1）下载 ClamAv 相关 RPM 包后，使用如下命令安装。

```
wget http://packages.sw.be/clamav/clamav-db-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-devel-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamd-0.95.3-1.el5.rf.i386.rpm
rpm -ivh *.rpm
```

（2）使用如下命令启动 clamd 服务，并设置为下次启动自动加载。

```
service clamd restart
chkconfig clamd on
```

（3）执行以下命令升级 ClamAV 病毒库。

```
freshclam
```

（4）使用 crontab -e 命令增加如下自动化任务。

```
*/50 * * * * /usr/bin/freshclam --quiet --daemon
```

（5）使用如下命令安装相关软件包。

```
yum -y install pcre-devel gmp-devel
```

（6）查询当前系统 Samba 服务器版本，然后下载（在 RHEL/ CentOS 5.4 中 Samba 版本为 3.0.33）解压并编译对应的 Samba 服务器源码包。

```
cd /usr/src
wget ftp://ftp.hkmirror.org/pub/samba/sambaftp/old-versions/samba-3.0.33.tar.gz
tar -xvzf samba-3.0.33.tar.gz
cd samba-3.0.33/source
./configure
make headers
```

（7）下载 Samba-vscan 源码包，解压后将其移动到 Samba 源码目录下 examples/VFS 目录。

```
cd /usr/src
wget http://www.openantivirus.org/download/samba-vscan-0.3.6c-beta5.tar.gz
tar -zxvf samba-vscan-0.3.6c-beta5.tar.gz
mv /usr/src/samba-vscan-0.3.6c-beta5 /usr/src/samba-3.0.33/examples/VFS/
```

（8）使用如下命令编译 Samba-vscan 后，将编译产生的库文件复制到/usr/lib/samba/vfs/目录，将病毒扫描配置文件到/etc/samba 目录。

```
cd /usr/src/samba-3.0.33/examples/VFS/samba-vscan-0.3.6c-beta5/
./configure
make
cp vscan-clamav.so /usr/lib/samba/vfs/
```

```
cp clamav/vscan-clamav.conf /etc/samba/
```

（9）修改/etc/samba/vscan-clamav.conf 中如下参数，以便 Samba-vscan 和 ClamAV 协同工作。

```
clamd socket name = /var/run/clamav/clamd.sock
infected file action = delete
```

其中/etc/samba/vscan-clamav.conf 中常用参数作用如下。

- max file size = 0：扫描文件的 size 上限，0 表示没有限制。
- verbose file logging = yes|no：log 文件的控制。如果指定为 yes 表示记录所有存取文件，如果是 no 只会记录感染了的文件。
- scan on open = yes|no：如果指定为 yes 表示每次打开文件都会扫描。
- deny access on error = yes|no：如果和 clamd 连接错误，是否不能存取那些被保护的文件。
- deny access on minor error = yes|no：如果和服务文件发生错误，是否不能存取那些被保护的文件。
- send warning message = yes|no：当发现感染病毒的文件时，是否向 Windows 客户端发送警告信息（在 Windows 客户端需要将“messager”服务启动才可收到警告信息）。
- infected file action = quarantine|delete|nothing：如何处理感染病毒的文件，quarantine 表示尝试移动去隔离区（如果移动不成功则会删除），delete 表示删除被感染病毒的文件，nothing 表示不做任何动作。
- quarantine directory = <目录>：指定隔离区的目录。
- quarantine prefix = <字符串>：被移动到隔离区的文件会自动在文件名前加上指定的字符串。

（10）在/etc/samba/smb.conf 文件“[global]”标签中，加入如下内容让 Samba 服务器能正确地调用 Samba-vscan 扫描病毒。

```
vfs object = vscan-clamav
vscan-clamav: config-file = /etc/samba/vscan-clamav.conf
```

（11）使用如下命令重新启动 smb 和 clamd 服务。

```
service smb restart
service clamd restart
```

当 Windows 客户端将带有病毒的文件复制到 Samba 的共享目录时，ClamAV 会根据/etc/samba/vscan-clamav.conf 中“infected file action”参数的配置处理感染病毒的文件。

3.4 文件服务器实用配置案例

文件服务应该说是目前企业使用最多的应用，而企业对文件服务器的需求也是多种多样，下面通过两个案例了解企业对文件服务器的一般需求及如果通过相应的文件服务器软件满足这样的需求。

3.4.1 企业全功能 FTP 配置案例

某企业需要配置一台 FTP 服务器，满足企业文件服务器的需求。该企业的网络拓扑如图 3-40 所示，企业采购的 FTP 服务器已安装 RHEL5.4，IP 地址为 192.168.159.14；FQDN 为 ftp.example.com，企业对 FTP 服务器的要求如下。

（1）只有 192.168.159.0/24 网段的客户端可以上传文件到 FTP 服务器，其他所有客户端只允许下载。

（2）每个用户最多可以上传 500MB 的数据。

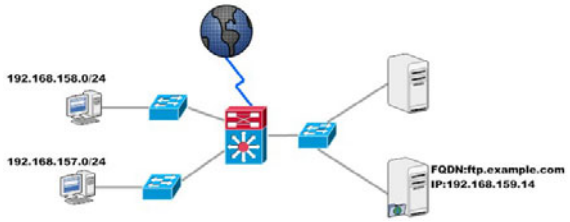


图 3-40 案例网络拓扑

（3）每个客户端只允许同时建立 2 个连接到 FTP 服务器。

（4）不允许采用匿名方式方式访问 FTP 服务器。

整个案例的实施过程如下。

（1）使用以下命令安装 MySQL。

```
yum -y install mysql-devel mysql-server pcre-devel gmp-devel
```

（2）下载 ClamAv 相关 RPM 包后，使用如下命令安装。

```
wget http://packages.sw.be/clamav/clamav-db-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-devel-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamd-0.95.3-1.el5.rf.i386.rpm
rpm -ivh *.rpm
```

（3）使用如下命令启动 clamd 服务，并设置为下次启动自动加载。

```
service clamd restart
chkconfig clamd on
```

（4）执行将以下命令升级 ClamAV 病毒库。

```
freshclam
```

（5）使用 crontab -e 命令增加如下自动化任务。

```
*/50 * * * * /usr/bin/freshclam --quiet --daemon
```

（6）启动 mysqld 服务，并设置为下次启动自动加载。

```
service mysqld restart
chkconfig mysqld on
```

（7）使用 mysqladmin 创建 MySQL 管理员及密码。

```
#建立名为 root 的 MySQL 管理员，并将密码设置为 redhat
mysqladmin -u root password redhat
```

（8）使用 root 用户登录 MySQL 数据库，建立用于保存虚拟用户的数据库。

```
create database proftpd;
```

（9）在 MySQL 环境中执行以下语句，建立保存虚拟用户及配额所需的表。

```
CREATE TABLE 'ftpgroups' (
 'groupname' varchar(30) NOT NULL default '',
 'gid' int(11) NOT NULL default '1000',
 'members' varchar(255) NOT NULL default ''
)ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE 'ftpusers'(
 'userid' varchar(30) NOT NULL default '',
 'passwd' varchar(80) NOT NULL default '',
 'uid' int(10) unsigned NOT NULL default '1000',
 'gid' int(10) unsigned NOT NULL default '1000',
 'homedir' varchar(255) NOT NULL default '',
 'shell' varchar(255) NOT NULL default '/sbin/nologin',
 'count' int(10) unsigned NOT NULL default '0',
 'host' varchar(30) NOT NULL default '',
 'lastlogin' varchar(30) NOT NULL default '',
 UNIQUE KEY 'userid' ('userid')
)ENGINE=MyISAM DEFAULT CHARSET=latin1
CREATE TABLE 'quotalimits'(
 'name' varchar(30) default NULL,
 'quota_type' enum('user','group','class','all') NOT NULL default 'user',
 'per_session' enum('false','true') NOT NULL default 'false',
 'limit_type' enum('soft','hard') NOT NULL default 'soft',
 'bytes_in_avail' float NOT NULL default '0',
 'bytes_out_avail' float NOT NULL default '0',
```

```
 'bytes_xfer_avail' float NOT NULL default '0',
 'files_in_avail' int(10) unsigned NOT NULL default '0',
 'files_out_avail' int(10) unsigned NOT NULL default '0',
 'files_xfer_avail' int(10) unsigned NOT NULL default '0'
)ENGINE=MyISAM DEFAULT CHARSET=latin1;
CREATE TABLE 'quotatallies'(
 'name' varchar(30) NOT NULL default '',
 'quota_type' enum('user','group','class','all') NOT NULL default 'user',
 'bytes_in_used' float NOT NULL default '0',
 'bytes_out_used' float NOT NULL default '0',
 'bytes_xfer_used' float NOT NULL default '0',
 'files_in_used' int(10) unsigned NOT NULL default '0',
 'files_out_used' int(10) unsigned NOT NULL default '0',
 'files_xfer_used' int(10) unsigned NOT NULL default '0'
)ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

（10）为了提高安全性，在 ProFTPd 读取 MySQL 数据库时，不应使用 root 用户，通过以下方式在 MySQL 建立一个专门用于读取 proftpd 数据库中表的用户，在 MySQL 环境中执行以下语句，本章中让 proftpd 用户使用 redhat 作为密码读取。

```
grant select,insert,update,delete,create,drop,index,alter,create temporary tables, \
             lock tables on proftpd.* to proftpd@localhost Identified by "redhat";
flush privileges;
```

（11）下载 ProFTPd 的 ClamAV 插件并使用以下命令解压。

```
cd /usr/src
wget        https://secure.thrallingpenguin.com/redmine/attachments/download/1/mod_clamav-0.11rc.tar.gz
tar -xvzf mod_clamav-0.11rc.tar.gz
```

（12）使用如下命令下载并安装 ProFTPd。

```
cd /usr/src
wget ftp://ftp.proftpd.org/distrib/source/proftpd-1.3.2a.tar.bz2
tar -xvjf proftpd-1.3.2a.tar.bz2
cd proftpd-1.3.2a
cp ../mod_clamav-0.11rc/mod_clamav.* proftpd-1.3.2a/contrib.
patch -p1 <../mod_clamav-0.11rc/proftpd.patch
./configure --with-modules=mod_sql:mod_sql_mysql:mod_quotatab:      \
               mod_quotatab_sql:mod_clamav \
               --with-includes=/usr/include/mysql --with-libraries=/usr/lib/mysql \
               --sysconfdir=/etc/proftpd
make && make install
```

（13）使用下方法制作 ProFTPd 在 RHEL/CentOS 中的启动脚本。

- 复制启动脚本例子文件。

```
cd /usr/src/proftpd-1.3.2a
cp contrib/dist/rpm/proftpd.init.d /etc/rc.d/init.d/proftpd
```

- 使用以下命令修改启动脚本权限。

```
chmod +x /etc/rc.d/init.d/proftpd
```

- 增加 proftpd 服务。

```
chkconfig --add proftpd
```

（14）修改/etc/proftpd/proftpd.conf 文件，内容如下。

```
Group        nobody
AllowRetrieveRestart on
AllowStoreRestart on
ServerName "ftp server"
ServerIdent on
RootLogin on
IdentLookups off
UseReverseDNS off
DeleteAbortedStores on
HideNoAccess on
RequireValidShell off
```

```
MaxInstances 30
MaxHostsPerUser 2
MaxClients 10
MaxClientsPerHost 1
MaxLoginAttempts 3
TimeoutStalled 600
TimeoutLogin 900
TimeoutIdle 600
TimeoutNoTransfer 600
<Limit LOGIN>
    DenyUser ftp
</Limit>
<Limit WRITE>
    order allow,deny
    Allow from 192.168.159.
    Deny from all
</Limit>
SQLConnectInfo proftpd@localhost proftpd  redhat
SQLAuthTypes Plaintext
SQLUserInfo ftpusers userid passwd uid gid homedir shell
SQLGroupInfo ftpgroups groupname gid members
SQLAuthenticate users groups
SQLNegativeCache on
SQLLogFile /var/log/proftpd.sql.log
SQLNamedQuery getcount SELECT "count from ftpusers where userid='%u'"
SQLNamedQuery getlastlogin SELECT "lastlogin from ftpusers where userid='%u'"
SQLNamedQuery updatelogininfo UPDATE                  \
          "count=count+1,host='%h',lastlogin=current_timestamp() \
          WHERE userid='%u'" ftpusers
SQLShowInfo PASS "230" "You've logged on %{getcount} times, last login    \
          at %{getlastlogin}"
SQLLog PASS updatelogininfo
QuotaDirectoryTally on
QuotaDisplayUnits "Mb"
QuotaEngine on
QuotaShowQuotas on
SQLNamedQuery get-quota-limit SELECT "name, quota_type, per_session,   \
       limit_type, bytes_in_avail,bytes_out_avail, bytes_xfer_avail,  \
       files_in_avail, files_out_avail, files_xfer_avail FROM quotalimits \
       WHERE name = '%{0}' AND quota_type = '%{1}'"
SQLNamedQuery get-quota-tally SELECT "name, quota_type, bytes_in_used,  \
       bytes_out_used,bytes_xfer_used, files_in_used, files_out_used,    \
       files_xfer_used FROM quotatallies \
       WHERE name = '%{0}' AND quota_type = '%{1}'"
SQLNamedQuery update-quota-tally UPDATE "bytes_in_used   \
       = bytes_in_used + %{0},bytes_out_used = bytes_out_used + %{1}, \
       bytes_xfer_used = bytes_xfer_used + %{2}, \
       files_in_used = files_in_used + %{3}, files_out_used =   \
       files_out_used + %{4}, files_xfer_used = files_xfer_used + %{5} \
       WHERE name = '%{6}' AND quota_type = '%{7}'" quotatallies
SQLNamedQuery insert-quota-tally INSERT      \
       "%{0}, %{1}, %{2}, %{3}, %{4}, %{5}, %{6}, %{7}" quotatallies
QuotaLimitTable sql:/get-quota-limit
QuotaTallyTable sql:/get-quota-tally/update-quota-tally/insert-quota-tally
```

（15）在 MySQL 环境中执行以下语句，建立虚拟用户（这里只建立一个用户名为 tonyzhang 密码为 12345 的例子）。

```
INSERT INTO 'ftpusers' VALUES ('tonyzhang', '12345', 1000, 1000, '/home/tonyzhang', \
                        '/sbin/nologin',0,'','');
```

（16）在 MySQL 环境中执行以下语句，建立虚拟用户的配额（这里只建立一个用户名为 tonyzhang 建立配额）。

```
insert   into   quotalimits   VALUES ('tonyzhang','user','false','soft','524288000','0','0',
'0','0','0');
```

（17）建立以下命令虚拟用户的家目录。

```
mkdir /home/tonyzhang
chmod 777 /home/tonyzhang
```

（18）启动 ProFTPd 服务，并设置为下次启动自动加载。

```
service proftpd restart
chkconfig proftpd on
```

到此 FTP 服务器的配置已可满足该企业的所有需求。

3.4.2 企业全功能 Samba 配置案例

某企业需要配置一台文件服务器，使企业内员工可以方便地进行资源共享。该企业的网络拓扑如图 3-41 所示，企业中所有客户端全部使用 Windows XP/Vista，其中设计部计算机位于 192.168.159.0/24 网段，市场部计算机位于 192.168.158.0/2，计划财务部计算机位于 192.168.157.0/24 网段。企业采购的文件服务器已安装 RHEL5.4，IP 地址为 192.168.159.14；FQDN 为 fs.example.com。每位员工用户建立完成并将/dev/sda10 挂载到/share 作为共享分区。企业对文件服务器的要求如下。

（1）计划财务部及设计部所有客户端，但公用计算机（pub.example.com）除外，可以使用该文件服务器。

（2）设计部所有客户端可以使用文件服务器上的光驱。

（3）需要一个存放内部资料的目录，所有用户只可读其中内容。

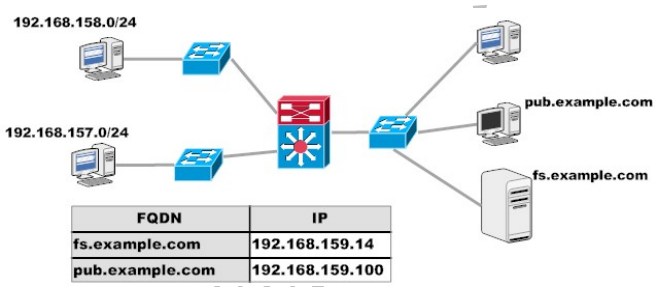


图 3-41 案例网络拓扑

（4）每个部门有一个需要存放资料的公用目录，只允许该部门员工可见、可读、可写。

（5）每个部门的公用目录中的内容，除了上传文件的用户及管理员以外其他用户不能删除。

（6）每位员工有一个自己的目录，除自己可读可写外，只有该部门经理可读。

整个案例的实施过程如下。

（1）在 fs.example.com 上安装 Samba 服务及相关软件包。

```
yum -y install samba pcre-devel gmp-devel
```

（2）下载 ClamAv 相关 RPM 包后，使用如下命令安装。

```
wget http://packages.sw.be/clamav/clamav-db-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-devel-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamav-0.95.3-1.el5.rf.i386.rpm
wget http://packages.sw.be/clamav/clamd-0.95.3-1.el5.rf.i386.rpm
rpm -ivh *.rpm
```

（3）使用如下命令启动 clamd 服务，并设置为下次启动自动加载。

```
service clamd restart
chkconfig clamd on
```

（4）执行将以下命令升级 ClamAV 病毒库。

```
freshclam
```

（5）使用 crontab -e 命令增加如下自动化任务。

```
*/50 * * * * /usr/bin/freshclam --quiet --daemon
```

（6）下载 Samba 源码包解压并编译对应的 Samba 服务器源码包。

```
cd /usr/src
wget ftp://ftp.hkmirror.org/pub/samba/sambaftp/old-versions/samba-3.0.33.tar.gz
tar -xvzf samba-3.0.33.tar.gz
cd samba-3.0.33/source
./configure
```

```
make headers
```

（7）下载 Samba-vscan 源码包，解压后将其移动到 Samba 源码目录下 examples/VFS 目录。

```
cd /usr/src
wget http://www.openantivirus.org/download/samba-vscan-0.3.6c-beta5.tar.gz
tar -zxvf samba-vscan-0.3.6c-beta5.tar.gz
mv /usr/src/ samba-vscan-0.3.6c-beta5 /usr/src/samba-3.0.33/examples/VFS/
```

（8）使用如下命令编译 Samba-vscan 后，将编译产生的库文件复制到/usr/lib/samba/vfs/目录，将病毒扫面配置文件到/etc/samba 目录。

```
cd /usr/src/samba-3.0.33/examples/VFS/samba-vscan-0.3.6c-beta5/
./configure
make

cp vscan-clamav.so /usr/lib/samba/vfs/
cp clamav/vscan-clamav.conf /etc/samba/
```

（9）修改/etc/samba/vscan-clamav.conf 中如下参数，以便 Samba-vscan 和 ClamAV 协同工作。

```
clamd socket name = /var/run/clamav/clamd.sock
infected file action = delete
```

（10）在 fs.example.com 上配置/dev/sda10 挂载参数，让该分区支持 acl 及硬盘配额。在/etc/fstab 中将/dev/sda10 的参数改为以下内容，修改完成后用 mount –o remount/share 命令重新载入挂载参数。

```
/dev/sda10   /share   ext3    defaults,acl,usrquota,grpquota       0  0
```

（11）在 fs.example.com 上根据部门建立用户组，并将用户加入相应组。

```
groupadd design          #设计部部门组
groupadd finance         #计划财务部部门组
groupadd manager         #部门经理组
#以下是将设计部所有员工初始组设置为 design
usermod -g design davidxu
usermod -g design petexu
#以下是将计划财务部所有员工初始组设置为 finance
usermod -g finance mikeliu
usermod -g finance janeli
#以下是将 manager 组加入计划财务部及设计部二位经理的额外组
usermod -aG manager davidxu
usermod -aG manager janeli
```

（12）在 fs.example.com 上建立所需目录。

```
mkdir /share/public              #内部资料公用目录
mkdir /share/design          #设计部专用目录
mkdir /share/finance         #计划财务部专用目录
mkdir /share/design/public       #设计部公用目录
mkdir /share/finance/public      #计划财务部公用目录
#以下是为每位员工建立专用目录
mkdir /share/design/davidxu /share/design/petexu /share/finance/mikeliu    \
     /share/finance/janeli
#以下是为通过 Sticky 实现每个部门公用目录除了上传文件的用户及管理员以外其他用户不#能删除
chmod o+t /share/design/public /share/finance/public
```

（13）在 fs.example.com 上将系统用户加入 Samba 服务器。

```
smbpasswd -a davidxu
smbpasswd -a petexu
smbpasswd -a mikeliu
smbpasswd -a janeli
```

（14）在 fs.example.com 上修改 smb.conf，在[global]标签下加入如下内容。

```
[global]
hosts allow = lo 192.168.159. 192.168.157. EXCEPT 192.168.159.100
include = /etc/samba/%G.smb.conf
include = /etc/samba/%U.smb.conf
vfs object = vscan-clamav
vscan-clamav: config-file = /etc/samba/vscan-clamav.conf
```

（15）在 fs.example.com 上修改 smb.conf 文件，内容如下。

```
[public]
    path = /share/public
 [cdrom]
    path = /mnt/cdrom
```

```
        root preexec = /bin/mount -t iso9660 /dev/cdrom /mnt/cdrom
        root postexec = /bin/umount /mnt/cdrom
```

（16）在 fs.example.com 上/etc/samba 目录下建立以下子配置文件。

- design.smb.conf

```
[design]
    path = /share/design/public
    write list = @design
```

- davidxu.smb.conf

```
[davidxu]
    path = /share/design/davidxu
    write list = davidxu
 [petexu]
    path = /share/design/petexu
    readonly = yes
```

- petexu.smb.conf

```
[petexu]
    path = /share/design/petexu
    write list = petexu
```

- finance.smb.conf

```
[finance]
    path = /share/finance/public
    write list = @finance
```

- mikeliu.smb.conf

```
[mikeliu]
    path = /share/finance/mikeliu
    write list = mikeliu
 [janeli]
    path = /share/finance/janeli
    read only = yes
```

- janeli.smb.conf

```
[janeli]
    path = /share/finance/janeli
    write list = janeli
```

（17）在 fs.example.com 上设置目录自身权限。

```
setfacl -R -m d:g:design:rwx /share/design
setfacl -R -m g:design:rwx/share/design
setfacl -R -m d:g:finance:rwx /share/finance
setfacl -R -m g:finance:rwx /share/finance
```

（18）在 fs.example.com 上启动 Samba 服务，并设置为下次启动自动加载。

```
service smb restart
chkconfig smb on
```

到此 Samba 服务器的配置已可满足该企业的所有需求。

3.5 文件服务器解决方案比较

企业选择在 Linux 系统上搭建文件服务器后，可以提供文件共享服务的方式也比较多，比如 FTP、NFS、Samba 等。这些服务虽然都是提供文件共享服务，但各有各的优势。

1．FTP 服务

FTP 在 Samba 出现之前几乎是类 UNIT 平台下实现异构系统数据共享唯一的解决方案，但随着 Samba 的成熟，FTP 在这方面已没有明显的优势，目前 FTP 服务主要在为 Internet 用户提供文件共享时使用得比较普遍。这主要是因为 FTP 服务的许多特性适合于在 Internet 中使用，比如所有 FTP 软件都支持对访问 IP 的

数量、单 IP 的并发数等，这些特性可以很好地保护 FTP 服务器自身。

在本章讲述的 vsftpd、ProFTPd 都应该算得上类 UNIT 平台比较优秀的 FTP 软件。vsftpd 更关注安全性、速度及稳定性，因此 vsftpd 提供的功能相对其他 FTP 软件要少一些或是默认不支持。ProFTPd 在设置其配置方法上更加合理，对于 Apache 比较熟悉的用户使用 ProFTPd 会更容易上手，而且 ProFTPd 所提供的功能更加丰富。在对这两种 FTP 软件的选择时应该更视 FTP 服务器要求而定，如果只需要提供一个 FTP 站点给用户下载数据而且并发数可能比较大，vsftpd 是非常不错的选择。如果需要使用比较丰富的功能而且 IT 人员对 Apache 比较熟悉，ProFTPd 是非常不错的选择。

2．Samba 服务

毫不夸张地说 Samba 是开源社区最有影响力的贡献之一，Samba 不但很好地解决了异构平台中数据共享问题，而且还可以提供比 Windows 文件服务器更加丰富的功能。如果在企业内部网络中需要使用文件/打印服务器，毫无疑问 Samba 是目前最好的选择，Samba 可以发挥类 UNIX 平台稳定性的优势，同时让 Windows 客户端方便地访问。即使企业中使用了微软的活动目录，Samba 同样可以加入到活动目录，实现单一登录访问。

3．NFS 服务

NFS 是网络文件系统（Network File System）的缩写，是 1980 年由 Sun 所发展出来在类 UNIX 系统间实现数据共享的一种方法，NFS 支持应用程序在客户端通过网络存取位于服务器磁盘中数据的一种文件系统协议。NFS 的基本原则是让不同的客户端及服务器通过一组 RPCs 共享相同的文件系统，NFS 独立于操作系统，允许不同硬件及操作系统共同进行文件的共享。NFS 功能上比较单一，而且虽然微软公司的 Windows 系统也提供对 NFS 的支持（具体方法见 2.1），但 NFS 还是更适合在类 UNIX 系统间实现数据共享。

联系方式