《使用 vsftpd 架设 FTP 服务器》

Smartraining 工作室

osmond 编著

引言

本文为笔者与友人合著的拙作《RedHat 8.X 网络服务》中的一节内容，此书将于下半年由机械工业出版社出版。目前此书仍在紧张的写作过程中。

目前，有关 vsftpd 配置方面的文档甚少，所以笔者决定先将本节公开，以饷广大的 Linux 爱好者。

由于此文为尚未出版的出版物中的一部分，存在版权问题，所以笔者将其制作为操作受限的 PDF 文件，见谅。

欢迎大家多提宝贵意见，笔者将不胜感激。

Osmond

osmond68@163.com

目录

10.2 使用 vsftpd 架设 FTP 服务器

本节内容
➢ vsftpd 的功能特性 ➢ vsftpd 的两种安装方法（RPM/tar.gz） ➢ 比较独立启动和 xinetd 启动的 vsftpd 服务器配置和启动 ➢ 比较 vsftpd 中的三类用户 ➢ 配置 vsftpd 的常用参数 ➢ 配置基于本地用户的访问控制和基于主机的访问控制 ➢ 配置基于 IP 的虚拟 FTP 服务器 ➢ 配置虚拟用户的 FTP 服务器
学习目标
➢ 掌握 vsftpd 的安装方法 ➢ 熟悉 RedHat 8.1 中的默认配置 ➢ 熟悉 vsftpd 的性能配置语句 ➢ 掌握基于本地用户的访问控制和基于主机的访问控制的配置 ➢ 掌握本地组（多个本地用户）访问的 FTP 服务器配置 ➢ 理解 vsftpd 的两种启动运行方式配置的不同并加以比较 ➢ 掌握安全的仅下载 FTP 服务器的配置 ➢ 掌握可上传的 FTP 服务器的配置 ➢ 掌握基于 IP 的虚拟 FTP 服务器的配置（两种启动方式） ➢ 理解 vsftpd 中三类用户的使用场合 ➢ 掌握多个虚拟用户访问（具有各自操作权限）的 FTP 服务器配置
估计用时：2 天

10.2.1 vsftpd 简介

什么是 vsftpd

vsftpd 是一个基于 GPL 发布的类 UNIX 系统上使用的 FTP 服务器软件。其中的 vs 是‘ Very Secure ”的缩写，从此名称缩写可以看出，编制者的初衷就是代码的安全性。

Vsftpd 的特性

安全性是编写 vsftpd 代码的初衷，除了与生俱来的安全性能之外，高速、稳定的性能是 vsftpd 的两个特性。

在速度方面：使用 ASCII 模式下载数据时，vsftpd 的速度是 WU-FTPd 的两倍；如果 Linux 的主机使用 2.4.X 版本的内核，在千兆以太网上的下载速度可达 86Mbyte/sec。

在稳定性方面：vsftpd 可以在单机（非集群）上支持 4000 个以上的并发用户同时连接。据 ftp.redhat.com 的数据，vsftpd 可以支持 15000 个并发用户。

除了安全、高速、稳定之外，vsftpd 还具有如下的特性：

- 支持基于 IP 的虚拟 FTP 服务器
- 支持虚拟用户
- 支持 PAM 或 xinetd / tcp_wrappers 的认证方式
- 支持两种运行方式：独立和 Xinetd

- 支持每个虚拟用具有独立的配置
- 支持带宽限制等

谁在使用 vsftpd

由于 vsftpd 具有上述的特性，现在越来越多的 FTP 服务器使用 vsftpd。例如：

- ftp.redhat.com
- ftp.suse.com
- ftp.debian.org
- ftp.gnu.org
- ftp.gnome.org
- ftp.ximian.com
- ftp.kde.org
- ftp.gimp.org
- ftp.openbsd.org
- ftp.sunet.se
- rpmfind.net

从 RPM 安装 vsftpd

Redhat 8.1 自带了 vsftpd，下面以 RPM 包的安装为例介绍 vsftpd 的安装。若用户在安装 Redhat 时已经安装了 FTP 服务器，则可跳过下面的安装步骤。

操作步骤 10.1　安装 vsftpd

```
//查看是否安装了 vsftpd 和 anonftp
# rpm -qa|grep vsftpd
# rpm -qa|grep anonftp
//将 Redhat8.1 的第 1 张安装光盘放入光驱后挂装
# mount /mnt/cdrom
//进入光盘的 RedHat/RPMS 目录
# cd /mnt/cdrom/RedHat/RPMS
//安装所需的 RPM 包
# rpm –ivh vsftpd*
# rpm –ivh anonftp*
//弹出光盘
# cd;eject
```

注意

anonftp 包用于创建匿名 FTP 服务器目录。若要架设匿名 FTP 服务器就应该安装此包。anonftp 包安装的匿名 FTP 服务器目录是/var/ftp，匿名下载目录为/var/ftp/pub。

安装完 vsftpd 后，下一步就是启动了。Redhat 默认 vsftpd 以 xinetd 方式启动，所以需要执行如下的操作步骤。

操作步骤 10.2　启动 vsftpd

```
//修改/etc/xinetd.d/vsftpd
#  vi /etc/xinetd.d/vsftpd
//键入 i 进入插入模式
//将 disable  = yes
//改为 disable  = no
//按 Esc 键返回编辑模式，再键入命令：wq 存盘退出 vi
#
//重新启动 xinetd 守护进程
# service xinetd restart
```

下面的操作用于检验 vsftpd 是否被启动。

操作步骤 10.3　检验 vsftpd 是否被启动

```
# telnet 127.0.0.1 21
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
220 ready, dude (vsFTPd 1.1.0: beat me, break me)
//从上面的内容可以确认 vsftpd 已经被启动，按 Ctrl+]中断会话
^]
//按 q 退出 telnet
telnet> q
Connection closed.
#
```

10.2.2　ReHat8.1 中 vsftpd 的默认配置

ReHat8.1 中的配置文件

在 RedHat8.1 中 vsftpd 的配置文件有三个，分别是：

- /etc/vsftpd.conf
- /etc/vsftpd.ftpusers
- /etc/vsftpd.user_list

其中，/etc/vsftpd.conf 是主配置文件。/etc/vsftpd.ftpusers 中指定了哪些用户不能访问 FTP 服务器。/etc/vsftpd.user_list 中指定的用户默认情况（即在/etc/vsftpd.conf 中设置了 userlist_deny=YES）下也不能访问 FTP 服务器，当在/etc/vsftpd.conf 中设置了 userlist_deny=NO 时，仅仅允许/etc/vsftpd.user_list 中指定的用户访问 FTP 服务器。

ReHat8.1 中的默认配置

使用如下的操作查看 vsftpd 的默认主配置文件。

操作步骤 10.4　查看 vsftpd 的默认主配置文件

```
# cat /etc/vsftpd.conf|grep =
//允许匿名登录
```

```
anonymous_enable=YES
//允许本地用户登录
local_enable=YES
//开放本地用户的写权限
write_enable=YES
//设置本地用户的文件生成掩码为 022，默认值为 077
local_umask=022
//当切换到目录时，显示该目录下的.message 隐含文件的内容
//这是由于默认情况下有 message_file=.message 的设置
dirmessage_enable=YES
//激活上传和下载日志
xferlog_enable=YES
//启用 FTP 数据端口的连接请求
connect_from_port_20=YES
//使用标准的 ftpd xferlog 日志格式
xferlog_std_format=YES
//设置 PAM 认证服务的配置文件名称，该文件存放在/etc/pam.d/目录下
pam_service_name=vsftpd
#
```


注意

在上面的配置文件中，忽略了被注释掉（即以#开头）的配置语句行。这些行将在下面逐步介绍。

测试 ReHat8.1 中的默认配置

执行下面的操作，测试 ReHat8.1 中 vsftpd 的默认配置。


警告

在默认情况下，匿名服务器下载目录/var/ftp/pub 中没有任何内容，为了进行测试，可以先向该目录复制些文件。笔者向此目录复制了 webmin-1.000-1.noarch.rpm。

操作步骤 10.5　测试 ReHat8.1 中 vsftpd 的默认配置－匿名账号

```
//生成目录信息文件/var/ftp/pub/.message
# echo "Welcome to this Directory.">/var/ftp/pub/.message
//用户也可以使用 vi 编辑此文件
//同时也可以写入更多的内容，如：
//# ls -F /var/ftp/pub/ >/var/ftp/pub/.message 等
# cd
//使用 FTP 客户端连接本地 FTP 服务器
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 ready, dude (vsFTPd 1.1.0: beat me, break me)
```

```
//使用匿名 FTP 账号（ftp 或 anonymous）登录
Name (127.0.0.1:root): ftp
331 Please specify the password.
//输入 Email 地址作为 FTP 匿名账号的口令
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
//列表显示匿名 FTP 服务器目录
ftp> ls
227 Entering Passive Mode (127,0,0,1,179,2)
150 Here comes the directory listing.
d--x--x--x   2 0       0          4096 Oct 09 08:36 bin
d--x--x--x   2 0       0          4096 Oct 09 08:36 etc
drwxr-xr-x   2 0       0          4096 Oct 09 08:36 lib
drwxr-sr-x   2 0       50         4096 Mar 09 19:45 pub
226 Directory send OK.
//进入匿名 FTP 服务器下载目录
ftp> cd pub
250-Welcome to this Directory.
//此处显示了.message 文件的内容
250 Directory successfully changed.
ftp> ls
227 Entering Passive Mode (127,0,0,1,45,163)
150 Here comes the directory listing.
-r-xr-xr-x   1 0       50       6477959 Mar 09 19:41 webmin-1.000-1.noarch.rpm
226 Directory send OK.
//下载文件 webmin-1.000-1.noarch.rpm
ftp> mget web*
mget webmin-1.000-1.noarch.rpm? y
227 Entering Passive Mode (127,0,0,1,37,151)
150 Opening BINARY mode data connection for webmin-1.000-1.noarch.rpm (6477959
bytes).
226 File send OK.
6477959 bytes received in 0.794 secs (8e+03 Kbytes/sec)
//显示本地文件
ftp> !ls
anaconda-ks.cfg  install.log  install.log.syslog  webmin-1.000-1.noarch.rpm
//确认已经将文件 webmin-1.000-1.noarch.rpm 下载到本地
//
//上传文件 install.log
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,184,207)
```

```
550 Permission denied.
//上传文件失败
//切换到根目录
ftp> cd /
250 Directory successfully changed.
//显示根目录下的内容
ftp> ls
227 Entering Passive Mode (127,0,0,1,150,224)
150 Here comes the directory listing.
d--x--x--x   2 0       0          4096 Oct 09 08:36 bin
d--x--x--x   2 0       0          4096 Oct 09 08:36 etc
drwxr-xr-x   2 0       0          4096 Oct 09 08:36 lib
drwxr-sr-x   2 0       50         4096 Mar 09 19:45 pub
226 Directory send OK.
//退出 FTP
ftp> bye
221 Goodbye.
#
//查看日志
# cat /var/log/vsftpd.log
Mon Mar 10 03:50:10 2003 1 127.0.0.1 6477959 /pub/webmin-1.000-1.noarch.rpm b
_ o a ah@dsa ftp 0 * c
#
```

操作步骤 10.6　测试 ReHat8.1 中 vsftpd 的默认配置－本地账号

```
//查看本地普通用户
# tail -1 /etc/passwd
lrj:x:501:501::/home/lrj:/bin/bash
//使用本地账号 lrj 登录
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 ready, dude (vsFTPd 1.1.0: beat me, break me)
Name (127.0.0.1:root): lrj
331 Please specify the password.
//输入 lrj 用户的口令
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
//显示远程主机 lrj 自家目录的内容
ftp> ls
227 Entering Passive Mode (127,0,0,1,103,231)
150 Here comes the directory listing.
-rw-r--r--   1 501     501       56683 Nov 23 23:21 cipe-1.4.5-11.i386.rpm
```

```
226 Directory send OK.
//下载文件 cipe-1.4.5-11.i386.rpm
ftp> get cipe-1.4.5-11.i386.rpm
local: cipe-1.4.5-11.i386.rpm remote: cipe-1.4.5-11.i386.rpm
227 Entering Passive Mode (127,0,0,1,136,170)
150 Opening BINARY mode data connection for cipe-1.4.5-11.i386.rpm (56683 bytes).
226 File send OK.
56683 bytes received in 0.0575 secs (9.6e+02 Kbytes/sec)
//显示本地文件
ftp> !ls -l
total 6428
-rw-r--r--    1 root    root        2306 10月  9 16:57 anaconda-ks.cfg
-rw-r--r--    1 root    root       56683  3月 10 04:06 cipe-1.4.5-11.i386.rpm
-rw-r--r--    1 root    root       17596 10月  9 16:49 install.log
-rw-r--r--    1 root    root        4096 10月  9 16:49 install.log.syslog
-rw-r--r--        1  root          root            6477959    3  月    10  03:50
webmin-1.000-1.noarch.rpm
//文件 cipe-1.4.5-11.i386.rpm 已经下载到本地
//上传本地文件 install.log，注意此文件的本地权限为 644
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,235,238)
150 Go ahead make my day^W^W^Wsend me the data.
226 File receive OK.
17596 bytes sent in -0.00114 secs (-1.5e+04 Kbytes/sec)
//显示远程主机上的文件
ftp> ls
227 Entering Passive Mode (127,0,0,1,136,160)
150 Here comes the directory listing.
-rw-r--r--    1 501     501        56683 Nov 23 23:21 cipe-1.4.5-11.i386.rpm
-rw-r--r--    1 501     501        17596 Mar 09 20:07 install.log
226 Directory send OK.
//文件 install.log 已经被上传
//由于此文件在本地就没有执行权限，所以此文件在远程主机上的权限为 644
//若此文件在本地具有执行权限，则此文件在远程主机上的权限为 777-022=755
//
//切换到根目录
ftp> cd /
250 Directory successfully changed.
ftp> ls
227 Entering Passive Mode (127,0,0,1,36,80)
150 Here comes the directory listing.
drwxr-xr-x    2 0       0          4096 Oct 09 08:35 bin
drwxr-xr-x    4 0       0          1024 Oct 09 08:30 boot
```

```
drwxr-xr-x   20 0       0         118784 Mar 07 11:42 dev
drwxr-xr-x   63 0       0           4096 Mar 09 19:41 etc
drwxr-xr-x    5 0       0           4096 Nov 23 23:19 home
drwxr-xr-x    2 0       0           4096 Jun 21  2001 initrd
drwxr-xr-x    7 0       0           4096 Oct 09 08:48 lib
drwx------    2 0       0          16384 Oct 09 08:23 lost+found
drwxr-xr-x    2 0       0           4096 Aug 27  2002 misc
drwxr-xr-x    4 0       0           4096 Oct 09 01:00 mnt
drwxr-xr-x    2 0       0           4096 Aug 23  1999 opt
dr-xr-xr-x   66 0       0              0 Mar 07 19:41 proc
drwxr-x---    5 0       0           4096 Mar 09 20:06 root
drwxr-xr-x    2 0       0           8192 Oct 09 08:48 sbin
drwxrwxrwt    5 0       0           4096 Mar 09 20:04 tmp
drwxr-xr-x   16 0       0           4096 Oct 09 08:48 usr
drwxr-xr-x   23 0       0           4096 Mar 04 05:34 var
226 Directory send OK.
ftp> cd tmp
250 Directory successfully changed.
ftp> ls
227 Entering Passive Mode (127,0,0,1,194,224)
150 Here comes the directory listing.
-r--r--r--    1 0       0        1431335 Nov 23 23:22 vmware-linux-tools.tar.gz
226 Directory send OK.
//下载远程主机/tmp 目录下的文件
ftp> mget vm*
mget vmware-linux-tools.tar.gz? y
227 Entering Passive Mode (127,0,0,1,242,86)
150 Opening BINARY mode data connection for vmware-linux-tools.tar.gz (1431335
bytes).
226 File send OK.
1431335 bytes received in 0.299 secs (4.7e+03 Kbytes/sec)
//下载成功
//上传文件到远程主机的/tmp 目录
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,243,2)
150 Go ahead make my day^W^W^Wsend me the data.
226 File receive OK.
17596 bytes sent in 0.00213 secs (8.1e+03 Kbytes/sec)
ftp> ls
227 Entering Passive Mode (127,0,0,1,96,214)
150 Here comes the directory listing.
-rw-r--r--    1 501     501         17596 Mar 09 20:12 install.log
-r--r--r--    1 0       0        1431335 Nov 23 23:22 vmware-linux-tools.tar.gz
```

```
226 Directory send OK.
//由于使用 write_enable=YES 开放了本地用户的写权限、
//且远程主机的/tmp 目录的权限为 1777，
//所以上传成功
ftp> bye
221 Goodbye.
#
//查看日志
# cat /var/log/vsftpd.log
Mon Mar 10 04:06:47 2003 1 127.0.0.1 56683 /home/lrj/cipe-1.4.5-11.i386.rpm b
_ o r lrj ftp 0 * c
Mon Mar 10 04:07:18 2003 1 127.0.0.1 17596 /home/lrj/install.log b _ i r lrj ftp
0 * c
Mon Mar 10 04:11:47 2003 1 127.0.0.1 1431335 /tmp/vmware-linux-tools.tar.gz b
_ o r lrj ftp 0 * c
Mon Mar 10 04:12:18 2003 1 127.0.0.1 17596 /tmp/install.log b _ i r lrj ftp 0
* c
#
//重新连接本地 FTP 服务器
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 ready, dude (vsFTPd 1.1.0: beat me, break me)
//以 root 用户登录
Name (127.0.0.1:root): root
331 Please specify the password.
//输入 root 用户口令
Password:
530 Login incorrect.
Login failed.
//登录失败
ftp> bye
221 Goodbye.
# cat /etc/vsftpd.ftpusers |grep root
root
//之所以登录失败，是由于 root 用户写在了/etc/vsftpd.ftpusers 文件中
#
```

重点

通过以上的操作，得出在 RedHat 默认设置下的结论：

（1）允许匿名用户和本地用户登录；

（2）匿名用户的登录名为 ftp 或 anonymous，口令为一个 Email 地址；

（3）匿名用户不能离开匿名服务器目录/var/ftp，且只能下载

不能上传；

（4）本地用户的登录名为本地用户名，口令为此本地用户的口令；

（5）本地用户可以离开自家目录切换至有权访问的其他目录，并在权限允许的情况下进行上传/下载；

（6）写在文件/etc/vsftpd.ftpusers 中的本地用户禁止登录。

10.2.3 进一步配置 vsftpd

允许匿名用户上传

为了使匿名用户能够上传，需要在/etc/vsftpd 中激活两个配置选项，分别是：

- anon_upload_enable
- anon_mkdir_write_enable

同时还要配置 anon_world_readable_only=NO 放开匿名用户对整个服务器的浏览权限。具体的操作步骤如下。

操作步骤 10.7 配置 vsftpd 允许匿名用户上传

```
//修改 vsftpd 的主配置文件/etc/vsftpd.conf
# vi /etc/vsftpd.conf
//将如下两行前的#删除
//①#anon_upload_enable=YES → 允许匿名用户上传
//②#anon_mkdir_write_enable=YES → 开启匿名用户的写和创建目录的权限
//若要以上两项设置生效，同时还要求：
//(1) write_enable=YES
//(2) 匿名用户对文件系统的上传目录具有写权限
//添加如下的配置语句
// anon_world_readable_only=NO
//上面的配置语句用于放开匿名用户的浏览权限
//修改后存盘退出 vi
//
//创建匿名上传目录
# mkdir /var/ftp/incoming
# 修改上传目录的权限
# chmod o+w /var/ftp/incoming/
#
//重新启动 xinetd
# service xinetd restart
#
```

下面进行测试。

操作步骤 10.8　测试匿名用户上传

```
//使用匿名用户连接本地 FTP 服务器
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 ready, dude (vsFTPd 1.1.0: beat me, break me)
Name (127.0.0.1:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (127,0,0,1,87,37)
150 Here comes the directory listing.
d--x--x--x   2 0       0          4096 Oct 09 08:36 bin
d--x--x--x   2 0       0          4096 Oct 09 08:36 etc
drwxr-xrwx   2 0       0          4096 Mar 09 23:57 incoming
drwxr-xr-x   2 0       0          4096 Oct 09 08:36 lib
drwxr-sr-x   2 0       50         4096 Mar 09 20:46 pub
226 Directory send OK.
ftp> cd incoming
250 Directory successfully changed.
ftp> !ls
anaconda-ks.cfg     install.log       vmware-linux-tools.tar.gz
cipe-1.4.5-11.i386.rpm install.log.syslog  webmin-1.000-1.noarch.rpm
//向 incoming 目录中上传文件
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,131,179)
150 Go ahead make my day^W^W^Wsend me the data.
226 File receive OK.
17596 bytes sent in 0.0012 secs (1.4e+04 Kbytes/sec)
//在 incoming 目录中创建目录
ftp> mkdir newdir
257 "/incoming/newdir" created
//列表显示/incoming 目录
ftp> ls
227 Entering Passive Mode (127,0,0,1,108,45)
150 Here comes the directory listing.
-rw-------   1 14      50      17596 Mar 11 08:21 install.log
drwx------   2 14      50       4096 Mar 11 08:52 newdir
226 Directory send OK.
//在新创建的目录中上传文件并创建目录
```

```
ftp> cd newdir
250 Directory successfully changed.
ftp> put anaconda-ks.cfg
local: anaconda-ks.cfg remote: anaconda-ks.cfg
227 Entering Passive Mode (127,0,0,1,112,83)
150 Go ahead make my day^W^W^Wsend me the data.
226 File receive OK.
2306 bytes sent in 0.00196 secs (1.1e+03 Kbytes/sec)
ftp> mkdir newdir2
257 "/incoming/newdir/newdir2" created
//显示/incoming/newdir 目录内容
ftp> ls
227 Entering Passive Mode (127,0,0,1,36,47)
150 Here comes the directory listing.
-rw-------    1 14    50    2306 Mar 11 08:53 anaconda-ks.cfg
drwx------    2 14    50    4096 Mar 11 08:54 newdir2
226 Directory send OK.
ftp> bye
221 Goodbye.
#
//测试结束
```


注意

（1）匿名用户对/var/ftp/incoming 目录而言是其他用户，所以必须为此目录添加对其他用户的可写权限才可上传，即此目录权限的数字表示是 707。

（2）只有设置 anon_world_readable_only=NO 后，才能开放匿名用户的读权限，即：浏览此服务器中全部的内容。

启用 ASCII 传输方式

默认情况下，vsftpd 为了提高传输效率，禁止了 ASCII 传输方式。虽然在 ftp 客户软件中可以使用 asc 命令，但是传输文件时仍然使用二进制传输方式。

可以分别启用上传和下载的 ASCII 传输方式，方法是编辑/etc/vsftpd 配置文件，将如下两行前的#去掉即可启用。

#ascii_upload_enable=YES

#ascii_download_enable=YES

设置连接服务器后的欢迎信息

为了使用户连接服务器后显示信息，vsftpd 提供了两个选项，分别是：

- ftpd_banner.
- banner_file

例如：可以设置如下的 ftpd_banner 选项的值：

ftpd_banner=Welcome to Smartraining FTP service.

即：用户连接服务器后显示信息“ Welcome to Smartraining FTP service. ”。

也可以设置如下的 banner_file 选项的值：

banner_file=/var/vsftpd_banner_file

即：用户连接服务器后显示文件/var/vsftpd_banner_file 中的信息。

警告

（1）如果设置了 ftpd_banner 的值，将覆盖 vsftpd 默认的服务器连接后的信息。

（2）如果 ftpd_banner 和 banner_file 同时设置，那么 banner_file 将覆盖 ftpd_banner 的设置。

配置基本的性能和安全选项

1. 设置空闲的用户会话的中断时间

例如下面的配置：

idle_session_timeout=600

将在用户会话空闲 10 分钟后被中断。

2. 设置空闲的数据连接的的中断时间

例如下面的配置：

data_connection_timeout=120

将在数据连接空闲 2 分钟后被中断。

3. 设置客户端空闲时的自动中断和激活连接的时间

例如下面的配置：

accept_timeout=60

connect_timeout=60

将使客户端空闲 1 分钟后自动中断连接，并在中断 1 分钟后自动激活连接。

4. 设置最大传输速率限制

例如下面的配置：

local_max_rate=50000

anon_max_rate=30000

将使本地用户的最大传输速率为 50kkbytes / sec，匿名用户的传输速率为 30 kbytes / sec。

5. 设置客户端连接时的端口范围

例如下面的配置：

pasv_min_port=50000

pasv_max_port=60000

将使客户端连接时的端口范围在 50000 和 60000 之间。这提高了系统的安全性。

6. 设置 chroot

在默认配置中，本地用户可以切换到自家目录以外的目录进行浏览，并在权限许可的范围内进行下载和上传。这样的设置对于一个 FTP 服务器来说是不安全的。

如果希望用户登录后不能切换到自家目录以外的目录，则需要设置 chroot 选项，涉及如下选项：

- chroot_local_user

- chroot_list_enable
- chroot_list_file

有两种设置 chroot 的方法：

（1）设置所有的本地用户执行 chroot

只要将 chroot_local_user 的值设为 YES 即可，即：

chroot_local_user=YES

（2）设置指定的用户执行 chroot

需要如下的设置：

chroot_local_user=NO

chroot_list_enable=YES

chroot_list_file= /etc/vsftpd.chroot_list

这样，只有/etc/vsftpd.chroot_list 文件中指定的用户才执行 chroot。


注意

上面所提及的文件/etc/vsftpd.chroot_list 和下面将要提及的文件 /etc/vsftpd.user_list 的格式要求均为每个用户名占一行。

配置基于本地用户的访问控制

要配置基于本地用户的访问控制，可以通过修改 vsftpd 的主配置文件/etc/vsftpd.conf 来进行，有如下两种限制方法：

1. 限制指定的本地用户不能访问，而其他本地用户可访问

例如下面的设置：

userlist_enable=NO

userlist_deny=YES

userlist_file= /etc/vsftpd.user_list

使文件/etc/vsftpd.user_list 中指定的本地用户不能访问 FTP 服务器，而其他本地用户可访问 FTP 服务器。

2. 限制指定的本地用户可以访问，而其他本地用户不可访问

例如下面的设置：

userlist_enable= YES

userlist_deny= NO

userlist_file= /etc/vsftpd.user_list

使文件/etc/vsftpd.user_list 中指定的本地用户可以访问 FTP 服务器，而其他本地用户不可以访问 FTP 服务器。

配置基于主机的访问控制

由于 vsftpd 有两种运行方式，即：由 xinetd 启动和独立启动。这两种运行方式的主机访问控制配置是不同的，下面介绍的是由 xinetd 启动的 vsftpd 的主机访问控制的配置。显然，要配置这种主机访问控制，需要修改配置文件/etc/xinetd.d/vsftpd。

1. 只允许指定的主机访问

在配置文件/etc/xinetd.d/vsftpd 的 {} 中添加如下的配置语句：

only_from <主机表>

例如：only_from 192.168.1.0

表示只允许 192.168.1.0 网段内的主机访问。

2. 指定不能访问的主机

在配置文件/etc/xinetd.d/vsftpd 的 {} 中添加如下的配置语句：

no_access <主机表>

例如：no_access 192.168.1.0

表示只有 192.168.1.0 网段内的主机不能访问。

关于主机表的书写形式，见表 10-1。

表 10-1 xinetd 配置访问控制表时主机表的书写语法

选项值	含义
Hostname	可解析的主机名
IP Address	点分十进制表示的 IP 地址
Net_name	在/etc/networks 中定义的网络名
x.x.x.0 x.x.0.0 x.0.0.0 0.0.0.0	0 作为通配符看待。如：191.72.61.0 匹配从 191.72.61.0 到 191.72.61.255 的所有 IP 地址。0.0.0.0 表示匹配所有的 IP 地址
x.x.x.{a,b,…} x.x.{a,b,…} x.{a,b,…}	指定主机表。如：191.72.61.{1,3,123}表示包含地址 191.72.61.1、191.72.61.2 和 191.72.61.123
IPAddress/netmask	定义要匹配的网络或子网。如：172.19.16/20 匹配从 172.19.16.0 到 172.19.31.255

3. 配置每个客户机的最大连接数

在配置文件/etc/xinetd.d/vsftpd 的 {} 中添加如下的配置语句：

per_source = 数值

例如：per_source = 5

表示每个客户机的最大连接数为 5。

4. 配置服务器总的并发连接数

在配置文件/etc/xinetd.d/vsftpd 的 {} 中添加如下的配置语句：

instances = 数值

例如：instances = 200

表示 FTP 服务器总共支持的最高连接数为 200。

5. 配置访问时间限制

在配置文件/etc/xinetd.d/vsftpd 的 {} 中添加如下的配置语句：

access_time = hour:min-hour:min

例如：access_time = 18:00-23:59

表示只有在下午 6 点到午夜 0 点之前才能访问此 FTP 服务器；

又如：access_time = 8:30-11:30 13:00-18:00

表示只有在上午 8 点半到 11 点半和下午 1 点到下午 6 点才能访问此 FTP 服务器。

6. 指定连接失败时显示的信息

在配置文件/etc/xinetd.d/vsftpd 的 {} 中添加如下的配置语句：

banner_fail = 文件名

例如：banner_fail = /etc/vsftpd.busy_banner

表示当连接失败时显示文件/etc/vsftpd.busy_banner 中的内容。

可以用下面的命令生成文件/etc/vsftpd.busy_banner：

技巧

```
# echo "421 Server busy, please try later." \
> /etc/vsftpd.busy_banner
```

10.2.4 配置本地组访问的 FTP 服务器

本节将解决类似这样的问题：

本地组 softgrp 有三个用户 soft、soft1 和 soft2，其中 soft 对 FTP 有读写（包括列文件目录、上传、下载）权限，而 soft1 和 soft2 对 FTP 只有读（包括列文件目录、下载）的权限。

为了实现这种功能，我们要借助于本地文件系统的权限设置来实现，具体操作步骤如下：

操作步骤 10.9　配置本地组访问的 FTP 服务器

```
//创建本地组的 FTP 服务器目录
# mkdir -p /var/local-ftp/softgrp
//创建本地用户和组
# groupadd softgrp
# useradd -G softgrp -d /var/local-ftp/softgrp -M soft
# useradd -G softgrp -d /var/local-ftp/softgrp -M soft1
# useradd -G softgrp -d /var/local-ftp/softgrp -M soft2
//设置用户口令
# passwd soft
# passwd soft1
# passwd soft2
//修改/var/local-ftp/softgrp 的属主和权限
# chown soft.softgrp /var/local-ftp/softgrp
# chmod 750 /var/local-ftp/softgrp
# ll -d /var/local-ftp/softgrp
drwxr-x---    2 soft     softgrp     4096  3 月 12 00:46 /var/local-ftp/softgrp
#
//设置了上面对目录/var/local-ftp/softgrp 的文件系统权限之后
//（1）soft 用户是该目录的属主，因此具有读写权限和进入目录的权限
//（2）soft1 和 soft2 用户属于 softgrp 组，因此只具有读权限和进入目录的权限
//
//配置结束，下面进行测试
# cd
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 Welcome to Smartraining FTP service.
//以 soft 用户身份登录
Name (127.0.0.1:root): soft
331 Please specify the password.
Password:
230 Login successful. Have fun.
```

```
Remote system type is UNIX.
Using binary mode to transfer files.
//上传文件
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,68,177)
150 Go ahead make my day^W^W^Wsend me the data.
226 File receive OK.
17596 bytes sent in 0.00176 secs (9.7e+03 Kbytes/sec)
ftp> ls
227 Entering Passive Mode (127,0,0,1,134,104)
150 Here comes the directory listing.
-rw-r--r--    1 503     504        17596 Mar 11 17:17 install.log
226 Directory send OK.
//用户 soft 写文件成功，并可以浏览
//
//关闭连接
ftp> close
221 Goodbye.
//重新连接
ftp> open 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 Welcome to Smartraining FTP service.
//以 soft1 用户身份登录
Name (127.0.0.1:root): soft1
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
//显示文件
ftp> ls
227 Entering Passive Mode (127,0,0,1,31,75)
150 Here comes the directory listing.
-rw-r--r--    1 503     504        17596 Mar 11 17:17 install.log
226 Directory send OK.
ftp> put anaconda-ks.cfg
local: anaconda-ks.cfg remote: anaconda-ks.cfg
227 Entering Passive Mode (127,0,0,1,184,222)
553 Could not create file.
//拒绝用户 soft1 上传
ftp> bye
221 Goodbye.
#
```


注意

上面的配置主要是在系统中的文件系统上进行的，同时 vsftpd 的配置文件/etc/vsftpd.conf 中要确保以下选项的配置正确：

local_enable=YES

write_enable=YES

chroot_local_user=YES

10.2.5 从源代码安装 vsftpd

用户也可以到 vsftpd 的主站下载 TAR 包自行编译安装 vsftpd。


参考

Vsftpd 的主站为：http://vsftpd.beasts.org/

Vsftpd 的 FTP 服务器为：ftp://vsftpd.beasts.org/users/cevans/

笔者下载了 vsftpd-1.1.3.tar.gz，将其上传到 Linux 服务器，并将其移动到/usr/local/src 目录下。下面介绍安全的步骤：

操作步骤 10.10　从源代码安装 vsftpd

```
//备份 RedHat 8.1 的配置文件
# mkdir –p ~/conf-bak/vsftpd
# cd ~/conf-bak/vsftpd
# cp /etc/vsftpd.conf vsftpd.conf.rh
# cp /etc/xinetd.d/vsftpd vsftpd.xinetd.rh
# cp /etc/pam.d/vsftpd vsftpd.pam.rh
//进入存放 vsftpd 源代码的目录
# cd /usr/local/src
# ls
vsftpd-1.1.3.tar.gz
//解包
# tar -zxvf vsftpd-1.1.3.tar.gz
# cd vsftpd-1.1.3
//编译 vsftpd
# make
//查看是否编译成功
# ls -l vsftpd
-rwxr-xr-x    1 root     root       65996  3 月 12 18:56 vsftpd
//安装 vsftpd
# make install
//安装过程执行了如下操作
// cp vsftpd /usr/local/sbin/vsftpd
```

```
// cp vsftpd.conf.5 /usr/local/man/man5
// cp vsftpd.8 /usr/local/man/man8
// cp xinetd.d/vsftpd /etc/xinetd.d/
#
//复制默认配置文件到/etc 目录
# cp vsftpd.conf /etc
//复制本地用户所需的 PAM 配置文件
# cp RedHat/vsftpd.pam /etc/pam.d/ftp
//重新启动 xinetd
# service xinetd restart
//安装结束
```

10.2.6 使用 vsftpd 配置高安全级别的匿名服务器

高安全级别匿名 FTP 服务器的配置要求

- 仅仅允许匿名用户访问
- 不允许本地用户访问
- 关闭所有写权限
- 不允许匿名用户上传
- 设置客户端连接时的端口范围
- 设置 匿名用户的最大传输速率限制
- 设置空闲的数据连接的的中断时间
- 设置客户端空闲时的自动中断和激活连接的时间
- 配置每个主机的最大连接数
- 配置总的并发连接数
- 配置禁止访问的主机
- 配置安全日志

配置高安全级别的匿名 FTP 服务器


谨慎

vsftpd 推荐使用这种近乎于偏执的安全配置，如果用户只想架设匿名 FTP 下载服务器，出于安全性的考虑，请参考使用这种配置。

配置这种安全匿名服务器，可以从 vsftpd 的源代码树中获得，具体操作步骤如下。

操作步骤 10.11　配置高安全级别的匿名服务器

```
//进入 vsftp 源代码树的 EXAMPLE/INTERNET_SITE 目录
# cd /usr/local/src/vsftpd-1.1.3/EXAMPLE/INTERNET_SITE
//复制配置文件
# cp vsftpd.conf /etc
# cp vsftpd.xinetd /etc/xinetd.d/vsftpd
```

```
//修改主配置文件/etc/vsftpd.conf，添加一个连接后的信息
# cat >>/etc/vsftpd.conf <<!
> ftpd_banner=This FTP server is anonymous only.
> !
#
//显示主配置文件的内容如下
# cat /etc/vsftpd.conf
# Access rights
anonymous_enable=YES
local_enable=NO
write_enable=NO
anon_upload_enable=NO
anon_mkdir_write_enable=NO
anon_other_write_enable=NO
# Security
anon_world_readable_only=YES
connect_from_port_20=YES
hide_ids=YES
pasv_min_port=50000
pasv_max_port=60000
# Features
xferlog_enable=YES
ls_recurse_enable=NO
ascii_download_enable=NO
async_abor_enable=YES
# Performance
one_process_model=YES
idle_session_timeout=120
data_connection_timeout=300
accept_timeout=60
connect_timeout=60
anon_max_rate=50000
ftpd_banner=This FTP server is anonymous only.
#
//此文件中的选项在前面已经介绍过了，此处不再赘述
//接下来查看/etc/xinetd.d/vsftpd
# cat /etc/xinetd.d/vsftpd
service ftp
{
       disable              = no
       socket_type           = stream
       wait                 = no
       user                 = root
       server               = /usr/local/sbin/vsftpd
```

```
        per_source              = 5
        instances               = 200
        no_access               = 192.168.1.3
        banner_fail             = /etc/vsftpd.busy_banner
        log_on_success          += PID HOST DURATION
        log_on_failure          += HOST
}
#
//重新启动 xinetd
# service xinetd restart
#
//配置结束
```

10.2.7 配置基于 IP 的虚拟 FTP 服务器

vsftpd 支持基于 IP 的虚拟 FTP 服务器配置。其主要配置步骤为：

- 配置虚拟 IP 地址
- 建立虚拟 FTP 的服务器目录并设置适当的权限
- 建立虚拟 FTP 的服务器的 xinetd 配置文件
- 建立虚拟 FTP 的服务器的主配置文件


注意

虚拟 FTP 的服务器要有单独的 xinetd 配置文件和单独的主配置文件，这两个文件不能与原配置文件重名。

具体操作过程如下：

操作步骤 10.11　配置基于 IP 的虚拟 FTP 服务器

```
//查看本机现有的 IP 地址
# ifconfig |grep -1 eth0
eth0      Link encap:Ethernet  HWaddr 00:50:56:C7:22:DF
          inet addr:192.168.1.222  Bcast:192.168.1.255  Mask:255.255.255.0
//可以看出本机的第一个网络接口 eth0 的 IP 为 192.168.1.222
//下面配置一个虚拟网络接口 eth0:1
# ifconfig eth0:1 192.168.1.234 up
# # ifconfig |grep -1  eth0:1
eth0:1    Link encap:Ethernet  HWaddr 00:50:56:C7:22:DF
          inet addr:192.168.1.234  Bcast:192.168.1.255  Mask:255.255.255.0
#
//可以看出本机的虚拟网络接口 eth0:1 的 IP 为 192.168.1.234
//
//下面建立虚拟 FTP 的服务器目录
# mkdir -p /var/ftp2/pub
```

```
//确保目录具有如下的权限
# ll -d /var/ftp2
drwxr-xr-x   3 root   root    4096  3月 12 03:00 /var/ftp2
# ll -d /var/ftp2/pub
drwxr-xr-x   2 root   root    4096  3月 12 03:00 /var/ftp2/pub
#
//在下载目录中生成测试文件
# echo "hello">/var/ftp2/pub/test_file
#
//下面创建此虚拟服务器的匿名用户所映射的本地用户 ftp2
# useradd -d /var/ftp2 -M ftp2
#
//更改现有的配置文件/etc/xinetd.d/vsftpd
# vi /etc/xinetd.d/vsftpd
// 在此文件的 {} 中添加如下的配置语句：
// bind = 192.168.1.222
// 将原 FTP 服务绑定到 eth0 接口，之后保存退出 vi
#
//生成新的虚拟 FTP 服务器的配置文件/etc/xinetd.d/vsftpd2
# cp /etc/xinetd.d/vsftpd /etc/xinetd.d/vsftpd2
//更改新的配置文件/etc/xinetd.d/vsftpd2
# vi /etc/xinetd.d/vsftpd2
//在此文件的 {} 中添加如下的配置语句：
// bind = 192.168.1.234
//将虚拟 FTP 服务绑定到 eth0:1 接口
//另外添加如下的配置语句：
// server_args = /etc/vsftpd_site2.conf
//使 vsftpd 读取虚拟 FTP 服务器的主配置文件，之后保存退出 vi
#
//生成虚拟 FTP 服务器的主配置文件/etc/vsftpd_site2.conf
# cp /etc/vsftpd.conf /etc/vsftpd_site2.conf
//修改新的主配置文件
# vi /etc/vsftpd_site2.conf
//将如下的配置语句行：
// ftpd_banner=This FTP server is anonymous only.
//修改为：
// ftpd_banner=This is the alternative FTP site.
//添加如下的配置语句：
// ftp_username=ftp2
//使此虚拟服务器的匿名用户映射到本地用户 ftp2
//这样匿名用户登录后才能进入本地用户 ftp2 的/var/ftp2 目录
//修改后，保存退出 vi
#
//重新启动 xinetd
```

```
# service xinetd restart
#
//配置结束
//
//下面进行测试
//首先以 127.0.0.1 连接本地 FTP 服务器
# ftp 127.0.0.1
ftp: connect: Connection refused
ftp> bye
//由于配置的两个 FTP 服务器，一个被绑定在 192.168.1.222，令一个被绑定在
//192.168.1.234，所以连接失败
#
//连接原 FTP 服务器
# ftp 192.168.1.222
Connected to 192.168.1.222 (192.168.1.222).
220 This FTP server is anonymous only.
//上面显示的是原 FTP 服务器的连接成功后的信息
Name (192.168.1.222:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bye
221 Goodbye.
#
//连接虚拟 FTP 服务器
# ftp 192.168.1.234
Connected to 192.168.1.234 (192.168.1.234).
220 This is the alternative FTP site.
//上面显示的是虚拟 FTP 服务器的连接成功后的信息
Name (192.168.1.234:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bye
221 Goodbye.
#
//测试结束
```

10.2.8 配置独立运行的 vsftpd

配置独立运行的 vsftpd

显然，这种启动方式将不再需要/etc/xinetd.d 下面的配置文件。要配置独立运行的 vsftpd 很简单，只需要在前面的主配置文件的基础上添加如下的配置即可。

设置 listen=YES

指明 vsftpd 以独立运行方式启动。

为了实现访问控制，需要添加如下的配置项：

设置 max_clients=200

指明服务器总的并发连接数

设置 max_per_ip=4

指明每个客户机的最大连接数。

具体操作步骤如下：

操作步骤 10.12　配置独立运行的 vsftpd

```
//复制一个新的主配置文件
# cp /etc/vsftpd.conf /etc/vsftpd.standalone.conf
//编辑新的配置文件/etc/vsftpd.standalone.conf
# vi /etc/vsftpd.standalone.conf
//在文件开始处插入下面的行
//listen=YES
//max_clients=200
//max_per_ip=4
//并将下面的配置语句
//ftpd_banner=This FTP server is anonymous only.
//改为：
//ftpd_banner=This FTP server is anonymous only,
// and vsftpd in "standalone" mode.
//(注意：要写在一行里)
//用:wq 退出 vi
#
//将由 xinetd 启动的配置停用
# vi /etc/xinetd.d/vsftpd
//将 disable = no
//设为 disable = yes
//保存后退出 vi
# vi /etc/xinetd.d/vsftpd2
//将 disable = no
//设为 disable = yes
//保存后退出 vi
// 重新启动 xinetd
# service xinetd restart
// 启动独立运行的 vsftpd 守护进程
# /usr/local/sbin/vsftpd /etc/vsftpd.standalone.conf  &
```

```
//配置结束，下面进行测试
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 This FTP server is anonymous only, and vsftpd in "standalone" mode.
Name (127.0.0.1:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bye
221 Goodbye.
#
```

配置独立运行的 vsftpd 在非标准端口下提供服务

为了配置独立运行的 vsftpd 在非标准端口下提供服务需要添加 listen_port 配置语句。具体操作步骤如下：

操作步骤 10.13　配置独立运行的 vsftpd 在非标准端口下提供服务

```
//修改独立运行的主配置文件，添加 listen_port 配置语句
# echo “listen_port=10021”>> /etc/vsftpd.standalone.conf
//重新启动 vsftpd 守护进程
# killall vsftpd
# /usr/local/sbin/vsftpd /etc/vsftpd.standalone.conf  &
//下面进行测试
# ftp 127.0.0.1 10021
Connected to 127.0.0.1 (127.0.0.1).
220 This FTP server is anonymous only, and vsftpd in "standalone" mode.
Name (127.0.0.1:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bye
221 Goodbye.
#
```

比较独立启动和 xinetd 启动的 vsftpd 服务器配置和启动

下面比较两种启动方式的 vsftp 配置，见表 10-2。

表 10-2 独立启动和 xinetd 启动的 vsftpd 服务器配置和启动的比较

	独立启动	xinetd 启动	
所需的配置文件	主配置文件	主配置文件	xinetd 配置文件
配置	listen		
	listen_address		bind
	listen_port		
	max_per_ip		per_source
	max_clients		instances
			only_from 和 no_access
			access_time
			server_args
启动	将配置文件作为参数运行 vsftpd	启动 xinetd	

用 vsftpd 配置独立运行的虚拟 FTP 服务器

为了配置独立运行的虚拟 FTP 服务器需要使用 listen_address 配置语句。笔者以 10.2.7 节的配置和上面的配置为基础进行下面的配置，具体步骤如下：

操作步骤 10.14 用 vsftpd 配置独立运行的虚拟 FTP 服务器

```
//修改原独立运行的服务器配置文件，
# vi /etc/vsftpd.standalone.conf
//添加 listen_address=192.168.1.222 的配置行
//将原 FTP 服务绑定到 eth0 接口。之后用:wq 退出 vi
#
//由 xinetd 运行的虚拟服务器的配置文件创建一个新的配置文件
# cp /etc/vsftpd_site2.conf /etc/vsftpd.standalone2.conf
//修改新的配置文件
# vi /etc/vsftpd.standalone2.conf
//在文件开始处插入下面的行
//listen=YES
//listen_address=192.168.1.234 → 将虚拟服务器绑定到 eth0:1 接口
//max_clients=200
//max_per_ip=4
//并将下面的配置语句
//ftpd_banner=This is the alternative FTP site.
//改为：
//ftpd_banner=This is the alternative FTP site,
// and vsftpd in "standalone" mode.
```

```
//(注意：要写在一行里)
//用:wq 退出 vi
#
//让原 FTP 服务器重新读取配置文件
# ps auxw|grep vsftpd
root          7164   0.0   0.2   1424   376  pts/0      S      22:06     0:00
/usr/local/sbin/vsftpd /etc/vsftpd.standalone.conf
# kill -HUP 7164
//上面的 7164 是由 ps 命令查得的 PID
//启动新的虚拟 FTP 服务器
# /usr/local/sbin/vsftpd /etc/vsftpd.standalone2.conf  &
#
//下面进行测试
//首先以 127.0.0.1 连接本地 FTP 服务器
# ftp 127.0.0.1
ftp: connect: Connection refused
ftp> bye
//由于配置的两个 FTP 服务器，一个被绑定在 192.168.1.222，令一个被绑定在
//192.168.1.234，所以连接失败
#
//以 192.168.222 连接原 FTP 服务器
# ftp 192.168.1.222
ftp: connect: Connection refused
ftp> bye
//由于原服务器的配置已经被修改为监听 10021 端口，所以连接失败
#
//以 10021 端口连接原 FTP 服务器
# ftp 192.168.1.222 10021
Connected to 192.168.1.222 (192.168.1.222).
220 This FTP server is anonymous only, and vsftpd in "standalone" mode.
Name (192.168.1.222:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd pub
250 Directory successfully changed.
ftp> ls
227 Entering Passive Mode (192,168,1,222,223,159)
150 Here comes the directory listing.
-r-xr-xr-x    1 ftp     ftp      6477959 Mar 09 19:41 webmin-1.000-1.noarch.rpm
226 Directory send OK.
ftp> bye
```

```
221 Goodbye.
#
//连接虚拟 FTP 服务器
# ftp 192.168.1.234
Connected to 192.168.1.234 (192.168.1.234).
220 This is the alternative FTP site,  and vsftpd in "standalone" mode.
Name (192.168.1.234:root): ftp
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd pub
250 Directory successfully changed.
ftp> ls
227 Entering Passive Mode (192,168,1,234,224,31)
150 Here comes the directory listing.
-rw-r--r--    1 ftp     ftp           6 Mar 12 12:09 test_file
226 Directory send OK.
ftp> bye
221 Goodbye.
#
//测试结束
```

10.2.9 配置虚拟用户的 FTP 服务器

本节讲述虚拟用户（virtual users）的 FTP 服务器的配置。虚拟用户只能访问为其提供的 FTP 服务，虚拟用户不能像本地的实用户那样登录系统而访问系统的其他资源。若用户对 FTP 服务器站内具有写权限并且不允许访问系统的其他资源，则该用户应该使用虚拟用户才能提高系统的安全性。

传统的 FTP 服务器采用如下的方法实现虚拟用户：

- 在本地建立普通用户账号并设置密码
- 将其登录 shell 设为不可登录
- 由 passwd/shadow 口令系统进行认证

vsftpd 的虚拟用户采用了不与系统账户口令文件合二为一的方法，也就是说，为了认证这些虚拟用户 vsftpd 使用单独的口令库文件（pam_userdb），由可插拔认证模块（PAM）进行认证。使用这种方式更加安全，并且配置更灵活。

比较 vsftpd 中的三类用户

vsftp 中有三类用户，即：本地用户、虚拟用户和匿名用户。下面将这三类用户进行比较，见表 10-3。

表 10-3 比较 vsftpd 中的三类用户

	本地用户	虚拟用户	匿名用户
用户名	本地用户名	虚拟用户口令库中指定的用户名	Anonymous 或 ftp
登录用户名	本地用户名	虚拟用户口令库中所有用户名	Anonymous 或 ftp
用户口令	本地用户的口令	虚拟用户口令库中指定的口令	Email 地址
口令的认证方式	由基于 passwd/shadow 的口令系统认证	DB 口令库由 PAM 进行认证	由 vsftpd 认证
登录映射的本地用户名	本地用户名	guest_username 所指定的本地用户，默认为空	ftp_username 所指定的本地用户，默认为 ftp
登录后进入的目录	本地用户的自家目录	guest_username 所指定的本地用户的自家目录	ftp_username 所指定的本地用户的自家目录，默认为/var/ftp
对登录后的目录是否可浏览	可以	anon_world_readable_only=NO 时可以	anon_world_readable_only=NO 时可以
对登录后的目录是否可上传	write_enable=YES 时可以	write_enable=YES，同时 anon_upload_enable=YES 时可以	write_enable=YES ，同时 anon_upload_enable=YES 时可以
对登录后的目录是否可创建目录	write_enable=YES 时可以	write_enable=YES，同时 anon_mkdir_write_enable=YES 时可以	write_enable=YES ，同时 anon_mkdir_write_enable=YES 时可以
对登录后的目录是否可改名和删除	write_enable=YES 时可以	write_enable=YES，同时 anon_other_write_enable=YES 时可以	write_enable=YES ，同时 anon_other_write_enable=YES 时可以
是否有用户自家目录	有	无	无
是否能切换到登录目录以外的目录	chroot_local_user=NO 时能，其值为 YES 时不能	不能，即设置 chroot_local_user=YES	不能
激活此类用户的命令	local_enable=YES	guest_enable=YES	anonymous_enable=YES

配置简单的虚拟用户 FTP 服务器

为了配置虚拟用户的 FTP 服务器，其主要配置步骤为：

- 生成虚拟用户口令库文件

- 配置生成 vsftpd 的认证文件
- 建立虚拟用户所要访问的目录并设置相应权限
- 建立配置文件

下面以独立运行的 vsftpd 为例进行配置，具体的操作过程如下：

操作步骤 10.15　配置简单的虚拟用户 FTP 服务器

```
//生成虚拟用户口令库文件，为了建立此口令库文件，先要生成一个文本文件
# cat <<! >logins.txt
> tom
> foo
> fred
> bar
> valid
> lwd
> dede
> jy
> !
#
//此文本文件的格式是：
//单数行为用户名，偶数行为口令，
//即：用户 tom 的口令为 foo；用户 fred 的口令为 bar
//
//下面使用 db_load 命令生成口令库文件
# db_load -T -t hash -f ~/logins.txt /etc/vsftpd_login.db
//修改口令库文件的权限
# chmod 600 /etc/vsftpd_login.db
//下面编辑生成虚拟用户所需的 PAM 配置文件
# vi /etc/pam.d/ftp.vu
//插入如下两行
//auth required /lib/security/pam_userdb.so db=/etc/vsftpd_login
//account required /lib/security/pam_userdb.so db=/etc/vsftpd_login
//保存后退出 vi
//用户也可以用下面的命令从源代码分发包中复制此文件
// cd /usr/local/src/vsftpd-1.1.3/EXAMPLE/
// cp VIRTUAL_USERS/vsftpd.pam /etc/pam.d/ftp.vu
#
//下面建立虚拟用户所要访问的目录并设置仅 virtual 用户访问的权限
# useradd -d /home/ftpsite virtual
# chmod 700 /home/ftpsite/
#
//在目录中生成测试文件
# su - virtual -c "echo hello>/home/ftpsite/test_file"
#
```

```
//下面生成主配置文件
# vi /etc/vsftpd.standalone.vu.conf
//在此文件中插入下面的配置语句
//listen=YES
//anonymous_enable=NO
//local_enable=YES
//write_enable=NO
//anon_upload_enable=NO
//anon_mkdir_write_enable=NO
//anon_other_write_enable=NO
//chroot_local_user=YES
//guest_enable=YES → 启用虚拟用户
//guest_username=virtual → 将虚拟用户映射为本地 virtual 用户
// 这样虚拟用户登录后才能进入本地用户 virtual 的目录/home/ftpsite/
//pasv_min_port=30000
//pasv_max_port=30999
//ftpd_banner=This FTP server is virtual user only.
//pam_service_name=ftp.vu → 指定 PAM 配置文件为 ftp.vu
//插入完毕，保存退出。
#
//下面先关闭原来的服务，而后启动新的服务
# killall vsftpd
# /usr/local/sbin/vsftpd /etc/vsftpd.standalone.vu.conf &
//配置结束
```

配置好后，新的口令库中的所有用户就都可以登录此 FTP 服务器了。下面是测试过程：

操作步骤 10.16　测试简单的虚拟用户 FTP 服务器

```
//以虚拟用户 tom 测试
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 This FTP server is virtual user only.
Name (127.0.0.1:root): tom
331 Please specify the password.
//下面输入虚拟用户 tom 的口令 foo
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> pwd
257 "/"
ftp> ls
227 Entering Passive Mode (127,0,0,1,120,186)
150 Here comes the directory listing.
```

```
226 Transfer done (but failed to open directory).
//无法列表显示文件
//这是由于配置语句 anon_world_readable_only 的默认值为 YES。
//这样的设置是最安全的，是建议的配置
//若要虚拟用户口令库中的人都能看到登录目录的内容，
//应该在配置文件/etc/vsftpd.standalone.vu.conf 中添加如下配置语句
// anon_world_readable_only=NO
//
ftp> size test_file
213 6
//能够显示以存在的文件的大小
ftp> bye
221 Goodbye.
#
//测试结束
```

对不同的虚拟用户进行不同权限的配置

在 10.2.4 节中讲述了配置本地组访问的 FTP 服务器，即使组长（soft）具有读写（包括列文件目录、上传、下载）权限，组员（soft1 和 soft2）具有只读（包括列文件目录、下载）权限。这是借助在本地文件系统上分配权限来实现的，但是若要让两个用户同时都具有读写权限，则用这种方法就不能实现了。

vsftpd 支持对不同的虚拟用户进行不同配置的方式。例如，vsftpd 可以实现如下的配置：

- 虚拟用户 valid 具有浏览目录、上传和下载的权限
- 虚拟用户 dede 具有浏览目录、上传、下载、文件改名和删除的权限
- 虚拟用户 tom 和 fred 具有浏览目录和下载的权限

下面以简单的虚拟用户 FTP 服务器的配置为基础进行，具体配置步骤为：

操作步骤 10.17　对 vsftpd 的不同虚拟用户分配不同权限的配置

```
//首先编辑配置文件/etc/vsftpd.standalone.vu.conf
//激活对不同的虚拟用户进行不同权限配置的配置语句
# vi /etc/vsftpd.standalone.vu.conf
//在文件中添加如下的配置行
// user_config_dir=/etc/vsftpd_user_conf
//指定不同虚拟用户配置文件的存放路径
//添加后保存退出 vi
//接下来创建此目录
# mkdir /etc/vsftpd_user_conf
#
//下面分别创建虚拟用户 tom 和 fred 的配置文件
//开放 tom 和 fred 的读权限
#echo "anon_world_readable_only=NO">/etc/vsftpd_user_conf/tom
#echo "anon_world_readable_only=NO">/etc/vsftpd_user_conf/fred
```

```
//下面分别创建虚拟用户 valid 和 dede 的配置文件
//开放 valid 和 dede 的读写权限
# cat <<! >/etc/vsftpd_user_conf/valid
> anon_world_readable_only=NO
> write_enable=YES
> anon_upload_enable=YES
> !
# cp /etc/vsftpd_user_conf/valid /etc/vsftpd_user_conf/dede
# echo “anon_other_write_enable=YES”>>/etc/vsftpd_user_conf/dede
//经过以上的设置虚拟用户 valid 能浏览、下载和上传
//而虚拟用户 dede 除此之外还具有文件改名和删除文件的权限
#
//下面让守护进程 vsftpd 重新读取配置文件
//  /etc/vsftpd.standalone.vu.conf
# ps auxw|grep vsftpd
root          7218   0.0   0.2   1424   376  pts/0      S      01:37     0:00
/usr/local/sbin/vsftpd /etc/vsftpd.standalone.vu.conf
# kill -HUP 7218
#
//配置结束
```

下面进行测试，步骤如下：

操作步骤 10.18　对 vsftpd 的不同虚拟用户分配不同权限的配置测试

```
# cd
//连接本地 FTP 服务器，以虚拟用户 tom 进行测试
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 This FTP server is virtual user only.
Name (127.0.0.1:root): tom
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (127,0,0,1,120,188)
150 Here comes the directory listing.
-rw-rw-r--    1 507     508           6 Mar 12 17:49 test_file
226 Directory send OK.
//浏览成功
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,119,172)
```

```
550 Permission denied.
//上传被拒绝
ftp> bye
221 Goodbye.
#
//连接本地 FTP 服务器，以虚拟用户 valid 进行测试
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 This FTP server is virtual user only.
Name (127.0.0.1:root): valid
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> put install.log
local: install.log remote: install.log
227 Entering Passive Mode (127,0,0,1,117,75)
150 Ok to send data.
226 File receive OK.
17596 bytes sent in 0.00456 secs (3.8e+03 Kbytes/sec)
//上传成功
ftp> ls
227 Entering Passive Mode (127,0,0,1,117,59)
150 Here comes the directory listing.
-rw-------   1 507     508        17596 Mar 12 19:15 install.log
-rw-rw-r--   1 507     508            6 Mar 12 17:49 test_file
226 Directory send OK.
//浏览成功
ftp> delete test_file
550 Permission denied.
//拒绝删除文件
ftp> rename test_file test
550 Permission denied.
//拒绝对文件改名
ftp> bye
221 Goodbye.
#
//连接本地 FTP 服务器，以虚拟用户 dede 进行测试
# ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 This FTP server is virtual user only.
Name (127.0.0.1:root): dede
331 Please specify the password.
```

```
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
//在本地主机生成一个新文件
ftp> !touch newfile
ftp> put newfile
local: newfile remote: newfile
227 Entering Passive Mode (127,0,0,1,118,178)
150 Ok to send data.
226 File receive OK.
//上传成功
ftp> ls
227 Entering Passive Mode (127,0,0,1,117,59)
150 Here comes the directory listing.
-rw-------    1 507     508        17596 Mar 12 19:15 install.log
-rw-------    1 507     508            0 Mar 12 19:26 newfile
-rw-rw-r--    1 507     508            6 Mar 12 17:49 test_file
226 Directory send OK.
//浏览成功
ftp> rename test_file test
350 Ready for RNTO.
250 Rename successful.
//文件改名成功
ftp> delete newfile
250 Delete operation successful.
//删除文件成功
ftp> ls
227 Entering Passive Mode (127,0,0,1,118,11)
150 Here comes the directory listing.
-rw-------    1 507     508      17596 Mar 12 19:15 install.log
-rw-rw-r--    1 507     508          6 Mar 12 17:49 test
226 Directory send OK.
ftp> bye
221 Goodbye.
#
//测试结束
```

重点

当一个虚拟用户登录 FTP 服务器时，vsftpd 的守护进程首先查看主配置文件的权限配置，然后再用此用户单独的配置文件中的配置覆盖主配置文件中的配置。

vsftpd 的这种配置机制，类似于 Apache 配置中的 httpd.conf

和.htaccess。只不过前者是针对不同虚拟用户的，而后者是针对不同目录的。

因此在对不同虚拟用户进行配置时要注意：

（1）在主配置文件中设置最低的权限，这些设置对虚拟用户口令库中的所有用户均生效；

（2）分别在不同的虚拟用户的配置文件中开放此虚拟用户应该具有的权限。

注意

本节讲述的虚拟用户 FTP 服务器的配置是基于独立运行的 vsftpd 守护进程讲解的。

用户可以配置为基于 xinetd 启动的 FTP 服务器，同时也可以将虚拟用户的 FTP 服务器配置为基于 IP 的虚拟 FTP 服务器。请读者自行练习。