# **vsftp**

# 1 软件信息

**服务端软件名：**vsftpd

**客户端软件名：**ftp

**服务名：**vsftpd

**端口号：**20、21、指定范围内随机端口

**配置文件：**/etc/vsftpd/vsftpd.conf

# 2登录验证方式

## 2.1匿名用户验证：

用户账号名称：ftp或anonymous

用户账号密码：无密码

工作目录：/var/ftp

默认权限：默认可下载不可上传，上传权限由两部分组成（主配置文件和文件系统）

## 2.2本地用户验证：

用户账号名称：本地用户（/etc/passwd）

用户账号密码：用户密码（

/etc/shadow）

工作目录：登录用户的宿主目录

权限：最大权限（

drwx------）

## 2.3 虚拟（virtual）用户验证：

1. 创建虚拟用户用来代替本地用户，减少本地用户曝光率

2. 使用本地用户作为虚拟用户的映射用户，为虚拟用户提供工作目录和权限控制

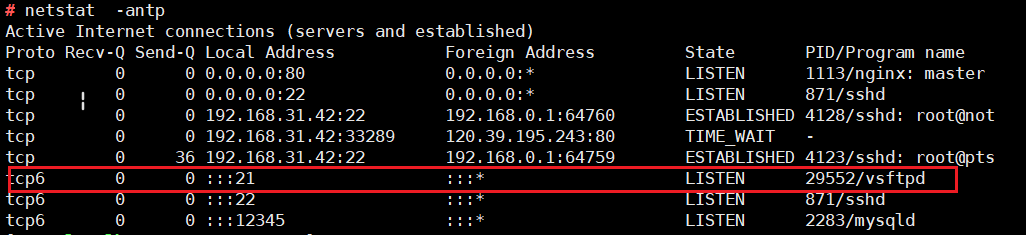
3. 能够设置严格的权限（为每一个用户生成单独的配置文件）

# 3 vsftp部署

安装  
yum install -y vsftpd   ftp #vsftpd服务端 ftp客户端  
   
开机自启  
systemctl enable vsftpd  
   
启动  
systemctl start vsftpd

查看端口开放情况

netstat -antp



## **3.1 匿名用户验证实验：**

**匿名权限控制：**

|  |
| --- |
| anonymous\_enable=YES #启用匿名访问  anon\_umask=022 #匿名用户所上传文件的权限掩码  anon\_root=/var/ftp #匿名用户的 FTP 根目录  anon\_upload\_enable=YES #允许上传文件  anon\_mkdir\_write\_enable=YES #允许创建目录  anon\_other\_write\_enable=YES #开放其他写入权(删除、覆盖、重命名)  anon\_max\_rate=0 #限制最大传输速率（0 为不限速，单位：bytes/秒） |

**实验需求与流程：**

注意：在客户端登录后，默认情况下是可以下载的，但不能上传

**1. 实现可以上传**

a. anon\_upload\_enable=YES

b. 在/var/ftp/下创建上传目录 # mkdir upload

c. 修改上传目录的权限或所有者，让匿名用户有写入权限 # chmod o+w upload

d. 重启vsftpd # systemctl restart vsftpd





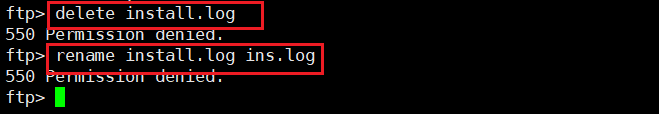
**2. 实现创建目录和文件其他操作**

|  |
| --- |
| anon\_mkdir\_write\_enable=YES  #允许创建目录  anon\_other\_write\_enable=YES  #删除文件、文件改名、文件覆盖 |

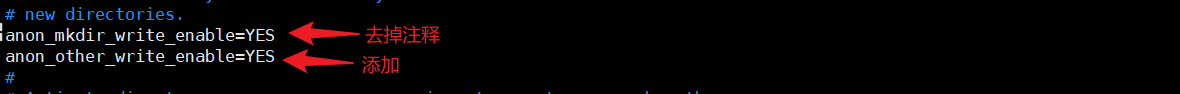
未添加配置前，没有权限创建目录



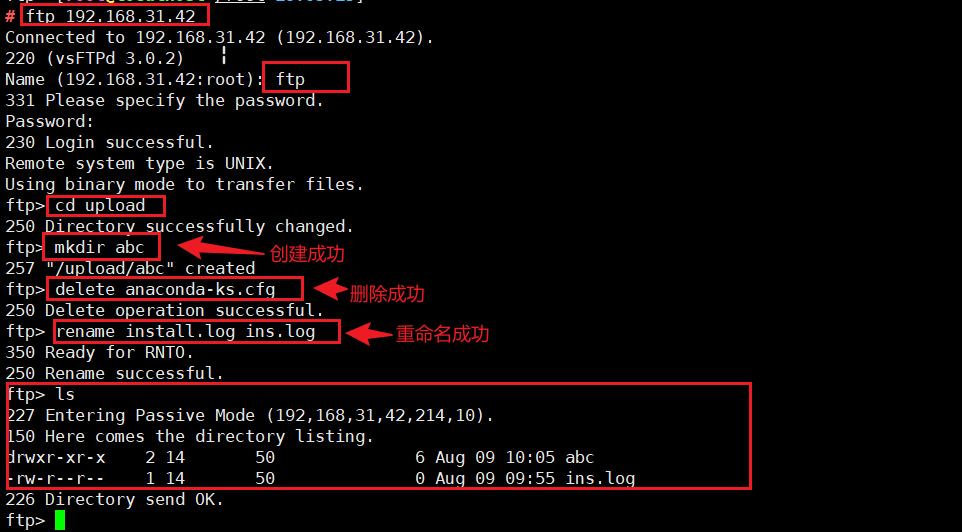
删除，重命名也没有权限



# vim /etc/vsftpd/vsftpd.conf



# systemctl restart vsftpd



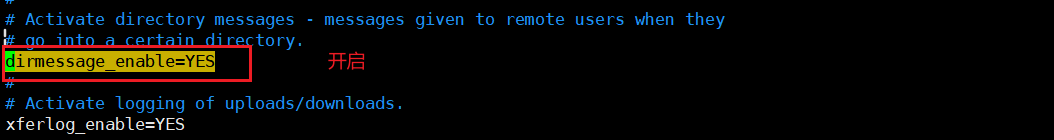
**3. 用户进入某个文件夹时，弹出相应的说明**

|  |
| --- |
| a. 在对应目录下创建 .message 文件，并写入相应内容  b. 确认dirmessage\_enable=YES是否启用  c. 尝试却换目录查看效果（同一次登录仅提示一次） |

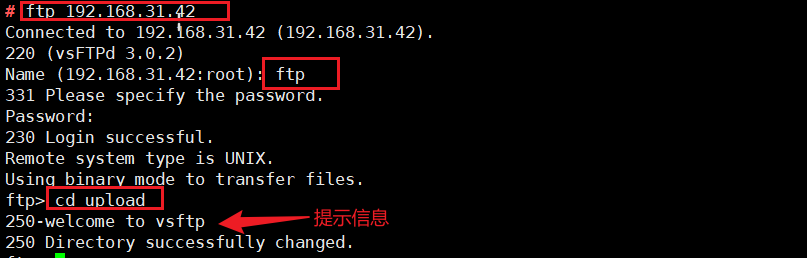
**# vim /var/ftp/upload/.message**



# vim /etc/vsftpd/vsftpd.conf



# systemctl restart vsftpd



**4. 实现上传的文件可下载**

|  |
| --- |
| 默认情况下开放上传权限后，上传的文件是无法被下载的，因为文件的其他人位置没有r权限  设置anon\_umask=022，可以让上传的文件其他人位置拥有r权限，然后才能被其他人下载 |

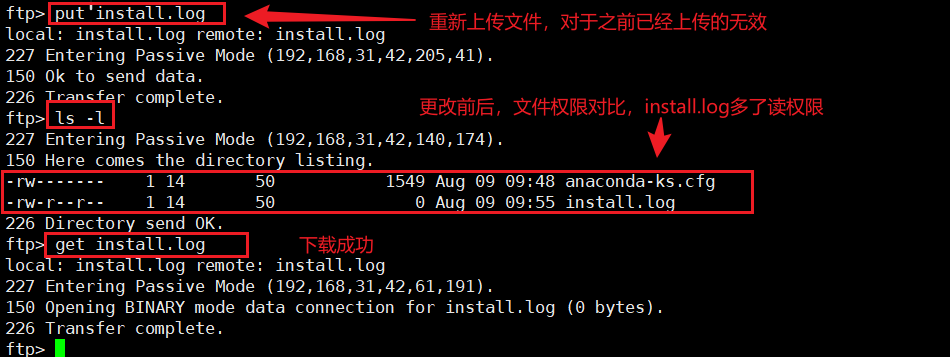
# vim /etc/vsftpd/vsftpd.conf

添加anon\_umask=022

# systemctl restart vsftpd

重新连接vsftp

# ftp 192.168.31.42



### 3.1.2匿名登陆

# ftp 192.168.31.42



## **3.2. 本地用户验证实验：**

**Centos7 本地用户无法登陆530 Login incorrect.**

**本地用户权限控制：**

|  |
| --- |
| local\_enable=YES #是否启用本地系统用户  local\_umask=022 #本地用户所上传文件的权限掩码  local\_root=/var/ftp #设置本地用户的 FTP 根目录  chroot\_local\_user=YES #是否将用户禁锢在主目录  local\_max\_rate=0 #限制最大传输速率  ftpd\_banner=Welcome to blah FTP service #用户登录时显示的欢迎信息  userlist\_enable=YES & userlist\_deny=YES #禁止/etc/vsftpd/user\_list 文件中出现的用户名登录 FTP  userlist\_enable=YES & userlist\_deny=NO #仅允许/etc/vsftpd/user\_list 文件中出现的用户名登录 FTP  配置文件：ftpusers #禁止/etc/vsftpd/ftpusers 文件中出现的用户名登录 FTP,权限比 user\_list 更高，即时生效 |

**实验需求与流程：**

**1. 服务端需要创建用户并设置密码(所创建的用户，不需要登录操作系统，仅用来登录VSFTP)**

useradd -s /sbin/nologin username

#useradd -s /sbin/nologin meta

#passwd meta

#useradd -s /sbin/nologin lisi

#passwd lisi

**2. 将所有用户禁锢在自己的家目录下**

注：默认没有禁锢用户时，客户端登录后可以随意切换目录，查看文件所在位置和文件名

chroot\_local\_user=YES

#开启用户家目录限制，限制所有用户不能随便切换目录

**3. 将部分用户禁锢在自己的家目录下**

chroot\_list\_enable=YES

#开启白名单功能，允许白名单中的用户随意切换目录

chroot\_list\_file=/etc/vsftpd/chroot\_list

#白名单文件所在位置（需自己创建）

**4. 配置文件：/etc/vsftpd/ftpusers**

所有写入此文件内的用户名都不允许登录ftp，立刻生效。

**5. 修改被动模式数据传输使用端口**

pasv\_enable=YES

pasv\_min\_port=30000

pasv\_max\_port=35000

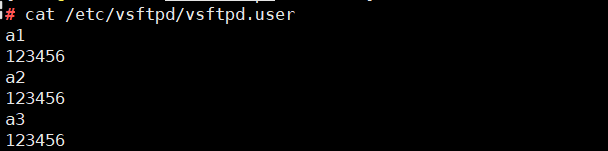
## 3.3虚拟用户验证实验：

**1. 建立 FTP 的虚拟用户的用户数据库文件（在/etc/vsftpd）**

vim vsftpd.user

#vim /etc/vsftpd/vsftpd.user

#cat /etc/vsftpd/vsftpd.user



注：该文件名可以随便定义，文件内容格式：奇数行用户，偶数行密码

db\_load -T -t hash -f vsftpd.user vsftpd.db

# db\_load -T -t hash -f /etc/vsftpd/vsftpd.user /etc/vsftpd/vsftpd.db

#将用户密码的存放文本转化为数据库类型，并使用 hash 加密

chmod 600 vsftpd.db

# chmod 600 /etc/vsftpd/vsftpd.db

#修改文件权限为 600，保证其安全性

**2. 创建 FTP 虚拟用户的映射用户，并指定其用户家目录**

useradd -d /var/ftproot -s /sbin/nologin virtual

# useradd -d /var/ftproot -s /sbin/nologin virtual

#创建 virtual 用户作为 ftp 的虚拟用户的映射用户

**3. 建立支持虚拟用户的 PAM 认证文件，添加虚拟用户支持**

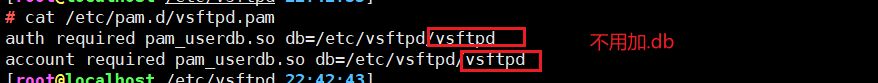
cp –a /etc/pam.d/vsftpd /etc/pam.d/vsftpd.pam

#使用模板生成自己的认证配置文件，方便一会调用

**编辑新生成的文件 vsftpd.pam (清空原来内容，添加下列两行)**

auth required pam\_userdb.so db=/etc/vsftpd/vsftpd

account required pam\_userdb.so db=/etc/vsftpd/vsftpd



此处不用加.db

**在 vsftpd.conf 文件中添加支持配置**

修改：

pam\_service\_name=vsftpd.pam

添加：

guest\_enable=YES

guest\_username=virtual

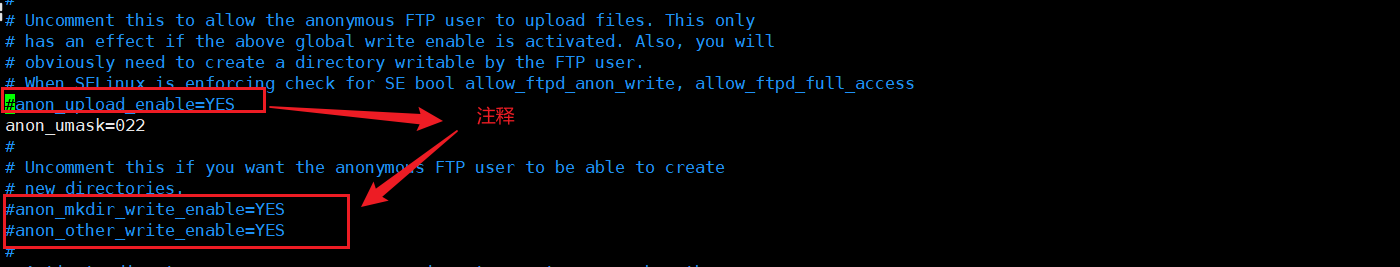
user\_config\_dir=/etc/vsftpd/dir

allow\_writeable\_chroot=yes

# mkdir /etc/vsftpd/dir

**4. 为虚拟用户建立独立的配置文件，启动服务并测试**

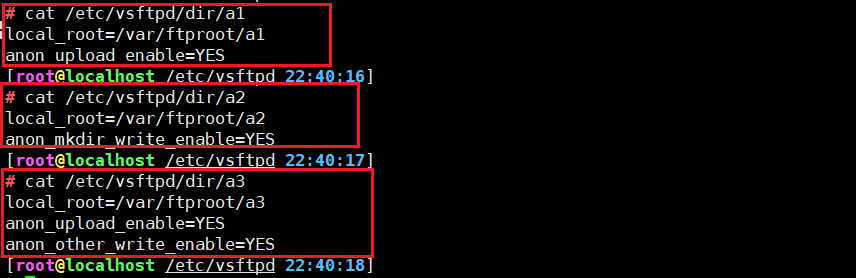
**注：做虚拟用户配置文件设置时，将主配置文件中自定义的匿名用户相关设置注释掉。**



**创建3个用户，权限如下**

**#mkdir /var/ftproot/{a1,a2,a3}**

**#chown virtual.virtual /var/ftproot/{a1,a2,a3}**



**用户可以上传:**

anon\_upload\_enable=YES

#允许上传文件

**用户可以创建目录或文件:**

anon\_mkdir\_write\_enable=YES

#允许创建目录

**用户可以修改文件名:**

anon\_upload\_enable=YES

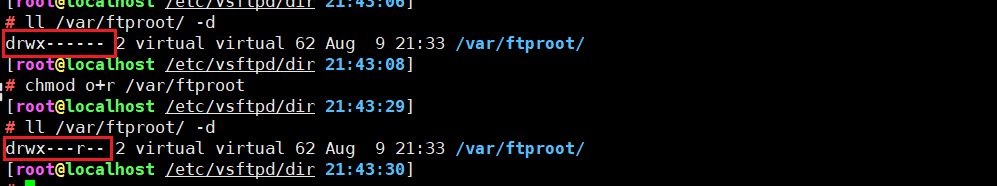
#允许上传文件（为了覆盖开启的）

anon\_other\_write\_enable=YES

#允许重名和删除文件、覆盖

**注：给映射用户的家目录 设置 o+r 让虚拟用户有读权限。**

**#** chmod o+r /var/ftproot



完整配置文件如下

|  |
| --- |
| anonymous\_enable=YES  local\_enable=YES  write\_enable=YES  local\_umask=022  anon\_umask=022  dirmessage\_enable=YES  xferlog\_enable=YES  connect\_from\_port\_20=YES  xferlog\_std\_format=YES  ftpd\_banner=Welcome to my FTP service.  chroot\_local\_user=YES  chroot\_list\_enable=YES  chroot\_list\_file=/etc/vsftpd/chroot\_list  listen=NO  listen\_ipv6=YES  userlist\_enable=YES  tcp\_wrappers=YES  local\_max\_rate=0  pam\_service\_name=vsftpd.pam  guest\_enable=YES  guest\_username=virtual  user\_config\_dir=/etc/vsftpd/dir  allow\_writeable\_chroot=yes |

## **3.4. openssl+vsftpd 加密验证方式：**

**拓展：使用tcpdump 工具进行指定端口抓包，抓取ftp登录过程中的数据包**

tcpdump -i eth0 -nn -X -vv tcp port 21 and ip host 来源ip

-i

#interface：指定tcpdump需要监听的接口

-n

#对地址以数字方式显式，否则显式为主机名

-nn

#除了-n的作用外，还把端口显示为数值，否则显示端口服务名

-X

#输出包的头部数据，会以16进制和ASCII两种方式同时输出

-vv

#产生更详细的输出

**1. 查看是否安装了 openssl**

rpm -q openssl

**2. 查看 vsftpd 是否支持 openssl**

ldd /usr/sbin/vsftpd | grep libssl

**3. 生成加密信息的秘钥和证书文件**

位置：/etc/ssl/certs/

a. openssl genrsa -out vsftpd.key 1024

#建立服务器私钥，生成 RSA 密钥

b. openssl req -new -key vsftpd.key -out vsftpd.csr

#需要依次输入国家，地区，城市，组织，组织单位，

Email 等信息。最重要的是有一个 common name，

可以写你的名字或者域名。如果为了 https 申请，这个必须和域名吻合，否则会引发浏览器警报。

生成的 csr 文件交给 CA 签名后形成服务端自己的证书

c. openssl x509 -req -days 365 -sha256 -in vsftpd.csr -signkey vsftpd.key -out

vsftpd.crt

#使用 CA 服务器签发证书，设置证书的有效期等信息

**注意 1：生成完秘钥和证书文件后，将本目录{/etc/ssl/certs/}的权限修改为 500.**

**注意 2：在实验环境中可以用命令生成测试，在生产环境中必须要在 https 证书厂商注册（否则浏览器不识别）**

**4. 修改主配置文件/etc/vsftpd/vsftpd.conf**

ssl\_enable=YES

#启用 ssl 认证

ssl\_tlsv1=YES

ssl\_sslv2=YES

ssl\_sslv3=YES

#开启 tlsv1、sslv2、sslv3 都支持

allow\_anon\_ssl=YES

#允许匿名用户{虚拟用户}

force\_anon\_logins\_ssl=YES

force\_anon\_data\_ssl=YES

#匿名登录和传输时强制使用 ssl

force\_local\_logins\_ssl=YES

force\_local\_data\_ssl=YES

#本地登录和传输时强制使用 ssl

rsa\_cert\_file=/etc/ssl/certs/itdance.cn.cert

#rsa格式的证书

rsa\_private\_key\_file=/etc/ssl/certs/itdance.cn.key

#rsa格式的密钥

**注：密钥文件要在配置文件中单独声明（写入配置文件时，注释要单独一行，否则会报错）**

**5. 重启服务**

service vsftpd restart

**6. 测试(使用第三方客户端连接)**