网络服务-VSFTP

1. VSFTP 概述

FTP 是 File Transfer Protocol（文件传输协议）的英文简称，用于 Internet 上的文件的双向传输。使用 FTP 来传输时，是具有一定程度的危险性， 因为数据在因特网上面是完全没有受到保护的明文传输方式！

VSFTP 是一个基于 GPL 发布的类 Unix 系统上使用的 FTP 服务器软件，它的全称是 Very Secure FTP，从名称定义上基本可以看出，这是为了解决 ftp 传输安全性问题的。

1.1 安全特性

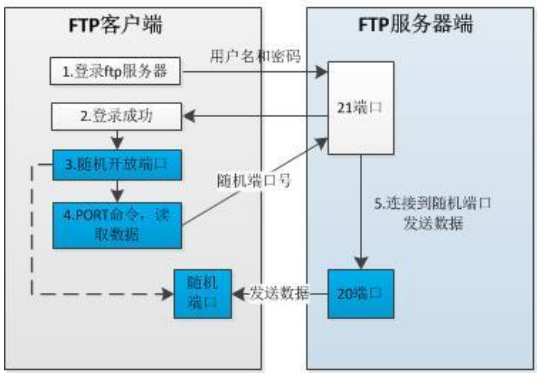
1. vsftp 程序的运行者一般是普通用户，降低了相对应进程的权限，提高了安全性
2. 任何需要执行较高权限的指令都需要上层程序许可
3. ftp 所需要使用的绝大多数命令都被整合到了 vsftp 中，基本不需要系统额外提供命令
4. 拥有 chroot 功能，可以改变用户的根目录，限制用户只能在自己的家目录

2. VSFTP 连接类型

控制连接（持续连接） → TCP 21（命令信道） → 用户收发FTP命令
数据连接（按需连接） → TCP 20（数据信道） → 用于上传下载数据

3. VSFTP 工作模式

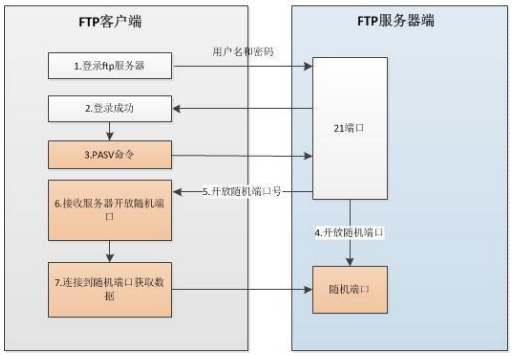
主动模式



Port模式

FTP 客户端首先和服务器的 TCP 21 端口建立连接，用来发送命令，客户端需要接收数据的时候在这个通道上发送 PORT 命令。PORT 命令包含了客户端用什么端口接收数据。在传送数据的时候，服务器端通过自己的 TCP 20 端口连接至客户端的指定端口发送数据。FTP server 必须和客户端建立一个新的连接用来传送数据。

被动模式



Passive 模式

FTP 客户端首先和服务器的 TCP 21 端口建立连接，用来建立控制通道发送命令，但建立连接后客户端发送 Pasv 命令。服务器收到 Pasv 命令后，打开一个临时端口（端口大于 1023 小于 65535）并且通知客户端在这个端口上传送数据的请求，客户端连接 FTP 服务器的临时端口，然后 FTP 服务器将通过这个端口传输数据。

注意：由于VSFTP的被动模式是随机端口进行数据传输，所以在设置防火墙时需要刻意放行。

4. VSFTP 传输模式

Binary模式：不对数据进行任何处理，适合进行可执行文件、压缩文件、图片等
ASCII模式：进行文本传输时，自动适应目标操作系统的结束符，如回车符等

Linux的红帽发行版中VSFTP默认采用的是Binary模式，这样能保证绝大多数文件传输后能正常使用
切换方式：在ftp>提示符下输入ascii即转换到ACSII方式，输入bin，即转换到Binary方式。

5. VSFTP 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

6. 登录验证方式

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）

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------）

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）

7. VSFTP 实验部署

注：先关闭服务器和客户机上的防火墙和 SELinux

7.1. 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
   c. 修改上传目录的权限或所有者，让匿名用户有写入权限
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
```

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）
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

7.2. 本地用户验证实验：

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
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

7.3. 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
   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

```
#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
#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，并制定其用户家目录**

```
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
```

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
```

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。

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。

7.4. 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/vsftpd.crt

   #rsa格式的证书

   rsa_private_key_file=/etc/ssl/certs/vsftpd.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)

FileZilla-FTP（第三方客户端工具）

连接测试时选择：

服务器类型：显式 TLS/SSL

登录类型：　一般或匿名