

Trabalhando com a Base de Dados

+ Podemos usar o nmap da mesma maneira que usávamos fora do metasploit, com a única diferença que agora começa com

`db_nmap`

```
msf6 auxiliary(scanner/portscan/tcp) > back
msf6 > db_nmap -v --open -sV -Pn 172.16.1.7
[*] Nmap: 'Host discovery disabled (-Pn). All addresses
[*] Nmap: Starting Nmap 7.94SVN ( https://nmap.org ) a
[*] Nmap: NSE: Loaded 46 scripts for scanning.
[*] Nmap: Initiating Parallel DNS resolution of 1 host
[*] Nmap: Completed Parallel DNS resolution of 1 host.
[*] Nmap: Initiating SYN Stealth Scan at 23:23
[*] Nmap: Scanning 172.16.1.7 [1000 ports]
[*] Nmap: Discovered open port 111/tcp on 172.16.1.7
[*] Nmap: Discovered open port 8080/tcp on 172.16.1.7
[*] Nmap: Discovered open port 22/tcp on 172.16.1.7
[*] Nmap: Discovered open port 21/tcp on 172.16.1.7
[*] Nmap: Completed SYN Stealth Scan at 23:23, 2.48s e
[*] Nmap: Initiating Service scan at 23:23
[*] Nmap: Scanning 4 services on 172.16.1.7
[*] Nmap: Completed Service scan at 23:23, 10.37s elap
[*] Nmap: NSE: Script scanning 172.16.1.7.
[*] Nmap: Initiating NSE at 23:23
```

+ Podemos também realizar a varredura fora do metasploit e depois importar o resultado pra dentro dele. Basta que durante a realização nmap, salvemos seu output em um arquivo xml

```
nmap -v --open -sV -Pn 172.16.1.4 -oX /opt/host4.xml
```

Agora, no metasploit:

```
db_import /opt/host4.xml
```

+ Para vermos o que já foi salvo de nossas varreduras, executamos

```
services
```

```
msf6 > services
Services
=====
```

host	port	proto	name	state	info
172.16.1.7	21	tcp	ftp	open	ProFTPD 1.3.3a
172.16.1.7	22	tcp	ssh	open	OpenSSH 5.5p1 Debian 6+squeeze5 protocol 2.0
172.16.1.7	111	tcp	rpcbind	open	2 RPC #100000
172.16.1.7	8080	tcp	http	open	Apache httpd 2.2.16 (Debian)

+ Para explorar vulnerabilidades de maneira automática, fazemos

```
vulns
```

→ Claro, os resultados só vão ser mais efetivos conforme formos

enumerando os serviços