LAB - SEM 06 - Mecanismos de Defesa

LAB01: 443

+ Aqui fizemos uma busca pelas 1000 principais portas no nmap e encontramos um arquivo na internet que tivesse todas elas: o nome era "nmap-top-ports.txt"
+ Executamos então uma varredura com o nmap somente com essas portas especificadas no arquivo (A varredura porta à porta da 1 à 65535) demorou demais. Portanto, foi melhor usar as portas comuns

```
for porta in $(cat nmap-top-ports.txt);
do echo "TESTANDO A PORTA $porta";
nmap -v -sS -Pn -g $porta 172.16.1.59| grep "Discovered";
done
```

→ Lembrando que a opção -g especifica a porta de origem

```
TESTANDO A PORTA 443
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan time
s may be slower.
Discovered open port 8080/tcp on 172.16.1.59
Discovered open port 22/tcp on 172.16.1.59
```

+ Como disse, o teste porta a porta também funcionou, mas demorou demais

```
for porta in $(seq 1 65535);
do;
nmap -v -sS -Pn -g $porta 172.16.1.59 | grep "Discovered";
echo "TENTATIVA DA PORTA $porta";
done
```

```
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan time
s may be slower.
Discovered open port 8080/tcp on 172.16.1.59
Discovered open port 22/tcp on 172.16.1.59
```

LAB02: 120537778998

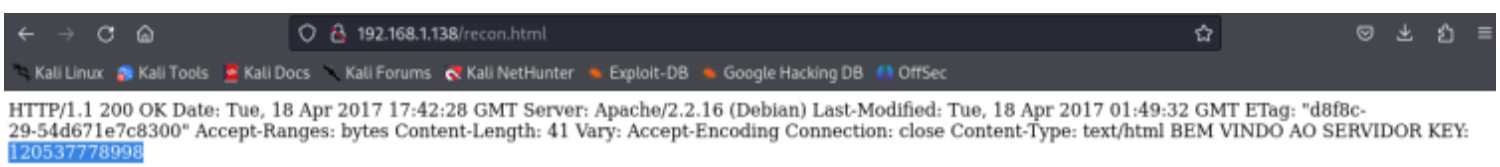
+ Com o resultado do lab passado, fizemos a conexão via netcat enviando uma porta de origem (sim, o netcat tem essa opção), fizemos uma requisição do tipo GET e enviamos para um arquivo xml em /var/www/html/recon.html (recon.html é o nome do arquivo criado)

```
nc -vn -p 443 172.16.1.59 8080 > /var/www/html/recon.html
```

+ Depois, iniciamos nosso serviço do apache

```
service apache2 start
```

+ Por fim, acessamos no navegador o endereço 192.168.1.138/recon.html



LAB03: 80,2222,10000
Como o objetivo é fazer o mínimo de barulho possível, então usar a opção que manda IP's aleatórios como -D RND:20 não é a mais furtiva.
A melhor opção é que façamos o teste com as --top-ports e subamos a qtd pouco a pouco

```
nmap -Pn -sS --top-ports=350 intranet.businesscorp.com.br --open
```

```
┌──(root㉿DESKTOP-NJHHNK6)-[/home/kali]
└─# nmap -Pn -sS --top-ports=350 intranet.businesscorp.com.br --open
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-25 17:50 -03
Nmap scan report for intranet.businesscorp.com.br (37.59.174.228)
Host is up (0.18s latency).
rDNS record for 37.59.174.228: ip228.ip-37-59-174.eu
Not shown: 344 closed tcp ports (reset), 3 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
80/tcp    open  http
2222/tcp  open  EtherNetIP-1
10000/tcp open  snet-sensor-mgmt
```

LAB04: debian4
Quando fazemos a varredura dos serviços com a opção -V do nmap, vemos que o serviço rodando na porta 2222 é o ssh

```
nmap -v -sSV -p 80,2222,10000 intranet.businesscorp.com.br
```

```
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.2.22 ((Debian))
2222/tcp  open  ssh     OpenSSH 6.0p1 Debian 4+deb7u6 (protocol 2.0)
10000/tcp open  http    MiniServ 0.01 (Webmin httpd)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Portanto, se fizermos a interação com o netcat pela porta 2222, vamos capturar o banner do OS:

```
nc -v intranet.businesscorp.com.br 2222
```

De onde recebemos o mesmo banner do acusado pelo nmap e concluimos ser um Debian 4

LAB05: Apache/2.2.22
Basta ver o lab4

LAB06: openssh6.0p1
Basta ver o lab4
o serviço de acesso à distância é o ssh

LAB07: miniserv/0.01
Basta ver o lab4

LAB08: publickey,password

Ao pesquisarmos na internet, achamos a seguinte requisição que se propõe a fazer o que buscamos

```
nmap --script ssh-auth-methods intranet.businesscorp.com.br
```