SSI
(Computer Systems Security)

Penetration Testing

João Azevedo & Paulo Araújo

University of Minho, Braga, PT
{a85227,a85729}@alunos.uminho.pt

Abstract. Este documento tem como principal objetivo a experimentação da fase de *footprinting* presente nos testes de penetração associados a sistemas e/ou redes em pequena ou larga escala. A realização destes testes pressupõe a utilização de um conjunto de ferramentas *open-source* para *scanning* de redes e sistemas com vista a aumentar o reportório de segurança das máquinas envolvidas na prestação de diversos serviços aplicacionais duma dada organização.

Keywords: Systems Security · Penetration Testing · Footprinting · Reconnaissance and Scanning

1 Caso de estudo

Os testes de intrusão (1) - *"Penetration Test"* ou *"pentest"* - são métodos que têm por base ataques simulados a um sistema com vista a descobrir vulnerabilidades que possam ser exploradas. O vetor de ataque pode envolver vários serviços conhecidos como, por exemplo, interfaces de aplicação (APIs), servidores *frontend* e *backend*, servidores de DNS, entre outros.

O planeamento para a realização destes testes pressupõe 5 fases, cíclicas, de reconhecimento do sistema, *scanning*, ganho de acesso, entre outras, no entanto, nesta análise apenas vão ser abordadas as duas primeiras, correspondendo à fase de *footprinting*.

2 Endereços a analisar

A fase de *footprinting* pode ser dividida em duas principais: *passive* (reconhecimento) e *active* (exploração), com vista a obter informações públicas sobre um certo *target*, dentro dos *IPs* apresentados de seguida.

As informações obtidas podem envolver informação de DNS, espionagem ativa, *port scanning* ou até uma simples pesquisa por *websites* de determinadas organizações.

De seguida, em cada secção vamos apresentar um determinado *IP* e todas as informações que conseguimos retirar do mesmo, catalogando cada passo e ferramenta usada.

2.1 IPv4: "137.74.187.100"

- ***host*** - *DNS lookup utility*: A primeira ferramenta que testamos permite-nos fazer operações de *lookup* do *DNS - Domain Name System*, mapeando um dado domínio para um endereço IP ou, como usado neste caso em particular, obter um dado domínio a partir de um endereço IP(v4).
  Assim, o comando *host* pode ser executado da seguinte forma:

  ```
  $ host 137.74.187.100
  100.187.74.137.in-addr.arpa domain name pointer hackthissite.org.
  ```

  Este produz no *stdout* uma linha que nos indica um registo do tipo *PTR* presente no servidor de nomes reverso. Temos então que este IP está mapeado para o domínio *hackthissite.org.*
  Através de uma pequena pesquisa descobrimos o servidor *web www.hackthissite.org* deste domínio que foi feito para treinar, de forma legal e segura, mecanismos e ferramentas de exploração de vulnerabilidades por *hackers* num ambiente aberto e mantido por membros desta comunidade.

- ***whois*** - *Client for the whois directory service*: Este segundo comando é muito completo e tem por base a utilização de um protocolo da pilha *TCP/IP* chamado *WHOIS*(3) para consultar informações de contacto de um sistema, blocos de endereços IP alocados e informações do(s) servidor(es) de DNS desta e de outras entidades na *Internet*.
  De forma geral, o cliente *whois* do nosso sistema *Linux* comunica com o servidor *WHOIS* (porta 43) da organização em questão que mantém uma base de dados com os conteúdos descritos anteriormente, providenciando, entre outras, as seguintes informações:

  ```
  $ whois 137.74.187.100
  ```

  1. Informações do registo regional da *Internet* para a Europa (*RIPE*):

     Esta tabela contém algumas das entradas obtidas com o *whois*, que nos indicam a subrede dedicada para a organização que estamos a explorar (*hackthissite.org.*), vendidas pela organização *RIPE*:

NetRange:	137.74.0.0 - 137.74.255.255
CIDR:	137.74.0.0/16
NetName:	RIPE
Organization:	RIPE Network Coordination Centre (RIPE)
RegDate:	2016-08-29

2. Informações técnicas e de contacto:

   Por outro lado, conseguimos também obter alguma informação administrativa relativa à organização do domínio original, incluindo informações técnicas, administrativas e contactos:

organisation:	ORG-SH80-RIPE
org-name:	Staff HackThisSite
address:	Stadtmitte 1
address:	10117 Berlin
address:	DE
phone:	+49.151011011

– ***nslookup*** - *Interactive client for Internet name servers queries*: Uma forma de estabelecer o perfil deste domínio passa também por obter informações específicas de servidores de nomes questionando o DNS através do *queries* com o *nslookup*:

```
$ nslookup
> hackthissite.org.
```

DNS1:	**137.74.187.100**
DNS2:	137.74.187.101
DNS3:	137.74.187.102
DNS4:	137.74.187.103
DNS5:	137.74.187.104

Isto permite-nos perceber que o endereço IP original corresponde a um (de um total de 5) servidor de DNS que o domínio possui (representado a negrito).

– ***nmap****(2) - Network exploration tool and security / port scanner*: Esta ferramenta permite realizar esta segunda fase do *footprinting* através de um *scanning* ativo de redes grandes. Neste caso, o objetivo é encontrar *hosts* que estejam à escuta na rede, isto é, possíveis serviços aplicacionais e as respetivas portas:

```
# TCP SYN/Connect()/ACK/Window/Mainmon scans
$ nmap -sS hackthissite.org
```

O comando *nmap* mostra-nos que existem vários endereços disponíveis para este domínio, passíveis de ser analisados e, para os quais obtivemos as seguintes informações:

```
#output resumed
Nmap scan report for hackthissite.org (137.74.187.104)
Host is up (0.0073s latency).
Other addresses for hackthissite.org (not scanned):
137.74.187.100 137.74.187.102 (...)
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 5.53 seconds
```

Existem, portanto, duas portas *tcp* abertas para comunicações cujo serviço é *http/https*, o que indica que muito provavelmente esta máquina também serve o *website* principal da organização, o que se confirma executando o comando *host*:

```
$ host www.hackthissite.org
www.hackthissite.org has address 137.74.187.100
# other addresses (...)
```

Numa segunda fase, passamos à identificação do sistema em si, visto que muitos *exploits* são escritos para certas versões de sistemas operativos e, assim, seria interessante perceber se esta informação é detetável.

```
# Enable OS detection (w/ Guess OS more aggressively option)
$ nmap -O hackthissite.org --osscan-guess
```

O resultado da execução mostra-nos, mais uma vez, duas portas com o estado *OPEN* como no comando anterior. No entanto, agora mostra-nos possíveis OS (com as probabilidades de ser ou não):

```
#output resumed
Device type: bridge|general purpose
Running (JUST GUESSING): Oracle Virtualbox (91%), QEMU (86%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu
Aggressive OS guesses: Oracle Virtualbox (91%),
QEMU user mode network gateway (86%) (...)
```

Por fim, seria também interessante perceber que serviços aplicionais estão a correr, incluindo as suas versões e, para isso, temos a opção *-sV*:

```
# Probe open ports to determine service/version info
$ nmap -sV hackthissite.org
```

Assim, obtivemos o seguinte *output*:

```
PORT    STATE SERVICE        VERSION
80/tcp  open  http-proxy     HAProxy http proxy 1.3.1 or later
443/tcp open  ssl/http-proxy HAProxy http proxy 1.3.1 or later
Service Info: Device: load balancer
```

Os servidores encontram-se a utilizar um *software proxy* chamado *HAProxy*(4), versão 1.3.1 ou superior, informação essa que será bastante importante na altura de procurar vulnerabilidades.

2.2 IPv4: "216.58.215.148"

- ***host***: O *output* deste comando indica um registo do tipo PTR presente no servidor de nomes reverso. Sabemos então que este IP está mapeado para o domínio/*host* mad41s04-in-f20.1e100.net. :

  ```
  $ host 216.58.215.148
  148.215.58.216.in-addr.arpa domain name pointer mad41s04-in-f20.1e100.net.
  ```

  O que não nos dá grande informação visto que não existe nenhum website acessível a partir do mesmo, nem qualquer tipo informação que nós possamos usar para explorar.

- ***whois***: Visto que não nos foi possível obter nenhuma informação relevante com o comando anterior sentimos a necessidade de executar o comando *whois* não com o *name server* mas sim com o IP disponibilizado.

  ```
  $ whois 216.58.215.148
  ```

  Obtemos agora dados que nos permitem determinar quem é a organização que detém o domínio deste ip e muitas outras informações que iremos apresentar em seguida.

NetRange:	216.58.192.0 - 216.58.223.255
CIDR:	216.58.192.0/19
NetName:	GOOGLE
NetType:	Direct Allocation
Organization:	Google LLC (GOGL)
RegDate:	2012-01-27

Sabemos que estamos perante serviços da Google que estão hospedados no intervalo de ip's 216.58.192.0 - 216.58.223.255, e que este domínio foi registado a 27 de Janeiro de 2012.

OrgName:	Google LLC
OrgId:	GOGL
Address:	1600 Amphitheatre Parkway
City:	Mountain View
PostalCode:	94043
Country:	US

Já nesta tabela observamos a mesma informação sobre a organização que opera neste IP, e ainda o endereço físico do mesmo, incluindo informações sobre a rua, cidade, código postal e o país.
Através deste comando as organizações podem disponibilizar várias informações sobre os seus diversos departamentos. Relativamente ao nosso caso de estudo a *Google* continha dois departamentos: *OrgTech* e *OrgAbuse*. Deste modo, na tabela seguinte são apresentadas algumas informações pertinentes referentes ao primeiro departamento:

OrgTechHandle:	ZG39-ARIN
OrgTechName:	Google LLC
OrgTechPhone:	+1-650-253-0000
OrgTechEmail:	arin-contact@google.com

Aqui estão disponíveis informações referentes ao departamento de tecnologia da google, onde se pode encontrar dados para encontrar em contacto com o mesmo através de telemóvel e email.

– ***nmap***:

```
$ nmap -O 216.58.215.148
Running (JUST GUESSING): Linux 1.0.X (88%), Cisco embedded (87%),
Nokia Symbian OS (87%), Ouya embedded (85%)
(...)
```

O comando *nmap* utilizado em cima com a flag -O dá como resultado um conjunto de sistemas operativos possíveis de estarem a correr na máquina que está a alojar este IP, e as suas probabilidades. Neste caso o sistema operativo mais provável de estar a correr na máquina é o Linux.

```
$ nmap -sS 216.58.215.148
#output resumed
```

```
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https
```

Já este comando *nmap* com a flag *-sS* permite-nos verificar que portas da máquina, para este IP, estão a comunicar via protocolos TCP. Conseguimos verificar então que as portas 80 e 443 estabelecem comunicações via TCP, com serviços http e https, respetivamente.
De salientar ainda o teste do *nmap* com a flag *-sU* para o varrimento das portas da máquina em busca daquelas que estabelecem comunicação via UDP, que não teve grande efeito visto não terem sido encontradas nenhumas portas no resultado do comando, o que nos depreender que não existem comunicações UDP.

2.3 IPv4: "45.33.32.156"

- ***host***:

```
$ host 45.33.32.156
156.32.33.45.in-addr.arpa domain name pointer scanme.nmap.org.
```

A finalidade do comando *host* foi a mesma aquando da sua utilização nos IP's anteriores a de encontrar um domínio, neste caso foi scanme.nmap.org. Conseguimos aceder desta vez ao website que tem como finalidade disponibilizar aos seus visualizadores uma série de documentação e informação sobre o *nmap* e a sua instalação.

- ***whois***

```
$ whois 45.33.32.156
```

Conseguimos assim verificar o intervalo de IP's que este domínio aloca 45.33.0.0 - 45.33.127.255, e que está alocado num servidor virtual de uma empresa denominada Linode, empresa essa que presta serviços de hospedagem na *cloud*. Dá para perceber também que este domínio foi registado em 20 de Março de 2015:

NetRange:	45.33.0.0 - 45.33.127.255
CIDR:	45.33.0.0/17
NetName:	LINODE-US
NetHandle:	NET-45-33-0-0-1
OriginAS:	AS3595, AS21844, AS6939, AS8001
Organization:	Linode (LINOD)
RegDate:	2015-03-20

A informação da próxima tabela é assim referente à empresa que presta a hospedagem do IP. Com isto é possível observar a sua localização física e ainda a data em que foi feito o seu primeiro registo:

OrgName:	Linode
OrgId:	LINOD
Address:	249 Arch St
City:	Philadelphia
PostalCode:	19106
Country:	US
RegDate:	2008-04-24

Seria ainda possível ver os vários departamentos desta empresa e a forma como é possível entrar em contacto com os mesmos, mas não achamos pertinente incidir novamente sobre esse tema visto não se tratar do nosso objeto de estudo.

– ***nmap***:

Já com este último comando não fomos felizes visto que não conseguimos obter nenhuma resposta por parte do servidor. Visto que o comando faz o varrimento por todas as portas não foi possível encontrar qualquer tipo de informação sobre as mesmas, mesmo tentando várias flags do comando.

3 Conclusão

A realização deste trabalho prático foi crucial para complementar a teoria leccionada nas aulas teóricas, que incidia sobre os testes de penetração mais concretamente na fase de *footprinting*.

Foram portanto apresentadas diversas ferramentas as quais ficamos familiarizados e demos a conhecer no decorrer deste trabalho prático. O nosso grupo adotou estas e outras ferramentas, como por exemplo o *Nessus*, na realização das várias análises, no entanto, esta última não produziu resultados que fossem novos em comparação com o *nmap*, por exemplo.

Concluimos então que este relatório representa uma possível documentação da fase de *footprinting* para o trabalho proposto e será também uma preparação para as próximas fases do processo cíclico que temos vindo a estudar.

References

[1] Learning Center. 2020. What Is Penetration Testing — Step-By-Step Process Methods — Imperva. [online] Available at: https://www.imperva.com/learn/application-security/penetration-testing/.

[2] HackerTarget.com. 2020. Nmap Tutorial: From The Basics To Advanced Tips. [online] Available at: https://hackertarget.com/nmap-tutorial/.

[3] Pt.wikipedia.org. 2020. WHOIS. [online] Available at: https://pt.wikipedia.org/wiki/WHOIS.

[4] En.wikipedia.org. 2020. Haproxy. [online] Available at: https://en.wikipedia.org/wiki/HAProxy.