



ATTAQUE D'UN SERVEUR PRISE D'EMPREINTE

[Floux]

L'attaque d'un serveur à «l'aveuglette» à toute ses chances de rater. La prise d'empreinte (ou pentest pour les intimes ;)) permet au pirate de mieux connaître sa victime et ainsi de préparer au mieux son attaque, en se renseignant sur les failles potentielles du serveur par exemple. Pour cela, une poignée d'outils, un peu d'imagination et Internet feront l'affaire...

LES BASES

Tout d'abord, un petit rappel sur les IPs.

Nos ordinateurs utilisent l'«Internet Protocole», d'où IP, pour communiquer entre eux. Pour cela, une adresse IP leur est attribuée et celle-ci sert ensuite de «numéro de téléphone» pour le poste. Lorsque vous rentrez une URL (http://www.google.fr par exemple) vous allez contacter un serveur DNS qui va vous renvoyer l'IP d'un des serveurs de Google afin de pouvoir s'y connecter.

L'IP change normalement à chaque connexion à moins que vous n'en ayez une fixe. Elle est composée de 4 octets (allant de 0 à 255) et s'écrit sous cette forme x.x.x.x (ex: 192.168.0.1).

C'est bien beau d'avoir l'URL d'un site, mais si on veut maintenant commencer notre prise d'empreinte, il sera plus pratique d'avoir l'IP du serveur, sachant que certains outils n'accepte pas l'URL en argument. Pour cela nous pouvons utiliser la commande «ping».

Celle-ci enverra des requêtes au serveur pour vérifier qu'il est bien joignable. Le résultat de cette commande nous donnera l'adresse IP du serveur et entre autre le temps de réponse de celui-ci.

```
floux@floux-laptop:~$ ping www.google.fr
PING www.l.google.com (66.249.91.104): 56 bytes of data:
64 bytes from www.l.google.com: icmp_seq=1 ttl=238 time=83.4 ms
64 bytes from www.l.google.com: icmp_seq=2 ttl=238 time=79.8 ms
64 bytes from www.l.google.com: icmp_seq=3 ttl=238 time=82.6 ms
64 bytes from www.l.google.com: icmp_seq=4 ttl=238 time=79.7 ms
64 bytes from www.l.google.com: icmp_seq=5 ttl=238 time=72.7 ms
64 bytes from www.l.google.com: icmp_seq=6 ttl=238 time=80.7 ms
64 bytes from www.l.google.com: icmp_seq=7 ttl=238 time=79.7 ms
64 bytes from www.l.google.com: icmp_seq=8 ttl=238 time=72.7 ms
64 bytes from www.l.google.com: icmp_seq=9 ttl=238 time=79.7 ms
64 bytes from www.l.google.com: icmp_seq=10 ttl=238 time=79.7 ms
64 bytes from www.l.google.com: icmp_seq=11 ttl=238 time=80.7 ms
64 bytes from www.l.google.com: icmp_seq=12 ttl=238 time=76.7 ms

--- www.l.google.com ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 1201ms
rtt min/avg/mx = 72.7/77.8/103.4 ms
```

Une petite «astuce» qui peu s'avérer très utile pour tout ceux qui utilisent un système basé sur Unix (Linux, Mac OS, FreeBSD...), n'oubliez pas que vous disposez de la commande «man» qui vous permettra d'afficher le manuel de la commande passée en argument et donc de pouvoir ainsi découvrir des options plus ou moins utiles selon l'utilisation que vous en ferez.

Cela offrira un plus grand panel de commande avec toutes les explications qui vont avec, par rapport à un «-h ou --help». RTFM quoi! =)

Une dernière chose, vous cherchez des informations sur un logiciel, une faille ou je ne sais quoi encore, rappelez-vous que Google est votre ami est que grâce à lui vous pourrez trouver bien plus que ce que vous cherchez (faut-il savoir chercher...).

Toutes vos questions ont de grandes chances d'avoir déjà été posées un grand nombre de fois sur des forums, alors demandez l'aide de votre ami Google qui devrait vous trouver ça en moins de 2s ;-)

TRACE TA ROUTE !

Lorsque notre ordinateur communique avec un serveur distant, la liaison n'est pas direct. Nos paquets passent par plusieurs noeuds (serveurs, routeurs...), il est donc intéressant de voir la route empruntée par la communication.

L'utilisation des commandes «traceroute» et «tracert», respectivement pour Linux et Windows, permettent de réaliser cette tâche. A noter qu'il se peut que «traceroute» ne soit pas installé par défaut sous certaines distribution, un «apt-get install traceroute» permettra de l'installer sous les distributions basées sur Debian. Cette commande peut notamment être intéressante pour comprendre la structure d'un réseau local. Si le hacker réussit à pénétrer dans le réseau interne de l'entreprise, il peut ainsi avoir une «vue» de celui-ci.

```
floux@floux-laptop:~$ traceroute www.google.fr
traceroute to www.google.fr (66.249.91.104): 30 hops max, 40 byte packets
 1  flouxquas-tour-mishone.net (192.168.0.1)  0.328 ms  0.182 ms  0.698 ms
 2  Atlantex-258-1-2-1-w90-49.abn.wanadoo.fr (88.49.25.1)  47.686 ms  46.986 ms  58.929 ms
 3  10.125.229.74 (10.125.229.74)  58.938 ms  58.923 ms  58.984 ms
 4  103.253.92.98 (103.253.92.98)  78.798 ms  78.818 ms  78.797 ms
 5  ge-2-1-0-0.nirman201.Nantes.francetelecom.net (193.252.99.198)  78.779 ms  78.762 ms  82.662 ms
 6  81.253.131.69 (81.253.131.69)  94.606 ms  93.834 ms  93.811 ms
 7  francetelecom-level3-10ge.Paris1.Level3.net (4.68.111.254)  93.802 ms  59.566 ms  59.550 ms
 8  * te-2-4.car2.Paris1.Level3.net (4.68.111.203)  59.576 ms *
 9  ae-31-53.ebr1.Paris1.Level3.net (4.68.188.98)  71.548 ms  71.537 ms  83.491 ms
10  ae-1-100.ebr2.Paris1.Level3.net (4.69.185.88)  83.418 ms  83.413 ms  83.394 ms
11  ae-2.ebr1.Frankfurt1.Level3.net (4.80.132.382)  95.348 ms  107.337 ms  107.328 ms
12  ae-1-55.edge1.Frankfurt1.Level3.net (4.80.118.146)  95.284 ms  ae-1-53.edge1.Frankfurt1.Level3.net (4.80.118.82)  107.192 ms  ae-1-51.edge1.Frankfurt1.Level3.net (4.80.118.181)  107.149 ms
13  62.67.33.114 (62.67.33.114)  107.158 ms  78.797 ms  71.826 ms
14  209.85.249.180 (209.85.249.180)  71.891 ms  209.85.249.178 (209.85.249.178)  71.795 ms  209.85.249.180 (209.85.249.180)  71.775 ms
15  72.14.232.288 (72.14.232.288)  83.732 ms  209.85.248.182 (209.85.248.182)  83.721 ms  71.867 ms
16  64.233.175.246 (64.233.175.246)  83.784 ms  209.85.248.79 (209.85.248.79)  83.781 ms  83.698 ms
17  72.14.233.83 (72.14.233.83)  83.678 ms  83.682 ms  95.594 ms
18  66.249.94.146 (66.249.94.146)  83.850 ms  98.956 ms  72.14.233.79 (72.14.233.79)  95.903 ms
19  1x-in-f104.google.com (66.249.91.104)  83.823 ms  83.942 ms  84.060 ms
```

LISTER LES MACHINES D'UNE ENTREPRISE

Nous avons vu que le rôle d'un serveur DNS était de nous donner l'IP d'un serveur à partir d'une URL. Si il est mal configuré, un service peu nous lister l'ensemble des machines du réseau interne de l'entreprise. Ceci est très utile pour un hacker, donc extrêmement dangereux pour l'entreprise.

Cela est donc réalisable par la commande «nslookup»(normalement présente par défaut sous Windows et Linux), et «dig» sous Linux seulement. Il faut savoir que la commande «nslookup» n'est plus mise-à-jour sous Linux, il est donc préférable d'utiliser «dig», par contre, pas de problème avec la première sous Windows.

```
> floux@floux-laptop:~$ nslookup
> google.fr
Server:          192.168.0.1
Address:         192.168.0.1#53

Non-authoritative answer:
Name:   google.fr
Address: 72.14.221.104
Name:   google.fr
Address: 66.249.93.104
Name:   google.fr
Address: 216.239.59.104
>
```

```
; <<> DiG 9.4.2 <<> google.fr
;; global options: printcmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 51349
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;google.fr.                IN      A

;; ANSWER SECTION:
google.fr.                1444    IN      A      66.249.93.104
google.fr.                1444    IN      A      216.239.59.104
google.fr.                1444    IN      A      72.14.221.104

;; Query time: 72 msec
;; SERVER: 192.168.0.1#53(192.168.0.1)
;; WHEN: Sat May 10 17:57:45 2008
;; MSG SIZE rcvd: 75

floux@floux-laptop:~$
```

Pour notre recherche un peu plus loin, on peut tenter de faire un transfert de zone. Si cela réussit, le serveur DNS nous renverra la liste des serveurs associé au domaine.

```
floux@floux-laptop:~$ dig axfr @floux.com @floux.com
; <<> DiG 9.4.2 <<> axfr @floux.com @floux.com
;; global options: printcmd
floux.com.                3600    IN      SOA     floux.com. hostmaster.floux.com. 2006030100 10800 3600 60
4800 3600
floux.com.                3600    IN      NS      floux.com.
floux.com.                3600    IN      NS      floux.com.
floux.com.                3600    IN      MX      10 messenger.floux.com.
www.floux.com.            3600    IN      A      217.147.147.147
preview.floux.com.        3600    IN      A      217.147.147.147
floux.com.                3600    IN      SOA     floux.com. hostmaster.floux.com. 2006030100 10800 3600 60
4800 3600
;; Query time: 276 msec
;; SERVER: 217.147.147.147(217.147.147.147)
;; WHEN: Sat May 10 18:03:12 2008
;; XFR size: 7 records (messages 3, bytes 345)

floux@floux-laptop:~$
```

SCAN DE PORTS

Maintenant que nous avons une liste de serveurs en main, nous allons pouvoir procéder à un scan de ports. L'objectif est ici d'établir une liste des ports ouverts ou non et de regarder les services qui tournent derrière. Cela nous permettra de rechercher des vulnérabilités exploitables via ces services.

```
flou@flou-laptop:~$ sudo nmap -sS -i 192.168.136.126
Starting Nmap 4.53 [ http://nmap.org ] at 2008-05-14 15:36 CDT
SCRIPT ENGINE: ipinfo.nse is not a file.
SCRIPT ENGINE: Aborting script scan.
Interesting ports on 192.168.136.126:
Not shown: 1700 closed ports
PORT      STATE SERVICE
53/tcp    open  dns
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ss
445/tcp   open  microsoft-ss
445/tcp   open  ipatp
445/tcp   open  msrpc
445/tcp   open  msrpc
445/tcp   open  msrpc
1026/tcp  open  strpc
1027/tcp  open  msrpc
1043/tcp  open  strpc
1258/tcp  open  ldap
1259/tcp  open  tcpwrapped
MAC Address: 00:0C:29:00:13:01 (VMware)
Device type: general purpose
Running: Microsoft Windows 2003
OS details: Microsoft Windows Server 2003 SP2 or SP3
Network Distance: 1 hop
Service Info: OS: Windows

OS and Service detection performed. Please report any incorrect results at http://nmap.org/bin/submit/
Nmap done: 1 IP address (1 host up) scanned in 50.630 seconds
flou@flou-laptop:~$
```

Pour cette étape, j'ai choisi nmap, qui est bien connu (on le retrouve même dans Matrix ;-), mais pas forcément le meilleur !

Ainsi la commande:
`nmap -sS -A x.x.x.x`

va scanner l'hôte x.x.x.x avec la méthode SYN SCAN (-sS), on n'aura donc pas de log sur la machine distante. L'option -A va permettre de détecter la version des services tournant sur les ports du serveur.

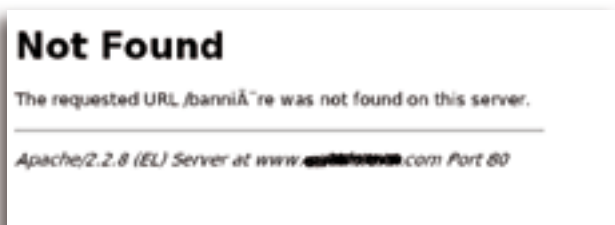
FINGERPRINTING

Nous venons de faire un scan de ports, il me semble donc logique d'enchaîner directement sur le fingerprinting.

Cette étape consiste à relever la version de l'OS (Operating System) tournant sur le serveur. On appelle cela aussi, relevé de bannière.

Cette étape ne peut être sautée du fait que les attaques ne seront pas les mêmes d'un OS à un autre, normal =). Déjà, avec l'argument -A passé dans nmap, on peut voir que l'OS a été récupéré (voir Illustration 6). Une autre commande permettant de réaliser cette tâche est l'option -O.

Une autre manière, est de demander une page inexistante sur le serveur, dans le but que celui-ci nous renvoie un message d'erreur, et là, avec un peu de chance, on pourra en savoir un peu plus sur lui.



Ici, on peut voir que le serveur fait tourner Apache 2.2.8, à partir de là on peut essayer de rechercher des failles sur des sites www.milw0rm.com.

De plus, il y a le très bon p0f, qui lui fait du fingerprinting passif. C'est-à-dire qu'il lira les paquets du réseau et à partir de ceux là, il en déduira l'OS correspondant à l'IP du paquet.

Cette technique est donc totalement furtive, l'auditeur ne pourra pas être détecté.

UNE BALADE SUR LE SITE PAR UNE CHAUDE SOIRÉE D'ÉTÉ

Après avoir exploité les failles techniques, nous allons attaquer les failles humaines. L'entreprise cible et son personnel vont donc être notre première cible pour la récupération d'informations.

Tout d'abord une petite visite sur le site de l'entreprise peut être un très bon début.

Souvent dans les rubriques « contacts », « notre groupe » ou encore « offres d'emploi », nous pouvons avoir le nom, prénom de personnes qui sont employées dans l'entreprise, ainsi nous allons pouvoir trouver les adresses mails, les numéros de téléphones des personnes susceptibles de nous délivrer un maximum d'informations sur les systèmes.

De nos jours il existe de plus en plus de sites communautaires qui détiennent des informations personnelles sur ses membres (ex : Facebook).

Ainsi en connaissant par cœur notre cible, nous pourrions plus facilement la mettre en confiance et lui soutirer des éléments cruciaux pour notre attaque.

WHOIS (T'ES QUI TOI ?)

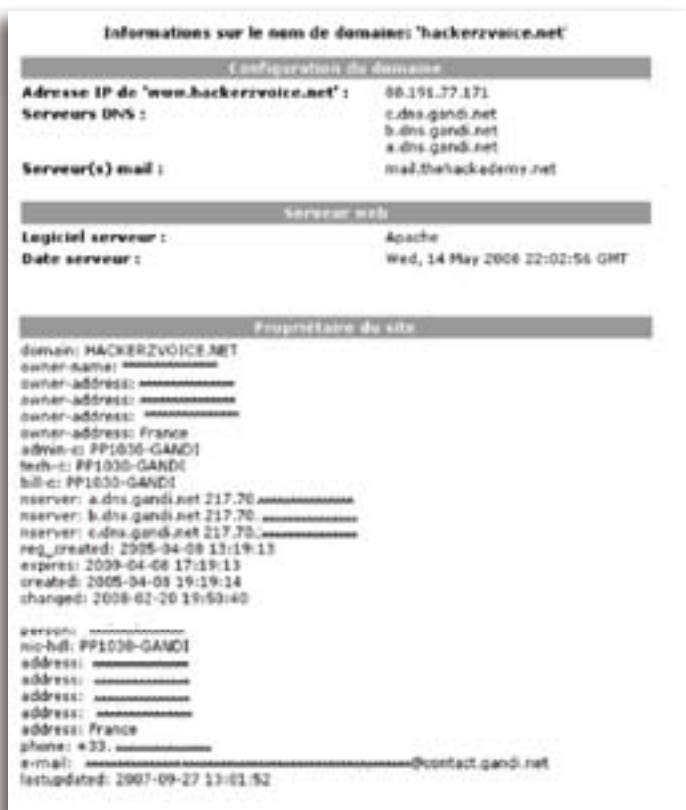
La notion de « whois » est également très importante dans la prise d'empreintes.

Cela permet de donner des indications pertinentes sur le serveur hébergeant le site internet de l'entreprise cible.

En effet lors de l'acquisition d'un serveur web, celui-ci est référencé avec les informations de son propriétaire à contacter en cas de litiges, l'endroit où il est implémenté physiquement, etc...

Il existe plusieurs types de fonction «whois» :

La première est la fonction Web. Par exemple sur le site <http://www.raynette.fr/services/whois/>



La seconde est la fonction sous un terminal Linux grace à la commande : « whois »



Il existe beaucoup d'autres solutions notamment la création de script avec des fonctions propres au langage utilisé.

Ce qu'il faut retenir de cette expérience, c'est que nous pouvons avoir le nom, le prénom, l'adresse postale, l'adresse mail et le numéro de téléphone du propriétaire du serveur.

Avec un peu de chance, nous pouvons avoir beaucoup de renseignement sur le directeur de l'entreprise. Ceci étant notre attaque va pouvoir cibler une personne. Voyons par la suite ce que nous pouvons avoir comme informations supplémentaires.

LE SOCIAL ENGINEERING OU L'ART DE TROMPER LES GENS

Après avoir récupérer assez d'informations sur l'entreprise, et sur son personnel en particulier, nous allons pouvoir tester notre aptitude à tromper les gens. La technique du social engineering demande beaucoup de préparation, aucun paramètre ne doit être laissé pour compte.

Il faut ainsi créer son propre scénario, envisager le maximum de situations, et toujours garder son sang froid.

Tout d'abord nous allons prendre un premier contact avec la personne susceptible d'être faillible pour son entreprise. Le but étant de la mettre en confiance pour qu'elle nous divulgue le maximum d'informations à son propre insu.

Par exemple, nous recevons de nos jours de plus en plus d'appels téléphoniques pour des sondages, ou pour connaître le type de parpaing que nous avons utilisé pour construire notre maison.

Il va être d'autant plus facile de s'appuyer sur ce système pour soutirer des informations cruciales (entreprises partenaires, marque des serveurs, ...).

ET POUR RÉSUMER ?

Ce qui est important de toujours garder à l'esprit est le fait qu'une entreprise n'est pas totalement infaillible.

Il est toujours possible de trouver la petite brèche que ce soit au niveau technique que niveau humain.

Lors d'un pentest il est donc crucial de vérifier chaque entité susceptible d'être en contact avec l'information que l'entreprise traite (allant de la secrétaire, au sys-admin, en passant par la femme de ménage, et en revenant par les mainframes traitent les données).

Plus l'entreprise est consciente des dangers qu'elle court en s'exposant sur le net, et à la face du monde, plus l'attaque sera mise à l'épreuve et plus elle aura de chances d'échouer.