

Techniques de Hacking – N1

SEC-HACK

m2information.fr



TECHNIQUES DE HACKING – NIVEAU 1

SURFACE D'ATTAQUE ET VULNÉRABILITÉS

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

- Surface d'attaque et défense en profondeur
- Standard de classification des vulnérabilités
- Framework ATT&CK
- Collecte d'information passive
- Collecte d'information active

- Surface d'attaque

Lorsque l'on s'intéresse à la sécurité d'une machine ou d'un système d'information de manière générale, il est nécessaire de commencer par définir son niveau d'exposition à la menace.

C'est en premier lieu, ce niveau d'exposition qui va nous permettre de définir la surface attaquable de notre machine.

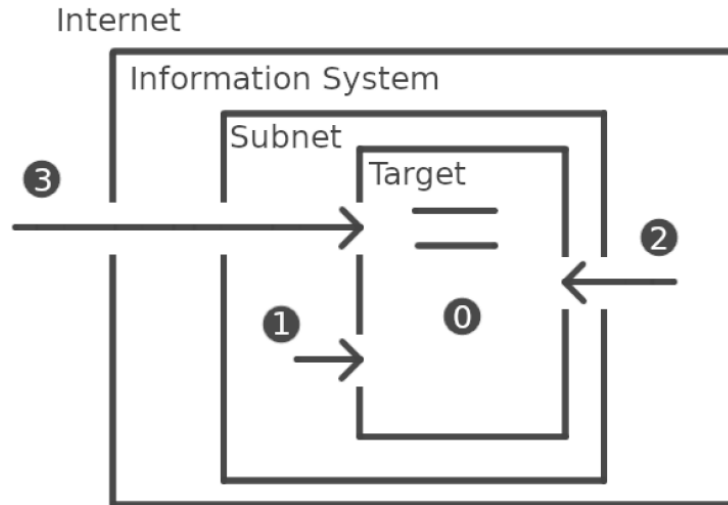
En fonction l'intégration ou non-intégration de l'attaquant au sein du système support de la machine attaquée, les possibilités d'exploitation et surtout de découverte vont potentiellement être très limitées.

Il va donc exister plusieurs niveau de qualification de cette surface d'attaque.

Techniques de Hacking Niveau 1 – Surface d’attaque et Vulnérabilités

Cette capacité à l’exploitation potentielle ou surface d’attaque sera donc qualifiée d’un point de vue périmétrique (chaque périmètre dépendant du niveau de proximité possible entre l’attaquant et la cible).

Le périmètre va de l’exposition maximale possible pour un système (internet) jusqu’à de l’exploitation potentielle directement en local sur la machine (élévation de privilège, ...



La couche « 0 » de notre surface d'attaque est la seule ne nécessitant pas de mise en réseau de notre machine.

Le besoin de sécurité à ce niveau la réside principalement dans la protection de la machine contre tous les moyens pouvant être exploités par un attaquant pour opérer de l'élévation de privilège.

Ensuite, pour des raisons de réponse incidente, il va être nécessaire au niveau de la machine de mettre en œuvre tous les mécanismes nécessaires à l'imputabilité et à la traçabilité des activités utilisateurs.

En effet, il est nécessaire d'être en mesure de posséder tous les éléments nécessaires à l'étude de l'attaque et des failles exploitées si ces événements devaient se produire.

Pour toutes les couches au dessus, elles impliquent des interactions avec la machine cible par le réseau, ce qui implique que le durcissement le plus basique qui peut y être opéré passe par du filtrage de flux.

Ce filtrage peut aussi bien s'opérer au niveau de la machine qu'au niveau des équipements réseau :

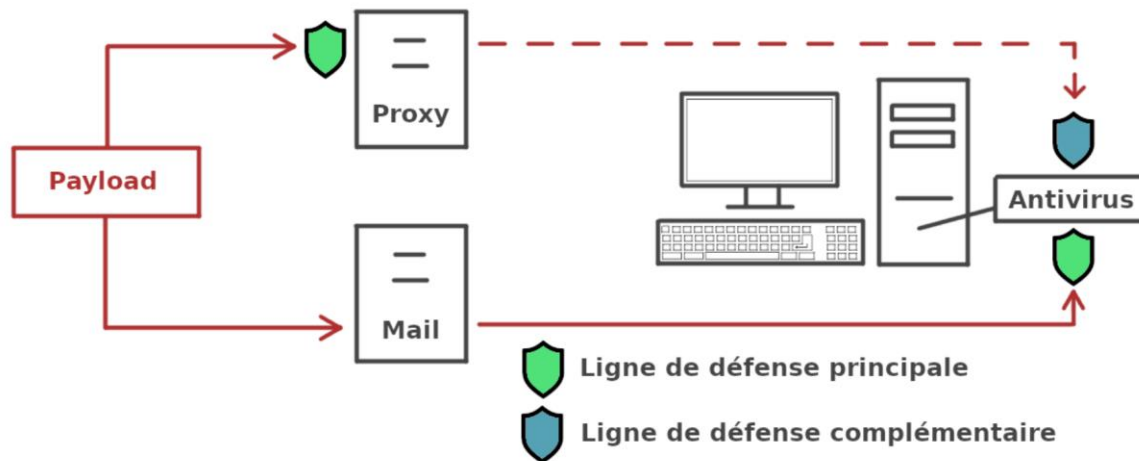
- Pare-feu local (Iptables, ...)
- Network Access List (NACL)
- Security Group (SG)

Mais aussi au niveau d'un filtrage applicatif (White listing au niveau du reverse proxy, ...)

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Du point de vue du pentest, il s'agit de venir déconstruire le concept de défense en profondeur pour chercher à causer des incidents pour par la suite les prévenir.

« La démarche de défense en profondeur consiste à déterminer les barrières à mettre en place en fonction des menaces et des biens à protéger puis à déterminer le niveau de gravité des incidents provoqués par le franchissement des barrières et ainsi tracer une ligne de défense composée des différentes barrières nécessaires » - ANSSI



- Gestion des vulnérabilités

Le but de toutes nos recherches (actives comme passives) est la découverte de vulnérabilités à des fins d'exploitation.

On appelle vulnérabilité, une faiblesse dans le schéma de fonctionnement d'un système ou d'une application.

Ces vulnérabilités peuvent être exploitées par un attaquant pour parvenir à opérer des actions initialement interdites, non prévues ou à accéder à des informations normalement interdites.

Il y a beaucoup de facteurs au sein d'un système pouvant causer une vulnérabilité au sein de celui-ci. On considère les grandes catégories de vulnérabilités suivantes :

- Système d'exploitation (failles au plus bas niveau, élévation de privilège)
- Authentification (comptes par défaut, mots de passes faibles, ...)
- Configuration (configuration permettant la divulgation d'information, ...)
- Logique applicative (mécanismes applicatifs permettant l'exploitation de ceux-ci)
- Facteur humain (hors système informatique, phishing, social engineering, ...)

La classification des vulnérabilités joue un rôle vital dans le schéma de gestion d'une chaîne de sécurité et un rôle tout aussi important dans un schéma d'attaque.

C'est ce qui va nous servir à déterminer vers quel service se tourner pour procéder à une exploitation réussie.

Le système de notation le plus couramment utilisé est le CVSS (Common Vulnerability Scoring System).

Ce système apporte des informations facilement lisible incluant :

- La facilité d'exploitation de la vulnérabilité par rapport aux ressources nécessaires
- L'existence d'un exploit public
- L'impact réel sur la Confidentialité, l'Intégrité et la Disponibilité

Il existe des vulnérabilités pour tous les types de systèmes (système d'exploitation, application, service, ...), il est donc nécessaire de se baser sur des flux de données fiables et complets pour traquer les vulnérabilités des installations attaquées.

Il existe plusieurs bases de données de recensement de ces vulnérabilités :

- Exploit-DB (Offensive Security)
- NVD (National Vulnerability Database) (NIST)

L'objectif de ces bases est de fournir un flux d'information fiable permettant la prise de décision avec une complétion d'analyse très basique.

La base du NIST (NVD) est plus basée sur un modèle de listing des vulnérabilités, sous la forme de CVE (Common Vulnerabilities and Exposures)

Ces CVE contiennent toutes les vulnérabilités confirmées (~2000 par mois), il s'agit d'une base très fournie mais pas le plus adéquat pour opérer des scénarios opérationnels d'attaque.

Exploit-DB va être beaucoup plus intéressant pour du pentest. La base se focalise sur la rétention de tous les exploits existants.

On va pouvoir également y trouver des PoC (bouts de code faits pour démontrer l'exploitabilité d'une vulnérabilité).

Note : Lorsqu'une nouvelle vulnérabilité présentant une exploitation intéressante, il est également très facile de trouver des PoC sur Github avant qu'ils ne soient déposés sur Exploit-DB.

Techniques de Hacking Niveau 1 – Surface d’attaque et Vulnérabilités

Au travers d’Exploit-DB, on pourra facilement partir à la recherche d’un exploit en fonction des informations que l’on aura pu récupérer au niveau de nos systèmes.

Note : Cette recherche aurait été impossible sans la version du service. Il aurait fallu tenter tout ce qui existe ou faire des suppositions pseudo aléatoires.





Exploit Database Advanced Search

PORT	STATE	SERVICE	VERSION
21/tcp	open	ftp	vsftpd 2.3.4

Title

vsftpd 2.3.4

Date  D A V Title

2021-04-12			vsftpd 2.3.4 - Backdoor Command Execution
2011-07-05			vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)

Plutôt que de rechercher à la main des informations sur l'ensemble de ces vulnérabilités, et aux vues de la quantité de vulnérabilités existantes, nous allons privilégier l'automatisation à la recherche manuelle.

Pour ce faire, nous allons notamment pouvoir nous baser sur des systèmes automatisés (script NMAP, ...) qui vont venir s'indexer sur ces bases de vulnérabilité à la recherche d'une possible exploitation.

Dans le cas où nous ne serions pas en mesure de récupérer d'information utile (un système durci n'affichant pas sa bannière par exemple), il restera nécessaire de venir établir des suppositions quant à l'état de l'art d'implémentation de chaque service, de faire des suppositions techniques sur les versions et de chercher manuellement

Pour récupérer un rendu plus « professionnel » dans la recherche et la gestion de nos vulnérabilités (et surtout sur l'évolution de celles-ci), on se reposera sur des solutions graphiques comme Nessus.

Nessus est une solution Tenable, il s'agit d'un outil pouvant opérer des scans de vulnérabilités automatisés (que ça soit par le réseau, ou localement sur une machine en lui fournissant des credentials pour s'y connecter).



Note : Pour un outil open source => OpenVAS

Nessus va être un bon point de départ pour tout pentest ou tout besoin de gestion de vulnérabilité au sein d'un système d'information.

Les outils de ce type vont venir à la fois trouver les machines sur un segment réseau donné mais également les services / versions présents ainsi que les vulnérabilités exploitables ou non.

L'outil se base sur les même flux de données que nos différents scripts NMAP, il s'agit simplement d'intégrer une solution pleinement automatisable et fournissant des rapports « prêts à l'emploi », sans avoir à traiter avec des informations brutes.

Il existe également des modules dédiés à la recherche de vulnérabilités spécifiques et même auditer la compliance (PCI DSS, ...).

- Framework MITRE ATT&CK

Au-delà de la détection des vulnérabilités, il est nécessaire de classifier les différentes techniques et tactiques d’attaques pour mieux appréhender les approches des attaquants.

Pour compléter notre KillChain basique, nous allons être en mesure de traquer toutes les étapes intermédiaires au travers du Framework MITRE ATT&CK.

Il s’agit d’une base de connaissance générale sur l’ensemble des techniques d’attaques connues et exploitées par les attaquants (le modèle de menace associé, les contres mesures à mettre en place, la manière de détecter, ...).

Techniques de Hacking Niveau 1 – Surface d’attaque et Vulnérabilités

Le Framework identifie 14 tactiques d’attaques et pas loin de 202 techniques (435 sous techniques) qui leur sont associées.

Illustration : Matrice de base du Framework

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery
10 techniques	8 techniques	10 techniques	14 techniques	20 techniques	14 techniques	43 techniques	17 techniques	32 techniques
Active Scanning (3)	Acquire Access	Content Injection	Cloud Administration Command	Account Manipulation (6)	Abuse Elevation Control Mechanism (6)	Abuse Elevation Control Mechanism (6)	Adversary-in-the-Middle (3)	Account Discovery (4)
Gather Victim Host Information (4)	Acquire Infrastructure (3)	Drive-by Compromise	Command and Scripting Interpreter (10)	BITS Jobs	Access Token Manipulation (5)	Access Token Manipulation (5)	Brute Force (4)	Application Window Discovery
Gather Victim Identity Information	Compromise Accounts (3)	Exploit Public-Facing Application	Container Administration	Boot or Logon Autostart Execution (14)	Account Manipulation	BITS Jobs	Credentials from Password Stores (6)	Browser Information Discovery
	Compromise Infrastructure							

L’objectif est d’avoir une vision complète des capacités d’un attaquant quelque soit le contexte dans lequel il se trouve. Dans l’idéal, il faudrait être en mesure de proposer une contre mesure pour toute technique applicable à nos environnements.

- Reconnaissance

La reconnaissance peut être définie comme le simple fait de récupérer un maximum d'informations sur la cible (sur nous même dans un schéma de défense).

On fait la distinction entre la reconnaissance passive et la reconnaissance active. La reconnaissance passive va consister à récupérer les informations « publiques »

Travailler avec ce type d'information va permettre de ne pas alerter la cible en n'opérant pas de trafic direct vers celle-ci. L'objectif va donc être de :

- Traquer les enregistrements DNS disponibles
- Glaner des informations sur le personnel (LinkedIn, ...)
- Actualité de l'entreprise (Médias, ...)

Il s'agit principalement de qualifier et de contextualiser la cible.

La contextualisation va d'ailleurs être le point central de notre activité. En effet, si nous n'avons pas suffisamment de renseignement sur le contexte dans lequel s'inscrit le service ou la machine qui est ciblée, nous allons passer à côté de la grande majorité des capacités d'attaques.

Sans contexte, il n'est pas non plus possible de comprendre l'étendu de l'impact des éléments que nous avons trouvé.

C'est le contexte qui fera la distinction entre les capacités de découvertes de vulnérabilités entre un système automatisé (Nessus, ...) et notre test de pénétration.

Note : La reconnaissance active participe également à la contextualisation.

La reconnaissance active de son côté consiste à aller chercher l'information directement où elle se trouve. Par exemple, se connecter directement aux serveurs (HTTP, FTP, SMTP, ...) sur la base d'un scan du matériel réseau exposé (matériel potentiellement découvert au préalable au travers d'une étape de reconnaissance passive).

L'engagement d'une reconnaissance active peut également signifier tenter de s'introduire physiquement dans l'entreprise afin de collecter de l'information là où elle est directement stockée (mais aussi par social engineering en contactant directement la société ou ses salariés).

La reconnaissance active est invasive et surtout traçable. Ce qui peut à la fois poser des problématiques légales et de découverte de l'attaque en cours.

Un simple nom de domaine peut suffire pour récupérer un grand nombre d'informations sur sa cible.

Les registrars (instances de gestion qui délivre les noms de domaine) sont responsables de maintenir à jour des bases WHOIS. Cette information WHOIS contient toutes les données en relation avec le nom de domaine enregistré.

Par exemple :

- Informations de contact : Nom, adresse mail, numéro de téléphone, ...
- Informations temporelles : Dates d'enregistrement, d'expiration, ...
- Serveur de nom : Le serveur auprès duquel les résolutions de noms sont opérées

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Le système DNS est un excellent point de départ pour la reconnaissance et la recherche de cibles potentielles. En effet, de manière classique, on effectue des requêtes DNS afin de traduire un nom de domaine vers une adresse IP.

Cependant, il peut nous apporter beaucoup plus que ça. Il existe plus d'un type d'enregistrement DNS pour les différents usages qu'il est possible, notamment la création d'enregistrements dédiés aux serveurs de mails.

Type d'Enregistrement	Donnée
A	IPv4 Address
AAAA	IPv6 Address
CNAME	Alias
MX	Serveur de mail
TXT	Texte

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Sur la base d'un simple client DNS, nous allons donc être en mesure de trouver de nouvelles adresses IP et donc de nouveaux points d'entrée potentielle (toujours dans une démarche de qualification de la surface d'attaque).

Le serveur de mail est un bon point de départ mais il est également possible de simplement opérer un transfert de zone ou même brute force les enregistrements existants (la plupart des enregistrements créés sont basés sur des noms génériques).

```
(kali@kali)~$ dig lessonsharing.fr MX

; <<> DiG 9.18.0-2-Debian <<> lessonsharing.fr MX
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 7166
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:;, udp: 512
;; QUESTION SECTION:
;lessonsharing.fr.                IN      MX

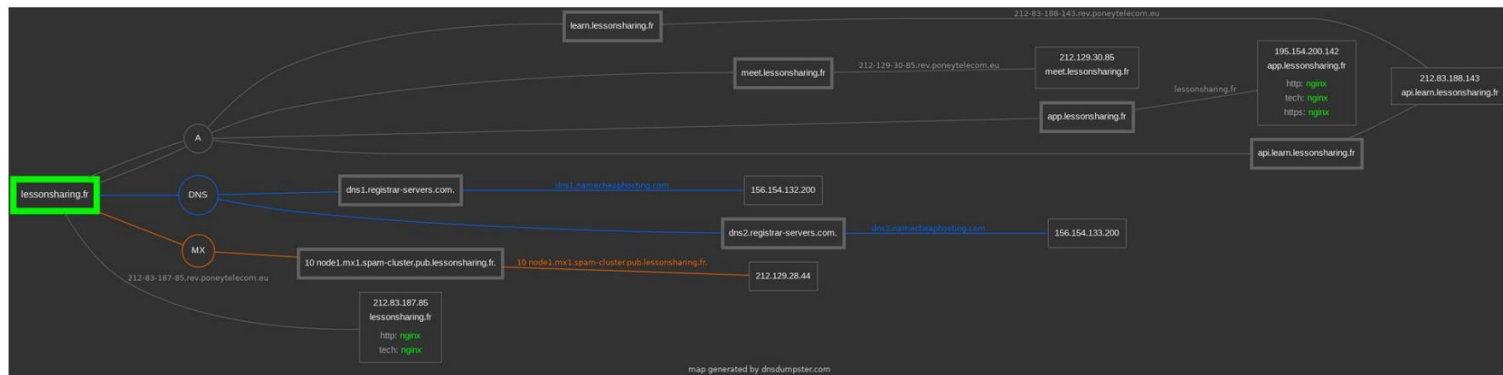
;; ANSWER SECTION:
lessonsharing.fr.                170     IN      MX      10 node1.mx1.spam-cluster.pub.lessonsharing.fr.

;; Query time: 8 msec
```

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Les outils de résolution de noms à notre disposition (nslookup, dig, ...) ne sont pas en mesure de trouver de sous domaines par eux même. Ces sous-domaines font pourtant partie de la surface d'attaque qui nous intéresse.

Passer par une méthode de force brute peut être long et non pertinent. Heureusement, il existe des outils comme DNSDumpster pour nous simplifier la vie et ainsi cartographier plus simplement les assets exposés d'un domaine.



On peut voir que DNSDumpster tente de commencer à nous donner un aperçu des services et des technologies qui les supporte cependant, ces informations sont assez rudimentaires.

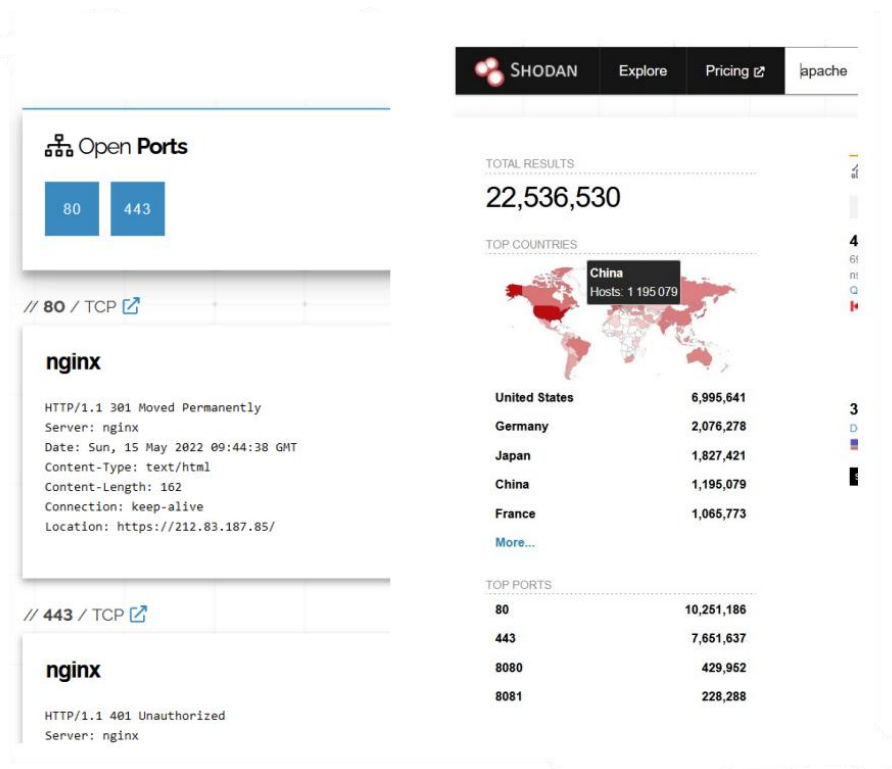
Pour aller plus loin, nous allons pouvoir nous baser sur des outils tel que Shodan qui scanne en continue « Internet » afin de cartographier et de fournir un moteur de recherche des systèmes connectés.

Notre intérêt ici est de venir chercher des informations que l'on ne pourrait en temps normal récupérer qu'avec de la reconnaissance active (scan de ports, ...) sans laisser d'empreinte étant donné que notre machine n'opère pas directement ces scans.

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

On y retrouve des informations sur les ports ouverts et les services qui y répondent.

On y trouve également des informations géographiques, des données concernant l'hébergement de la ressource scannée, ..



Tous les mécanismes présentés jusqu'à présents permettent de récupérer des informations sans informer la chaîne de sécurité qu'une attaque est en cours.

Ces informations ne suffisant pas pour opérer une attaque, nous allons devoir opérer de la reconnaissance active pour aller plus loin.

Note : Une reconnaissance passive peut dans certains cas être suffisante pour exploiter un système directement (notamment grâce à Shodan).

A partir du moment où l'on opère de la reconnaissance active sur un système, il est nécessaire d'avoir une autorisation légale de le faire.

« En droit français, la tentative de commettre une infraction est punie au même titre que l'infraction elle-même »

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Dans la plupart des cas, la reconnaissance active va passer par l'opération d'une simple connexion avec les outils présents sur la plupart des machines (ping, traceroute, telnet, un navigateur internet, ...)

Par exemple, un simple ping peut nous aiguiller sur le type d'OS présent de l'autre côté du ping.

Initialement, un ping sert à tester la connectivité entre deux machines, nous allons pouvoir nous en servir pour d'autres dessins (ici exploitation du TTL) mais aussi pour opérer des cartographies de tout un segment réseau

```
C:\Users\Alex>echo "Pinging Windows" && ping -n 1 127.0.0.1 |  
"Pinging Windows"  
Réponse de 127.0.0.1 : octets=32 temps<1ms TTL=128  
  
C:\Users\Alex>echo "Pinging Linux" && ping -n 1 192.168.56.95  
"Pinging Linux"  
Réponse de 192.168.56.95 : octets=32 temps<1ms TTL=64
```

Le Time To Live défini combien de routeurs un paquet peut traverser avant d'être jeté (et ainsi éviter les boucles réseau), il s'agit d'un mécanisme IP basique.

Il est décrémenté à chaque fois que le paquet passe un périphérique de couche 3.

Il peut cependant être utilisé de différentes manières afin de collecter des informations sur les systèmes présents au sein de l'infrastructure. Notamment, en regardant le TTL d'un paquet revenant, nous allons pouvoir connaître le nombre de routeur entre nous et notre cible.

Avec un simple trace route, nous allons pouvoir également connaître l'adresse IP de chaque routeur traversé (et ainsi procéder à de la reconnaissance facilement).

Il s'agit d'un mécanisme assez peu bruyant et donc très appréciable pour du pivoting.

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Tous les services opérant un travail en clair, à savoir la majorité des services qu'on trouve dans une infrastructure par défaut (HTTP, FTP, SMTP, POP, ...) fonctionnent sur la base d'un simple socket TCP en clair avec une surcouche protocolaire applicative uniquement.

Il est donc possible de les interroger manuellement avec un simple telnet et de suivre le protocole applicatif standard pour récupérer les informations qui nous intéressent (en l'occurrence, la footprint du service, c'est à dire sa bannière).

Ici : Apache/2.4.52 (Debian)

```
(kali@kali)-[~]  
$ telnet 127.0.0.1 80  
Trying 127.0.0.1...  
Connected to 127.0.0.1.  
Escape character is '^]'.  
GET / HTTP/1.0  
host: toto.com  
  
HTTP/1.1 200 OK  
Date: Sun, 22 May 2022 15:08:29 GMT  
Server: Apache/2.4.52 (Debian)  
Last-Modified: Mon, 07 Feb 2022 17:26:12 GMT  
ETag: "29cd-5d770e59af500"  
Accept-Ranges: bytes  
Content-Length: 10701  
Vary: Accept-Encoding  
Connection: close  
Content-Type: text/html
```


Techniques de Hacking Niveau 1 – Surface d’attaque et Vulnérabilités

Une alternative à telnet est netcat (nc) qui va nous permettre en plus de pouvoir opérer un client TCP simple, de s’en servir pour créer un serveur TCP (et donc faire du tunneling par exemple).

Ce mécanisme nous sera particulièrement utile dans la mesure où l’on va donc pouvoir faire du forwarding FIFO entre un serveur TCP basique et un processus présent sur la machine (au hasard un /bin/bash par exemple).

```
(kali@kali)-[~]  
$  
(kali@kali)-[~]  
$ nc localhost 4444  
Hello World !
```

```
(kali@kali)-[~]  
$ cat toto.sh  
#!/bin/bash  
  
echo "Hello World !";  
exit;
```

```
(kali@kali)-[~]  
$ nc -lvp 4444 -e ./toto.sh  
listening on [any] 4444 ...  
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 34120  
(kali@kali)-[~]  
$
```

```
(kali@kali)-[~]  
$ nc localhost 4444  
Hello world !  
Bonjour à tous !  
^C
```

```
(kali@kali)-[~]  
$ nc -lvp 4444  
listening on [any] 4444  
connect to [127.0.0.1]  
Hello World !  
Bonjour à tous !  
(kali@kali)-[~]  
$
```

Techniques de Hacking Niveau 1 – Surface d'attaque et Vulnérabilités

Dans la mesure où l'on peut exécuter un programme arbitraire comme réponse au micro service TCP que l'on vient de déployer, rien ne nous empêche de le transformer en une backdoor fonctionnelle (accessible par un simple nc).

```
(kali@kali)-[~]  
$ nc localhost 4444  
whoami  
root
```

```
(kali@kali)-[~]  
$  
(kali@kali)-[~]  
$ sudo nc -lvp 4444 -e /bin/bash  
listening on [any] 4444 ...  
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 34124
```

Cependant, cela implique à la fois de posséder en amont un accès à la machine pour lancer la backdoor et surtout d'être intégré dans un environnement réseau permettant un accès à la backdoor (même système d'information, même subnet, ...)

TECHNIQUES DE HACKING – NIVEAU 1

TRAVAUX PRATIQUES