

R316-CYBER

L'outil nmap

Sami Evangelista
IUT de Villetaneuse
Département Réseaux et Télécommunications
2023–2024

<http://www.lipn.univ-paris13.fr/~evangelista/cours/R316-CYBER>

Ce document est mis à disposition selon les termes de la licence Creative Commons "Attribution – Pas d'utilisation commerciale – Partage dans les mêmes conditions 3.0 non transposé".



Objectifs :

- ▶ trouver les hôtes actifs (cartographies)
- ▶ trouver les ports ouverts (balayage de ports)
- ▶ trouver les services disponibles
- ▶ identifier les versions des OS et/ou services
- ...

Objectifs :

- ▶ trouver les hôtes actifs (cartographies)
- ▶ trouver les ports ouverts (balayage de ports)
- ▶ trouver les services disponibles
- ▶ identifier les versions des OS et/ou services
- ...

Quelle utilité ?

- ▶ Pour l'administrateur :
 - ▶ inventaire du réseau (IP utilisées, services disponibles)
 - ▶ dans le cadre d'un audit de sécurité : trouver les failles potentielles (p.ex., versions de logiciels, services non autorisés) pour les corriger
- ▶ Pour l'attaquant :
 - ▶ trouver les failles pour les exploiter

`https://insecure.org/`

- ▶ logiciel libre
- ▶ première version : 1997
- ▶ outil de référence pour le scan de réseaux
- ▶ disponible sur de nombreux systèmes (Linux, Mac, Windows, ...)
- ▶ nombreuses possibilités de scan (détection d'hôtes, balayage de ports, détection de version, prise d'empreinte TCP/IP, ...)
- ▶ interface en mode texte, ou graphique (via l'outil zenmap)

Syntaxe générale :

```
# nmap [options] hôte(s)
```

avec

- ▶ options = options de découverte et/ou de scan
- ▶ hôte(s) = IP ou nom(s) à scanner.

Par exemple :

- ▶ 10.0.0.1
- ▶ nom.domaine.tld
- ▶ 10.0.0.0/24 ⇒ toutes les machines du réseau
- ▶ 10.0.0.1-10 ⇔ 10 IP : 10.0.0.1, 10.0.0.2, ... 10.0.0.10
- ▶ 10.0.0.1,10 ⇒ 2 IP : 10.0.0.1 et 10.0.0.10

nmap procède en deux temps :

1. découverte des hôtes
nmap va identifier les hôtes actifs parmi ceux fournis en argument
2. scan des hôtes
lance les scans sur les hôtes actifs

Remarques :

- ▶ La première étape peut permettre d'éviter de lancer de nombreux scans sur des hôtes à l'arrêt.
- ▶ L'étape de découverte peut être sautée : option `-P0`.
 - ▶ Dans ce cas tous les hôtes sont considérés comme actifs (et nmap lance l'étape 2 sur tous les hôtes passés en argument).
- ▶ Idem pour l'étape de scan : option `-sP`.
 - ▶ Dans ce cas nmap va juste détecter les hôtes actifs.

Il y a de nombreuses façons de découvrir/scanner les hôtes.

Options de découverte

- ▶ découverte avec un ICMP echo : `-PE`
- ▶ découverte avec TCP : `-PT <numéro-de-port>`
- ▶ découverte avec UDP : `-PU <numéro-de-port>`

Options de scan

- ▶ balayage de port : `-sS`, `-sT`, `-sX`, `-sN`
 - ▶ `-p` pour préciser les ports à scanner. (par défaut, nmap scanne 1000 ports parmi les plus courants).
- ▶ détection de version des services : `-sV`
- ▶ prise d'empreinte TCP/IP pour la détection d'OS : `-O`

Remarques :

- ▶ Avec une même commande, le trafic observé pourra changer selon que
 - ▶ l'utilisateur est root ou non (possibilité de forger des paquets ou pas)
 - ▶ la cible est sur le même réseau ou non (possibilité d'utiliser ARP ou pas)

Suite à un balayage de port nmap associe un des états suivants à un port :

- ▶ ouvert — service actif sur ce port
- ▶ fermé — port accessible mais aucun service actif sur ce port
- ▶ filtré — port inaccessible (p.ex., à cause d'un pare-feu)
- ▶ non filtré — port accessible mais nmap est incapable de déterminer si le port est ouvert ou fermé
- ▶ ouvert | filtré — le port est soit ouvert soit filtré
- ▶ fermé | filtré — le port est soit fermé soit filtré

Comment nmap peut-il déterminer la version d'un service écoutant sur un port ouvert ?

- ▶ utilisation d'une base de données d'empreinte de services : `nmap-service-probes` qui est constamment enrichie par les utilisateurs/développeurs de nmap avec de nouvelles empreintes.
- ▶ une empreinte = réponse typique d'un service (outil + version) à une requête particulière
- ▶ Cette base contient une suite de tests à effectuer sous la forme :
 - ▶ envoyer telle requête sur tel port
 - ▶ si la réponse du serveur contient telle chaîne de caractères, alors le serveur est probablement de tel type
- ▶ Le premier test effectué est le test de la bannière (test NULL) :
 - ▶ connexion TCP au serveur
 - ▶ attente d'un message envoyé par le serveur (la bannière)
 - ▶ La bannière (si elle est envoyée) contient parfois l'empreinte du serveur.

Le test de la bannière fonctionne avec de nombreux services (ftp, ssh, ...).

- ▶ Pour détecter l'OS et la version de l'OS d'un hôte le principe est le même.
- ▶ Nmap effectue une série de tests pour prendre l'empreinte TCP/IP de la cible (comment la cible répond à divers paquets TCP, UDP, ICMP, ...).
- ▶ Tests documentés sur
<https://nmap.org/book/osdetect-methods.html>
- ▶ L'empreinte prise par nmap est comparée à celle de la base de données `nmap-os-db` pour déterminer l'OS et la version de l'OS de la cible.