|  |  |  |
| --- | --- | --- |
| nmap  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)    http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png  nmap (suite)  http://upload.wikimedia.org/wikipedia/commons/thumb/3/35/Tux.svg/512px-Tux.svg.png |  | nmap – Le balayeur de ports  *nmap* est un balayeur de ports *open source* qui permet l’exploration réseau et obtenir des informations sur le système d’exploitation d’un ordinateur distant. Ce logiciel est de plus en plus utilisé par les administrateurs réseaux car l’audit des résultats de *nmap* fournit des indications bien précises d’un réseau.  **Exemples de balayages**  Voici quelques exemples d'utilisation de *nmap*, du plus simple et un peu plus complexe et ésotérique.  De véritables adresses IP et noms de domaine sont utilisés pour rendre les choses plus concrètes.  Il est possible de les substituer avec celles de votre propre réseau.  **Bien qu’il soit généralement admis que balayer d'autres réseaux soit ou devrait être illégal, certains administrateurs de réseau n'apprécient pas les balayages non sollicités de leur réseau et peuvent s'en plaindre.**  **La meilleure approche est donc d'obtenir d'abord leur autorisation.**  Pour des raisons de tests, il est autorisé de balayer l'hôte *scanme.nmap.org*.  Cette permission inclus seulement les balayages avec *nmap* et non pas l'essai d'exploits ou d'attaques de dénis de service.  Afin de préserver la bande passante, Il est suggéré de ne lancer qu'une douzaine de balayages sur cet hôte au maximum par jour.  En cas d'abus de ce libre-service de cible de balayage, il serait fermé et *nmap* afficherait le message suivant …  *Failed to resolve given hostname/IP: scanme.nmap.org*.  Ces permissions s'appliquent aussi à l'hôte *scanme2.nmap.org*, à *scanme3.nmap.org*, et ainsi de suite, même si ces hôtes n'existent présentement pas.  **Identification du système d’exploitation d’une machine (distante ou non)**  Cette commande essaie d’identifier le système d’expiation de l’hôte local.  # **nmap -O 127.0.0.1**  Si *nmap* n’arrive pas à identifier la version du système d’exploitation, il est possible d’afficher une liste des systèmes d’exploitation qui pourraient potentiellement correspondre …  # **nmap -O --osscan-guess 127.0.0.1**  **Balayage générique**  Cette commande balaye tous les ports réservés TCP sur la machine *scanme.nmap.org*.  L'option -v active le mode verbeux.  # **nmap -v scanme.nmap.org**  **Détection des hôtes sur un réseau**  Si on souhaite identifier toutes les hôtes présents sur un réseau …  # **nmap -sP 192.168.0.0/10**  **Visualisation des ports ouverts sur un hôte**  Un balayage de ports permet de voir comment votre machine est vue depuis l’extérieur, quels sont les services qui sont accessibles.  Par contre un balayage de ports permet un pirate informatique de voir les services qui tournent à distance afin de trouver une faille sur l’un de ces services et comme ça il pourra aller plus loin dans l’attaque.  # **nmap 127.0.0.1**  Pour balayer un port précis. Ici, c’est le port HTTP :  # **nmap –p 80 127.0.0.1**  **Balayage furtif d’un réseau**  Cette commande lance un balayage furtif (*stealth SYN scan*) contre chaque machine active parmi les 255 machines du réseau de “classe C” sur lequel l’hôte *scanme.nmap.org* réside.  Il essaie également de déterminer le système d'exploitation sur chaque hôte actif.  Cette démarche nécessite les privilèges de *root* puisqu'on utilise un SYN scan et une détection de SE.  # **nmap -sS -O scanme.nmap.org/24**  **Détermination des services (démons ou *deamons*) en fonction sur une partie d’un réseau**  Cette commande lance une recherche des hôtes et un balayage TCP dans la première moitié de chacun des 255 sous‑réseaux à 8 bits dans l'espace d'adressage de classe B 198.116.0.0.  Elle permet de déterminer si les systèmes ont les services sshd, DNS, pop3d, imapd en fonction ou si le port 4564 est ouvert.  Pour chacun de ces ports qui sont ouverts, la détection de version est utilisée pour déterminer quelle application est actuellement lancée.  # **nmap -sV -p 22,53,110,143,4564 198.116.0-255.1-127**  **Balayage aléatoire pour un service donné**  Cette commande demande à *nmap* de choisir 100 000 hôtes de façon aléatoire et de les balayer dans le but de trouver les serveurs Web (port 80).  L'énumération des hôtes est désactivée avec -P0 puisqu’envoyer quelques essais (*probes*) pour déterminer si un hôte est actif est inutile lorsque l’on ne cherche qu’à tester un port sur chaque hôte.  # **nmap -v -iR 100000 -P0 -p 80**  **Balayage spécifique pour un service donné**  La prochaine procédure balaye 4 096 adresses IP à la recherche de serveurs Web (sans leur envoyer un *ping* au préalable) et sauvegarde la sortie en format grepable et XML.  # **nmap -PN -p80 -oX logs/pb-port80scan.xml -oG logs/pb-port80scan.gnmap 216.163.128.20/20**  *nmap* – Le résumé des principaux commutateurs  **Spécifications des cibles**  Les cibles peuvent être spécifiées par …  • des noms d'hôtes (scanme.nmap.org) ;  • des adresses IP (192.168.10.100) ;  • des adresses de réseaux (cmaisonneuve.qc.ca/24, 192.168.10.0/24) ;  • dés étendues spécifiques (10.0-255.0-255.1-254) ;  • etc.  **Lire une liste d’hôtes/réseaux cibles à partir du fichier**  # **nmap -iL <nom\_du\_fichier>**  **Choisir des cibles au hasard**  # **nmap -iR <nombre\_d’hôtes>**  **Exclure des hôtes/réseaux du balayage**  # **nmap --exclude <hôte1[,hôte2][,hôte3],...>**  **Exclure des hôtes/réseaux des cibles à partir d’un fichier**  # **nmap --excludefile <fichier\_exclusions>**  **Découverte des hôtes**  **Lister des hôtes – Liste simplement les cibles à balayer**  # **nmap -sL**  **Balayer avec ping – Détermination des hôtes en ligne**  # **nmap -sP**  **Considèrer que tous les hôtes sont en ligne -- évite la découverte des hôtes**  # **nmap -P0**  # **nmap -PN**  **Découvrir les drapeaux (*flags*) TCP SYN/ACK ou UDP des ports en paramètre**  # **nmap -PS/PA/PU [liste\_de\_ports]**  **Découvrir à l’aide de requête ICMP *echo*, *timestamp* ou *netmask***  # **nmap -PE/PP/PM**  **Découvrir à l’aide de ping IP (par type)**  # **nmap -PO [numéro\_de\_protocole]**  **Ne jamais résoudre les noms DNS/Toujours résoudre** [résout les cibles actives par défaut]  # **nmap -n/-R**  **Spécifier les serveurs DNS particuliers**  # **nmap --dns-servers <serveur1[,serveur2],...>**  **Techniques de balayage**  **Balayer à l’aide des TCP SYN/Connect()/ACK/Window/Maimon**  # **nmap -sS/sT/sA/sW/sM**  **Balayer à l’aide de TCP Null, FIN et Xmas**  # **nmap -sN/sF/sX**  **Balayer le service UDP**  # **nmap -sU**  **Personnaliser les drapeaux (*flags*) des balayages TCP**  # **nmap --scanflags <drapeaux>**  **Balayar passivement (*Idlescan*)**  # **nmap -sI < hôte\_zombie [:probeport]>**  **Balayer les protocoles supportés par la couche IP**  # **nmap -sO**  **Balayer par rebond FTP**  # **nmap -b <hôte\_de\_relais\_FTP>**  **Déterminer une route vers chaque hôte**  # **nmap --traceroute**  **Donner** **la raison pour laquelle tel port apparait à tel état**  # **nmap --reason**  Spécifications des ports et ordre de balayage  **Ne balayer que les ports spécifiés**  # **nmap -p <étendue\_de\_ports>**  Exemple …  # **nmap -p22; -p1-65535; -pU:53,111,137,T:21-25,80,139,8080**  **Rapide - Ne balayer que les ports listés dans le fichier *nmap-services***  # **nmap –F**  **Balayer séquentiellement les ports en ne mélangeant pas leur ordre**  # **nmap –r**  **Balayer un nombre de ports parmi les plus courants**  # **nmap --top-ports <nombre>**  **Balayer un ratio (pourcentage) des ports les plus courants**  # **nmap --port-ratio <ratio>**  Détection de service/version  **Tester les ports ouverts afin de déterminer le service est en écoute et sa version**  # **nmap –sV**  **Limiter les tests aux plus probables pour une identification plus rapide**  # **nmap --version-light**  **De 0 (léger) à 9 (tout essayer)**  # **nmap --version-intensity <niveau>**  **Essayer un à un tous les tests possibles pour la détection des versions**  # **nmap** **--version-all**  **Afficher des informations détaillées du balayage de versions (pour débogage)**  # **nmap 1/**  Évasion pare-Feu/IDS et usurpation d'identité  **Fragmenter les paquets (en spécifiant éventuellement la MTU)**  # **nmap -f; --mtu <valeur>**  **Obscurcir le balayage avec des leurres**  # **nmap -D <decoy1,decoy2[,ME],...>**  **Usurper l'adresse source**  # **nmap -S <adresse\_ip>**  **Utiliser l'interface réseau spécifiée**  # **nmap -e <interface>**  **Utiliser le numéro de port comme source**  # **nmap -g/--source-port <numéro\_de\_port>**  **Ajouter des données au hasard aux paquets émis**  # **nmap --data-length <numéro>**  **Envoyer des paquets avec les options IP spécifiées**  # **nmap --ip-options <options>**  **Spécifier le champ durée de vie (time-to-live) IP**  # **nmap --ttl <valeur>**  **Usurper une adresse physique (MAC)**  # **nmap --spoof-mac <adresse MAC, préfixe ou nom du fabriquant>**  Envoer des paquets TCP/UDP avec une somme de contrôle erronée.  # **nmap –badsum**  **Sortie**  **Diriger la sortie vers un fichier des résultats du balayage au format normal, *XML*, *s|<rIpt kIddi3* et *grepable***  # **nmap -oN/-oX/-oS/-oG <fichier>**  **Diriger la sortie dans les trois formats majeurs en même temps**  # **nmap -oA <nom\_de-base>**  **Rendre nmap plus verbeux (*-vv* pour plus d'effet)**  # **nmap –v**  **Sélectionner ou augmenter le niveau de débogage (significatif jusqu'à 9)**  # **nmap -d[level**  **Afficher tous les paquets émis et reçus**  # **nmap --packet-trace**  **Afficher les interfaces et les routes de l'hôte (pour débogage)**  # **nmap –iflist**  **Journaliser les erreurs/alertes dans un fichier au format normal**  # **nmap --log-errors**  **Ajouter la sortie au fichier plutôt que de l'écraser**  # **nmap --append-output**  **Reprendre un scan interrompu**  # **nmap --resume <nom\_de\_fichier>**  **de styles XSL afin de transformer la sortie XML en HTML**  # **nmap --stylesheet <chemin/URL>**  **Appliquer une feuille de styles de références de Insecure.Org pour un XML plus portable**  # **nmap –webxml**  **Ne pas appliquer une feuille de styles XSL à la sortie XML**  # **nmap --no\_stylesheet**  Détection de système d'exploitation:  **Activer la détection d'OS**  # **nmap –O**  **Limiter la détection aux cibles prometteuses *--osscan-guess*  Détecter l'OS de façon plus agressive**  # **nmap --osscan-limit**  **Deviner l'OS de façon plus agressive**  # **nmap --osscan-guess**  Divers  **Activer le balayage IPv6**  # **nmap -6**  **Activer la détection du système d'exploitation et des versions**  # **nmap –A**  **Spécifier un dossier pour les fichiers de données de *nmap***  # **nmap --datadir <nom\_du\_répertoire>**  **Envoyer des paquets en utilisant des trames *Ethernet* ou des paquets IP bruts**  # **nmap --send-eth/--send-ip**  **Afficher le numéro de version**  Suppose que l'utilisateur est entièrement privilégié -V  **nmap –privileged**  Suppose que l'utilisateur n'a pas les privilèges d'usage des *raw* socket  **nmap –unprivileged**  **Afficher le résumé de l'aide**  **nmap –h** |
|  |  |  |