Nmap (Network Mapper)
Outil d'exploration réseau et scanneur de ports/sécurité

1. Présentation

Nmap est un outil open source d'exploration réseau et d'audit de sécurité, utilisé pour scanner de grands réseaux mais fonctionne également pour une cible unique. Il utilise à cette fin des paquets IP bruts afin de déterminer :

- Les hôtes actifs sur le réseau,
- Les services (nom de l'application et la version),
- Les systèmes d'exploitation et leurs versions,
- Les types de dispositifs de filtrage / Pare-feux,
- …

Le résultat de retour est un rapport présentant une table des ports, les services qui leurs correspondent et leurs états, l'adresse MAC etc...

```
Starting nmap 3.81(http://www.insecure.org/nmap/) at 2006-05-03 10:05UTC
Interesting ports on 192.160.100.1
PORT         STATE      SERVICE
80/tcp         open        http
135/tcp        open        msrpc
…
MAC Address: 0E:0A:FF:18:56:32 (Dell)
Running: Microsoft Windows 95/98/ME INT/2K/XP
…
Nmap finished: 1 IP address (1 host up) scanned in 1,483 seconds
```

Figure 1 rapport d'une opération de scan

L'état d'un port est soit :

- *Open* : l'application de la machine cible est en écoute de paquets/connexions sur ce port.
- *Closed* : pas d'application en écoute, bien qu'ils puissent quand même s'ouvrir n'importe quand.

- *Filtred* : indique qu'un pare-feu, un dispositif de filtrage ou un autre obstacle réseau bloque ce port, empêchant ainsi Nmap de déterminer s'il s'agit d'un port ouvert ou fermé
- *Unfiltred* : lorsqu'ils répondent aux paquets de tests (probes) de Nmap, mais Nmap ne peut déterminer s'ils sont ouverts ou fermés.
- Nmap renvoie également les combinaisons d'états ouverts|filtré et fermés|filtré lorsqu'il n'arrive pas à déterminer dans lequel des deux états possibles se trouve le port.

Syntaxe générale :

Nmap [*Types de scans* ...] [*Options*] { *spécifications des cibles* }

a) Types de Scans :

Nmap offre différents types de scan tel que scan UDP, scan TCP, scan du protocole IP…

b) Options :

Permettent de déterminer les caractéristiques désirées de la cible.

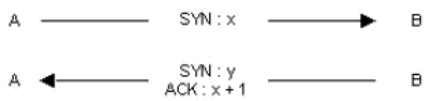
c) Spécification des cibles:

Les cibles peuvent être spécifiées par des noms d'hôtes, des adresses IP, des adresses de réseaux, etc…

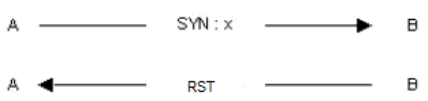
2. Technique de Scan

Scan TCP SYN -sS

Appelé scan demi-ouvert car il n'établit pas pleinement une connexion. Envoi un paquet SYN et attente de réponse ACK.



port ouvert



port fermé

Si aucune réponse n'est reçue le port est considéré comme *filtré*.

(+) Rapide, discret, furtif.
(--) Pas de résultat en présence d'obstacles (Pare-feux).

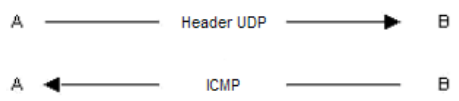
Scan TCP connect () –sT

Nmap demande au système d'exploitation qui l'exécute d'établir une connexion au port de la machine cible grâce à l'appel système *connect ()*.

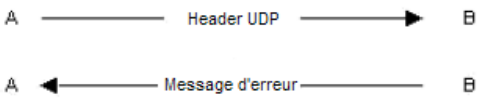
(+) la probabilité que les cibles activent la connexion est plus grande.

(--) plus long que le scan TCP SYN, demande plus de paquets pour obtenir la même information.

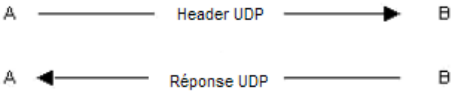
Scan UDP -sU



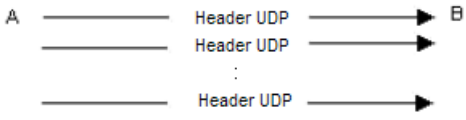
port fermé



port filtré



port ouvert



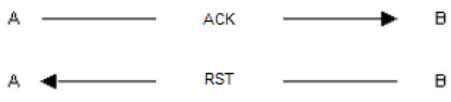
port ouvert | filtré

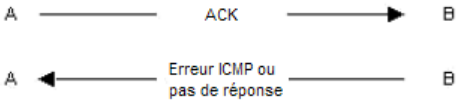
Pour différencier les ports filtrés des ports ouverts on peut utiliser le scan (–sV).

(--) Difficulté et retard de l'exécution vu que les ports ouverts et filtrés ne renvoient que rarement des réponses. Nmap expire de ce fait son délai de retransmission au cas où les paquets se soient perdus.

Scan TCP ACK -sA

Utilisé pour établir les règles des pare-feux, déterminant s'ils sont avec ou sans états et quels ports sont filtrés.

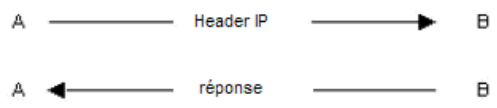




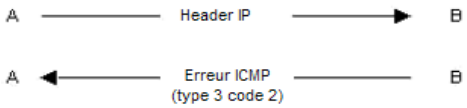
Systèmes non filtrés (pas de pare-feu) ports filtrés

Scan du protocole IP –sO

Ce scan permet de déterminer quels protocoles IP sont supportés par la cible.



port ouvert



port fermé

Scan par rebond FTP –b<ftp relay host>

Pour scanner un port, le serveur ftp envoie un fichier au port de la cible. Et il se charge d'effectuer le scan et il décide selon le message d'erreur si le port est ouvert ou fermé.

Scan paresseux -- idlescan sI <zombie host[:probeport]>

C'est un scan de port TCP en aveugle ; la technique employée consiste à récolter des informations sur les ports ouverts de la cible en utilisant un exploit basé sur la prédictibilité de la génération des identifiants de fragmentation IP de l'hôte relais (le zombie).

(+) Possibilité d'utiliser différentes machines zombies.
(--) Nécessité d'une relation de confiance entre la machine zombie et la machine cible.

Scan de fenêtre TCP –sW

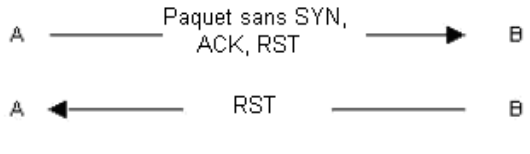
Ce type de scan est similaire au scan TCP ACK à la différence prés qu'il parvient à déterminer -quand les ports sont non filtrés suite à la réception d'un RST – s'ils sont ouverts ou fermés en utilisant la taille de la fenêtre TCP :

- Si la taille de la fenêtre est positif alors le port est ouvert.
- Si la taille de la fenêtre est nulle alors le port est fermé.

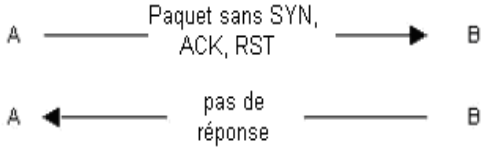
(--) utilisé uniquement sur certains systèmes, vous ne pouvez donc pas toujours vous y fier.

Scan TCP Null, FIN et Xmas -sN; -sF; -sX

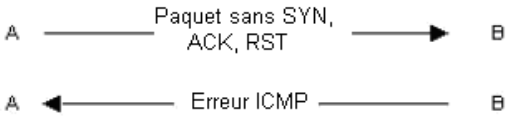
Ces trois types de scans exploitent une subtile faille de la RFC TCP : chaque paquet ne contenant ni SYN, ni RST, ni ACK se voit renvoyé un RST.



port fermé



port `ouvert|filtré`



port filtré

Scan TCP personnalisé –scanflags

L'option --scanflags vous permet de créer votre propre type de scan en spécifiant vos propres combinaisons de drapeaux TCP.

3. Limites de Nmap :

- Même avec tous les tests vus plus haut, Nmap est incapable de distinguer les piles TCP de Win95, WinNT ou Win98.

Solution : On peut simplement commencer avec les premières attaques de DoS contre Windows (Ping of Death, Winnuke, etc…) et évoluer vers des attaques plus avancées comme Teardrop et Land. Apres chaque attaque on les teste pour voir s'ils plantent. Quand nous les planterons finalement, nous serons à même de déterminer ce qu'ils utilisent au service pack ou patch prés.

- Technique de SYN Flood : Certains systèmes d'exploitation arrêterons d'accepter de nouvelles connexions si on leur envoi trop de paquet SYN. Beaucoup de systèmes d'exploitation géreront uniquement 8 paquets. Les noyaux Linux récents (parmi d'autres OS) autorisent plusieurs méthodes comme les cookies SYN empêchant cela de devenir un problème sérieux. Ainsi on peut apprendre quelques informations sur l'OS cible en envoyant 8 paquets forgés à un port ouvert et tester ensuite si on peut établir une connexion sur ce port. Cela n'a pas été implémenté dans Nmap car certaines personnes n'apprécient pas de subir un SYN flood.

4. commandes utiles :

Nmap -sP : Ping les hôtes. Détecte les machines actives.

Nmap -sS : Voir tous les ports TCP ouverts sur une machine, utilisation de messages SYN, donc pas de log sur la machine cible.

Nmap -D : Donne l'impression aux hôtes scannés d'être scannés par toutes ces adresses IP spécifiées.

Nmap -sU: Voir tous les ports UDP ouverts sur une machine.

Nmap -O : Connaitre le système d'exploitation de la machine (TCP/IP fingerprint).

Nmap -O –osscan-guess : Si nmap n'arrive pas à déterminer la version, on pourra lui demander de nous donner une liste des systèmes qui pourraient potentiellement Correspondre.

Nmap -b : Scan par rebon ftp, permet de demander à un serveur FTP de scanner les ports à votre place (envoie des fichiers pour tester les ports ouverts). Cette fonctionnalité est souvent désactivée des serveurs FTP afin d'éviter les abus. Ici on passe par le serveur ftp qui a pour adresse 127.0.0.1 pour scanner une plage d'adresses ip.

Nmap -sV : Détermine les services sur les ports ouverts: nom, version…

Nmap -T Insane : Scanne pas à la même fréquence tout les ports

Nmap -P0 : Ne ping pas les machines, utile face à un Firewall.

Nmap 192.168.0.0-255 : Scanner une plage d'adresses. Ici toutes les adresses de 192.168.0 à 192.168.255

Nmap -n 127.0.0.1 : Désactiver la résolution DNS inverse des hôtes, augmente la rapidité.

Nmap -p 80 127.0.0.1 : Scanner un port précis. Ici, c'est le port http.

Nmap -p 0-80,60000 127.0.0.1 : Scanner une plage de ports. Ici on scan du port 0 au 80 et tous ceux supérieurs à 60000.

Nmap -v -sS -iR 0 -p 80 : Scanner des serveurs web au hasard sur le réseau.

Usurper l'adresse ip source:

Ici on scan 127.0.0.1, par l'interface réseau eth0, en se faisant passer pour 10.0.0.0 depuis le port80: nmap -S 10.0.0.0 -g 80 -e eth0 -P0 127.0.0.1

Usurper l'adresse MAC:

nmap --spoof-mac 01:02:03:04:05:06 127.0.0.1

nmap --spoof-mac Cisco 127.0.0.1

Trace les paquets et les données envoyés et Reçus:

Pratique pour vérifier qu'une usurpation Fonctionne:

nmap --packet-trace -S 10.0.0.0 -eth0 127.0.0.1