Exemplos comandos Nmap

Fonte: Ezaul Zillmer
Adaptado por: Kelton Costa

O Nmap é um dos mais utilizados e completos programas para se fazer uma análise/rastreio de uma servidor, rede ou subnet. Sendo um dos melhores, isso implica que tenha várias facetas, utilidades e uma lista enorme de comandos e opções. Foi desenvolvido por Gordon Lyon que, com este programa, tentou resolver algumas questões em relação aos testes que fazia: Que computadores estão ligados na rede local? Que IPs se encontram na rede? Qual o sistema operacional do alvo? Que portas tem o alvo abertas? Descobrir se o sistema está infectado com vírus ou malware. Pesquisar por computadores ou serviços não autorizados na rede.

EXEMPLO 1: ANALISAR UM IP OU DOMÍNIO
Análise a um IP:
nmap 192.168.2.2
Análise de um domínio:
nmap teste.com
Análise com mais informações:
nmap -v 192.168.2.2

EXEMPLO 2: ANALISAR MÚLTIPLOS IPS OU UMA REDE SUBNET
Vários IPs , separando-os com um espaço:
nmap 192.168.1.1 192.168.1.2 192.168.1.
Dentro da rede:
nmap 192.168.1.1,2,3
De x a x, numa seleção de IPs:
nmap 192.168.1.1-20
Uma seleção com um wildcard:
nmap 192.168.1.*
Ou, uma rede completa:
nmap 192.168.1.0/24

EXEMPLO 3: SELECIONAR OS ALVOS A PARTIR DE UMA PASTA
Criar uma pasta em que são introduzidos os alvos:
* cat > /root/Desktop/alvos.txt
E dentro desta pasta os nossos alvos:
test.com 192.168.1.0/24 google.pt facebook.com 8.8.8.8
E o comando:
nmap -iL /root/Desktop/alvos.txt

EXEMPLO 4: EXCLUIR ALVOS DE UMA REDE
Quando se analisa uma rede grande através do Exemplo 2, pode-se remover alguns hosts:
nmap 192.168.1.0/24 --exclude 192.168.1.5
nmap 192.168.1.0/24 --exclude 192.168.1.5,192.168.1.254
Ou, através de uma pasta de exclusão como mostrado no exemplo 3:
nmap -iL /tmp/scanlist.txt --excludefile /tmp/exclude.txt

EXEMPLO 5: TENTAR DETECTAR O SISTEMA OPERACIONAL E A SUA VERSÃO
Com o comando "-A":
nmap -A 192.168.1.254
nmap -v -A 192.168.1.1
nmap -A -iL/root/Desktop/alvos.txt

EXEMPLO 6: DESCOBRIR SE O ALVO É PROTEGIDO POR UMA FIREWALL
Com o comando "-sA":
nmap -sA 192.168.1.254
nmap -sA facebook.com

EXEMPLO 7: COMO ANALISAR QUANDO O ALVO É PROTEGIDO POR UMA FIREWALL
Com o:
nmap -PN 192.168.1.1
nmap -PN server1.cyberciti.biz

EXEMPLO 8: ANALISAR UM ALVO COM O IPV6
nmap -6 IPv6-Address-Here nmap -6 google.pt nmap -6 2607:f0d0:1002:51::4
nmap -v -A -6 2607:f0d0:1002:51::4

EXEMPLO 9: ANÁLISE E DESCOBERTA DE HOST LIGADOS
Técnica conhecida por ping ou descoberta de alvos:
nmap -sP 192.168.1.0/24

EXEMPLO 10: ANÁLISE RÁPIDA
Utilizando o argumento "-F":
nmap -F 192.168.1.1

EXEMPLO 11: ADICIONA A RAZÃO PELO QUAL O NMAP DIZ QUE TEM A PORTA ABERTA
nmap --reason 192.168.2.2
nmap --reason google.pt

EXEMPLO 12: APENAS MOSTRA PORTAS ABERTAS (OU POSSÍVEIS ABERTAS)
nmap --open 192.168.1.1
nmap --open google.pt

EXEMPLO 13: MOSTRA TODOS OS PACOTES ENVIADOS E RECEBIDOS
nmap --packet-trace 192.168.1.1
nmap --packet-trace facebook.com

EXEMPLO 14: MOSTRA AS SAÍDAS INSTALADAS ASSIM COMO OS CAMINHOS PERCORRIDOS
nmap --iflist
nmap –iflist

EXEMPLO 15: ANALISAR PORTAS ESPECÍFICAS
Utilizando o comando "nmap -p [port] hostName". Analisar porta 80: # nmap -p 80 192.168.2.2 Analisar porta 80 por TCP:
nmap -p T:80 192.168.2.2
Analisar porta 53 por UDP:
nmap -p U:53 192.168.1.1
Analizar duas portas:
nmap -p 80,443 192.168.1.1
Analisar de x a x porta:
nmap -p 80-200 192.168.1.1
Combinação de todas:
nmap -p U:53,111,137,T:21-25,80,139,8080 192.168.2.2
nmap -p U:53,111,137,T:21-25,80,139,8080 facebook.com
nmap -v -sU -sT -p U:53,111,137,T:21-25,80,139,8080 192.168.2.2
Analisar com um "* wildcard":
nmap -p "*" 192.168.1.1
Analisar as portas mais comuns:

nmap --top-ports 5 192.168.1.1
nmap --top-ports 10 192.168.1.1

EXEMPLO 16: A MANEIRA DE ANÁLISE MAIS RÁPIDA DE PORTAS ABERTAS
nmap -T5 facebook.com

EXEMPLO 17: DETECTAR SISTEMA OPERACIONAL DO ALVO
nmap -O 192.168.2.2

EXEMPLO 18: ANALISAR QUE PROGRAMAS E VERSÃO CORREM NAS PORTAS ABERTAS
nmap -sv 192.168.2.2

EXEMPLO 19: ANALISAR UM ALVO UTILIZANDO TCP ACK (PA) E TCP SYN (PS) PING
Se o Firewall estiver bloqueando os pings normais (ICMP), utilizar o seguinte método de descoberta:
nmap -PS 192.168.2.2
nmap -PS 80,21,443 192.168.2.2
nmap -PS 192.168.2.2
nmap -PS 80,21,200-512 192.168..2.2

EXEMPLO 20: ANALISAR UM ALVO UTILIZANDO O PROTOCOLO PELO IP
nmap -PO 192.168.2.2

EXEMPLO 21: ANALISAR UM ALVO UTILIZANDO UDP PING
nmap -PU 192.168.2.2

EXEMPLO 22: DESCOBRIR AS PORTAS MAIS UTILIZADAS USANDO A ANÁLISE TCP SYN
Análise camuflada:
nmap -sS 192.168.1.1
A portas TCP mais utilizadas:
nmap -sT 192.168.1.1

EXEMPLO 23 ANALISAR SERVIÇOS UDO (ANALISE UDP)
nmap -sU 192.268.2.2

EXEMPLO 24: ANALISAR O PROTOCOLO IP
Esta análise permite detectar quais são os protocolos (TCP, ICMP, IGMP etc) que o alvo suporta:
nmap -sO 192.168.2.2
nmap -sO facebook.com

EXEMPLO 25: PROCURAR FALHAS NO FIREWALL
Uma análise nula para fazer o Firewall gerar uma resposta:
nmap -sN 192.168.2.2
Verificação de Firewall:
nmap -sF 192.168.2.2
Faz os sets FIN, PSH, e URG, serem analisados:
nmap -sX 192.168.2.2

EXEMPLO 26: ANALISAR O FIREWALL COM PACOTES FRAGMENTADOS
nmap -f 192.168.2.2
nmap -f fw2.nixcraft.net.in
nmap -f 15 google.pt
Escolha o vosso offset com a opçãp mtu:
nmap --mtu 32 192.168.1.1

EXEMPLO 27: A OPÇÃO -D FAZ COM QUE O ALVO PENSE QUE ESTÁ SENDO ANALISADO POR MAIS MÁQUINAS
O IDS fará com que se reporte entre 5 a 10 portas a cada IP, mas nunca sabe quais são os verdadeiros e os falsos:
nmap -n -Ddecoy-ip1,decoy-ip2,your-own-ip,decoy-ip3,decoy-ip4 remote-host-ip
nmap -n -D192.168.1.5,10.5.1.2,172.1.2.4,3.4.2.1 192.168.1.5

EXEMPLO 28: ANALISAR COM O ENDEREÇO MAC ALTERADO
nmap --spoof-mac MAC-ADDRESS-HERE 192.168.2.2
Utilizar o endereço MAC aleatório:
nmap -v -sT -PN --spoof-mac 0 192.168.1.1

EXEMPLO 29: SALVAR AS INFORMAÇÕES OBTIDAS PARA UMA PASTA
nmap 192.168.2.2 > output.txt
nmap -oN /path/to/filename 192.168.2.2
nmap -oN output.txt 192.168.2.2

Verificando a Segurança de um Endereço IP

Estudo de Caso

Ambiente para aplicação do estudo

1. servidor virtual com o CentOS 6.7
2. gateway da rede
3. firewall de entrada e saída
4. servidor WEB
5. servidor FTP
6. servidor DNS
7. servidor proxy

Para acessar todos os recursos é utilizado o serviço SSH.

Kernel: 2.6.32-573.el6.x86_64

Iptables: iptables v1.4.7

Servidor Web:

Software utilizado: HTTPD

Versão: 2.2.15 (Unix)

Servidor FTP:

Software utilizado: VSFTPD

Versão: 2.2.2

Servidor DNS:

Software utilizado: BIND

Versão: 9.8.2rc1-RedHat-9.8.2-0.47.rc1.el6

Servidor proxy:

Software utilizado: SQUID

Versão: 3.1.23

Serviço SSH:

Software utilizado: SSH

Versão: OpenSSH_5.3p1, OpenSSL 1.0.1e-fips

Estado das portas
Conceitos para a aplicação do estudo

Aberto (open)

Uma aplicação está ativamente aceitando conexões TCP ou pacotes UDP na referida porta. Identificar tal estado é o objetivo principal de um escaneamento de portas.

Fechado (closed)

Uma porta fechada está acessível (ela recebe e responde a pacotes de sondagens do Nmap), mas não há nenhuma aplicação "ouvindo" nela. Elas podem ser úteis para mostrar que um host está ativo em um determinado endereço IP (descoberta de hosts, ou scan usando ping), e como parte de uma detecção de SO.

Importante: Pelo fato de portas fechadas serem alcançáveis, pode valer a pena escanear mais tarde no caso de alguma delas abrir.

Filtrado (filtered)

O Nmap não consegue determinar se a porta está aberta porque uma filtragem de pacotes impede que as sondagens alcancem a porta. A filtragem poderia ser de um dispositivo firewall dedicado, regras de roteador, ou um software de firewall baseado em host. Essas portas frustram os atacantes pois elas fornecem poucas informações.

Não-filtrado (unfiltered)

O estado não-filtrado significa que uma porta está acessível, mas que o Nmap é incapaz de determinar se ela está aberta ou fechada.

open|filtered

O Nmap coloca portas neste estado quando é incapaz de determinar se uma porta está aberta ou filtrada. Isso acontece para tipos de scan onde as portas abertas não dão nenhuma resposta.

closed|filtered

Este estado é usado quando o Nmap é incapaz de determinar se uma porta está fechada ou filtrada.

Importante: É apenas usado para o scan IPID Idle scan.

Técnicas de escaneamento de portas

-sS (scan TCP SYN)

O scan SYN é a opção padrão e mais popular. Pode ser executada rapidamente, escaneando milhares de portas por segundo em uma rede rápida não bloqueada por firewalls intrusivos. Esta técnica é frequentemente chamada de escaneamento de porta entreaberta (half-open scanning), porque não abre uma conexão TCP completamente.

-sT (scan TCP connect)

O scan TCP connect é o scan padrão do TCP quando o scan SYN não é uma opção. Quando um scan SYN está disponível é normalmente a melhor escolha.

-sU (scans UDP)

Scan de portas rodando no protocolo UDP. Ele pode ser combinado com um tipo de escaneamento TCP como o scan SYN (-sS) para averiguar ambos protocolos na mesma execução.

-sN; -sF; -sX (scans TCP Null, FIN, e Xmas)

Estes três tipos de scan exploram uma brecha sutil na RFC do TCP para diferenciarem entre portas abertas e fechadas.

-sA (scan TCP ACK)

Este scan nunca determina se uma porta está aberta (ou mesmo aberta|filtrada). Ele é utilizado para mapear conjuntos de regras do firewall, determinando se eles são orientados à conexão ou não, e quais portas estão filtradas.

-sO (Scans do protocolo IP)

Scans do protocolo IP permitem determinar quais protocolos IP (*TCP, ICMP, IGMP, etc.)* são suportados pelas máquina-alvo. Isso não é, tecnicamente, um scan de portas, pois ele varia os números do protocolo IP ao invés dos números de portas TCP e UDP.

Como especificar a(s) porta(s) de análise

-p <faixa de portas> (escaneia apenas as portas especificadas)

Esta opção especifica quais portas que você deseja escanear e prevalece sobre o padrão. Números de portas individuais são suportadas, bem como as faixas separadas por um hífen (p.ex.: 1-1023). Os valores iniciais e/ou finais da faixa podem ser omitidos, o que faz com que o Nmap use 1 e 65535.

-F (Scan Rápido (portas limitadas)

Especifica que você deseja apenas escanear as portas listadas no arquivo nmap-services que vem com o nmap (ou o arquivo de protocolos para o -sO). Isto é muito mais rápido do que escanear todas as 65535 portas de um host pelo fato desta lista conter uma grande quantidade de portas TCP (> 1200).

Detecção de SO

-O (habilita a detecção de SO)

Habilita a detecção de SO. Alternativamente, pode-se usar -A para habilitar tanto a detecção de SO quanto a detecção de versão.

Opções adicionais

-v **(aumenta o nível de verbosidade)**

Aumenta o nível de verbosidade, fazendo com que o Nmap apresente mais informações sobre o progresso do scan. Portas abertas são mostradas conforme são encontradas, e estimativas de tempo para o término são fornecidas quando o Nmap acredita que um scan irá demorar mais do que alguns minutos. Use duas vezes para uma verbosidade ainda maior.

Importante: Usar mais do que duas vezes não surte nenhum efeito.

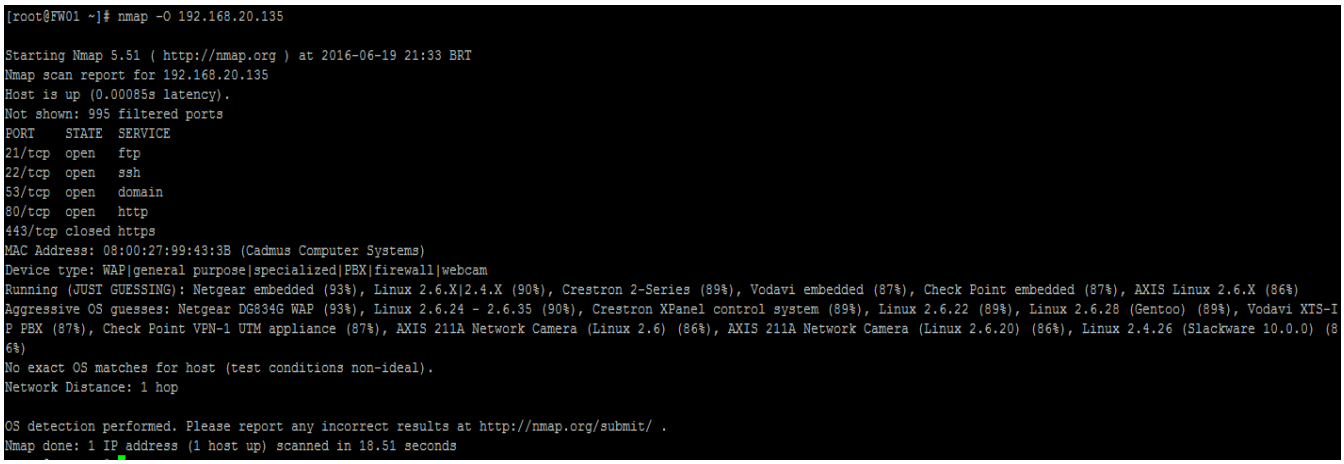
--top-ports <number>

Comando utilizado para analisar as portas mais comuns. A sintaxe do comando é sempre utilizada no fim do nmap, e inserido um número no fim do comando, que é referente a quantidade de “top ports” a ser analisado.

Exemplos

Comando para ter conhecimento do OS que estamos fazendo a análise.

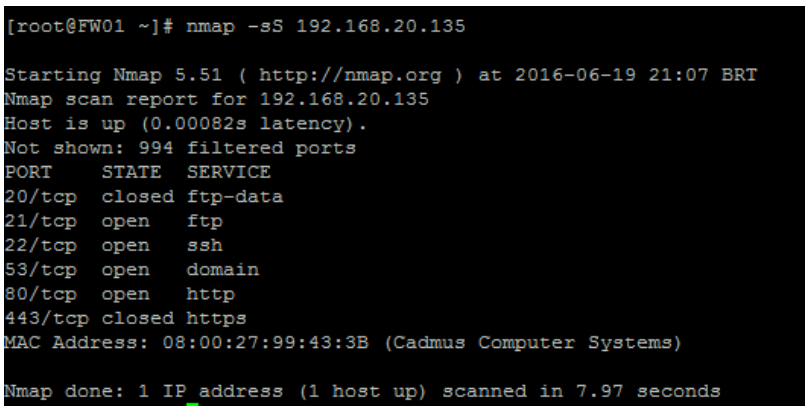
nmap -O 192.168.20.135



A figura capturada como exemplo traz algumas portas abertas, mas o principal é a verificação do SO que possivelmente está sendo executado no IP analisado. Observar “Running” que apresenta em um de seus resultados “Linux 2.6.x/2.4.x (90%)”. Que no caso é realmente o Kernel utilizado.

Comando para uma análise básica nas portas padrão TCP .

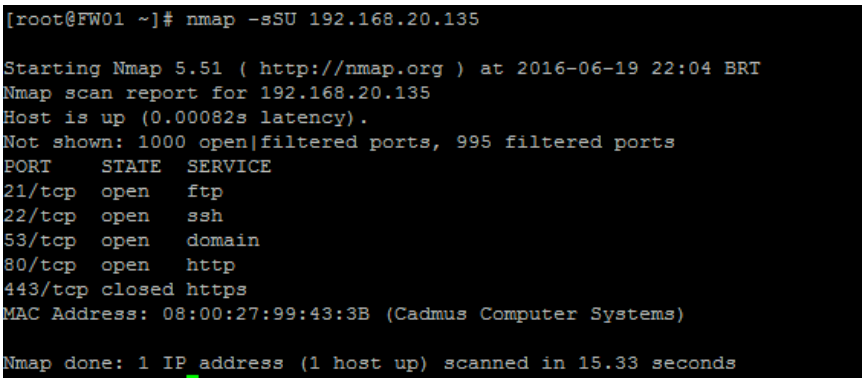
nmap -sS 192.168.20.135



Este resultado apresenta algumas portas padrões, e o resultado do estado delas, que no caso são Open e Closed, dependendo da porta.

Comando para uma análise básica nas portas padrão TCP e UDP

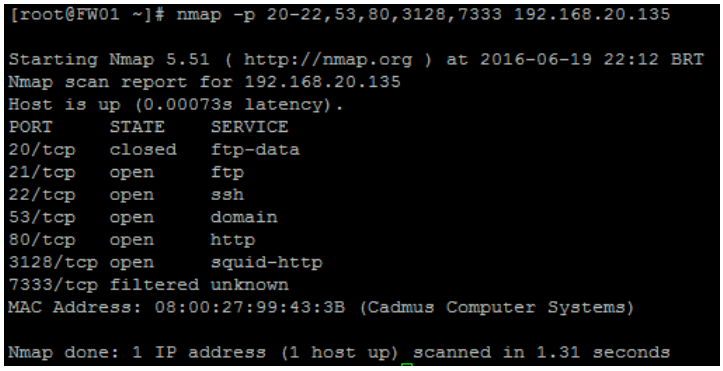
nmap -sSU 192.168.20.135



Como pode ser observado no comando utilizando o parâmetro “U”, é realizdo uma varredura em portas TCP e UDP. O resultado é como o esperado, uma vez que uma análise UDP é muito mais difícil de detectar algo, devido a forma com uma conexão UDP é feita.

Comando para uma análise em portas especificas TCP, definidas pelo usuário

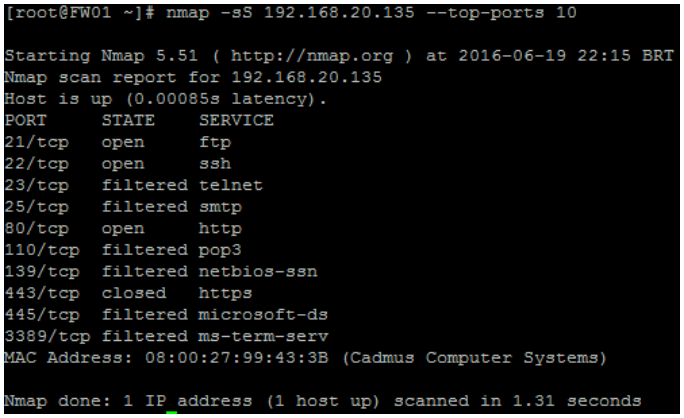
nmap -p 20-22,53,80,3128 192.168.20.135



Neste exemplo de teste foi realizado uma varredura no protocolo TCP nas portas 20,21,22,53,80,3128,7333, somente a porta 20 apresentou estar fechada. Esse comando é muito importante quando se quer buscar por alguma porta de algum serviço especifico, ou até mesmo portas diferentes dos padrões, como foi o caso da 7333 testada, que está aberta.

Comando para analisar as portas mais comuns TCP.

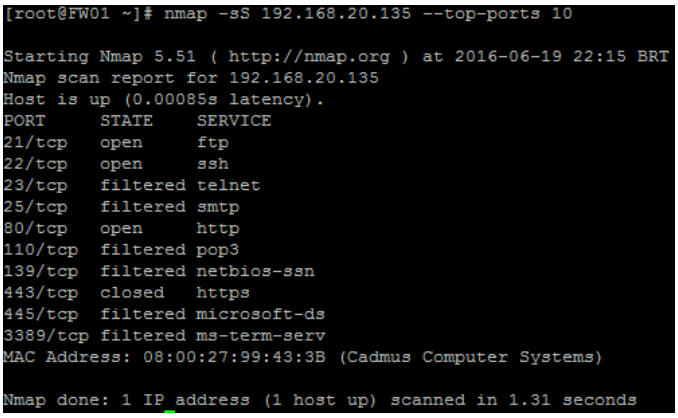
nmap -sS 192.168.20.135 --top-ports 10



Neste exemplo tem-se como resultado algumas portas filtradas, fechadas e aberta, exemplo importante para análise, pois foi utilizado o comando –sS para análise TCP, (--top-ports 10 lista as 10 portas mais comuns). Com esta análise identificam-se resultados interessantes, que mostram que as portas já antes abertas se mantiveram, portas fechadas como a 443 apareceram bem como algumas filtradas.

Esta análise é importante pois, no local testado, tem-se um servidor apache, com a porta 80 aberta e a 443 fechada, assim o resultado da 443 se apresenta como fechada, para outras portas que não possuem nenhum serviço rodando no servidor, se mostram filtrada, isso é, estas podem estar abertas ou não, mas que não foi encontrado nenhuma aplicação sendo rodada nelas.

nmap -sS 192.168.20.135 --top-ports 10



Análise linha a linha:

1 Linha: Mostra a versão do nmap a ser executada, data e hora do teste.

2 Linha: Mostra o alvo do scan, no caso é um IP único.

3 Linha: Informa que o host está ativo, e a latência até encontrá-lo.

4--14 Linha: Resultado de portas, estado da porta e o serviço que comumente é executado naquela porta.

15 Linha: Mac address do destino, interessante caso tenha interesse em realizar um ataque via *mac address.*

17 linha: Resultado da pesquisa, e o tempo para o resultado.