Varrendo portas e serviços com Nmap (Scanning + Firewall Bypass)

17 DE JUNHO DE 2015

O ***Nmap*** ("Network Mapper") é uma ferramenta de código aberto para exploração de rede e auditoria de segurança. Ela tem como função escanear rapidamente redes amplas, embora também funcione muito bem contra hosts individuais. O Nmap utiliza pacotes IP em estado bruto (raw) de maneira inovadora para determinar quais hosts estão disponíveis na rede, quais serviços (nome da aplicação e versão) os hosts oferecem, quais sistemas operacionais (e versões de SO) eles estão executando, que tipos de filtro de pacotes/firewalls estão em uso, e dezenas de outras características.

Embora o Nmap seja normalmente utilizado para auditorias de segurança, muitos administradores de sistemas e rede consideram-no útil para tarefas rotineiras tais como inventário de rede, gerenciamento de serviços de atualização agendados, e monitoramento de host ou disponibilidade de serviço. Abordaremos aqui algumas funcionalidades e técnicas as quais podemos usufruir dela como fazer varreduras nas portas, bypass/identificação de firewall, entre outros recursos.

Para obter o Nmap é necessário instala-lo (pode ser feito através da ferramenta Organon feita pela nossa equipe) ou obter através de algumas distribuições que vem instalada como o Kali Linux, Parrot Security, Backtrack, entre outras. Caso já tenha instalado na sua distro siga para o Passo 2, agora instalaremos ela.

Passo 1 Instalação

```
# apt-get install nmap
```

Passo 2 Executando comandos com Nmap

Destacaremos abaixo os comandos mais utilizados explicando cada um deles, ressaltando que isso é apenas a ponta do iceberg dessa incrível ferramenta, a qual existe milhares de outros recursos nela. Para conhecer melhor ela, acesse o site oficial ou de no terminal o comando *help* na ferramenta para obter todas as opções dela.

Como já dito podemos fazer um scan numa máquina apenas (host individual) como exemplo o IP *192.168.1.50* ou fazer na rede toda varrendo todas as máquinas contidas nela usando exemplo 192.168.1.0/24. O scan do Nmap além de fazer consulta IP, ele pode também estabelecer um varredura com DNS como exemplo de sites. Então daremos continuidade dando exemplo como

host individual, sendo o mesmo processo como em qualquer outro tipo de alvo. Segue abaixo alguns dos comandos que podemos explorar com a ferramenta:

-sP (Ping scan)

Muito utilizado para ver se a rede está no ar, ele envia pacotes ICMP “echo request” para verificar se determinado host ou rede está ativa. Hoje em dia, existem muitos filtros que rejeitam os pacotes ICMP “echo request”, então envia um pacote TCP ACK para a porta 80 (default) e caso receba RST o alvo está ativo. A outra maneira é enviar um pacote SYN e esperar um RST ou SYN-ACK.

```
# nmap –sP 192.168.1.50
```

-sR (RCP scan)

Ele trabalha como um conjunto de várias técnicas utilizadas no Nmap. Ele considera todas as portas TCP e UDP abertas e envia comandos NULL SunRPC, para determinar se realmente são portas RPC. É como se o comando “rpcinfo -p” estivesse sendo utilizado, mesmo através de um firewall (ou protegido por TCPwrappers).

```
# nmap –sR 192.168.1.50
```

-sS (TCP SYN scan)

Técnica também conhecida como “*half-open*”, pois não abre uma conexão TCP completa. É enviado um pacote SYN, como se ele fosse uma conexão real, e é aguardado uma resposta. Caso um pacote SYN-ACK seja recebido, a porta está aberta, enquanto um como resposta indica que a porta está fechada. A vantagem dessa abordagem é que poucos irão detectar esse scanning de portas.

```
# nmap –sS 192.168.1.50
```

-sT (TCP connect() scan)

É a técnica mais básica de TCP scanning. É utilizada a chamada de sistema (system call) “*connect()*” que envia um sinal as portas ativas. Caso a porta esteja aberta recebe como resposta “*connect()*”. É um dos scans mais rápidos, porém mais fácil de ser detectado.

```
# nmap –sT 192.168.1.50
```

-sU (UDP scan)

Este método é utilizado para determinar qual porta UDP está aberta em um host. A técnica consiste

em enviar um pacote UDP de0 byte para cada porta do host. Se for recebido uma mensagem ICMP “port unreachable” então a porta está fechada, senão a porta *pode* estar aberta. Para variar um pouco, a Microsoft ignorou a sugestão da RFC e com isso a varredura de máquinas Windows é muito rápida.

```
# nmap –sU 192.168.1.50
```

-sV (Version detection)

Após as portas TCP e/ou UDP serem descobertas por algum dos métodos, o nmap irá determinar qual o serviço está rodando atualmente. O arquivo nmap-service-probes é utilizado para determinar tipos de protocolos, nome da aplicação, número da versão e outros detalhes.

```
# nmap –sV 192.168.1.50
```

-p (porta)

Especifica quais portas devem ser verificadas na varredura. Por default, todas as portas entre 1 e 1024 são varridas.

```
# nmap -p 22,80 192.168.1.50 ou nmap -p U:53,111,137,T:21-25,80,139,8080
```

-P0 (Sem ping)

Não tenta pingar o host antes de iniciar a varredura. Isto permite varrer alvos que bloqueiam ICMP “echo request (ou responses)” através de firewall.

```
# nmap -P0 192.168.1.50
```

-PT (Ping com TCP)

Usa TCP “ping” para determinar se o host está ativo.

```
# nmap –PT 80 alvo
```

-R (resolução DNS para todos os alvos)

Irá resolver nomes de hosts a serem varridos.

```
# nmap -R 192.168.1.50
```

-r (aleatório)

A varredura será feita nas portas randomicamente, não seguindo uma ordem crescente.

```
# nmap -r 192.168.1.50
```

-ttl<valor>

Altera o valor do TTL (Time to Live), dessa forma dificulta a origem do pacote.

```
# nmap -ttl 55 192.168.1.50
```

-v (verbose)

Mostra tudo o que está acontecendo no scan.

```
# nmap -v 192.168.1.50
```

Passo 3 Dicas de Varredura + Identificação e Bypass de Firewall

No processo de scan do Nmap ele acaba enviando muitos pacotes para o alvo, isso acaba gerando um certo “barulho” que é a expressão que usamos para definir quando se pode alertar algum dispositivo de segurança do alvo como um firewall. Utilizando o *Wireshark* (a qual usaremos para monitorar os pacotes na rede) vemos a quantidade de pacote enviada para o alvo, lembrando que quanto menor o número de pacote menor é o barulho dele, sendo menos perceptível a um firewall.

nmap -sV = 2125 pacotes (listar portas)

nmap -O = 2130 pacotes (pega a versão do sistema operacional)

nmap -sS = 2090 pacotes (mais requisitado; envia script para o alvo)

nmap -A = 2290 pacotes (modo avançado)

nmap -D = 2090 pacotes (incrimina o IP para o ataque)

nmap -P0 = 2007 pacotes (não utiliza ping)

Lembrando que se você faz uma varredura numa determinada máquina ele gera menos barulho do

que numa host inteira. Como no caso abaixo:

```
# nmap -p135 -sV 192.165.2.103 (pegou 25 pacotes)
```

```
# nmap -p135 -sV -sS 192.165.2.103 (pegou 25 pacotes)
```

Uma boa forma de varrer um alvo driblando algum mecanismo de segurança é utilizando o comando –g a qual se passa pela porta de origem para fazer a varredura assim podemos driblar o firewall. Como no exemplo abaixo que ele varre o alvo utilizando a porta de *Origem* 53 para tentar enganar o Firewall o fazendo achar que é uma consulta DNS.

-g é a porta de Origem

-p é a porta de Destino

```
Ex: # nmap -g53 –p80 192.168.0.50
```

Nesse exemplo se passamos pela porta de origem 53 para varrer a porta 80 deixando o firewall pensar que é uma consulta DNS. Podendo também se demonstrar dessa maneira:

```
# nmap -p80 -g53 www.cienciahacker.com.br (pega a porta 53 que é da IDS e faz a varredura na 80, assim não registra como tentativa de ataque; porta 53 é IDS)
```

Em algumas rede o pacote ICMP (ping) são bloqueadas pelo firewall impossibilitando boa parte dos scans, com a opção -P0 ele faz a varredura sem usa-lo.

```
# nmap -P0 www.cienciahacker.com.br
```

Vale a pena ressaltar que antes de qualquer conexão com qualquer é porta é necessário verificar a existência de algum firewall pois pode barrar sua entrada. Com a opção -sA ele realiza essa consulta.

```
# nmap -sA -p80  www.cienciahacker.com.br
```

Fonte bibliográfica: https://nmap.org/man/pt_BR/index.html#man-description

FILTRAR FIREWALL NETWORK MAPPER NMAP SCAN SCANNING REDES VARRENDO PORTAS