C|EH™ Certified Ethical Hacker

TRAINNING EDUCATION CENTER

Curso:

(C|EH) V12

CERTIFIED ETHICAL HACKER - SECURITY IMPLEMENTATION

Progresso do curso

Módulo 1. Introdução ao Hacking Ético

Módulo 2. Footprinting e Reconhecimento

Módulo 3. Scanning de Redes

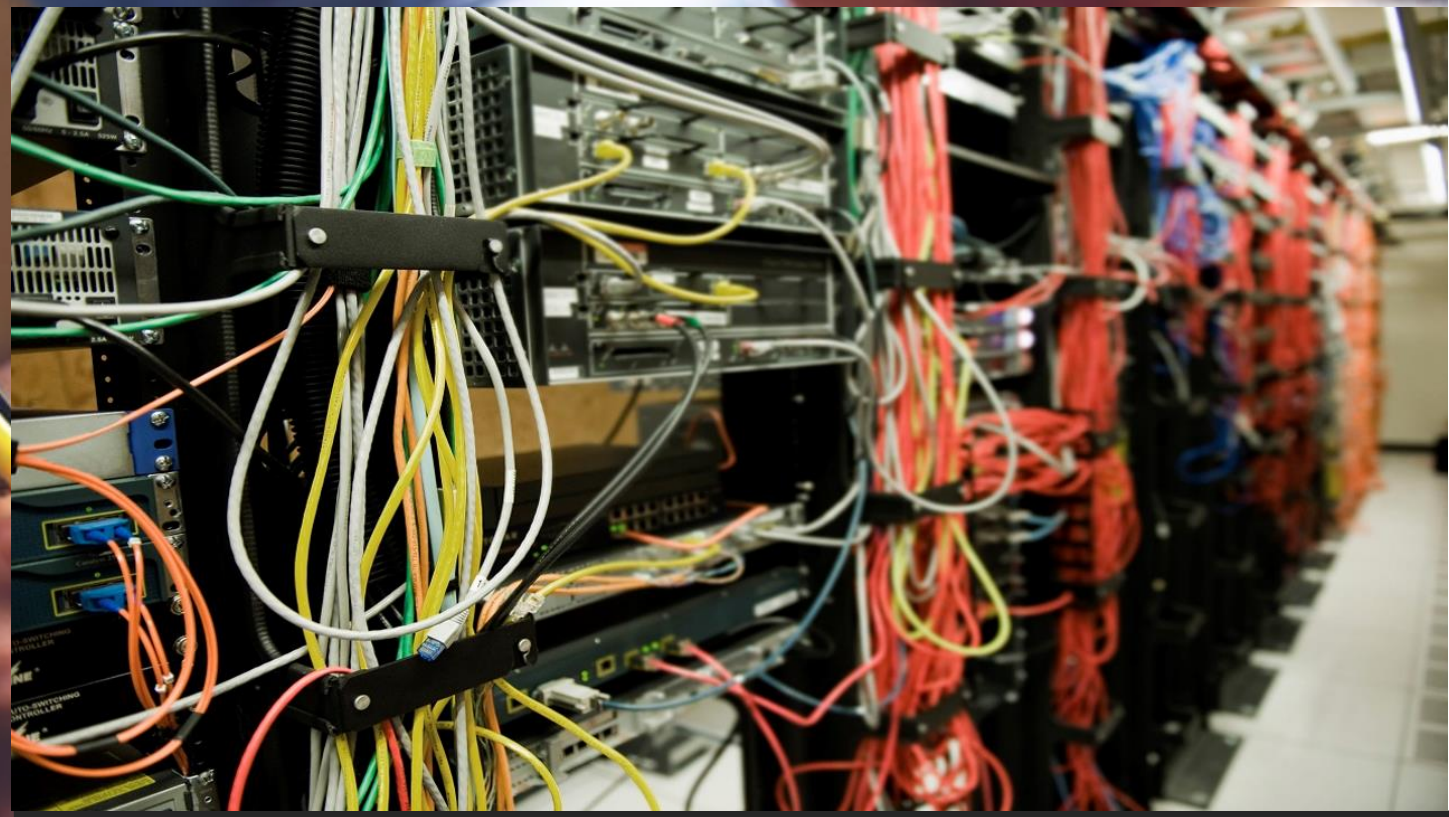
Módulo 4. Enumeração

Módulo 5. Análise de Vulnerabilidade

Conceitos sobre Scanning:

Na fase de scanning, podemos encontrar várias formas de intrusão no sistema alvo. Podemos também descobrir mais sobre o sistema de destino, tais como o sistema operacional utilizado, quais os serviços que estão em execução, e se existem ou não quaisquer lapsos de configuração no sistema de destino.

A ideia é descobrir canais de comunicação exploráveis, para descobrir o máximo de portas ativas possível, e se manter a par dos que são sensíveis ou úteis para hacking.



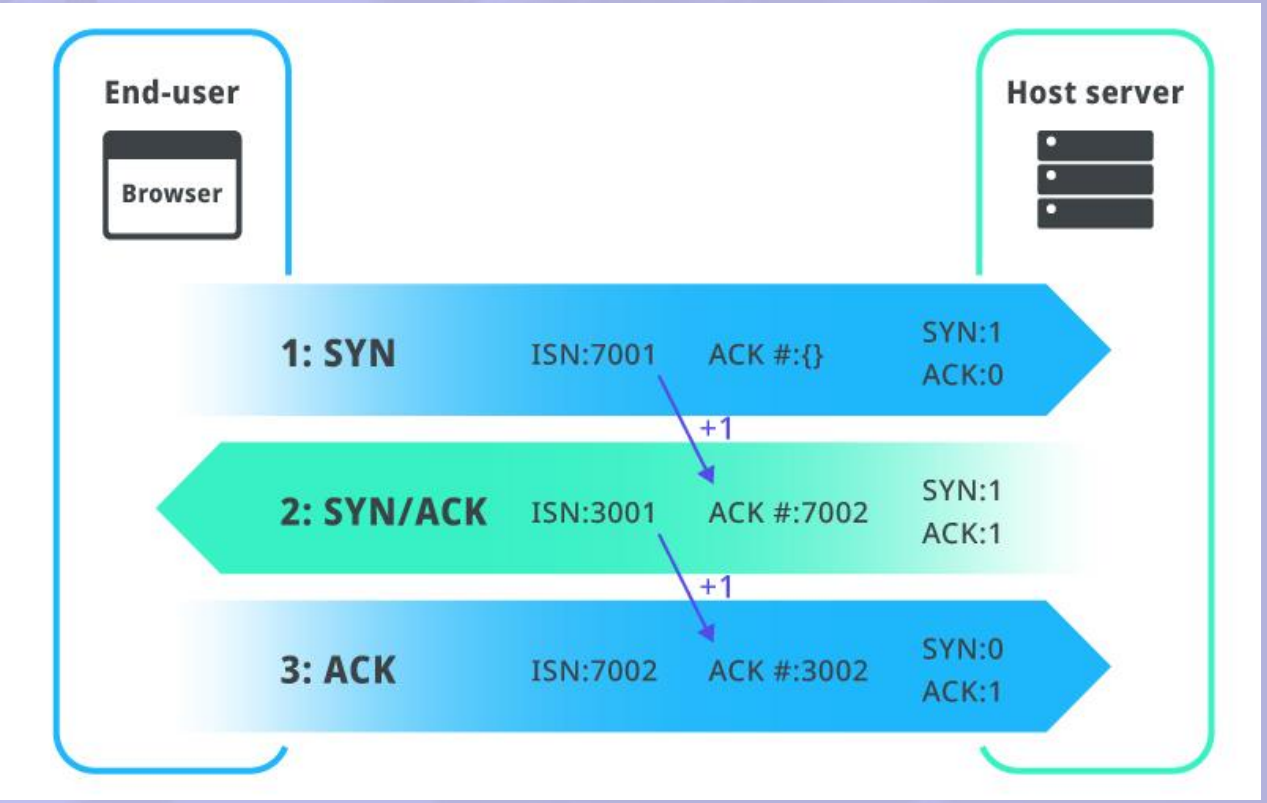
CEHv12

03 - Scanning de Redes

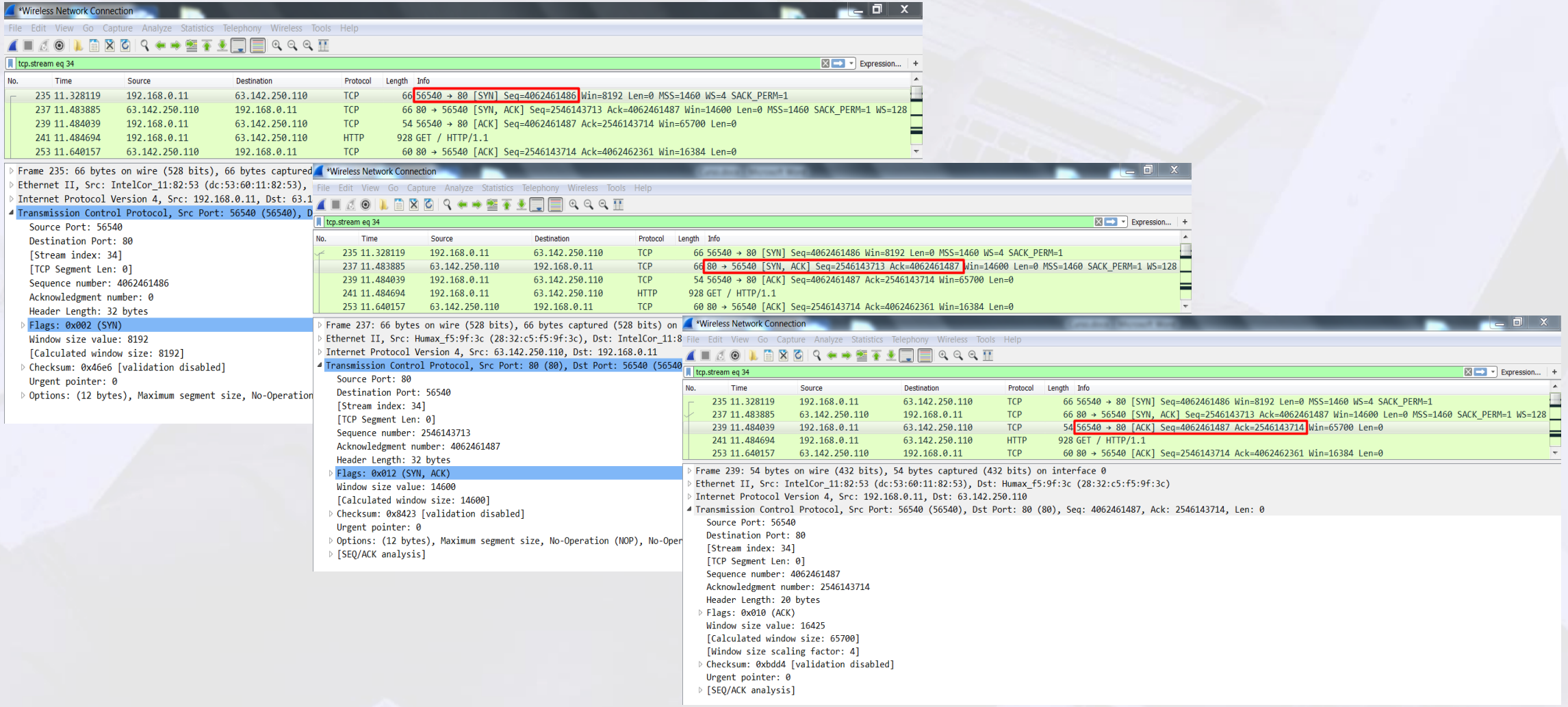
TCP Three-way handshake

- TCP ou Protocolo de Controle de Transmissão é um protocolo confiável e orientado à conexão, e com o TCP os dados podem ser entregues com precisão. Muitos aplicativos, como web (HTTP), e-mail (SMTP) e transferência de arquivos (FTP) utilizam TCP.
- Antes que o TCP transmita segmentos de dados de um dispositivo para outro na Internet, ele primeiro utilizará um handshake de 3 vias para estabelecer uma conexão e ser sincronizado. O TCP 3-way handshake é uma série de comunicação entre 2 dispositivos (por exemplo, computador e servidor) para estabelecer uma conexão de rede e garantir que os dados sejam transferidos sem erros e completos.

"Número de sequência inicial" (**ISN**) e "número de confirmação" (**ACK** #) são identificadores únicos aleatórios na forma de um número de 32 bits utilizado para marcar a sequência de pacotes de dados que um dispositivo transmitirá de um cliente (ou um servidor) e vice-versa. Isso permitirá que os dispositivos identifiquem a ordem correta dos pacotes de dados ao solicitar/enviar segmentos e também ao reformar todos os dados. Os números de sequência são apenas um dos dados incluídos durante o encapsulamento de pacotes de dados; números de porta e endereços IP também estão incluídos.



TCP Three-way handshake



Flags TCP

- **Sincronização (SYN) –** É utilizado na primeira etapa da fase de estabelecimento da conexão ou processo de handshake de 3 vias entre os dois hosts. Somente o primeiro pacote do remetente e do destinatário deve ter esse sinalizador definido. É utilizado para sincronizar o número de sequência, ou seja, para informar à outra extremidade qual número de sequência eles devem aceitar.

- **Reconhecimento (ACK) –** É utilizado para reconhecer pacotes recebidos com sucesso pelo host. O sinalizador é definido se o campo de número de confirmação contiver um número de confirmação válido. O receptor envia um ACK = 1, bem como SYN = 1, na segunda etapa do estabelecimento da conexão, para informar ao remetente que recebeu seu pacote inicial.

- **Finish (FIN) –** É utilizado para solicitar o término da conexão, ou seja, quando não há mais dados do remetente, ele solicita o término da conexão. Este é o último pacote enviado pelo remetente. Ele libera os recursos reservados e encerra a conexão normalmente.

- **Reset (RST) –** É utilizado para encerrar a conexão se o remetente sentir que algo está errado com a conexão TCP ou que a conversa não deveria existir. Ele pode ser enviado do lado do receptor quando o pacote é enviado para um host específico que não o esperava.

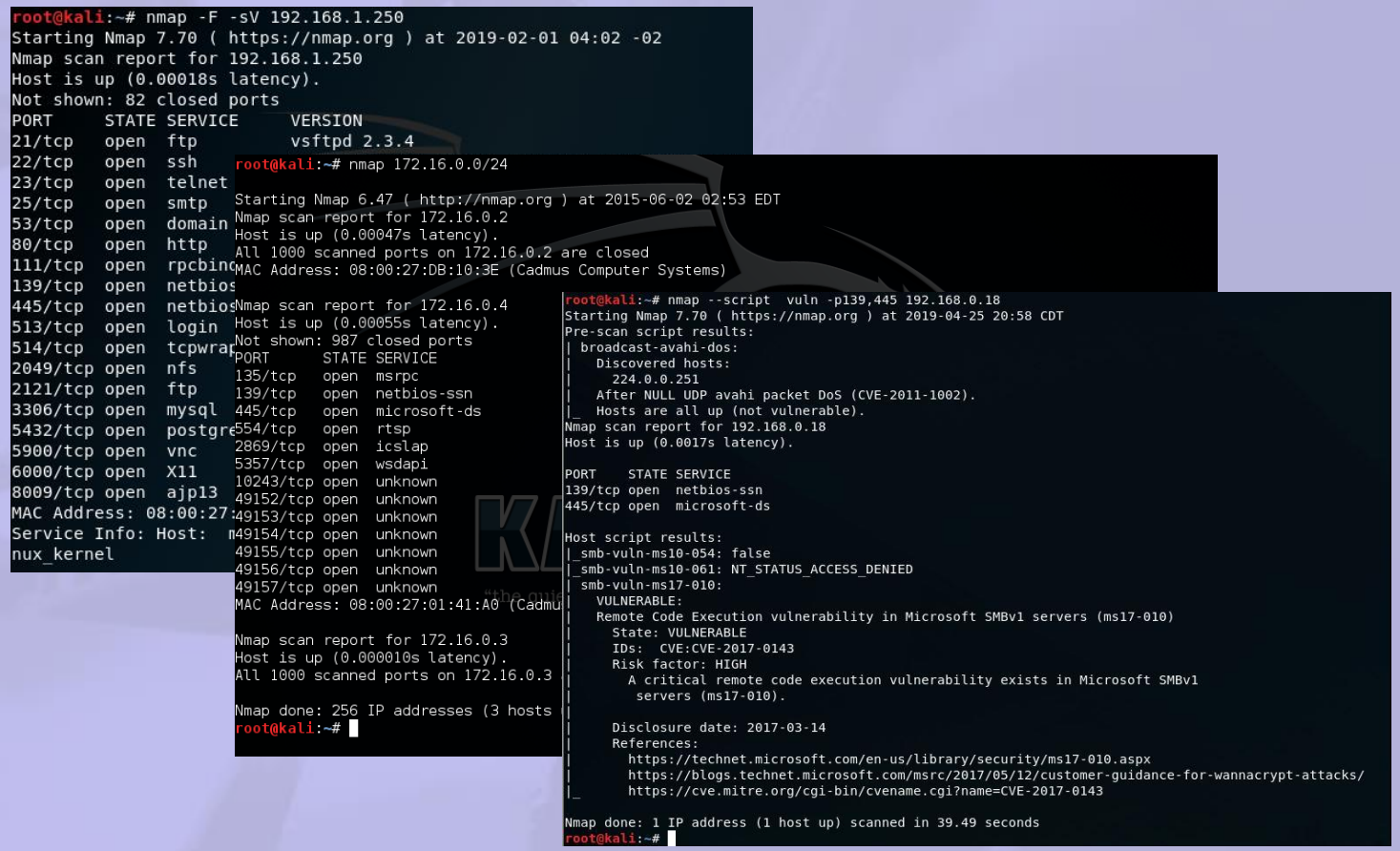
- **Push (PSH) –** A camada de transporte, por padrão, espera algum tempo até que a camada de aplicação envie dados suficientes iguais ao tamanho máximo do segmento, para que o número de pacotes transmitidos na rede seja minimizado, o que não é desejável por alguns aplicativos, como aplicativos interativos (ex: bate-papo). Da mesma forma, a camada de transporte na extremidade do receptor armazena os pacotes e transmite para a camada de aplicação se atender a determinados critérios.

- **Urgente (URG) –** Os dados dentro de um segmento com sinalizador URG = 1 são encaminhados para a camada de aplicação imediatamente, mesmo que haja mais dados a serem fornecidos à camada de aplicação. É usado para notificar o receptor para processar os pacotes urgentes antes de processar todos os outros pacotes. O destinatário será notificado quando todos os dados urgentes conhecidos forem recebidos.

Tipos de Scanning de redes

- Em um sentido tradicional, os pontos de acesso que um ladrão olha são as portas e janelas.
- Estes são geralmente os pontos de vulnerabilidade da casa por causa de sua acessibilidade relativamente fácil.
- Quando se trata de sistemas de computadores e redes, os pontos são as portas e janelas do sistema que um intruso utiliza para obter acesso.

- **Varredura de porta** – Busca por portas e serviços disponíveis.
- **Varredura de rede** - Busca por endereços range de redes e endereços IP.
- **Varredura de vulnerabilidade** - Busca por vulnerabilidades já conhecidas.



Objetivos do Scanning

Descobrir hosts ativos, endereços IP, portas abertas e serviços dos hosts em tempo real na rede.

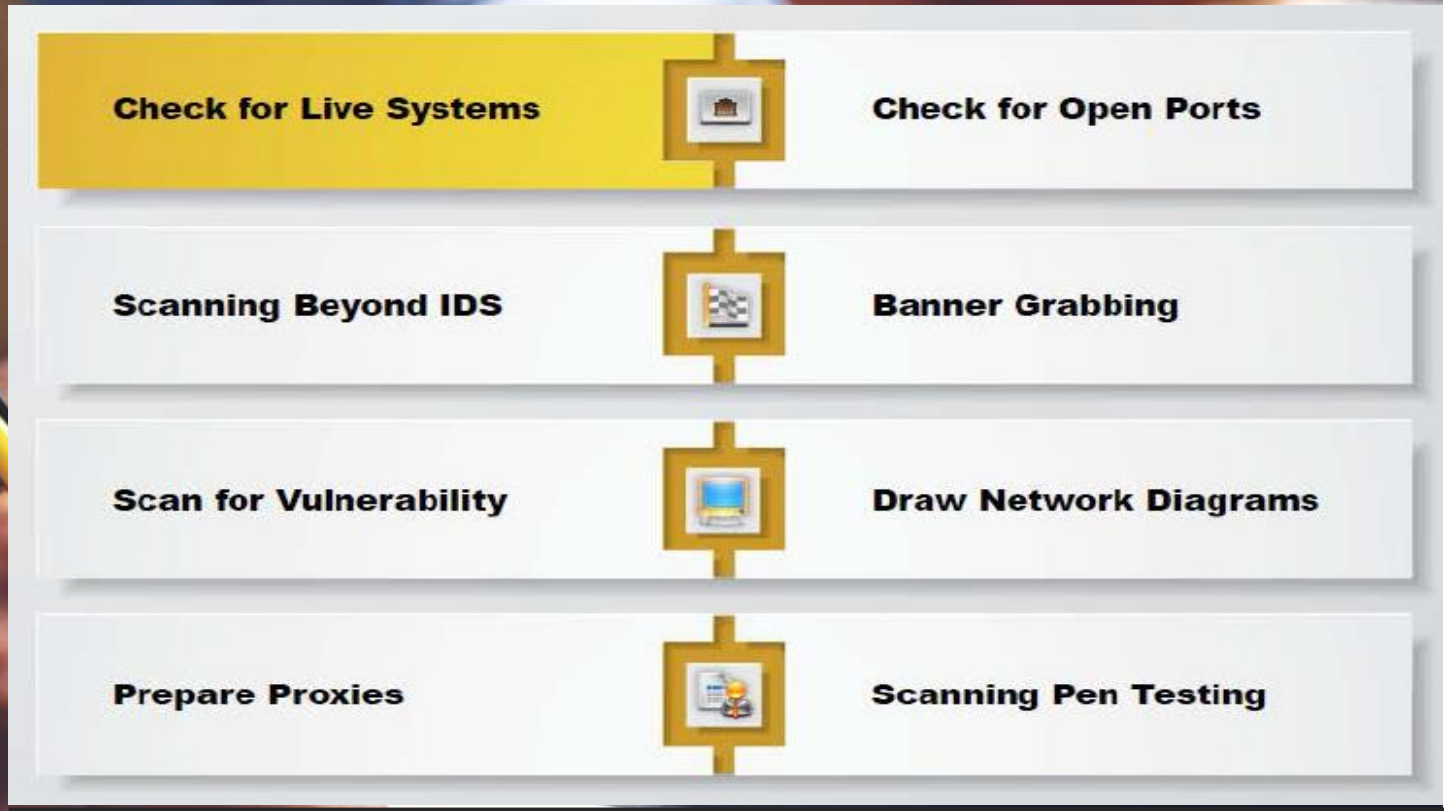
- **Descobrir as portas abertas:** Portas abertas são os melhores meios para invadir um sistema ou rede. Você pode encontrar maneiras fáceis de invadir a rede da empresa-alvo, descobrindo portas abertas em sua rede.
- **Descobrir sistemas operacionais e arquitetura do sistema do alvo:** Isto também é referido como impressões digitais. Aqui, o atacante vai tentar lançar o ataque com base nas vulnerabilidades dos sistemas operacionais.
- **Identificar as vulnerabilidades e ameaças:** vulnerabilidades e ameaças são os riscos de segurança presentes em qualquer sistema. Você pode comprometer o sistema ou rede, explorando essas vulnerabilidades e ameaças.
- **Detectar o serviço de rede de cada porta associada.**

Metodologia:

Scanning: Inclui na fase de scanning os procedimentos de verificação das portas alvo do sistema, após a análise destas portas executa-se a análise de vulnerabilidades, visando identificar quais delas podem estar comprometidas na rede.

A maioria dos servidores de rede possuem portas TCP em estado de escuta. Considere uma porta aberta se estiver em um estado de escuta, e fechada se não estiver.

O protocolo TCP utiliza o handshake de três vias (tree way handshake): SYN, SYN-ACK, ACK para comunicações, bem como alguns outros sinalizadores que incluem FIN, RST, URGENT e PUSH.



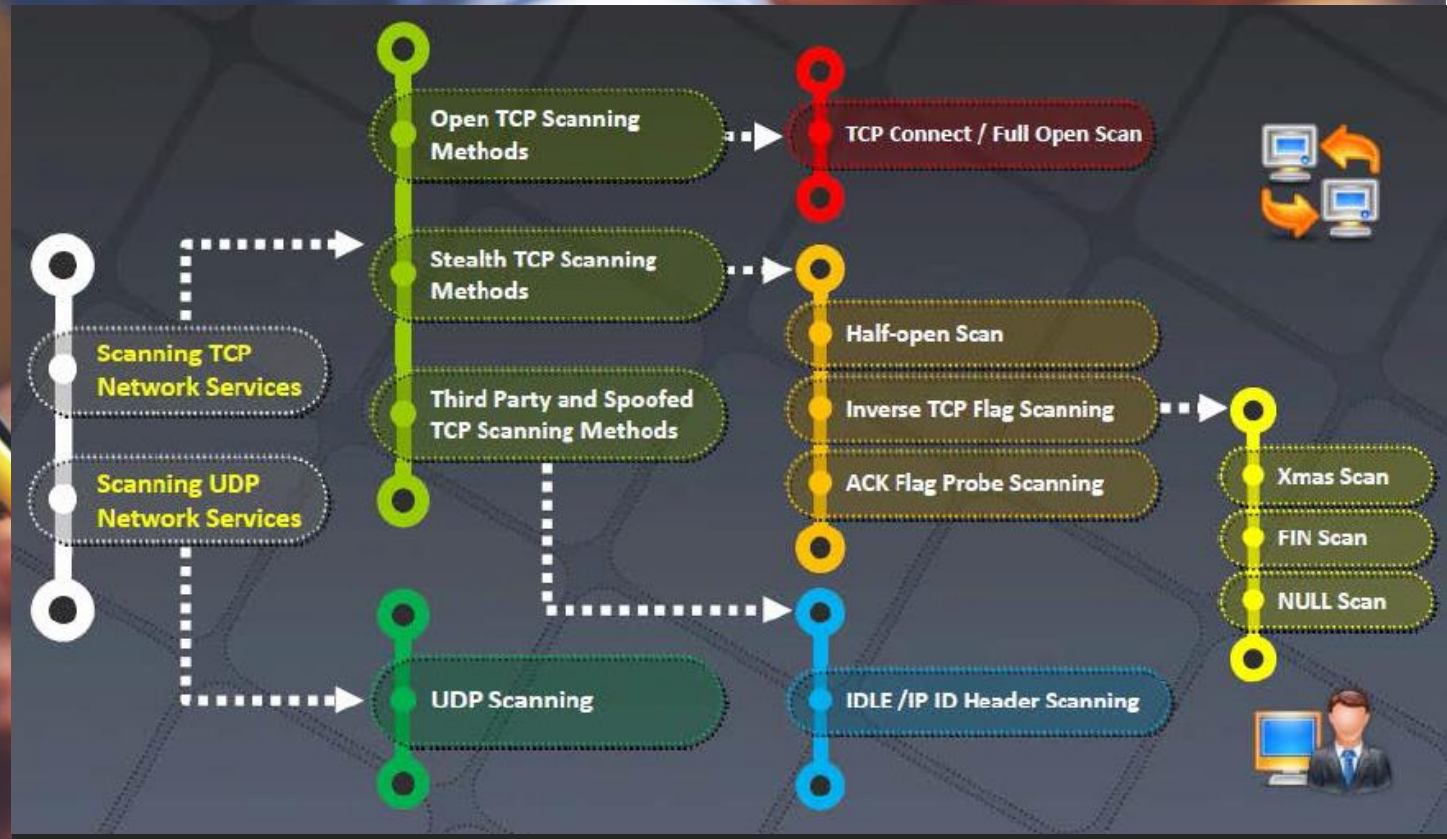
CEHv12

03 - Scanning de Redes

Ferramentas de Scanning:

NMAP: o Nmap é um scanner de rede. que permite descobrir os hosts e serviços em uma rede de computadores, criando assim um "mapa" da rede. Ele envia pacotes falsos para o host de destino e em seguida, analisar as respostas.

Hping2/Hping3: o Hping2/HPing3 é uma ferramenta de linha de comando que monta/analisa pacotes TCP/IP, envia requisições ICMP do tipo echo e suporta os protocolos TCP, UDP, ICMP e raw-IP. Ele tem um modo Traceroute, que permite enviar arquivos entre canais secretos.

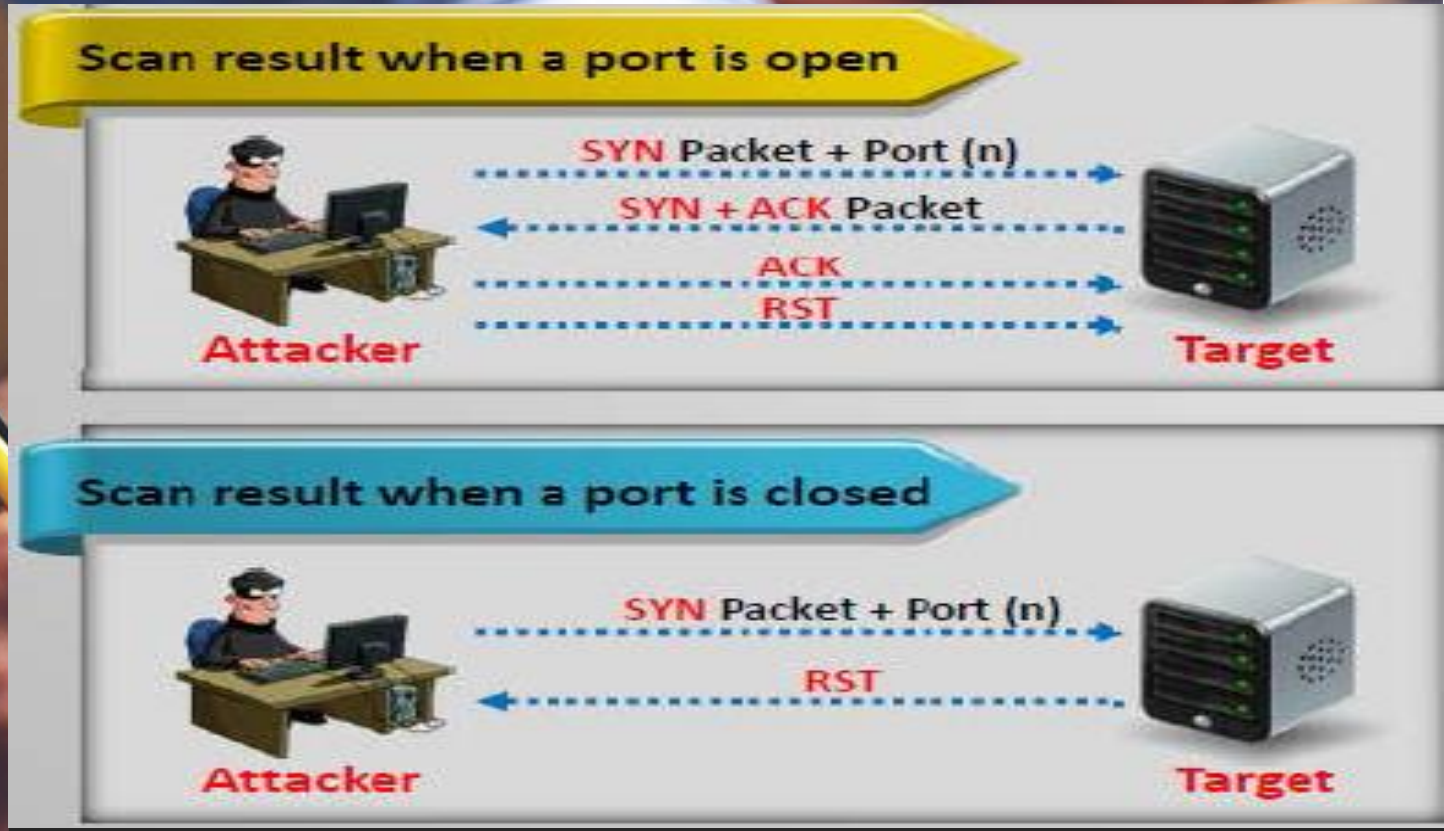


CEHv12

03 - Scanning de Redes

TCP Connect / Full Open Scan:

Esse método é a forma mais confiável de TCP scanning. A chamada connect() do sistema fornecida pelo SO é usada para abrir uma conexão com cada porta desejada com o alvo. A desvantagem desse tipo de scan é que ele é facilmente detectável pois gera muito ruído no alvo.



CEHv12

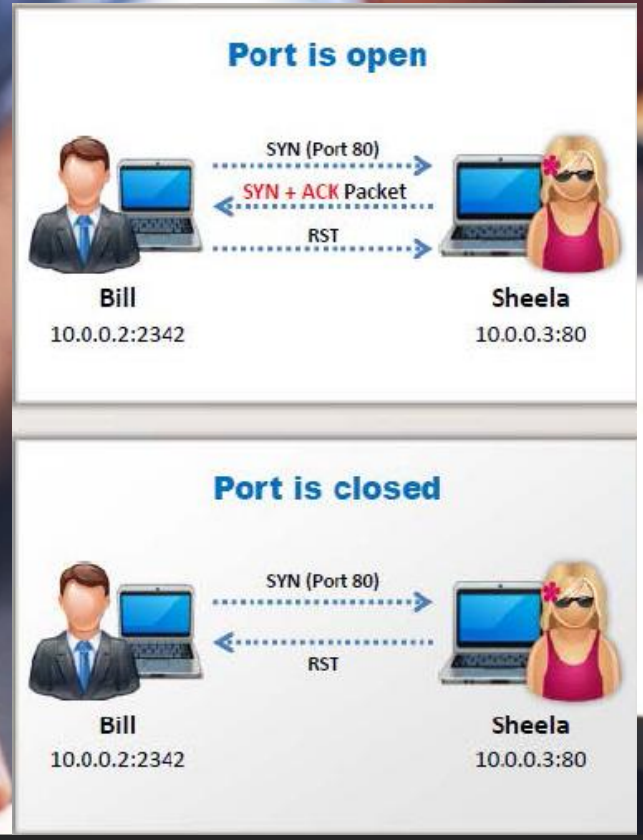
03 - Scanning de Redes

Stealth Scan (Half-open Scan):

Esta técnica é frequentemente chamada de escaneamento de porta entre aberta (half-open scanning), porque não abre uma conexão TCP completamente.

É enviado um pacote SYN, como se fosse abrir uma conexão real e então espera uma resposta.

Um SYN/ACK indica que a porta está ouvindo (aberta), enquanto um RST (reset) é indicativo de que uma porta está fechada.



CEHv12

03 - Scanning de Redes

Xmas Scan:

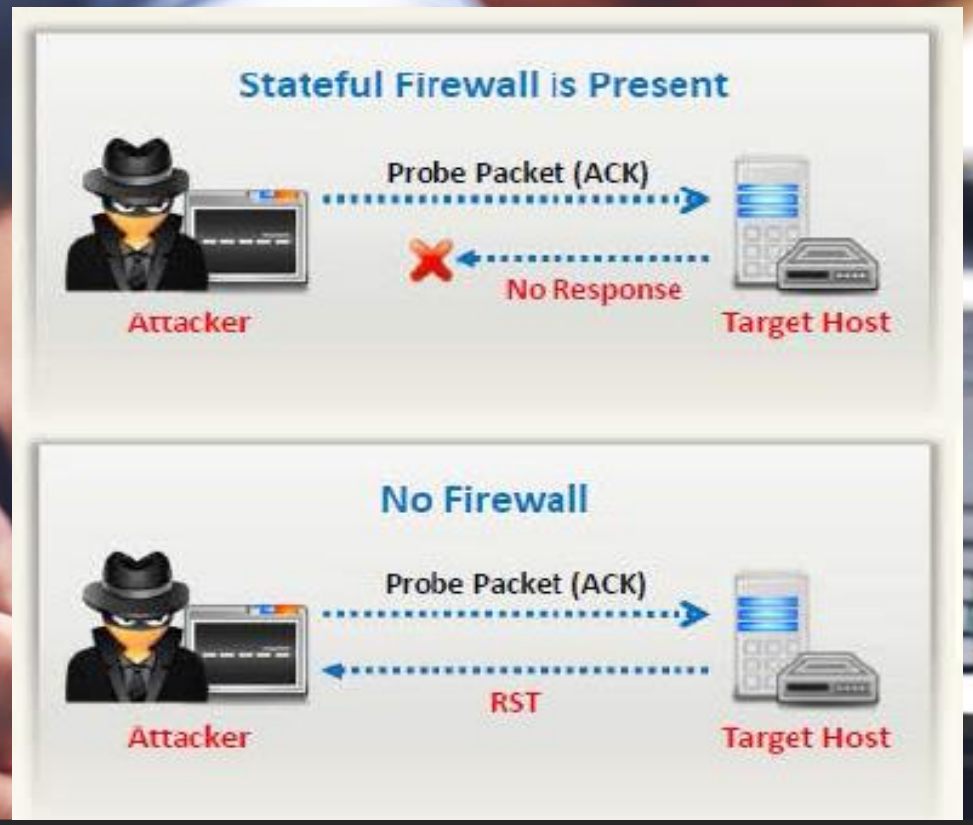
Este tipo de scan explora uma brecha sutil na RFC 793 do TCP para diferenciar entre porta aberta e fechada. Quando se faz um scan em sistemas padronizados com o texto desta RFC, qualquer pacote que não contenha os bits SYN, RST, ou ACK irá resultar em um RST como resposta se a porta estiver fechada, e nenhuma resposta se a porta estiver aberta.

CEHv12

03 - Scanning de Redes

ACK Flag Scanning:

Esse scan é diferente dos outros discutidos até agora pelo fato de que ele nunca determina se uma porta está aberta. Ele é utilizado para mapear conjuntos de regras do firewall, determinando se eles são orientados à conexão ou não e quais portas estão filtradas.

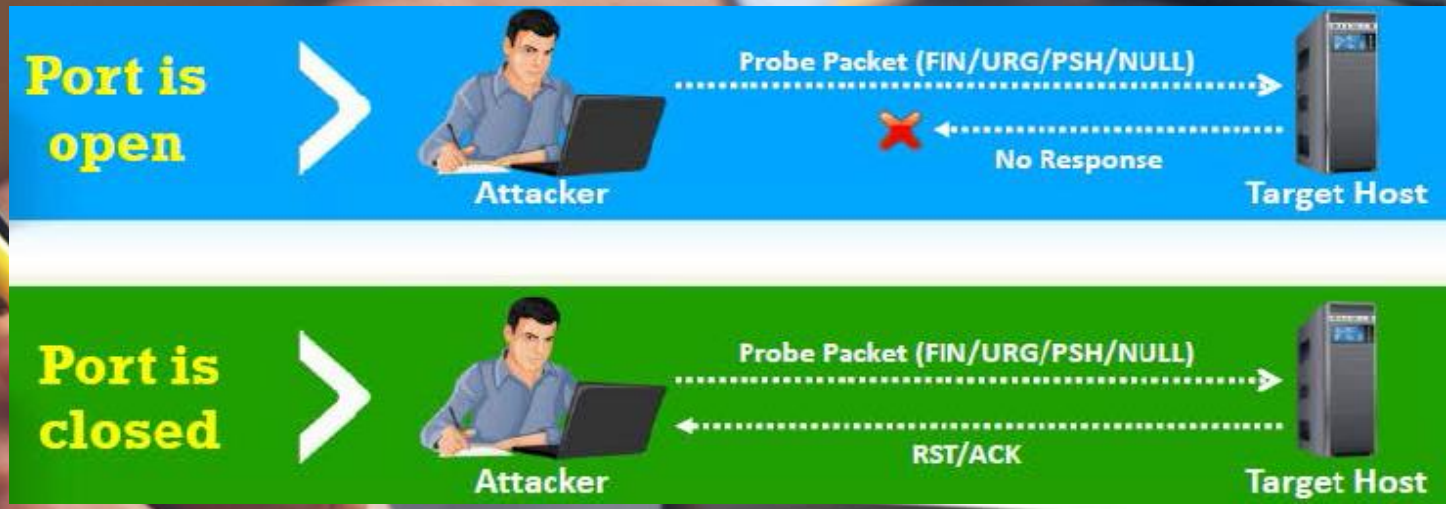


CEHv12

03 - Scanning de Redes

Inverse TCP Flag Scanning:

Atacantes enviam os pacotes TCP habilitando várias flags TCP (FIN, URG, PSH) ou sem nenhuma flag. Quando a porta está aberta, o atacante não recebe qualquer resposta do host, enquanto que, quando a porta está fechada, ele recebe um RST/ACK do host de destino.



CEHv12

03 - Scanning de Redes

UDP Scanning:

Embora os serviços mais populares na Internet trafeguem sobre o protocolo TCP, os serviços UDP são amplamente difundidos. O DNS, o SNMP, e o DHCP são três dos mais comuns. Pelo fato do escaneamento UDP ser normalmente mais lento e mais difícil que o TCP, alguns auditores de segurança ignoram essas portas. Isso é um erro, pois serviços UDP passíveis de exploração são bastante comuns e invasores certamente não ignoram o protocolo inteiro.

CEHv12

03 - Scanning de Redes

Modus Operandi de Scanning

- Sistema Operacional: kernel, serviços do sistema, serviços de comunicação (rede) e aplicações dos usuários, que podem se utilizar de serviços.
- A forma de identificação de um ponto de acesso de serviço de rede é a porta de protocolo TCP/IP.
- Sockets TCP/IP = (IP, portas)
- A porta é a unidade que permite identificar o tráfego de dados destinado a diversas aplicações.
- A identificação única de um processo acessando os serviços de rede TCP/IP é o socket TCP/IP , formado pelo par IP da máquina e a porta utilizada para acessar um serviço de rede utilizado por uma aplicação.

1. Time to Live
2. Detecção de hosts ativos
3. Detecção do sistema operacional
4. Detecção de portas abertas
5. Detecção de versão dos serviços
6. Evasão de Firewall / IDS
7. Banner Grabbing
8. Diagrama de rede
9. Utilização de Proxies
10. Scanner de vulnerabilidades

Time to Live (TTL)

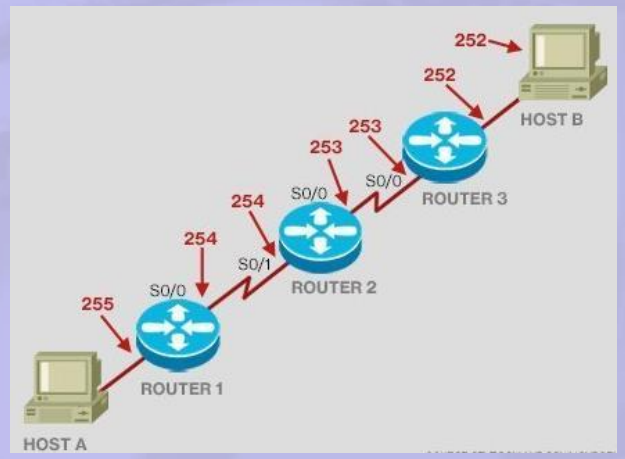
- O Time to Live (TTL) se refere à quantidade de tempo ou de "saltos" que se define que um pacote deve existir dentro de uma rede antes de ser descartado por um roteador. O TTL também é usado em outros contextos, incluindo armazenamento da CDN em cache e armazenamento de DNS em cache.

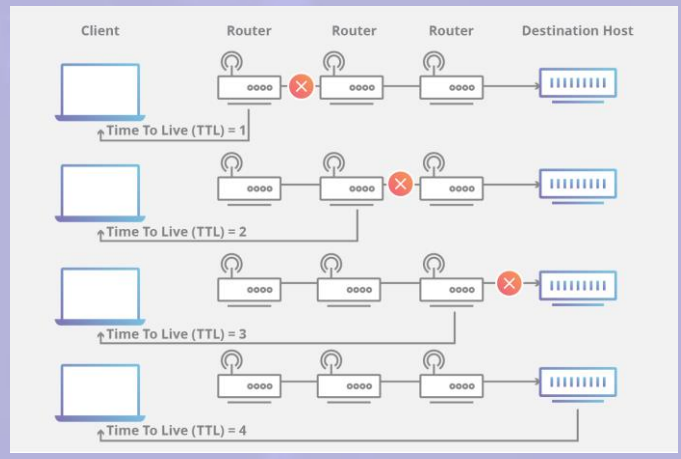
OS	TTL	Window size (bytes)
Linux 2.4 and 2.6	64	5,840
Google customized Linux	64	5,720
Linux kernel 2.2	64	32,120
FreeBSD	64	65,535
OpenBSD, AIX 4.3	64	16,384
Windows 2000	128	16,384
Windows XP	128	65,535
Windows 7, Vista, and Server 8	128	8,192
Cisco Router IOS 12.4	255	4,128
Solaris 7	255	8,760
MAC	64	65,535

1 **Time to Live**

- Quando um pacote de informações é criado e enviado pela internet, há o risco de que ele continue a passar de um roteador para outro indefinidamente. Para mitigar essa possibilidade, os pacotes são desenvolvidos com um prazo de validade chamado tempo até entrar no ar ou **limite de salto**. O pacote TTL também pode ser útil para determinar há quanto tempo um pacote está em circulação e permitir que o remetente receba informações sobre o caminho percorrido por um pacote na internet.

- Cada pacote tem um local no qual armazena um valor numérico que determina quanto tempo mais ele deve continuar a se mover pela rede. Cada vez que um roteador recebe um pacote, ele subtrai "um" da contagem do TTL e depois o passa para o próximo local de rede. Se em um determinado ponto, a contagem do TTL for igual a zero depois da subtração, o roteador descartará o pacote e enviará uma mensagem ICMP de volta ao host de origem.





Detecção de hosts ativos

- Um dos primeiros passos em qualquer missão de reconhecimento de uma rede é reduzir um conjunto (às vezes enorme) de faixas de endereços IP, em uma lista de hosts ativos e interessantes.

```
┌──(root㉿kali)-[~]
└─# nmap -sn 192.168.0.0/24
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-22 17:12 -03
Nmap scan report for dlinkrouter.Dlink (192.168.0.1)
Host is up (0.020s latency).
MAC Address: 34:0A:33:1C:3D:FB (D-Link International)
Nmap scan report for TuxGTR.Dlink (192.168.0.143)
Host is up (0.00080s latency).
MAC Address: 4C:44:5B:F7:D2:6C (Intel Corporate)
Nmap scan report for kali.Dlink (192.168.0.190)
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 4.66 seconds
```

2 Detecção de hosts ativos

Selecionar um endereço ou vários endereços:

- **Único IP:** `nmap 192.168.1.1`
- **Mais de um IP:** `nmap 192.168.1.1 192.168.1.2 192.168.1.3`
- **Mais de um IP:** `nmap 192.168.1.1,2,3`
- **Mesmo range de IP's:** `nmap 192.168.1.1-50`
- **Mesma sub-rede:** `nmap 192.168.1.0/24`
- **Caractere coringa:** `nmap 192.168.1.*`
- **A partir de lista:** `nmap -iL alvos.txt`

Excluindo hosts ou sub-redes:

- `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`

Excluir em uma lista:

- `nmap -iL /tmp/scanlist.txt --excludefile /tmp/exclude.txt`

Detecção do sistema operacional

- O funcionamento interno da detecção do sistema operacional é bastante complexo, mas é um dos recursos mais fáceis de utilizar.
- Observe que a detecção do SO requer que o Nmap seja executado como um usuário privilegiado.
- O modo de detecção do sistema operacional é muito poderoso devido à comunidade de usuários do Nmap, que gentilmente contribui com impressões digitais que identificam uma ampla variedade de sistemas, incluindo roteadores residenciais, webcams IP, sistemas operacionais e muitos outros dispositivos de hardware.

3 **Detecção do sistema operacional**

Para habilitar a detecção do SO, adicione a opção Nmap -O ao seu comando scan.

```
#nmap -O <target>
```

Podemos aumentar ou diminuir a intensidade durante a detecção de versão alterando o nível de intensidade da varredura com o argumento --version-intensity [0-9], conforme a seguir:

```
# nmap -sV --version-intensity 9 <target>
```

Caso a detecção do SO falhe, podemos utilizar o argumento --osscan-guess para forçar o Nmap a adivinhar o sistema operacional.

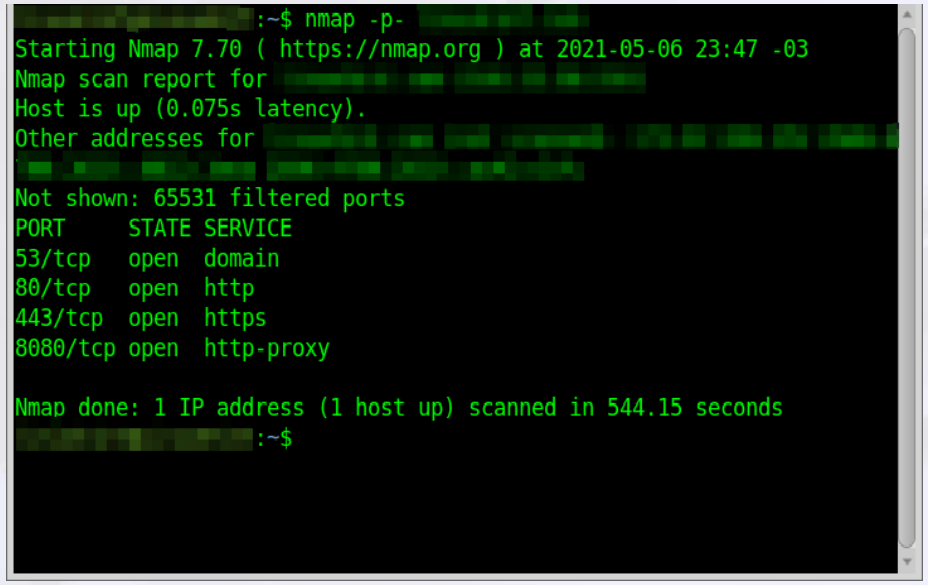
```
#nmap -O --osscan-guess <target>
```

Temos a possibilidade de utilizar o script nmap smb-os-discovery.nse, ele deve, na maioria das vezes, fornecer as respostas certas. Não funciona em algumas versões do Windows 10.

```
nmap --script smb-os-discovery.nse -p445 <target>
```

Detecção de portas abertas

- O Nmap oferece opções para especificar quais portas serão escaneadas e se a ordem de escaneamento é aleatória ou sequencial.



4 **Detecção de portas abertas**

Selecionar uma porta ou intervalo de portas

- Porta única: `-p 53`
- Portas específicas: `-p 23, 53, 80, 445`
- Intervalo de portas: `-p 1-1024`
- Intervalo + portas específicas: `-p 1-1024, 3389`
- Todas as 65.535 portas: `-p-`
- 100 portas mais populares: `-F`
- Especifique as portas que NÃO devem ser verificadas: `--exclude-ports`
- Todas as portas iguais ou inferiores a um número: `-p [-1024]`
- Todas as portas iguais ou maiores que um número: `-p [1024-]`
- Não utiliza as portas de forma aleatória: `-r`

Exemplo: `nmap -sS -O -p 1-1024, 1701, 3306, 3389 192.168.1.10-100`

Detecção de versão dos serviços

- O Nmap ao escanear um host poderá dizer que as portas 25/tcp, 80/tcp e 53/udp estão abertas. Utilizando o ***nmap-services,*** banco de dados de cerca de 2.200 serviços conhecidos, O Nmap relataria que essas portas provavelmente correspondem a um servidor de correio (SMTP), servidor web (HTTP) e servidor de nomes (DNS), respectivamente.
- Depois que as portas TCP e/ou UDP são descobertas utilizando um dos outros métodos de verificação(***nmap-service-probe***), a detecção de versão interroga essas portas para determinar mais sobre o que está realmente em execução.

Detecção de versão do serviço

- Versão: `-sV`
- Verificação do Windows: `-sW`

```
root@kali:~# nmap -sV -O 192.168.56.102

Starting Nmap 6.47 ( http://nmap.org ) at 2015-06-09 21:43 CDT
Nmap scan report for 192.168.56.102
Host is up (0.00026s latency).
Not shown: 991 closed ports
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 5.3p1 Debian 3ubuntu4 (Ubuntu Linux; protocol 2
80/tcp   open  http        Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-
1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/...)
139/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
143/tcp  open  imap        Courier Imapd (released 2008)
443/tcp  open  ssl/http    Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-
1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/...)
445/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
5001/tcp open  ovm-manager Oracle VM Manager
8080/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
8081/tcp open  http        Jetty 6.1.25
MAC Address: 08:00:27:3F:C5:C4 (Cadmus Computer Systems)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.17 - 2.6.36
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at http://
Nmap done: 1 IP address (1 host up) scanned in 14.14 seconds
```

Evasão de Firewall / IDS

- Há diversas formas de realizarmos Bypass em sistemas de defesa como Firewalls, IDS e IPS.
- Exemplo de comandos:
- `nmap -g 53 [ip do alvo]`
- `nmap -T 2 [ip do alvo]`
- `nmap --max-rate 30 [ip do alvo]`
- `nmap -D 172.16.20.20,10.0.0.20,192.168.0.20 [ip do alvo]`
- `nmap -D RND:20 [ip do alvo]`
- `nmap -g 53 -T 2 -D RND:30 [ip do alvo]`
- `nmap -f 192.168.1.12`

-g: Com o parâmetro -g podemos determinar portas de saída da máquina atacante, ou seja, se o alvo não bloquear portas como 80,443,53 poderemos bypassar um firewall.

-T: Com o parâmetro -T, conseguimos alterar o tempo em que o nmap fará consultas na máquina alvo. Temos de 0 a 5 como opção, podemos por exemplo, evitar a detecção em um IDS botando por exemplo -T 2 ou -T 1. Esses parâmetros farão que o Nmap mande requisições em tempos maiores. Como ponto negativo é que seu scan irá demorar mais tempo para ser concluído.

--max-rate: Com este parâmetro conseguimos determinar a quantidade de pacotes por segundo que serão enviados. Assim como o -T podemos reduzir a quantidade de pacotes que serão enviados. Se botarmos por exemplo --max-rate 30, será reduzida drasticamente a varredura, podem bypassar um IDS.

-D: Com o parâmetro -D conseguimos fazer com que o scan engane o alvo, como se a requisição viesse de outro IP, ou seja, podemos inserir -D 192.168.0.10,10.10.10.20,172.16.200,30, assim podemos enganar um IDP/IPS devido a vir requisição de outros IPS. Uma forma muito útil é executar com -D RND: 20 neste caso o Nmap utilizará randomicamente 20 outros ips para fazer a requisição, assim dificultará a análise do IDS/IPS.

Evasão de Firewall / IDS

- Há diversas formas de realizarmos Bypass em sistemas de defesa como Firewalls, IDS e IPS.
- Exemplo de comandos:
- `nmap -g 53 [ip do alvo]`
- `nmap -T 2 [ip do alvo]`
- `nmap --max-rate 30 [ip do alvo]`
- `nmap -D 172.16.20.20,10.0.0.20,192.168.0.20 [ip do alvo]`
- `nmap -D RND:20 [ip do alvo]`
- `nmap -g 53 -T 2 -D RND:30 [ip do alvo]`
- `nmap -f 192.168.1.12`

6 Evasão de Firewall / IDS

O comando -f induz nossa varredura a implantar pacotes IP fragmentados em pequenos pedaços. Os pacotes fragmentados consistem em enviar vários pacotes minúsculos em vez de um pacote de tamanho normal.

```
nmap -f 192.168.1.12
```

O comando nmap --mtu nos permite especificar nosso próprio tamanho de deslocamento. Lembre-se de que o tamanho do deslocamento deve ser um múltiplo de 8.

O Nmap está dando a opção ao usuário de definir um MTU (Maximum Transmission Unit) específico para o pacote.

É semelhante à técnica de fragmentação de pacotes.

Durante a varredura, o Nmap criará pacotes com um tamanho baseado no número que forneceremos.

Neste exemplo, demos o número 24, então o Nmap criará pacotes de 24 bytes, causando confusão no firewall.

Tenha em mente que o número MTU deve ser um múltiplo de 8 (8, 16, 24, 32, etc.).

```
nmap --mtu 24 192.168.1.12
```

Evasão de Firewall / IDS

- Há diversas formas de realizarmos Bypass em sistemas de defesa como Firewalls, IDS e IPS.
- Exemplo de comandos:
- `nmap -g 53 [ip do alvo]`
- `nmap -T 2 [ip do alvo]`
- `nmap --max-rate 30 [ip do alvo]`
- `nmap -D 172.16.20.20,10.0.0.20,192.168.0.20 [ip do alvo]`
- `nmap -D RND:20 [ip do alvo]`
- `nmap -g 53 -T 2 -D RND:30 [ip do alvo]`
- `nmap -f 192.168.1.12`

6 Evasão de Firewall / IDS

O comando Badsum induz a implantação de uma soma de verificação TCP/UDP/SCTP inválida para pacotes transmitidos ao nosso destino. Como praticamente toda pilha de IP de host descartaria corretamente os pacotes, cada resposta aceita possivelmente se origina de um firewall ou sistema de detecção de intrusão que não estava preocupado em confirmar a soma de verificação. Além disso, tentamos utilizar alguns scripts do Nmap NSE como "firewall-bypass", porém os resultados do uso desse script podem ser um falso positivo com uma porcentagem alta.

```
nmap --badsum 192.168.1.12
```

```
nmap -sS -T2 192.168.1.12 --script firewall-bypass
```

Banner Grabbing

- Na fase de coleta de informações do teste de invasão, a captura de banner é muito útil, pois ajuda a ter conhecimento das versões de software do alvo. Posteriormente, isso servirá como ponto de partida ao procurar vulnerabilidades no sistema de destino.
- Existem várias ferramentas que podemos utilizar para captura de banner durante o estágio de reconhecimento, dependendo do tipo de captura de banner que você deseja utilizar. Ou seja, captura de banner ativa ou passiva.

7 **Banner Grabbing**

Captura de Banner Ativa – O invasor cria ou modifica seus próprios pacotes e os envia para o servidor host remoto e analisa os dados de resposta para obter as informações do sistema operacional e os serviços executados com suas versões.
Captura de Banner Passiva – O invasor coleta dados sobre nosso alvo utilizando informações disponíveis publicamente.

Ferramentas:

```
whatweb <URL do site>
curl -s -I <URL do site>
wget -q -S <URL do site>
nikto -h <URL do site>
telnet  <alvo> <porta>
nc <alvo> <porta>
nmap -sV  <alvo>
nmap -p 22 --script=banner <alvo>
dmitry -pb <alvo>
Netcraft
Wappalyzer
```

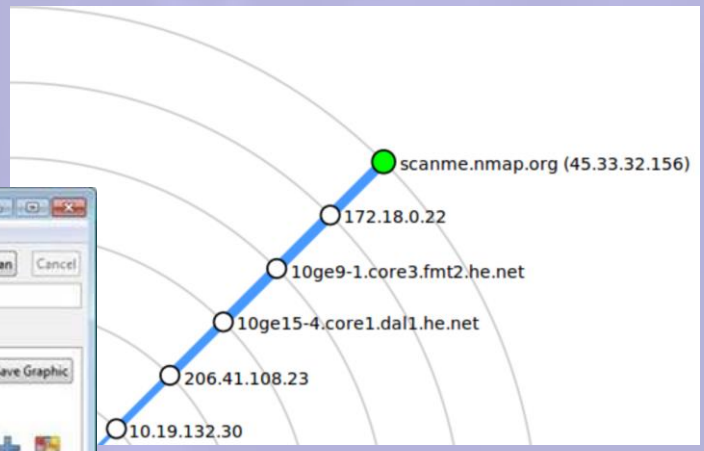
Diagrama de rede

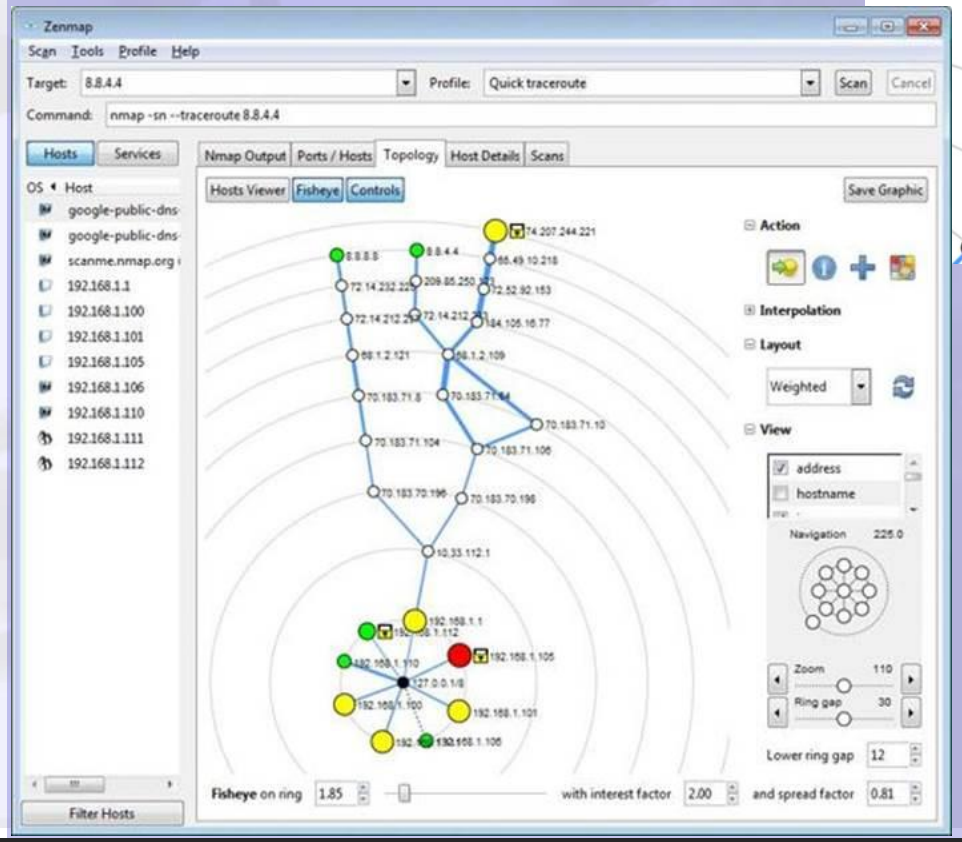
- A guia de topologia do Zenmap permite que os usuários obtenham uma representação gráfica da rede. Os diagramas de rede são utilizados para várias tarefas em TI, e podemos evitar ter que desenhar a topologia com ferramentas de terceiros exportando o gráfico de topologia do Nmap.

8

Diagrama de rede

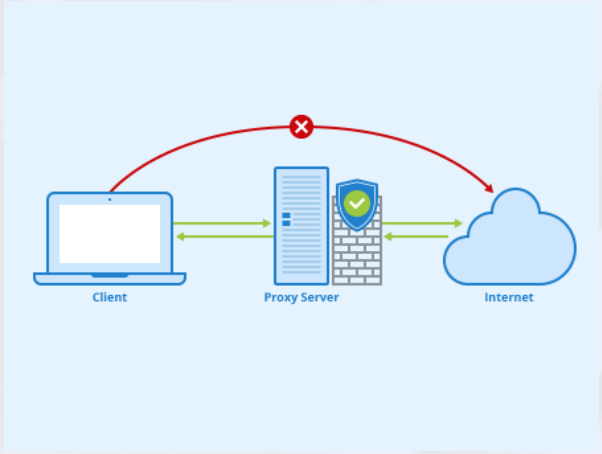
```
# nmap -sV --traceroute scanme.nmap.org
```





Utilização de Proxies

- Quando se utiliza um proxy, é como se estivéssemos utilizando um outro computador conectado à internet que passa a servir como intermediário entre você e o servidor que deseja acessar. Desse modo, o acesso é feito através do endereço de IP do servidor proxy, e não através do seu próprio.

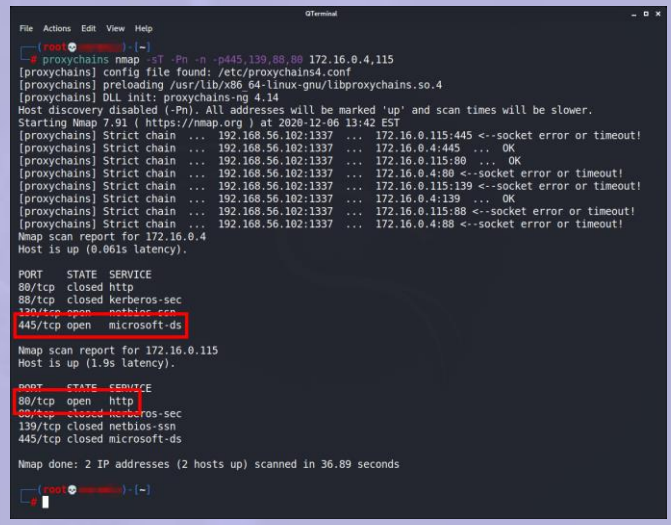


9

Utilização de Proxies

- Anônimidade com TOR, Proxychains e Privoxy

```
# apt-get install tor privoxy proxychains
```



- Configurar o privoxy

```
# vim /etc/privoxy/config
```

- No fim do arquivo config, inserir as linhas:

```
forward-socks5 / 127.0.0.1:9050 .

forward-socks4 / 127.0.0.1:9050 .

forward-socks4a / 127.0.0.1:9050 .
```

Detalhe: Depois da porta 9050, espaço em branco e ponto.

- Executar:

```
# privoxy /etc/privoxy/config

#proxychains nmap -sT -PN -n -sV -p21,25,80 10.0.2.15
```

Scanner de vulnerabilidades

- Utilizando Scripts de Análise de Vulnerabilidades do NMAP.
- O parâmetro é bem simples de ser utilizado através do script com o parâmetro `--script`.
- `nmap -p445 --script smb-vuln-ms17-010`

```
kali@kali:~$ nmap -p445 --script smb-vuln-ms17-010 192.168.171.129
Starting Nmap 7.80 ( https://nmap.org ) at 2020-03-08 11:46 EDT
Nmap scan report for 192.168.171.129
Host is up (0.0012s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|       servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143

Nmap done: 1 IP address (1 host up) scanned in 0.22 seconds
kali@kali:~$
```

10 **Scanner de vulnerabilidades**

Utilizando o NMAP para encontrar vulnerabilidades:

```
# nmap -sS -sC -Pn --script vuln scanme.nmap.org
```

Utilizando o NMAP para buscar exploits:

```
# nmap -Pn -sS -sC --script exploit scanme.nmap.org
```

Utilizando o NMAP para testar vulnerabilidade a ataques DoS:

```
# nmap -Pn -sS -sC --script dos scanme.nmap.org
```

Varredura de vulnerabilidade individual: (malware, intrusive, fuzzer, intrusive, etc.)

```
nmap -p 80 --script dns-brute.nse vulnweb.com
nmap -sV --script http-csrf <target>
nmap -sV --script http-sherlock <target>
nmap -sV --script http-slowloris-check <target>
nmap -sV --script http-vmware-path-vuln <target>
nmap -sV --script http-passwd <target>
nmap -sV --script http-internal-ip-disclosure <target>
```

Obrigado!

“QUEM NÃO SABE O QUE PROCURA, NÃO PERCEBE QUANDO ENCONTRA”.