Redes de Computadores

Introdução:

Nmap é um software livre que realiza port scan desenvolvido pelo Gordon Lyon, autoproclamado hacker *"Fyodor"* É muito utilizado para avaliar a segurança dos computadores, e para descobrir serviços ou servidores em uma rede de computadores.

Nmap foi primeiramente publicado em setembro de 1997, em um artigo na revista Phrack com o código fonte incluso. Com a ajuda e contribuições da comunidade de segurança de computadores, o desenvolvimento continuou. Atualizações do programa incluem detecção do sistema operacional, detecção de serviço, código reescrito de C para C++, tipos adicionais de scanning, suporte a novos protocolos e novos programas que complementam o núcleo do Nmap.

Recursos:

Os recursos do Nmap incluem:

Descoberta de hosts - Identificando hosts na rede. Por exemplo, recebendo respostas de Ping ou de uma porta aberta.

Scanner de portas - Mostrando as portas TCP e UDP abertas.

Detecção de versão - Interrogando serviços na rede para determinar a aplicação e o número da versão.

Detecção do sistema operacional - Remotamente determina o sistema operacional e as características de hardware do host.

Interação com scripts com o alvo - Usando Nmap Scripting Engine e Lua.

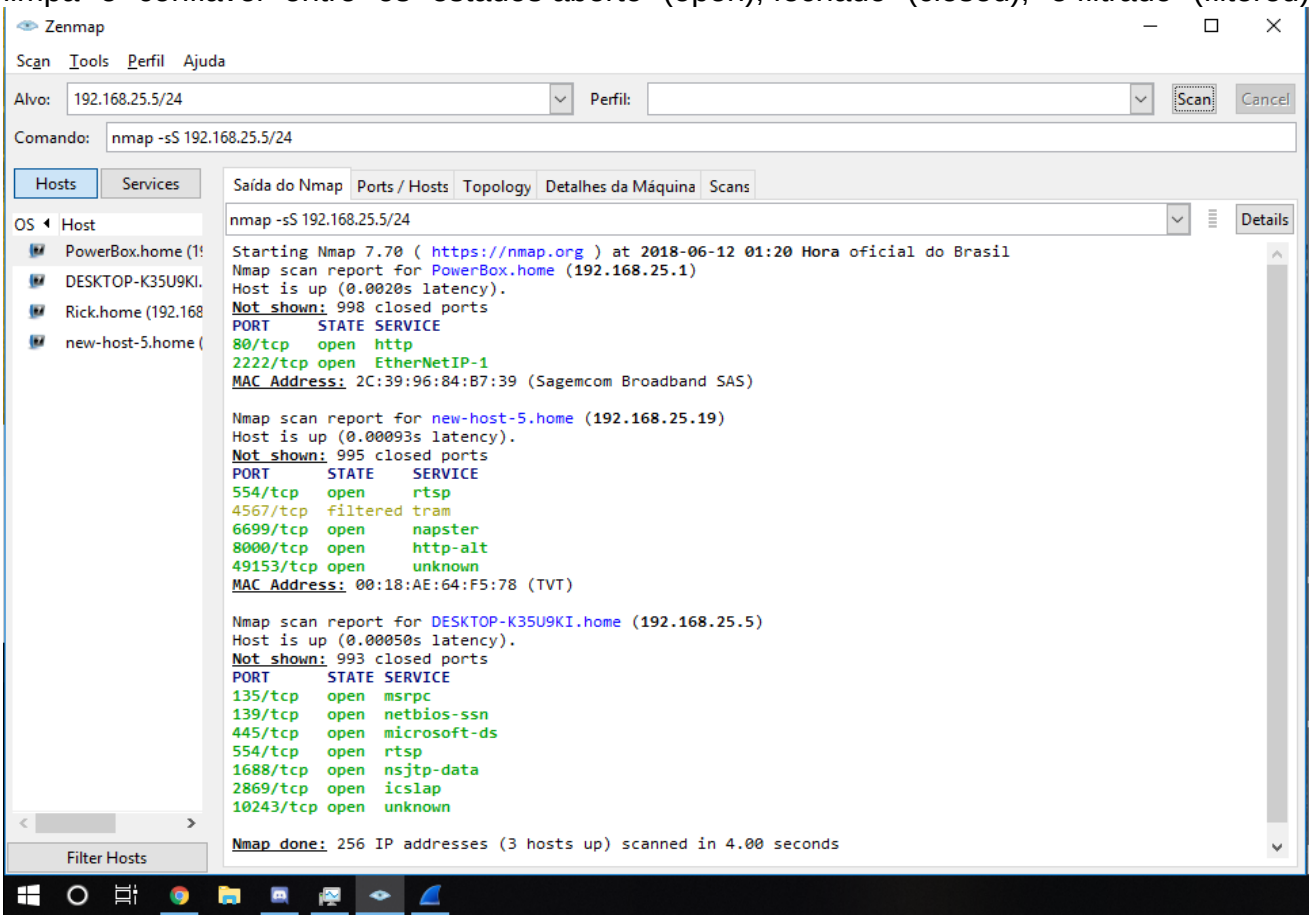
Além desses recursos, Nmap pode prover informações furtivas do alvo, incluindo DNS reverso, tipos de dispositivos, e endereços MAC.

Principais técnicas de escaneamento de portas:

Escaneamento por TCP SYN scan:

O scan SYN é a opção de scan padrão e mais popular por boas razões. Pode ser executada rapidamente, escaneando milhares de portas por segundo em uma rede rápida, não bloqueada por firewalls intrusivos. O scan SYN é relativamente não-obstrusivo e camuflado, uma vez que ele nunca completa uma conexão TCP. Ele também trabalha contra qualquer pilha TCP padronizada ao invés de depender de idiossincrasias de plataformas específicas como os scans Fin/Null/Xmas, Maimon e Idle fazem. Ele também permite uma diferenciação

limpa e confiável entre os estados aberto (open), fechado (closed), e filtrado (filtered).

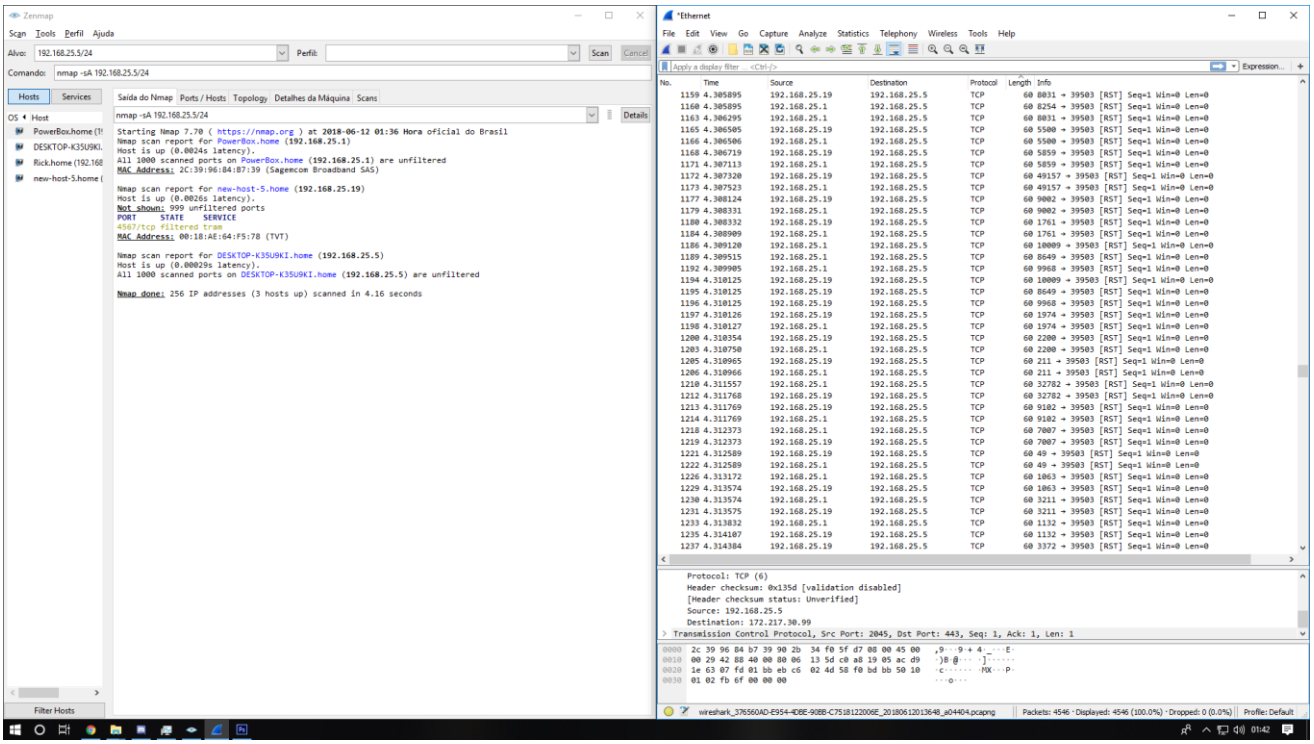


Escaneamento com WireShark: https://mega.nz/#!wJtiTAbS!pGwv-TdvmT9HvN5_xrVLFA4YK4rjS-GEhZp-5X4yzwU

Escaneamento por TCP ACK:

Esse scan é diferente dos outros discutidos até agora pelo fato de que ele nunca determina se uma porta estáaberta (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.

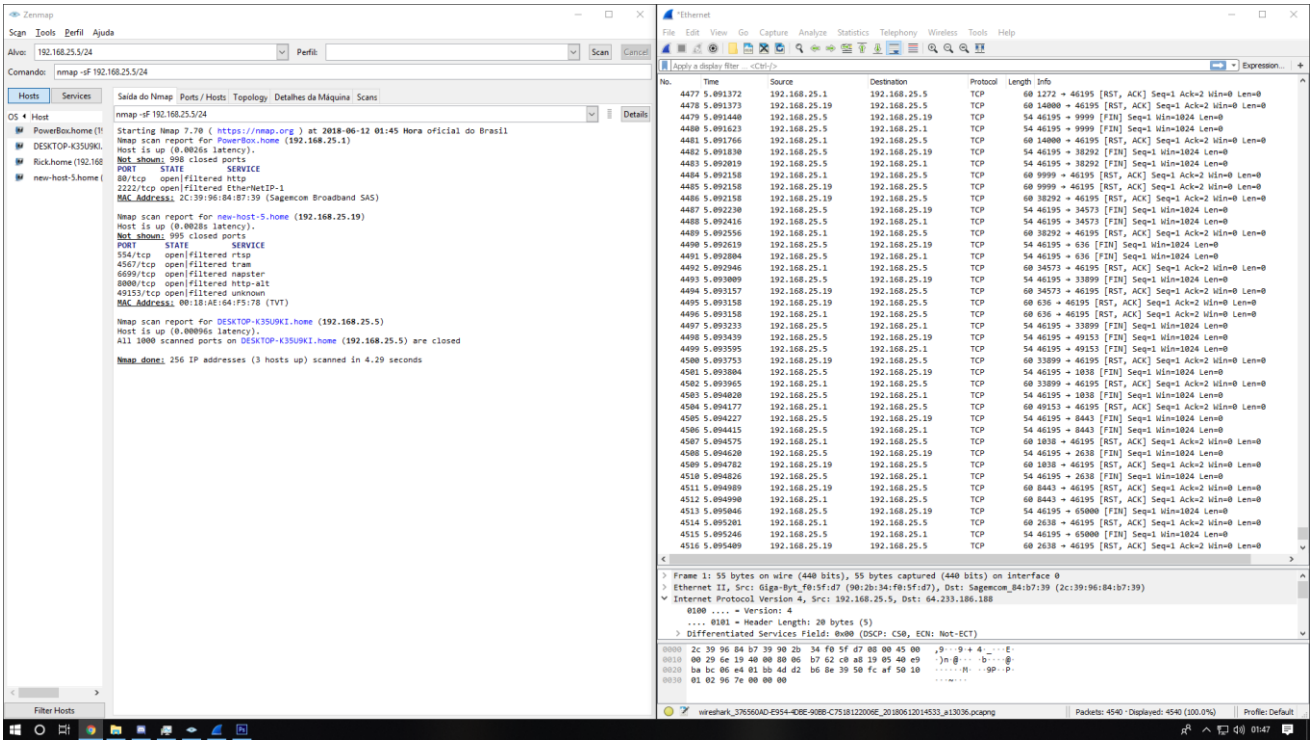
O pacote de sondagem do scan ACK tem apenas a flag ACK marcada (a menos que você use --scanflags). Quando se escaneia sistemas não-filtrados, as portas abertas e fechadas irão devolver um pacote RST. O Nmap então coloca nelas o rótulo não-filtradas (unfiltered), significando que elas estão alcançáveis pelo pacote ACK, mas se elas estão abertas ou fechadas é indeterminado. Portas que não respondem, ou que devolvem certas mensagens de erro ICMP (tipo 3, código 1, 2, 3, 9, 10, ou 13), são rotuladas como filtradas.



Escaneamento com WireShark: https://mega.nz/#!hQ82TIQB!hzOzhzIEi-SSu6CpP7IYXoexZBlceWRhJ81jiZnuZmw

Escaneamento por TCP FIN scan:

Marca apenas o bit FIN do TCP.



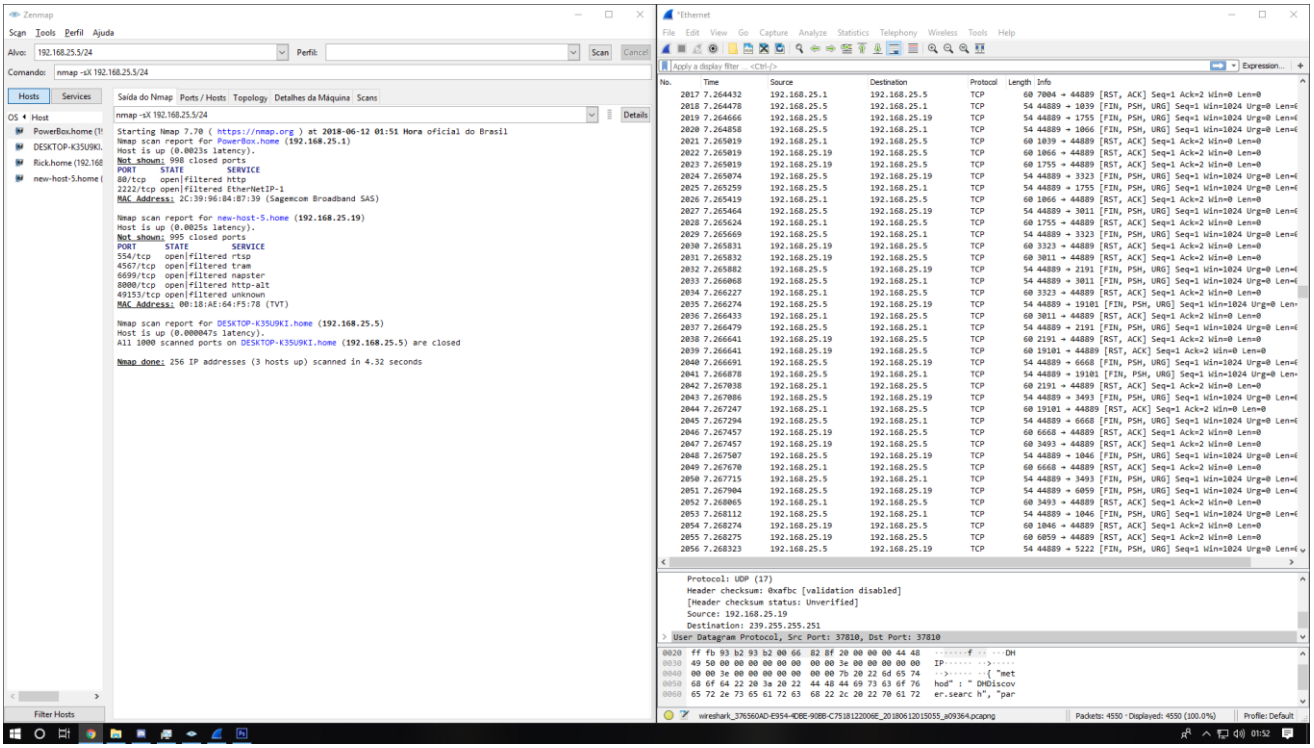
Escaneamento com WireShark: https://mega.nz/#!oEcwhCAL!B_YvGGbpRXSrCl-QyUcCE_-M2LODCrnmPU4kNPOdyfI

Escaneamento por TCP Xmas scan:

Marca as flags FIN, PSH e URG, iluminando o pacote como uma árvore de Natal.

Esses três tipos de scan são exatamente os mesmos em termos de comportamento, exceto pelas flags TCP marcadas no pacotes de sondagem. Se um pacote RST for recebido, a porta é considerada fechada, e nenhuma resposta significa que está aberta|filtrada. A porta é marcada como filtrada se um erro ICMP do tipo inalcançável (tipo 3, código 1, 2, 3, 9, 10, ou 13) for recebido.

A vantagem principal desses tipos de scan é que eles podem bisbilhotar através de alguns firewalls não-orientados à conexão e de roteadores que filtram pacotes. Outra vantagem é que esses tipos de scan são um pouco mais camuflados do que o scan SYN.

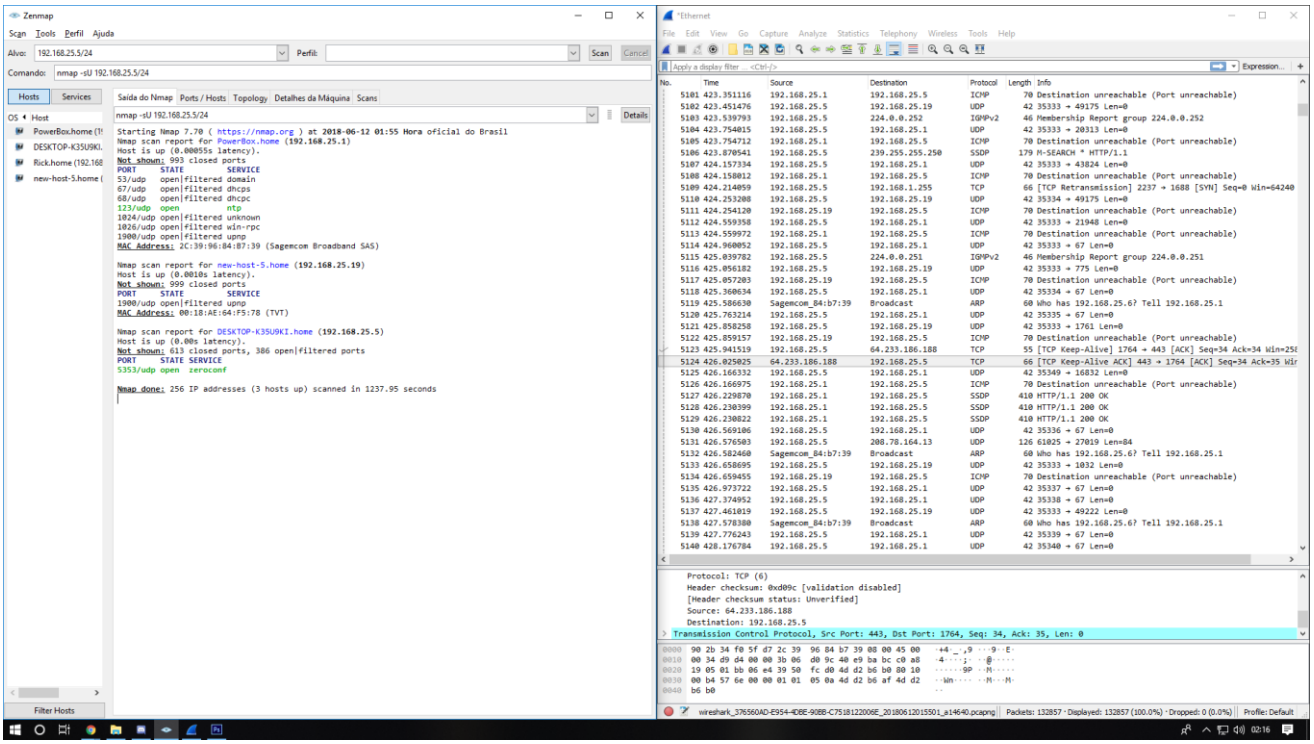


Escaneamento com WireShark:
https://mega.nz/#!RFNiARJT!dI54GZs0vPH68WhOYNMM4Tob4JtUCzgkCN3RYiQZ9EQ

Escaneamento UDP:

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 (registrados nas portas 53, 161/162, e 67/68) 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. Felizmente o Nmap pode ajudar a inventariar as portas UDP.

O scan UDP é ativado com a opção -sU. Ele pode ser combinado com um tipo de escaneamento TCP como o scan SYN (-sS) para averigüar ambos protocolos na mesma execução.

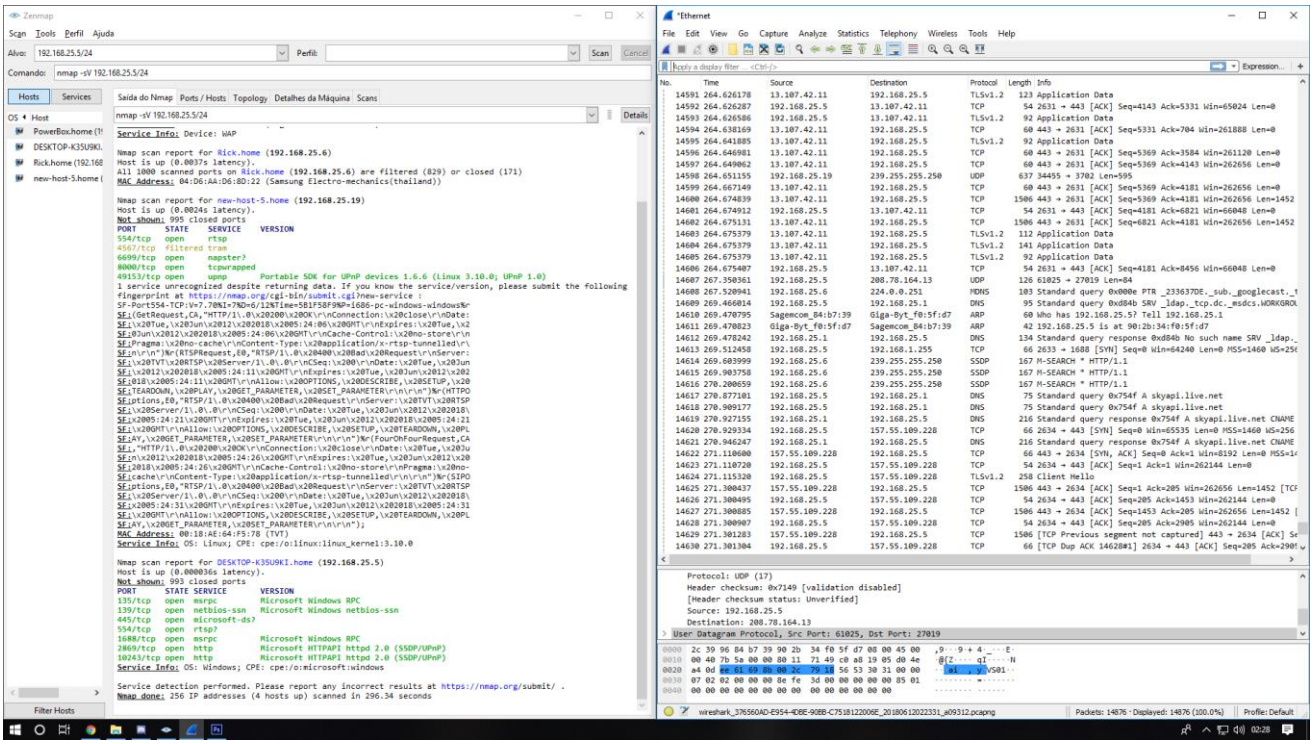


Escaneamento com WireShark:
https://mega.nz/#!MdVH3QBa!jNbqr4N0AP7zP2x91pGSAGNvg4RQr0bQ0J7rVw5a6Ag

Escaneamento de versões dos softwares:

A opção Nmap -sV permite a detecção de serviço, que retorna informações adicionais de serviço e versão. A detecção de serviço é um dos recursos mais amados do Nmap porque é muito útil em muitas situações, como identificar vulnerabilidades de segurança ou garantir que um serviço esteja sendo executado em uma determinada porta ou um patch foi aplicado com sucesso.

Já a opção -O indica ao Nmap que tente a detecção do SO enviando várias sondas usando os protocolos TCP, UDP e ICMP contra portas abertas e fechadas. O modo de detecção do sistema operacional é muito poderoso devido à comunidade de usuários do Nmap, que contribui com impressões digitais que identificam uma grande variedade de sistemas, incluindo roteadores residenciais, webcams IP, sistemas operacionais e muitos outros dispositivos de hardware. É importante notar que a detecção do sistema operacional requer pacotes brutos, portanto, o Nmap precisa ser executado com **privilégios suficientes**.



Escaneamento com WireShark: https://mega.nz/#!BR1giIKI!YvUFszAwyvuTS6nMme-PzfSz49B_bggVuq96-Fs1MH4

NSE (NMAP Scripting Engine):

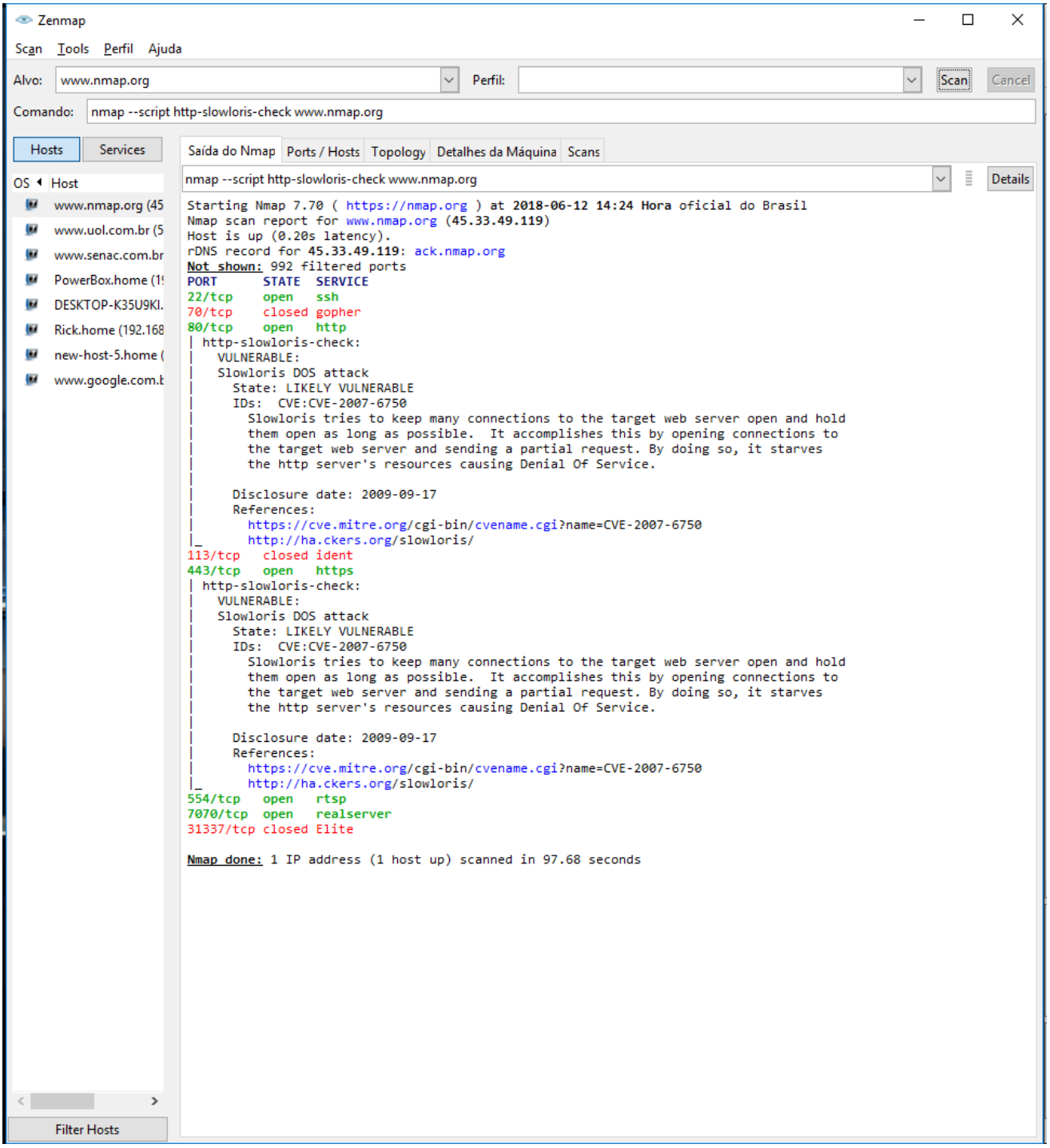
O que é ? - *NSE* é uma funcionalidade do *Nmap* poderosa, versátil e flexível, que permite a seus usuários desenvolver e compartilhar scripts simples e integrá-los às varreduras tradicionais do *Nmap*, a fim de automatizar tarefas variadas. Os usuários podem somente utilizar os scripts disponibilizados junto com o *Nmap*, modificar scripts existentes ou ainda desenvolver seus próprios scripts personalizados que atendam suas necessidades.

O *NSE* foi desenvolvido inicialmente com o objetivo de melhorar a descoberta de rede, incluir métodos mais sofisticados de detecção de versões e permitir a identificação de vulnerabilidades. Em sua versão atual, além destas funções, o *NSE* é capaz de detectar *backdoors*, explorar vulnerabilidades, realizar ataques de dicionário e de negação de serviço, detectar malwares remotamente, entre outros. Por se tratar de uma ferramenta tão versátil, é possível que surjam ainda scripts aplicáveis em novas situações não previstas pelos desenvolvedores e mantenedores do *Nmap*.

Linguagem utilizada - Os scripts executados pelo *NSE* são escritos na linguagem de script Lua. A linguagem Lua foi Criada em 1993 no Laboratório de Tecnologia em Computação Gráfica da Pontifícia Universidade Católica do Rio de Janeiro (TecGraf/PUC-Rio) e continua em desenvolvimento ativo ainda hoje. Lua é uma linguagem extensível, segura e portável, muito usada e já bastante depurada, além disto é considerada uma linguagem fácil de aprender (principalmente se já se conhece outras linguagens de script como perl, python ou outras linguagens como C/C++, Java, etc) e pequena para embutir (segundo a documentação da linguagem, uma distribuição completa, com código fonte, manual e binários para algumas plataformas cabem confortavelmente em um disquete). Atualmente, Lua é bastante usada no desenvolvimento de jogo, figurando em títulos como "*World of Warcraft*" e "*Crysis*", e, mais recentemente, em ferramentas de segurança, como o *Nmap*, naturalmente, o *Wireshark* e o *Snort* 3.0.

Scripts para encontrar vulnerabilidades HTTP:

http-vuln-cve2006-3392	Explora uma vulnerabilidade de divulgação de arquivos em Webmin (CVE-2006-3392)
http-vuln-cve2009-3960	Exploits cve-2009-3960 também conhecido como Adobe XML External Entity Injection.
http-vuln-cve2010-0738	Testa se um destino do JBoss é vulnerável ao bypass de autenticação do console jmx (CVE-2010-0738).
http-vuln-cve2010-2861	Executa um ataque de passagem de diretórios contra um servidor ColdFusion e tenta capturar o hash de senha para o usuário administrador. Em seguida, ele usa o valor salt (oculto na página da web) para criar o hash SHA1 HMAC que o servidor da Web precisa para autenticação como admin. Você pode passar esse valor para o servidor ColdFusion como o administrador sem quebrar o hash da senha.
http-vuln-cve2011-3192	Detecta uma vulnerabilidade de negação de serviço na maneira como o servidor da web Apache manipula solicitações para vários intervalos de sobreposição / simples de uma página.
http-vuln-cve2011-3368	Testa a vulnerabilidade CVE-2011-3368 (Bypass de proxy reverso) no modo de proxy reverso do servidor HTTP Apache. O script executará 3 testes: • o teste de loopback, com 3 cargas para lidar com diferentes regras de reescrita • o teste de hosts internos. Segundo Contextis, esperamos um atraso antes de um erro no servidor. • O teste do site externo. Isso não significa que você pode alcançar um ip de LAN, mas este é um problema relevante de qualquer maneira.
http-vuln-cve2012-1823	Detecta instalações PHP-CGI que são vulneráveis a CVE-2012-1823, Esta vulnerabilidade crítica permite que atacantes recuperem o código-fonte e executem o código remotamente.
http-vuln-cve2013-0156	Detecta servidores Ruby on Rails vulneráveis a injeção de objetos, execuções de comandos remotos e ataques de negação de serviço. (CVE-2013-0156)
http-vuln-cve2013-6786	Detecta um redirecionamento de URL e reflete a vulnerabilidade de XSS no servidor da Web Allegro RomPager. A vulnerabilidade foi atribuída a CVE-2013-6786.
http-vuln-cve2013-7091	Um dia 0 foi lançado no dia 6 de dezembro de 2013 por rubina119 e foi corrigido no Zimbra 7.2.6.



Scripts para encontrar vulnerabilidades SMB:

smb-vuln-conficker	Detecta os sistemas Microsoft Windows infectados pelo worm Conficker. Essa verificação é perigosa e pode falhar em sistemas.
smb-vuln-cve-2017-7494	Verifica se as máquinas de destino estão vulneráveis à vulnerabilidade de carga da biblioteca compartilhada arbitrária CVE-2017-7494.

smb-vuln-cve2009-3103	Detecta sistemas Microsoft Windows vulneráveis a negação de serviço (CVE-2009-3103). Este script irá travar o serviço se estiver vulnerável.
smb-vuln-ms06-025	Detecta sistemas Microsoft Windows com o serviço Ras RPC vulnerável ao MS06-025.
smb-vuln-ms07-029	Detecta os sistemas Microsoft Windows com o RPC do servidor de DNS vulnerável ao MS07-029.
smb-vuln-ms08-067	Detecta sistemas Microsoft Windows vulneráveis à vulnerabilidade de execução remota de código conhecida como MS08-067. Essa verificação é perigosa e pode falhar em sistemas.
smb-vuln-ms10-054	Testa se as máquinas de destino são vulneráveis à vulnerabilidade de corrupção de memória remota do SMB ms10-054.
smb-vuln-ms10-061	Testa se as máquinas de destino são vulneráveis à vulnerabilidade de representação do Spooler da impressora ms10-061.
smb-vuln-ms17-010	Tenta detectar se um servidor Microsoft SMBv1 está vulnerável a uma vulnerabilidade de execução remota de código (ms17-010, a.k.a. EternalBlue). A vulnerabilidade é ativamente explorada por WannaCry e Petya ransomware e outros malwares.
smb-vuln-regsvc-dos	Verifica se um sistema Microsoft Windows 2000 é vulnerável a uma falha no regsvc causada por uma referência de ponteiro nulo. Esta verificação irá travar o serviço se ele estiver vulnerável e exigir uma conta de convidado ou superior para funcionar.
smb2-vuln-uptime	Tenta detectar correções ausentes nos sistemas Windows, verificando o tempo de atividade retornado durante a negociação do protocolo SMB2.

Example Usage

- `nmap --script smb-vuln-cve-2017-7494 -p 445 <target>`
- `nmap --script smb-vuln-cve-2017-7494 --script-args smb-vuln-cve-2017-7494.check-version -p445 <target>`

Script Output

```
PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:16:04:53 (VMware)

| smb-vuln-cve-2017-7494:
|   VULNERABLE:
|   SAMBA Remote Code Execution from Writable Share
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-7494
|     Risk factor: HIGH  CVSSv3: 7.5 (HIGH) (CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H)
|       All versions of Samba from 3.5.0 onwards are vulnerable to a remote
|       code execution vulnerability, allowing a malicious client to upload a
|       shared library to a writable share, and then cause the server to load
|       and execute it.
|
|     Disclosure date: 2017-05-24
|     Check results:
|       Samba Version: 4.3.9-Ubuntu
|       Writable share found.
|        Name: \\192.168.15.131\test
|       Exploitation of CVE-2017-7494 succeeded!
|     Extra information:
|       All writable shares:
|        Name: \\192.168.15.131\test
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-7494
|_      https://www.samba.org/samba/security/CVE-2017-7494.html
```

Scripts para encontrar vulnerabilidades SMTP:

smtp-vuln-cve2010-4344	Verifica e / ou explora um estouro de heap nas versões do Exim anteriores à versão 4.69 (CVE-2010-4344) e uma vulnerabilidade de escalonamento de privilégios no Exim 4.72 e anterior (CVE-2010-4345).
smtp-vuln-cve2011-1720	Verifica se há uma corrupção de memória no servidor SMTP Postfix quando ele usa mecanismos de autenticação de biblioteca Cyrus SASL (CVE-2011-1720). Esta vulnerabilidade pode permitir negação de serviço e, possivelmente, execução remota de código.
smtp-vuln-cve2011-1764	Verifica se há uma vulnerabilidade de cadeia de formatação no servidor SMTP do Exim (versão 4.70 a 4.75) com suporte a DKIM (DomainKeys Identified Mail) (CVE-2011-1764). O mecanismo de log do DKIM não usou especificadores de cadeia de caracteres de formato ao registrar algumas partes do campo de cabeçalho DKIM-Signature. Um atacante remoto que é capaz de enviar e-mails, pode explorar esta vulnerabilidade e executar código arbitrário com os privilégios do daemon do Exim.

Example Usage

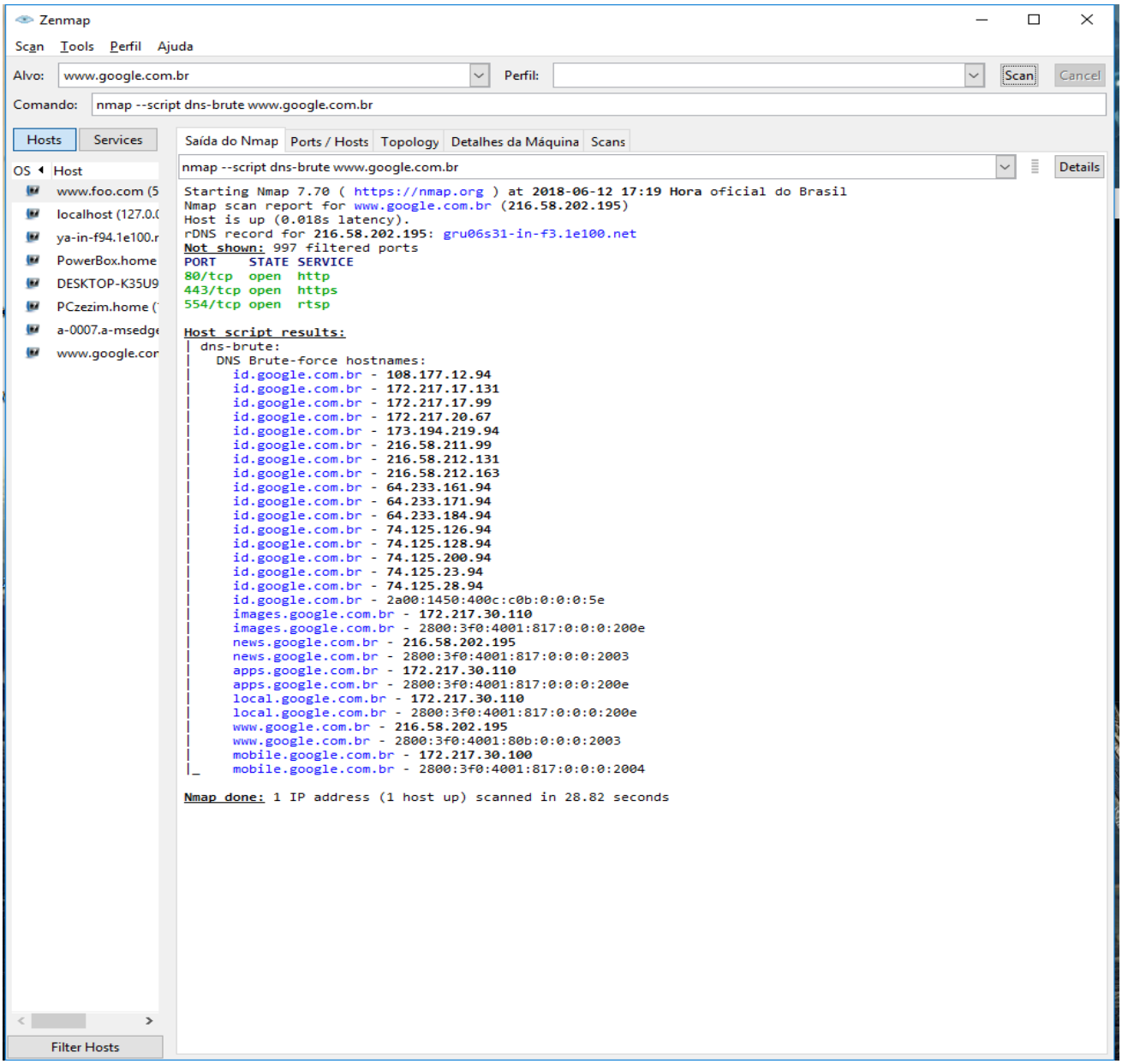
```
nmap --script=smtp-vuln-cve2011-1720 --script-args='smtp.domain=<domain>' -pT:25,465,587 <host>
```

Script Output

```
PORT   STATE SERVICE
25/tcp open  smtp
| smtp-vuln-cve2011-1720:
|   VULNERABLE:
|   Postfix SMTP server Cyrus SASL Memory Corruption
|     State: VULNERABLE
|     IDs:  CVE:CVE-2011-1720  OSVDB:72259
|     Description:
|       The Postfix SMTP server is vulnerable to a memory corruption vulnerability
|       when the Cyrus SASL library is used with authentication mechanisms other
|       than PLAIN and LOGIN.
|     Disclosure date: 2011-05-08
|     Check results:
|       AUTH tests: CRAM-MD5 NTLM
|     Extra information:
|       Available AUTH MECHANISMS:  CRAM-MD5 DIGEST-MD5 NTLM PLAIN LOGIN
|     References:
|       http://www.postfix.org/CVE-2011-1720.html
|       http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-1720
|_      http://osvdb.org/72259
```

Scripts para enumeração DNS:

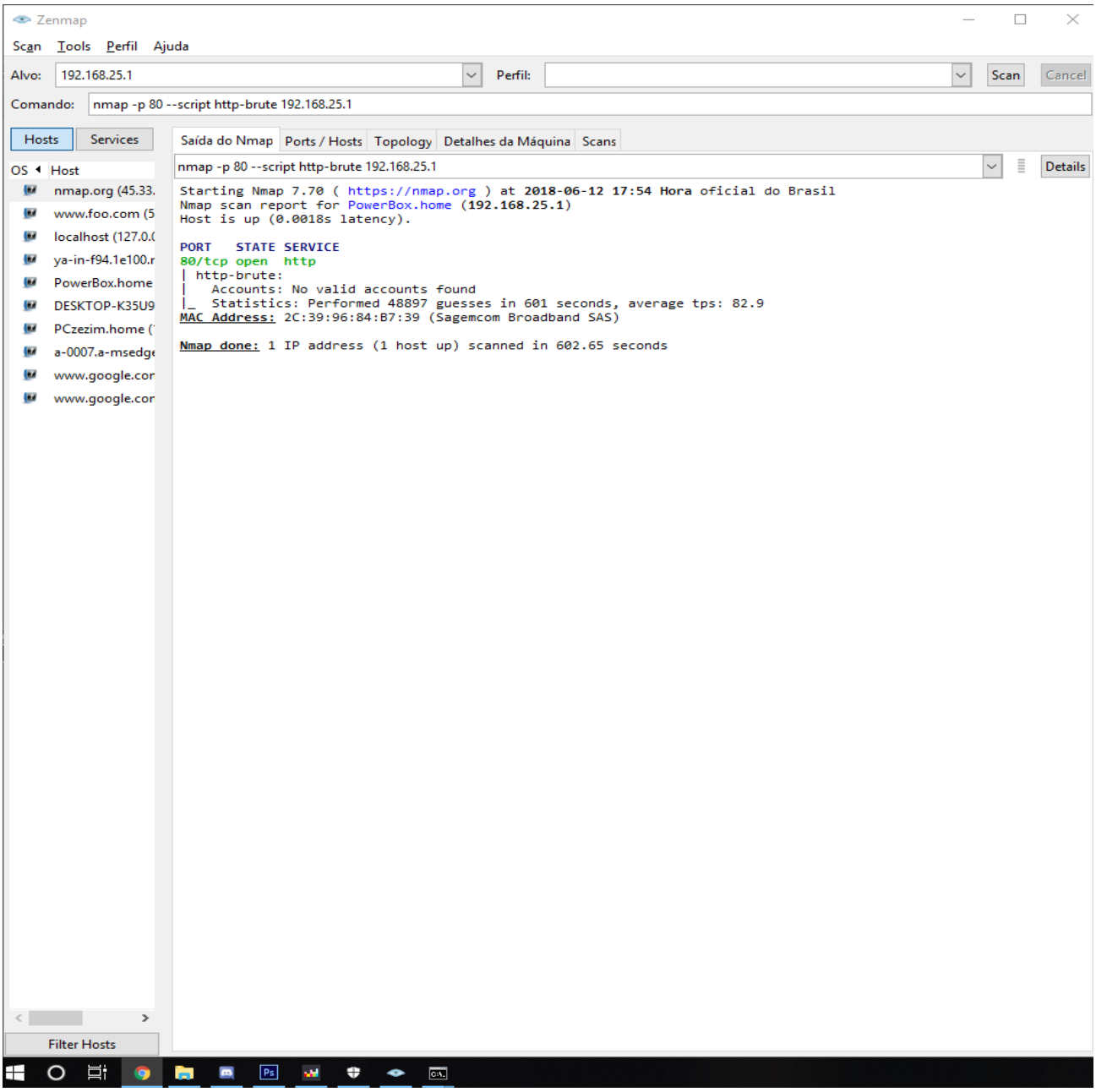
Attempts to enumerate DNS hostnames by brute force guessing of common subdomains. With the **dns-brute** argument, dns-brute will also try to enumerate common DNS SRV records.



Scripts de brute force:

afp-brute	Executa a adivinhação de senha contra o AFP (Apple Filing Protocol).
ajp-brute	Executa a auditoria de senhas de força bruta contra o protocolo Apache JServ. O Apache JServ Protocol é comumente usado por servidores da web para se comunicar com contêineres do servidor de aplicativos Java de backend.
backorifice-brute	Executa auditoria de senha de força bruta contra o serviço BackOrifice. O argumento do script backorifice-brute.ports é obrigatório (especifica portas para executar o script).

cassandra-brute	Executa a auditoria de senha de força bruta no banco de dados do Cassandra.
cics-enum	Enumerador de ID de transação do CICS para mainframes IBM. Este script é baseado no mainframe_brute de Dominic White (https://github.com/sensepost/mainframe_brute). No entanto, esse script não depende de nenhuma biblioteca ou ferramenta de terceiros e, em vez disso, usa a biblioteca NSE TN3270, que emula uma tela TN3270 em lua.
cics-user-brute	Script de força bruta de ID do usuário do CICS para a tela de login do CESL.
cics-user-enum	Script de enumeração de ID do usuário do CICS para a tela de login do CESL / CESN.
citrix-brute-xml	Tenta adivinhar credenciais válidas para o Serviço XML do Agente Web Citrix PN. O serviço XML é autenticado no servidor Windows local ou no Active Directory.
cvs-brute	Executa a auditoria de senha de força bruta contra a autenticação do pserver do CVS.
cvs-brute-repository	Tenta adivinhar o nome dos repositórios do CVS hospedados no servidor remoto. Com o conhecimento do nome correto do repositório, nomes de usuário e senhas podem ser adivinhados.
deluge-rpc-brute	Executa a auditoria de senha de força bruta contra o daemon DelugeRPC.
domcon-brute	Executa auditoria de senha de força bruta no Lotus Domino Console.
dpap-brute	Executa a auditoria de senha de força bruta contra uma biblioteca do iPhoto.
drda-brute	Executa a adivinhação de senha em bancos de dados que suportam o protocolo IBM DB2, como Informix, DB2 e Derby
ftp-brute	Executa auditoria de senha de força bruta contra servidores FTP.
http-brute	Executa auditoria de senha de força bruta contra autenticação básica básica, digest e ntlm.
http-form-brute	Executa auditoria de senha de força bruta contra a autenticação baseada em formulário http.
http-iis-short-name-brute	Tentativas de força bruta forçam os nomes de arquivos 8.3 (comumente conhecidos como nomes abreviados) de arquivos e diretórios na pasta raiz de servidores IIS vulneráveis. Este script é uma implementação do PoC "iis shortname scanner".



Conclusão:

O Nmap pode ser considerado uma ferramenta Hacker/Cracker, ou um excelente utilitário para consultores de segurança e administradores de rede, o fato é que ele realiza de forma extremamente eficiente o que se propõe.

Referências:

https://pt.wikipedia.org/wiki/Nmap

https://nmap.org/nsedoc/categories/discovery.html

https://nmap.org/man/pt_BR/man-port-scanning-techniques.html

https://crazybulletctfwriteups.wordpress.com/2015/09/05/nmap-vulnerability-discovery/