

FACULDADE SENAC GOIÁS
CURSO DE GESTÃO DE TECNOLOGIA DA INFORMAÇÃO
FUNDAMENTOS DE TECNOLOGIA DA INFORMAÇÃO

JOÃO NETTO PINHEIRO
JOÃO PAULO NASCIMENTO OLIVEIRA
PAULO ROBERTO VIEIRA

NMAP

Professor: Fernando Pirkel Tsukahara

GOIÂNIA
2018

Sumário

1. O que é o NMAP	3
2. História do NMAP	3
3. NMAP nos filmes	3
4. Principais técnicas de escaneamento de portas	4
4.1. Escaneamento por TCP SYN scan	4
4.1.1. Exemplos de uso	5
4.1.2. Captura do escaneamento com WireShark	5
4.1.3. Captura do escaneamento com Tcpdump	6
4.2. Escaneamento por TCP ACK scan	6
4.2.1. Exemplos de uso	7
4.2.2. Captura do escaneamento com WireShark	7
4.2.3. Captura do escaneamento com Tcpdump	8
4.3. Escaneamento por TCP FIN scan	8
4.3.1. Exemplos de uso	8
4.3.2. Captura do escaneamento com WireShark	9
4.3.3. Captura do escaneamento com Tcpdump	9
4.4. Escaneamento por TCP Xmas scan	10
4.4.1. Exemplos de uso	10
4.4.2. Captura do escaneamento com WireShark	10
4.4.3. Captura do escaneamento com Tcpdum	11
4.5. Escaneamento UDP	11
4.5.1. Exemplos de uso	12
4.5.2. Captura do escaneamento com WireShark	13
4.5.3. Captura do escaneamento com Tcpdump	14
4.6. Escaneamento de versões dos softwares	14
4.6.1. Exemplos de uso	18
4.6.2. Captura do escaneamento com WireShark	18
4.6.3. Captura do escaneamento com Tcpdump	19
5. Conclusão	19
6. Referências	20

1. O que é o NMAP?

Nmap ("Network Mapper") é uma fonte livre e aberta (licença) utilitário para descoberta de rede e auditoria de segurança. Muitos sistemas e administradores de rede também acham útil para tarefas como inventário de rede, gerenciamento de agendas de atualização de serviço e monitoramento de tempo de atividade de host ou serviço. O Nmap usa pacotes IP brutos para determinar quais hosts estão disponíveis na rede, quais serviços (nome do aplicativo e versão) esses hosts estão oferecendo, quais sistemas operacionais (e versões do SO) estão executando, que tipo de filtros de pacotes / firewalls estão em uso e dezenas de outras características.

2. História do NMAP.

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.

3. NMAP nos filmes.

Por razões desconhecidas, Hollywood decidiu que o Nmap é a ferramenta para mostrar sempre que cenas de hackers são necessárias. Pelo menos, é muito mais realista do que a abordagem boba de animação 3D usada em muitos filmes anteriores (por exemplo, "hacking the Gibson" em Hackers, ou os retratos muito piores em Swordfish). Nós sempre gostamos de ver o Nmap nos filmes. Embora o Nmap tenha sido usado em alguns filmes obscuros anteriores, foi o The Matrix Reloaded que realmente transformou o Nmap em uma estrela de cinema.

Em The Bourne Ultimatum, a CIA precisa hackear o servidor de e-mail de um jornal (The Guardian UK) para ler o e-mail de um repórter que eles assassinaram. Então eles se voltam para o Nmap e seu novo GUI Zenmap oficial para hackear o servidor de email! O Nmap relata que o servidor de e-mail está executando o SSH 3.9p1, Postfix smtpd e um servidor de nomes (presumivelmente bind). Eles também fazem uso substancial do Bash, o shell Bourne-again.

4. Principais técnicas de escaneamento de portas:

Uma vez que o Nmap é gratuito, a única barreira para a maestria em escaneamento de portas é o conhecimento. A maioria dos tipos de scan está disponível apenas para usuários privilegiados. Isso acontece porque eles enviam e recebem pacotes em estado bruto, o que requer acesso de root em sistemas Unix. Utilizar a conta de administrador no Windows é recomendado, embora o Nmap às vezes funcione com usuários sem privilégios nessa plataforma quando o WinPcap foi carregado no SO. Requerer privilégio de root era uma séria limitação quando o Nmap foi lançado em 1997, pois muitos usuários apenas tinham acesso a contas de shell compartilhadas. Agora o mundo é diferente. Computadores estão mais baratos, muito mais pessoas tem acesso direto e permanente à Internet, e computadores de mesa Unix (incluindo Linux e MAC OS X) são comuns. Dos scans listados nesta seção, os usuários não privilegiados podem apenas executar os scans connect e ftp bounce.

- sS (scan TCP SYN)
- sT (scan TCP connect)
- sU (scans UDP)
- sN; -sF; -sX (scans TCP Null, FIN, e Xmas)
- scan Null (-sN)
- scan FIN (-sF)
- scan Xmas(-sX)
- sA (scan TCP ACK)
- sW (scan da Janela TCP)
- sM (scan TCP Maimon)
- scanflags (scan TCP Personalizado)
- sl <hostzumbi[:portadesondagem]> (scan Idle)
- sO (Scans do protocolo IP)
- b <host para relay de ftp> (Scan de FTP bounce)

4.1. 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-obstrutivo 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).

Esta técnica é frequentemente chamada de escaneamento de porta entreaberta (half-open scanning), porque você não abre uma conexão TCP completamente. Você envia 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 uma não-ouvinte. Se nenhuma resposta é recebida após diversas retransmissões, a porta é marcada como filtrada. A porta também é marcada como filtrada se um erro ICMP de inalcançável é recebido (tipo 3, código 1,2, 3, 9, 10, ou 13).

4.1.1. Exemplos de uso

-sS (scan TCP SYN)

4.1.2. Captura do escaneamento com WireShark

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	13.78.168.230	192.168.50.220	TCP	60	443 → 24750 [ACK] Seq=1 Ack=1 Win=0 Len=0
2	0.001720	13.78.168.230	192.168.50.220	TLV1.2	185	Change Cipher Spec, Encrypted Handshake Message
3	0.001998	13.78.168.230	192.168.50.220	TLV1.2	123	Application Data
4	0.002043	192.168.50.220	13.78.168.230	TCP	54	24750 → 443 [ACK] Seq=94 Ack=221 Win=256 Len=0
5	0.002452	192.168.50.220	13.78.168.230	TCP	54	24750 → 443 [RST] Seq=94 Win=0 Len=0
6	0.003157	192.168.50.220	13.78.168.230	TCP	60	24751 → 443 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
7	0.079203	192.168.50.220	65.52.188.27	TLV1.2	427	Application Data
8	0.190802	13.78.168.230	192.168.50.220	TCP	60	443 → 24751 [SYN, ACK] Seq=0 Ack=1 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
9	0.190843	192.168.50.220	13.78.168.230	TCP	54	24751 → 443 [ACK] Seq=1 Ack=1 Win=66048 Len=0
10	0.191219	192.168.50.220	13.78.168.230	TLV1.2	261	Client Hello
11	0.191933	192.168.50.220	192.168.50.2	DNS	141	Standard query 0x6038 A 0.19-0.3000001.0.170c.22da.2f4a.210.0.0p0ffifuslmw8ck3jrhce5thji.avts.mcafee.com
12	0.199714	192.168.50.2	192.168.50.220	DNS	183	Standard query response 0x6038 A 0.19-0.3000001.0.170c.22da.2f4a.210.0.0p0ffifuslmw8ck3jrhce5thji.avts.mcafee.com A 127.192.0.128 NS local.cloud.mcafee.com
13	0.233559	65.52.188.27	192.168.50.220	TLV1.2	429	Application Data
14	0.233601	192.168.50.220	65.52.188.27	TCP	54	24700 → 443 [ACK] Seq=374 Ack=376 Win=1013 Len=0
15	0.234162	192.168.50.220	23.75.147.163	TCP	60	24752 → 443 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
16	0.244918	23.75.147.163	192.168.50.220	TCP	60	443 → 24752 [SYN, ACK] Seq=0 Ack=1 Win=29280 Len=0 MSS=1460 SACK_PERM=1 WS=128
17	0.244994	192.168.50.220	23.75.147.163	TCP	54	24752 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
18	0.245307	192.168.50.220	23.75.147.163	TLV1.2	211	Client Hello
19	0.255399	23.75.147.163	192.168.50.220	TCP	60	443 → 24752 [RST] Seq=1 Win=0 Len=0
20	0.255799	23.75.147.163	192.168.50.220	TCP	60	443 → 24752 [RST] Seq=1 Win=0 Len=0
21	0.255910	192.168.50.220	23.75.147.163	TCP	60	24753 → 443 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
22	0.261320	23.75.147.163	192.168.50.220	TCP	60	443 → 24753 [SYN, ACK] Seq=0 Ack=1 Win=29280 Len=0 MSS=1460 SACK_PERM=1 WS=128
23	0.261392	192.168.50.220	23.75.147.163	TCP	54	24753 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
24	0.261637	192.168.50.220	23.75.147.163	TLV1.2	183	Client Hello
25	0.266313	23.75.147.163	192.168.50.220	TCP	60	443 → 24753 [RST] Seq=1 Win=0 Len=0
26	0.266332	192.168.50.220	23.75.147.163	TCP	60	24754 → 443 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
27	0.266814	23.75.147.163	192.168.50.220	TCP	60	443 → 24753 [RST] Seq=1 Win=0 Len=0
28	0.271817	23.75.147.163	192.168.50.220	TCP	60	443 → 24754 [SYN, ACK] Seq=0 Ack=1 Win=29280 Len=0 MSS=1460 SACK_PERM=1 WS=128
29	0.271854	192.168.50.220	23.75.147.163	TCP	54	24754 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
30	0.271915	192.168.50.220	23.75.147.163	TCP	54	24754 → 443 [FIN, ACK] Seq=1 Ack=1 Win=262144 Len=0
31	0.272767	192.168.50.220	192.168.50.2	DNS	77	Standard query 0x03c4 A upad.facsnet.edu
32	0.272948	192.168.50.2	192.168.50.220	DNS	152	Standard query response 0x03c4 No such name A upad.facsnet.edu SOA a.edu-servers.net
33	0.276639	23.75.147.163	192.168.50.220	TCP	60	443 → 24754 [RST] Seq=1 Win=0 Len=0
34	0.276648	23.75.147.163	192.168.50.220	TCP	60	443 → 24754 [RST] Seq=1 Win=0 Len=0
35	0.379476	13.78.168.230	192.168.50.220	TCP	1514	443 → 24751 [ACK] Seq=1 Ack=200 Win=262656 Len=1660 [TCP segment of a reassembled PDU]
36	0.379757	13.78.168.230	192.168.50.220	TCP	1514	443 → 24751 [ACK] Seq=1461 Ack=200 Win=262656 Len=1660 [TCP segment of a reassembled PDU]
37	0.379759	13.78.168.230	192.168.50.220	TLV1.2	1041	Server Hello, Certificate, Server Key Exchange, Server Hello Done
38	0.379818	192.168.50.220	13.78.168.230	TCP	54	24751 → 443 [ACK] Seq=200 Ack=3908 Win=66048 Len=0
39	0.381000	192.168.50.220	13.78.168.230	TLV1.2	147	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
40	0.381245	65.52.188.27	192.168.50.220	TCP	60	1159 [TCP Seq 1341] 443 → 24700 [ACK] Seq=376 Ack=376 Win=1013 Len=0
41	0.392541	192.168.50.220	192.168.50.2	DNS	147	Standard query 0xd78c A 0.19-0.3000071.9030001.170c.22da.2f4a.210.0.0testjsunber37zg8th5jck6.avts.mcafee.com
42	0.393248	192.168.50.2	192.168.50.220	DNS	189	Standard query response 0xd78c A 0.19-0.3000071.9030001.170c.22da.2f4a.210.0.0testjsunber37zg8th5jck6.avts.mcafee.com A 127.129.0.128 NS local.cloud.mcafee.com
43	0.504747	13.78.168.230	192.168.50.220	TCP	60	1159 [TCP Seq 1341] 443 → 24751 [ACK] Seq=1988 Ack=200 Win=262656 Len=0

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on Interface 0
 > Ethernet II, Src: 7c:4d:23:44:cc:18 (Pcs4d2344cc18), Dst: Avc0aIn, 08:90:af (78:05:c2:00:90:af)
 > Internet Protocol Version 4, Src: 13.78.168.230, Dst: 192.168.50.220
 > Transmission Control Protocol, Src Port: 443, Dst Port: 24750, Seq: 1, Ack: 1, Len: 0

0000 70 85 c2 00 90 af 7c 4d 23 44 cc 8a 00 00 45 00 p.....M #D....E.
 0010 00 28 6c 70 40 00 00 00 fe a0 0d 4e a5 e5 c0 a5 .[log.f.N...
 0020 32 4c 01 30 00 00 01 0e 71 76 76 09 12 64 50 10 2.....0.v....P.
 0030 04 02 f0 89 00 00 00 00 e7 2c 26 25 ,&N

Wireshark 7.2.0 (32-bit) - 64-bit
 Packets: 8365 - Displayed: 8365 (100.0%)

4.1.3. Captura do escaneamento com Tcpcmdump

```

root@aluno-DC2C-S:~/home/aluno# tcpdump -i enp8s0f0 "tcp[tcpflags] & (tcp-syn) != 0"
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp8s0f0, link-type EN10MB (ethernet), capture size 262144 bytes
19:13:32.403686 IP aluno-DC2C-S.54784 > pc-pool-ngn.flickr.vip.bfi.yahoo.com.https: Flags [S], seq 1114150151, win 29200, options [mss 1460,sackOK,TS val 3801351439, ecr 0,nop,wscale 7], length 0
19:13:32.584571 IP aluno-DC2C-S.42240 > pc-pool-ngn.flickr.vip.bfi.yahoo.com.https: Flags [S], seq 249859089, ack 1114150152, win 5840, options [mss 1460,nop,nop,sackOK,nop,wscale 8], length 0
19:14:00.749332 IP aluno-DC2C-S.47912 > gru10s10-ln-f10.1e100.net.https: Flags [S], seq 4277829466, win 29200, options [mss 1460,sackOK,TS val 4028401949, ecr 0,nop,wscale 7], length 0
19:14:00.742916 IP gru10s10-ln-f10.1e100.net.https > aluno-DC2C-S.47912: Flags [S], seq 1884407354, ack 4277829467, win 42408, options [mss 1380,sackOK,TS val 1649785805, ecr 4028401949,nop,wscale 8], length 0
19:14:00.832204 IP aluno-DC2C-S.50810 > gru10s10-ln-f10.1e100.net.https: Flags [S], seq 2194723896, win 29200, options [mss 1460,sackOK,TS val 4028402062, ecr 0,nop,wscale 7], length 0
19:14:37.026437 IP aluno-DC2C-S.50120 > gru10s10-ln-f10.1e100.net.https: Flags [S], seq 3099670796, win 29200, options [mss 1460,sackOK,TS val 3455779481, ecr 0,nop,wscale 7], length 0
19:14:37.105950 IP gru10s10-ln-f10.1e100.net.https > aluno-DC2C-S.50120: Flags [S], seq 2189785565, ack 3099670797, win 42408, options [mss 1380,sackOK,TS val 70983, ecr 3455779481,nop,wscale 8], length 0
19:14:38.171087 IP aluno-DC2C-S.48176 > nycp-hlb26.doubleverifry.com.https: Flags [S], seq 160649742, win 29200, options [mss 1460,sackOK,TS val 3932477209, ecr 0,nop,wscale 7], length 0
19:14:38.173127 IP aluno-DC2C-S.48178 > nycp-hlb26.doubleverifry.com.https: Flags [S], seq 1651271773, win 29200, options [mss 1460,sackOK,TS val 3932477211, ecr 0,nop,wscale 7], length 0
19:14:38.321487 IP nycp-hlb26.doubleverifry.com.https > aluno-DC2C-S.48176: Flags [S], seq 2127250985, ack 160649742, win 42340, options [mss 1460,nop,wscale 12], length 0
19:14:38.325601 IP aluno-DC2C-S.57664 > nycp-hlb26.doubleverifry.com.https: Flags [S], seq 2289938320, win 29200, options [mss 1460,sackOK,TS val 3932477363, ecr 0,nop,wscale 7], length 0
19:14:38.337476 IP nycp-hlb26.doubleverifry.com.https > aluno-DC2C-S.48178: Flags [S], seq 458482745, ack 1651271774, win 42340, options [mss 1460,nop,wscale 12], length 0
19:14:49.726284 IP aluno-DC2C-S.51688 > gru10s10-ln-f10.1e100.net.https: Flags [S], seq 3600877383, win 29200, options [mss 1460,sackOK,TS val 1842540521, ecr 0,nop,wscale 7], length 0
19:14:49.726011 IP aluno-DC2C-S.35854 > gru10s10-ln-f10.1e100.net.https: Flags [S], seq 1273276405, win 29200, options [mss 1460,sackOK,TS val 314771815, ecr 0,nop,wscale 7], length 0
19:14:49.765808 IP gru10s10-ln-f10.1e100.net.https > aluno-DC2C-S.51688: Flags [S], seq 232562423, ack 3600877384, win 42408, options [mss 1380,sackOK,TS val 825956485, ecr 1842540521,nop,wscale 8], length 0
19:14:49.775901 IP gru10s10-ln-f10.1e100.net.https > aluno-DC2C-S.35854: Flags [S], seq 3084717780, ack 1275726406, win 42408, options [mss 1380,sackOK,TS val 3338778801, ecr 314771815,nop,wscale 8], length 0
19:14:49.874872 IP aluno-DC2C-S.39122 > gru10s11-ln-f2.1e100.net.https: Flags [S], seq 1582616523, win 29200, options [mss 1460,sackOK,TS val 2130707492, ecr 0,nop,wscale 7], length 0
19:14:49.902866 IP gru10s11-ln-f2.1e100.net.https > aluno-DC2C-S.39122: Flags [S], seq 2787178722, ack 1582616524, win 42408, options [mss 1380,sackOK,TS val 1829643766, ecr 2130707492,nop,wscale 8], length 0
19:14:49.991775 IP aluno-DC2C-S.39914 > gru10s11-ln-f2.1e100.net.https: Flags [S], seq 2069594016, win 29200, options [mss 1460,sackOK,TS val 2130707609, ecr 0,nop,wscale 7], length 0
tcpdump tcp syn scan:16:31.412105 IP aluno-DC2C-S.45984 > 186-192-81-5.prt.globo.com.https: Flags [S], seq 1058916492, win 29200, options [mss 1460,sackOK,TS val 2750278871, length 0]
19:16:31.463877 IP 186-192-81-5.prt.globo.com.https > aluno-DC2C-S.45984: Flags [S], seq 2716198565, ack 1058916493, win 14480, options [mss 1460,sackOK,TS val 2244683404, ecr 2750278871,nop,wscale 10], length 0
19:16:31.518379 IP aluno-DC2C-S.56456 > 186-192-81-5.prt.globo.com.https: Flags [S], seq 2424865183, win 29200, options [mss 1460,sackOK,TS val 2750278977, ecr 0,nop,wscale 7], length 0
19:16:31.597913 IP aluno-DC2C-S.54334 > 186-192-91-9.prt.globo.com.https: Flags [S], seq 3028557497, win 29200, options [mss 1460,sackOK,TS val 2344101223, ecr 0,nop,wscale 7], length 0
19:16:31.590183 IP aluno-DC2C-S.55446 > 186-192-81-62.prt.globo.com.https: Flags [S], seq 1335581133, win 29200, options [mss 1460,sackOK,TS val 2692524477, ecr 0,nop,wscale 7], length 0
19:16:31.598340 IP aluno-DC2C-S.34274 > 186-192-91-9.prt.globo.com.https: Flags [S], seq 10806648525, win 29200, options [mss 1460,sackOK,TS val 1415435899, ecr 0,nop,wscale 7], length 0
19:16:31.599431 IP aluno-DC2C-S.35870 > gru10s10-ln-f14.1e100.net.https: Flags [S], seq 2013457439, win 29200, options [mss 1460,sackOK,TS val 314873700, ecr 0,nop,wscale 7], length 0
19:16:31.656804 IP gru10s10-ln-f14.1e100.net.https > aluno-DC2C-S.35870: Flags [S], seq 249195743, ack 2013457440, win 42408, options [mss 1380,sackOK,TS val 1062907110, ecr 314873700,nop,wscale 8], length 0
19:16:31.659567 IP 186-192-81-62.prt.globo.com.https > aluno-DC2C-S.55446: Flags [S], seq 3585292726, ack 1335581134, win 14480, options [mss 1460,sackOK,TS val 380679336, ecr 2692524477,nop,wscale 10], length 0
19:16:31.660897 IP 186-192-91-9.prt.globo.com.https > aluno-DC2C-S.34274: Flags [S], seq 2314298929, ack 1006648526, win 14480, options [mss 1460,sackOK,TS val 2249206578, ecr 1415435899,nop,wscale 10], length 0
19:16:31.801647 IP aluno-DC2C-S.38136 > 186-192-91-9.prt.globo.com.https: Flags [S], seq 182418873, win 29200, options [mss 1460,sackOK,TS val 1415436094, ecr 0,nop,wscale 7], length 0
19:16:31.854744 IP aluno-DC2C-S.35780 > 201.7.182.243.https: Flags [S], seq 1035734086, win 29200, options [mss 1460,sackOK,TS val 345963597, ecr 0,nop,wscale 7], length 0
19:16:31.854772 IP aluno-DC2C-S.35782 > 201.7.182.243.https: Flags [S], seq 498063033, win 29200, options [mss 1460,sackOK,TS val 345963597, ecr 0,nop,wscale 7], length 0
19:16:31.873250 IP aluno-DC2C-S.37654 > 201.7.182.243.https: Flags [S], seq 609020267, win 29200, options [mss 1460,sackOK,TS val 345963615, ecr 0,nop,wscale 7], length 0
19:16:31.918824 IP aluno-DC2C-S.32828 > a23-75-156-244.deploy.static.akamaitechnologies.com.https: Flags [S], seq 163388348, win 29200, options [mss 1460,sackOK,TS val 2038394565, ecr 0,nop,wscale 7], length 0
19:16:31.934328 IP aluno-DC2C-S.54926 > 186-192-90-3.prt.globo.com.https: Flags [S], seq 417758644, win 29200, options [mss 1460,sackOK,TS val 2892988215, ecr 0,nop,wscale 7], length 0
19:16:32.003183 IP 186-192-90-3.prt.globo.com.https > aluno-DC2C-S.54926: Flags [S], seq 4157153909, ack 4177586445, win 20844, options [mss 1460,sackOK,TS val 1516629505, ecr 2892988215,nop,wscale 7], length 0
19:16:32.004760 IP a23-75-156-244.deploy.static.akamaitechnologies.com.https > aluno-DC2C-S.32828: Flags [S], seq 16408339701, ack 163388349, win 28960, options [mss 1460,sackOK,TS val 62507886, ecr 2038394565,nop,wscale 7], length 0
19:16:33.813937 IP aluno-DC2C-S.50852 > 186-192-81-31.prt.globo.com.https: Flags [S], seq 2874184941, win 29200, options [mss 1460,sackOK,TS val 1161799055, ecr 0,nop,wscale 7], length 0
19:16:33.824189 IP aluno-DC2C-S.35148 > 186-192-81-31.prt.globo.com.https: Flags [S], seq 1165835457, win 29200, options [mss 1460,sackOK,TS val 1161799065, ecr 0,nop,wscale 7], length 0
19:16:33.874461 IP aluno-DC2C-S.55578 > 186-192-81-168.prt.globo.com.https: Flags [S], seq 1019871527, win 29200, options [mss 1460,sackOK,TS val 89802331, ecr 0,nop,wscale 7], length 0

```

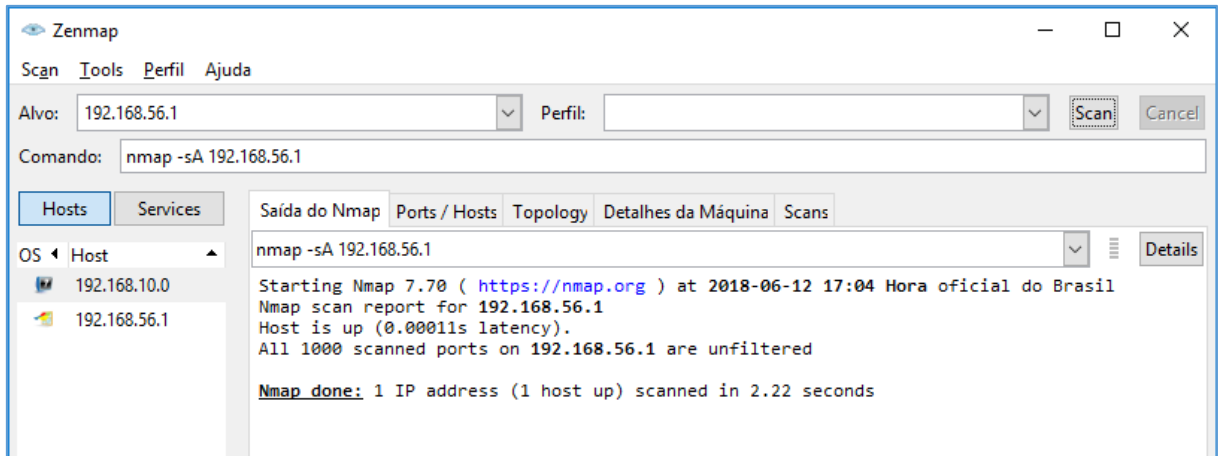
4.2. Escaneamento por TCP ACK scan

Esse scan é diferente dos outros, pelo fato de que ele 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.

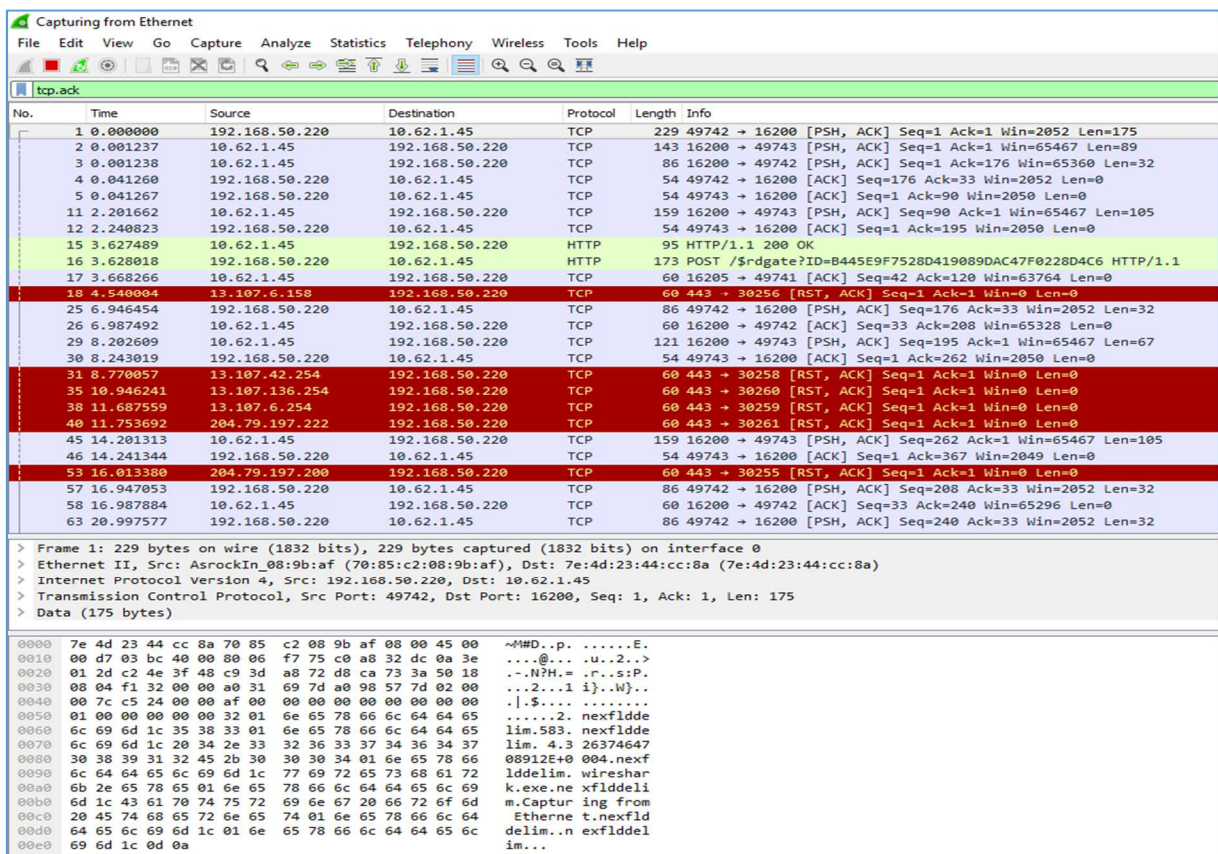
O pacote de sondagem do scan ACK tem apenas a flag ACK marcada (a menos que você use --scanflags). Quando se escaneia sistemas não-filtrados, as portas abertas e fechadas irão devolver um pacote RST. O Nmap então coloca nelas o rótulo não-filtradas (unfiltered), significando que elas estão alcançáveis 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.

4.2.1. Exemplos de uso

-sA (scan TCP ACK)



4.2.2. Captura do escaneamento com WireShark



4.2.3. Captura do escaneamento com Tcpdump

```

root@aluno-DC2C-S:/home/aluno
Arquivo Editar Ver Pesquisar Terminal Ajuda

root@aluno-DC2C-S:/home/aluno# tcpdump -i enp8s0f0 "tcp[tcpflags] & (tcp-ack) != 0"
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp8s0f0, link-type EN10MB (Ethernet), capture size 262144 bytes
19:30:13.837692 IP aluno-DC2C-S.55874 > 186-192-81-62.prt.globo.com.https: Flags [P.], seq 3218028186:3218028224, ack 2410531252, win 245, options [nop,nop,TS val 2693346742 ecr 4082139640], length 38
19:30:13.838050 IP aluno-DC2C-S.55874 > 186-192-81-62.prt.globo.com.https: Flags [P.], seq 38:61, ack 1, win 245, options [nop,nop,TS val 2693346742 ecr 4082139640], length 23
19:30:13.838269 IP aluno-DC2C-S.55874 > 186-192-81-62.prt.globo.com.https: Flags [F.], seq 61, ack 1, win 245, options [nop,nop,TS val 2693346742 ecr 4082139640], length 0
19:30:13.886391 IP 186-192-81-62.prt.globo.com.https > aluno-DC2C-S.55874: Flags [.], ack 62, win 20, options [nop,nop,TS val 4082192883 ecr 2693346742], length 0
19:30:13.886407 IP 186-192-81-62.prt.globo.com.https > aluno-DC2C-S.55874: Flags [F.], seq 1, ack 62, win 20, options [nop,nop,TS val 4082192883 ecr 2693346742], length 0
19:30:13.886417 IP aluno-DC2C-S.55874 > 186-192-81-62.prt.globo.com.https: Flags [.], ack 2, win 245, options [nop,nop,TS val 2693346791 ecr 4082192883], length 0
19:30:17.838799 IP aluno-DC2C-S.39340 > gru10s11-in-f2.1e100.net.https: Flags [P.], seq 2712836620:2712836666, ack 4044944817, win 263, options [nop,nop,TS val 2131635493 ecr 963461180], length 46
19:30:17.838892 IP aluno-DC2C-S.39338 > gru10s11-in-f2.1e100.net.https: Flags [P.], seq 989132319:989132365, ack 2029272081, win 262, options [nop,nop,TS val 2131635493 ecr 2482273702], length 46
19:30:17.839304 IP aluno-DC2C-S.39340 > gru10s11-in-f2.1e100.net.https: Flags [P.], seq 46:77, ack 1, win 263, options [nop,nop,TS val 2131635493 ecr 963461180], length 31
19:30:17.839326 IP aluno-DC2C-S.39340 > gru10s11-in-f2.1e100.net.https: Flags [F.], seq 77, ack 1, win 263, options [nop,nop,TS val 2131635493 ecr 963461180], length 0
19:30:17.839529 IP aluno-DC2C-S.39338 > gru10s11-in-f2.1e100.net.https: Flags [P.], seq 46:77, ack 1, win 262, options [nop,nop,TS val 2131635494 ecr 2482273702], length 31
19:30:17.839544 IP aluno-DC2C-S.39338 > gru10s11-in-f2.1e100.net.https: Flags [F.], seq 77, ack 1, win 262, options [nop,nop,TS val 2131635494 ecr 2482273702], length 0
19:30:17.911405 IP gru10s11-in-f2.1e100.net.https > aluno-DC2C-S.39340: Flags [.], ack 77, win 183, options [nop,nop,TS val 963514423 ecr 2131635493], length 0
19:30:17.911418 IP gru10s11-in-f2.1e100.net.https > aluno-DC2C-S.39340: Flags [F.], seq 1, ack 77, win 183, options [nop,nop,TS val 963514423 ecr 2131635493], length 0
19:30:17.911424 IP aluno-DC2C-S.39340 > gru10s11-in-f2.1e100.net.https: Flags [.], ack 2, win 263, options [nop,nop,TS val 2131635566 ecr 963514423], length 0
19:30:17.911427 IP gru10s11-in-f2.1e100.net.https > aluno-DC2C-S.39340: Flags [.], ack 78, win 183, options [nop,nop,TS val 963514423 ecr 2131635493], length 0
19:30:17.962973 IP aluno-DC2C-S.39338 > gru10s11-in-f2.1e100.net.https: Flags [F.], seq 77, ack 1, win 262, options [nop,nop,TS val 2131635617 ecr 2482273702], length 0
19:30:18.028515 IP gru10s11-in-f2.1e100.net.https > aluno-DC2C-S.39338: Flags [.], ack 78, win 185, options [nop,nop,TS val 2482327069 ecr 2131635617,nop,nop,seq 1 (77:78)], length 0
19:30:18.164495 IP gru10s11-in-f2.1e100.net.https > aluno-DC2C-S.39338: Flags [F.], seq 1, ack 78, win 185, options [nop,nop,TS val 2482327203 ecr 2131635617], length 0
19:30:18.164522 IP aluno-DC2C-S.39338 > gru10s11-in-f2.1e100.net.https: Flags [.], ack 2, win 262, options [nop,nop,TS val 2131635819 ecr 2482327203], length 0
19:30:18.574982 IP aluno-DC2C-S.43308 > gru10s03-in-f206.1e100.net.https: Flags [P.], seq 1443671643:1443671689, ack 2278144654, win 4902, options [nop,nop,TS val 1619988520 ecr 479411969], length 46

21 packets captured
30 packets received by filter
3 packets dropped by kernel
root@aluno-DC2C-S:/home/aluno#

```

4.3. Escaneamento por TCP FIN scan

Esse tipo de scan explora uma brecha sutil na RFC do TCP para diferenciarem entre portas abertas e fechadas. Se a porta [destino] estiver FECHADA, um segmento entrante que não contenha um RST irá causar o envio de um RST como resposta. Então a página seguinte discute os pacotes enviados às portas abertas sem os bits SYN, RST ou ACK marcados, afirmando que: “é pouco provável que você chegue aqui, mas se chegar, descarte o segmento, e volte.”

Quando se escaneia 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. Contanto que nenhum desses três bits estejam incluídos, qualquer combinação dos outros três (FIN, PSH e URG) é válida.

4.3.1. Exemplos de uso

-sF (scans TCP FIN)

4.3.2. Captura do escaneamento com WireShark

*Ethernet						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
tcp_fin_retransmission						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.50.220	10.62.1.45	TCP	222	49742 → 16200 [PSH, ACK] Seq=1 Ack=1 Win=2048 Len=168
2	0.001174	10.62.1.45	192.168.50.220	TCP	141	16200 → 49743 [PSH, ACK] Seq=1 Ack=1 Win=65467 Len=87
3	0.001503	10.62.1.45	192.168.50.220	TCP	86	16200 → 49742 [PSH, ACK] Seq=1 Ack=169 Win=64247 Len=32
4	0.041660	192.168.50.220	10.62.1.45	TCP	54	49742 → 16200 [ACK] Seq=169 Ack=33 Win=2048 Len=0
5	0.041696	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=88 Win=2052 Len=0
6	0.689075	10.62.1.45	192.168.50.220	TCP	197	16200 → 49743 [PSH, ACK] Seq=88 Ack=1 Win=65467 Len=143
7	0.729239	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=231 Win=2052 Len=0
10	1.957339	192.168.50.220	10.62.1.45	TCP	86	49742 → 16200 [PSH, ACK] Seq=169 Ack=33 Win=2048 Len=32
11	1.997847	10.62.1.45	192.168.50.220	TCP	60	16200 → 49742 [ACK] Seq=33 Ack=201 Win=64215 Len=0
12	2.029982	10.62.1.45	192.168.50.220	TCP	214	16200 → 49743 [PSH, ACK] Seq=231 Ack=1 Win=65467 Len=160
13	2.069916	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=391 Win=2051 Len=0
15	2.998095	192.168.50.220	10.62.1.45	TCP	229	49742 → 16200 [PSH, ACK] Seq=201 Ack=33 Win=2048 Len=175
16	2.999220	10.62.1.45	192.168.50.220	TCP	86	16200 → 49742 [PSH, ACK] Seq=33 Ack=376 Win=65535 Len=32
17	2.999521	10.62.1.45	192.168.50.220	TCP	143	16200 → 49743 [PSH, ACK] Seq=391 Ack=1 Win=65467 Len=89
18	3.039463	192.168.50.220	10.62.1.45	TCP	54	49742 → 16200 [ACK] Seq=376 Ack=65 Win=2047 Len=0
19	3.039490	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=480 Win=2051 Len=0
37	8.626139	10.62.1.45	192.168.50.220	HTTP	95	HTTP/1.1 200 OK
38	8.626728	192.168.50.220	10.62.1.45	HTTP	173	POST /\$rdgate?ID=B445E9F7528D419089DAC47F0228D4C6 HTTP/1.1
39	8.667093	10.62.1.45	192.168.50.220	TCP	60	16205 → 49741 [ACK] Seq=42 Ack=120 Win=63407 Len=0
40	9.006043	192.168.50.220	10.62.1.45	TCP	86	49742 → 16200 [PSH, ACK] Seq=376 Ack=65 Win=2047 Len=32
41	9.006611	10.62.1.45	192.168.50.220	TCP	86	16200 → 49742 [PSH, ACK] Seq=65 Ack=408 Win=65503 Len=32
42	9.047092	192.168.50.220	10.62.1.45	TCP	54	49742 → 16200 [ACK] Seq=408 Ack=97 Win=2047 Len=0
54	10.213873	fe80::b566:f3af:45d... fe80::647e:cb32:247...		TCP	86	54143 → 5357 [SYN] Seq=0 Win=64800 Len=0 MSS=1440 WS=256 SACK_PERM=1
55	10.214254	fe80::647e:cb32:247... fe80::b566:f3af:45d...		TCP	86	5357 → 54143 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1440 WS=256 SACK_PERM=1
56	10.214530	fe80::b566:f3af:45d... fe80::647e:cb32:247...		TCP	74	54143 → 5357 [ACK] Seq=1 Ack=1 Win=529920 Len=0
> Frame 1: 222 bytes on wire (1776 bits), 222 bytes captured (1776 bits) on interface 0 > Ethernet II, Src: AsrockIn_08:9b:af (70:85:c2:08:9b:af), Dst: 7e:4d:23:44:cc:8a (7e:4d:23:44:cc:8a) > Internet Protocol Version 4, Src: 192.168.50.220, Dst: 10.62.1.45 > Transmission Control Protocol, Src Port: 49742, Dst Port: 16200, Seq: 1, Ack: 1, Len: 168 > Data (168 bytes)						
0000	7e 4d 23 44 cc 8a 70 85	c2 08 9b af 08 00 45 00	~MHD.p.E.			
0010	00 d0 04 ba 40 00 80 06	f6 7e c0 a8 32 dc 0a 3e	...@... ~..2..>			
0020	01 2d c2 4e 3f 48 c9 3d	b9 0b d8 ca 77 5a 50 18	..N?H.= ...wZP.			
0030	08 00 3c b2 00 00 a0 31	69 7d a0 98 57 7d 02 00	..<...i j}..W}..			
0040	00 7c c5 24 00 00 a8 00	00 00 00 00 00 00 00 00	. \$.			
0050	01 00 00 00 00 00 32 01	6e 65 78 66 6c 64 64 652. nexfldde			
0060	6c 69 6d 1c 35 38 33 01	6e 65 78 66 6c 64 64 65	lim.583. nexfldde			
0070	6c 69 6d 1c 20 34 2e 33	32 36 33 37 35 30 31 31	lim. 4.3 26375011			
0080	36 37 32 34 35 45 2b 30	30 30 34 01 6e 65 78 66	67245E+0 004.nexfl			
0090	6c 64 64 65 6c 69 6d 1c	77 69 72 65 73 68 61 72	lddelim. wireshar			
00a0	6b 2e 65 78 65 01 6e 65	78 66 6c 64 64 65 6c 69	k.exe.ne xflddeli			
00b0	6d 1c 55 6e 73 61 76 65	64 20 70 61 63 6b 65 74	m.Unsave d packet			
00c0	73 85 01 6e 65 78 66 6c	64 64 65 6c 69 6d 1c 01	s..nexfl ddelim..			
00d0	6e 65 78 66 6c 64 64 65	6c 69 6d 1c 0d 0a	nexfldde lim...			

4.3.3. Captura do escaneamento com Tcpcdump

```

root@aluno-DC2C-S: /home/aluno
Arquivo Editar Ver Pesquisar Terminal Ajuda
root@aluno-DC2C-S: /home/aluno# tcpdump -l enp8s0f0 "tcp[tcpflags] & (tcp-fin) != 0"
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp8s0f0, link-type EN10MB (Ethernet), capture size 262144 bytes
19:34:40.572939 IP aluno-DC2C-S.50764 > 151.101.92.175.https: Flags [F.], seq 3377161584, ack 3058444996, win 254, options [nop,nop,TS val 94255794 ecr 575433066], length 0
19:34:40.637660 IP 151.101.92.175.https > aluno-DC2C-S.50764: Flags [F.], seq 32, ack 0, win 63, options [nop,nop,TS val 575433790 ecr 94255794], length 0
19:34:42.574241 IP aluno-DC2C-S.50774 > 151.101.92.175.https: Flags [F.], seq 3621527592, ack 3388075720, win 248, options [nop,nop,TS val 94257796 ecr 574108892], length 0
19:34:42.623007 IP 151.101.92.175.https > aluno-DC2C-S.50774: Flags [F.], seq 32, ack 1, win 60, options [nop,nop,TS val 574109732 ecr 94257796], length 0
19:34:45.575577 IP aluno-DC2C-S.41572 > a104-105-212-212.deploy.static.akamaitechnologies.com.https: Flags [F.], seq 512728936, ack 4028731077, win 245, options [nop,nop,TS val 3334060521 ecr 3075377580], length 0
19:34:45.575763 IP aluno-DC2C-S.41570 > a104-105-212-212.deploy.static.akamaitechnologies.com.https: Flags [F.], seq 506192130, ack 4053479844, win 245, options [nop,nop,TS val 3334060521 ecr 3075377583], length 0
19:34:45.575870 IP aluno-DC2C-S.41574 > a104-105-212-212.deploy.static.akamaitechnologies.com.https: Flags [F.], seq 3927602202, ack 4053280748, win 245, options [nop,nop,TS val 3334060521 ecr 3075377579], length 0
19:34:45.600843 IP a104-105-212-212.deploy.static.akamaitechnologies.com.https > aluno-DC2C-S.41572: Flags [F.], seq 32, ack 1, win 243, options [nop,nop,TS val 3075380357 ecr 3334060521], length 0
19:34:45.607742 IP a104-105-212-212.deploy.static.akamaitechnologies.com.https > aluno-DC2C-S.41570: Flags [F.], seq 32, ack 0, win 244, options [nop,nop,TS val 3075380360 ecr 3334060521], length 0
19:35:31.576604 IP aluno-DC2C-S.55952 > 186-192-81-62.prt.globo.com.https: Flags [F.], seq 2692573964, ack 2410046690, win 562, options [nop,nop,TS val 2693664480 ecr 381766001], length 0
19:35:31.627838 IP 186-192-81-62.prt.globo.com.https > aluno-DC2C-S.55952: Flags [F.], seq 1, ack 0, win 20, options [nop,nop,TS val 381819335 ecr 2693664479], length 0
19:35:34.576780 IP aluno-DC2C-S.40268 > gru10s13-ln-f2.1e100.net.https: Flags [F.], seq 2137950785, ack 4194846908, win 385, options [nop,nop,TS val 207990456 ecr 1871375478], length 0
19:35:34.576863 IP aluno-DC2C-S.40266 > gru10s13-ln-f2.1e100.net.https: Flags [F.], seq 3316609905, ack 3746382303, win 362, options [nop,nop,TS val 207990456 ecr 2776590958], length 0
19:35:34.576955 IP aluno-DC2C-S.39348 > gru10s11-ln-f2.1e100.net.https: Flags [F.], seq 2371944460, ack 2381610317, win 1105, options [nop,nop,TS val 2131952230 ecr 2482589348], length 0
19:35:34.619412 IP gru10s11-ln-f2.1e100.net.https > aluno-DC2C-S.39348: Flags [F.], seq 1, ack 0, win 506, options [nop,nop,TS val 2482643682 ecr 2131952230], length 0
19:35:34.623786 IP gru10s13-ln-f2.1e100.net.https > aluno-DC2C-S.40268: Flags [F.], seq 1, ack 1, win 179, options [nop,nop,TS val 1871428536 ecr 207990456], length 0
19:35:34.623885 IP gru10s13-ln-f2.1e100.net.https > aluno-DC2C-S.40266: Flags [F.], seq 1, ack 1, win 177, options [nop,nop,TS val 2776644290 ecr 207990456], length 0
19:35:35.577210 IP aluno-DC2C-S.56644 > gru10s10-ln-f2.1e100.net.https: Flags [F.], seq 3451444850, ack 874454636, win 237, options [nop,nop,TS val 3619013628 ecr 1981965826], length 0
19:35:35.577593 IP aluno-DC2C-S.42414 > 104.16.14.243.https: Flags [F.], seq 3842211856, ack 1274709327, win 245, length 0
19:35:35.631912 IP gru10s10-ln-f2.1e100.net.https > aluno-DC2C-S.56644: Flags [F.], seq 1, ack 1, win 175, options [nop,nop,TS val 1982019158 ecr 3619013628], length 0
19:35:35.648519 IP 104.16.14.243.https > aluno-DC2C-S.42414: Flags [F.], seq 1, ack 0, win 33, length 0
^C
21 packets captured
21 packets received by filter
0 packets dropped by kernel
root@aluno-DC2C-S: /home/aluno#

```

4.4. Escaneamento por TCP Xmas scan

Esse tipo de scan explora uma brecha sutil na RFC do TCP para diferenciarem entre portas abertas e fechadas. Se a porta [destino] estiver FECHADA, um segmento entrante que não contenha um RST irá causar o envio de um RST como resposta. Então a página seguinte discute os pacotes enviados à portas abertas sem os bits SYN, RST ou ACK marcados, afirmando que: “é pouco provável que você chegue aqui, mas se chegar, descarte o segmento, e volte.”

Quando se escaneia 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. Contudo que nenhum desses três bits estejam incluídos, qualquer combinação dos outros três (FIN, PSH e URG) é válida.

4.4.1. Exemplos de uso

scan Xmas(-sX)

4.4.2. Captura do escaneamento com WireShark

The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains various icons for file operations, capture control, and analysis. The packet list pane shows a series of captured packets, with the selected packet (No. 2) highlighted. The packet details pane shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol fields. The packet bytes pane shows the raw data of the selected packet, with a hex dump and ASCII representation.

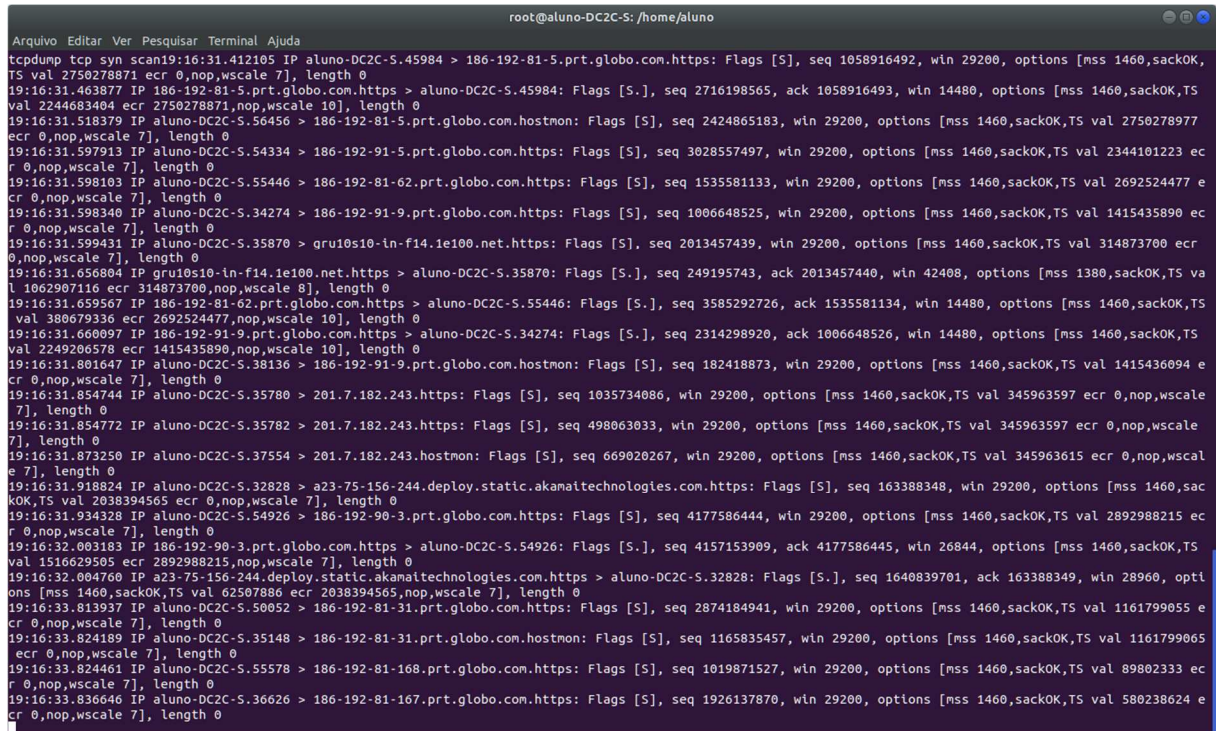
No.	Time	Source	Destination	Protocol	Length	Info
2	0.556004	10.62.1.45	192.168.50.220	TCP	136	16200 → 49743 [PSH, ACK] Seq=1 Ack=1 Win=65467 Len=82
3	0.595915	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=83 Win=2052 Len=0
4	0.653842	192.168.50.220	52.173.28.179	TLSv1.2	127	Application Data
7	0.700771	192.168.50.220	35.186.224.62	TCP	66	30304 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
8	0.723238	35.186.224.62	192.168.50.220	TCP	66	443 → 30304 [SYN, ACK] Seq=0 Ack=1 Win=42780 Len=0 MSS=1380 SACK_PERM=1 WS=256
9	0.723372	192.168.50.220	35.186.224.62	TCP	54	30304 → 443 [ACK] Seq=1 Ack=1 Win=66048 Len=0
10	0.723930	192.168.50.220	35.186.224.62	TLSv1.2	242	Client Hello
11	0.746305	35.186.224.62	192.168.50.220	TCP	60	443 → 30304 [ACK] Seq=1 Ack=189 Win=44032 Len=0
12	0.747871	35.186.224.62	192.168.50.220	TLSv1.2	1484	Server Hello
13	0.748105	35.186.224.62	192.168.50.220	TLSv1.2	1484	Certificate [TCP segment of a reassembled PDU]
14	0.748154	192.168.50.220	35.186.224.62	TCP	54	30304 → 443 [ACK] Seq=189 Ack=2861 Win=66048 Len=0
15	0.748671	35.186.224.62	192.168.50.220	TLSv1.2	71	Server Key Exchange, Server Hello Done
16	0.750015	192.168.50.220	35.186.224.62	TLSv1.2	147	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
17	0.772769	35.186.224.62	192.168.50.220	TLSv1.2	338	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
18	0.773285	192.168.50.220	35.186.224.62	TLSv1.2	229	Application Data
19	0.826756	52.173.28.179	192.168.50.220	TLSv1.2	179	Application Data
20	0.834787	35.186.224.62	192.168.50.220	TCP	60	443 → 30304 [ACK] Seq=3162 Ack=457 Win=45056 Len=0
21	0.866882	192.168.50.220	52.173.28.179	TCP	54	49674 → 443 [ACK] Seq=74 Ack=126 Win=257 Len=0
22	0.940951	35.186.224.62	192.168.50.220	TLSv1.2	1484	Application Data
23	0.941332	35.186.224.62	192.168.50.220	TLSv1.2	444	Application Data
24	0.941360	192.168.50.220	35.186.224.62	TCP	54	30304 → 443 [ACK] Seq=457 Ack=4982 Win=66048 Len=0
28	1.236105	192.168.50.220	23.75.147.137	TCP	66	30305 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
29	1.241093	23.75.147.137	192.168.50.220	TCP	66	443 → 30305 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
30	1.241133	192.168.50.220	23.75.147.137	TCP	54	30305 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
31	1.242634	192.168.50.220	23.75.147.137	TLSv1.2	276	Client Hello

Frame 2: 136 bytes on wire (1088 bits), 136 bytes captured (1088 bits) on interface 0
 Ethernet II, Src: Dell_61:40:ed (f4:8e:38:61:40:ed), Dst: AsrockIn_08:9b:af (70:85:c2:08:9b:af)
 Internet Protocol Version 4, Src: 10.62.1.45, Dst: 192.168.50.220
 Transmission Control Protocol, Src Port: 16200, Dst Port: 49743, Seq: 1, Ack: 1, Len: 82
 Data (82 bytes)

```

0000  70 85 c2 08 9b af f4 8e 38 61 40 ed 08 00 45 00  p.....8a...E.
0010  00 7a 11 dc 40 00 7e 06 eb b2 0a 3e 01 2d c0 a8  .Z..@..>...-..
0020  32 dc 3f 48 c2 4f cf 41 d6 63 26 ab 11 3d 50 18  2.?H.O.A.c&..=P.
0030  ff bb 88 bb 00 00 00 00 00 00 a0 98 57 7d 02 00  .....W}...
0040  00 09 ed 24 00 00 52 00 00 00 00 00 00 00 00 00  ...$.R.....
0050  03 00 00 00 00 00 11 2b 00 00 29 00 00 00 01 00  ....+..).....
0060  00 00 01 31 01 06 0d 50 72 6f 67 72 61 6d 61 41  ....!..P rogramaA
0070  74 75 61 6c 06 10 4e 65 74 42 65 61 6e 73 20 49  tual,.Ne tBeans I
0080  44 45 20 38 2e 32 00 00  DE 8.2..
  
```


4.4.3. Captura do escaneamento com Tcpdump



```

Arquivo Editar Ver Pesquisar Terminal Ajuda
tcpdump tcp syn scan19:16:31.412105 IP aluno-DC2C-S.45984 > 186-192-81-5.prt.globo.com.https: Flags [S], seq 1058916492, win 29200, options [mss 1460,sackOK,
TS val 2750278871 ecr 0,nop,wscale 7], length 0
19:16:31.463877 IP 186-192-81-5.prt.globo.com.https > aluno-DC2C-S.45984: Flags [S.], seq 2716198565, ack 1058916493, win 14480, options [mss 1460,sackOK,TS
val 2244683404 ecr 2750278871,nop,wscale 10], length 0
19:16:31.518379 IP aluno-DC2C-S.56456 > 186-192-81-5.prt.globo.com.hostmon: Flags [S], seq 2424865183, win 29200, options [mss 1460,sackOK,TS val 2750278977
ecr 0,nop,wscale 7], length 0
19:16:31.597913 IP aluno-DC2C-S.54334 > 186-192-91-5.prt.globo.com.https: Flags [S], seq 3028557497, win 29200, options [mss 1460,sackOK,TS val 2344101223 ec
r 0,nop,wscale 7], length 0
19:16:31.598103 IP aluno-DC2C-S.55446 > 186-192-81-62.prt.globo.com.https: Flags [S], seq 1535581133, win 29200, options [mss 1460,sackOK,TS val 2692524477 e
cr 0,nop,wscale 7], length 0
19:16:31.598340 IP aluno-DC2C-S.34274 > 186-192-91-9.prt.globo.com.https: Flags [S], seq 1006648525, win 29200, options [mss 1460,sackOK,TS val 1415435890 ec
r 0,nop,wscale 7], length 0
19:16:31.599431 IP aluno-DC2C-S.35870 > gru10s10-ln-f14.1e100.net.https: Flags [S], seq 2013457439, win 29200, options [mss 1460,sackOK,TS val 314873700 ecr
0,nop,wscale 7], length 0
19:16:31.656804 IP gru10s10-ln-f14.1e100.net.https > aluno-DC2C-S.35870: Flags [S.], seq 249195743, ack 2013457440, win 42408, options [mss 1380,sackOK,TS va
l 1062907116 ecr 314873700,nop,wscale 8], length 0
19:16:31.659567 IP 186-192-81-62.prt.globo.com.https > aluno-DC2C-S.55446: Flags [S.], seq 3585292726, ack 1535581134, win 14480, options [mss 1460,sackOK,TS
val 380679336 ecr 2692524477,nop,wscale 10], length 0
19:16:31.660097 IP 186-192-91-9.prt.globo.com.https > aluno-DC2C-S.34274: Flags [S.], seq 2314298920, ack 1006648526, win 14480, options [mss 1460,sackOK,TS
val 2249206578 ecr 1415435890,nop,wscale 10], length 0
19:16:31.801647 IP aluno-DC2C-S.38136 > 186-192-91-9.prt.globo.com.hostmon: Flags [S], seq 182418873, win 29200, options [mss 1460,sackOK,TS val 1415436094 e
cr 0,nop,wscale 7], length 0
19:16:31.854744 IP aluno-DC2C-S.35780 > 201.7.182.243.https: Flags [S], seq 1035734086, win 29200, options [mss 1460,sackOK,TS val 345963597 ecr 0,nop,wscal
e 7], length 0
19:16:31.854772 IP aluno-DC2C-S.35782 > 201.7.182.243.https: Flags [S], seq 498063033, win 29200, options [mss 1460,sackOK,TS val 345963597 ecr 0,nop,wscal
e 7], length 0
19:16:31.873250 IP aluno-DC2C-S.37554 > 201.7.182.243.hostmon: Flags [S], seq 669020267, win 29200, options [mss 1460,sackOK,TS val 345963615 ecr 0,nop,wscal
e 7], length 0
19:16:31.918824 IP aluno-DC2C-S.32828 > a23-75-156-244.deploy.static.akamaitechnologies.com.https: Flags [S], seq 163388348, win 29200, options [mss 1460,sac
kOK,TS val 2038394565 ecr 0,nop,wscale 7], length 0
19:16:31.934328 IP aluno-DC2C-S.54926 > 186-192-90-3.prt.globo.com.https: Flags [S], seq 4177586444, win 29200, options [mss 1460,sackOK,TS val 2892988215 ec
r 0,nop,wscale 7], length 0
19:16:32.003183 IP 186-192-90-3.prt.globo.com.https > aluno-DC2C-S.54926: Flags [S.], seq 4157153909, ack 4177586445, win 26844, options [mss 1460,sackOK,TS
val 1516629505 ecr 2892988215,nop,wscale 7], length 0
19:16:32.004760 IP a23-75-156-244.deploy.static.akamaitechnologies.com.https > aluno-DC2C-S.32828: Flags [S.], seq 1640839701, ack 163388349, win 28960, opti
ons [mss 1460,sackOK,TS val 62507880 ecr 2038394565,nop,wscale 7], length 0
19:16:33.813937 IP aluno-DC2C-S.50052 > 186-192-81-31.prt.globo.com.https: Flags [S], seq 2874184941, win 29200, options [mss 1460,sackOK,TS val 1161799055 e
cr 0,nop,wscale 7], length 0
19:16:33.824190 IP aluno-DC2C-S.35148 > 186-192-81-31.prt.globo.com.hostmon: Flags [S], seq 1165835457, win 29200, options [mss 1460,sackOK,TS val 1161799065
ecr 0,nop,wscale 7], length 0
19:16:33.824461 IP aluno-DC2C-S.55578 > 186-192-81-168.prt.globo.com.https: Flags [S], seq 1019871527, win 29200, options [mss 1460,sackOK,TS val 89802333 ec
r 0,nop,wscale 7], length 0
19:16:33.836646 IP aluno-DC2C-S.36626 > 186-192-81-167.prt.globo.com.https: Flags [S], seq 1926137870, win 29200, options [mss 1460,sackOK,TS val 580238624 e
cr 0,nop,wscale 7], length 0

```

4.5. 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 averiguar ambos protocolos na mesma execução.

O scan UDP funciona enviando um cabeçalho UDP vazio (sem dados) para cada porta almejada. Se um erro ICMP de porta inalcançável (tipo 3, código 3) é retornado, a porta está fechada. Outros erros do tipo inalcançável (tipo 3, códigos 1, 2, 9, 10, ou 13) marcam a porta como filtrada. Ocasionalmente um serviço irá responder com um pacote UDP, provando que está aberta. Se nenhuma resposta é recebida após as retransmissões, a porta é classificada como aberta|filtrada. Isso significa que a porta poderia estar aberta, ou talvez que filtros de pacotes estejam

bloqueando a comunicação. Scans de versões (-sV) podem ser utilizados para ajudar a diferenciar as portas verdadeiramente abertas das que estão filtradas.

Um grande desafio com o escaneamento UDP é fazê-lo rapidamente. Portas abertas e filtradas raramente enviam alguma resposta, deixando o Nmap esgotar o tempo (time out) e então efetuar retransmissões para o caso de a sondagem ou a resposta ter sido perdida. Portas fechadas são, normalmente, um problema ainda maior. Elas costumam enviar de volta um erro ICMP de porta inalcançável. Mas, ao contrário dos pacotes RST enviados pelas portas TCP fechadas em resposta a um scan SYN ou connect, muitos hosts limitam a taxa de mensagens ICMP de porta inalcançável por padrão. O Linux e o Solaris são particularmente rigorosos quanto a isso. Por exemplo, o kernel 2.4.20 do Linux limita a quantidade de mensagens de destino inalcançável a até uma por segundo (no net/ipv4/icmp.c).

O Nmap detecta a limitação de taxa e diminui o ritmo de acordo para evitar inundar a rede com pacotes inúteis que a máquina-alvo irá descartar. Infelizmente, um limite como o do Linux de um pacote por segundo faz com que um scan de 65.536 portas leve mais de 18 horas. Idéias para acelerar o escaneamento UDP incluem escanear mais hosts em paralelo, fazer um scan rápido apenas das portas mais comuns primeiro, escanear por detrás de um firewall, e utilizar --host-timeout para pular os hosts lentos.

4.5.1. Exemplos de uso

-sU (scans UDP)

4.5.2. Captura do escaneamento com WireShark

WireShark interface showing a packet capture on the Ethernet interface. The capture is filtered by 'tcp.flags.syn'. The packet list shows a series of TCP and TLSv1.2 packets. The selected packet (No. 43) is a TLSv1.2 Application Data packet.

No.	Time	Source	Destination	Protocol	Length	Info
3	0.982609	192.168.50.220	10.62.1.45	TCP	86	49742 → 16200 [PSH, ACK] Seq=1 Ack=1 Win=2048 Len=32
4	1.024042	10.62.1.45	192.168.50.220	TCP	60	16200 → 49742 [ACK] Seq=1 Ack=33 Win=65215 Len=0
5	1.024125	192.168.50.220	10.62.1.45	TCP	365	49742 → 16200 [PSH, ACK] Seq=33 Ack=1 Win=2048 Len=311
6	1.025304	10.62.1.45	192.168.50.220	TCP	141	16200 → 49743 [PSH, ACK] Seq=1 Ack=1 Win=65467 Len=87
7	1.025780	10.62.1.45	192.168.50.220	TCP	86	16200 → 49742 [PSH, ACK] Seq=1 Ack=344 Win=64904 Len=32
8	1.066143	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=88 Win=2052 Len=0
9	1.066164	192.168.50.220	10.62.1.45	TCP	54	49742 → 16200 [ACK] Seq=344 Ack=33 Win=2048 Len=0
10	1.066670	10.62.1.45	192.168.50.220	TCP	143	16200 → 49743 [PSH, ACK] Seq=88 Ack=1 Win=65467 Len=89
11	1.105982	192.168.50.220	10.62.1.45	TCP	54	49743 → 16200 [ACK] Seq=1 Ack=177 Win=2051 Len=0
20	3.451177	192.168.50.220	52.109.120.22	TCP	66	30314 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
21	3.790429	52.109.120.22	192.168.50.220	TCP	66	443 → 30314 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1440 WS=256 SACK_PERM=1
22	3.790585	192.168.50.220	52.109.120.22	TCP	54	30314 → 443 [ACK] Seq=1 Ack=1 Win=66048 Len=0
23	3.791286	192.168.50.220	52.109.120.22	TLSv1.2	244	Client Hello
25	4.132939	52.109.120.22	192.168.50.220	TCP	1514	443 → 30314 [ACK] Seq=1 Ack=191 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
26	4.133278	52.109.120.22	192.168.50.220	TCP	1514	443 → 30314 [ACK] Seq=1461 Ack=191 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
27	4.133359	192.168.50.220	52.109.120.22	TCP	54	30314 → 443 [ACK] Seq=191 Ack=2921 Win=66048 Len=0
28	4.133637	52.109.120.22	192.168.50.220	TCP	1514	443 → 30314 [ACK] Seq=2921 Ack=191 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
29	4.133638	52.109.120.22	192.168.50.220	TLSv1.2	1345	Server Hello, Certificate, Certificate Status, Server Key Exchange, Server Hello Done
30	4.133735	192.168.50.220	52.109.120.22	TCP	54	30314 → 443 [ACK] Seq=191 Ack=5672 Win=66048 Len=0
31	4.144843	192.168.50.220	52.109.120.22	TLSv1.2	268	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
34	4.486304	52.109.120.22	192.168.50.220	TLSv1.2	161	Change Cipher Spec, Encrypted Handshake Message
35	4.487650	192.168.50.220	52.109.120.22	TLSv1.2	459	Application Data
36	4.488093	192.168.50.220	52.109.120.22	TLSv1.2	459	Application Data
38	4.827072	52.109.120.22	192.168.50.220	TCP	60	443 → 30314 [ACK] Seq=5779 Ack=1215 Win=130304 Len=0
43	6.872815	52.109.120.22	192.168.50.220	TLSv1.2	779	Application Data

Frame 3: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
 Ethernet II, Src: AsrockIn_08:9b:af (70:85:c2:08:9b:af), Dst: 7e:4d:23:44:cc:8a (7e:4d:23:44:cc:8a)
 Internet Protocol Version 4, Src: 192.168.50.220, Dst: 10.62.1.45
 Transmission Control Protocol, Src Port: 49742, Dst Port: 16200, Seq: 1, Ack: 1, Len: 32
 Data (32 bytes)

```

0000  7e 4d 23 44 cc 8a 70 85 c2 08 9b af 08 00 45 00  ~#D..p. ....E.
0010  00 48 08 33 40 00 00 06 f3 8d c0 a8 32 dc 0a 3e  .H.3@... ..2..>
0020  01 2d c2 4e 3f 48 c9 3d f6 75 d8 ca 87 fa 50 18  .-.N?H.= .u....P.
0030  08 00 5c 81 00 00 a0 31 69 7d 00 00 00 02 00    ..\....1 i}.....
0040  00 7c 00 00 00 00 20 00 00 00 00 00 00 00 00  .|.... . ....
0050  fe 00 00 00 00 00                                     .....
  
```


4.5.3. Captura do escaneamento com Tcpcmdump

```

root@aluno-DC2C-S: /home/aluno
Arquivo Editar Ver Pesquisar Terminal Ajuda

root@aluno-DC2C-S:/home/aluno# tcpdump -n udp dst portrange 1-1023
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp8s0f0, link-type EN10MB (Ethernet), capture size 262144 bytes
21:04:06.785251 IP 192.168.19.136.43356 > 192.168.19.255.137: NBT UDP PACKET(137): QUERY; REQUEST; BROADCAST
21:04:50.267404 IP 192.168.19.202.138 > 192.168.19.255.138: NBT UDP PACKET(138)
21:06:34.691769 IP 192.168.19.243.68 > 192.168.19.2.67: BOOTP/DHCP, Request from 70:85:c2:08:9a:0f, length 300
21:06:34.692967 IP 192.168.19.2.67 > 192.168.19.243.68: BOOTP/DHCP, Reply, length 300
21:06:34.737195 IP 192.168.19.243.52553 > 192.168.19.2.53: 45194+ [1au] A? horizon.globo.com. (46)
21:06:34.742184 IP 192.168.19.243.57483 > 192.168.19.2.53: 45767+ [1au] A? s3.glbimg.com. (42)
21:06:34.742912 IP 192.168.19.243.33306 > 192.168.19.2.53: 3235+ [1au] AAAA? pagead46.l.doubleclick.net. (55)
21:06:34.743130 IP 192.168.19.243.33664 > 192.168.19.2.53: 61431+ [1au] A? tag.navdmp.com. (43)
21:06:34.743306 IP 192.168.19.243.60984 > 192.168.19.2.53: 37802+ [1au] AAAA? tag.navdmp.com. (43)
21:06:34.767402 IP 192.168.19.243.49062 > 192.168.19.2.53: 41805+ [1au] A? cdn.krxd.net. (44)
21:06:34.767606 IP 192.168.19.243.43616 > 192.168.19.2.53: 12382+ [1au] AAAA? cdn.krxd.net. (41)
21:06:34.783389 IP 192.168.19.243.46705 > 192.168.19.2.53: 37379+ [1au] A? e1879.e7.akamaiedge.net. (52)
21:06:34.783893 IP 192.168.19.243.33078 > 192.168.19.2.53: 54504+ [1au] AAAA? e1879.e7.akamaiedge.net. (52)
21:06:34.809073 IP 192.168.19.243.33532 > 192.168.19.2.53: 63907+ [1au] A? c.apl.globo.com. (44)
21:06:34.809407 IP 192.168.19.243.53082 > 192.168.19.2.53: 8970+ [1au] AAAA? c.apl.globo.com. (44)
21:06:34.819837 IP 192.168.19.243.36735 > 192.168.19.2.53: 65804+ [1au] A? tags.tiqcdn.com. (44)
21:06:34.835047 IP 192.168.19.243.38429 > 192.168.19.2.53: 38158+ [1au] A? apl.globo.com. (42)
21:06:34.862380 IP 192.168.19.243.56084 > 192.168.19.2.53: 39166+ [1au] A? vitrines.globo.com. (47)
21:06:34.896658 IP 192.168.19.243.41584 > 192.168.19.2.53: 27852+ [1au] A? cocoon.globo.com. (45)
21:06:34.900407 IP 192.168.19.243.44491 > 192.168.19.2.53: 16498+ [1au] A? recomendacao.globo.com. (51)
21:06:34.924629 IP 192.168.19.243.42277 > 192.168.19.2.53: 11557+ [1au] AAAA? e8091.b.akamaiedge.net. (51)
21:06:34.937462 IP 192.168.19.243.60679 > 192.168.19.2.53: 60813+ [1au] AAAA? www-google-analytics.l.google.com. (62)
21:06:34.967274 IP 192.168.19.243.35198 > 192.168.19.2.53: 23748+ [1au] A? s04.video.glbimg.com. (49)
21:06:34.996965 IP 192.168.19.243.48203 > 192.168.19.2.53: 22727+ [1au] A? s02.video.glbimg.com. (49)
21:06:35.017592 IP 192.168.19.243.41174 > 192.168.19.2.53: 29654+ [1au] A? s01.video.glbimg.com. (49)
21:06:35.030913 IP 192.168.19.243.37267 > 192.168.19.2.53: 36627+ [1au] A? s03.video.glbimg.com. (49)
21:06:35.062906 IP 192.168.19.243.57240 > 192.168.19.2.53: 35287+ [1au] A? stats.l.doubleclick.net. (52)
21:06:35.063229 IP 192.168.19.243.44054 > 192.168.19.2.53: 46202+ [1au] AAAA? stats.l.doubleclick.net. (52)
21:06:35.064051 IP 192.168.19.243.45658 > 192.168.19.2.53: 51756+ [1au] A? adservice.google.com.br. (52)
21:06:35.064331 IP 192.168.19.243.46508 > 192.168.19.2.53: 38069+ [1au] AAAA? adservice.google.com.br. (52)
21:06:35.072871 IP 192.168.19.243.46273 > 192.168.19.2.53: 32504+ [1au] AAAA? cdn-fastly.krxd.net.c.global-ssl.fastly.net. (72)
21:06:35.072733 IP 192.168.19.243.44978 > 192.168.19.2.53: 2278+ [1au] A? adservice.google.com. (49)
21:06:35.072923 IP 192.168.19.243.49872 > 192.168.19.2.53: 50703+ [1au] AAAA? adservice.google.com. (49)
21:06:35.114958 IP 192.168.19.243.42853 > 192.168.19.2.53: 36855+ [1au] A? partnerad.l.doubleclick.net. (56)
21:06:35.135290 IP 192.168.19.243.40812 > 192.168.19.2.53: 65399+ [1au] AAAA? partnerad.l.doubleclick.net. (56)
21:06:35.135994 IP 192.168.19.243.36804 > 192.168.19.2.53: 22319+ [1au] A? beacon.krxd.net. (44)
21:06:35.136181 IP 192.168.19.243.41745 > 192.168.19.2.53: 43063+ [1au] AAAA? beacon.krxd.net. (44)
21:06:35.161602 IP 192.168.19.243.41904 > 192.168.19.2.53: 49271+ [1au] A? cdn.navdmp.com. (43)
21:06:35.161857 IP 192.168.19.243.46120 > 192.168.19.2.53: 26185+ [1au] AAAA? cdn.navdmp.com. (43)
21:06:35.201558 IP 192.168.19.243.37625 > 192.168.19.2.53: 12773+ [1au] A? star.c10r.facebook.com. (51)
21:06:35.202807 IP 192.168.19.243.51887 > 192.168.19.2.53: 65123+ [1au] AAAA? star.c10r.facebook.com. (51)
21:06:35.216526 IP 192.168.19.243.49283 > 192.168.19.2.53: 61074+ [1au] AAAA? beacon-17-537698933.us-east-1.elb.amazonaws.com. (76)
21:06:36.438421 IP 192.168.19.243.53357 > 192.168.19.2.53: 63893+ [1au] A? scontent.xx.fbcdn.net. (50)
21:06:36.438743 IP 192.168.19.243.40777 > 192.168.19.2.53: 5013+ [1au] AAAA? scontent.xx.fbcdn.net. (50)
21:06:36.447577 IP 192.168.19.243.35558 > 192.168.19.2.53: 5530+ [1au] A? consumer.krxd.net. (46)
21:06:36.447839 IP 192.168.19.243.53400 > 192.168.19.2.53: 24128+ [1au] AAAA? consumer.krxd.net. (46)
21:06:36.506996 IP 192.168.19.243.33333 > 192.168.19.2.53: 1576+ [1au] AAAA? c.global-ssl.fastly.net. (52)
^C
47 packets captured
47 packets received by filter
0 packets dropped by kernel
root@aluno-DC2C-S:/home/aluno#

```

4.6. Escaneamento de versões dos softwares

Aponte o Nmap para uma máquina remota e ele poderá lhe dizer que as portas 25/tcp, 80/tcp e 53/udp estão abertas. Utilizar o banco de dados nmap-services, com cerca de 2.200 serviços bastante conhecidos, do Nmap iria relatar que aquelas portas provavelmente correspondem a um servidor de correio eletrônico (SMTP), a um servidor de páginas web (HTTP) e a um servidor de nomes (DNS) respectivamente. Essa pesquisa normalmente é precisa -- a grande maioria de daemons escutando na porta TCP 25 é, de fato, de servidores de correio eletrônico. Entretanto, você não deveria apostar a sua segurança nesta informação! As pessoas podem e executam serviços em portas estranhas.

Mesmo que o Nmap esteja certo, e o servidor hipotético acima esteja executando os serviços SMTP, HTTP e DNS, isso não é informação o bastante. Quando fizer uma avaliação de vulnerabilidades (ou mesmo um simples inventário da rede) de sua empresa ou clientes, você realmente deseja saber qual o programa-servidor de correio eletrônico ou de nomes e as versões que estão rodando. Ter um

número de versão exato ajuda substancialmente na determinação de quais explorações (exploits) o servidor está vulnerável. A detecção de versão ajuda a obter esta informação.

Depois que as portas TCP e/ou UDP forem descobertas usando qualquer um dos outros métodos de scan, a detecção de versão interroga essas portas para determinar mais informações sobre o que realmente está sendo executado nessas portas. O banco de dados nmap-service-probes do Nmap contém sondagens para pesquisar diversos serviços e expressões de acerto (match expressions) para reconhecer e destrinchar as respostas. O Nmap tenta determinar os protocolos de serviços (p.ex.: ftp, ssh, telnet, http), o nome da aplicação (p.ex.: ISC Bind, Apache httpd, Solaris telnetd), o número da versão, o nome do host, tipo de dispositivo (p.ex.: impressora, roteador), a família do SO (p.ex.: Windows, Linux) e às vezes detalhes diversos do tipo, se um servidor X está aberto para conexões, a versão do protocolo SSH ou o nome do usuário do KaZaA. É claro que a maioria dos serviços não fornece todas essas informações. Se o Nmap foi compilado com o suporte ao OpenSSL, ele irá se conectar aos servidores SSL para deduzir qual o serviço que está escutando por trás da camada criptografada. Quando os serviços RPC são descobertos, o "amolador" de RPC (RPC grinder) do Nmap (-sR) é automaticamente utilizado para determinar o nome do programa RPC e o número da versão. Algumas portas UDP são deixadas no estado aberta|filtrada depois que scan de porta UDP não consegue determinar se a porta está aberta ou filtrada. A detecção de versão irá tentar provocar uma resposta dessas portas (do mesmo jeito que faz com as portas abertas), e alterar o estado para aberta se conseguir. Portas TCP do tipo aberta|filtrada são tratadas da mesma forma. **Note que a opção -A do Nmap** habilita a detecção de versão, entre outras coisas. Um trabalho documentando o funcionamento, uso e customização da detecção de versão está disponível em <http://insecure.org/nmap/vscan/>.

Quando o Nmap recebe uma resposta de um serviço, mas não consegue encontrá-la em seu banco de dados, ele mostra uma identificação (fingerprint) especial e uma URL para que você envie informações se souber com certeza o que está rodando nessa porta. Por favor, considere dispor de alguns minutos para mandar essa informação de forma que sua descoberta possa beneficiar a todos. Graças a esses envios, o Nmap tem cerca de 3.000 padrões de acerto para mais de 350 protocolos, tais como o smtp, ftp, http, etc.

A detecção de versão é habilitada e controlada com as seguintes opções:

`-sV` (detecção de versão)

Habilita a detecção de versão, conforme discutido acima. Alternativamente, você pode usar a opção `-A` para habilitar tanto a detecção de SO como a detecção de versão.

`--allports` (Não exclui nenhuma porta da detecção de versão)

Por padrão, a detecção de versão do Nmap pula a porta TCP 9100 por causa de algumas impressoras que imprimem qualquer coisa que seja enviada para essa porta, levando a dezenas de páginas com requisições HTTP, requisições de sessões SSL binárias, etc. Esse comportamento pode ser alterado modificando-se ou removendo a diretiva `Exclude nonmap-service-probes`, ou você pode especificar `--allports` para escanear todas as portas independente de qualquer diretiva `Exclude`.

`--version-intensity <intensidade>` (Estabelece a intensidade do scan de versão)

Quando está executando um scan de versão (`-sV`), o nmap envia uma série de sondagens, cada qual com um valor atribuído de raridade, entre 1 e 9. As sondagens com números baixos são efetivas contra uma ampla variedade de serviços comuns, enquanto as com números altos são raramente úteis. O nível de intensidade especifica quais sondagens devem ser utilizadas. Quando mais alto o número, maiores as chances de o serviço ser corretamente identificado. Entretanto, scans de alta intensidade levam mais tempo. A intensidade deve estar entre 0 e 9. O padrão é 7. Quando uma sondagem é registrada na porta-alvo através da diretiva `nmap-service-probes ports`, essa sondagem é tentada independentemente do nível de intensidade. Isso assegura que as sondagens DNS sempre serão tentadas contra qualquer porta 53 abertas, e a sondagem SSL será realizada contra a 443, etc.

`--version-light` (Habilita o modo leve (light))

Esse é um apelido conveniente para `--version-intensity 2`. Esse modo leve torna o escaneamento de versão muito mais rápido, mas é ligeiramente menos provável que identifique os serviços.

`--version-all` (Tenta simplesmente todas as sondagens)

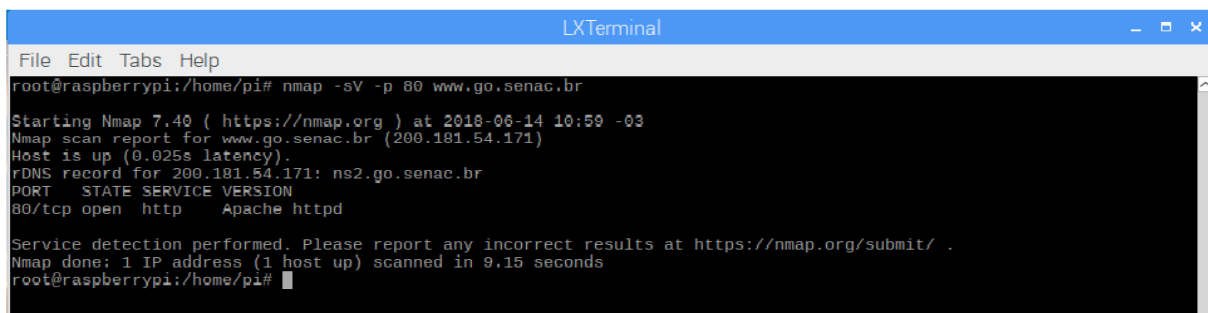
Um apelido para `--version-intensity 9`, assegurando que todas as sondagens sejam tentadas contra cada porta.

`--version-trace` (Monitora as atividades do scan de versão)

Isto faz com que o Nmap mostre informações de depuração extensivas sobre o que o escaneamento de versão está fazendo. É um sub-conjunto do que você obteria com `--packet-trace`.

-sR (Scan RPC)

Este método trabalha em conjunto com os vários métodos de escaneamento de portas do Nmap. Ele pega todas as portas TCP/UDP descobertas no estado aberta e inunda-as com comandos NULL do programa SunRPC, em uma tentativa de determinar se elas são portas RPC e, se forem, quais programas e números de versão elas mostram. Dessa forma você pode obter efetivamente a mesma informação que o `rpcinfo -p` mesmo se o portmapper do alvo estiver atrás de um firewall (ou protegido por TCP wrappers). Chamarizes não funcionam ainda com o scan RPC. Isso é habilitado automaticamente como parte do scan de versão (`-sV`) se você o solicitar. Como a detecção de versão inclui isso e é muito mais abrangente, o `-sR` raramente é necessário.

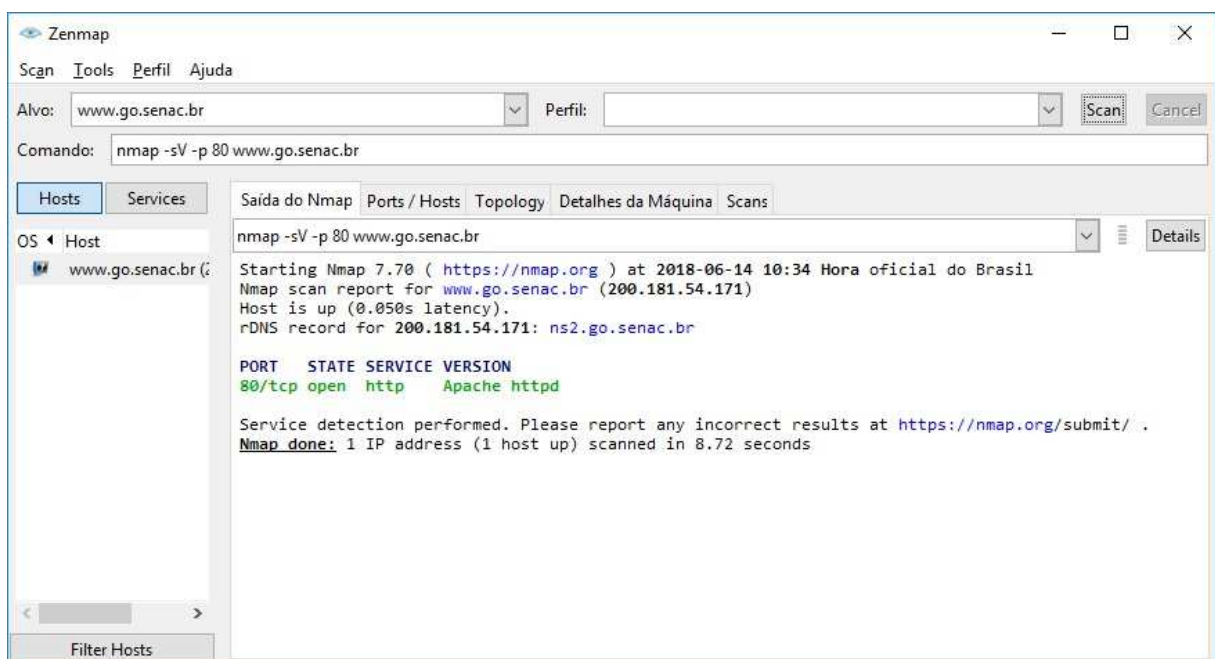


```

LXTerminal
File Edit Tabs Help
root@raspberrypi:/home/pi# nmap -sV -p 80 www.go.senac.br

Starting Nmap 7.40 ( https://nmap.org ) at 2018-06-14 10:59 -03
Nmap scan report for www.go.senac.br (200.181.54.171)
Host is up (0.025s latency).
rDNS record for 200.181.54.171: ns2.go.senac.br
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.15 seconds
root@raspberrypi:/home/pi#
  
```



4.6.1. Exemplos de uso

- sV (detecção de versão)
- allports (Não exclui nenhuma porta da detecção de versão)
- version-intensity <intensidade> (Estabelece a intensidade do scan de versão)
- version-light (Habilita o modo leve (light))
- version-all (Tenta simplesmente todas as sondagens)
- version-trace (Monitora as atividades do scan de versão)
- sR (Scan RPC)

4.6.2. Captura do escaneamento com WireShark

The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains various icons for file operations, capture control, and analysis. The packet list pane shows a series of captured packets, with packet 4148 selected. The packet details pane for packet 4148 shows the following structure:

- Frame 4148: 575 bytes on wire (4600 bits), 575 bytes captured (4600 bits) on interface 0
- Ethernet II, Src: Tp-Link_T29:fd:90 (90:f6:52:29:fd:90), Dst: Dell_bc:fc:e9 (84:8f:69:bc:fc:e9)
- Internet Protocol Version 4, Src: 200.181.54.171, Dst: 192.168.25.100
- Transmission Control Protocol, Src Port: 80, Dst Port: 51478, Seq: 1, Ack: 164, Len: 521
- Hypertext Transfer Protocol
 - HTTP/1.1 302 Found\r\n
 - Date: Thu, 14 Jun 2018 13:33:43 GMT\r\n
 - Server: Apache/2.4.10 (Debian)\r\n
 - Location: https://www.go.senac.br/evox/about\r\n
 - Content-Length: 299\r\n
 - Content-Type: text/html; charset=iso-8859-1\r\n
 - Connection: close\r\n
 - \r\n
 - [HTTP response 1/1]

The packet bytes pane at the bottom shows the raw data in hexadecimal and ASCII. The status bar at the bottom indicates: HTTP Server (http.server), 32 bytes | Packets: 4754 · Displayed: 143 (3.0%) · Dropped: 0 (0.0%) | Profile: Default

4.6.3. Captura do escaneamento com Tcpdump

```
LXTerminal
File Edit Tabs Help
root@raspberrypi:/home/pi# tcpdump -i eth0 -n -s0 -X port 80 and host 200.181.54.171
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

```
LXTerminal
File Edit Tabs Help
ecr 386851050], length 0
 0x0000: 4500 0034 3aa0 4000 4006 26b0 c0a8 196b E..4:..@.&....k
 0x0010: c8b5 36ab 93f4 0050 70db 1a12 1919 828a ..6....Pp.....
 0x0020: 8010 00ed d99a 0000 0101 080a b018 b1db .....
 0x0030: 170e e0ea .....
10:59:42.996050 IP 200.181.54.171.80 > 192.168.25.107.37878: Flags [..], ack 42, win 227, options [nop,nop,TS val 386851070 e
cr 2954408394], length 0
 0x0000: 4500 0034 352e 4000 3906 3322 c8b5 36ab E..45.@.9.3"...6.
 0x0010: c0a8 196b 0050 93f6 37db d5d0 2869 45ab ...k.P..7...(iE.
 0x0020: 8010 00e3 326f 0000 0101 080a 170e e0fe ....2o.....
 0x0030: b018 b1ca .....
10:59:42.998199 IP 200.181.54.171.80 > 192.168.25.107.37878: Flags [P.], seq 1:483, ack 42, win 227, options [nop,nop,TS val
386851070 ecr 2954408394], length 482: HTTP: HTTP/1.1 302 Found
 0x0000: 4500 0216 352f 4000 3906 313f c8b5 36ab E...5/@.9.1?...6.
 0x0010: c0a8 196b 0050 93f6 37db d5d0 2869 45ab ...k.P..7...(iE.
 0x0020: 8018 00e3 02f5 0000 0101 080a 170e e0fe .....
 0x0030: b018 b1ca 4854 5450 2f31 2e31 2033 3032 ....HTTP/1.1.302
 0x0040: 2046 6f75 6e64 0d0a 4461 7465 3a20 5468 .Found..Date:.Th
 0x0050: 752c 2031 3420 4a75 6e20 3230 3138 2031 u,.14.Jun.2018.1
 0x0060: 333a 3538 3a33 3620 474d 540d 0a53 6572 3:58:36.GMT..Ser
 0x0070: 7665 723a 2041 7061 6368 652f 322e 342e ver:.Apache/2.4.
 0x0080: 3130 2028 4465 6269 616e 290d 0a4c 6f63 10.(Debian)..Loc
 0x0090: 6174 696f 6e3a 2068 7474 7073 3a2f 2f77 ation:.https://w
 0x00a0: 7777 2e67 6f2e 7365 6e61 632e 6272 2f0d ww.go.senac.br/.
 0x00b0: 0a43 6f6e 7465 6e74 2d4c 656e 6774 683a .Content-Length:
 0x00c0: 2032 3839 0d0a 436f 6e74 656e 742d 5479 .289..Content-Ty
 0x00d0: 7065 3a20 7465 7874 2f68 746d 6c3b 2063 pe:.text/html;.c
 0x00e0: 6861 7273 6574 3d69 736f 2d38 3835 392d harset=iso-8859-
 0x00f0: 310d 0a0d 0a3c 2144 4f43 5459 5045 2048 1....<!DOCTYPE.H
 0x0100: 544d 4c20 5055 424c 4943 2022 2d2f 2f49 TML.PUBLIC."-//I
 0x0110: 4554 462f 2f44 5444 2048 544d 4c20 322e ETF//DTD.HTML.2.
 0x0120: 302f 2f45 4e22 3e0a 3c68 746d 6c3e 3c68 0//EN">.<html><h
 0x0130: 6561 643e 0a3c 7469 746c 653e 3330 3220 ead>.<title>302.
 0x0140: 466f 756e 643c 2f74 6974 6c65 3e0a 3c2f Found</title>.</
 0x0150: 8865 6164 3e3c 626f 6479 3e0a 3c68 313e head><body>.<h1>
 0x0160: 466f 756e 643c 2f68 313e 0a3c 703e 5468 Found</h1>.<p>Th
 0x0170: 6520 646f 6375 6d65 6e74 2068 6173 206d e.document.has.m
 0x0180: 6f76 6564 203c 6120 6872 6566 3d22 6874 oved.<a.href="ht
 0x0190: 7470 733a 2f2f 7777 772e 676f 2e73 656e tps://www.go.sen
 0x01a0: 6163 2e62 722f 223e 6865 7265 3c2f 613e ac.br/">here</a>
```

5. Conclusão

Conforme descrito no site da Nmap.org a única barreira para a maestria em escaneamento de portas é o conhecimento, utilizando o sistema Nmap Security. Neste trabalho tivemos uma introdução tendo contato com as ferramentas (Nmap, Zenmap, WireShark e TCPdump) e ver as possibilidades de uso das mesmas, cabe a nós, aprofundar a pesquisa e ampliar o conhecimento neste universo em si que é a disciplina fundamentos de redes de computadores.

6. Referências

LYON, Gordon. **NMAP**. 2018. Disponível em: <<https://nmap.org/>>. Acesso em: 12 jun. 2018.

LYON, Gordon Fyodor. **Digitalização em Rede Nmap**. Califórnia: Amazon, 2011. 468 p. Disponível em: <<http://nmap.org/book/>>. Acesso em: 12 jun. 2018.

NMAP.ORG (Califórnia). **O NMAP**. 2018. Disponível em: <<https://nmap.org/>>. Acesso em: 12 jun. 2018.