

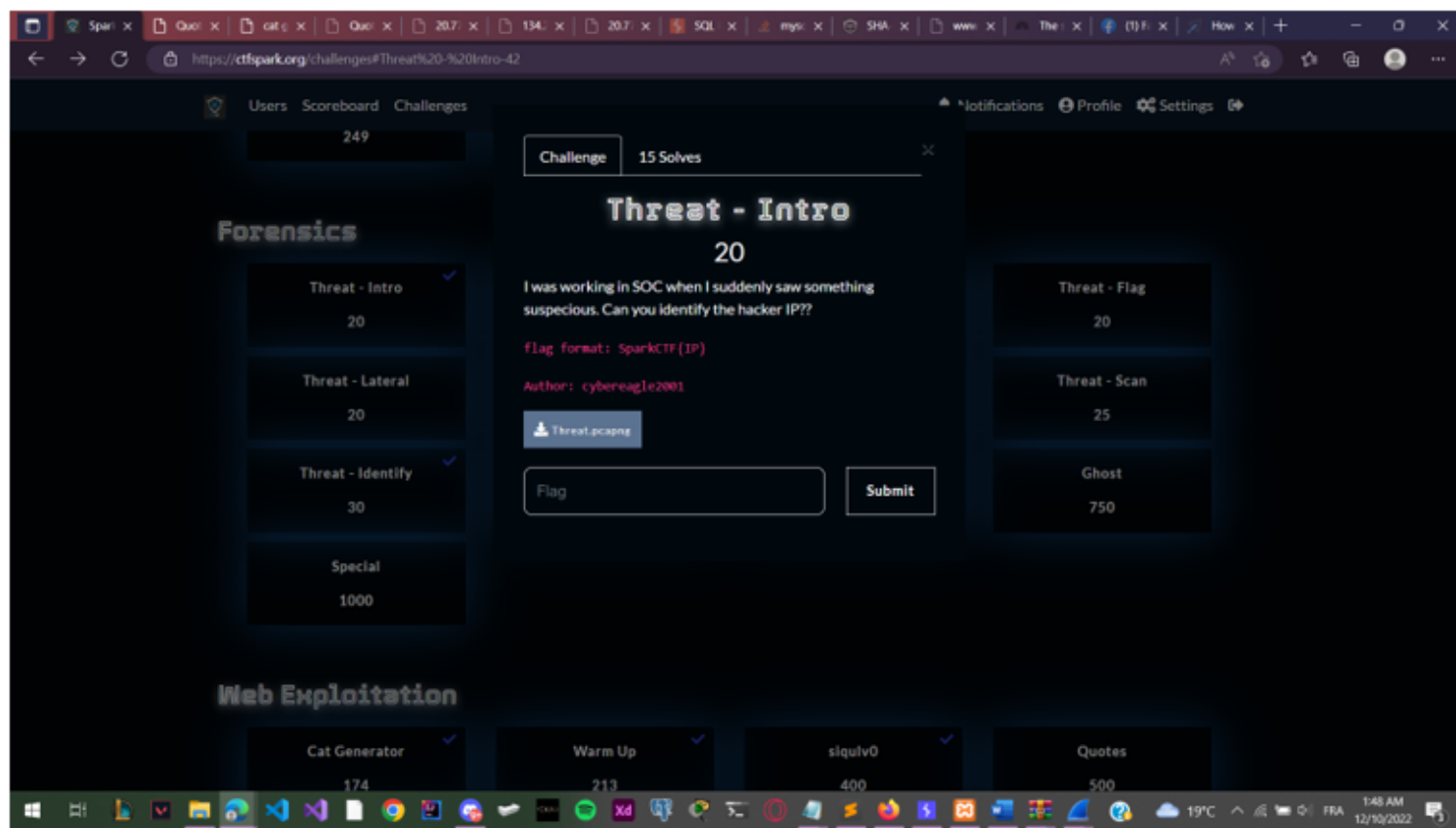
# **Threat - Forensics Series - Writeup**

These are the writeups for my Threat forensics series tasks that were published during SparkCTF 2022. These tasks are based on real life scenarios where the CTF players should have basic understanding of network analysis. To create the tasks I used two VM's (kali linux for the attacker server and CentOS 7 for the victim machine). I used a simple reverse shell written in python and some known utilities like Nmap.

This Event was Organised by [Engineers Spark Community](#) :



## **Threat-Intro**

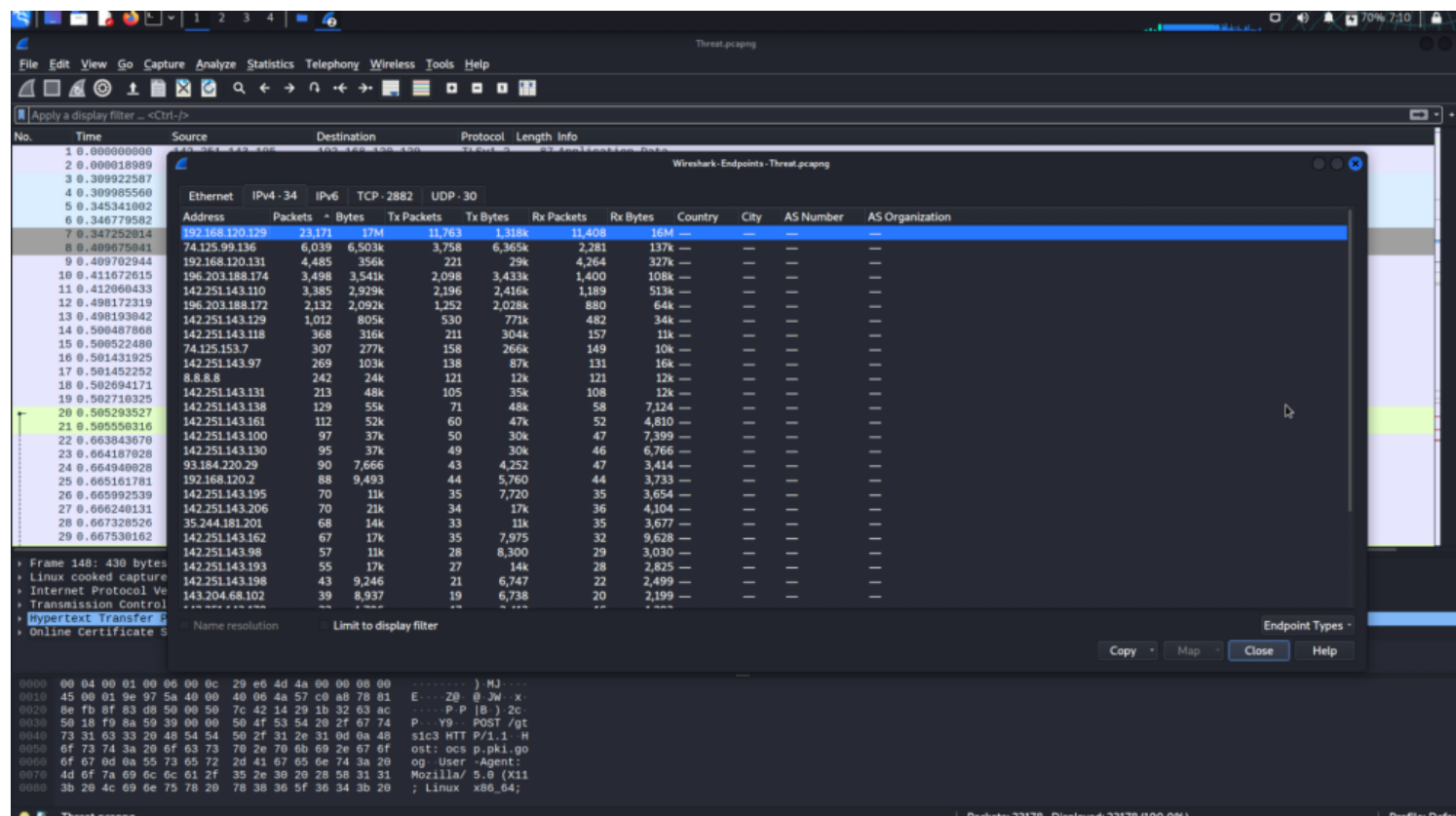


In all the tasks of the Threat Series the players will use the same capture file.

To identify the IP address of the hacker the player must identify the most suspicious IP in the network. It is the most redundant IP in the capture file.

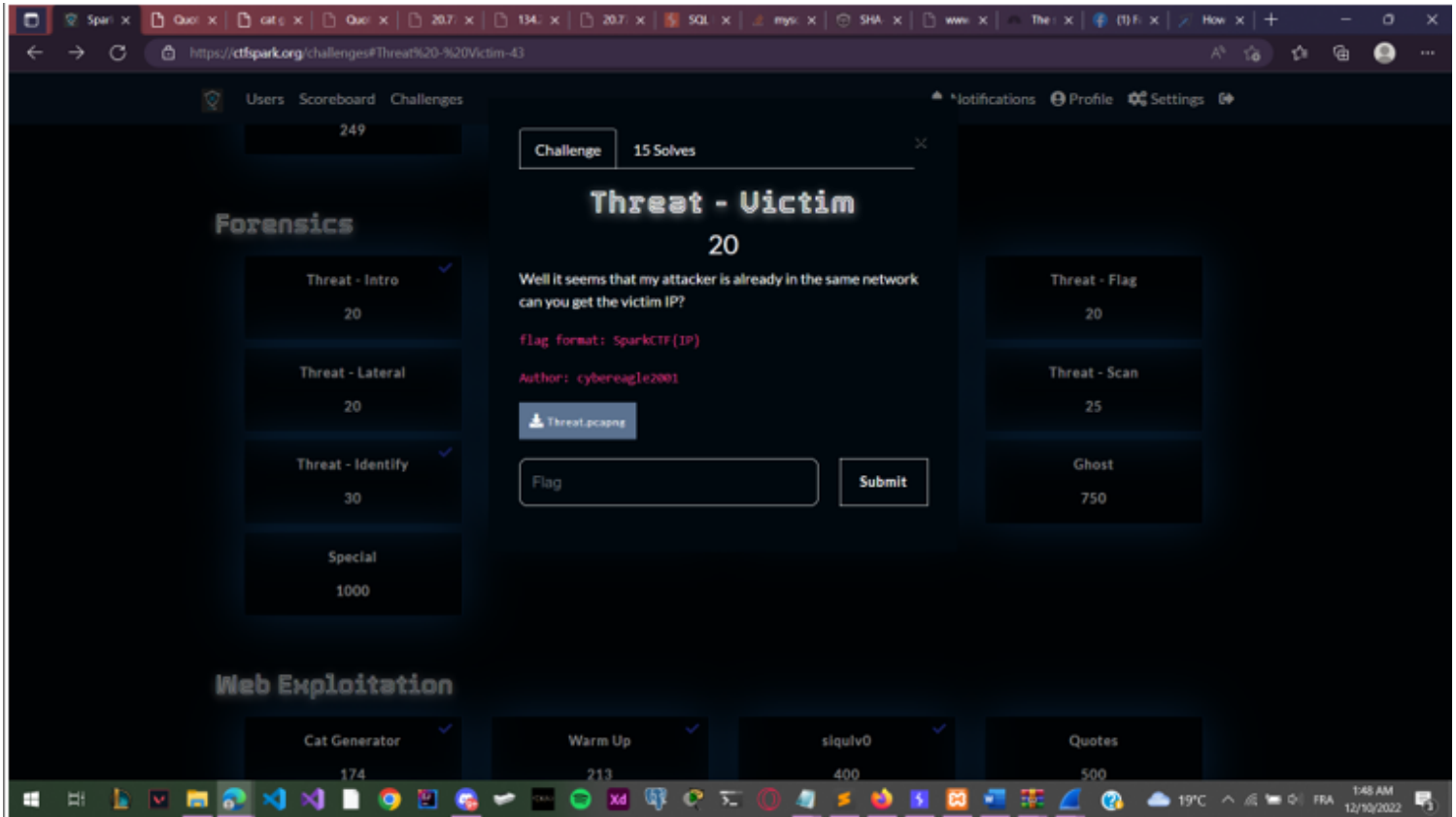
To do so you will need to analyse the endpoints of the capture. You can visit > statistics > endpoints in Wireshark menu. then we will choose IPv4 - 34 and make sure that the number of the packets is decreasing.

We can easily identify the most active IP in the network:



our flag is : SparkCTF{192.168.120.129}

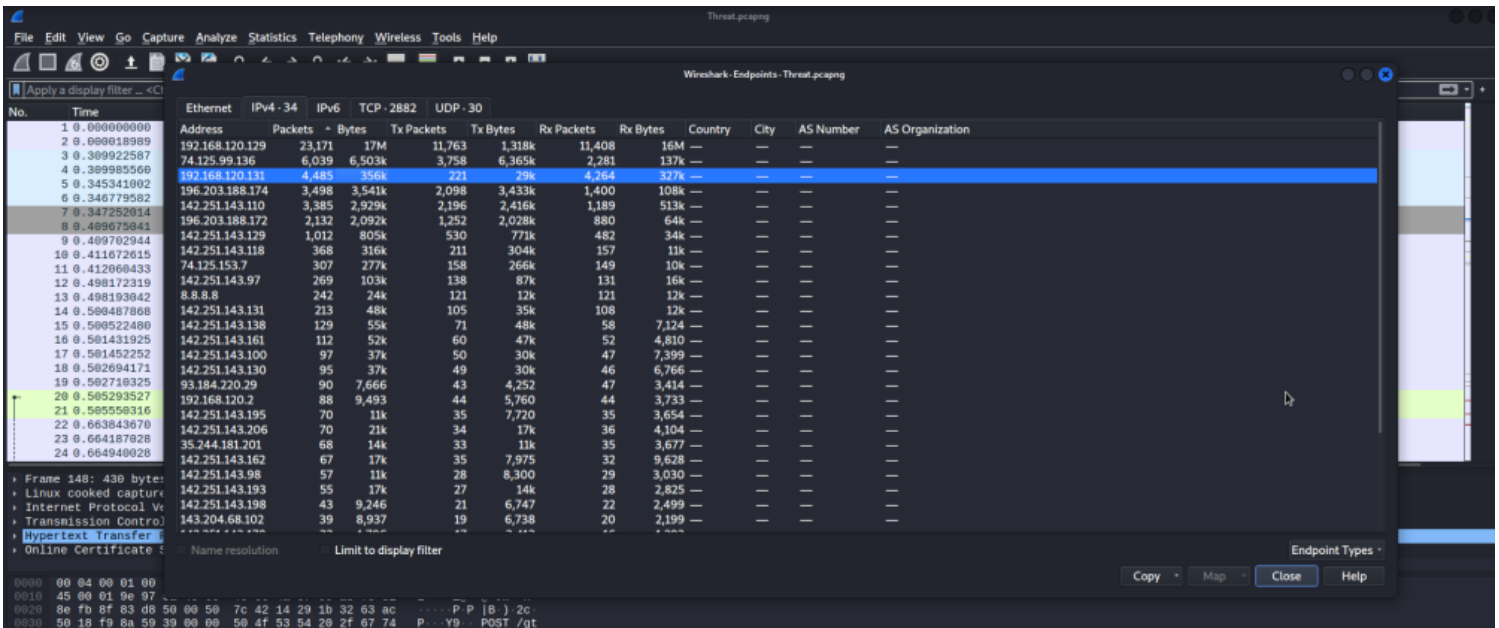
## ***Threat-Victim***



Using the same techniques in the first task the players will identify two Other IP's which are the second and third most active in our network.

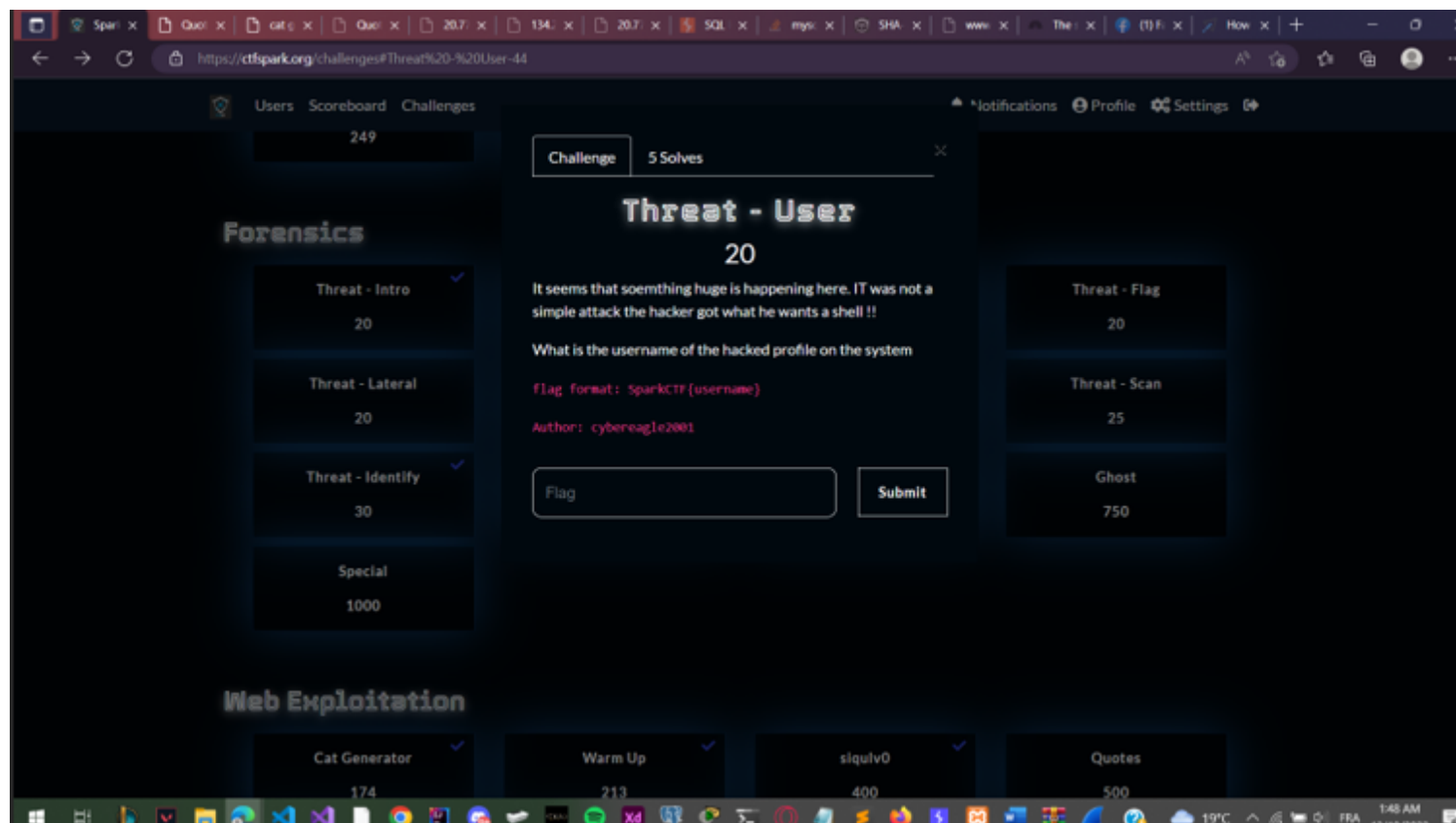
To identify the victim we will use the hint available in the task description. "my attacker is already in the same network".

that means that the victim IP should have the same IP class as our attacker's IP this is why we can neglect the second most active IP address in the capture:



our flag is : SparkCTF{192.168.120.131}

# Threat-user

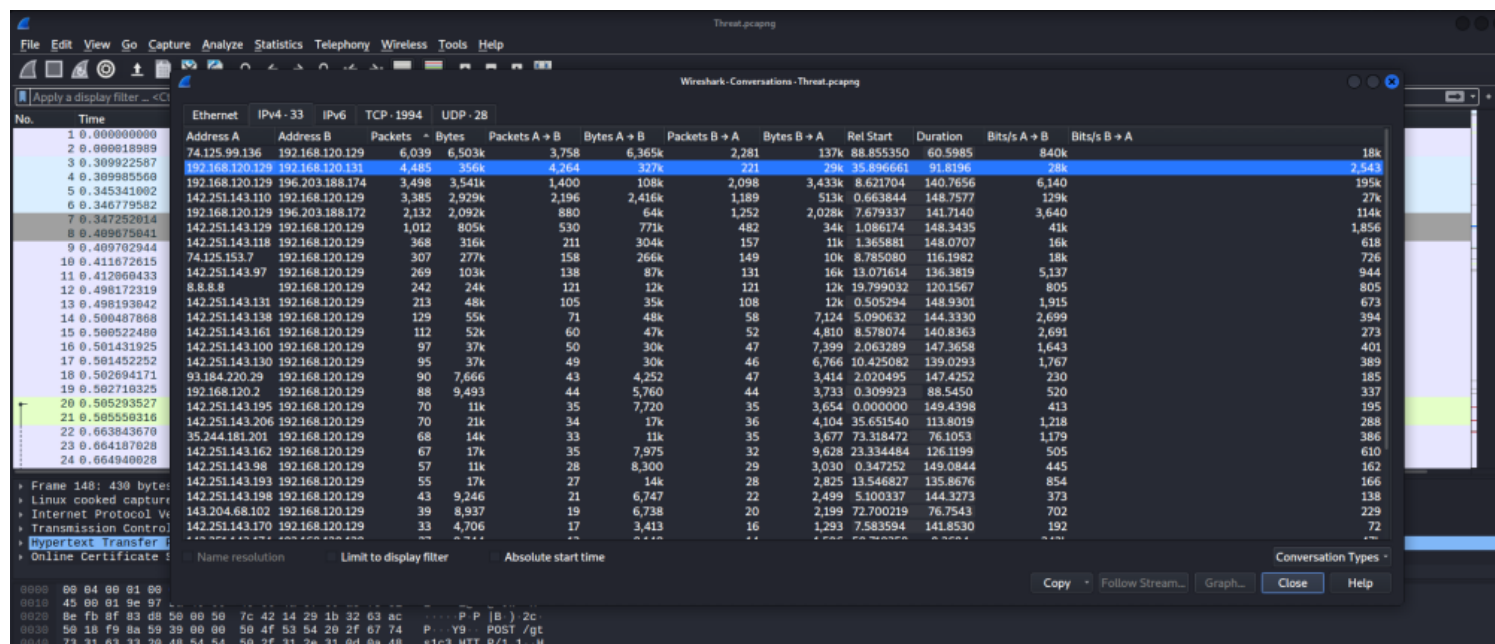


It seems that something huge is happening here, It was not a simple attack. The hacker got what he wanted, a shell!!

The players must identify the victim's username. To do so we need to find the packets which the IP's found in the previous tasks.

We already knew that 192.168.120.129 is the hacker's IP and 192.168.120.131 is the victim's IP. Let's give a look to statistics again and try and understand the conversations we have captured.

> statistics > conversations



the second biggest number of packets is sent from 192.168.120.129 to 192.168.120.131 which is quite logica if we will tie it to the previous tasks.

Let's filter and see the packets where the IP source is 192.168.120.129. we can see that we still have a lot of packets to analyse.



Threat\_captng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.src == 192.168.120.129

No.	Time	Source	Destination	Protocol	Length	Info
5205	13.242702747	192.168.120.129	142.251.143.97	TLSv1.3	174	Application Data
5208	13.244476276	192.168.120.129	142.251.143.97	TCP	56	37128 → 443 [ACK] Seq=518 Ack=5601 Win=63000 Len=0
5210	13.245967867	192.168.120.129	142.251.143.97	TCP	56	37128 → 443 [ACK] Seq=518 Ack=7001 Win=63000 Len=0
5212	13.247206881	192.168.120.129	142.251.143.97	TCP	56	37128 → 443 [ACK] Seq=518 Ack=7710 Win=63000 Len=0
5214	13.249180250	192.168.120.129	142.251.143.97	TCP	56	37126 → 443 [ACK] Seq=518 Ack=1401 Win=63000 Len=0
5215	13.250000278	192.168.120.129	142.251.143.97	TLSv1.3	120	Change Cipher Spec, Application Data
5216	13.250318664	192.168.120.129	142.251.143.97	TLSv1.3	243	Application Data
5219	13.250802170	192.168.120.129	142.251.143.97	TLSv1.3	173	Application Data
5222	13.255076460	192.168.120.129	142.251.143.97	TCP	56	37126 → 443 [ACK] Seq=518 Ack=5601 Win=61320 Len=0
5224	13.259253396	192.168.120.129	142.251.143.97	TCP	56	37126 → 443 [ACK] Seq=518 Ack=7710 Win=62780 Len=0
5225	13.261578295	192.168.120.129	142.251.143.97	TLSv1.3	120	Change Cipher Spec, Application Data
5226	13.261065944	192.168.120.129	142.251.143.97	TLSv1.3	243	Application Data
5229	13.262190027	192.168.120.129	142.251.143.97	TLSv1.3	174	Application Data
5231	13.262358527	192.168.120.129	142.251.143.97	TCP	56	37124 → 443 [ACK] Seq=518 Ack=2001 Win=62780 Len=0
5234	13.263596256	192.168.120.129	142.251.143.97	TCP	56	37124 → 443 [ACK] Seq=518 Ack=5601 Win=62780 Len=0
5236	13.266190350	192.168.120.129	142.251.143.97	TCP	56	37124 → 443 [ACK] Seq=518 Ack=7001 Win=62780 Len=0
5238	13.267038033	192.168.120.129	142.251.143.97	TCP	56	37124 → 443 [ACK] Seq=518 Ack=7710 Win=62780 Len=0
5240	13.267083273	192.168.120.129	142.251.143.97	TCP	56	37130 → 443 [ACK] Seq=518 Ack=1401 Win=63000 Len=0
5241	13.269815984	192.168.120.129	142.251.143.97	TLSv1.3	120	Change Cipher Spec, Application Data
5243	13.270189169	192.168.120.129	142.251.143.97	TLSv1.3	243	Application Data
5245	13.270609350	192.168.120.129	142.251.143.97	TLSv1.3	198	Application Data
5248	13.271499302	192.168.120.129	142.251.143.97	TCP	56	37130 → 443 [ACK] Seq=518 Ack=5601 Win=61320 Len=0
5250	13.272708922	192.168.120.129	142.251.143.97	TCP	56	37130 → 443 [ACK] Seq=518 Ack=7001 Win=62780 Len=0
5252	13.273602241	192.168.120.129	142.251.143.97	TCP	56	37130 → 443 [ACK] Seq=518 Ack=7710 Win=62780 Len=0

Frame 8102: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface any, id 0

- Linux cooked capture v1
- Internet Protocol Version 4, Src: 192.168.120.129, Dst: 192.168.120.131
- Transmission Control Protocol, Src Port: 42610, Dst Port: 139, Seq: 0, Len: 0

```

0000  00 04 00 01 00 05 00 0c 29 e0 4d 4a 93 c2 88 00  .....).MJ....
0010  45 00 00 3c 47 99 40 00 40 06 08 cd c0 a8 78 b1  E...<G@ @...x.
0020  c0 a8 78 b3 a6 72 00 8b eb 28 4c 56 00 00 00 00  x...r...@...
0030  a8 02 fa f0 72 84 00 00 02 a4 05 b4 84 02 00 0a  r...r...
0040  63 cf a6 90 00 00 00 01 03 03 07  .....C.....

```

this is why we will need to combine our filter with the destination IP in order to retrieve the only communication we have between the hacker and the victim:

Threat\_captng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.src == 192.168.120.129 && ip.dst == 192.168.120.131

No.	Time	Source	Destination	Protocol	Length	Info
8108	35.896933241	192.168.120.129	192.168.120.131	TCP	76	41792 → 25 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552984 TSecr=0 WS=128
8109	35.896977656	192.168.120.129	192.168.120.131	TCP	76	56928 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552984 TSecr=0 WS=128
8110	35.897000703	192.168.120.129	192.168.120.131	ICMP	104	Destination unreachable (Host administratively prohibited)
8111	35.897019507	192.168.120.129	192.168.120.131	TCP	76	53034 → 507 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552984 TSecr=0 WS=128
8112	35.897059533	192.168.120.129	192.168.120.131	TCP	76	47380 → 1723 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552984 TSecr=0 WS=128
8113	35.897084924	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
8114	35.897084908	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
8115	35.897140899	192.168.120.129	192.168.120.131	TCP	76	39018 → 995 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552984 TSecr=0 WS=128
8116	35.897155182	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
8117	35.897155234	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
8118	35.897189611	192.168.120.129	192.168.120.131	TCP	76	50646 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552984 TSecr=0 WS=128
8119	35.897210303	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
8120	35.897229336	192.168.120.129	192.168.120.131	TCP	76	45358 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8121	35.897273236	192.168.120.129	192.168.120.131	TCP	76	38492 → 143 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8122	35.897338980	192.168.120.129	192.168.120.131	TCP	76	50184 → 3389 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8123	35.897376164	192.168.120.129	192.168.120.131	TCP	76	60918 → 5900 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8125	35.897449770	192.168.120.129	192.168.120.131	TCP	76	55286 → 3396 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8126	35.897471283	192.168.120.129	192.168.120.131	TCP	68	50646 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1674552985 TSecr=1505788
8127	35.897562723	192.168.120.129	192.168.120.131	TCP	76	56100 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8128	35.897597833	192.168.120.129	192.168.120.131	TCP	76	45976 → 199 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8130	35.897583204	192.168.120.129	192.168.120.131	TCP	68	45358 → 21 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1674552985 TSecr=1505788
8131	35.897607108	192.168.120.129	192.168.120.131	TCP	76	36626 → 993 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8132	35.897608120	192.168.120.129	192.168.120.131	TCP	76	46996 → 113 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128
8133	35.897624166	192.168.120.129	192.168.120.131	TCP	76	55186 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1674552985 TSecr=0 WS=128

Frame 8110: 104 bytes on wire (832 bits), 104 bytes captured (832 bits) on interface any, id 0

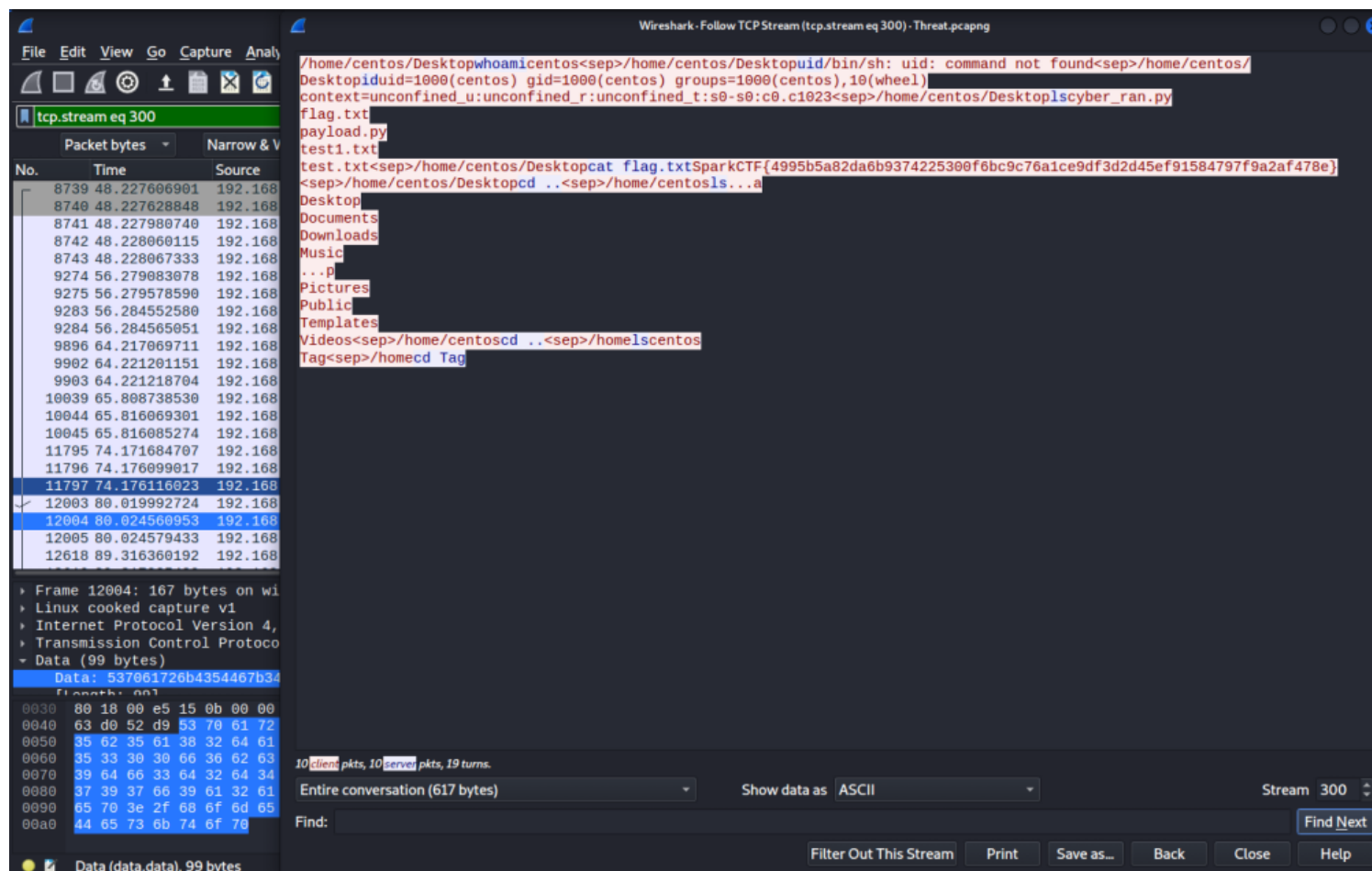
- Linux cooked capture v1
- Internet Protocol Version 4, Src: 192.168.120.131, Dst: 192.168.120.129
- Internet Control Message Protocol

```

0000  00 00 00 01 00 05 00 0c 29 2e 5c 41 f6 93 88 00  .....).VA....
0010  45 c0 00 58 a5 a0 00 00 40 01 01 ef c0 a8 78 b3  E...x...@ a...x.
0020  c0 a8 78 b1 03 0a 0f 7a 00 00 00 00 45 00 00 3c  x...oz...E...<
0030  47 99 40 00 40 00 00 cd c0 a8 78 b1 c0 a8 78 b3  G @ @...x...x.
0040  a0 72 00 8b eb 28 4c 56 00 00 00 00 a0 02 fa f0  r...r...

```

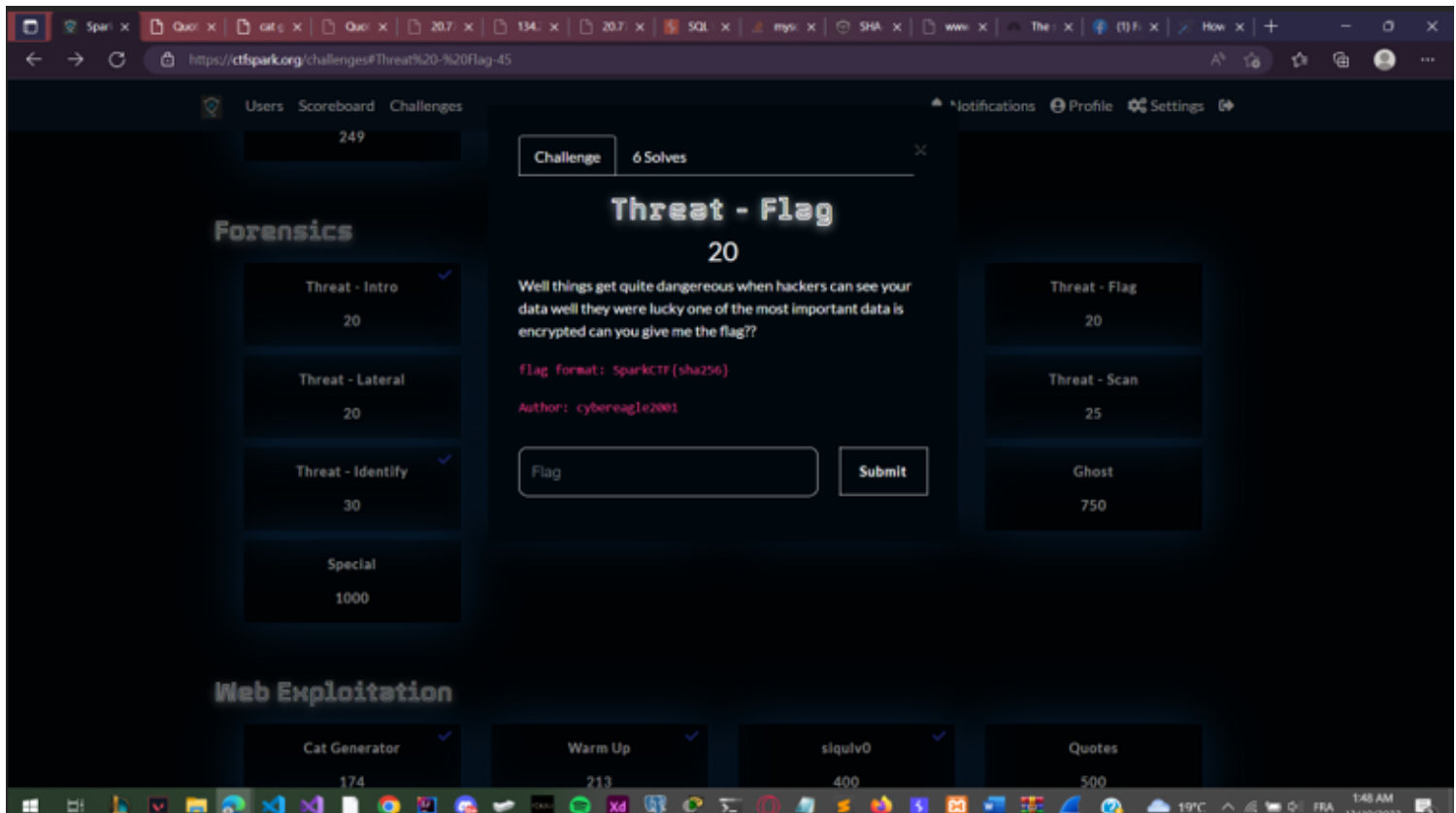
Things are getting more and more promising with fewer packets and less protocols. Let's follow the TCP streams and try to find if we captured a shell communication.



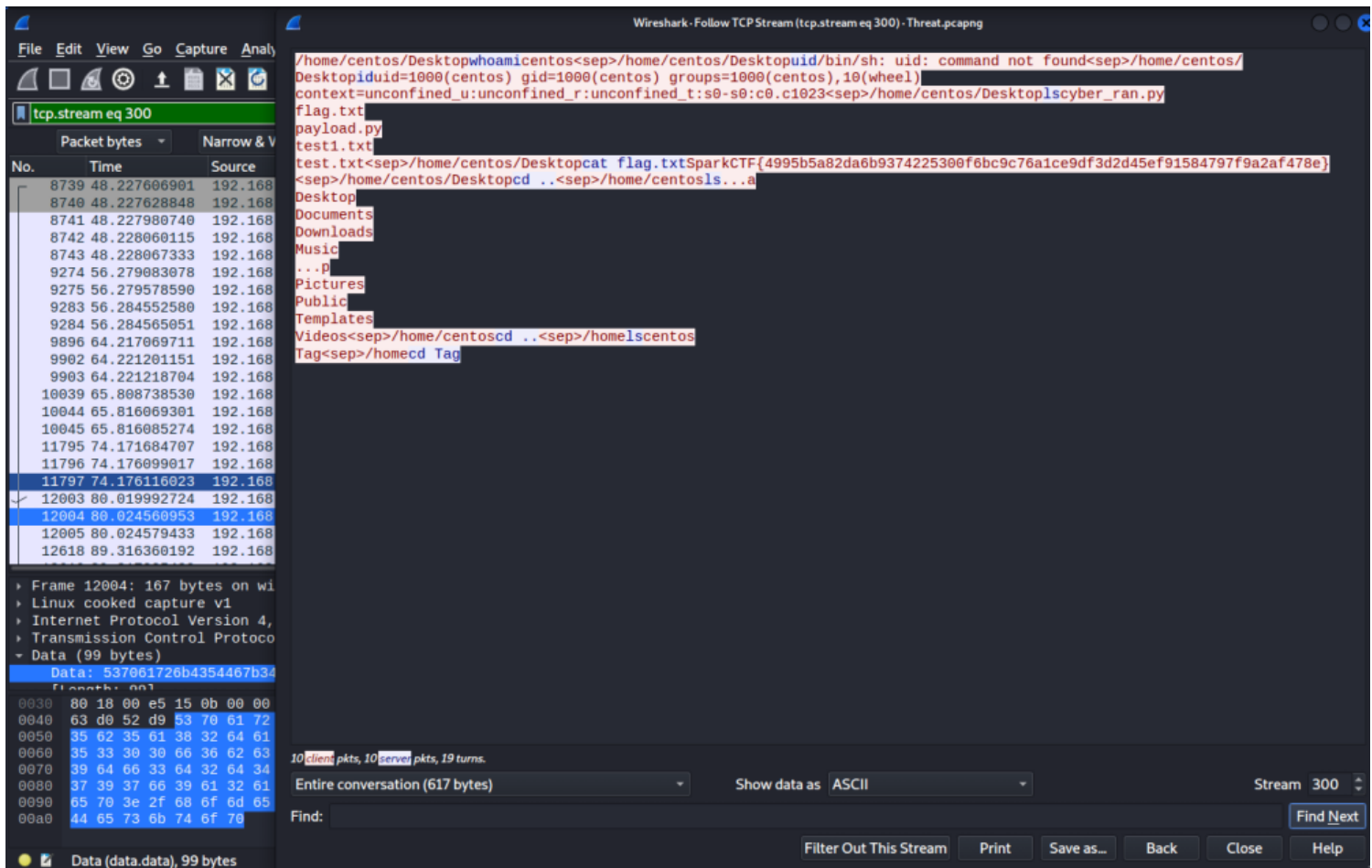
you can see that the hacker executed "whoami" on the victims machine and got "centos" as response.

The flag is : SparkCTF{centos}

## ***Threat-Flag***



This task is quite easy if we have already solved the user task.  
Examining the shell commands we will see that the hacker run the following : cat flag.txt



the flag is :  
SparkCTF{4995b5a82da6b9374225300f6bc9c76a1ce9df3d2d45ef91584797f9a2af478e}

## Threat-Lateral

Challenge

0 Solves

×

Threat - Lateral

20

can you get the username of the second computer user ??

flag format:SparkCTF{username}

Author: cybereagle2001

Flag

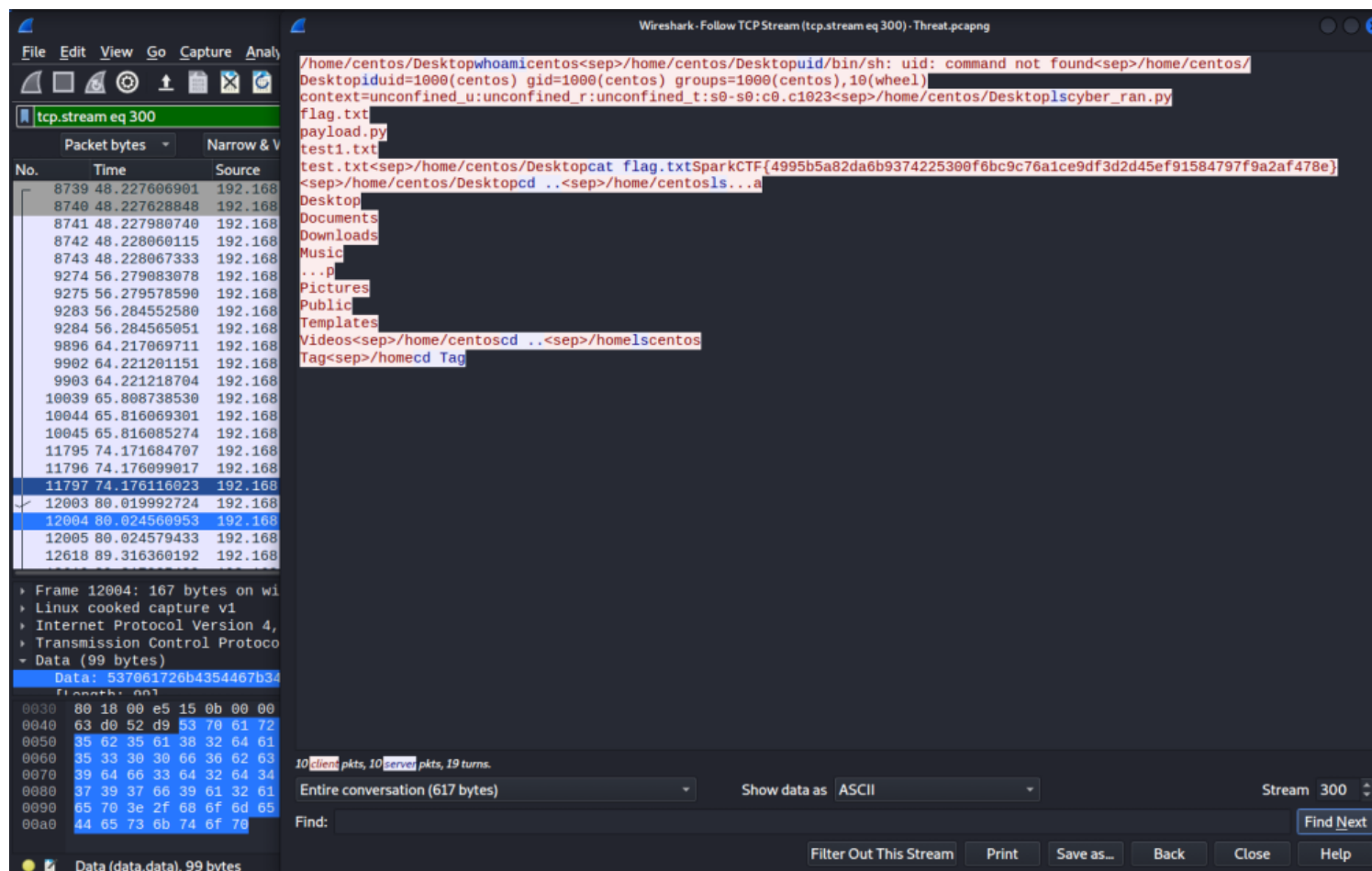
Submit

Lateral movement refers to the techniques that a cyberattacker uses, after gaining initial access, to move deeper into a network in search of sensitive data and other high-value assets. So let's see if our hacker

tried to reach another user account.

This task is also easy if we have already solved the previous tasks.

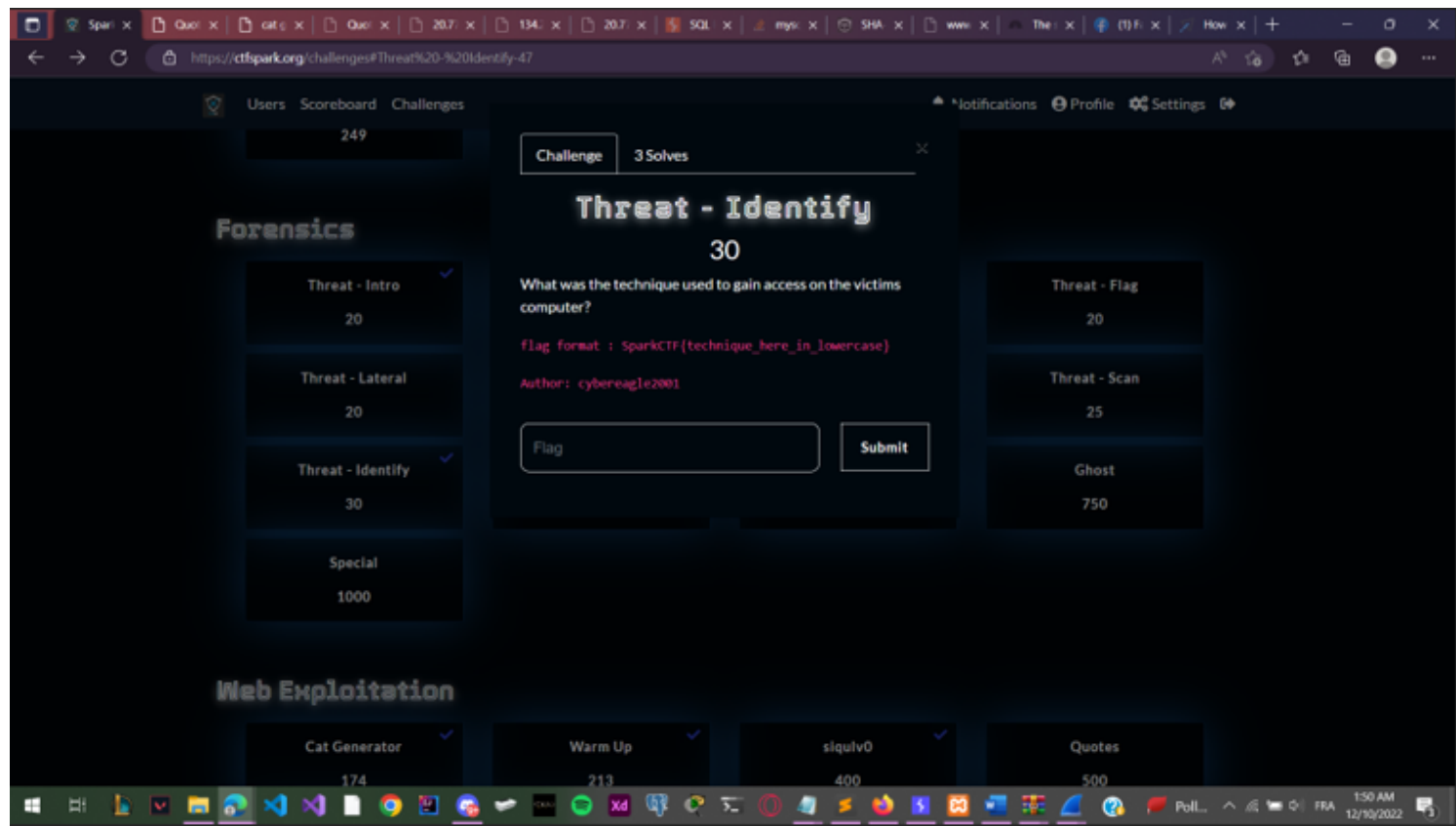




we can see that our hacker wanted to gain access to Tag /home directory.

Flag : SparkCTF{Tag}

## Threat-Identify



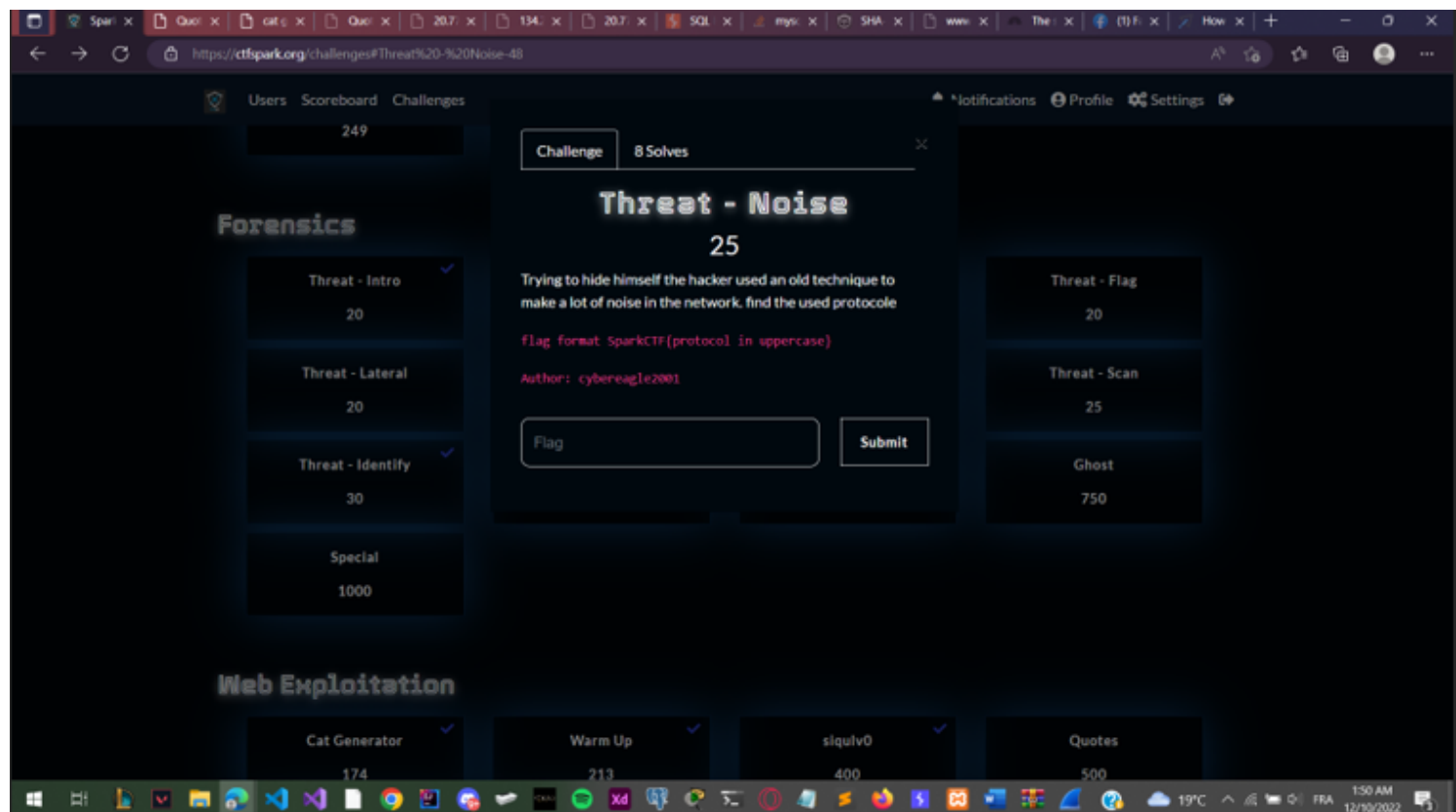
This is not a technical challenge, it's more about your knowledge. What was the technique used to gain access.

The hacker actually had the chance to execute commands on the victim computer but we can see that the communication was established from the victim IP.

this is what it's called reverse shell

Flag: SparkCTF{reverse\_shell}

## ***Threat-Noise***



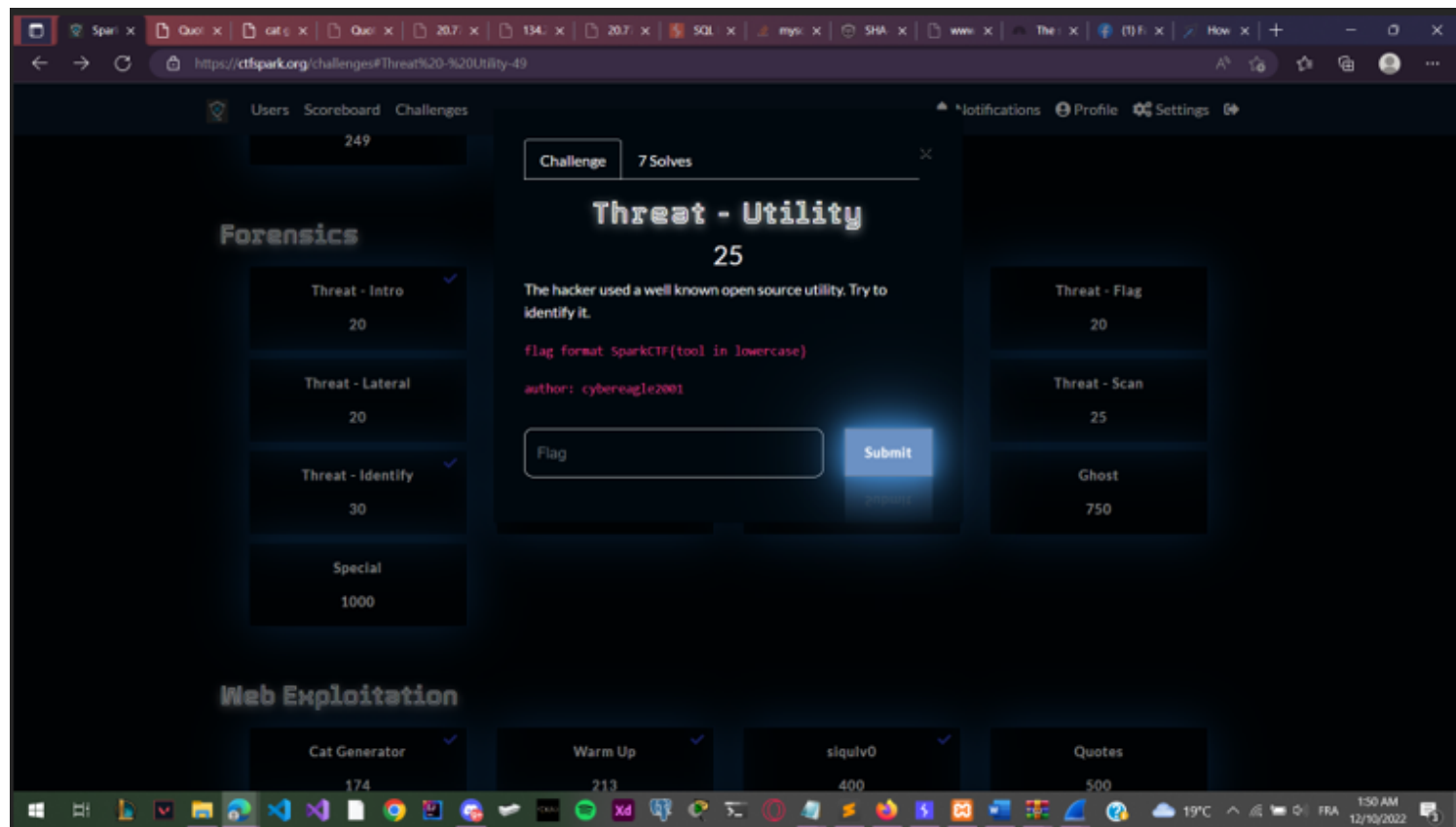
To hide himself the hacker used an old technique.

This is what the description says. But let's go back and see why our network capture is quite big. If you will analyse the file precisely you can easily identify the huge ammount of ping request sent from the hacker ip.

No.	Time	Source	Destination	Protocol	Length	Info
11960	77.867526181	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=59/15104, ttl=64 (reply in 11961)
11961	77.925539210	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=59/15104, ttl=128 (request in 11960)
11964	78.068173371	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
11969	78.867545900	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=60/15360, ttl=64 (reply in 11970)
11970	78.925841784	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=60/15360, ttl=128 (request in 11969)
11974	79.092302447	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12081	79.867534212	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=61/15616, ttl=64 (reply in 12082)
12082	79.925929452	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=61/15616, ttl=128 (request in 12081)
12087	80.116424321	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12012	80.867389162	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=62/15872, ttl=64 (reply in 12013)
12013	80.925566640	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=62/15872, ttl=128 (request in 12012)
12016	81.148127855	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12081	81.867484102	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=63/16128, ttl=64 (reply in 12084)
12084	81.928335959	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=63/16128, ttl=128 (request in 12081)
12088	82.022259324	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12093	82.867549211	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=64/16384, ttl=64 (reply in 12094)
12094	82.925598578	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=64/16384, ttl=128 (request in 12093)
12098	83.068232207	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12104	83.868418846	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=65/16640, ttl=64 (reply in 12105)
12105	83.925907360	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=65/16640, ttl=128 (request in 12104)
12108	84.084891941	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12117	84.869265579	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=66/16896, ttl=64 (reply in 12118)
12118	84.928180120	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=66/16896, ttl=128 (request in 12117)
12197	85.031516710	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12202	85.869894612	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=67/17152, ttl=64 (reply in 12205)
12205	85.928327214	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=67/17152, ttl=128 (request in 12202)
12209	86.064149630	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12214	86.876277126	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=68/17408, ttl=64 (reply in 12217)
12217	86.927916774	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=68/17408, ttl=128 (request in 12214)
12219	87.028145884	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)
12226	87.871345965	192.168.120.129	8.8.8.8	ICMP	100	Echo (ping) request id=0x181b, seq=69/17664, ttl=64 (reply in 12228)
12228	87.928885965	8.8.8.8	192.168.120.129	ICMP	100	Echo (ping) reply id=0x181b, seq=69/17664, ttl=128 (request in 12226)
12238	87.958007603	192.168.120.131	192.168.120.129	ICMP	104	Destination unreachable (Host administratively prohibited)

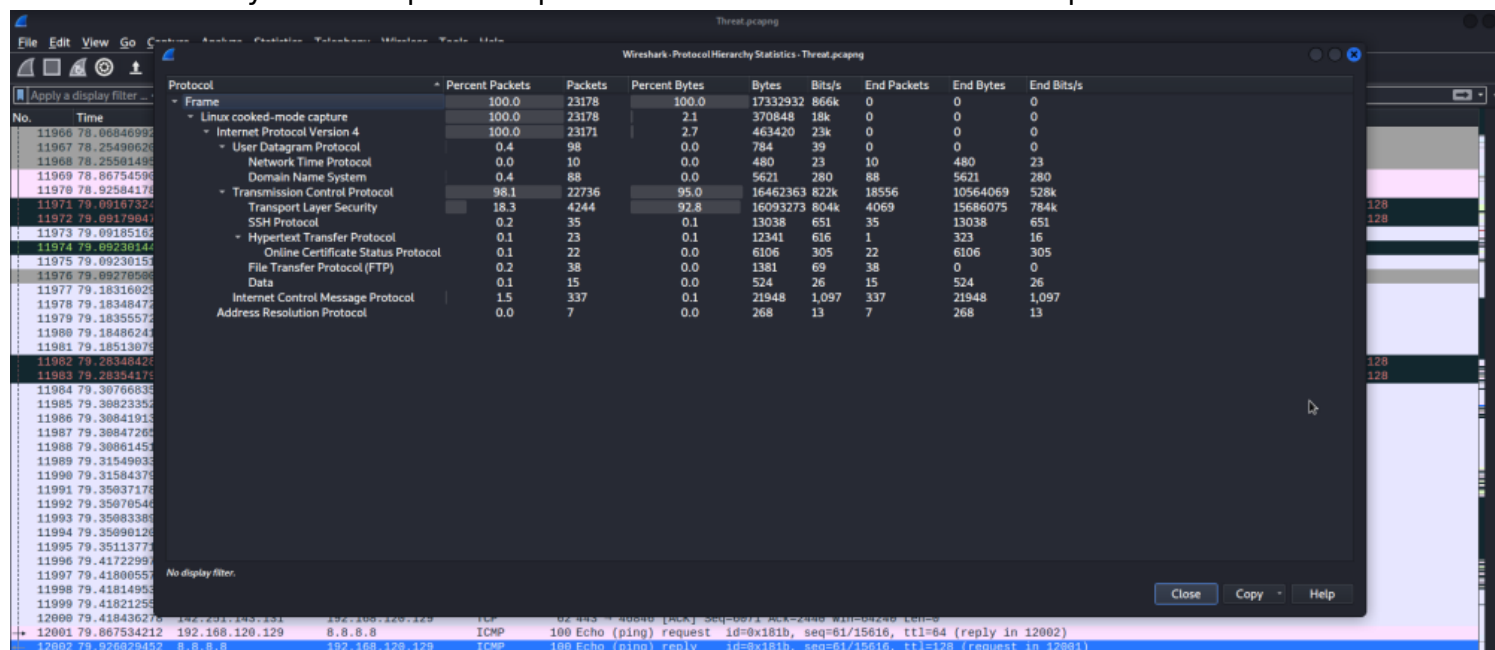
flag: SparkCTF{ICMP}

## Threat-Utility



A well known utility used during the attacks???? Le's see what the network capture is hiding!

first we will analyse all the possible protocols that are available at our capture.



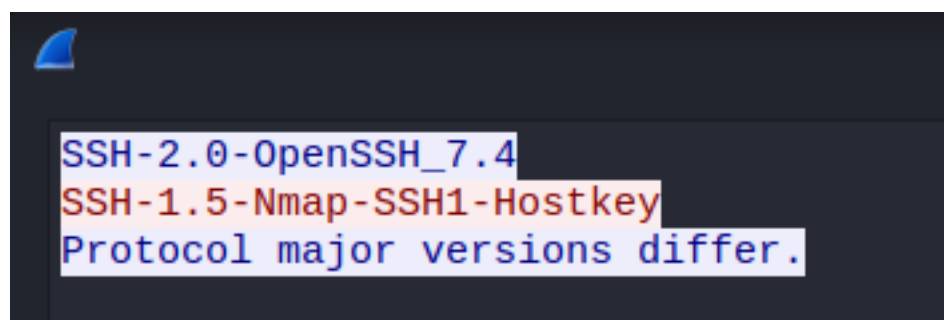
the most attractive ones are the FTP and the SSH. Why?? Well because the attacker didn't use SSH as a protocol to gain access to the shell and he didn't use the FTP in order to tranfert any type of files. Let's analyse them one by one to try and understand why they are present in our capture:

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 top toolbar contains icons for various functions like opening files, saving, and filtering. The main window is divided into three panes: Packet List, Packet Details, and Packet Bytes.

The Packet List pane shows a list of captured packets. The selected packet is 20483, which is an SSH-2.0-OpenSSH\_7.4 Server: Protocol. The Packet Details pane shows the structure of the selected packet, which is an SSH-2.0-OpenSSH\_7.4 Server: Protocol. The Packet Bytes pane shows the raw data of the packet.

No.	Time	Source	Destination	Protocol	Length	Info
20392	126.596596320	192.168.120.131	192.168.120.129	SSHv1	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20392	126.596596320	192.168.120.131	192.168.120.129	SSHv1	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20394	126.612512333	192.168.120.131	192.168.120.129	SSHv1	95	Client: Protocol (SSH-1.5-Nnap-SSH1-Hostkey)
20404	126.616778137	192.168.120.129	192.168.120.131	SSHv1	88	Client: Protocol (SSH-1.5-Nnap-SSH1-Hostkey)
20407	126.616945769	192.168.120.129	192.168.120.131	SSHv2	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20443	126.725645628	192.168.120.131	192.168.120.129	SSHv2	95	Client: Protocol (SSH-2.0-Nnap-SSH2-Hostkey)
20445	126.817136104	192.168.120.129	192.168.120.131	SSHv2	1348	Server: Key Exchange Init
20447	126.825250172	192.168.120.131	192.168.120.129	SSHv2	580	Client: Key Exchange Init
20449	126.866649557	192.168.120.129	192.168.120.131	SSHv2	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20459	126.974092556	192.168.120.131	192.168.120.129	SSHv2	95	Client: Protocol (SSH-2.0-Nnap-SSH2-Hostkey)
20461	127.064211722	192.168.120.129	192.168.120.131	SSHv2	1348	Server: Key Exchange Init
20463	127.066626681	192.168.120.131	192.168.120.129	SSHv2	580	Client: Key Exchange Init
20465	127.113904571	192.168.120.129	192.168.120.131	SSHv2	212	Client: Diffie-Hellman Key Exchange Init
20467	127.213221332	192.168.120.129	192.168.120.131	SSHv2	788	Server: Diffie-Hellman Key Exchange Reply, New Keys
20469	127.215323004	192.168.120.131	192.168.120.129	SSHv2	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20477	127.325456442	192.168.120.131	192.168.120.129	SSHv2	95	Client: Protocol (SSH-2.0-Nnap-SSH2-Hostkey)
20479	127.412086049	192.168.120.129	192.168.120.131	SSHv2	1348	Server: Key Exchange Init
20481	127.415235946	192.168.120.131	192.168.120.129	SSHv2	596	Client: Key Exchange Init
20483	127.461650877	192.168.120.129	192.168.120.131	SSHv2	212	Client: Diffie-Hellman Key Exchange Init
20485	127.561080805	192.168.120.129	192.168.120.131	SSHv2	444	Server: Diffie-Hellman Key Exchange Reply, New Keys
20487	127.562207847	192.168.120.131	192.168.120.129	SSHv2	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20507	127.672380010	192.168.120.131	192.168.120.129	SSHv2	95	Client: Protocol (SSH-2.0-Nnap-SSH2-Hostkey)
20513	127.672951060	192.168.120.129	192.168.120.131	SSHv2	596	Client: Key Exchange Init
20518	127.673634799	192.168.120.129	192.168.120.131	SSHv2	1348	Server: Key Exchange Init
20526	127.676703556	192.168.120.131	192.168.120.129	SSHv2	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20532	127.685422875	192.168.120.131	192.168.120.129	SSHv2	95	Client: Protocol (SSH-2.0-Nnap-SSH2-Hostkey)
20534	127.685624117	192.168.120.129	192.168.120.131	SSHv2	596	Client: Key Exchange Init
20536	127.686070381	192.168.120.129	192.168.120.131	SSHv2	1348	Server: Key Exchange Init
20538	127.688106453	192.168.120.131	192.168.120.129	SSHv2	89	Server: Protocol (SSH-2.0-OpenSSH_7.4)
20544	127.697522980	192.168.120.131	192.168.120.129	SSHv2	95	Client: Protocol (SSH-2.0-Nnap-SSH2-Hostkey)
20546	127.697772947	192.168.120.129	192.168.120.131	SSHv2	580	Client: Key Exchange Init
20548	127.698219346	192.168.120.129	192.168.120.131	SSHv2	1348	Server: Key Exchange Init
20550	127.700715706	192.168.120.131	192.168.120.129	SSHv2	212	Client: Diffie-Hellman Key Exchange Init
20552	127.701361696	192.168.120.129	192.168.120.131	SSHv2	788	Server: Diffie-Hellman Key Exchange Reply, New Keys
20553	127.703156914	192.168.120.131	192.168.120.129	SSHv2	372	Server: Diffie-Hellman Key Exchange Reply, New Keys

Well it seems that SSH was obviously so present and well used !! if we examin the info of the packets we can see the version used which is SSH-2.0-OpenSSH-7.4.  
Let's follow the streams and see what's happening .



It seems that someone is scanning the network using Nmap. Let's verify our theory by analysing the ftp service.

Threat pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

FTP

No.	Time	Source	Destination	Protocol	Length	Info
20396	126.579610110	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20396	126.613498526	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20398	126.614477783	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20400	126.615365756	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20402	126.616188533	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20405	126.616859558	192.168.120.129	192.168.120.131	FTP	84	Request: USER anonymous
20406	126.616899286	192.168.120.129	192.168.120.131	FTP	84	Request: USER anonymous
20408	126.616982424	192.168.120.129	192.168.120.131	FTP	74	Request: SYST
20409	126.617000595	192.168.120.129	192.168.120.131	FTP	74	Request: AUTH TLS
20422	126.620638844	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20424	126.620638932	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20430	126.667562184	192.168.120.129	192.168.120.131	FTP	84	Request: USER anonymous
20431	126.667661886	192.168.120.129	192.168.120.131	FTP	74	Request: QUIT
20432	126.668303275	192.168.120.131	192.168.120.129	FTP	82	Response: 221 Goodbye.
20489	127.618577625	192.168.120.131	192.168.120.129	FTP	92	Response: 530 Permission denied.
20491	127.621180794	192.168.120.131	192.168.120.129	FTP	92	Response: 530 Permission denied.
20496	127.661340633	192.168.120.129	192.168.120.131	FTP	74	Request: QUIT
20502	127.662055160	192.168.120.131	192.168.120.129	FTP	82	Response: 221 Goodbye.
20509	127.672380072	192.168.120.131	192.168.120.129	FTP	92	Response: 530 Permission denied.
20512	127.672810796	192.168.120.129	192.168.120.131	FTP	74	Request: STAT
20515	127.673380920	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20519	127.673757284	192.168.120.129	192.168.120.131	FTP	74	Request: QUIT
20522	127.674519999	192.168.120.131	192.168.120.129	FTP	82	Response: 221 Goodbye.
20567	127.708789677	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20569	127.708907708	192.168.120.129	192.168.120.131	FTP	78	Request: AUTH TLS
20571	127.709561811	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20573	127.709714026	192.168.120.129	192.168.120.131	FTP	74	Request: QUIT
20576	127.710952898	192.168.120.131	192.168.120.129	FTP	82	Response: 221 Goodbye.
20580	127.712261879	192.168.120.129	192.168.120.131	FTP	585	Request: /026/003/001/002/000/001/000/0010/003/003000/03300/03600/0260/0010200/00005;0005 f\03000\Z10\001\b\k000\02400=
20582	127.712324837	192.168.120.129	192.168.120.131	FTP	88	Response: 220 (vsFTPd 3.0.2)
20584	127.713224886	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20586	127.713336910	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20591	127.713805558	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20593	127.713937872	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.
20595	127.714205784	192.168.120.129	192.168.120.131	FTP	160	Request: /026/003/001/000/001/000/0005/003/003000/03600/0260/00210/00005/000/000
20597	127.715951687	192.168.120.131	192.168.120.129	FTP	88	Response: 220 (vsFTPd 3.0.2)
20599	127.715951744	192.168.120.131	192.168.120.129	FTP	106	Response: 530 Please login with USER and PASS.

this one looks like a brute force attack where someone is trying to guess the password and the username of the FTP service running on the 192.168.120.131 host.  
Let's follow the FTP and try to verify the tool used :



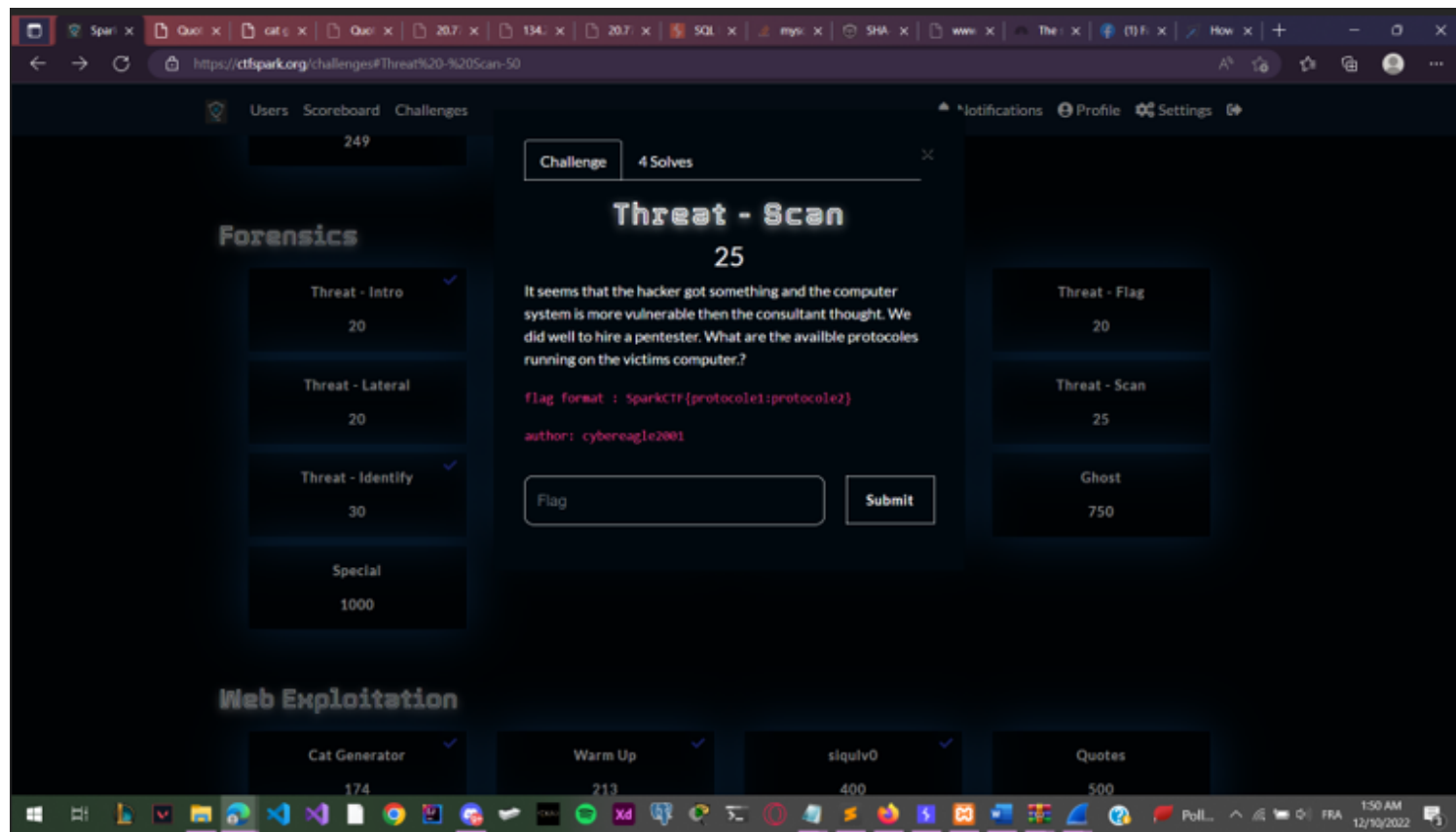
```
Wireshark · Follow TCP Stream (tcp.stream eq 1982) · Threat.pcapng  
220 (vsFTPD 3.0.2)  
SYST  
530 Please login with USER and PASS.  
USER anonymous  
530 Permission denied.  
STAT  
530 Please login with USER and PASS.  
QUIT  
221 Goodbye.
```

The vsFTPD 3.0.2 version of FTP is quite known to be vulnerable to a well known CVE.  
CVE-2015-1419: Unspecified vulnerability in vsftpd 3.0.2 and earlier allows remote attackers to bypass access restrictions via unknown vectors, related to deny\_file parsing.  
This CVE is used by the NSE script engine of Nmap. We can obviously relate because we have already found some Nmaps fingerprints in SSH.

```
Wireshark · Follow TCP Stream (tcp.stream eq 1981) · Threat.pcapng  
SSH-2.0-OpenSSH_7.4  
SSH-1.5-NmapNSE_1.0  
Protocol major versions differ.
```

our flag is : SparkCTF{nmap}

## ***Threat-scan results***



This is A gift challenge if you have solved the Utility task.

flag: SparkCTF{ftp:ssh}

**author:** cybereagle2001 (Oussama Ben Hadj Dahman)