WRITE - UP

IT NUMBER: IT22345332

NAME: G.P DINUJAYA THAMARA

WEEKEND BATCH

MALABE CAMPUS

1. where are the robots

where are the robots | 100 points

Tags: picoCTF 2019 Web Exploitation

AUTHOR: ZARATEC/DANNY

Description

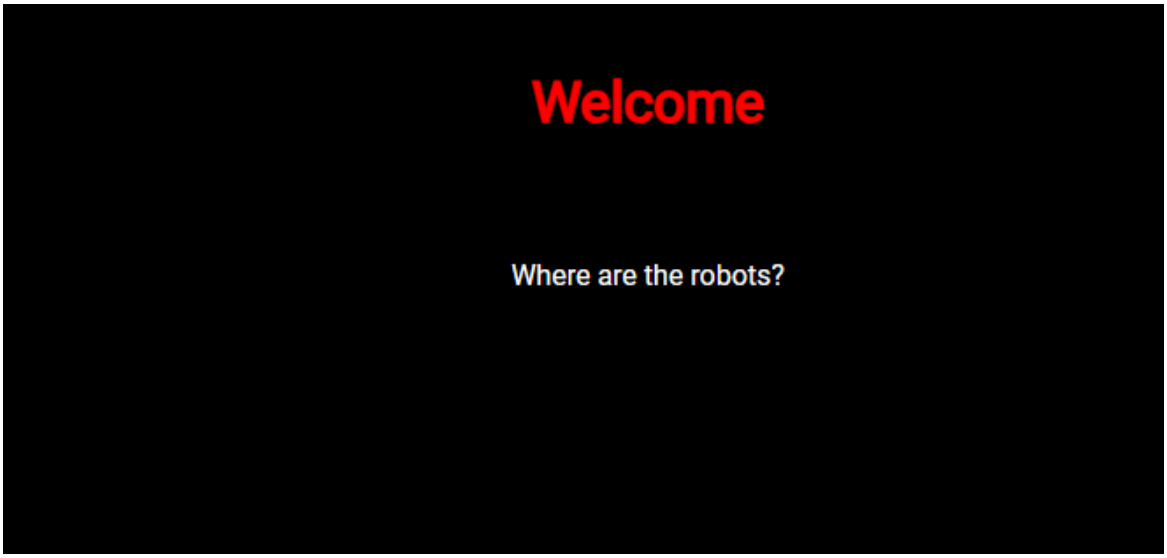
Can you find the robots? https://jupiter.challenges.picoctf.org/problem/60915/ (link) or http://jupiter.challenges.picoctf.org:60915

Hints 1

55,779 solves / 59,505 users attempted (94%) 84% Liked

picoCTF{FLAG} Submit Flag

this is where link directs to

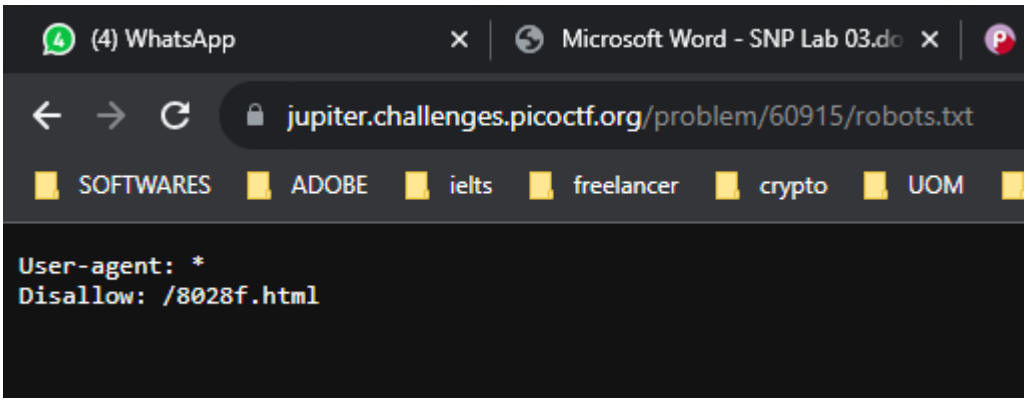


What part of the website could tell you where the creator doesn't want you to look? Is the hint given by them which makes sense to move to **robots.txt** file?

/robots.txt file is a standard used to provide instructions to web crawlers or "robots" about which parts of a website should or should not be crawled, indexed, or accessed. It is commonly used to control how search engines and other automated agents interact with a website

https://jupiter.challenges.picoctf.org/problem/60915/robots.txt

After I moved to robots.txt file it gives me the hint where to go next



could be a URL pointing to a specific web page that contains information relevant to the challenge. So visit that page by changing the last part of the URL as follows.

https://jupiter.challenges.picoctf.org/problem/60915/8028f.html after I change the URL I got the flag.

2. Scavenger Hunt

Scavenger Hunt | 50 points

Tags: picoCTF 2021 Web Exploitation

AUTHOR: MADSTACKS

Description

There is some interesting information hidden around this site http://mercury.picoctf.net:39698/. Can you find it?

Hints

1

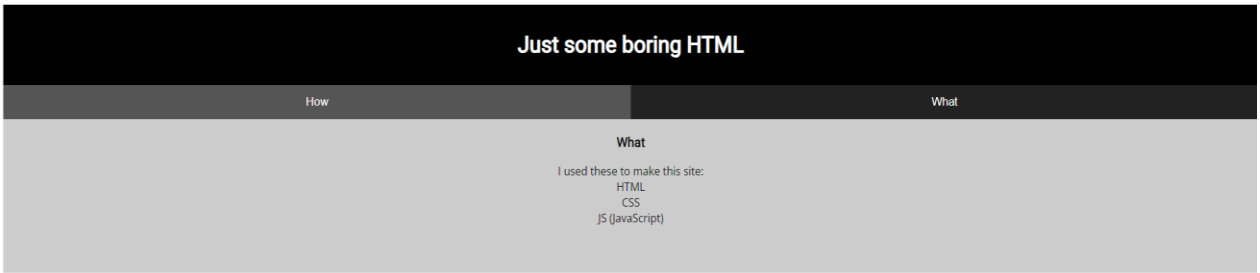
36,471 solves / 42,556 users attempted (86%)

63% Liked

picoCTF{FLAG}

Submit Flag

When the following link is clicked it directs to a below webpage



So, when I visited the website as for the first step, I checked with the inspect element.

There were no clues to be found in that, so after that I moved to page source which gives the first part of the flag.

```html
<!doctype html>
<html>
  <head>
    <title>Scavenger Hunt</title>
    <link href="https://fonts.googleapis.com/css?family=Open+Sans|Roboto" rel="stylesheet">
    <link rel="stylesheet" type="text/css" href="mycss.css">
    <script type="application/javascript" src="myjs.js"></script>
  </head>

  <body>
    <div class="container">
      <header>
        <h1>Just some boring HTML</h1>
      </header>

      <button class="tablink" onclick="openTab('tabintro', this, '#222')" id="defaultOpen">How</button>
      <button class="tablink" onclick="openTab('tababout', this, '#222')">What</button>

      <div id="tabintro" class="tabcontent">
        <h3>How</h3>
        <p>How do you like my website?</p>
      </div>

      <div id="tababout" class="tabcontent">
        <h3>What</h3>
        <p>I used these to make this site: <br/>
          HTML <br/>
          CSS <br/>
          JS (JavaScript)
        </p>
    <!-- Here's the first part of the flag: picoCTF{t -->
      </div>

    </div>

  </body>
</html>
```

After I click mycss.css file I got the second part of the flag

```
}

h1 {
    color: white;
}

p {
    font-family: "Open Sans";
}

.tablink {
    background-color: #555;
    color: white;
    float: left;
    border: none;
    outline: none;
    cursor: pointer;
    padding: 14px 16px;
    font-size: 17px;
    width: 50%;
}

.tablink:hover {
    background-color: #777;
}

.tabcontent {
    color: #111;
    display: none;
    padding: 50px;
    text-align: center;
}

#tabintro { background-color: #ccc; }
#tababout { background-color: #ccc; }

/* CSS makes the page look nice, and yes, it also has part of the flag. Here's part 2: h4ts_4_l0 */
```

After that I clicked the myjs.js file to get some more clues in that there was a comment which gives a certain hint to move to robots.txt file

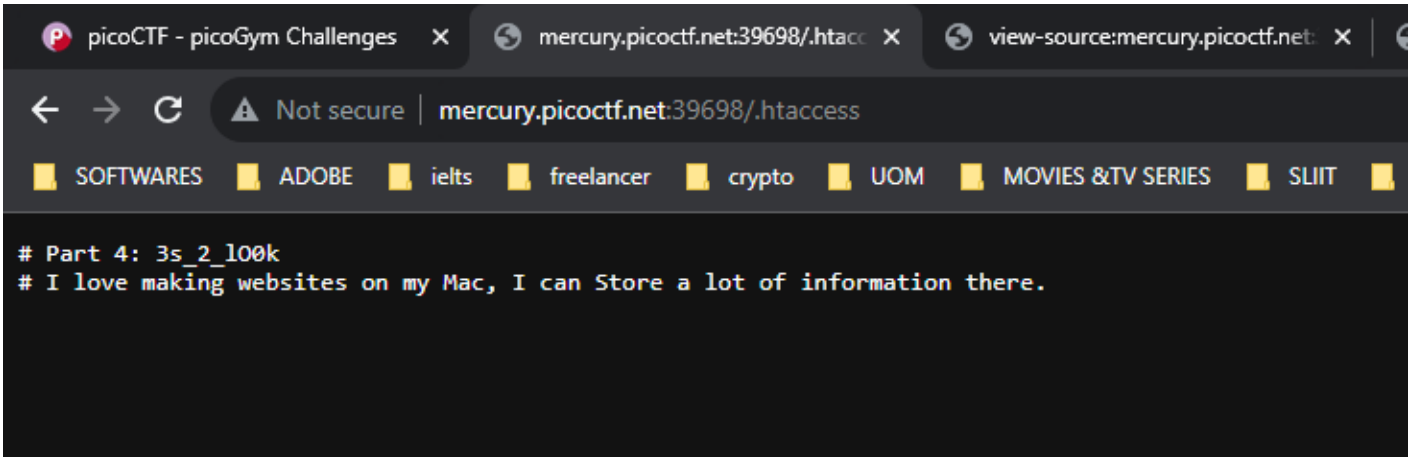
```
function openTab(tabName,elmnt,color) {
    var i, tabcontent, tablinks;
    tabcontent = document.getElementsByClassName("tabcontent");
    for (i = 0; i < tabcontent.length; i++) {
        tabcontent[i].style.display = "none";
    }
    tablinks = document.getElementsByClassName("tablink");
    for (i = 0; i < tablinks.length; i++) {
        tablinks[i].style.backgroundColor = "";
    }
    document.getElementById(tabName).style.display = "block";
    if(elmnt.style != null) {
        elmnt.style.backgroundColor = color;
    }
}

window.onload = function() {
    openTab('tabintro', this, '#222');
}

/* How can I keep Google from indexing my website? */
```

http://mercury.picoctf.net:39698/robots.txt so when I moved to the robots file it gives me third part of the flag with some additional hints.

```
User-agent: *
Disallow: /index.html
# Part 3: t_0f_pl4c
# I think this is an apache server... can you Access the next flag?
```

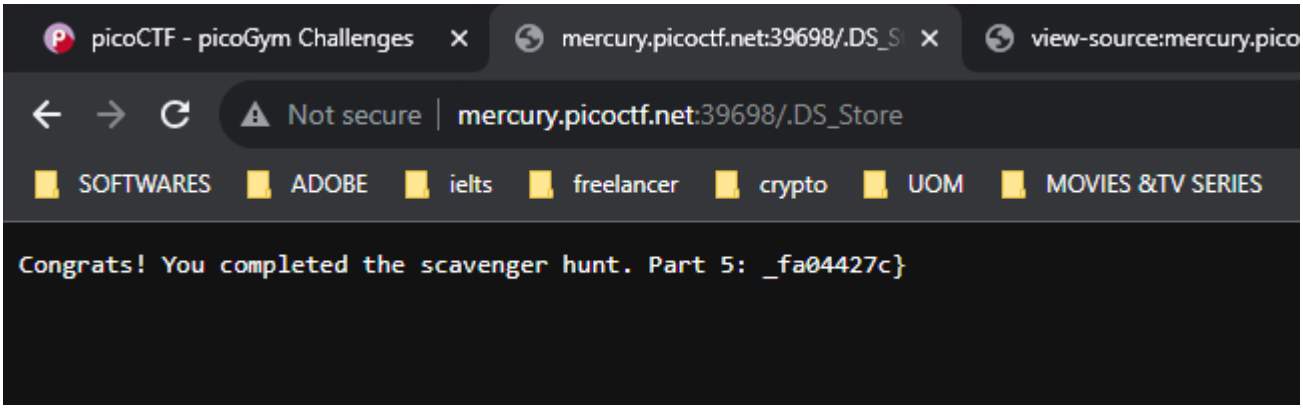
Since it has given a hint, **you have to access the Apache server.**

Through this file configuration **(/.htaccess)** we can access Apache web servers, The **.htaccess** (hypertext access) file is a configuration file used by Apache web servers to control various aspects of how a directory and its subdirectories behave. It can be used to set up authentication, URL rewriting, access control, and more.

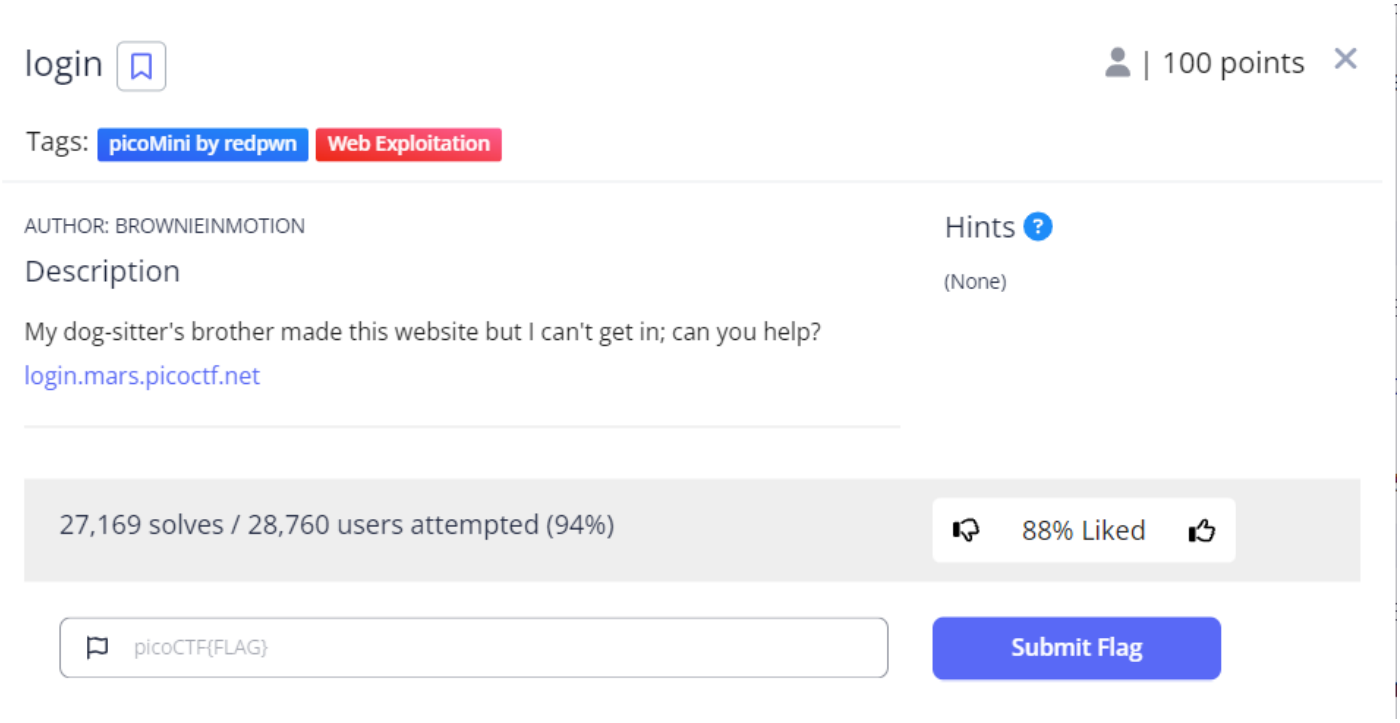


In that I found the fourth part of the flag and another hint which says that he likes to store information on a mac. Which is probably a reference for the **.DS_Store**

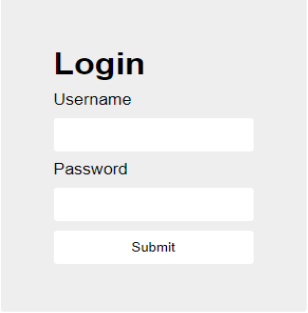
.DS_Store is a file created by macOS operating systems to store custom attributes of a folder, such as the position of icons or background images. **These files are usually hidden and contain metadata about the folder's appearance and arrangement**.



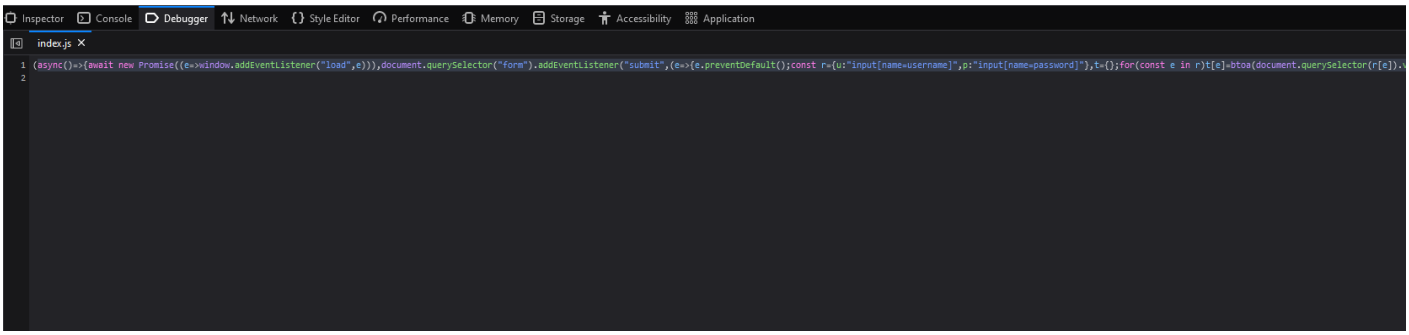
3. login



When the link is clicked, it directs to the following page



As the first step check the inspect element to find clues in there were no clues in the html and css page and then lastly I check the js file



Since all the codes are in one line, its bit hard to identify so I copied in the vs and click format document which looks more clearer which is like this

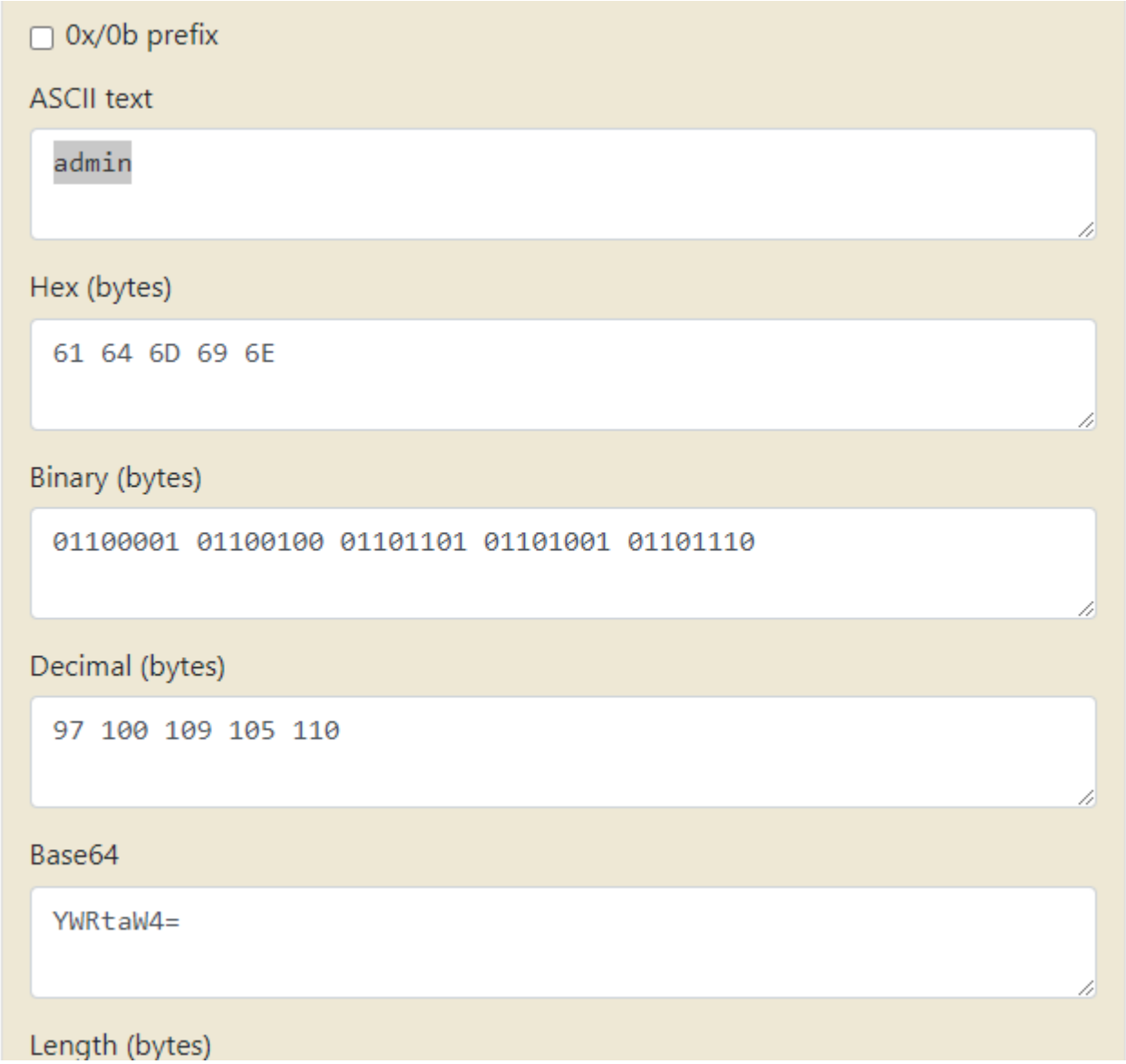
```
(async () => {
  await new Promise((e) => window.addEventListener("load", e)),
    document.querySelector("form").addEventListener("submit", (e) => {
      e.preventDefault();
      const r = { u: "input[name=username]", p: "input[name=password]" },
        t = {};
      for (const e in r)
        t[e] = btoa(document.querySelector(r[e]).value).replace(/=/g, "");
      return "YWRtaW4" !== t.u
        ? alert("Incorrect Username")
        : "cGljb0NURns1M3J2M3JfNTNydjNyXzUzcnYzcl81M3J2M3JfNTNydjNyfQ" !== t.p
        ? alert("Incorrect Password")
        : void alert(`Correct Password! Your flag is ${atob(t.p)}.`);
    });
})();
```

In here **btoa means** encoding a string in base-64

for example btoa("abc") gives the output as "YWJj" **where "abc" in encoded in base-64**

t[e] = btoa(document.querySelector(r[e]).value).replace(/=/g, ""); This code seems to be designed to encode values from HTML input elements and remove any equal signs (=) from the encoded output.

YWRtaW4 this in **base 64** when it is converted to **ASCII text** it gives **admin**



`Correct Password! Your flag is ${atob(t.p)}.` from this string we can identify that we should convert the base 64 password to ASCII text to get the flag

Number delimiter

Space

0x/0b prefix

ASCII text

```
picoCTF{53rv3r_53rv3r_53rv3r_53rv3r_53rv3r}
```

Hex (bytes)

```
70 69 63 6F 43 54 46 7B 35 33 72 76 33 72 5F 35 33 72 76 33
72 5F 35 33 72 76 33 72 5F 35 33 72 76 33 72 5F 35 33 72 76
```

Binary (bytes)

```
01110000 01101001 01100011 01101111 01000011 01010100
01000110 01111011 00110101 00110011 01110010 01110110
```

Decimal (bytes)

```
112 105 99 111 67 84 70 123 53 51 114 118 51 114 95 53 51 114
118 51 114 95 53 51 114 118 51 114 95 53 51 114 118 51 114 95
```

Base64

cGljb0NURns1M3J2M3JfNTNydjNyXzUzcnYzcl81M3J2M3JfNTNydjNyfQ==

Length (bytes)

43

Checksum

8-bit | Sum | C5

4. GET aHEAD

GET aHEAD | 20 points

Tags: picoCTF 2021 Web Exploitation

AUTHOR: MADSTACKS

Description

Find the flag being held on this server to get ahead of the competition

http://mercury.picoctf.net:34561/

Hints

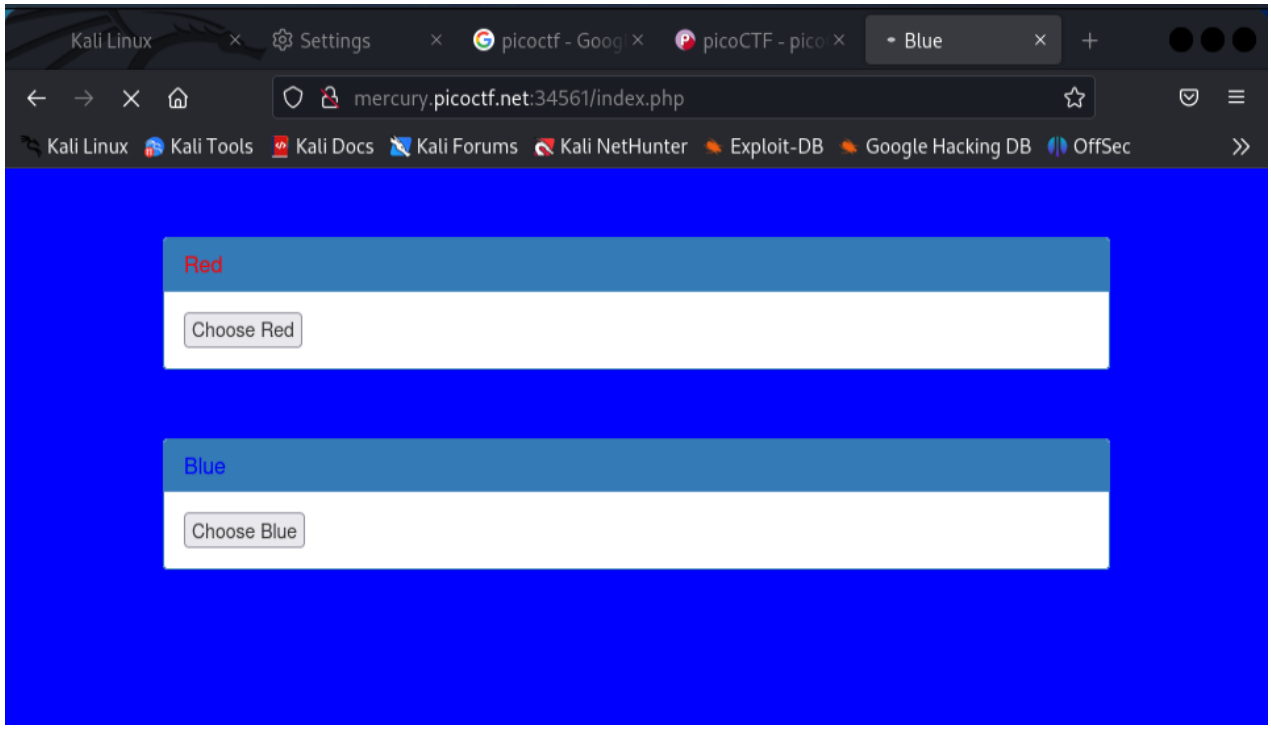
1 2

Check out tools like Burpsuite to modify your requests and look at the responses

59,445 solves / 64,327 users attempted (92%) 82% Liked

picoCTF{FLAG} Submit Flag

Once I clicked the above link it redirected me to the following page



As usual, I moved to the inspect element and page sources there was a clue, these two colors use two different request methods.

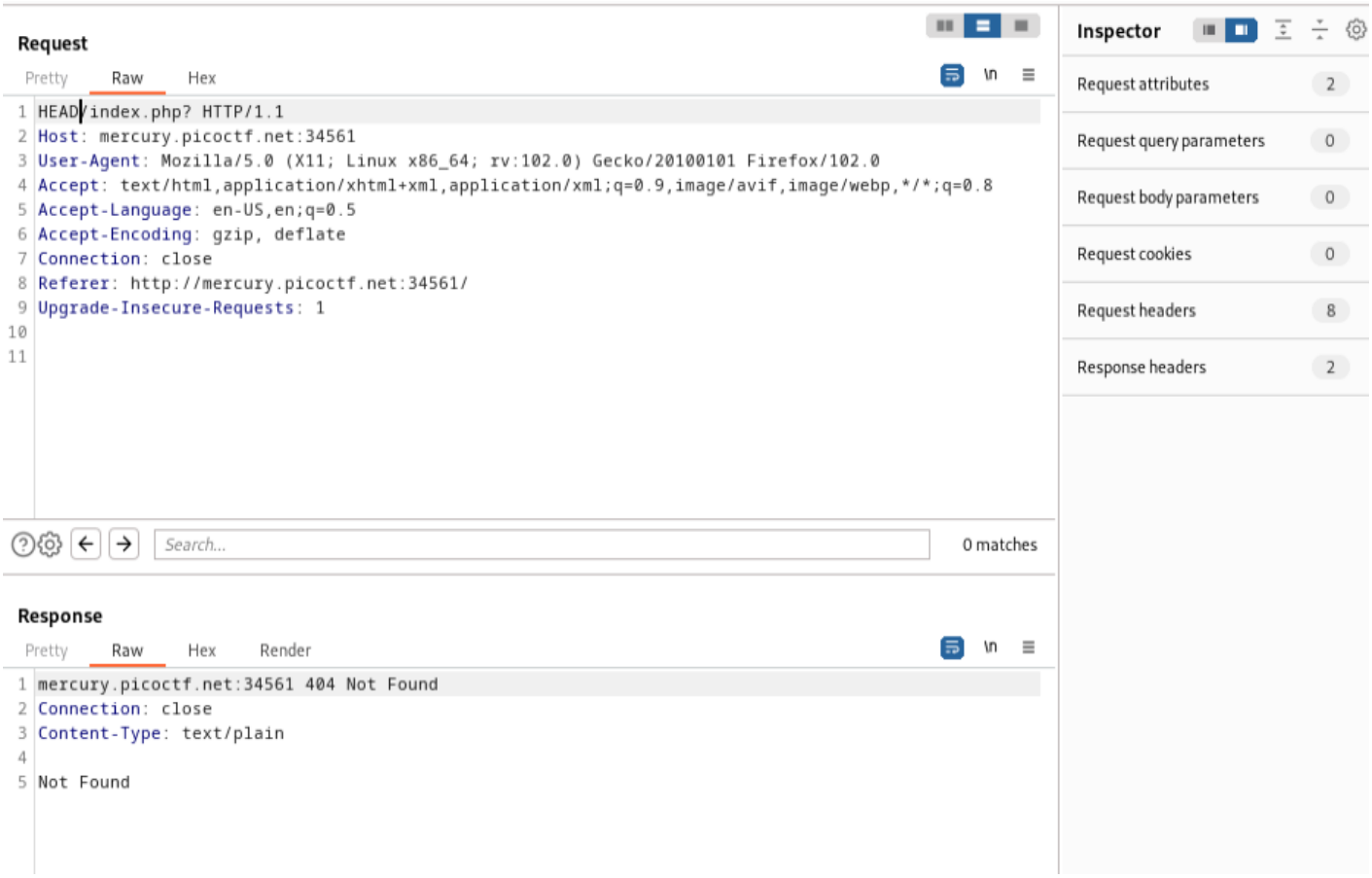
Red uses GET method.

Blue use POST method.

When we look at Request and respond headers inspectors they look perfectly normally

Since the challenge name also GET aHEAD I thought of changing the request method from GET to HEAD for that **I click send to repeater** then we can change any request. And click Send button

It gives me nothing.



After that I change the request blue too from POST to HEAD, it gives me the flag.

1 × 2 × 3 × 4 × +

Send Cancel < > Target: http://

Request

Pretty Raw Hex

```
HEAD /index.php HTTP/1.1
Host: mercury.picoctf.net:34561
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
Origin: http://mercury.picoctf.net:34561
Connection: close
Referer: http://mercury.picoctf.net:34561/
Upgrade-Insecure-Requests: 1
```

Search... 0 matches

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
flag: picoCTF{r3j3ct_th3_du411ty_8f878508}
Content-type: text/html; charset=UTF-8
```

5. Mind your Ps and Qs

Mind your Ps and Qs | 20 points

Tags: picoCTF 2021 Cryptography

AUTHOR: SARA

Description

In RSA, a small e value can be problematic, but what about N? Can you decrypt this? values

Hints

1

Bits are expensive, I used only a little bit over 100 to save money

27,729 solves / 32,396 users attempted (86%)

73% Liked

picoCTF{FLAG}

Submit Flag

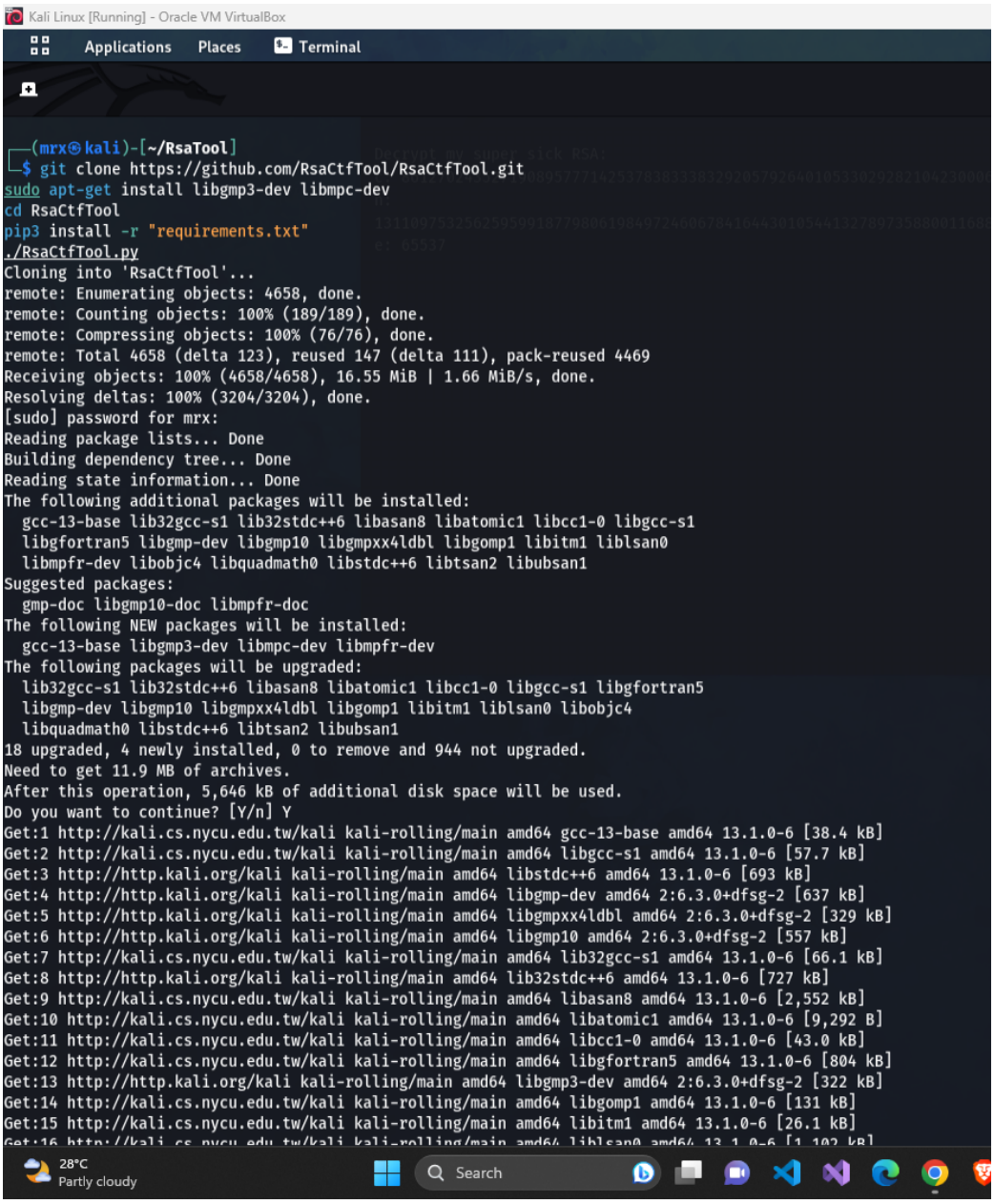
When I click the link values it downloaded this file

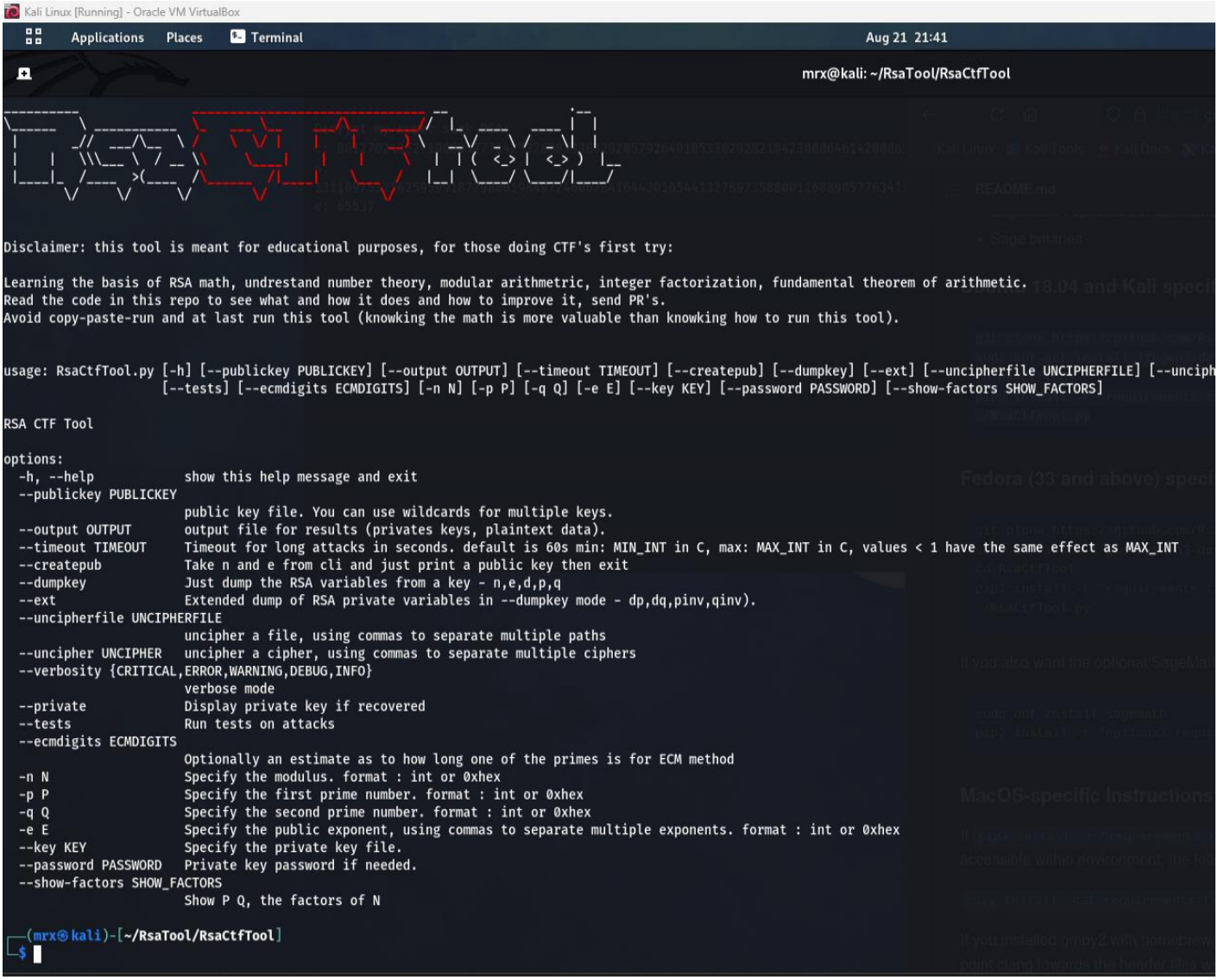
Open values ~/Downloads

```
Decrypt my super sick RSA:
c: 861270243527190895777142537838333832920579264010533029282104230006461420086153423
n:
1311097532562595991877980619849724606784164430105441327897358800116889057763413423
e: 65537
```

In here they have used RSA Algorithm where **RSA algorithm** is an asymmetric cryptography algorithm. Asymmetric means that it works on two different keys i.e., **Public Key** and **Private Key.** As the name describes, the Public Key is given to everyone, and the Private key is kept private.The RSA algorithm involves four steps: key generation, key distribution, encryption, and decryption

To solve this challenge, I found and online tool. I installed it in my kali machine.





After that using it requirement.txt file I found the different options enter the required values and got the flag

```
options:
  -h, --help            show this help message and exit
  --publickey PUBLICKEY
                        public key file. You can use wildcards for multiple
                        keys.
  --output OUTPUT       output file for results (privates keys, plaintext
                        data).
  --timeout TIMEOUT     Timeout for long attacks in seconds. default is 60s
                        min: MIN_INT in C, max: MAX_INT in C, values < 1 have
                        the same effect as MAX_INT
  --createpub           Take n and e from cli and just print a public key then
                        exit
  --dumpkey             Just dump the RSA variables from a key - n,e,d,p,q
  --ext                 Extended dump of RSA private variables in --dumpkey
                        mode - dp,dq,pinv,qinv).
  --uncipherfile UNCIPHERFILE
                        uncipher a file, using commas to separate multiple
                        paths
  --uncipher UNCIPHER   uncipher a cipher, using commas to separate multiple
                        ciphers
  --verbosity {CRITICAL,ERROR,WARNING,DEBUG,INFO}
                        verbose mode
  --private             Display private key if recovered
  --tests               Run tests on attacks
  --ecmdigits ECMDIGITS
                        Optionally an estimate as to how long one of the
                        primes is for ECM method
  -n N                  Specify the modulus. format : int or 0xhex
  -p P                  Specify the first prime number. format : int or 0xhex
  -q Q                  Specify the second prime number. format : int or 0xhex
  -e E                  Specify the public exponent, using commas to separate
                        multiple exponents. format : int or 0xhex
  --key KEY             Specify the private key file.
  --password PASSWORD   Private key password if needed.
┌──(mrx㉿kali)-[~/RsaTool/RsaCtfTool]
└─$ python3 RsaCtfTool.py -n 131109753256259599187798061984972460678416443010544132789735880011688905776341 3423 -e 65537 --uncipher 861270243527190895777142537838333832920579264010533029282104230006461420086153423
```

```
[+] loading prime list file data/ti_rsa_signing_keys.txt...
100%|
[+] Time elapsed: 0.0021 sec.
[*] Performing mersenne_primes attack on /tmp/tmpvn9jyjlb.
 24%|
[+] Time elapsed: 0.0003 sec.
[*] Performing smallq attack on /tmp/tmpvn9jyjlb.
[+] Time elapsed: 0.1545 sec.
[*] Performing lucas_gcd attack on /tmp/tmpvn9jyjlb.
100%|
[+] Time elapsed: 0.0272 sec.
[*] Performing factordb attack on /tmp/tmpvn9jyjlb.
[*] Attack success with factordb method !
[+] Total time elapsed min,max,avg: 0.0003/0.1545/0.0351 sec.

Results for /tmp/tmpvn9jyjlb:

Unciphered data :
HEX : 0x007069636f4354467b736d6131315f4e5f6e305f67306f645f31333638363637397d
INT (big endian) : 130163825294491060659272914253425354379962221353529052566395739590028494157397 73
INT (little endian) : 37119719776712686220408522365100361254955019426847706732211053811485132026256711 68
utf-8 : picoCTF{sma11_N_n0_g0od_13686679}
utf-16 : 瀰捩羆褥獻愁ㄱ也湟弭で摯ㄱ韭堊孅紹
STR : b'\x00picoCTF{sma11_N_n0_g0od_13686679}'

┌──(mrx㉿kali)-[~/RsaTool/RsaCtfTool]
└─$
```

6.Insp3ct0r

Insp3ct0r | 50 points

Tags: picoCTF 2019 Web Exploitation

AUTHOR: ZARATEC/DANNY

Description

Kishor Balan tipped us off that the following code may need inspection:

https://jupiter.challenges.picoctf.org/problem/44924/ (link) or

http://jupiter.challenges.picoctf.org:44924

Hints

1 2

83,228 solves / 87,729 users attempted (95%)

87% Liked

picoCTF{FLAG}

Submit Flag

In this challenge from the name even we can guess where to look and the hints are given as how you would inspect a web code on browser for that we must simply go to the inspect element or must view the page source, so in a webpage there are basically three parts which is html part , css part and the js part since the second given as there are three parts I thought of first looking those three files so as I thought the three parts of the flag were in these three separate files

```
  </head>
<body>
  <div class="container"> overflow
    <header>
      <h1>Inspect Me</h1>
    </header>
    <button id="defaultOpen" class="tablink" onclick="openTab('tabintro', this, '#222')">What</button> event
    <button class="tablink" onclick="openTab('tababout', this, '#222')">How</button> event
    <div id="tabintro" class="tabcontent" style="display: block;">
      <h3>What</h3>
      <p>I made a website</p>
    </div>
    <div id="tababout" class="tabcontent" style="display: none;">
      <h3>How</h3>
      <p>...</p>
      <!--Html is neat. Anyways have 1/3 of the flag: picoCTF{tru3_d3-->
    </div>
  </div>
```

```
        cursor: pointer;
        padding: 14px 16px;
        font-size: 17px;
        width: 50%;
    }

    .tablink:hover {
        background-color: #777;
    }

    .tabcontent {
        color: #111;
        display: none;
        padding: 50px;
        text-align: center;
    }

    #tabintro { background-color: #ccc; }
    #tababout { background-color: #ccc; }

    /* You need CSS to make pretty pages. Here's part 2/3 of the flag: t3ct1ve_0r_ju5t */
```

```
        document.getElementById(tabName).style.display = "block";
        if(elmnt.style != null) {
        elmnt.style.backgroundColor = color;
        }
    }

    window.onload = function() {
        openTab('tabintro', this, '#222');
    }

    /* Javascript sure is neat. Anyways part 3/3 of the flag: _lucky?f10be399} */
```

7.Cookies

Cookies | 40 points

Tags: picoCTF 2021 Web Exploitation

AUTHOR: MADSTACKS

Description

Who doesn't love cookies? Try to figure out the best one.

http://mercury.picoctf.net:29649/

Hints

(None)

44,699 solves / 48,484 users attempted (92%)

61% Liked

picoCTF{FLAG}

Submit Flag

Once I click this link it gives me this page

Cookies

Home

Welcome to my cookie search page. See how much I like different kinds of cookies!

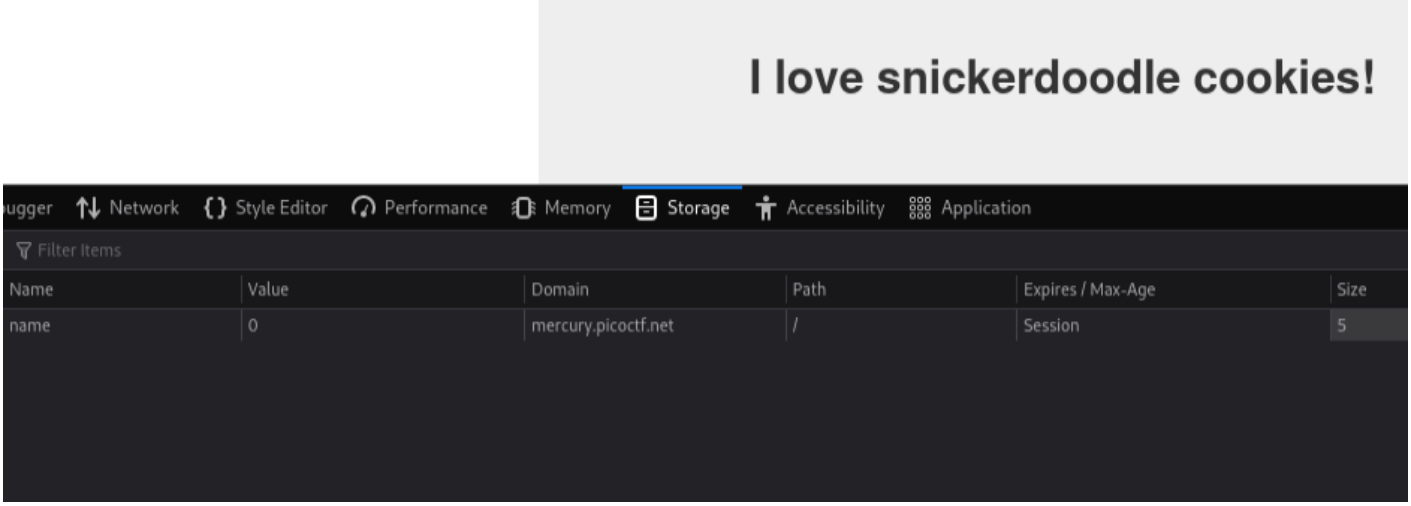
snickerdoodle

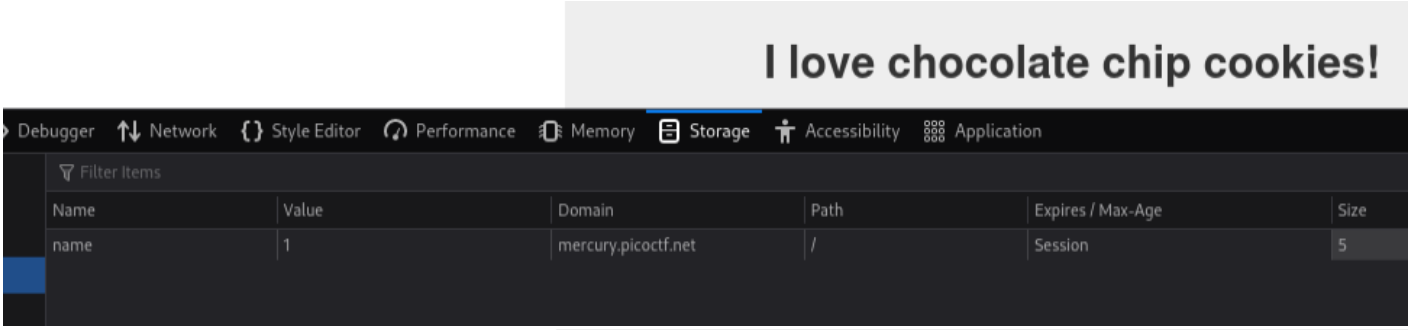
Search

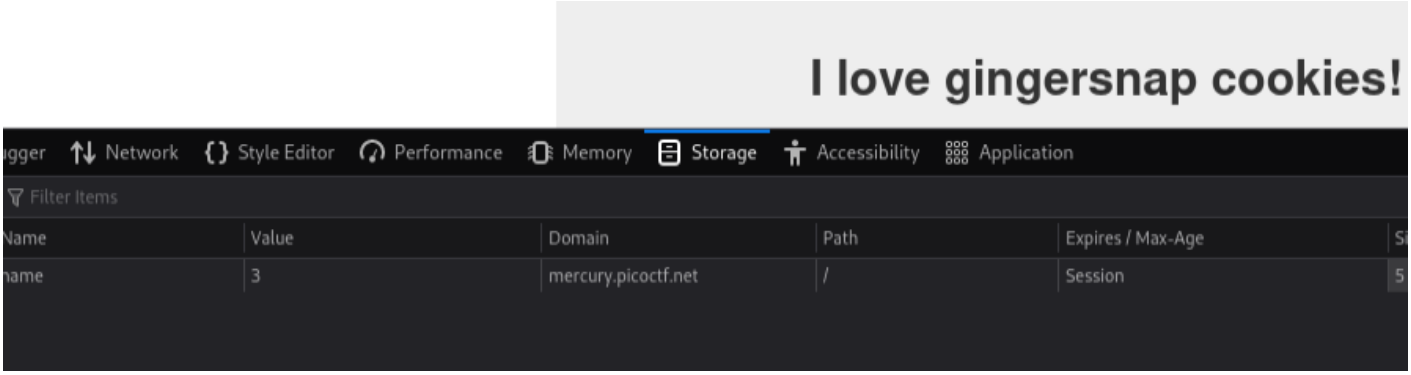
© PicoCTF

Cookie is a way to have users store information about what they are doing and present it back to you.

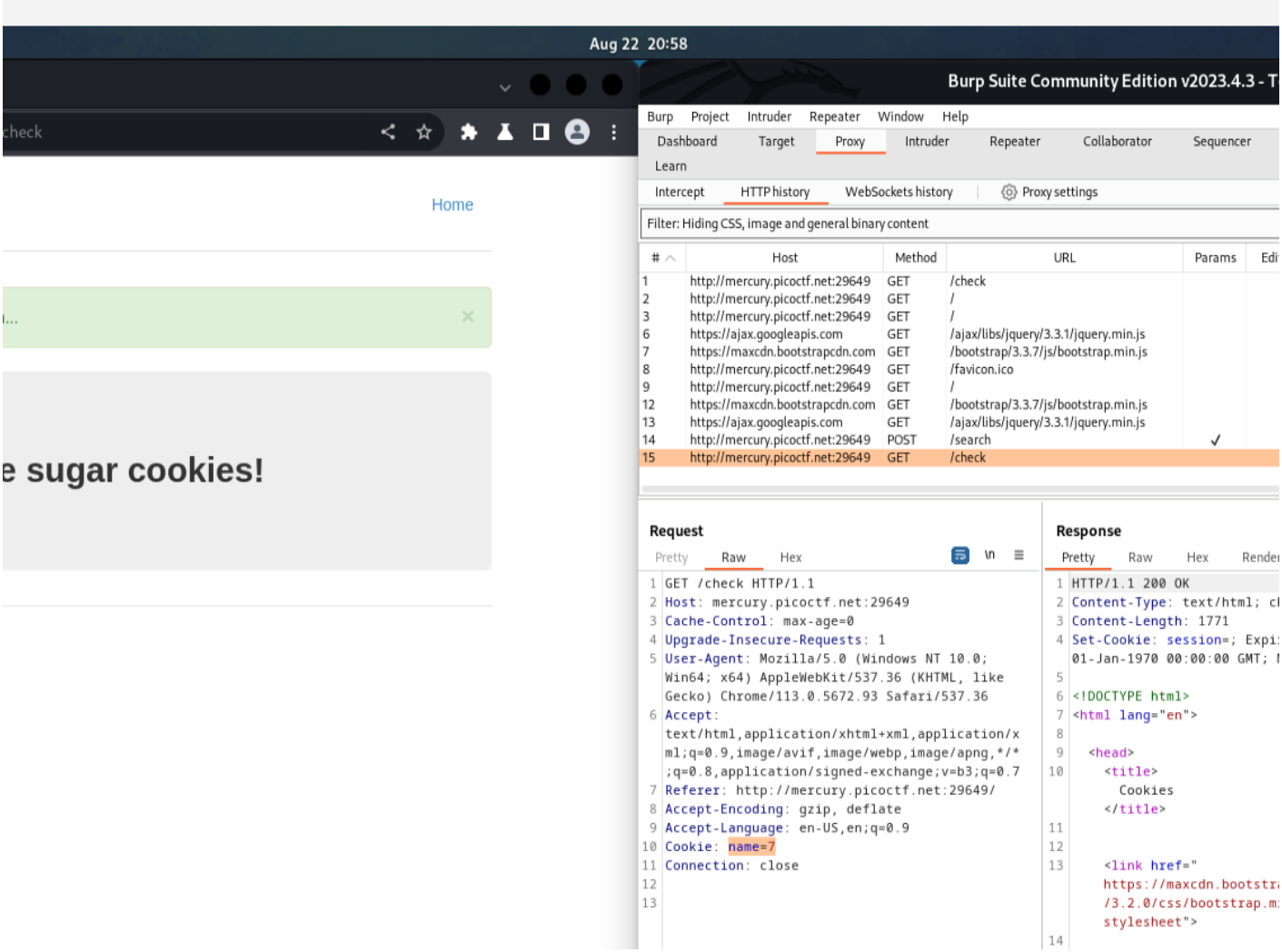
In here when you type different type of cookies the value changes that it has a array of cookies we cannot go through one by one so need to use Burpsuite to automate this function



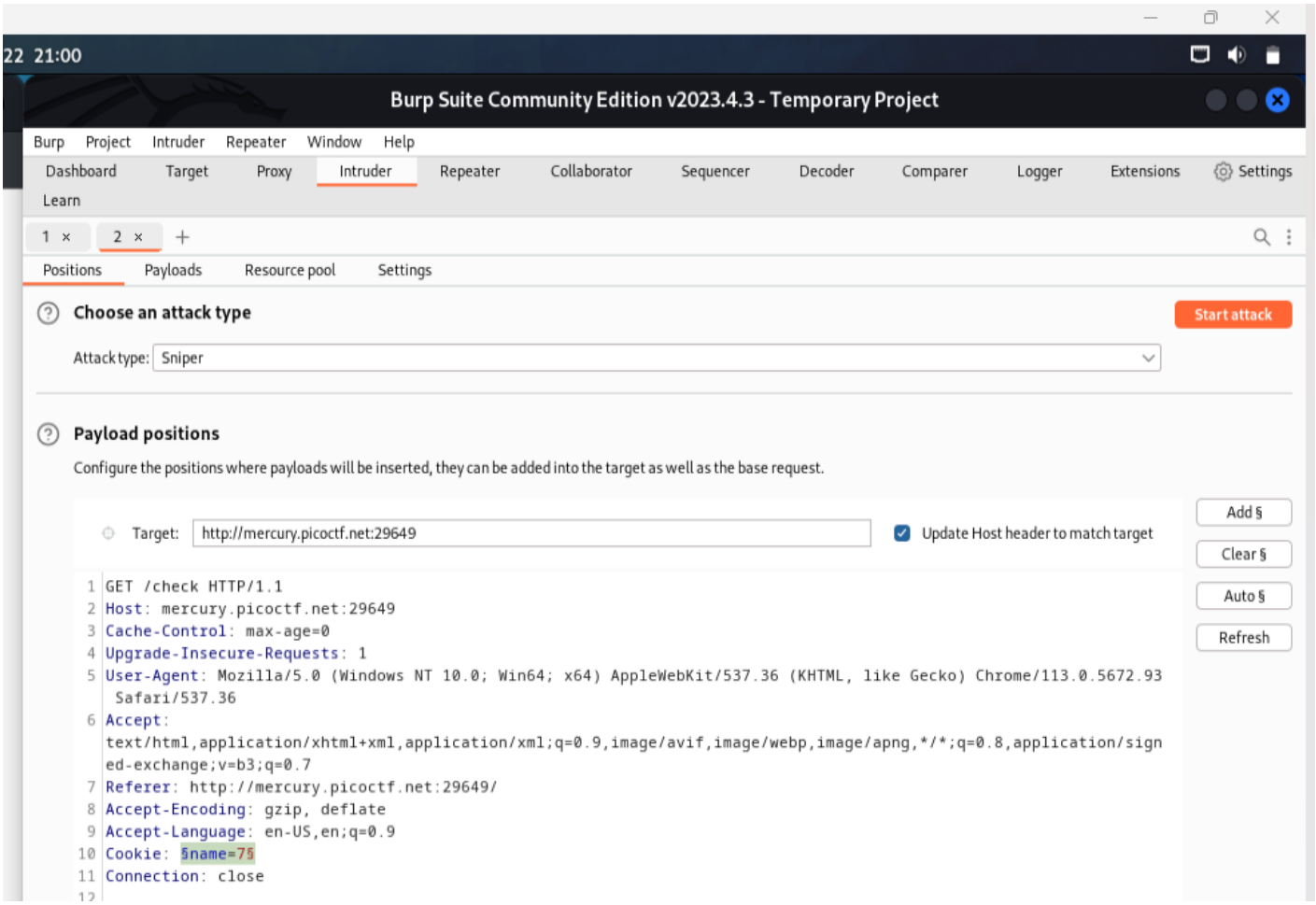




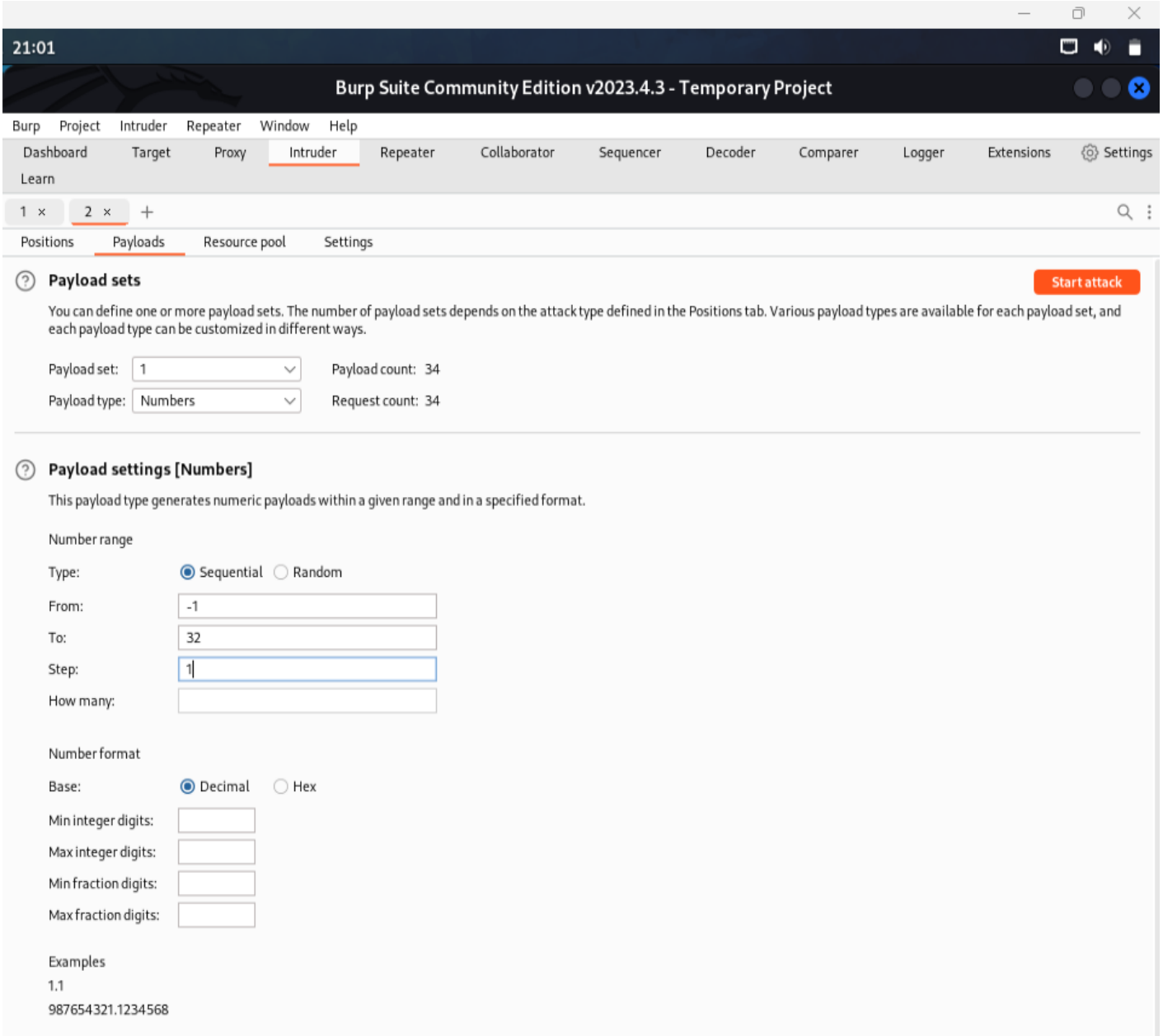
So I opened burpsuite moved proxy tab click intercept is off to on the intercept and click open browser and paste the link



So since we need to send different values we need to send it to intruder



Then move to payloads tab do the following changes to it and click start attack.


21:01
Burp Suite Community Edition v2023.4.3 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Extensions Settings
Learn
1 × 2 ×
Positions Payloads Resource pool Settings
Payload sets
Start attack
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: 34
Payload type: Numbers
Request count: 34
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:
Max integer digits:
Min fraction digits:
Max fraction digits:
Examples
1.1
987654321.1234568

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length
0		200			1932
1	-1	302			557
2	0	200			1940
3	1	200			1941
4	2	200			1941
5	3	200			1937
6	4	200			1937
7	5	200			1940
8	6	200			1938
9	7	200			1932
10	8	200			1935
11	9	200			1931
12	10	200			1935
13	11	200			1933
14	12	200			1933

From this we must select the one with unusual length then it will give the flag.

8.Power Cookie

Power Cookie | 200 points

Tags: picoCTF 2022 Web Exploitation cookie

AUTHOR: LT 'SYREAL' JONES

Description

Can you get the flag?

Go to this website and see what you can discover.

Hints

1

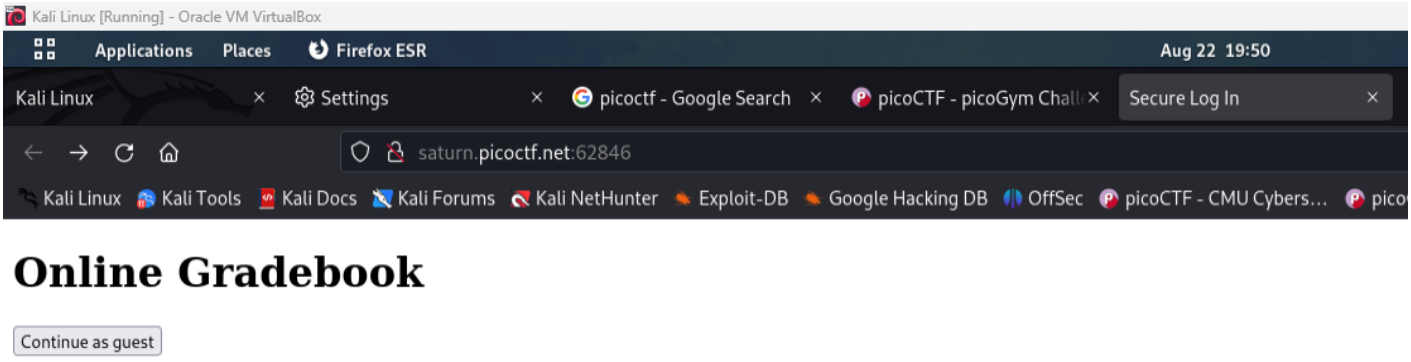
16,627 solves / 16,960 users attempted (98%)

93% Liked

picoCTF{FLAG}

Submit Flag

The above link redirects me to this page



As the page is plane and as usually for the first I went to the source page which some hint about an js file.

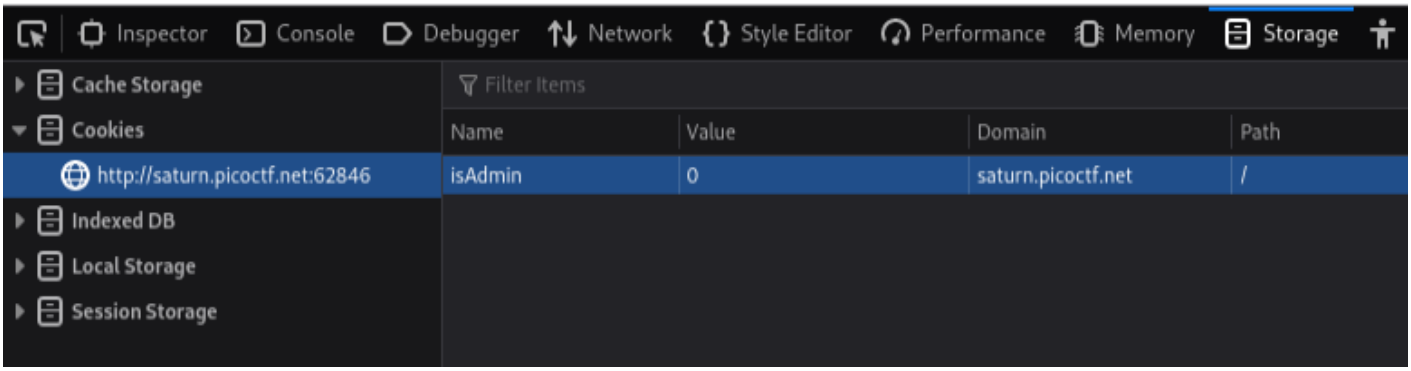
```
Kali Linux  Kali Tools  Kali Docs  Kali Forums  Kali NetHunter  Exploit-DB  Google Hac

1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>Secure Log In</title>
8    </head>
9    <body>
10     <script src="guest.js"></script>
11
12     <h1>Online Gradebook</h1>
13     <button type="button" onclick="continueAsGuest();">Continue as guest</button>
14   </body>
15 </html>
16
```

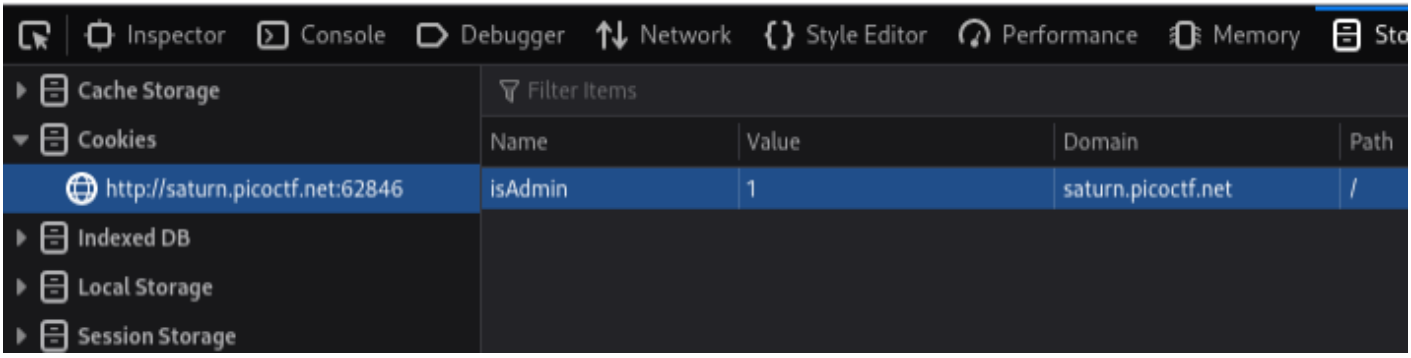
In the guest.js file it gives me a hint about cookies

```
Applications  Places  Firefox ESR

Kali Linux ×  Settings ×  picoctf - Google Sear ×

view-source:http://saturn.picoctf.net:62846/g

Kali Linux  Kali Tools  Kali Docs  Kali Forums  Kali NetHunter  Exploit-D

function continueAsGuest()
{
  window.location.href = '/check.php';
  document.cookie = "isAdmin=0";
}
```

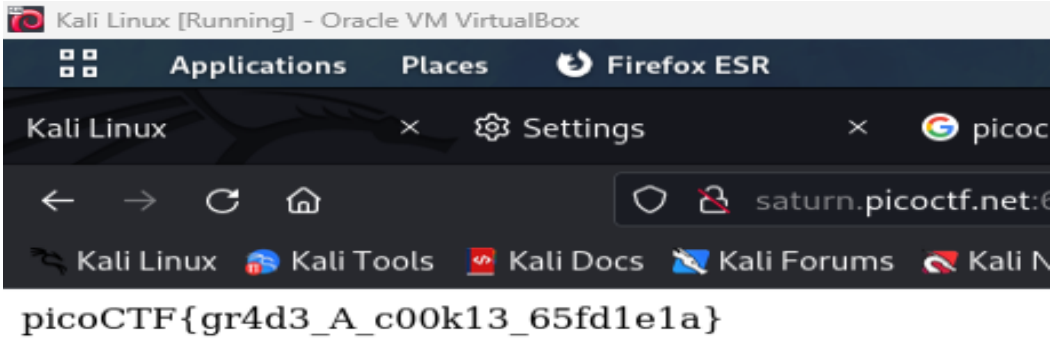
In document.cookie = “isAdmin = 0” it’s a Boolean value where it is whether it is an Admin or not so if we change this value to 1 (modified this cookie)in developer option. we can get closer to the flag. And also, from the code segment ‘./check.php’ we can clearly identify that it is running php. It is server-side language it looking for the cookie that user carries

So I opened the inspector element and moved to the storage tab there in Cookies sections I change the value from zero to one which like making “isAdmin” true by making “isAdmin = 1”

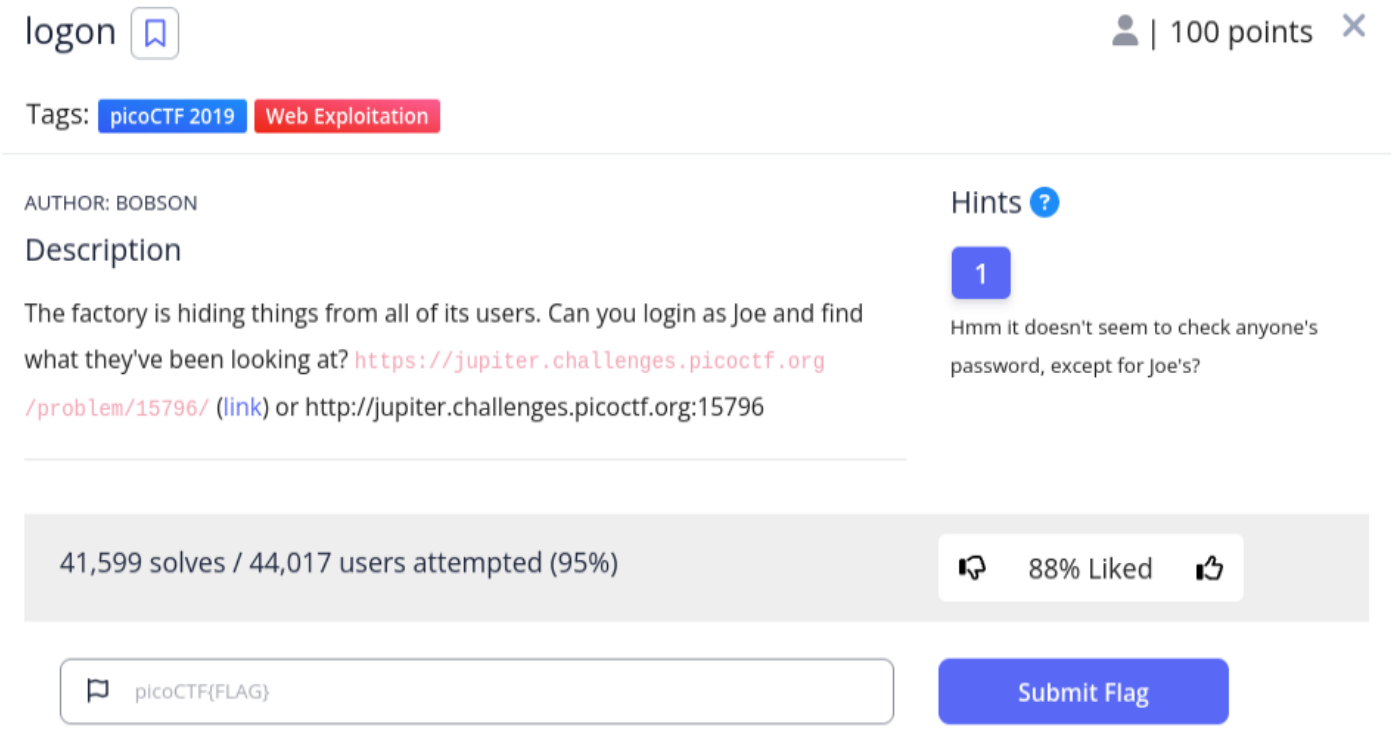




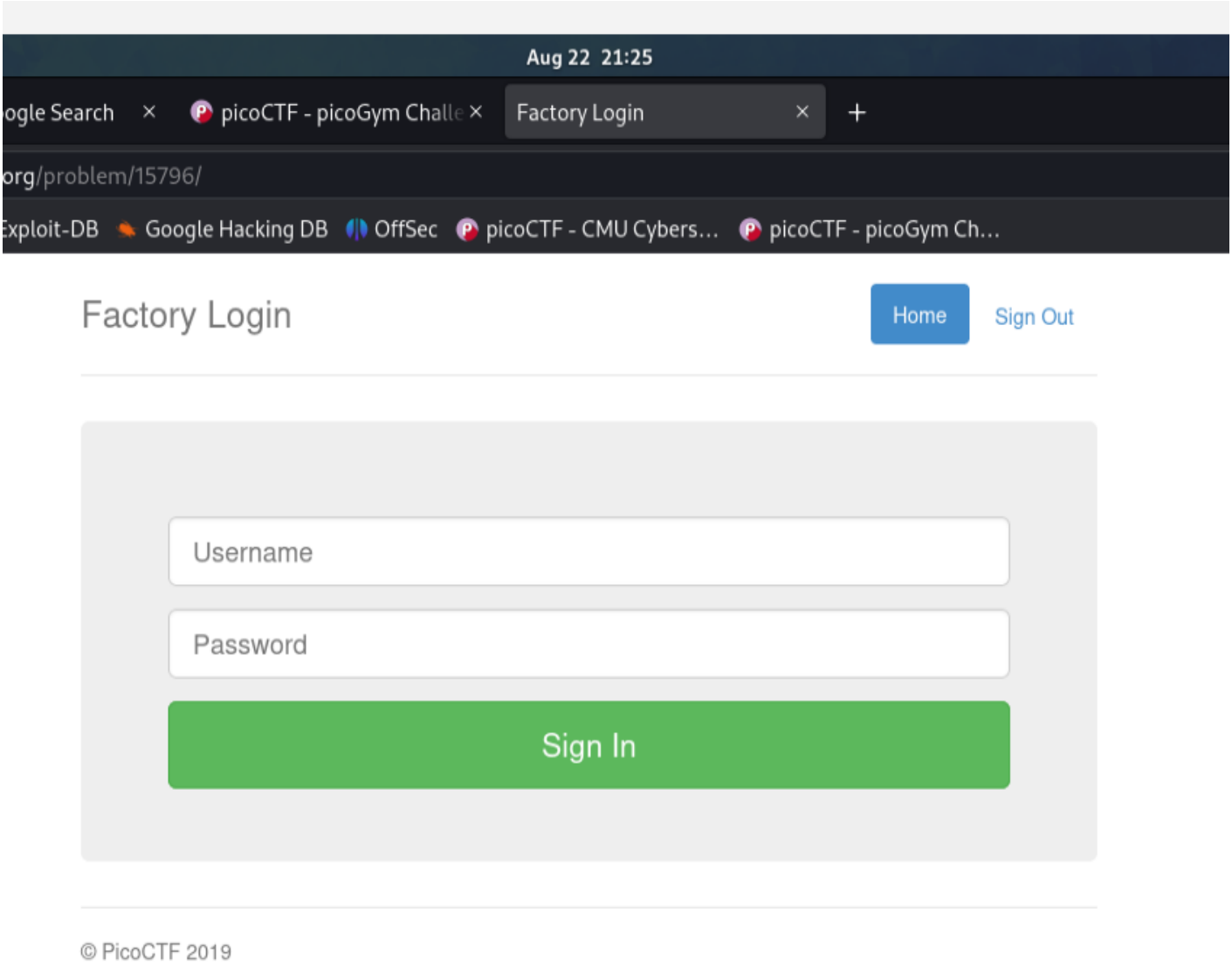
After this reloaded the page using CTRL+SHIFT+R which is hard refresh reasking the webserver to give the required page



9. logon

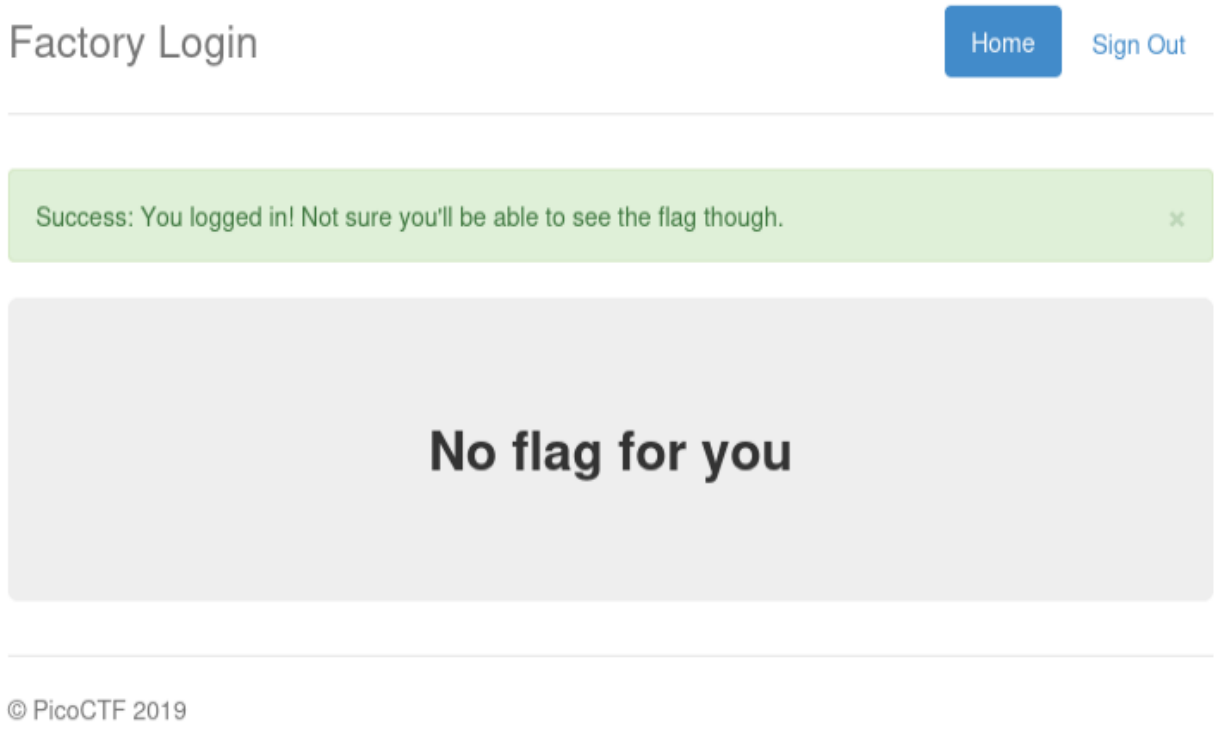


Once I click the above, I was redirected to the below page

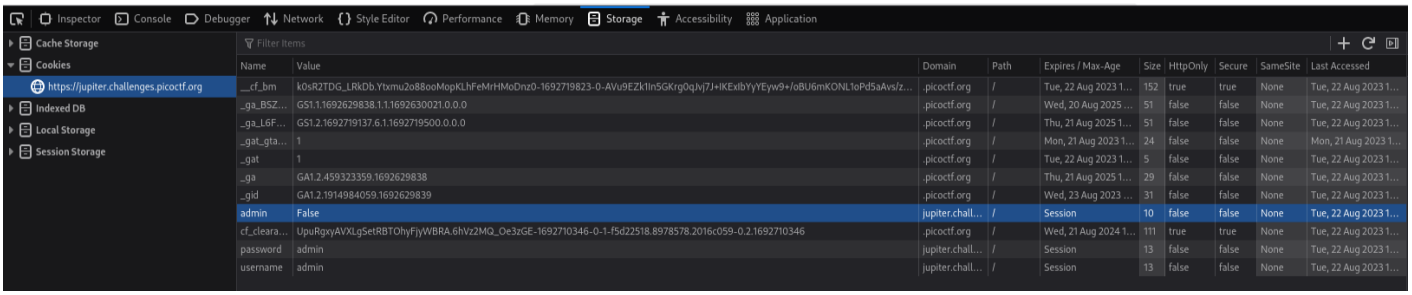


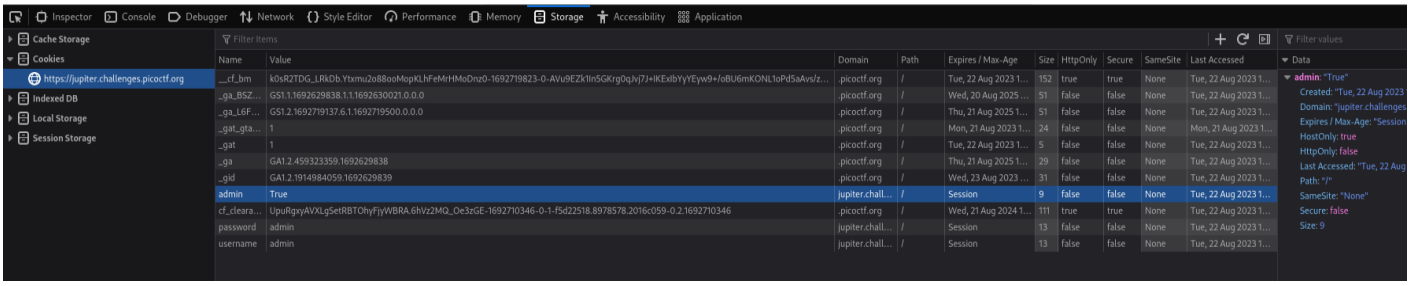
It gives a hint that it doesn't seem to check anyone's password, except for Joe's?

So I entered username as admin and password also as admin page directed to this page



Then moved I to the inspect element then in the storage tab I changed the cookie settings as follows and got the flag.





Factory Login

Home Sign Out

Flag:

picoCTF{th3_c0nsp1r4cy_l1v3s_6edb3f5f}

© PicoCTF 2019

10.Some Assembly Required 1

Some Assembly Required 1 | 70 points

Tags: picoCTF 2021 Web Exploitation

AUTHOR: SEARS SCHULZ

Description

http://mercury.picoctf.net:36152/index.html

Hints

(None)

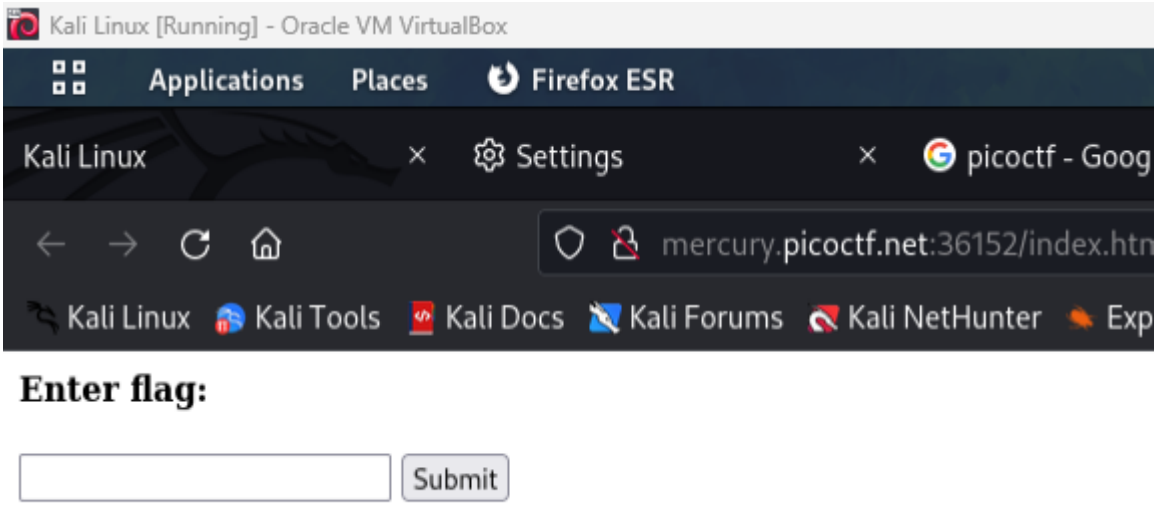
20,752 solves / 22,016 users attempted (94%)

62% Liked

picoCTF{FLAG}

Submit Flag

Once I click the link it directed to this page



I moved to the inspect element and found a js file.

```
50    }
51  })(_0x402c, 0x994c3);
52  let exports;
53  (async () => {
54    const _0x48c3be = _0x4e0e;
55    let _0x5f0229 = await fetch(_0x48c3be(0x1e9)),
56      _0x1d99e9 = await WebAssembly[_0x48c3be(0x1df)](
57        await _0x5f0229[_0x48c3be(0x1da)]()
58      ),
```

We must consider this also because title of the challenge also some assembly required

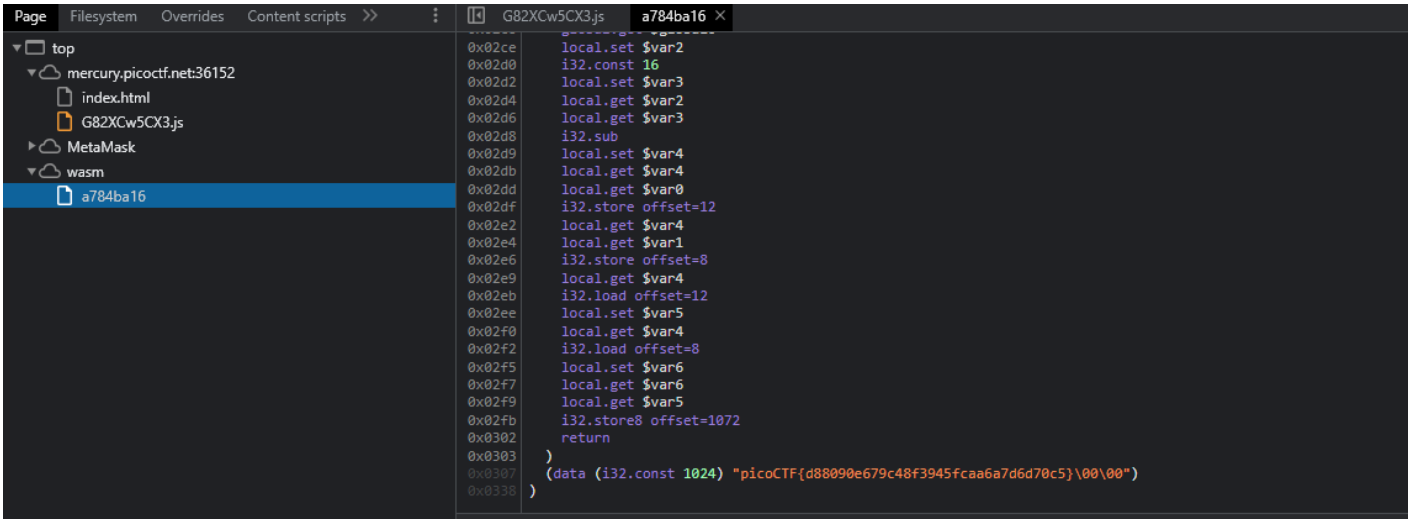
WebAssembly means that we are fetching something from some address that defined by this code.

await fetch(_0x48c3be(0x1e9))

and in sources there is a wasm file too.

wasm = WebAssembly is a binary instruction format for a stack-based virtual machine. Wasm is designed as a portable compilation target for programming languages, enabling deployment on the web for client and server applications.

In the wasm file I was able to get the flag



Terminal Help assembly.js - CTF - Visual Studi

login.js | assembly.js

assembly.js > _0x402c

```js
const _0x402c = [
  "value",
  "2wfTpTR",
  "instantiate",
  "275341bEPcme",
  "innerHTML",
  "1195047NznhZg",
  "1qfevql",
  "input",
  "1699808QuoWhA",
  "Correct!",
  "check_flag",
  "Incorrect!",
  "./JIFxzHyW8W",
  "23SMpAuA",
  "802698XOMSrr",
  "charCodeAt",
  "474547vVoGDO",
  "getElementById",
  "instance",
  "copy_char",
  "43591XxcWUl",
  "504454llVtzW",
  "arrayBuffer",
  "2NIQmVj",
  "result",
];
const _0x4e0e = function (_0x553839, _0x53c021) {
  _0x553839 = _0x553839 - 0x1d6;
  let _0x402c6f = _0x402c[_0x553839];
  return _0x402c6f;
};
(function (_0x76dd13, _0x3dfcae) {
  const _0x371ac6 = _0x4e0e;
  while (!![]) {
    try {
      const _0x478583 =
        -parseInt(_0x371ac6(0x1eb)) +
        parseInt(_0x371ac6(0x1ed)) +
        -parseInt(_0x371ac6(0x1db)) * -parseInt(_0x371ac6(0x1d9)) +
        -parseInt(_0x371ac6(0x1e2)) * -parseInt(_0x371ac6(0x1e3)) +
        -parseInt(_0x371ac6(0x1de)) * parseInt(_0x371ac6(0x1e0)) +
        parseInt(_0x371ac6(0x1d8)) * parseInt(_0x371ac6(0x1ea)) +
        -parseInt(_0x371ac6(0x1e5));
      if (_0x478583 === _0x3dfcae) break;
      else _0x76dd13["push"](_0x76dd13["shift"]());
```