

# WRITE UP JOINTS 2023

~ by Heker 1MISSU ~



**TEAMS :**

**IndianaJones**

**BayzLightyear**

**UkiyoAgusta**

# WEB

## Vision

### Vision

### 100

Jota visited a strange website that asked him to enter a secret code, can you help Jota to explore the website ?

Author: Jears #8964

34.101.234.148:8239

Submit

Pada chall ini kita diberikan sebuah website yang berisi form input. Yang dimana jika di view page source akan terlihat jika kita menginputkan secret code dengan “mantapujiwa” maka akan mengeluarkan pop up

```
<script>
  let popup = document.getElementById("popup");
  function inputCode() {
    let input= document.getElementById("userInput").value;
    let message = document.querySelector("#message")
    if(input == "mantapujiwa"){
      popup.classList.add("showPopup");
      message.innerHTML = "Your code is right!";
    }
    else{
      message.innerHTML = "Your code is wrong!";
    }
  }
</script>
```

pop up tidak berisi flag, namun terdapat link ke webpage selanjutnya, dan jika diperhatikan terdapat gambar yang di sembunyikan, cara untuk menampilkannya tinggal menghapus code visibility: hidden pada css sehingga didapatkan gambar flag di tampilan flagnya

**Flag: JCTF2023{s0\_e4sy\_w3b\_3xPl0itation}**



## Web of the Gods

# Web of the Gods

## 300

The power of a god... One could only dream it.

Author: Giga - Infinicus#6867

34.101.234.148:8069

► View Hint

Submit

pada chall ini terdapat sebuah website yang berisi flag. Objectivenya sebenarnya adalah untuk mengganti header terus menerus sesuai dengan yang diminta oleh web. Jadi berikut adalah command untuk mengganti header-header yang diperlukan dengan menggunakan curl untuk mendapatkan flagnya

```
indianajones@bukan-heker:~/Documents/CTF/picoctf/2022/web/Who_Are_You$ curl -s -H 'Accept-Language: el' --referer "https://www.jointsugm.id/" -H "DNT:1" -XGET "http://34.101.234.148:8069/Domain-of-Gods/secrypt.js"
```

flag berada pada file secrypt.js

```
lumn|addTableRow|addTableColumnGroup|addTableBody|printRandomLetters|ABCDEFGHIJKLMNOPQRSTUVWXYZ  
XYZabcdefghijklmnopqrstuvwxyz0123456789|printRandomNumbers|0123456789|cdnjs|addTableData|t4k  
Ar4pUt0_P0p0ruN64_p1R1T0P4R0|JCTF2023|cloudflare|com|ajax|libs|animejs|anime|min|js|addLink|  
addList|addListItem|addTable|printFlag|ReadTextFile'.split('|'),0,{}))
```

Flag: JCTF2023{t4kAr4pUt0\_P0p0ruN64\_p1R1T0P4R0}

## OSINT

whereIsThis

### whereIsThis 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



Tinggal cari Pentol Mbokdhe, lalu cari Indomaretnya di mana memakai layanan Maps. Ditemukan alamat Indomaret Jl. DR. Sardjito No.31, Terban, Kec. Gondokusuman, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55223 dengan Pluscode 69FC+8V Terban, Kota Yogyakarta, Daerah Istimewa Yogyakarta.

**Flag: JCTF2023{69FC+8V\_TERBAN}**

## MISC

### Mega SUS

Challenge

43 Solves

×

## Mega SUS

### 100

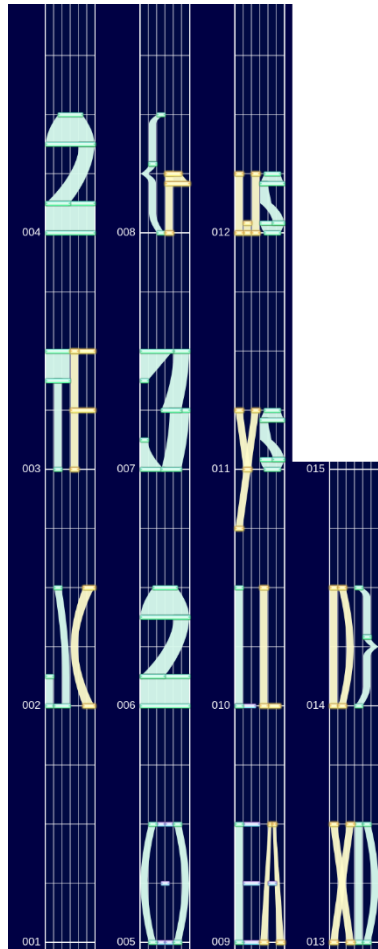
My friend who really like a rythm game send this file to me, claiming he got it from a rhytm game called Project Sekai. It's really *sus*.

Author: Arif ('saj#6550)

[https://drive.google.com/file/d/1zowdSmoGXeZns2J0l8lkRe/view?usp=share\\_link](https://drive.google.com/file/d/1zowdSmoGXeZns2J0l8lkRe/view?usp=share_link)

Submit

Untuk menyelesaikan chall ini, cukup dengan melakukan konversi dari file .SUS ke gambar. Kami menggunakan online converter yang dapat diakses di <https://sus2img.palettetool.com/>. Cukup dengan input file maka akan ditampilkan gambar yang akan membentuk flag yang dicari.



Flag: JCTF2023{rEALLYsusXDD}

# FORENSIC

## Dinosaur

### Dinosaur

100

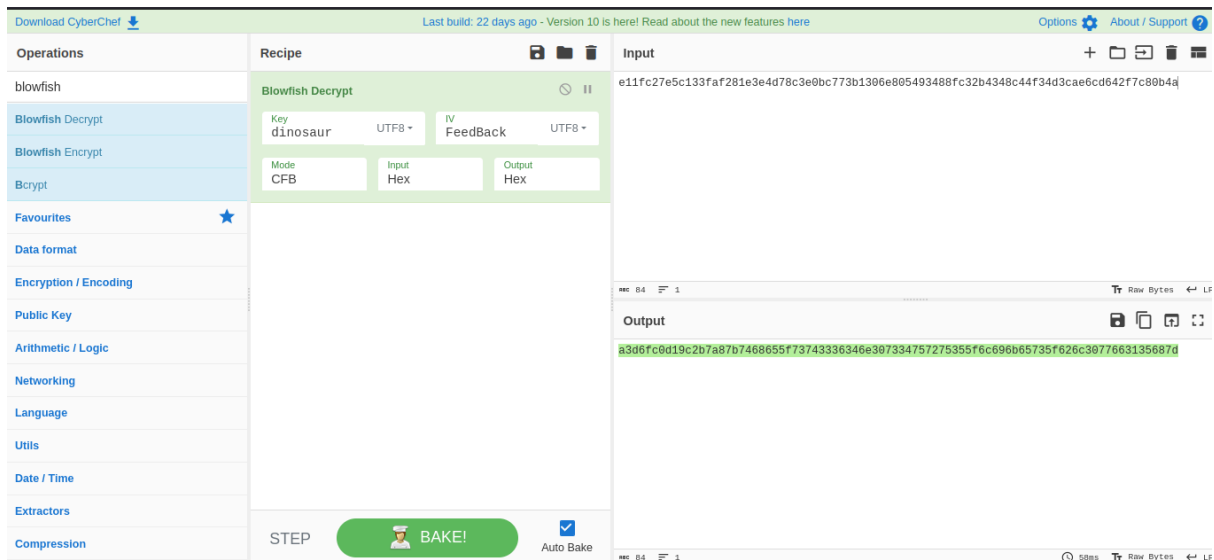
The stegosaurus is one of the few creatures that likes to eat blowfish. The key of its favorable taste to a blowfish is dinosaur. It initializes his day by using blowfish. Although it wasn't the best food of the prehistoric era, the stegosaurus always leaves a FeedBack which until now, is still a Cipher 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/1ymEPI2oZOLubN3VD8SKJHNfKkzhzhtiY/view?usp=share\\_link](https://drive.google.com/file/d/1ymEPI2oZOLubN3VD8SKJHNfKkzhzhtiY/view?usp=share_link)

Pada chall ini kami diberikan sebuah gambar jpg. Berdasarkan hint yang ada pada desc soal, kami hanya perlu mengekstrat data dalam gambar dengan steghide tanpa passphrase. Setelah di steghide, kami mendapatkan file txt yang berisi hex. Ternyata dalam hint pada soal sudah jelas apa” saja yang perlu kita lakukan, jadi kami menggunakan cyberchef untuk mendecrypt hex tadi dengan blowfish cipher, dan menggunakan key berupa “dinosaur”, IV berupa “FeedBack” dan mode CFB.



Selanjutnya tinggal di decode menjadi ascii text dan didapatkan flagnya

**Flag: JCTF2023{the\_st364n0s4uru5\_likes\_bl0wf15h}**

# CYRPTOGRAPHY

## Easy CBC

## 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)

```
# !pip install certifi==2021.10.8
# !pip install cffi==1.15.0
# !pip install cryptography==36.0.2
# !pip install Pillow==9.0.1
# !pip install wincertstore==0.2
import os
from cryptography.hazmat.primitives.ciphers import Cipher, algorithms,
modes
from cryptography.hazmat.backends import default_backend
import PIL.Image as Image

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')

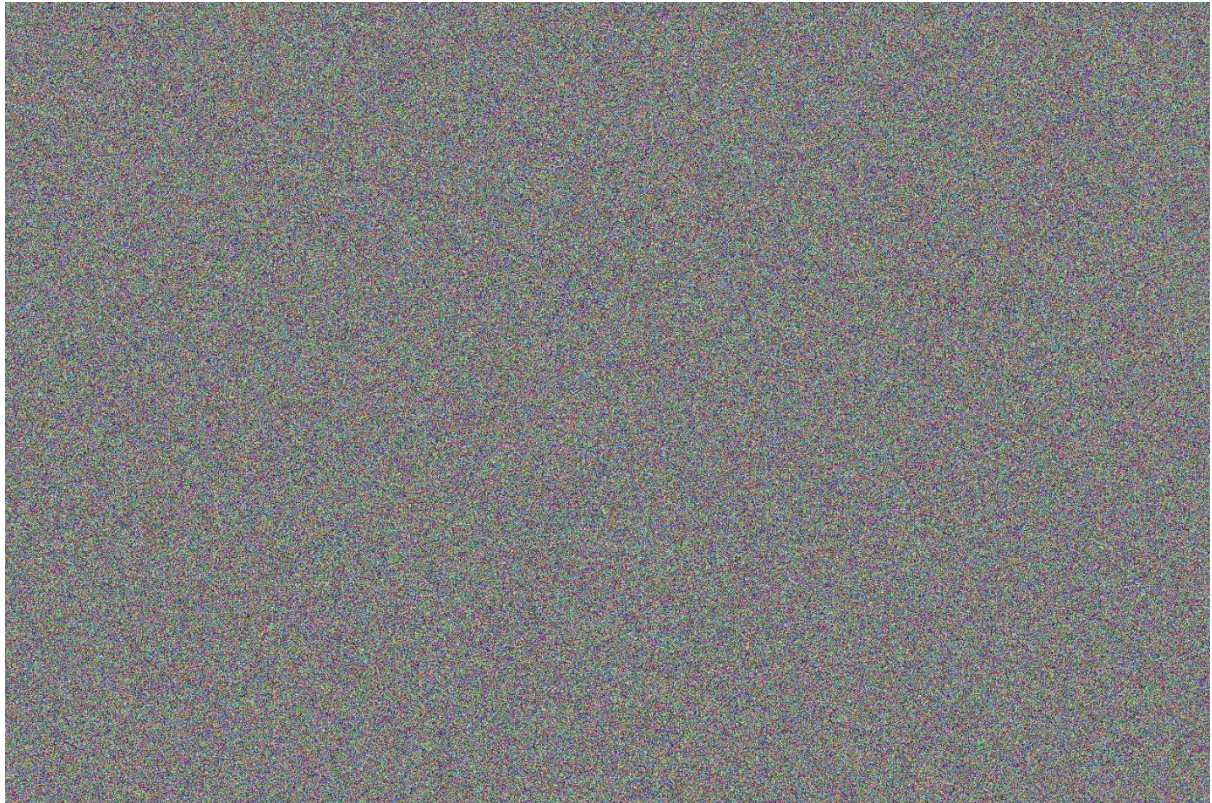
def main():
    key = b'JOINTSCTF2023'
    key = key.ljust(32, b'\x35')

    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()

```



Cara solvenya tinggal reverse kode Python yang di soalnya saja.

```
import os
from cryptography.hazmat.primitives.ciphers import Cipher, algorithms,
modes
from cryptography.hazmat.backends import default_backend
import PIL.Image as Image

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

    def decrypt(self, image):
        return self.decryptor.update(image)

    def finalize_decrypt(self):
        return self.decryptor.finalize()

def DecryptImage(encryption, image, output):
    output = output + '.jpg'
    with open(image, 'rb') as reader:
        with open(output, 'wb') as writer:
            image_data = reader.read()
            header, body = image_data[:54], image_data[54:]
            body = encryption.decrypt(body)
            encryption.finalize_decrypt()
            body = body.rstrip(b'\x35')
            writer.write(header + body)
            writer.close()
            reader.close()
```

```
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)
    DecryptImage(encryption=AesCbc, image='out.bmp', output='decrypted')

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



Flag: JCTF2023{n4rim0\_in9\_pAndum}