



2019杭电CTF HGAME Writeup

原创

可乐  于 2019-01-27 15:00:26 发布  2641  收藏 2

分类专栏: [CTFwrite](#) 文章标签: [2019杭电CTF HGAME Writeup CTF](#)

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_30464257/article/details/86663069

版权



[CTFwrite](#) 专栏收录该内容

22 篇文章 0 订阅

订阅专栏

前言

□

WEB

谁吃了我的flag

呜呜呜, Mki一起床发现写好的题目变成这样了, 是因为昨天没有好好关机吗T_T hint: 据当事人回忆, 那个夜晚他正在用vim编写题目页面, 似乎没有保存就关机睡觉去了, 现在就是后悔, 十分的后悔。

hint 是后来增加的 一开始做的时候就想到类vim文件泄露 可是当时没有

第二天看了出了hint 说是vim 再进行 /.index.html.swp

成功下载出来 得到flag

换头大作战

1. 默认是GET提交 改为POST提交

□

2. 设置X-Forwarded-For:127.0.0.1

□

3. 修改UA

□

4. 修改referer

□

very easy web

```
<?php
error_reporting(0);
include("flag.php");

if(strpos("vidar",$_GET['id'])!==FALSE)
    die("<p>干巴爹</p>");

$_GET['id'] = urldecode($_GET['id']);
if($_GET['id'] === "vidar")
{
    echo $flag;
}
highlight_file(__FILE__);
?>
```

```
$_GET['id'] = urldecode($_GET['id']);
```

对id进行url解码了一次

在数据传入php脚本进行数据处理时本身会被解码一次

这样就导致了二次编码注入

将vidar进行url编码两次提交

?id=%2576%2569%2564%2561%2572

can u find me?

在源代码看到f12.php

访问

□

在返回包看到密码,提交

□

访问iamflag.php

□

easy_php

□

正则可以用双写绕过

□

看到include_once函数 试一下读读flag.php源码

利用伪协议 php://filter

□

base64解码得到

□

php trick

misc

Hidden Image in LSB

用stegsolve进行分析得到flag

□

Broken Chest

是一个压缩包文件，解压发现文件错误

用winhex打开压缩包分析分析

发现文件头不是以 50 4B 03 04 开头

修改为50 4B 03 04 开头 即可 保存

解压发现要密码, 在注释里面看到密码 提交得到flag

参考:

<https://ctf-wiki.github.io/ctf-wiki/misc/archive/zip/>