




# WeChall CTF Writeup (三)

原创

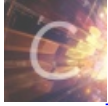
[lmn\\_](#)  于 2022-02-26 21:05:12 发布  1088  收藏

分类专栏: [CTF](#) 文章标签: [web安全](#) [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/weixin\\_43211186/article/details/123156011](https://blog.csdn.net/weixin_43211186/article/details/123156011)

版权



[CTF 专栏收录该内容](#)

15 篇文章 0 订阅

订阅专栏

## 文章目录

[0x11 2 Training: PHP LFI by Gizmore](#)

[0x12 2 PHP 0817 by Gizmore](#)

[0x13 2 Training: Crypto - Transposition I by Gizmore](#)

[0x14 2 Training: Crypto - Substitution I by Gizmore](#)

以下题目标题组成:

[Score] [Title] [Author]

[0x11 2 Training: PHP LFI by Gizmore](#)

## Training: PHP LFI (Exploit, PHP, Training)

### PHP - Local File Inclusion

Your mission is to exploit this code, which has obviously an LFI vulnerability:

#### GeSHi`ed PHP code

```
1 $filename = 'pages/'.(isset($_GET["file"])?$_GET["file"]:"welcome").'.html';
2 include $filename;
```

There is a lot of important stuff in ../solution.php, so please include and execute this file for us.

Here are a few examples of the script in action (in the box below):

[index.php?file=welcome](#)

[index.php?file=news](#)

[index.php?file=forums](#)

For debugging purposes, you may look at the whole source again, also as highlighted version.

### The vulnerable script in action (pages/welcome.html)

**Welcome** to my site!

Dude, you got hacked by ZeroCool :D Contact me...

Thanks go out to [minus](#) for his alpha testing, great thoughts and motivation!

© 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021 and 2022 by [Gizmore](#)

CSDN@lmm\_

题目意思:

您的任务是利用此代码, 该代码显然存在LFI漏洞:

#### GeSHi`ed PHP code

```
$filename = 'pages/'.(isset($_GET["file"])?$_GET["file"]:"welcome").'.html';
include $filename;
```

../solution.php 中有很多重要的东西, 所以请包含并执行此文件。

下面是一些正在运行的脚本示例 (在下面的框中):

[index.php?file=welcome](#)

[index.php?file=news](#)

[index.php?file=forums](#)

出于调试目的, 可以再次查看整个源代码, 也作为高亮版本。

高亮版本:

<https://www.wechall.net/challenge/training/php/lfi/up/index.php?highlight=christmas>

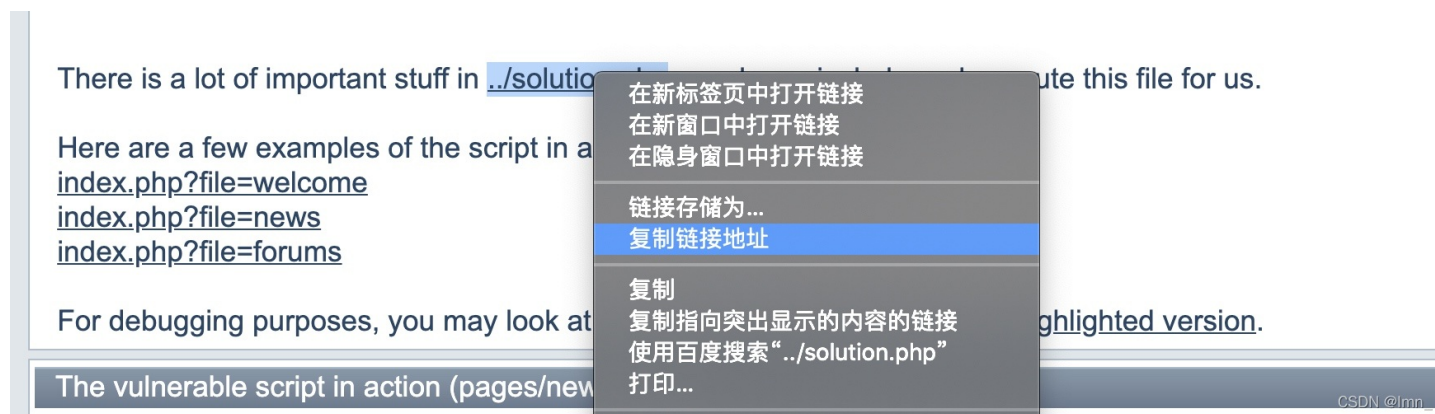
源代码:

<https://www.wechall.net/challenge/training/php/lfi/up/index.php?show=source>

LFI漏洞又称为本地文件包含漏洞

<https://www.wechall.net/challenge/training/php/lfi/up/index.php?file=news>

isset() 函数用于检测变量是否已设置并且非 NULL



看到地址

<https://www.wechall.net/challenge/training/php/lfi/solution.php>

判断出solution.php前应该是.../.../

因为后面有一个.html，我们可以用%00进行截断

最终结果：

<http://www.wechall.net/challenge/training/php/lfi/up/index.php?file=.../.../solution.php%00>

**0x12 2 PHP 0817 by Gizmore**

## PHP 0817 (PHP, Exploit)

### PHP-0817

I have written another include system for my dynamic webpages, but it seems to be vulnerable to LFI. Here is the code:

#### GeSHi`ed PHP code

```
1 <?php
2 if (isset($_GET['which']))
3 {
4     $which = $_GET['which'];
5     switch ($which)
6     {
7     case 0:
8     case 1:
9     case 2:
10         require_once $which.'.php';
11         break;
12     default:
13         echo GWF_HTML::error('PHP-0817', 'Hacker NoNoNo!', false);
14         break;
15     }
16 }
17 ?>
```

Your mission is to include [solution.php](#). Here is the script in action: [News](#), [Forum](#), [Guestbook](#).

Good Luck!

题目意思:

我为我的动态网页编写了另一个包含系统，但它似乎容易受到 LFI 的攻击。

这是代码:

```
<?php
if (isset($_GET['which']))
{
    $which = $_GET['which'];
    switch ($which)
    {
    case 0:
    case 1:
    case 2:
        require_once $which.'.php';
        break;
    default:
        echo GWF_HTML::error('PHP-0817', 'Hacker NoNoNo!', false);
        break;
    }
}
?>
```

您的任务是包含solution.php。

以下是正在运行的脚本：

News

Forum

Guestbook

目标是包含solution.php文件

<https://www.wechall.net/challenge/php0817/solution.php>

change language: [Chinese (Simplified)]

[Submit a Pull Request](#) [Report a Bug](#)

## switch

---

(PHP 4, PHP 5, PHP 7, PHP 8)

switch 语句类似于具有同一个表达式的一系列 if 语句。很多场合下需要把同一个变量（或表达式）与很多不同的值比较，并根据它等于哪个值来执行不同的代码。这正是 switch 语句的用途。

**注意：**注意和其它语言不同，[continue](#) 语句作用到 switch 上的作用类似于 break。如果在循环中有一个 switch 并希望 continue 到外层循环中的下一轮循环，用 continue 2。

CSDN @lmm\_

当非数字开头的字符串与数字 0 进行比较时，结果返回true

<http://www.wechall.net/challenge/php0817/index.php?which=solution>

## 0x13 2 Training: Crypto - Transposition I by Gizmore

? | score: 2 | [2.41](#) [3.30](#) [4.07](#) | Solved By [3730 People](#) | 83696 views | since Nov 27, 2010 - 21:25:40

### Training: Crypto - Transposition I (Crypto, Training)

#### Crypto - Transposition I

It seems that the simple substitution ciphers are too easy for you.  
From my own experience I can tell that [transposition ciphers](#) are more difficult to attack.  
However, in this training challenge you should have not much problems to reveal the plaintext.

oWdnreuf.IY uoc nar ae dht eemssga eaw yebttrew eh nht eelttre sra enic roertco drre . lhtni koy uowlu dilekt oes eoypur sawsro don:wn lesoihofpc.p

#### Your solution for Training: Crypto - Transposition I

Answer

CSDN @lmm\_

题目意思：

看起来简单的替换密码对你来说太容易了。

根据我自己的经验，我可以看出转置密码更难攻击。

然而，在这个训练挑战中，你应该没有太多问题来揭示明文。

密文：

oWdnreuf.IY uoc nar ae dht eemssga eaw yebttrew eh nht eelttre sra enic roertco drre . lhtni koy uowlu dilekt oes eoypur sawsro don:wn lesoihofpc.p

可以很快发现每两个字母进行换位，C语言实现一下

```

int main()
{
    char arr[] = "oWdnreuf.lY uoc nar ae dht eemssga eaw yebttrew eh nht eelttre sra enic roertco drre . Ihtni k
oy uowlu dilekt oes eoyrup sawsro don:wn lesoihofpc.p";
    int sz = sizeof(arr)/sizeof(arr[0]);
    //printf("%d\n",sz);
    char tmp = '0';
    int i = 0;
    while(i<sz-1)
    {
        tmp = arr[i];
        arr[i] = arr[i+1];
        arr[i+1] = tmp;
        i = i+2;
    }
    for (i = 0; i < sz; i++) {
        printf("%c",arr[i]);
    }
    return 0;
}

```

```

1910 int main()
1911 {
1912     char arr[] = "oWdnreuf.lY uoc nar ae dht eemssga eaw yebttrew eh nht eelttre sra enic roertco drre . Ihtni koy uowlu dilekt oes eoyrup sawsro don:wn lesoihofpc.p";
1913     int sz = sizeof(arr)/sizeof(arr[0]);
1914     //printf("%d\n",sz);
1915     char tmp = '0';
1916     int i = 0;
1917     while(i<sz-1)
1918     {
1919         tmp = arr[i];
1920         arr[i] = arr[i+1];
1921         arr[i+1] = tmp;
1922         i = i+2;
1923     }
1924     for (i = 0; i < sz; i++) {
1925         printf("%c",arr[i]);
1926     }
1927     return 0;
1928 }

```

Run: untitled1 x

```

/Users/zhaoyifan/CLionProjects/untitled1/cmake-build-debug/untitled1
Wonderful. You can read the message way better when the letters are in correct order. I think you would like to see your password now: neloshifocpp.0
Process finished with exit code 0

```

明文:

Wonderful. You can read the message way better when the letters are in correct order. I think you would like to see your password now: neloshifocpp.

## 0x14 2 Training: Crypto - Substitution I by Gizmore

### Training: Crypto - Substitution I (Crypto, Training)

#### Crypto - Simple Substitution I

Oh dear, I guess you have cracked the two caesar cryptos...  
This one is more difficult. Although a simple substitution is easily cracked...  
Again the characters are limited to A-Z... But I think I can come up with a 256 version again.

Enjoy!

QH APD RLSCMPAH MXU HXF IRW TDRU APCB SH KTCDWU C RS CSNTDBBDU VDTH GDLL UXWD HXFT BXLFACXW ODH CB DNDBMWXLBSSL APCB LCAALD  
IPRLLDWMD GRB WXAAXX PRU GRB CA

#### Your solution for Training: Crypto - Substitution I

Answer

CSDN @imn\_

题目意思：

哦，亲爱的，我猜你已经破解了两个凯撒密码...

这个更难。虽然简单的替换很容易破解...

再次将字符限制为A-Z范围内.但我想我可以再想出256个版本。

密文：

UQ ZOT KXPJROZQ RND QNC BKE MTKD ZOJS PQ HMJTED J KP JPGMTSSTD LTMQ ITXX DNET QNCM SNXCZJNE ATQ  
JS TGTSRENXSPPX ZOJS XJZZXT BOKXXTERT IKS ENZ ZNN OKMD IKS JZ

# quipqiup beta3

*quipqiup* is a fast and automated cryptogram solver by [Edwin Olson](#). It can solve simple substitution ciphers often found in newspapers, including puzzles like cryptoquips (in which word boundaries are preserved) and patristocrats (in which word boundaries are not preserved).

Puzzle:

UQ ZOT KXPJROZQ RND QNC BKE MTKD ZOJS PQ HMJTED J KP JPGMTSSTD LTMQ ITXX DNET QNCM SNXCZJNE ATQ JS TGTSRENXSPPX ZOJS XJZZXT BOKXXTERT IKS ENZ ZNN OKMD IKS JZ

Clues: For example G=R QVW=THE

Solve

⊗ automatically selected statistics mode; you can override by using the drop down menu next to the solve button.

0	-1.877	BY THE ALMIGHTY GOD YOU CAN READ THIS MY FRIEND I AM IMPRESSED VERY WELL DONE YOUR SOLUTION KEY IS EPESGNOLSMML THIS LITTLE CHALLENGE WAS NOT TOO HARD WAS IT
1	-3.666	GB THE ULAIKHTB KOS BOY CUM REUS THIN AB PRIEMS I UA IADRENNES VERB FELL SOME BOYR NOLYTIOM WEB IN EDENKMOLNAAL THIN LITTLE CHULLENKE FUN MOT TOO HURS FUN IT
2	-3.667	PM THE ALBIDHTM DOS MOK WAR UEAS THIN BM QUIERS I AB IBGUENNES YEUM CELL SORE MOKU NOLKTIOR JEM IN EGENDROLNBBL THIN LITTLE WHALLERDE CAN ROT TOO HAUS CAN IT

CSDN @imn\_

quipqiup可以破解简单的替换

明文：

BY THE ALMIGHTY GOD YOU CAN READ THIS MY FRIEND I AM IMPRESSED VERY WELL DONE YOUR SOLUTION KEY IS  
EPESGNOLSMML THIS LITTLE CHALLENGE WAS NOT TOO HARD WAS IT